

INTERNATIONAL STANDARD

NORME INTERNATIONALE



Function blocks (FB) for process control and electronic device description language (EDDL) –

Part 2: Specification of FB concept

Blocs fonctionnels (FB) pour les procédés industriels et langage de description électronique de produit (EDDL) –

Partie 2: Spécification du concept de FB

IECNORM.COM : Click to view the full PDF of IEC 61804-2:2018



THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2018 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
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 - webstore.iec.ch/advsearchform

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 21 000 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

67 000 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: sales@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 - webstore.iec.ch/advsearchform

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 21 000 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

67 000 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: sales@iec.ch.

INTERNATIONAL STANDARD

NORME INTERNATIONALE



**Function blocks (FB) for process control and electronic device description
language (EDDL) –
Part 2: Specification of FB concept**

**Blocs fonctionnels (FB) pour les procédés industriels et langage de description
electronique de produit (EDDL) –
Partie 2: Spécification du concept de FB**

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

COMMISSION
ELECTROTECHNIQUE
INTERNATIONALE

ICS 25.040.40; 35.240.50

ISBN 978-2-8322-5128-7

**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.....	5
INTRODUCTION.....	7
1 Scope.....	9
2 Normative references	9
3 Terms, definitions, abbreviated terms and conventions	9
3.1 Terms and definitions.....	9
3.2 Abbreviated terms.....	18
3.3 Conventions for lexical structures	19
4 General function block (FB) definition and EDD model.....	20
4.1 Device structure (device model).....	20
4.1.1 Device model description	20
4.1.2 FB type	23
4.1.3 FB execution.....	25
4.1.4 Reference between IEC 61499 and IEC 61804 models	26
4.1.5 UML specification of the device model	26
4.1.6 Classification of the algorithms	28
4.1.7 Algorithm description	29
4.1.8 Input and output variables and parameter definition	29
4.1.9 Choice of variables and parameters	30
4.1.10 Mode, Status and Diagnosis	30
4.2 Block combinations	30
4.2.1 Measurement channel.....	30
4.2.2 Actuation channel	31
4.2.3 Application.....	32
4.3 EDD and EDDL model.....	32
4.3.1 Overview of EDD and EDDL	32
4.3.2 EDD architecture	33
4.3.3 Concepts of EDD	33
4.3.4 Principles of the EDD development process	33
4.3.5 Interrelations between the lexical structure and formal definitions	34
4.3.6 Builtins	35
4.3.7 Profiles	35
5 Detailed block definition	35
5.1 General.....	35
5.2 Application FBs.....	35
5.2.1 Analog Input FB	35
5.2.2 Analog Output FB	37
5.2.3 Discrete Input FB	38
5.2.4 On/Off Actuation FB Discrete Output FB	40
5.2.5 Calculation FB	41
5.2.6 Control FB	42
5.3 Component FBs	44
5.4 Technology Block.....	44
5.4.1 Temperature Technology Block.....	44
5.4.2 Pressure Technology Block.....	47
5.4.3 Modulating Actuation Technology Block	49

5.4.4	On/Off Actuation Technology Block	51
5.5	Device (Resource) Block	54
5.5.1	Identification	54
5.5.2	Device state	54
5.5.3	Message	56
5.5.4	Initialisation	56
5.6	Algorithms common to all blocks	56
5.6.1	Data Input/Data Output status	56
5.6.2	Validity	57
5.6.3	Restart Initialisation	57
5.6.4	Fail-safe	57
5.6.5	Remote Cascade Initialisation	58
6	FB Environment	58
7	Mapping to System Management	58
8	Mapping to Communication	59
Annex A (informative)	Parameter description	62
Annex B (informative)	Compatibility levels	68
B.1	General	68
B.2	Compatibility	69
B.3	Incompatibility	69
B.4	Coexistence	70
B.5	Interconnectability	70
B.6	Interworkability	70
B.7	Interoperability	70
B.8	Interchangeability	71
Annex C (informative)	Proxy concept and its use in FB applications	72
C.1	General proxy concept	72
C.2	Use of the proxy concept in FB applications	73
Bibliography		75
Figure 1	– Position of IEC 61804-2 related to other standards and products	7
Figure 2	– FB structure is derived out of the process (P&ID view)	20
Figure 3	– FB structure may be distributed between devices (according to IEC 61499-1)	21
Figure 4	– IEC 61804 FBs can be implemented in different devices	22
Figure 5	– General components of devices	22
Figure 6	– Block types of IEC 61804 (all parts)	23
Figure 7	– IEC 61804 block overview (graphical representation not normative)	24
Figure 8	– UML class diagram of the device model	27
Figure 9	– Measurement process signal flow	31
Figure 10	– Actuation process signal flow	31
Figure 11	– Application process signal flow	32
Figure 12	– EDD generation process	34
Figure 13	– Analog Input FB	36
Figure 14	– Analog Output FB	37
Figure 15	– Discrete input FB	39

Figure 16 – Discrete Output FB.....	40
Figure 17 – Calculation FB.....	42
Figure 18 – Control FB.....	43
Figure 19 – Temperature Technology Block.....	44
Figure 20 – Pressure Technology Block.....	48
Figure 21 – Modulating Actuation Technology Block.....	50
Figure 22 – On/Off Actuation Technology Block.....	52
Figure 23 – Harel state chart.....	55
Figure 24 – Application structure of ISO OSI Reference Model.....	59
Figure 25 – Client/Server relationship in terms of OSI Reference Model.....	60
Figure 26 – Mapping of IEC 61804 FBs to APOs.....	60
Figure B.1 – Levels of functional device compatibility.....	68
Figure C.1 – Proxy model class diagram.....	72
Figure C.2 – Proxy integration in DCS.....	73
Table 1 – Field attribute descriptions.....	19
Table 2 – Equivalences between IEC 61804 and IEC 61499 model elements.....	26
Table 3 – Variables and parameter description template.....	29
Table 4 – Example of temperature sensors of Sensor Type.....	46
Table 5 – Device status state table.....	54
Table 6 – Device status transition table.....	56
Table A.1 – Parameter description.....	62
Table B.1 – Functionality features.....	69

IECNORM.COM : Click to view the full PDF of IEC 61804-2:2018

INTERNATIONAL ELECTROTECHNICAL COMMISSION

**FUNCTION BLOCKS (FB) FOR PROCESS CONTROL AND
ELECTRONIC DEVICE DESCRIPTION LANGUAGE (EDDL) –****Part 2: Specification of FB concept**

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 61804-2 has been prepared by subcommittee 65E: Devices and integration in enterprise systems, of IEC technical committee 65: Industrial-process measurement, control and automation.

This third edition cancels and replaces the second edition published in 2006 and integrates parts of IEC 61804-1 which was withdrawn in January 2013. This edition constitutes a technical revision.

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

- a) added command communication mapping in Clause 8;
- b) moved and reword compatibility level definition from IEC 62804-1 to new Annex B and terms and definitions;

c) added proxy concept in new Annex C.

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

FDIS	Report on voting
65E/567/FDIS	65E/576/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 61804 series, published under the general title *Function blocks (FB) for process control and electronic device description language (EDDL)*, can be found on the IEC website.

Future standards in this series will carry the new general title as cited above. Titles of existing standards in this series will be updated at the time of the next edition.

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

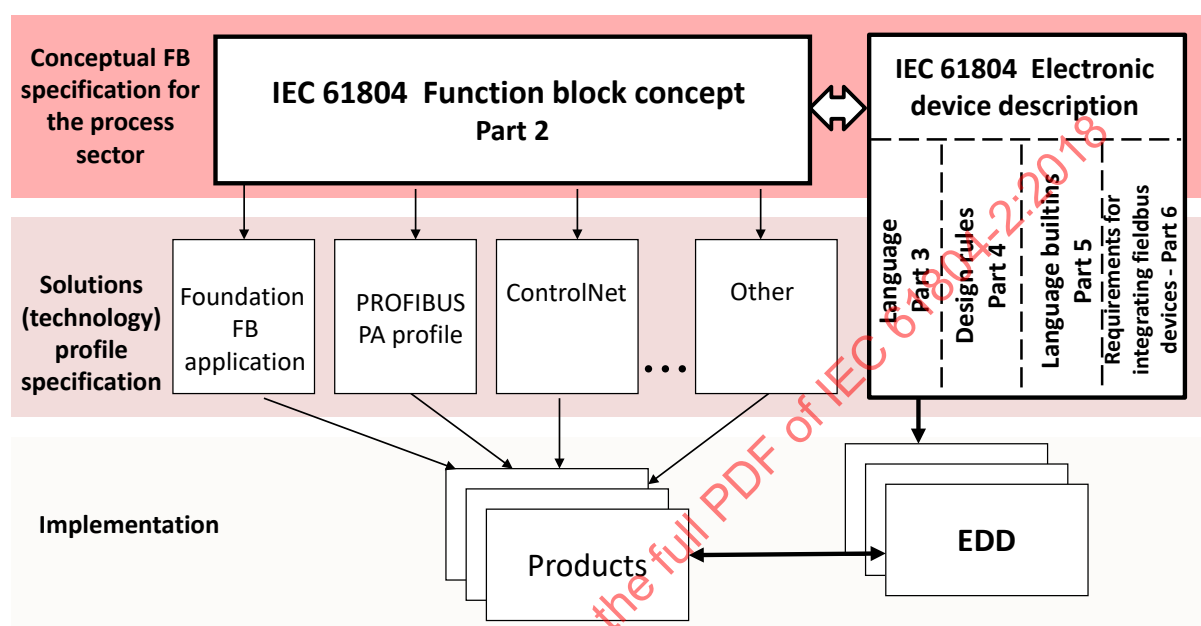
- reconfirmed,
- withdrawn,
- replaced by a revised edition, or
- amended.

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.

INTRODUCTION

This part of IEC 61804 provides a conceptual function block (FB) specification, which can be mapped to specific communication systems and their accompanying definitions by industrial groups.

The EDDL fills the gap between the conceptual FB specification of this document and a product implementation. Figure 1 shows these aspects.



IEC

Figure 1 – Position of IEC 61804-2 related to other standards and products

The International Electrotechnical Commission (IEC) draws attention to the fact that it is claimed that compliance with this document may involve the use of patents

U.S. Patent No. 5,333,114
 U.S. Patent No. 5,485,400
 U.S. Patent No. 5,825,664
 U.S. Patent No. 5,909,368
 U.S. Patent Pending No. 08/916,178
 Australian Patent No. 638507
 Canadian Patent No. 2,066,743
 European Patent No. 0495001
 Validated in:
 UK – Patent No. 0495001
 France – Patent No. 0495001
 Germany – Patent No. 69032954.7
 Netherlands – Patent No. 0495001
 Japan – Patent No. 3137643

IEC take no position concerning the evidence, validity and scope of this patent right. The holder of this patent right has assured the IEC that he is willing to negotiate licenses under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statement of the holder of this patent right is registered with IEC. Information may be obtained from:

FieldComm Group Inc.,
9430 Research Boulevard, Suite 1-120,
Austin, Texas, USA 78759,
Attention: President.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights other than those identified above. IEC shall not be held responsible for identifying any or all such patent rights.

ISO (www.iso.org/patents) and IEC (<http://patents.iec.ch>) maintain on-line data bases of patents relevant to their standards. Users are encouraged to consult the data bases for the most up to date information concerning patents.

The IEC 61804 series has the general title "Function blocks (FB) for process control and electronic device description language (EDDL)" and consists of the following parts:

Part 2: FB concept

Part 3: Electronic device description language (EDDL)

Part 4: EDD design rules

Part 5: EDDL Builtin library

Part 6: Meeting the requirements for integrating fieldbus devices in engineering tools for field devices

IECNORM.COM : Click to view the full PDF of IEC 61804-2:2018

FUNCTION BLOCKS (FB) FOR PROCESS CONTROL AND ELECTRONIC DEVICE DESCRIPTION LANGUAGE (EDDL) –

Part 2: Specification of FB concept

1 Scope

This part of IEC 61804 is applicable to function blocks (FB) for process control.

This document specifies FB by using the result of a harmonization work as regards several elements.

- a) The device model which defines the components of an IEC 61804-2 conformant device.
- b) Conceptual specifications of FBs for measurement, actuation and processing. This includes general rules for the essential features to support control whilst avoiding details which stop innovation as well as specialization for different industrial sectors.
- c) The electronic device description (EDD) technology, which enables the integration of real product details using the tools of the engineering life cycle.

The standardization work for FB was carried out by harmonizing the description of concepts of existing technologies. It results in an abstract level that allowed the definition of the common features in a unique way. This abstract vision is called here the "conceptual FB specification" and is mapped to specific communication systems and their accompanying definitions by the industrial groups.

NOTE This document can be mapped to ISO 15745-1.

There are solutions on the market today, which fulfil the requirements of this document and show how the conceptual specification is implemented in a given technology. New technologies will need to find equivalent solutions (see Figure 4).

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 61158 (all parts), *Industrial communication networks – Fieldbus specifications*

IEC 61499-1:2012, *Function blocks – Part 1: Architecture*

ISO/IEC 7498-1, *Information technology – Open Systems Interconnection – Basic Reference Model: The Basic Model*

3 Terms, definitions, abbreviated terms and conventions

3.1 Terms and definitions

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

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

- IEC Electropedia: available at <http://www.electropedia.org/>
- ISO Online browsing platform: available at <http://www.iso.org/obp>

3.1.1

performance

quantitative or qualitative level of a property at any point in time considered

[SOURCE: ISO 15686-1:2011, 3.15, modified – The second term "performance in use" has been deleted. In the definition, the words "quantitative or" have been added and the word "critical" has been deleted.]

3.1.2

semantics

relationships between the symbolic elements and their meanings, interpretation and use

[SOURCE: IEC 61131-3:2013, 3.85, modified – The words "of a programming language" have been deleted.]

3.1.3

algorithm

finite set of well-defined rules for the solution of a problem in a finite number of *operations*

3.1.4

application

software functional unit that is specific to the solution of a problem in industrial-process measurement and control

Note 1 to entry: An application may be distributed among *resources*, and may communicate with other applications.

3.1.5

application function block

application FB

FB which has no input or output to the process

3.1.6

attribute

property or characteristic of an entity, for instance, the version identifier of an FB type specification

Note 1 to entry: The formal description of attributes is part of the solution profiles to get domain specific interoperability. IEC 61804 (all parts) defines the general rules to define the attributes and specifies the EDDL to describe attributes, which may be described in solution profiles.

[SOURCE: IEC 61499-1:2012, 3.6, modified – A note to entry has been added.]

3.1.7

Builtin

predefined subroutine for communication and display executed by the EDD application

3.1.8

coexistence

ability of two or more devices to operate independently of one another in the same network respecting the common rules for sharing the same medium

3.1.9

compatibility

ability of a device to provide the set of functions and data required by an application for a specific role in the physical process

Note 1 to entry: Function comprises application and communication functions including the dynamic behavior.

Note 2 to entry: Data comprises communication frame format and order as well as data type definitions up to semantical description of the functions.

3.1.10

component function block

component FB

FB instance which is used in the specification of an algorithm of a composite FB type

Note 1 to entry: A component FB can be an FB or composite FB type.

3.1.11

composite FB type

FB type whose algorithms are expressed entirely in terms of interconnected component FBs and variables

[SOURCE: IEC 61499-1:2012, 3.16, modified – The words "and the control of their execution" and "event" have been deleted.]

3.1.12

configuration

<of a system or device> selecting functional units, assigning their locations and defining their interconnections

[SOURCE: IEC 61499-1:2012, 3.18]

3.1.13

data

representation of facts, concepts or instructions in a formalized manner suitable for communication, interpretation or processing by human beings or by automatic means

[SOURCE: IEC 61499-1:2012, 3.23, modified – The definition has been rephrased.]

3.1.14

data connection

association established between functional units for conveyance of data

[SOURCE: IEC 61499-1:2012, 3.24, modified – The words "two function blocks" have been replaced by "functional units".]

3.1.15

data input

interface of an FB which receives data from a data connection

[SOURCE: IEC 61499-1:2012, 3.25]

3.1.16

data output

interface of an FB which supplies data to a data connection

[SOURCE: IEC 61499-1:2012, 3.26]

3.1.17

data type

set of values together with a set of permitted operations

[SOURCE: IEC 61499-1:2012, 3.27]

3.1.18

device

independent physical entity capable of performing one or more specified functions in a particular context and delimited by its interfaces

[SOURCE: IEC 61499-1:2012, 3.29, modified – The note to entry has been deleted.]

3.1.19

Device Block

FB, which has no input and no output

3.1.20

device management application

application whose primary function is the management of multiple resources within a device

[SOURCE: IEC 61499-1:2012, 3.30]

3.1.21

EDD application

program using the EDD, or any translated form, which offers functionality such as communication representation, data representation, graphical representation, etc.

3.1.22

EDDL processor

processor or program, which translates the EDD into an executable form that can be processed by an EDD application

3.1.23

EDDL profile

selection of the supported elements of the EDDL lexical structure including the syntax definitions for a number of specific consortia

3.1.24

electronic device description language

EDDL

methodology for describing parameter(s) of a automation system component

3.1.25

electronic device description

EDD

data collection containing the device parameter(s), their dependencies, their graphical representation and a description of the data sets which are transferred.

Note 1 to entry: The electronic device description is created using the electronic device description language (EDDL).

3.1.26

electronic device description source

EDDS

ASCII file containing a specific device description

3.1.27

electronic device description technology

EDDT

technology which includes the EDD development process, the EDD usage and the involved tool chain

3.1.28**electronic device description language compiler**

tool which translates the EDD source in an internal format that is used by the EDD interpreter

3.1.29**electronic device description interpreter****EDDI**

tool which uses the EDD source or an internal format that is given by the EDDL compiler to provide the EDD information to the EDD user

3.1.30**entity**

particular thing, such as a person, place, process, object, concept, association, or event

[SOURCE: IEC 61499-1:2012, 3.31]

3.1.31**event**

instantaneous occurrence that is significant to scheduling the execution of an algorithm

Note 1 to entry: The execution of an algorithm may make use of variables associated with an event.

[SOURCE: IEC 61499-1:2012, 3.32]

3.1.32**exception**

event that causes suspension of normal execution

[SOURCE: IEC 61499-1:2012, 3.36]

3.1.33**function**

intended purpose of an entity or its characteristic action

[SOURCE: IEC 61499-1:2012, 3.44, modified – The word "specific" has been replaced by "intended".]

3.1.34**functional unit**

entity of hardware or software, or both, capable of accomplishing a specified purpose

[SOURCE: IEC 61499-1:2012, 3.48]

3.1.35**function block****function block instance**

software functional unit comprising an individual, named copy of a data structure and associated operations specified by a corresponding FB type

Note 1 to entry: Typical operations of an FB include modification of the values of the data in its associated data structure.

[SOURCE: IEC 61499-1:2012, 3.45, modified – The definition has been rephrased and the second note to entry has been deleted.]

3.1.36

function block diagram

FBD

network in which the nodes are function block instances, variables, literals, and events

Note 1 to entry: This is not the same as the function block diagram defined in IEC 61131-3.

3.1.37

hardware

physical equipment, as opposed to programs, procedures, rules and associated documentation

[SOURCE: IEC 61499-1:2012, 3.49]

3.1.38

implementation

development phase in which the hardware and software of a system become operational

[SOURCE: IEC 61499-1:2012, 3.51]

3.1.39

incompatibility

inability of devices to provide the functions and performance required by their role in a distributed application

3.1.40

input variable

variable whose value is supplied by a data input, and which may be used in one or more operations of an FB

Note 1 to entry: An input parameter of an FB, as defined in IEC 61131-3, is an input variable.

[SOURCE: IEC 61499-1:2012, 3.54]

3.1.41

instance

functional unit comprising an individual, named entity with the attributes of a defined type

[SOURCE: IEC 61499-1:2012, 3.55]

3.1.42

instance name

identifier associated with and designating an instance

[SOURCE: IEC 61499-1:2012, 3.56]

3.1.43

instantiation

creation of an instance of a specified type

[SOURCE: IEC 61499-1:2012, 3.57]

3.1.44

interchangeability

ability of one device to fulfil exactly the same role of another device in the physical process and distributed application as required by the process design

3.1.45**interconnectivity**

ability of two or more devices to operate with one another using the same communications protocols and communication interface

3.1.46**interface**

shared boundary between two functional units, defined by functional characteristics, signal characteristics, or other characteristics as appropriate

[SOURCE: IEC 60050-351:2013, 351-42-25, modified – The word "shared" has been added to the definition, and the notes to entry have been deleted.]

3.1.47**internal variable**

variable whose value is used or modified by one or more operations of a FB but is not supplied by a data input or to a data output

[SOURCE: IEC 61499-1:2012, 3.60]

3.1.48**interoperability**

ability of two or more devices to work together in one or more distributed applications with the shared knowledge of the data types and the semantics of the data transmitted

3.1.49**interworkability**

ability of two or more devices to support the transfer of data between devices with the shared knowledge of the data types of the data transmitted

Note 1 to entry: The content of the data is unknown, e.g. a text string that can be displayed but not interpreted by the receiver device itself, or the content is a value but the unit is unknown.

3.1.50**invocation**

process of initiating the execution of the sequence of operations specified in an algorithm

[SOURCE: IEC 61499-1:2012, 3.61]

3.1.51**management function block**

FB whose primary function is the management of applications within a resource

[SOURCE: IEC 61499-1:2012, 3.64]

3.1.52**management resource**

resource whose primary function is the management of other resources

[SOURCE: IEC 61499-1:2012, 3.65]

3.1.53**mapping**

set of values having defined correspondence with the quantities or values of another set

[SOURCE: IEC 61499-1:2012, 3.66, modified – The words "features or attributes" have been replaced by "values", and "members" by quantities or values".]

3.1.54

model

representation of a real world process, device, or concept

3.1.55

operation

well-defined action that, when applied to any permissible combination of known entities, produces a new entity

[SOURCE: IEC 61499-1:2012, 3.73]

3.1.56

output variable

variable whose value is established by one or more operations of a FB, and is supplied to a data output

Note 1 to entry: An output parameter of an FB, as defined in IEC 1131-3, is an output variable.

[SOURCE: IEC 61499-1:2012, 3.74]

3.1.57

parameter

variable that is given a constant value for a specified application and that may denote the application

[SOURCE: IEC 61499-1:2012, 3.75]

3.1.58

preprocessor

part of the runtime environment that transforms the information given in an EDD

3.1.59

preprocessor directive

description of conditions for filtering the EDD code before compilation or interpretation

Note 1 to entry: For example a preprocessor directive providing the facility to define names for constants or to write macros to make code easier to read.

3.1.60

proxy

computer process that relays a protocol between client and server computer systems, by appearing to the client to be the server and appearing to the server to be the client

[SOURCE: RFC 2828:2000]

3.1.61

resource

functional unit contained within a device which has independent control of its operation, and which provides various services to applications, including the scheduling and execution of algorithms

Note 1 to entry: The RESOURCE defined in IEC 61131-3 is a programming language element corresponding to the resource defined above.

Note 2 to entry: A device contains one or more resources.

[SOURCE: IEC 61499-1:2012, 3.80, modified – The words "contained within a device" have been added.]

3.1.62**resource management application**

application whose primary function is the management of a single resource

[SOURCE: IEC 61499-1:2012, 3.81]

3.1.63**service**

functional capability of a resource which can be modelled by a sequence of service primitives

[SOURCE: IEC 61499-1:2012, 3.87]

3.1.64**software**

intellectual creation comprising the programs, procedures, rules and any associated documentation pertaining to the operation of a system

[SOURCE: IEC 61499-1:2012, 3.92, modified – The word "configuration" has been deleted.]

3.1.65**system**

set of interrelated elements considered in a defined context as a whole and separated from its environment

Note 1 to entry: Such elements may be both material objects and concepts as well as the results thereof (e.g. forms of organization, mathematical methods, and programming languages).

Note 2 to entry: The system is considered to be separated from the environment and other external systems by an imaginary surface, which can cut the links between them and the considered system.

[SOURCE: IEC 60050-351:2013, 351-42-08, modified – The notes to entry 1,2,4 and 5 have been deleted and a new note 1 to entry added.]

3.1.66**Technology Block**

FB, which has at least one input or one output to the process

3.1.67**text dictionary**

collection of multilingual or other texts within the EDD

Note 1 to entry: References within an EDD are used to select an appropriate text dictionary.

3.1.68**type**

software element which specifies the common attributes shared by all instances of the type

[SOURCE: IEC 61499-1:2012, 3.99]

3.1.69**type name**

identifier associated with and designating a type

[SOURCE: IEC 61499-1:2012, 3.100]

3.1.70**variable**

software entity that may take different values, one at a time

Note 1 to entry: The values of a variable are usually restricted to a certain *data type*.

Note 2 to entry: Variables are described as input variables, output variables, and internal variables.

[SOURCE: IEC 61499-1:2012, 3.102, modified – In note 2, the words "internal variables and temporary variables" have been replaced by "and internal variables".]

3.2 Abbreviated terms

ADU	Analog digital unit
AFB	Application function block
ANSI	American National Standard Institute
ANSI C	American National Standard Institute for the programming language C (see ISO 9899-1990)
AP	Application process
ASCII	American Standard Code for Information Interchange (see ISO/IEC 10646)
ASN.1	Abstract Lexical Structure Notation 1
BNF	Backus Naur Format
CFB	Component function block
DAU	Digital analog unit
DD	Device description
DTD	Data type definition
EDD	Electronic Device Description
EDDI	Electronic Device Description Interpreter
EDDL	Electronic Device Description Language
EUC	Extended Unit Code (see ISO/IEC 2022:1994)
FB	Function block
FBD	Function block diagram
FMS	Fieldbus message specification
HMI	Human machine interface
HTML	Hypertext Mark-up Language
I/O	Input/output
IAM	Intelligent actuation and measurement
ID	Identifier
mA	Milliampere
NOAH	Network oriented application harmonisation
OSI	Open Systems Interconnection
P&ID	Piping and instrument diagram
PDU	Protocol data unit
SM	System management
TB	Technology Block
UML	Unified Modelling Language
wao	Write as one
XML	Extended Mark-up Language

3.3 Conventions for lexical structures

The EDDL is generally described using lexical structures in which the elements and the presence of fields are specified. A general form of the lexical structures is shown below.

ABC field1, field2

ABC is a lexical element. This element shall be coded in a concrete syntax. It is not required to code this element with the name "ABC". It is also possible to code this element for example with a tag number.

Field1 and field2 are fields of the lexical element ABC. Each field is mandatory and may have more than one attribute. If a field has attributes, the presence of the attributes is specified in a table. A comma separates field1 and field2. The comma is a lexical element and is not coded explicitly.

If a field has additional attributes, the attributes are defined in a table. The table layout and the possible usage qualifiers are shown in Table 1.

Table 1 – Field attribute descriptions

Usage	Attribute	Description
m	yyy	The presence of this attribute is mandatory
o	xxx	The presence of this attribute is optional
s	z1	The presence of this attribute is selectable with other attributes, which are also marked with "s" in the usage column. Only one of the selectable attributes (z1 or z2) is present
s	z2	The presence of this attribute is selectable with other attributes, which are also marked with "s" in the usage column. Only one of the selectable attributes (z1 or z2) is present
c	uuu	The presence of this attribute is conditional, and it is only present if the condition is true

The characters in the usage column have the following meanings:

- m: this attribute is mandatory and shall be present;
- o: this attribute is optional and need not be present;
- s: this attribute is a selection; one and only one of the fields marked with s (z1 or z2) shall be present;
- c: this attribute is conditional; the condition is described in the Description column.

If more than one attribute in the column Attribute exists, which have the same usage, the attributes are sorted in alphabetical order.

ABC field1+

The plus (+) behind field1 is used to indicate that field1 is used at least once. It may be used more than once.

ABC field2*

The asterisk (*) behind field2 is used to indicate that field2 is optional and need not be used. If it is used it may be used more than once.

ABC [field1, field2]+

The elements field1 and field2 in the square bracket [] are an unsorted list. The plus (+) behind the closing bracket indicates that field1 and field2 are used at least once and may be used more than once as a group.

ABC field1, (field2, field3)<exp>

<exp> indicates that field2 and field3 are used in conjunction with the conditional expression, (conditional constructs are specified in Table 1). The expression <exp> refers only to the fields within the preceding curved brackets (). Usage of conditional expressions is optional.

4 General function block (FB) definition and EDD model

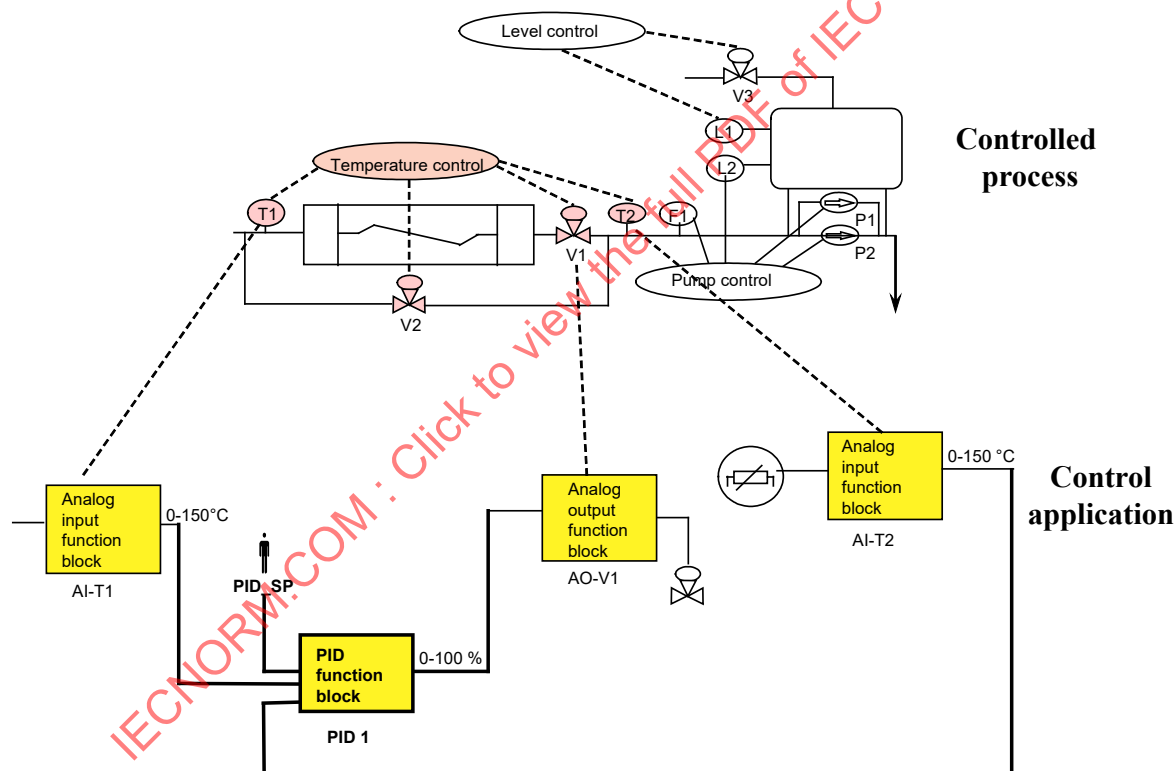
4.1 Device structure (device model)

4.1.1 Device model description

FBs are encapsulations of variables and their processing algorithms. The variables and algorithms are those required by the design of the process and its control system.

NOTE 1 FBs can be derived from the diagram in Figure 2.

FBs perform the application (measurement, actuation, control and monitoring) by connecting their data inputs and data outputs.

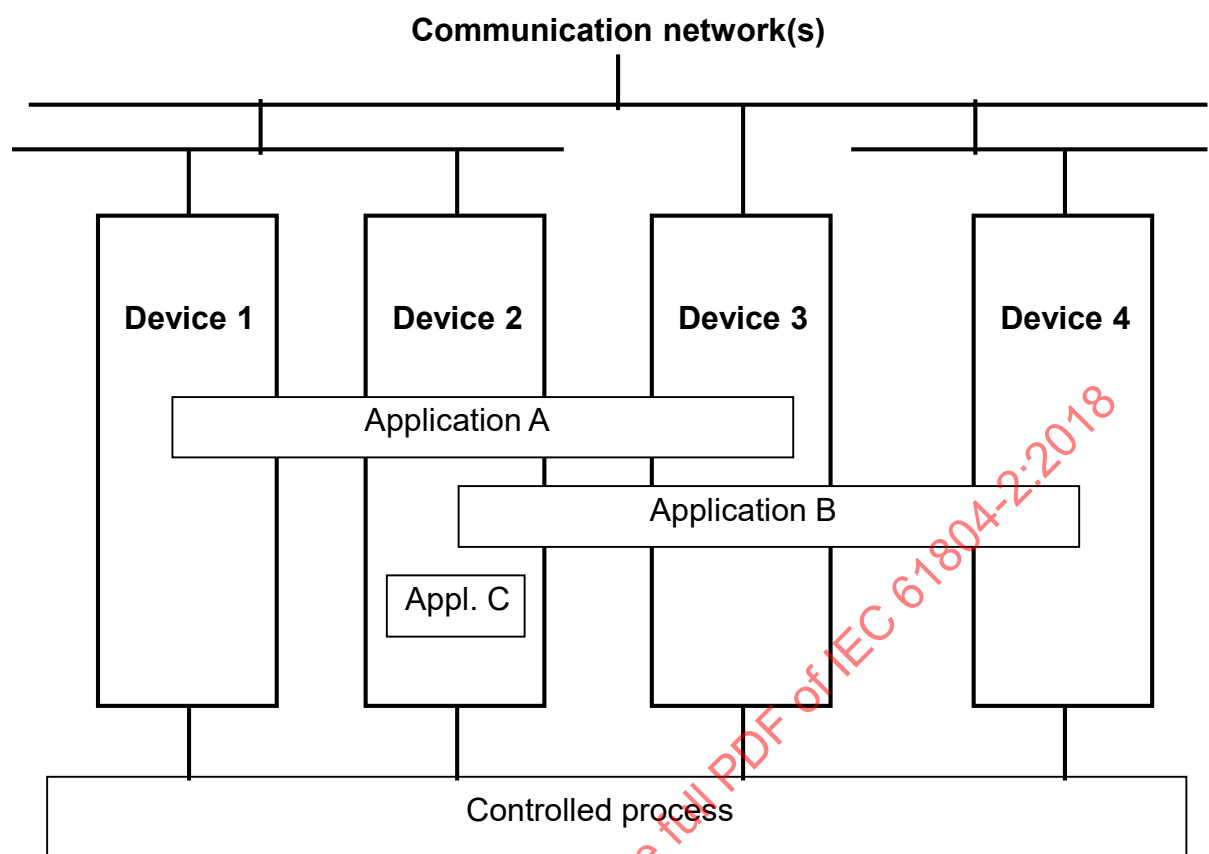


IEC

Figure 2 – FB structure is derived out of the process (P&ID view)

The devices are connected via a communication network or a hierarchy of communication networks.

NOTE 2 The application can be distributed among several devices, see for example Figure 3.



IEC

Figure 3 – FB structure may be distributed between devices (according to IEC 61499-1)

The FBs resulting from the design of the control system are abstract representations.

NOTE 3 These can be implemented in different ways in different device types, see Figure 4. FBs can be implemented, for example in field devices, programmable logic controller, visualization stations and device descriptions.

Additionally other applications such as system engineering and supervisory systems have to handle or interact with the FBs.

NOTE 4 Algorithms defined for an FB in the conceptual model are not necessarily mapped one-to-one to the device; they can be mapped to the device, a proxy or a supervisory station if the current technology does not solve it in the device.

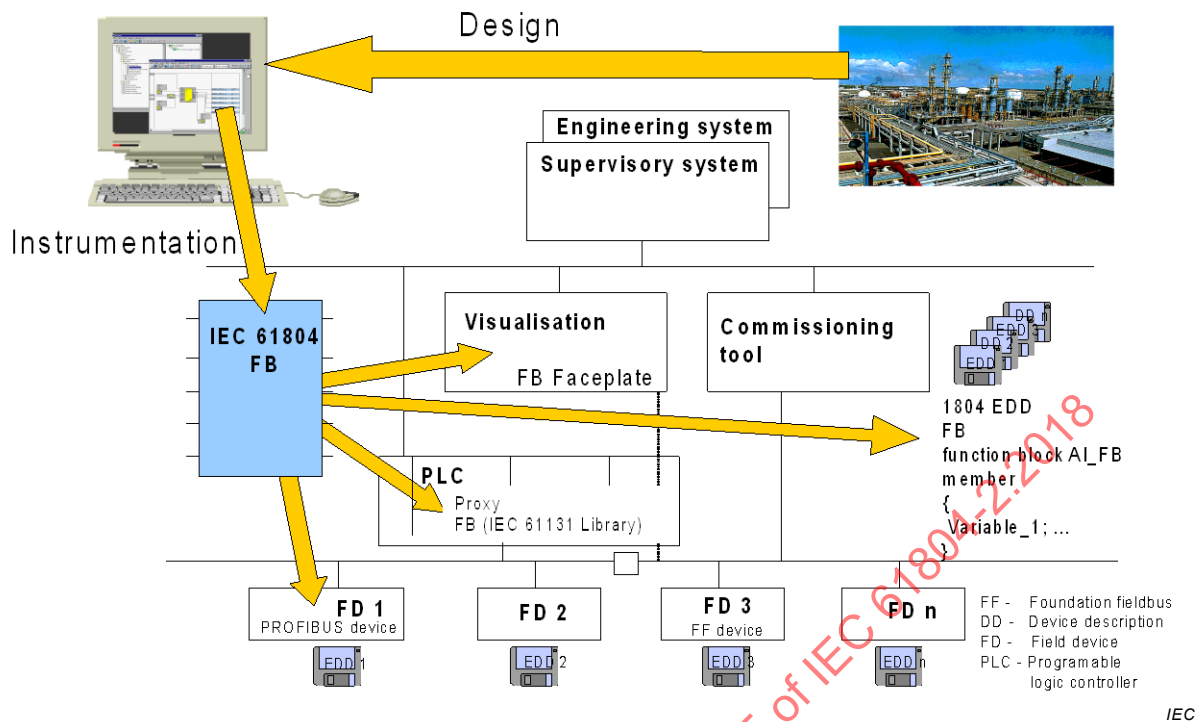


Figure 4 – IEC 61804 FBs can be implemented in different devices

For the purposes of this document, devices implement algorithms derived out of the design of the controlled process in terms of FBs. The devices are hardware and software modular, see for example, Figure 5. The components of devices are Modules, Blocks, Variables and Algorithms. There are defined relations between the components that are specified in the UML class diagram below, see Figure 8.

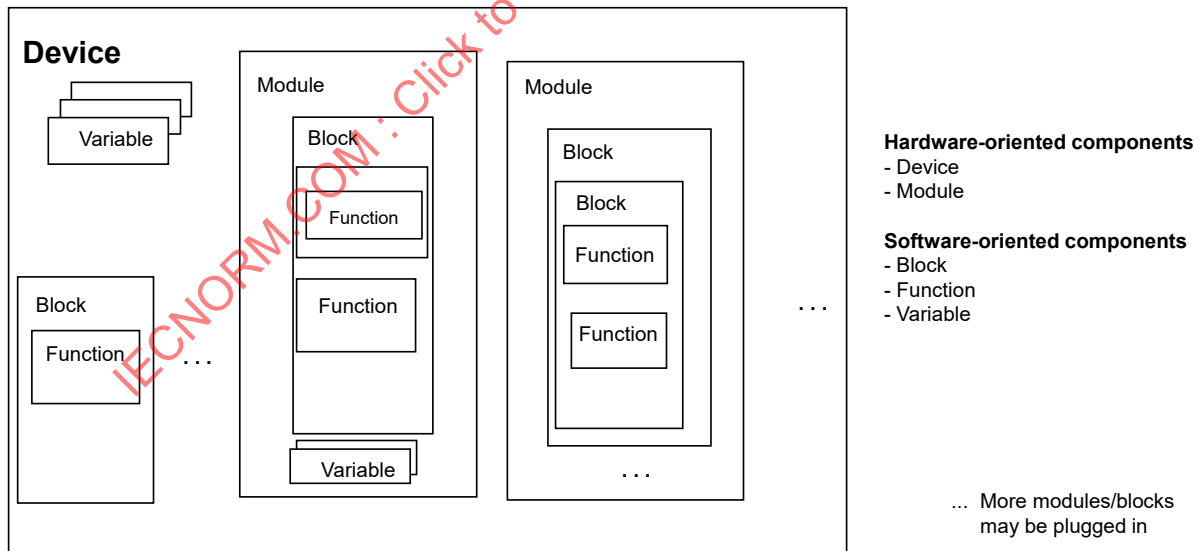
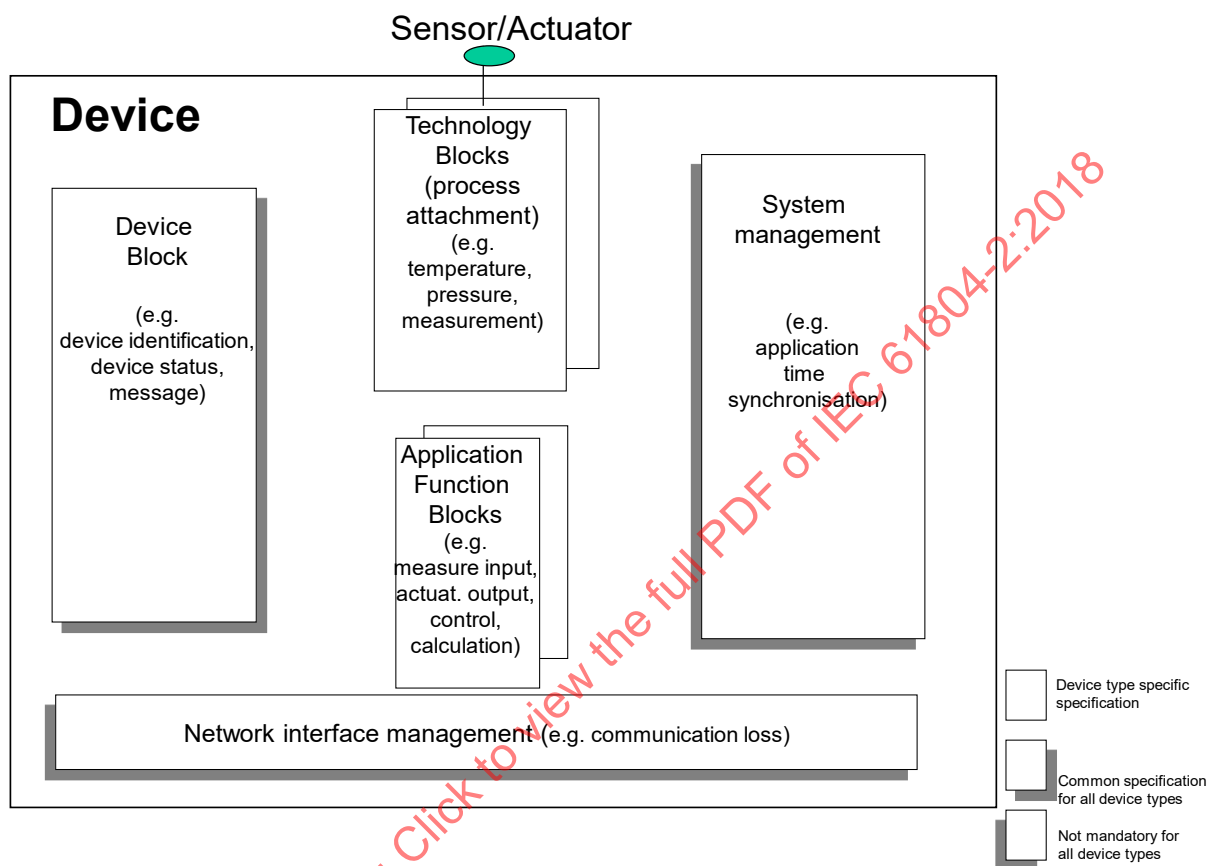


Figure 5 – General components of devices

For the purposes of this document, there are different block types (see Figure 6), which encapsulate the specific functionality of devices performing an automation application. The Technology Block represents the process attachment of a device. It contains the measurement or actuation principles of a device. The Technology Block is composed of acquisition or output and transformation parts. The application FB (hereafter called FB) contains application related signal processing, such as scaling, alarm detection or control and calculation. Component FBs may perform mathematical and logical processing with specific

additional exception handling procedures such as non-authorized parameter values. They shall be encapsulated within composite FBs.

The Device Block represents the resource of the device that contains information and function about the device itself, the operation system of the device and the device hardware. The device shall have an interface to the communication system and may have system management functionalities.



IEC

Figure 6 – Block types of IEC 61804 (all parts)

All devices within the scope of this document shall have the same logical device structure, see Figure 6. The number and types of blocks, which are instantiated in a device, are device and manufacturer specific. At least, the device shall have one Device Block, one application FB and one network interface management.

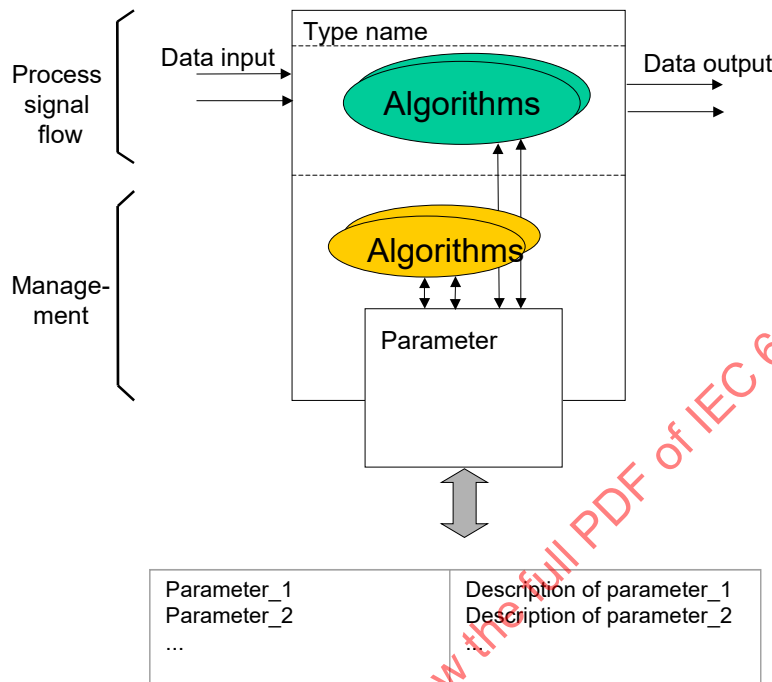
There is a data flow chain from signal detection through the Technology Block and FBs and vice versa. The signals between the parts of the chain are internal within the blocks or visible as linkages between blocks. The logical chain of technology and FB is called a channel. This concept is clarified in 4.2.1 and 4.2.2.

4.1.2 FB type

FBs are functional units in software, which encapsulate variables and algorithms. An FB type is defined by its behaviour. One FB contains one or more than one algorithm. The description of an FB is a list of algorithms, which are encapsulated in the FB together with the related data inputs and data outputs and parameters. There are algorithms, which are related to the process signal flow and those, which are related to other block specific algorithms. These other algorithms are called management. Parameters are related to process signal flow and management.

Graphical representation is not normative, see Figure 7. In other words, the data inputs and data outputs represent the intention of the process signal flow (conceptual definition), not the specific data that carry the corresponding values.

The parameter table specifies all the needed accessible data inputs, data outputs and parameters of the FB.



IEC

Figure 7 – IEC 61804 block overview (graphical representation not normative)

The FB is summarised by the following components:

- a) data inputs¹ which support status² and are related to the process signal flow only;
- b) data outputs¹ which support status² and are related to the process signal flow only;
- c) parameters² which are related to the process signal flow and management;
- d) parameters² to influence functions;
- e) parameters² to notify and make visible internal behaviour;
- f) parameters² to select functions in the signal flow;
- g) internal variables with memory for support of for example initialisation;
- h) mathematical/logical algorithm.

The influence of the FB behaviour is possible by data inputs and parameters only. The data inputs and parameters are used in the following ways:

- i) data, which are used as inputs or outputs of functions (e.g. setpoint for scaling functions);
- j) data, which are used as parameter of functions (e.g. limits for alarms and warnings);
- k) changes of parameter data are interpreted as events which switch transitions of state automata (e.g. start, stop, resumption of operation modus of devices);

¹ The decision which data of an FB is a data input, data output or parameter depends on specific implementations.

² For consistency reasons, data input/data output and status are in one structure, so that both belong to each other.

- l) changes of parameter data are interpreted as events, which start the transactions of sequences of algorithms (e.g. start of calibration procedures).

The data name and their description have to be checked to understand the purpose of the data.

4.1.3 FB execution

There are different execution control methods within devices. Execution control of FB algorithms is a feature of each device. There can be different execution policies within devices and in a distributed system.

NOTE For example, combinations of the following execution control methods are possible and others can be added:

- a) free running;
- b) device internal time schedule (time synchronization);
- c) device internal event triggered;
- d) parameter data changes are interpreted as events (see 4.1.2);
- e) system wide time synchronization (time synchronization across the communication system);
- f) communication service triggered;
- g) system wide event triggered (e.g. IEC 61499-1);
- h) distributed execution control;

The FB execution control within a device is only one aspect of the overall application execution control. The overall execution control is determined for example:

- i) sequence order (sequential or parallel):
 - 1) execution order of blocks along the signal flow;
 - 2) piping of data in parallel execution;
 - 3) handling of loss of communication between devices;
- j) synchronization:
 - 1) time synchronization between device;
 - 2) use of time in scheduling;
- k) time constraints; the following elements are covered:
 - 1) block execution time;
 - 2) communication time delay;
 - 3) scan rate of measurement;
 - 4) actuation time;
 - 5) choice of block algorithms;
 - 6) time delay resulting of communication behaviour;
- l) block execution time:
 - 1) communication time delay;
 - 2) scan rate of measurement;
 - 3) actuation time;
 - 4) choice of block algorithms;
- m) impact of exception handling:
 - 1) clock error;
 - 2) device error;
 - 3) communication error.

The decision of which technology fulfils the requirements has to be based on a detailed check at least of all these aspects. The choice of execution control method also depends on the technology level used to build the devices. So the method of FB execution control is also constrained to those available in the fieldbus used by the system.

4.1.4 Reference between IEC 61499 and IEC 61804 models

The relations to IEC 61499-1 are given in Table 2.

Table 2 – Equivalences between IEC 61804 and IEC 61499 model elements

IEC 61804 model element	IEC 61499 model element
Reference of block types	
Application FB	Application FB
Technology Block	Technology Block
Device (Resource) Block	Device (Resource) Block
Reference of Function Block elements	
Component Block	Component Block
Type Name	Type Name
Data Input ^a	Data Input ^a
Data Output ^a	Data Output ^a
Algorithms	Algorithms
Parameter	Parameter
Internal Variable	Internal Variable
Principle relations between EDDL elements and IEC 61499 transfer syntax elements ^b	
BLOCK_A, BLOCK_B	FUNCTION BLOCK
VARIABLE and CLASS INPUT	VAR_INPUT, END_VAR
VARIABLE and CLASS OUTPUT	VAR_OUTPUT, END_VAR
-	ALGORITHM
VARIABLE and CLASS CONTAINED	-
VARIABLE	VAR, END_VAR
^a The data inputs and data outputs represent the source and sink points for the process signal flow (conceptual definition) not the specific variables, which carry the according data. ^b This is not an exact syntax reference. It is intended to show the general relations.	

An IEC 61804 FB is an IEC 61499-1 FB without execution control and therefore has no event inputs and event outputs. The execution control of the IEC 61804 FBs algorithms are hidden, see 4.1.3.

4.1.5 UML specification of the device model

The device model definitions in 4.1.1, in Figure 5, and Figure 6, are general. To solve the ambiguity, the model is described as a UML class diagram (see ISO/IEC 19501-1). The components are transformed to the UML language elements in Figure 8.

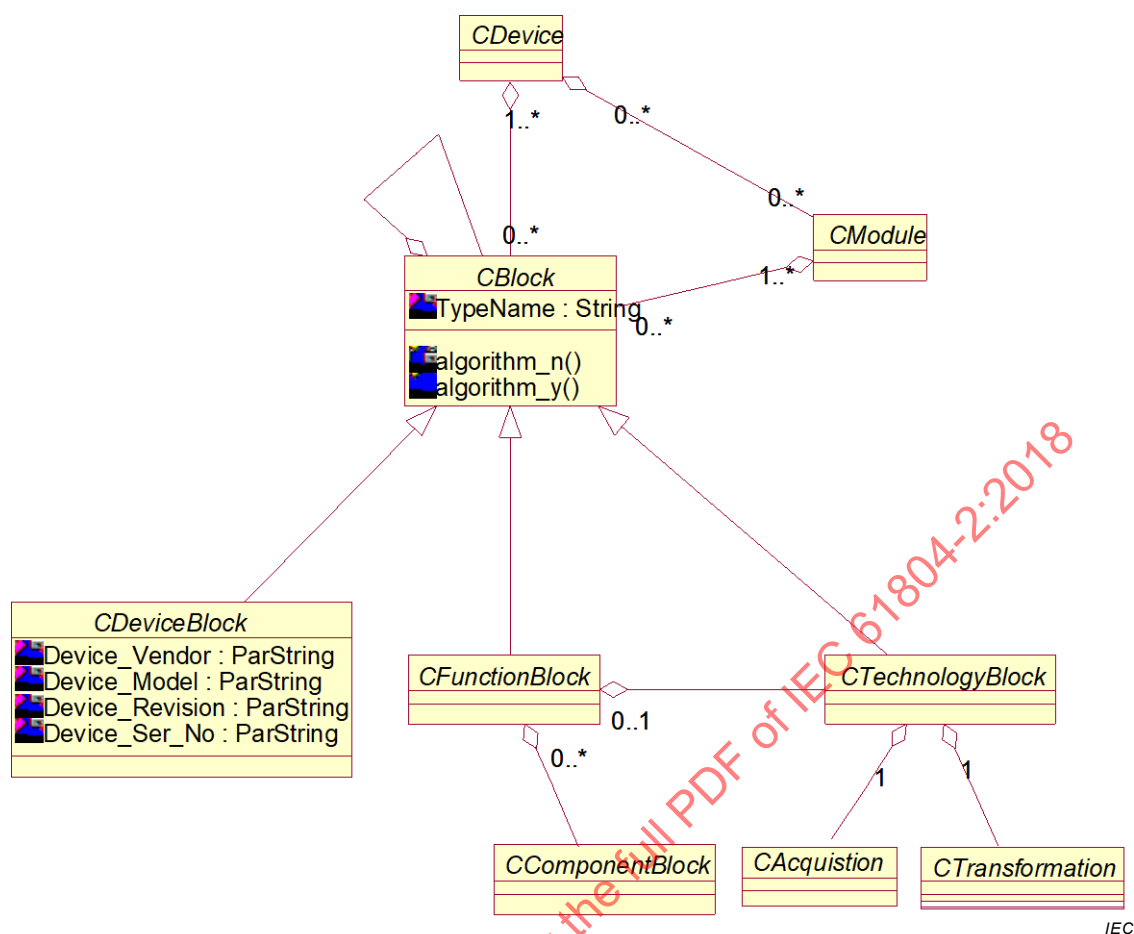


Figure 8 – UML class diagram of the device model

The following major steps are used to convert the device model into a UML class diagram:

- Device becomes the class **CDevice**;
- Module becomes the class **CModule**;
- Device Block, FB, Component FB and Technology Block becomes **CDeviceBlock**, **CFunctionBlock**, **CComponentFunctionBlock** and **CTechnologyBlock**;
- the block types are of the type Block which becomes **CBlock**;
- a Device contains a minimum of one block;
- a Device may contain modules;
- a Module contains a minimum of one block;
- blocks can be composed out of other blocks, i.e. may be of composite FB type;
- a Block contains a minimum of zero or more parameter;
- a Block shall have algorithms which can be internal only or visible from the outside (i.e. private or public);
- a Device Block contains the attribute **Device_Vendor**, **Device_Model**, **Device_Revision** and **Device_Ser_No** which are parameters;
- the FB, Component FB and Technology Block contain the attribute **TypeName**.

NOTE The **CBlock** class can be referenced to the Basic FB Type Declaration of the IEC 61499-1. The IEC 61804 block type has no aggregation to the **ECCDeclaration** class.

4.1.6 Classification of the algorithms

The following list provides common algorithms for use in application FBs, transducer blocks and Device Blocks.

a) Process signal algorithms:

1) measurement acquisition:

- i) sensor connection;
- ii) sensor range/calibration;
- iii) AD conversion;
- iv) status estimation.

2) measurement transformation:

- i) linearization;
- ii) filtering;
- iii) compensation;
- iv) scaling.

3) Measurement application:

- i) limit;
- ii) unit;
- iii) scaling;
- iv) linearization;
- v) simulation.

4) Actuation provision:

- i) amplification;
- ii) conversion;
- iii) status estimation.

5) Actuation acquisition:

See measurement acquisition for readback of actuator output value.

6) Actuation transformation:

- i) scaling;
- ii) compensation;
- iii) transition or activity limits.

7) Actuation application:

- i) limit;
- ii) unit;
- iii) scaling;
- iv) linearization;
- v) simulation.

b) Management:

- 1) estimation of Device status;
- 2) test;
- 3) diagnosis;
- 4) operating mode.

4.1.7 Algorithm description

The algorithm description is done individually for each algorithm in the appropriate language, for example plain English, Harel State Diagram or one of the IEC 61131-3 languages (for example FBD (function block diagram) or IEC 61131 ST (structured text)).

The objective of the profile description is to define a general set of rules allowing identification of a device together with classification and specification of the algorithms supported by the device.

4.1.8 Input and output variables and parameter definition

For the description of the block parameters, Table 3 table shall be used. This table provides a template for describing the interface to a block. It is comparable with a data dictionary or a database.

Table 3 – Variables and parameter description template

Parameter name	Description	Data type	User Access Read/Write	Class m/o/c
Block class				

Parameter name:

Identifier of the variable/parameters that are accessed within the FB. The name is valid within this specification but not normative for products on the market. The decision if a data is an input, output or parameter is application dependent.

Description:

Informative text, describing the purpose of the variable/parameter.

Data type:

The following data types are conceptual ones, i.e. they identify the signal type not the implementable data type. These will be mapped by technology profile to supported data in the following categories:

- a) numeric (e.g. float, real, long real, integer);
- b) enumerated;
- c) Boolean;
- d) string (e.g. visible string, octet string);
- e) array;
- f) structure.

User Access Read/Write

This specifies that the variable/parameter is changeable by a remote device or not.

Class m/o/c

This specifies if the variable/parameter shall be supported within the block or not, the states are: mandatory (m), optional (o), conditional (c).

Additional parameter attributes which shall be specified when mapping of IEC 61804 blocks to other FB specifications are:

g) Class of recovery after power fails shall have the value N or D as follows:

- N indicates a non-volatile parameter which shall be remembered through a power cycle, but which is not under the static update code;
- D indicates a dynamic parameter which is calculated by the process, the block or read from another block.

h) Default value:

indicates the value is assigned to parameter in the initialisation process for an unconfigured block.

4.1.9 Choice of variables and parameters

The block variables, parameters and algorithms included in a block will be those that are significant for the algorithm and device. As a minimum, FBs will include the variables and parameters defined in the P&ID. The names of parameters and variables are not normative.

4.1.10 Mode, Status and Diagnosis

These parameters manage and indicate channel performance. They can be reported, however the report mechanisms are technology dependent. Reported values may also include additional items such as time stamps, priorities, indication of possible reasons.

Mode describes the operation state of a channel or FB and influences the signal flow within the channel. Examples of modes are: Manual, Automatic, Local Override, Out of Service.

Status is a characteristic aspect of a channel which may accompany information transferred within the channel i.e. FB data inputs and data outputs.

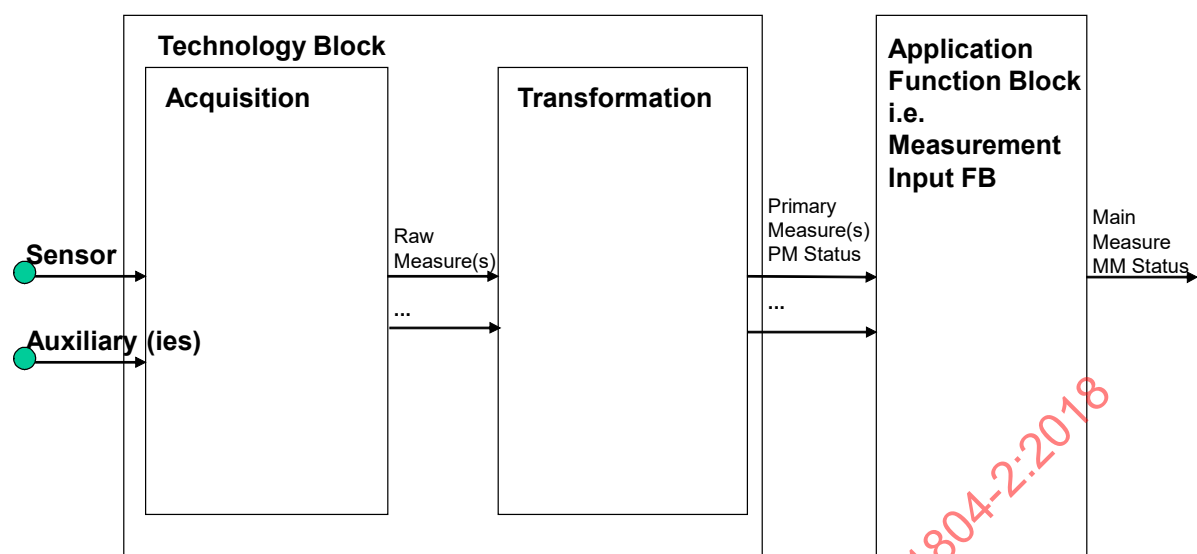
Device State describes the operational state of a device and interacts with the device technology and application blocks, it is maintained within a device by the Device Block.

Diagnosis is a report available from algorithms which assess channel or device internal performance. The results of these internal assessments may be used to construct generic measurement, control and actuation status information.

4.2 Block combinations

4.2.1 Measurement channel

The technology and application FBs provide a functional chain along which the process signals flow. Together they comprise a measurement channel (see Figure 9) or an actuation channel (see Figure 10).



IEC

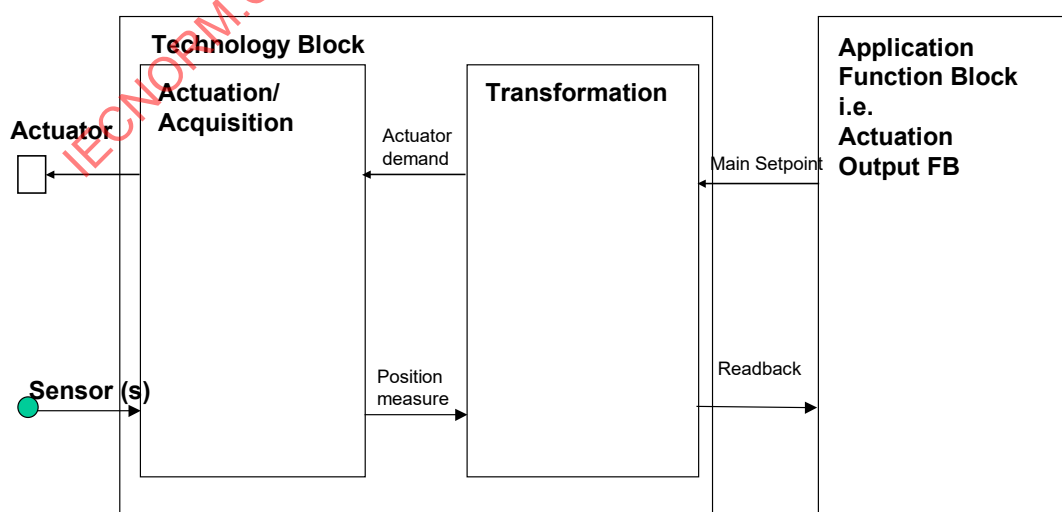
Figure 9 – Measurement process signal flow

A measurement may be accompanied by optional additional auxiliary measurements, for purposes such as compensation. The Technology Block provides a primary measured value and its accompanying status. Additionally, the Technology Block may provide other outputs, for example diagnosis or validation information.

NOTE Additional sensor inputs can also be used and transferred by a Technology Block.

The application FB uses the outputs of the Technology Block and other internal data to generate the main measure and its accompanied status. The status is accomplished by every function in the signal flow starting with the sensor(s) until the last function in the application FB. Information from one Technology Block is offered to more than one application FB. A measurement channel shall consist of at least one application FB. Channels without a Technology Block are possible.

4.2.2 Actuation channel



IEC

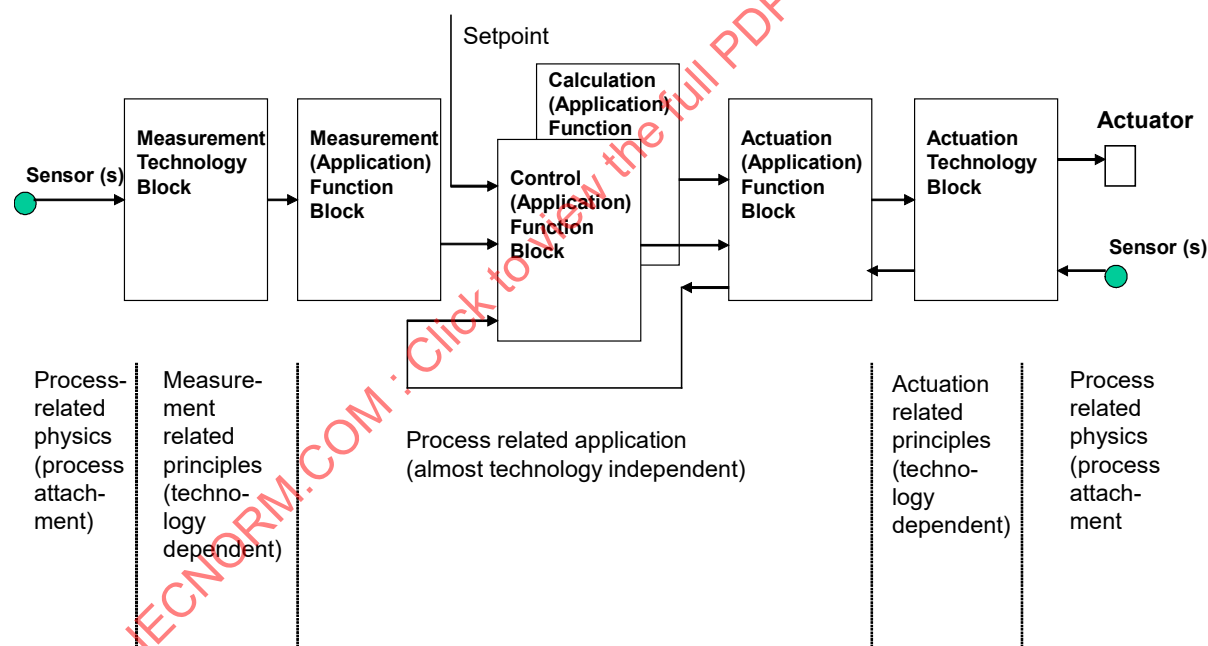
Figure 10 – Actuation process signal flow

The actuation channel is performed out of the function of the actuation signal flow and the additional measurement functions for the measurement of the current position of the actuator. If there is not a sensor for the position measurement, then the actuator demand will be used in the transformation to determine the readback value. Optionally, status values may accompany both signal flow directions and include information about the involved entities. The status accompanying the main setpoint carries information to give the Technology Block the opportunity to go in fail-safe position, if the main setpoint is not good. The status accompanying the readback carries information if the measure value is good or not. An actuation channel shall consist of at least one application FB. Channels without a Technology Block are possible.

4.2.3 Application

A complete application is supported by combinations of measurement and actuation channels together with control and calculation FBs (see Figure 11). The Technology Blocks are technology dependent and the other FBs are technology independent. There may be many different implementations of an application, depending on the technology used within the devices. The application may be performed by implementations using only measurement and actuation devices (i.e. complex devices able to perform measurement, control and actuation) or the application may be built from measurement and actuation devices together with controller devices and other system components.

NOTE A controller can be for instance integrated in the application as one calculation FB or an actuation device can take parts of programmable functions from controller devices in terms of calculation FBs.



IEC

Figure 11 – Application process signal flow

4.3 EDD and EDDL model

4.3.1 Overview of EDD and EDDL

An electronic device description (EDD) contains all the device parameters of an automation system component. The technology used to describe an EDD is called electronic device description language (EDDL). The EDDL provides a set of scalable basic language elements to handle simple, complex or modular devices. The EDDL is a descriptive language based on an ASCII format with clear separation between data and program.

NOTE Data in a text field, which is marked with a country code like Japanese can use multi byte code.

4.3.2 EDD architecture

From the viewpoint of the ISO/OSI model (ISO/IEC 7498-1), an EDD is above Layer 7. However, the EDD application uses the communication system to transfer its information. An EDD contains constructs that support mapping to a supporting communication system.

The device manufacturer defines the objects, which are reflected by the logical representation of the objects within an EDD application. For that reason, EDDL has language elements, which map the EDD data to the data representation of the communication system, so that the user of an EDD does not need to know the physical location (address) of a device object.

EDD describes the management of information to be displayed to the user. The specific representation of such visualisation is not part of EDD or EDDL definitions.

4.3.3 Concepts of EDD

The manufacturer of a device or of an automation system component describes the properties of the device by using the EDDL. The resulting EDD contains information such as:

- description of the device parameters
- description of parameter dependencies
- logical grouping of the device parameters
- selection and execution of supported device functions
- description of the transferred data sets

Depending on the required usage, the EDD may be physically located:

- in a device
- in an external data storage medium such as a compact disk, floppy or a server
- partially distributed in the device and an external storage medium

EDD supports text strings (common terms, phrases etc.) in more than one language (English, German, French, etc.). Text strings may be stored in separate dictionaries. There may be more than one dictionary for one EDD.

An EDD implementation includes sufficient information about the target device, for example manufacturer, device type, revision. This is used to match a specific EDD to a specific device.

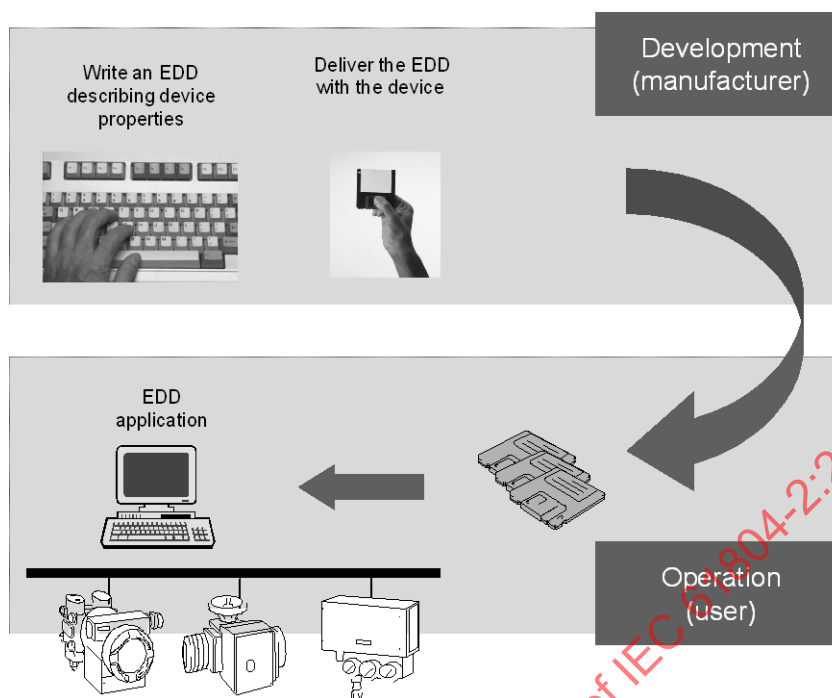
4.3.4 Principles of the EDD development process

4.3.4.1 General

Creation of an EDD is a three-stage process: EDD source generation, EDD pre-processing and EDD compilation.

4.3.4.2 EDD source generation

The EDD generation process is shown in Figure 12.



IEC

Figure 12 – EDD generation process

The device manufacturer writes an appropriate EDD for the device and delivers both (EDD and device). If the automation system supports the EDD method, a system integration can be made by the user.

The EDDs for devices may be embedded in the device memory, or delivered using separate storage media, or downloaded from an appropriate network server. The EDD is "interpreted" or "browsed" by an EDD-application. EDDs are normally stored as source files or preprocessed files.

4.3.4.3 EDD preprocessing

In the preprocessing stage, an EDD preprocessor generates a consistent EDD representation suitable for final compilation.

Preprocessing supports, for example, substitution of definitions and inclusion of external text. The output of the preprocessor is a complete EDD without any preprocessor directive.

4.3.4.4 EDD compilation

The EDD compilation stage produces an EDD application internal representation from a preprocessed EDD to be used in the EDD application.

4.3.5 Interrelations between the lexical structure and formal definitions

The lexical structure of the EDDL and its elements are described in IEC 61804-3 as well as the formal definitions and syntax for each EDDL element. The lexical structure and its formal description use the same name.

NOTE Instead of the specified formal definitions and syntax in IEC 61804-3 another specification can be developed as an additional option.

4.3.6 Builtins

Builtins are predefined subroutines which are executed in the EDD application.

EXAMPLE A hand held terminal is a simple device having a small display and limited cursor functions. For this type of device, a Builtin could be specified to provide display entry using only up/down right/left cursor actions.

The library of Builtins is defined in IEC 61804-5.

4.3.7 Profiles

EDDL is a harmonized specification of existing legacy EDD concepts. Concrete EDD applications use a subset of the EDDL specification. Selection of EDDL elements and Builtins is done in the profiles defined in IEC 61804-3.

In addition to EDD profiles, implementing consortia also publish "Device Profiles", which are used to support interchangeability of compliant devices. These Device Profiles may be described using the EDDL specified in this document.

5 Detailed block definition

5.1 General

This selection of blocks do not intent to be complete. It is a selection of very common measurement and actuation .

5.2 Application FBs

5.2.1 Analog Input FB

5.2.1.1 Analog Input FB overview

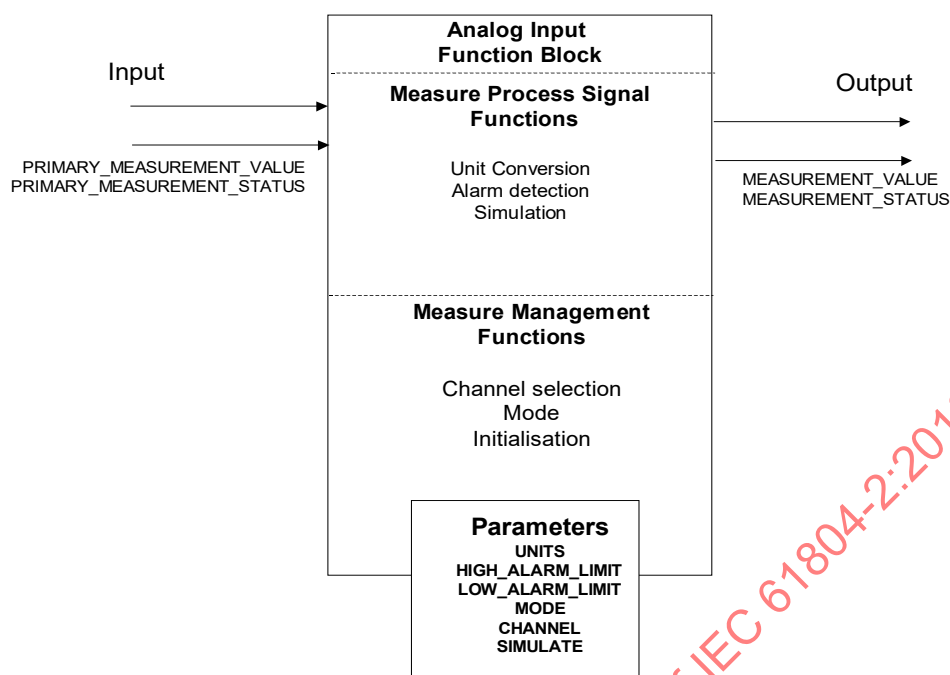
Figure 13 shows the block diagram of Analog Input FB.

The measure process signal function shall be used to convert signal(s) from a Technology Block to units appropriate for the primary measurement required for an application. The result is the MEASUREMENT_VALUE.

NOTE For example, conversion from inches of water to litres per minute. Also, this block can be used to provide operator notification that the primary measurement has detected a high or low alarm. The ability can be provided to simulate the process measurement during system checkout and testing.

Each process signal involves more information than only the value of the signal; the management parameters are generally required. Each measurement has a status, which indicates quality of the measurement value.

The status provided by the Technology Block is propagated to measurement (Input) FB by the PRIMARY_MEASUREMENT_STATUS. The status is a piece of information provided with every measurement to assist the user of measurement data (typically control functions) in assessing its utility. For example it may be a Boolean value (valid/non-valid), a continuous value (measurement uncertainty), a discrete value, or a combination thereof (see 5.6.1).



IEC

NOTE For the parameters description, see Annex A.

Figure 13 – Analog Input FB

5.2.1.2 Unit conversion

This algorithm converts the signal from a Technology Block into an understandable value. That may be used directly by the operator.

The user uses the UNITS to select the engineering units in which the MEASUREMENT_VALUE is to be displayed, for example bar or mbar.

NOTE This algorithm can also provide information on the channel and device operating state to assist in diagnostic of management activities.

5.2.1.3 Alarm detection

The FB shall provide the optional alarm detection inside.

EXAMPLE Low alarm, high alarm, deviation, update.

When implemented, the LOW_ALARM_LIMIT and HIGH_ALARM_LIMIT values shall be compared with the MEASUREMENT_VALUE of the FB. The results are high and low alarm notification, for example for an operator.

NOTE The way of reporting the detected alarms is technology dependent, therefore it is not described in this document and shown in the corresponding figure.

5.2.1.4 Simulation

This algorithm shall be used to simulate the MEASUREMENT_VALUE value to an assigned value using the SIMULATE parameter. This operation is usually carried out during commissioning, adjustment phases, or test purposes, and allows the running application to be temporarily uncoupled from the process.

5.2.1.5 Channel selection

One Technology Block will be used for primary final element data. Channel numbers (CHANNEL) will be defined for the measurement device when using more than one Technology Block.

5.2.1.6 Mode

The mode algorithm determines the source of the output for a measurement input FB based on the MODE parameter value. In Automatic mode, the measurement algorithm determines the output. When mode is set to Manual, the output of the FB is set by another source, for example it may be set by the operator.

5.2.1.7 Initialisation

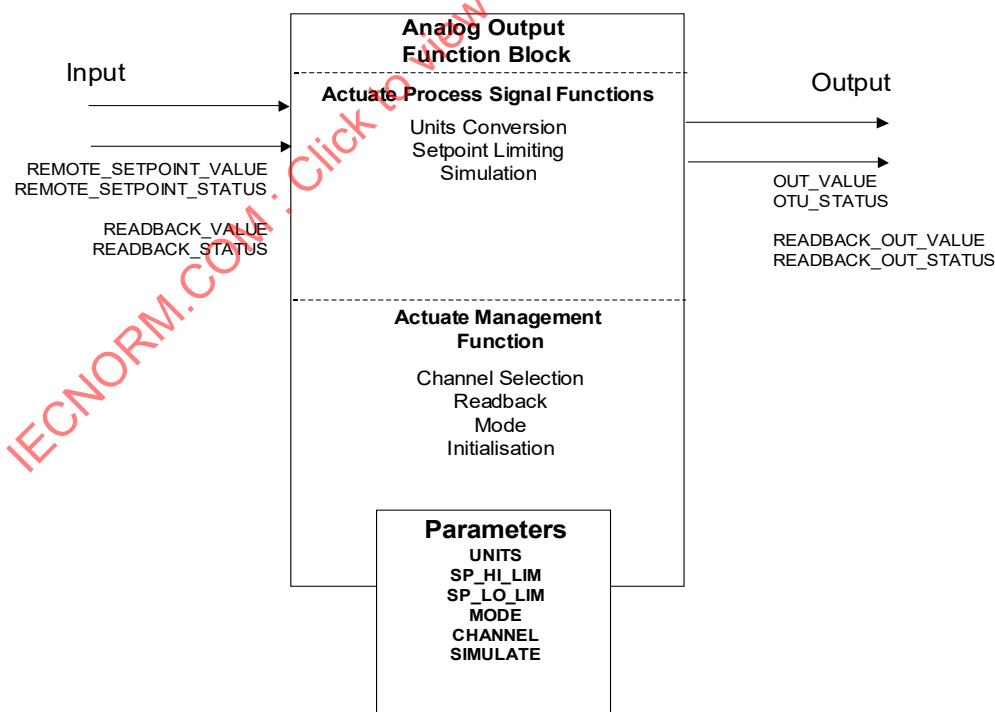
The initialisation algorithm is applied to this block and described in 5.6.3.

5.2.2 Analog Output FB

5.2.2.1 Analog Output FB overview

Figure 14 shows the block diagram of the Analog output FB.

The actuation process signal algorithm converts REMOTE_SETPOINT_VALUE into a useful value (OUT_VALUE) for the hardware specified by the channel selection for the Technology Block. The feedback value (received from the actuator) is provided as the READBACK_VALUE. If the Analog Output FB is part of a cascade chain, then the READBACK_OUT_VALUE provides the actual value to the upstream FB. All these input and output parameters shall be accompanied by their status (see 5.6.1).



IEC

NOTE For the parameters description, see Annex A.

Figure 14 – Analog Output FB

5.2.2.2 Unit conversion

This algorithm converts the REMOTE_SETPOINT_VALUE to a value which can be used by the actuator. UNITS of the REMOTE_SETPOINT_VALUE main setpoint value define the units of the setpoint. The READBACK_VALUE (i.e. the actual delivered value or the final demanded value) is also provided in the units of the setpoint.

5.2.2.3 Setpoint limiting

The REMOTE_SETPOINT_VALUE that is provided to the FB will be limited to the setpoint lower (SP_LO_LIM) and higher (SP_HI_LIM) range limits.

5.2.2.4 Simulation

This algorithm is used to force the READBACK_VALUE and the READBACK_STATUS to the assigned values through the SIMULATE parameter. The simulation can be used for example to simulate Technology Block faults. In simulation mode, the Technology Block ignores the Analog Actuation FB output value(s) and maintains the last value. This operation is usually carried out during commissioning, adjustment phases, or test purpose, and allows the running application to be temporarily uncoupled from the process.

5.2.2.5 Channel selection

One Technology Block will be used for primary final element data. Channel numbers (CHANNEL) will be defined for the Modulation Actuator Device when using more than one Technology Block.

5.2.2.6 Readback

This algorithm gives information about the actual delivered value of the actuator in the process.

The READBACK_STATUS information is provided to reflect the state of the actuating value. This may be a Boolean value (valid/non-valid), a continuous value (measurement uncertainty), a discrete value, or a combination thereof.

5.2.2.7 Fail safe

The fail safe algorithm is described in 5.6.4.

5.2.2.8 Mode

The mode algorithm determines the source of the output for the modulating actuation FB based on the MODE parameter value. In Automatic mode, the output is determined by the modulating actuation algorithm. When the mode is set to Manual, the output of the FB is set by another source, for example it may be set by the operator.

5.2.2.9 Initialisation

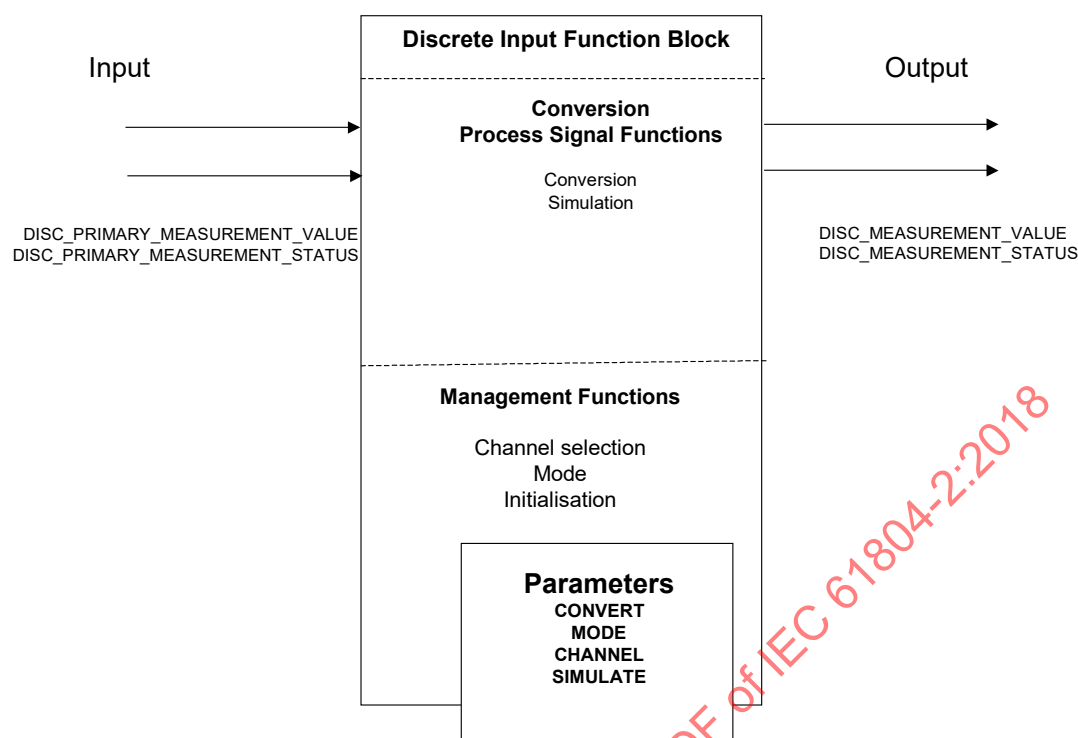
The initialisation algorithm is applied to this block and described 5.6.3.

5.2.3 Discrete Input FB

5.2.3.1 Discrete Input overview

Figure 15 shows the block diagram of the Discrete input FB.

Discrete Inputs represent for example inductive, optical, capacitive, ultrasonic proximity switches. When the digital input changes state the discrete output changes state too.



IEC

NOTE For the parameters description, see Annex A.

Figure 15 – Discrete input FB

5.2.3.2 Conversion

This algorithm converts the Boolean or discrete measure in a logical signal.

The result is the DISC_MEASUREMENT_VALUE accompanied by the DISC_MEASUREMENT_STATUS.

5.2.3.3 Channel selection

One Technology Block will be used for primary final element data. Channel numbers (CHANNEL) will be defined for the discrete detection device when using more than one Technology Block.

5.2.3.4 Simulation

This algorithm is used to force the main discrete value to an assigned value using the SIMULATE parameter. This operation is usually carried out during commissioning, adjustment phases, or test purposes, and allows the running application to be temporarily uncoupled from the process.

5.2.3.5 Mode

The mode algorithm determines the source of the measure input FB output (main discrete measure) based on the MODE parameter value. In Automatic mode, the discrete measure algorithm determines the output. When mode is set to Manual, the output of the FB is set by another source, for example it may be set by the operator.

5.2.3.6 Initialisation

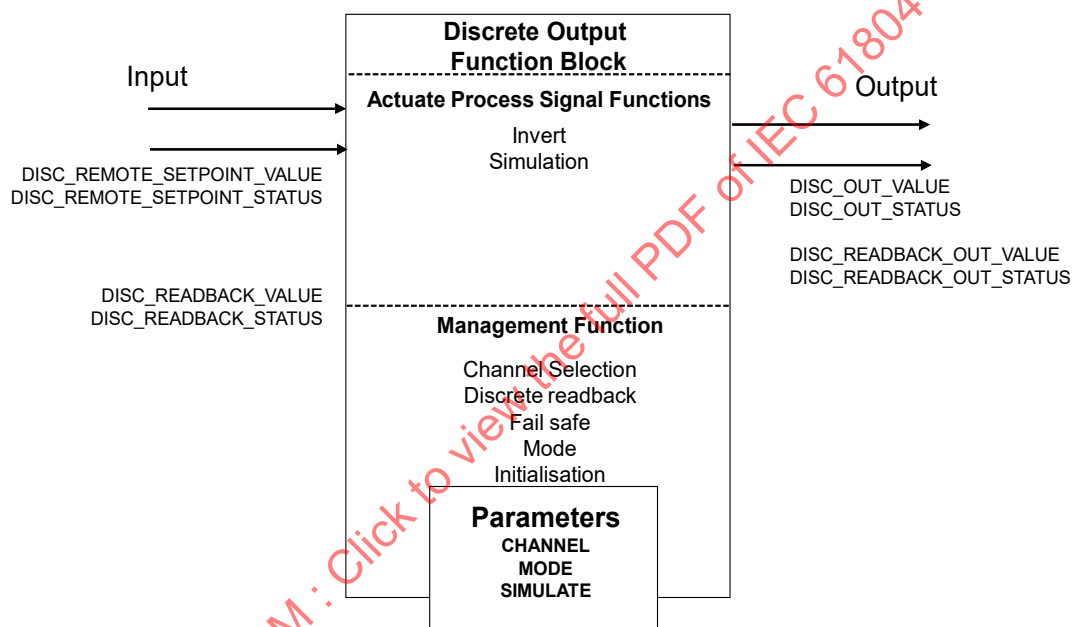
The initialisation algorithm is applied to this block and described in 5.6.3.

5.2.4 On/Off Actuation FB Discrete Output FB

5.2.4.1 On/Off Actuation FB Discrete Output FB overview

Figure 16 shows the block diagram of the Discrete output FB.

The actuation process signal algorithm converts the DISC_REMOTE_SETPOINT_VALUE value to a useful value (DISC_OUT_VALUE) for the hardware at the channel selection to the Technology Block. The DISC_READBACK_VALUE defines the target value of the final element. Is the Discrete Output FB part of a cascade chain the DISC_READBACK_OUT_VALUE provides the actual value to the upstream FB. All these input and output parameters shall be accompanied by their status (see 5.6.1).



IEC

NOTE For the parameters description, see Annex A.

Figure 16 – Discrete Output FB

5.2.4.2 Invert

Sometimes it is necessary to invert logically the DISC_REMOTE_SETPOINT_VALUE before forwarding it to the discrete actuation demand. This is done in this algorithm.

5.2.4.3 Simulation

This algorithm is used to force the DISC_READBACK_VALUE and the DISC_READBACK_STATUS to assigned values using the SIMULATE setting. The simulation can be used for example to simulate Technology Block faults. In simulation mode, the Technology Block ignores the DISC_OUT_VALUE value and maintains the last value. This operation is usually carried out during commissioning, adjustment phases, or test purpose, and allows the running application to be temporarily uncoupled from the process.

5.2.4.4 Channel selection

One Technology Block will be used for the primary final element data. Channel numbers (CHANNEL) will be defined for the Modulation Actuator Device when using more than one Technology Block.

5.2.4.5 Fail safe

The fail safe algorithm is described in 5.6.4.

5.2.4.6 Mode

The mode algorithm determines the source of the on/off actuation FB output based on the MODE parameter value. In Automatic mode, the on/off actuation algorithm determines the output. When the mode is set to Manual, the output of the FB is set by another source, for example it may be set by the operator.

5.2.4.7 Initialisation

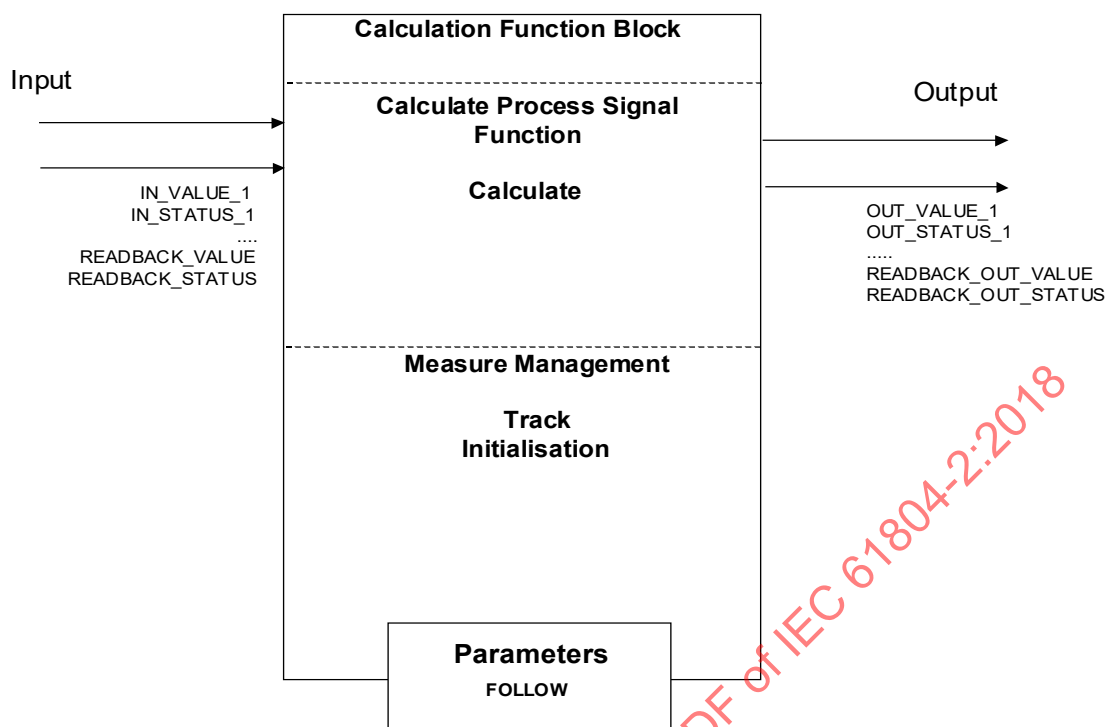
The initialisation algorithm is applied to this block and described in 5.6.3.

5.2.5 Calculation FB

5.2.5.1 Calculation FB overview

Figure 17 shows the block diagram of Calculation FB.

The Calculation FB acts upon the input signal(s) (IN_VALUE_x) from other FBs to provide an application value (OUT_VALUE_x). If the Calculation FB is part of a cascade chain the READBACK_OUT_VALUE provides the actual value to the upstream FB and the READBACK_VALUE is provided by a downstream FB. All these input and output parameters shall accompanied by their status (see 5.6.1).



IEC

NOTE For the parameters description see Annex A.

Figure 17 – Calculation FB

5.2.5.2 Calculate

This algorithm determines the output signal(s) based on a pre-defined algorithm and the input(s) to the FB. Example calculation functions are filtering, delay, input select.

5.2.5.3 Track

The track algorithm allows the FB output to be set to an input value when the FOLLOW parameter is active i.e. non-zero in value. For example this algorithm may be used to initialise a block or to force the calculation results to a specific value.

This does not apply to all blocks.

5.2.5.4 Initialisation

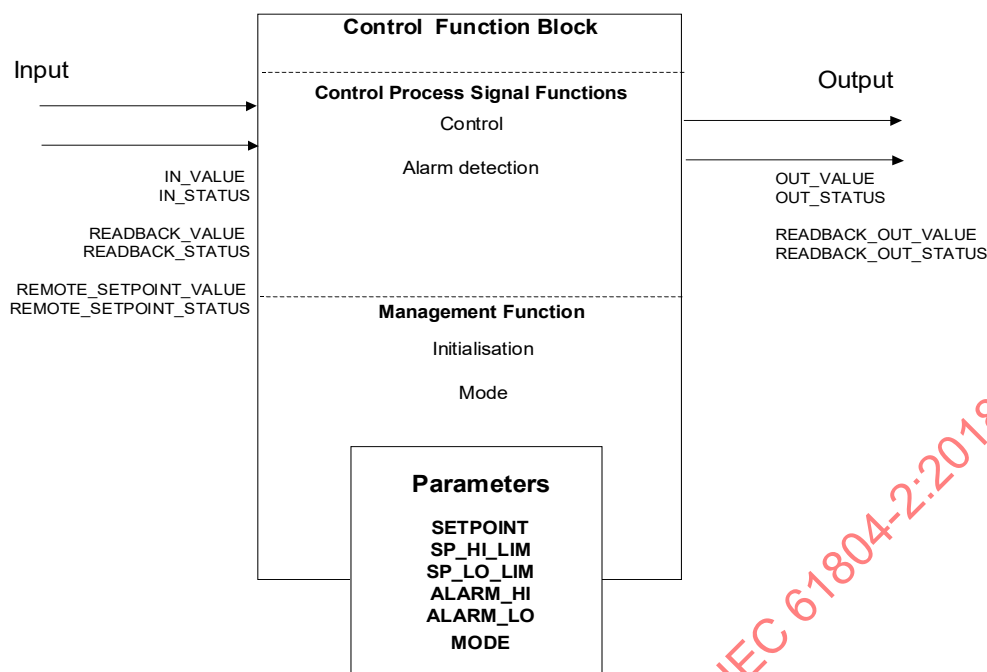
The initialisation algorithm is applied to this block and is described in 5.6.3.

5.2.6 Control FB

5.2.6.1 Control FB overview

Figure 18 shows the block diagram of the Control FB.

The control FB maintains a process input (IN_VALUE) at the setpoint value (SETPOINT) through the regulation of one or more process actuation outputs. The process input measurement is provided by an appropriate FB through the primary input connection. The primary output of the control FB regulates the process through an appropriate actuation FB. The SETPOINT defines the target value of the process measurement in mode "auto". The readback value and status provided by the downstream actuation block may be used in the initialisation and to modify the control action when the output is limited by a downstream condition.



IEC

Figure 18 – Control FB

5.2.6.2 Control

This algorithm determines the FB output value that is needed to drive the primary input value to the target value specified by the SETPOINT parameter. Changes in the SETPOINT value are limited to the range specified by the SP_HI and SP_LO limits. Actions may be modified when a readback input from the downstream block indicates that a downstream condition limits the adjustment of the block output.

5.2.6.3 Alarm detection

The alarm detection is optional. When implemented, the LOW_ALARM_LIMIT and HIGH_ALARM_LIMIT values shall be compared with the primary control measurement value of the block. The results are high and low alarm notification for example for an operator.

NOTE The way of reporting the detected alarms is technology dependent therefore it is not described in this document and shown in the according figure.

5.2.6.4 MODE

The mode algorithm determines the source of the control block output based on the MODE parameter value. In Automatic mode, the output is determined by the control algorithm and the SETPOINT is specified by the operator. When the mode is set to Manual, the output of the block is set by another source, for example it may be set by the operator.

In remote mode, the output is determined by the control algorithm and the setpoint is determined by the REMOTE_SETPOINT input from another FB.

5.2.6.5 Initialisation

When the feedback status indicates that the path to the process input is blocked, the output of the FB will be set based on the readback value to provide bumpless transfer when the downstream mode is changed to remote.

5.3 Component FBs

A process control application is built out of application FBs as defined above. In addition, the application can include component FBs combined in an application specific way and encapsulated by FBs of composite FB type as described 4.1.1. The exception handling and status handling is technology specific and is part of the component FB definitions.

5.4 Technology Block

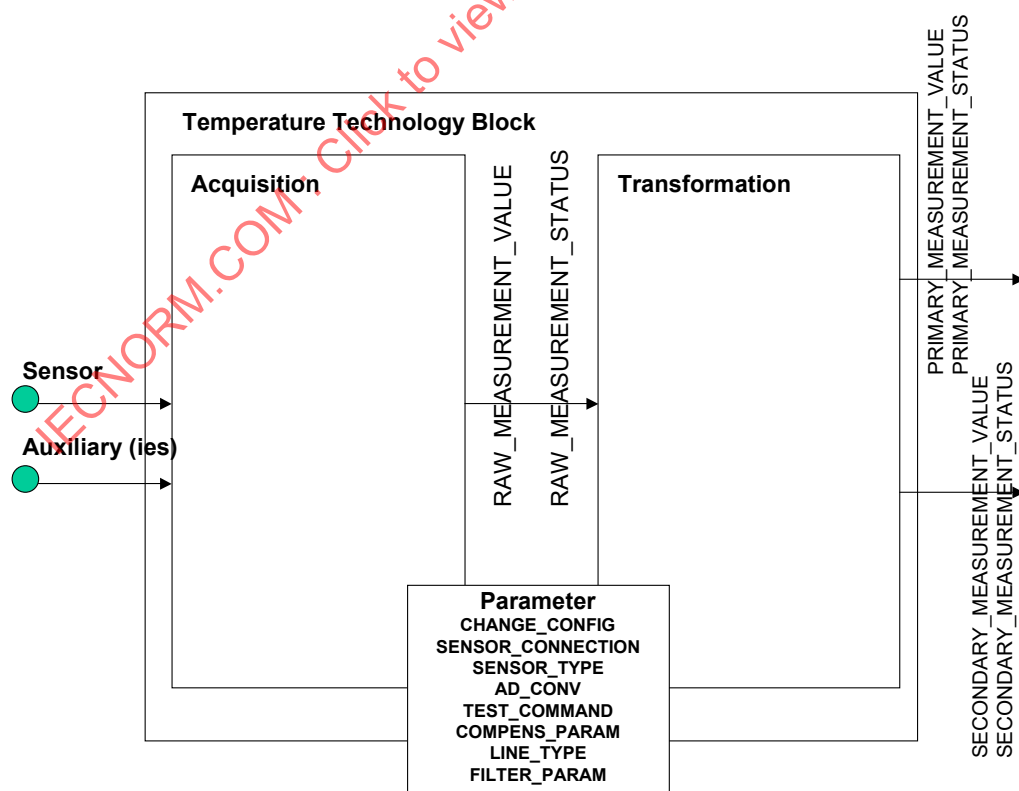
5.4.1 Temperature Technology Block

5.4.1.1 Temperature Technology Block overview

The algorithms of the Temperature Technology Block are summarized below:

- sensor connection
- channel range/scaling
- AD conversion
- test
- diagnosis
- cold junction compensation
- linearization
- filtering
- initialisation

The algorithms are encapsulated in the Acquisition and Transformation part of the Technology Block (see Figure 19).



IEC

Figure 19 – Temperature Technology Block

5.4.1.2 Temperature Acquisition Functions

5.4.1.2.1 Sensor connection

The process signal is connected directly to the interface module.

There is a possibility to connect the thermo resistance with 2, 3 or 4 wires. Compensation is chosen by the parameter `SENSOR_CONNECTION`.

This algorithm checks the sensor link and signals a fault if there is a short-circuit or an open circuit. The wiring check is enabled/disabled via the configuration (`CHAN_CONFIG`).

5.4.1.2.2 Channel range

This algorithm selects the sensor type which is connected to the device. According to the configuration (`SENSOR_TYPE`), it is necessary to differentiate between:

- electrical range (± 10 V, 0 V to 10 V, 0 V to 5 V, 1 V to 5 V, 0 mA to 20 mA or 4 mA to 20 mA),
- thermocouple,
- temperature probes.

Table 4 gives an example of several types of sensor.

IECNORM.COM : Click to view the full PDF of IEC 61804-2:2018

Table 4 – Example of temperature sensors of Sensor_Type

Symbol	Description
Type B	Platinum, 30% Rhodium/ Platinum, 6% Rhodium
Type C	Tungsten, 5% Rhenium/Tungsten, 26% Rhenium
Type D	Tungsten, 3% Rhenium/Tungsten, 25% Rhenium
Type E	Chromel/Constantan
Type G	Tungsten/Tungsten, 26% Rhenium
Type J	Iron/Constantan
Type K	Chromel/Alumel
Type L	Platinel 5355/Platinel 7674
Type N	Nicrosil/Nisil
Type R	Platinum, 13 % Rhodium/Platinum
Type S	Platinum, 10 % Rhodium/Platinum
Type T	Copper/Constantan
Pt50	Platinum 50 Ω
Pt100	Platinum 100 Ω
Pt200	Platinum 200 Ω
Pt500	Platinum 500 Ω
Pt1000	Platinum 1 000 Ω
Ni10	Nichel 10 Ω
Ni50	Nichel 50 Ω
Ni100	Nichel 100 Ω
Ni120	Nichel 120 Ω
Cu10	Copper 10 Ω
Cu25	Copper 25 Ω
Cu100	Copper 100 Ω
NOTE The temperature range can be the default range of the selected thermocouple or temperature probe defined in tenths of degree (e.g. –600 to +11 000 tenths of °C for a Ni 1 000 probe).	

5.4.1.2.3 AD conversion

Digitalisation of input measurement analogue signal, according to the parameter set during configuration (ADCONV).

5.4.1.2.4 Test

Many test strategies are possible for example, switching the input from the sensor to a reference signal and checking the output of the Technology Block against the expected value, in order to assess correct operation.

Test results then contribute to the status information processing.

During tests it is recommended that the output of the connected AB maintains the previous value or other "best estimate" of the true current value.

This algorithm is started by the TEST_COMMAND parameter which is optional and its implementation is manufacturer specific.

5.4.1.2.5 Diagnosis

This algorithm is device specific to assess the internal performance of the related channel. The results of internal assessments are used to construct the generic measurement status information. Technology specific report mechanisms provide the corresponding status information, for example to maintenance planning.

5.4.1.3 Initialisation

The initialisation algorithm is applied to this block and described in 5.6.3.

5.4.1.4 Temperature Transformation Functions

5.4.1.4.1 Cold junction compensation

The voltage generated from the thermocouple is compensated with a reference junction value.

COMPENS_PARAM defines the type of compensation. The type of cold junction compensation is either Internal or External (Internal: the device itself measures the Reference junction temperature via an internal mounted sensor).

5.4.1.4.2 Linearisation

Thermocouple and RTD values are linearised and compensated internally. The linearisation is done according to IEC 60584-1 and the DIN 43710 reference standard for the thermocouple curve. Optionally, the manufacturer may offer an additional user defined linearisation. LINE_TYPE defines the linearisation curve coefficients.

5.4.1.4.3 Filtering

A filtering is performed on the measure linearised and compensated.

With the FILTER_PARAM the filter efficiency shall be selected, for example 1 sec, 2 sec, 5 sec.

5.4.1.4.4 Initialisation

The initialisation algorithm is applied to this block and described in 5.6.3.

5.4.2 Pressure Technology Block

5.4.2.1 Pressure Technology Block overview

The algorithms of the Pressure Technology Block are summarised below:

- a) sensor connection
- b) channel range/scaling
- c) sensor calibration
- d) test
- e) diagnosis
- f) linearisation
- g) filtering
- h) temperature compensation
- i) initialisation

The algorithms are encapsulated in the Acquisition and Transformation part of the Technology Block (see Figure 20).

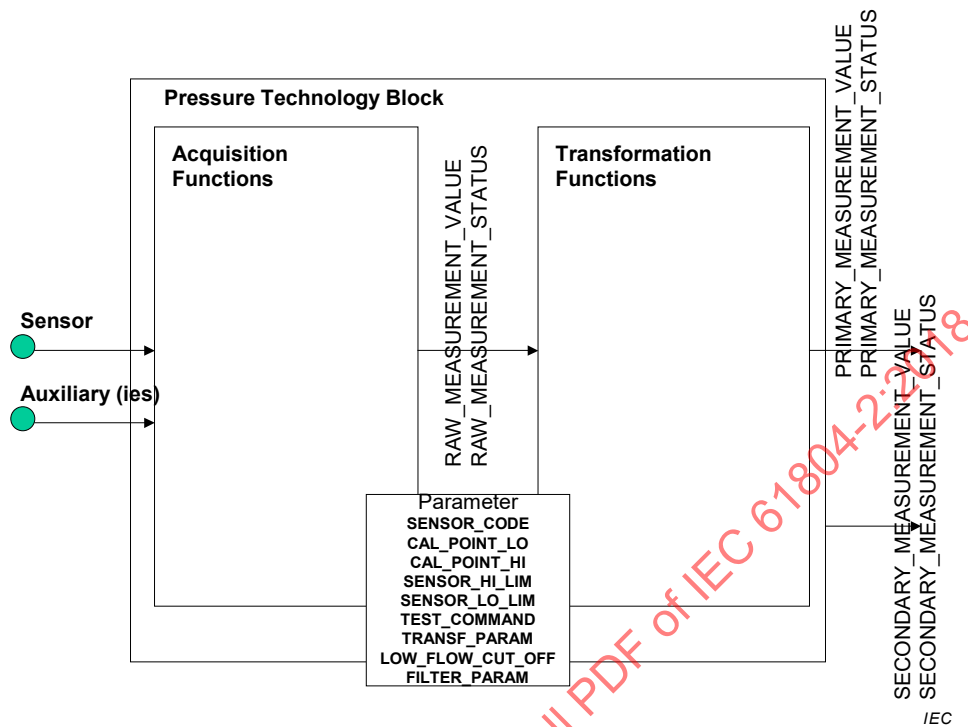


Figure 20 – Pressure Technology Block

5.4.2.2 Pressure Acquisition Functions

5.4.2.2.1 Sensor connection

There is a possibility to connect the different pressure or differential pressure sensors to the transmitter. Compensation is chosen by the parameter `SENSOR_CODE` depending on the measurement principle.

5.4.2.2.2 Channel range scaling

This algorithm selects the display format in which the measurements are supplied to the user. The `SENSOR_HI_LIM` and `SENSOR_LO_LIM` parameter define the maximum and minimum values the sensor is capable of indicating.

5.4.2.2.3 Sensor calibration

The calibration process is used to match the channel value combined with the applied input. The calibration of the sensor itself is not changed, because that is a factory procedure. Four parameters are defined to configure this process: `CAL_POINT_HI`, `CAL_POINT_LO`, `SENSOR_HI_LIM` and `SENSOR_LO_LIM`. The `CAL_*` parameters define the highest and lowest calibrated values for this sensor.

5.4.2.2.4 Test

Many test strategies are possible, for example, switching the input from the sensor to a reference signal and checking the output of the Technology Block against the expected value, in order to assess correct operation.

Test results then contribute to the status information processing.

During tests it is recommended that the output of the Application Block maintains the previous value or other "best estimate" of the true current value.

This algorithm is started by the TEST_COMMAND parameter which is optional and its implementation is manufacturer specific.

5.4.2.2.5 Diagnosis

This algorithm is device specific to assess the internal performance of the related channel. The results of internal assessments are used to construct the generic measurement status information. Technology specific report mechanisms provide corresponding status information for example to maintenance planning.

5.4.2.2.6 Initialisation

The initialisation algorithm is applied to this block and described in 5.6.3.

5.4.2.3 Pressure Transformation Functions

5.4.2.3.1 Linearisation

Pressure sensor values are linearised and compensated internally. Generally a linearisation is realised in the factory to meet initial accuracy. Additional linearisation is done by using the TRANSF_PARAM parameter if flow or level measurement is applied with the pressure transmitter. The square root function is chosen as well as user defined linearisation tables. LOW_FLOW_CUT_OFF parameter determines the starting point for flow measurement at the lowest level.

5.4.2.3.2 Filtering

Filter values are selected (no filter, low level of filtering, medium level of filtering, high level of filtering) in the corresponding FILTER_PARAM. The filtering is done on the measure that is linearised and compensated.

5.4.2.3.3 Temperature compensation

Usually the pressure of a liquid or gas is dependent on its temperature. The measured pressure value is compensated with the corresponding temperature using this algorithm.

5.4.2.3.4 Initialisation

The initialisation algorithm is applied to this block and described in 5.6.3.

5.4.3 Modulating Actuation Technology Block

5.4.3.1 Modulating Actuation Technology Block overview

The elementary algorithms and parameters of modulated actuation are summarised below. Valves as well as motor drives are represented because the functions do not include technology details:

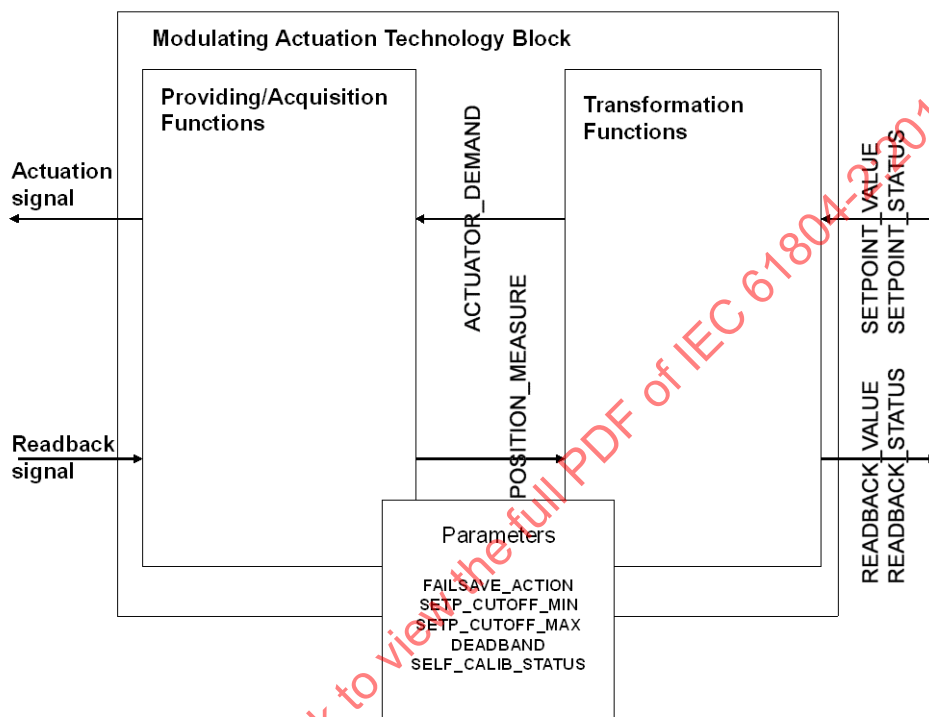
- a) amplification
- b) readback measurement
- c) output limits
- d) self-calibration
- e) fail safe
- f) diagnosis
- g) test

h) initialisation

A graphical representation with the inputs (left), the outputs (right) and the parameters (bottom) is used.

NOTE The Inputs and Outputs are logical connections and they do not always represent the signal flow from the view of the automation application (process control) or the process itself.

The algorithms are encapsulated in the Acquisition and Transformation part of the Technology Block (see Figure 21).



IEC

Figure 21. – Modulating Actuation Technology Block

5.4.3.2 Providing and Acquisition Functions

5.4.3.2.1 Amplification

The block provides as an output a signal (actuation signal) from the ACTUATOR_DEMAND for the final element (e.g. a valve or motor). The final element modifies the process in response to this actuation demand output sent from the AB to the Technology Block (SETPOINT_VALUE).

5.4.3.2.2 Readback measurement

The block measures the actual readback signal from the final element and converts it to the transfer part of the Technology Block (POSITION_MEASURE).

5.4.3.2.3 Fail safe

The fail safe algorithm is described in 5.6.4.

5.4.3.2.4 Test

Many test strategies are possible for example, driving the actuator in a defined range and checking the measured values, in order to assess correct operation.

Test results then contribute to the status information processing.

During tests it is recommended that the output of the Modulating Actuation Technology Block reflects the actual actions.

This algorithm is started by the TEST_COMMAND parameter, which is optional, and its implementation is manufacturer specific.

5.4.3.2.5 Diagnosis

This algorithm is device specific to assess the internal performance of the related channel. The results of internal assessments are used to construct the generic measurement status information. Technology specific report mechanisms provide additional status information, for example to maintenance planning.

5.4.3.2.6 Initialisation

The initialisation algorithm is applied to this block and described in 5.6.3.

5.4.3.3 Transformation Functions

5.4.3.3.1 Output limits

When the setpoint goes below the defined SETP_CUTOFF_MIN limit, the output (actuation signal) goes to the minimum value.

With an electro-pneumatic actuator, this is done by venting/filling the actuator. With a variable-speed actuator, the actuator goes to the stopped condition.

When the setpoint goes above the defined SETP_CUTOFF_MAX limit, the output (actuation signal) goes to the maximum value.

With an electro-pneumatic actuator, this is done by totally ventilating/filling the actuator. With a variable speed actuator, the actuator goes to the full value condition.

5.4.3.3.2 Self-calibration

The procedure of self-calibrating is manufacturer specific. The following status information (SELF_CALIB_STATUS) are typical examples:

Undetermined, Aborted, Error in mechanical system, Timeout, Aborted by means of Emergency override, Zero point error, Success.

5.4.3.3.3 Deadband

There is a deadband in which the changes of SETPOINT_VALUE does not affect the actuation signal. This is indicated in the DEADBAND parameter.

5.4.3.3.4 Initialisation

The initialisation algorithm is applied to this block and described in 5.6.3.

5.4.4 On/Off Actuation Technology Block

5.4.4.1 On/Off Actuation Technology Block overview

Figure 22 shows the On/Off actuation Technology Block.

The elementary algorithms and parameters of the device are summarized below:

- a) signal conversion
- b) signal detection
- c) self calibrating
- d) count limits
- e) fail safe
- f) test
- g) diagnosis
- h) initialisation

A graphical representation with the inputs (left), the outputs (right) and the parameters (bottom) is used.

NOTE Both simple and complex implementations are available using various technologies.

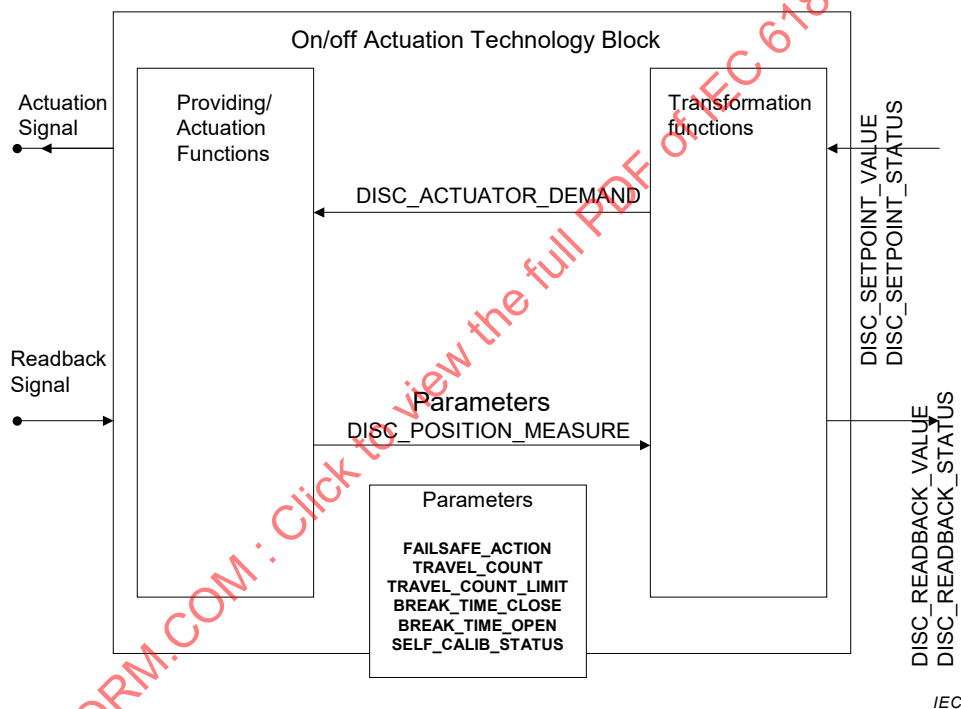


Figure 22 – On/Off Actuation Technology Block

5.4.4.1.1 Signal conversion

The block provides as an output a two-state signal (discrete actuation signal) from the DISC_ACTUATOR_DEMAND to the final element (e.g. a relay or valve). The final element modifies the process in response to this discrete actuation demand output sent from the application block to the Technology Block (DISC_SETPOINT_VALUE).

5.4.4.1.2 Signal detection

The block receives the actual demanded state (e.g. discrete position signal) from the final element and converts it to the transfer part of the Technology Block (DISC_POSITION_MEASURE).

5.4.4.1.3 Break time

The actuator needs a certain period of time to switch. The break time function in the Technology Block provides an adjustable dead time between a new DISC_SETPOINT_VALUE value and the change of the DISC_ACTUATOR_DEMAND by the parameters BREAK_TIME_CLOSE and BREAK_TIME_OPEN.

5.4.4.1.4 Fail safe

The fail safe algorithm is described in 5.6.4.

5.4.4.1.5 Test

Many test strategies are possible, for example, switching the actuator on and off and checking actual reached positions, in order to assess correct operation.

Test results then contribute to the status information processing.

During tests it is recommended that the output of the AB reflects the actual actions.

This algorithm is started by the TEST_COMMAND parameter, which is optional, and its implementation is manufacturer specific.

5.4.4.1.6 Diagnosis

This algorithm is device specific to assess internal performance of the related channel. The results of internal assessments are used to construct the generic measurement status information. Technology specific report mechanisms provide the corresponding status information, for example to maintenance planning.

5.4.4.1.7 Initialisation

The initialisation algorithm is applied to this block and described in 5.6.3.

5.4.4.2 Transformation Functions

5.4.4.2.1 Count limits

This algorithm counts the numbers of cycles (TRAVEL_COUNT) of an actuator. A cycle is two successive transitions from one state to the other and back to the first. The detection of transitions and the count function is manufacturer specific. The count is often used internally to assist diagnosis and the TRAVEL_COUNT_LIMIT can trigger a suitable maintenance report.

5.4.4.2.2 Self-calibrating

The procedure of self-calibrating is manufacturer specific. The following status information (SELF_CALIB_STATUS) is recommended:

Undetermined, Aborted, Error in mechanical system, Timeout, Aborted by means of Emergency override, Zero point error, Success.

5.4.4.2.3 Initialisation

The initialisation algorithm is applied to this block and described in 5.6.3.

5.5 Device (Resource) Block

5.5.1 Identification

The device provides documentation information electronically to assist the user of a device (in particular a Control operator/algorithm) in checking the device type and revision. For the different phases of the device life cycle (design, commissioning, documentation (on-line)) it is absolutely necessary to have an unambiguous identification of the devices. Therefore the following parameters are supported

- DEVICE_VENDOR
- DEVICE_MODEL
- DEVICE_REVISION
- DEVICE_SER_NO for identification of multiple devices of the same type is optional.

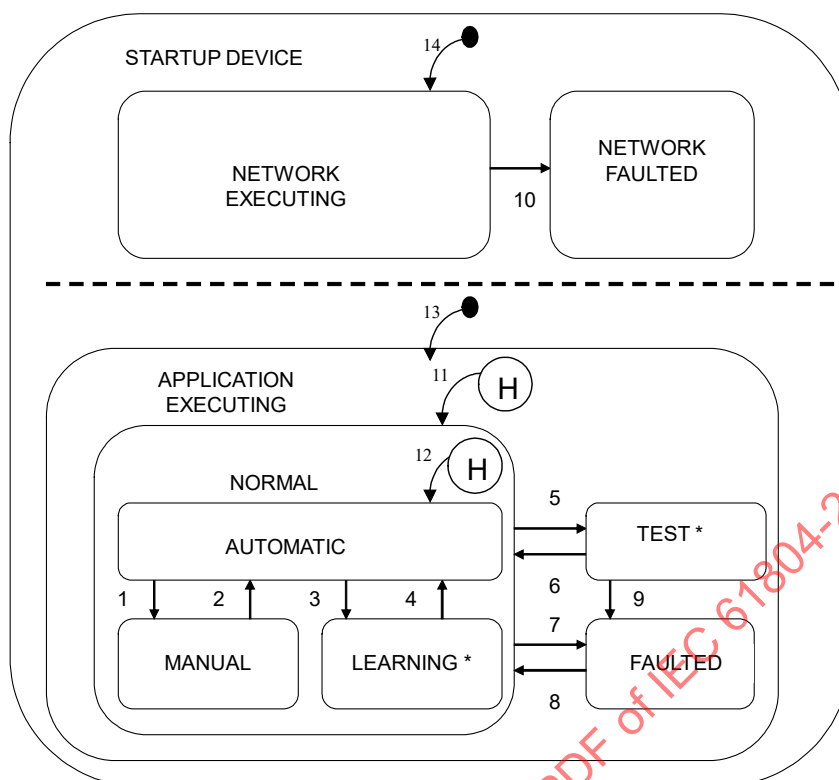
5.5.2 Device state

The device synthesized status is devoted to assist the user of a device (in particular a Control operator/algorithm) in assessing its remaining capabilities and to adapt accordingly its strategies. This status is called DEVICE_STATUS.

As an example the following state models are provided to aid in understanding the corresponding device behaviour. The behaviour is described using a state table (see Table 5), a Harel state model (see Figure 23) and a transition table (see Table 6).

Table 5 – Device status state table

State	Description
NETWORK EXECUTING	Initial state of the device. The device is capable of responding to network commands for normal operation. The processor is running.
NETWORK FAULTED	The normal operation of the device is not available, because the device functionality is not accessible through the network.
APPLICATION EXECUTING	Initial state of the application. The device is available for operation (normal, test and fault detection).
NORMAL	The device is available for normal operation including the reporting of detected diagnostics and process alarms.
AUTOMATIC	The device processes the value from the transmitter according to all algorithms (scaling, filtering, limit checks, engineering unit conversion).
MANUAL	This state is used to force the main measurement to an assigned value
LEARNING	The device is performing an automatic adjustment of some parameters (e.g. Functional Threshold). This state is optional, it depends on the device.
FAULTED	The device is not available for normal operation. Within this state, additional sub categories of fault status may be reported. Examples are: diagnosis, event time stamp and maintenance priority.
TEST	The device is performing the test. This state is optional, it depends on the device.



* Optional

NOTE The numbers in the figure are explained in Table 6.

Figure 23 – Harel state chart

Table 6 – Device status transition table

Transition	From state	To State	Description
1	AUTOMATIC	MANUAL	A control command with the operating mode "Manual" is received by the device
2	MANUAL	AUTOMATIC	A control command with the operating mode "Automatic" is received by the device
3	AUTOMATIC	LEARNING	Not mandatory for all devices
4	LEARNING	AUTOMATIC	Not mandatory for all devices
5	NORMAL	TEST	Not mandatory for all devices
6	TEST	NORMAL	Not mandatory for all devices
7	NORMAL	FAULTED	A fault is detected
8	FAULTED	NORMAL	Fault reset
9	TEST	FAULTED	Not mandatory for all devices
10	NETWORK EXECUTING	NETWORK FAULTED	Communication port failed, processor failed
11	APPLICATION EXECUTING	NORMAL or state before restart	Initialisation of application to provide diagnosis and alarm information
12	NORMAL	AUTOMATIC or state before restart	Application run now in AUTOMATIC or states LEARNING or MANUAL and recovering the states according to the device data
13	Power off	APPLICATION EXECUTING	Initialisation to device application
14	Power off	NETWORK EXECUTING	Initialisation of the communication

5.5.3 Message

The device provides memory space to store user information arising during the life time of the device. The user, for example service staff or maintenance operator, write textual information in this parameter. For example it can be used for documentation purposes.

5.5.4 Initialisation

The initialisation algorithm is applied to this block and described in 5.6.3.

5.6 Algorithms common to all blocks

5.6.1 Data Input/Data Output status

The synthesized output status is determined by a block based on its algorithm execution results, which for example consider block inputs, block parameters, Diagnosis and Device State. It is provided to assist the user of a device or a measurement (in particular a Control operator/algorithm) when assessing its current performance capabilities and to adapt accordingly its strategies.

For example, input status is used by some FBs to change the MODE and execute alternate algorithms.

5.6.2 Validity

Each FB can optionally offer a validity function which provides a more detailed information about the quality of the measurement than the one expressed with the input/output status. In this case the FB has to record in the parameter list the relevant contained parameters (e.g. Uncertainty_Value and Uncertainty_Status). Validity functions have to be separated from status functions.

NOTE The distribution of the validity information can be done in an acyclic or in a cyclic way.

5.6.3 Restart Initialisation

Many process control applications require control strategies to take pre-defined initialisation actions when restarting components and devices in the process control system. This capability is commonly called a Restart Initialisation function. The restart initialisation actions are highly dependent on the control system technologies, and are often configured uniquely for the particular process application.

The following optional behaviours may apply:

- first activation of a new device,
- cold restart of a device (extended power failure),
- warm restart of a device (short power failure),
- return of a device from fail-safe.

NOTE 1 This can be implemented as part of device management, FB management, mode or application program.

For example output Technology Blocks include defined default values for input (channel) parameters and the associated block functions to drive the output hardware to its un-powered state when the Technology Block input (channel) is not configured (i.e. the Technology Block input (channel) is not connected to an FB output).

NOTE 2 The physical device is represented by the Device Block. The initialisation of the Device Block is the visible initialisation of the physical device.

5.6.4 Fail-safe

In many process control applications, it is critical for control strategies and devices to take safe pre-defined actions in the event of a failure of strategies, components, or devices in the process control system. This capability is commonly called a fail-safe function. The following optional behaviours may apply.

- A resource fail-safe command, when set, will cause appropriate technology and FBs within the resource to execute their defined fail-safe actions.
- Also, a resource fail-safe disable command, when set, will disable all fail-safe actions within the resource.
- Initiating a fail-safe command on detection of lack of communication with other devices or resources within the system.

The particular pre-defined actions taken are highly dependent on the process application. The precise implementations of fail-safe functions are highly dependent on the control system technologies.

The following examples illustrate the use of fail safe.

- Resource blocks in some technologies and applications include parameters and functions to provide a fail-safe action of the device hardware and blocks associated with the resource. Fail-safe disable is enabled with a hardware jumper in this example profile. When the fail-safe disable is active, the resource sends fail-safe disabled notifications to other appropriate resources in the system.

- Technology Blocks in some technologies and applications include parameters and functions to provide a fail-safe action of the device hardware associated with the block.
- A Technology Block in one profile will execute pre-defined fail-safe actions on detection of bad channel or hardware values. The Technology Block will also execute pre-defined fail-safe actions on receipt of a resource block fail-safe command.
- Control, calculation, and output FBs in some technologies and applications include parameters and functions to provide a fail-safe action of the control functionality within the block.
- A FB in one technology profile will execute pre-defined fail-safe actions on detection of bad input, output, or transfer values. The FB will also execute pre-defined fail-safe actions on receipt of a resource fail-safe command. When fail-safe is active, these FBs send fail-safe notifications to appropriate resources in the system via their own resource.

5.6.5 Remote Cascade Initialisation

A control block Out value may connect to the Remote Setpoint of an output or control block. The downstream block will set its setpoint to this remote setpoint input value when the block mode parameter is set to the remote cascade. To prevent the block setpoint from changing when the mode transitions from auto or manual to remote cascade, the output of the block providing the remote setpoint value shall match the setpoint. To allow this co-ordination, the Readback Out value and status of the downstream block are connected to the Readback value and status of the upper block. The Readback Out value shall reflect the block Setpoint or In_Value value. The Readback Out status reflects the mode and initialisation state. Similarly, the control block Out status should reflect the action taken by the block based on its Readback input.

When the control block Readback status indicates that the downstream block is not in Cascade mode, then its Out value will be set to the Readback value. When the mode of the downstream block transitions to Remote Cascade, then its Readback status should indicate that initialisation is required. Only after the control block has taken action on this initialisation request should its Out status indicate that initialisation is complete. Once the Remote Input status reflects that initialisation is complete, then the block shall set the Setpoint to the Remote Setpoint value and provide a status indicating normal operation in its Readback Out status.

6 FB Environment

The FB Environment is composed of additional object and block types to the types defined in 4.1.1. These object and blocks are:

- Link Block;
- Alert Block;
- Trend Block.

NOTE The FB Environment is very platform and technology dependent.

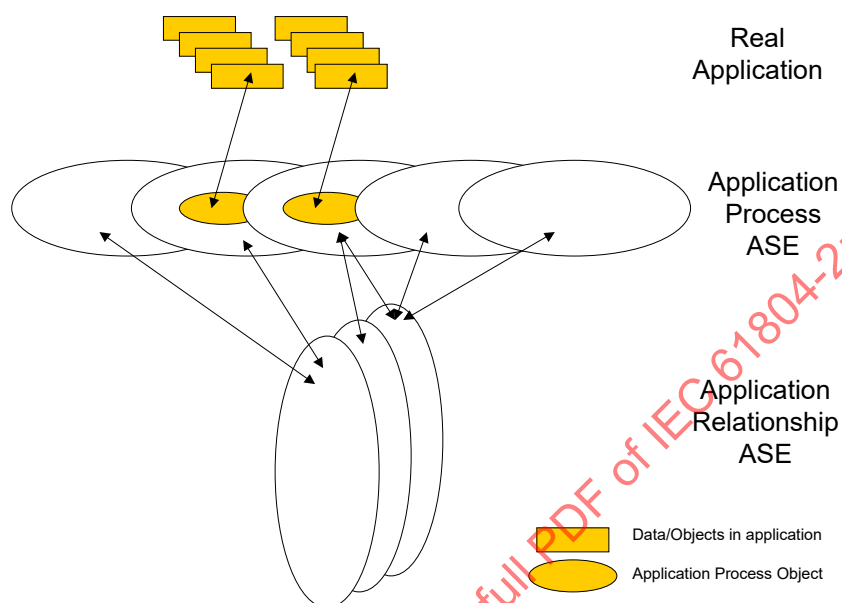
7 Mapping to System Management

The mapping to System Management is an open issue regarding IEC 61158 (all parts). Therefore, it is not done within this specification.

NOTE Fieldbus specific solutions can define their own mapping without changing the definition of this document.

8 Mapping to Communication

To provide a systematic mapping to communication networks the ISO OSI Reference Model has to be used. Regarding the application representation the model shown in Figure 24 is used.

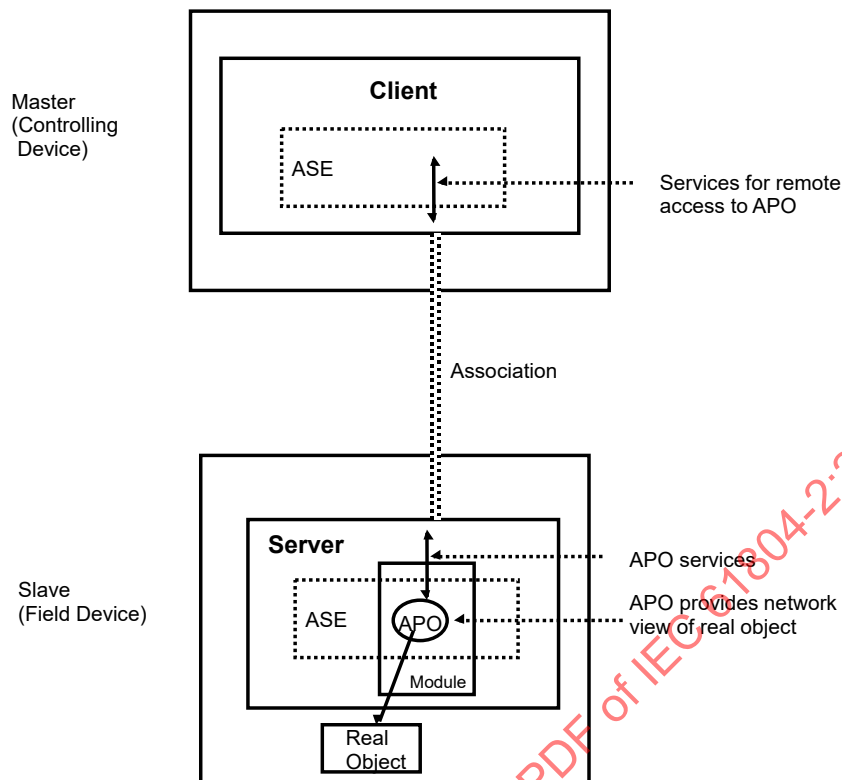


IEC

Figure 24 – Application structure of ISO OSI Reference Model

The real application data inputs, data outputs and parameters and object are represented by so called Application Process Objects (APOs) which are managed by so called Application Process Application Service Entities (AP ASEs) (see ISO OSI Reference Model). These AP ASEs communicate via so called Application Relationship ASEs.

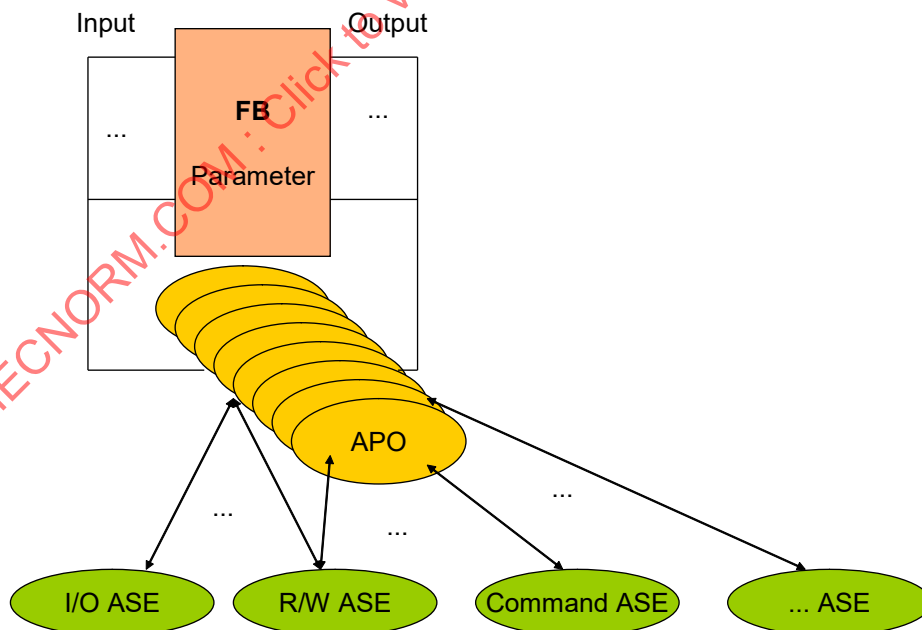
For example a client server relationship is modelled as shown in Figure 25.



IEC

Figure 25 – Client/Server relationship in terms of OSI Reference Model

IEC 61158 (all parts) uses exactly this model. Therefore the mapping has to use the same.



IEC

Figure 26 – Mapping of IEC 61804 FBs to APOs

There are various types of ASE in IEC 61158 (all parts). Different communication systems define and name their ASE types. The ASE types shown in Figure 26 are named general and represent cyclic communication (I/O ASE), client/server communication such as read and

write (R/W ASE) and command oriented communication with bidirectional data transport (Command ASE).

The proposed mapping rules are as follows.

- Inputs, Outputs, parameter and the blocks themselves should be mapped to corresponding APOs.
- For each APO the allowed ASEs have to be defined. More than one ASE per APO is possible.
- Multiple APO data can be transferred by one Command ASE service.

The mapping to a fieldbus according to IEC 61158 (all parts) or any other communication system has to be done by the appropriate expert group of the communication system.

IECNORM.COM : Click to view the full PDF of IEC 61804-2:2018

Annex A (informative)

Parameter description

Annex A describes the parameters of the different FBs, see Table A.1.

NOTE The acronyms have the following meaning: M = mandatory, O = optional, C = conditional, R = read, R/W = read/write.

Table A.1 – Parameter description

Parameter name	Description	Data type	User Access Read /Write	Class M/O/C
Analog Input FB				
MEASUREMENT_VALUE	Main measurement value as a result of the Measurement FB	Numeric	R	M
MEASUREMENT_STATUS	Status of the MEASUREMENT_VALUE	List of Boolean	R	M
PRIMARY_MEASUREMENT_VALUE	Primary measurement value as a result of the measurement Technology Block	Numeric	R	M
PRIMARY_MEASUREMENT_STATUS	Status of the PRIMARY_MEASUREMENT_VALUE parameter	List of Boolean	R	M
UNITS	Units of the main measurement value	Enumerated	R/W	O
HIGH_ALARM_LIMIT	Value for upper limit of alarms	Numeric	R/W	O
LOW_ALARM_LIMIT	Value for lower limit of alarms	Numeric	R/W	O
MODE	Operation mode of the block (e.g. Manual, Automatic, Remote Cascade)	Enumerated	R/W	O
CHANNEL	Logical reference to the Technology Block measurement	Enumerated	R/W	O
SIMULATE	Used to carry out internal tests	Enumerated	R/W	O
Analog Output FB				
REMOTE_SETPOINT_VALUE	Remote Setpoint from the output of an upstream application block	Numeric	R/W	M
REMOTE_SETPOINT_STATUS	Status of the REMOTE_SETPOINT_VALUE parameter	List of Boolean	R/W	M
OUT_VALUE	Primary output value of the analog actuation output function	Numeric	R/W	M
OUT_STATUS	Status of the OUT_VALUE parameter	List of Boolean	R/W	M
READBACK_VALUE	Feedback of the downstream Technology Block readback output value	Numeric	R/W	M
READBACK_STATUS	Status of the READBACK_VALUE parameter	List of Boolean	R	M
READBACK_OUT_VALUE	Feedback to the upstream application block readback value	Numeric	R/W	M
READBACK_OUT_STATUS	Status of the READBACK_OUT_VALUE parameter	List of Boolean	R/W	M
UNITS	Unit selection	Enumerated	R/W	O
SP_HI_LIM	Setpoint value high limit	Numeric	R/W	O

Parameter name	Description	Data type	User Access Read /Write	Class M/O/C
SP_LO_LIM	Setpoint value low limit	Numeric	R/W	O
MODE	Operation mode of the block (e.g. Manual, Automatic, Remote Cascade)	Enumerated	R/W	O
CHANNEL	Reference to the Technology Block actuator	Enumerated	R/W	O
SIMULATE	Used to carry out internal tests	Enumerated	R/W	O
Discrete Input FB				
DISC_MEASUREMENT_VALUE	Discrete input measurement value	Boolean	R	M
DISC_MEASUREMENT_STATUS	Status of the DISC_MEASUREMENT_VALUE parameter	List of Boolean	R	M
DISC_PRIMARY_MEASUREMENT_VALUE	Primary discrete measurement value as a result of the discrete input Technology Block	Boolean	R	M
DISC_PRIMARY_MEASUREMENT_STATUS	Status of the DISC_PRIMARY_MEASUREMENT_VALUE parameter	List of Boolean	R	M
CONVERT	Boolean invert of the discrete primary value or of the sensor value	Boolean	R/W	O
MODE	Operation mode of the block (e.g. Manual, Automatic, Remote Cascade)	Enumerated	R/W	O
CHANNEL	Reference to the Technology Block input	Enumerated	R/W	O
SIMULATE	Used to carry out internal tests	Enumerated	R/W	O
Discrete Output FB				
DISC_REMOTE_SETPOINT_VALUE	Discrete Remote Setpoint from the output of an upstream application block	Boolean	R/W	M
DISC_REMOTE_SETPOINT_STATUS	Status of DISC_REMOTE_SETPOINT_VALUE parameter	List of Boolean	R/W	M
DISC_OUT_VALUE	Primary output value of the on/off actuation output function	Numeric	R/W	M
DISC_OUT_STATUS	Status of the DISC_OUT_VALUE parameter	List of Boolean	R	M
DISC_READBACK_VALUE	Readback of the discrete readback output from a downstream Technology Block	Boolean	R/W	M
DISC_READBACK_STATUS	Status of the DISC_READBACK_VALUE parameter	List of Boolean	R/W	M
DISC_READBACK_OUT_VALUE	Feedback to the upstream application block discrete readback value	Numeric	R/W	M
DISC_READBACK_OUT_STATUS	Status of the DISC_READBACK_OUT_VALUE parameter	List of Boolean	R/W	M
MODE	Operation mode of the block (e.g. Manual, Automatic, Remote Cascade)	Enumerated	R/W	O
CHANNEL	Reference to the Technology Block of the actuator	Enumerated	R/W	O
SIMULATE	Used to carry out internal tests of the actuator	Enumerated	R/W	O

Parameter name	Description	Data type	User Access Read /Write	Class M/O/C
Calculation FB				
FOLLOW	Forces the output value to track a block input	Numeric	R/W	O
IN_VALUE	Primary input value to the calculation	Numeric	R	M
IN_STATUS	Status of the primary input value	List of Boolean	R	M
OUT_VALUE	Primary output value of the calculation	Numeric	R/W	M
OUT_STATUS	Status of the primary output value	List of Boolean	R	M
READBACK_VALUE	Feedback of the downstream block readback output value	Numeric	R/W	M
READBACK_STATUS	Status of the readback value	List of Boolean	R/W	M
READBACK_OUT_VALUE	Feedback to the upstream block readback value	Numeric	R/W	M
READBACK_OUT_STATUS	Status of the readback output value	List of Boolean	R/W	M
Control FB				
IN_VALUE	Primary input measurement	Numeric	R	M
IN_STATUS	Status of primary input measurement	List of Boolean	R	M
OUT_VALUE	Primary output value of the control function	Numeric	R/W	M
OUT_STATUS	Status of the OUT_VALUE parameter	List of Boolean	R	M
READBACK_VALUE	Feedback of the downstream block readback output value	Numeric	R/W	M
READBACK_STATUS	Status of the READBACK_VALUE parameter	Numeric	R/W	M
READBACK_OUT_VALUE	Feedback to the upstream block readback value	Numeric	R/W	M
READBACK_OUT_STATUS	Status of the READBACK_OUT_VALUE parameter	Numeric	R/W	M
REMOTE_SETPOINT_VALUE	Remote target value for a process output measurement from an upstream application block	Numeric	R/W	M
REMOTE_SETPOINT_STATUS	Status of the REMOTE_SETPOINT_VALUE parameter	List of Boolean	R	M
SETPOINT	Local target value for a process output measurement	Numeric	R/W	M
SP_HI_LIM	Upper limit for Setpoint value	Numeric	R/W	O
SP_LO_LIM	Lower limit for Setpoint value	Numeric	R/W	O
ALARM_HI	Upper alarm limit for the primary input value	Numeric	R/W	O
ALARM_LO	Lower alarm limit for the primary input value	Numeric	R/W	O
MODE	Operation mode of the block, (e.g. Manual, Automatic, Remote Cascade)	Enumerated	R/W	O

Parameter name	Description	Data type	User Access Read /Write	Class M/O/C
Temperature Technology Block				
RAW_MEASUREMENT_VALUE	Raw measurement value as result of measurement acquisition	Numeric	R	M
RAW_MEASUREMENT_STATUS	Status of RAW_MEASUREMENT_VALUE parameter	List of Boolean	R	M
PRIMARY_MEASUREMENT_VALUE	Primary measurement value as result of the transformation function	Numeric	R	M
PRIMARY_MEASUREMENT_STATUS	Status of PRIMARY_MEASUREMENT_VALUE parameter	List of Boolean	R	M
SECONDARY_MEASUREMENT_VALUE	Secondary measurement value(s) as result of the transformation function	Numeric	R	O
SECONDARY_MEASUREMENT_STATUS	Status of the corresponding SECONDARY_MEASUREMENT_VALUE parameters	List of Boolean	R	O
CHANGE_CONFIG	Wiring check	Enumerated	R	O
SENSOR_CONNECTION	2, 3 or 4 wires for RTD measurement	Enumerated	R/W	O
SENSOR_TYPE	Thermocouple, thermoresistance (RTD), low voltage i.e. in the range ± 25 mV or ± 100 mV	Enumerated	R/W	M
AD_CONV	A/D Conversion parameters	Numeric	R/W	O
TEST_COMMAND	Starts test procedure to check the sensor	Enumerated	R/W	O
COMPENS_PARAM	Cold junction compensation parameters	Numeric	R/W	O
LINE_TYPE	Linearisation curve coefficients, Supplementary Measure parameters	Enumerated	R/W	O
FILTER_PARAM	Filter parameters (e.g. Anti-aliasing pre-filtering)	Enumerated	R/W	O
Pressure Technology Block				
RAW_MEASUREMENT_VALUE	Raw measurement value as result of measurement acquisition	Numeric	R	M
RAW_MEASUREMENT_STATUS	Status of RAW_MEASUREMENT_VALUE parameter	List of Boolean	R	M
PRIMARY_MEASUREMENT_VALUE	Primary measurement value as result of the transformation function	Numeric	R	M
PRIMARY_MEASUREMENT_STATUS	Status of PRIMARY_MEASUREMENT_VALUE parameter	List of Boolean	R	M
SECONDARY_MEASUREMENT_VALUE	Secondary measurement value(s) as result of the transformation function	Numeric	R	O
SECONDARY_MEASUREMENT_STATUS(es)	Status of SECONDARY_MEASUREMENT_VALUE parameters	List of Boolean	R	O
SENSOR-CODE	Type of Sensor (it identifies the transformation curve to be used)	Enumerated	R/W	O
CAL_POINT_LO	This parameter contains the lowest calibrated value, which is put to the sensor and transfers this point as LOW to the transmitter	Numeric	R/W	O

Parameter name	Description	Data type	User Access Read /Write	Class M/O/C
CAL_POINT_HI	This parameter contains the highest calibrated value, which is put to the sensor and transfer this point as HIGH to the transmitter	Numeric	R/W	O
SENSOR_HI_LIM	Physical upper limit of the sensor	Numeric	R/W	O
SENSOR_LO_LIM	Physical lower limit of the sensor	Numeric	R/W	O
TEST_COMMAND	Starts test procedure to check the sensor	Enumerated	R/W	O
TRANSF_PARAM	Linearisation curve coefficients and Supplementary Measure parameters	Numeric	R/W	O
LOW_FLOW_CUT_OFF	Lowest flow value which is determined as the minimum value	Numeric	R/W	O
FILTER_PARAM	Filter parameters (e.g. anti-aliasing pre-filtering)	Enumerated	R/W	O
Modulating Actuation Technology Block				
SETPOINT_VALUE	Setpoint value for a process output from an upstream application block	Numeric	R/W	M
SETPOINT_STATUS	Status of the SETPOINT_STATUS parameter	List of Boolean	R/W	M
READBACK_VALUE	Feedback to the upstream AB readback value	Numeric	R	M
READBACK_STATUS	Status of the READBACK_VALUE parameter	List of Boolean	R	M
ACTUATOR_DEMAND	Demand to the actuator resulting from the transformation function	Enumerated	R	O
POSITION_MEASURE	Result feedback from the actuation/acquisition function	Numeric	R	O
FAILSAFE_ACTION	Fail-safe position for power-loss of the actuator respectively the valve	Enumerated	R/W	O
TEST_COMMAND	Starts test procedure to check the actuator	Enumerated	R/W	O
SETP_CUTOFF_MIN	When the setpoint (SETPOINT_VALUE) goes below the defined percent of span, the actuator signal goes to the minimum limit	Numeric	R/W	O
SETP_CUTOFF_MAX	When the setpoint (SETPOINT_VALUE) goes over the defined percent of span, the actuator signal goes to the maximum limit	Numeric	R/W	O
DEADBAND	Deadband of the actuator	Numeric	R/W	O
SELF_CALIB_STATUS:	Result of the calibration procedure (undetermined, aborted, success)	List of Boolean	R	O
On/Off Actuation Technology Block				
DISC_SETPOINT_VALUE	Local target value for the discrete actuation output	Boolean	R/W	M
DISC_SETPOINT_STATUS	Status of the discrete setpoint	List of Boolean	R/W	M
DISC_READBACK_VALUE	Feedback to the upstream application block readback value	Boolean	R	M

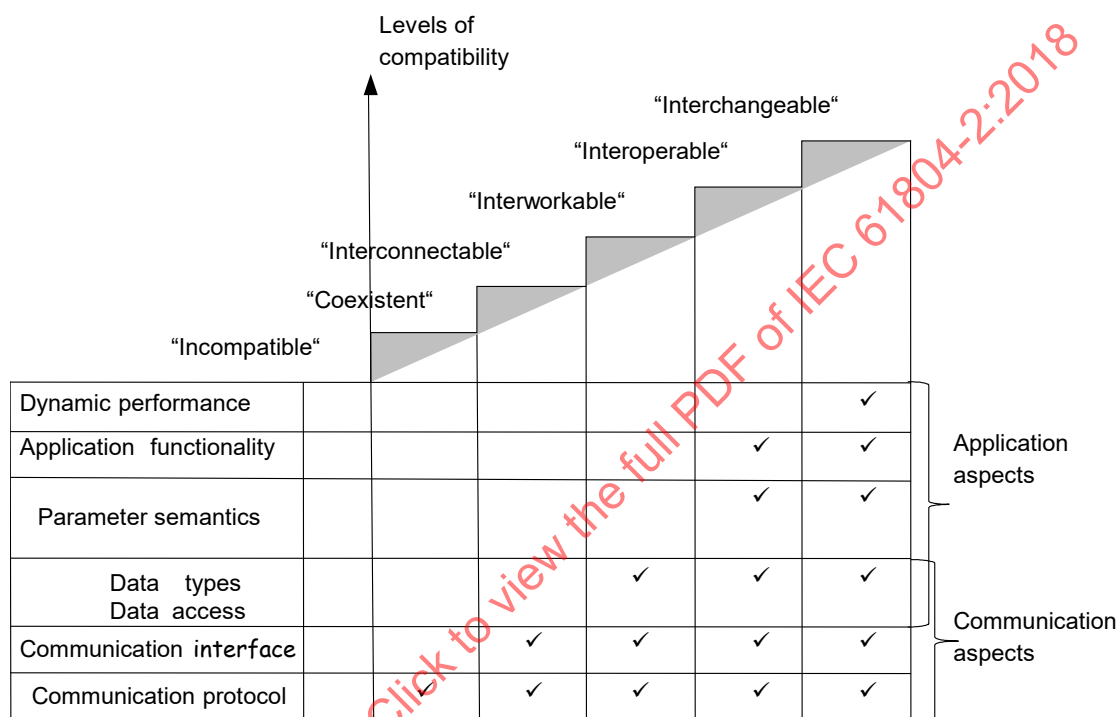
Parameter name	Description	Data type	User Access Read /Write	Class M/O/C
DISC_READBACK_STATUS	Status of the discrete readback output value	List of Boolean	R	M
DISC_ACTUATOR_DEMAND	Demand to the actuator resulting from the transformation function	Boolean	R	O
DISC_POSITION_MEASURE	Result feedback from the actuation/acquisition function	Boolean	R	O
FAILSAFE_ACTION	Fail-safe position for power-loss of the Actuator resp. the valve	Enumerated	R/W	O
TRAVEL_COUNT	Number of cycles from OPEN to CLOSE and CLOSE to OPEN	Numeric	R	O
TRAVEL_COUNT_LIMIT	Limit for TRAVEL_COUNT	Numeric	R/W	O
BREAK_TIME_CLOSE	Dead time between the change of the state(DISC_SETPOINT_VALUE) from CLOSE and the indication that the actuator starts its action	Numeric	R/W	O
BREAK_TIME_OPEN	Dead time between the change of the state (DISC_SETPOINT_VALUE) from OPEN and the indication that the actuator starts its action	Numeric	R/W	O
SELF_CALIB_STATUS	Result of the calibration procedure (undetermined, aborted, success)	List of Boolean	R	O
Device Block				
DEVICE_VENDOR	Company name of the manufacturer	String	R	M
DEVICE_MODEL	Name of the device model	String	R	M
DEVICE_REVISION	Device revision number	String	R	M
DEVICE_SER_NO	Serial number of the device	String	R	O
DEVICE_STATUS	Status of the device	List of Boolean	R	M

Annex B (informative)

Compatibility levels

B.1 General

There are certain levels of compatibility and corresponding levels of cooperation between FB-based devices. The levels are dependent on well-defined communication and application features. See Figure B.1 and Table B.1.



IEC

Figure B.1 – Levels of functional device compatibility

The following main features are used for the definition of compatibility levels (Table B.1).

Table B.1 – Functionality features

Feature	Description
Communication aspects	
Communication protocol	This feature is defined by all protocols of layer 1 to 7 of the ISO OSI Reference Model, i.e. from the physical medium access to the application layer protocol
Communication interface	This feature is defined by the communication service definition of the application layer including the services and the service parameters. Additional mapping mechanisms can be necessary. The dynamic performance of the communication system is part of this feature
Application aspects	
Data types	This feature is defined by the data type of the block data input, data output or parameter
Parameter semantics	This feature is defined by the characteristic features of the data (for example, this can be data name, data descriptions, the data range, the substitute value of the data, the default value, the persistence of the data after power loss and deployment)
Application functionality	This feature is defined by specifying the dependencies and consistency rules between the variables inside the blocks. This is done in the data description part or in a separate behaviour section
Dynamic performance	This feature is defined by time constraints which influence the data or the general device behaviour. For example, the update rate of a process value can influence block algorithms

Regarding these functional features, the following compatibility level names are used for the classification of devices. IEC 61804 (all parts) for distributed FB applications provides for, but does not require, coexistence, interconnectability, interworkability, interoperability and interchangeability between devices using FBs from different manufacturers. This allows the user to choose a device as part of a new system, or as a replacement, and to understand the consequences of that choice

B.2 Compatibility

Ability of a device to provide the set of functions and data required by an application for a specific role in the physical process.

If an application requires a set of functions and profile values "S1" at a specific role, then any device containing set "S1" or a superset of "S1" is compatible with that role in the application. The role represents both control and communication function requirements for a device and the mechanical attachments and environmental features for mounting in the specific part of the physical plant.

The compatibility level means the fulfillment of the intended compatibility level according to Figure B.1.

B.3 Incompatibility

Inability of at devices to provide the functions and performance required by its role in a distributed application.

Incompatibility can result from differences in device capability at any part of the profile in Figure B.1. An incompatible device may interfere with, or prevent, proper communication or functioning (possibly even destructively), if it is connected to distributed application network.

B.4 Coexistence

Ability of two or more devices to operate independently of one another in the same network respecting the common rules for sharing the same medium.

Devices sharing the same communication medium or channel can only operate independently within a defined allocation of bandwidth and schedule timing.

It is not necessary to have an agreement regarding the communication services. Application- and system-specific programming in one or both devices is generally required in order for coexistent devices to work together in the same distributed application.

B.5 Interconnectability

Ability of two or more devices to operate with one another using the same communications protocols, and communication interface.

The devices allow data exchange without agreements about the data types. A data type conversion could be necessary. Unique application-specific programming in one or both devices is generally required for interconnectable devices to function together in the same distributed application.

Devices in standby mode, with the same physical profile are interconnectable using a common medium. Devices need compatible features and settings within their communication protocol to move from standby to operational mode.

B.6 Interworkability

Ability of two or more devices to support the transfer of data between devices with the shared knowledge of the data types of the data transmitted.

If a device is replaced with a similar one of a different manufacture, it can be necessary to reprogram, reconfigure or reparameterise the device and/or the application to accommodate different semantic uses for data from the new device. The distributed application should be designed to accommodate any unique functionality and dynamic responses of the interworkable devices used in the implementation. These devices can exchange data and parameter values but the application may need updating to allow for different semantics, application functionality and dynamics.

B.7 Interoperability

Ability of two or more devices to work together in one or more distributed applications with the shared knowledge of the data types and the semantics of the data transmitted. The data input, data output, parameters, their semantics and the application-related functionality of each device is so defined that, should any device be replaced with a similar one of different manufacture, all distributed applications involving the replaced device will continue to operate as before the replacement, but with possible different dynamic responses.

Interoperability is achieved when both a field device and a system support the same combination of mandatory and optional parts of the same standard. Manufacturer-specific extensions in field devices or systems from different manufacturers may prevent interoperability.

B.8 Interchangeability

Ability of one device to fulfil exactly the same role of another device in the physical process and distributed application as required by the process design.

The interchanged device shall use the device mountings or attachments of its predecessor and comply with the environmental specification the location in the physical plant.

The configuration/setup of the replaced device is used for the new device. Following replacement and setup of an interchangeable device, any distributed applications involving the new device will continue to operate as before the replacement, including identical dynamic responses of the distributed applications.

There is a distinction between exchange of devices and the compatibility level of interchangeability. It is possible to exchange a compatible or coexistent device without influence to the application. The full interchangeability covers the entire features of the compatibility levels.

IECNORM.COM : Click to view the full PDF of IEC 61804-2:2018

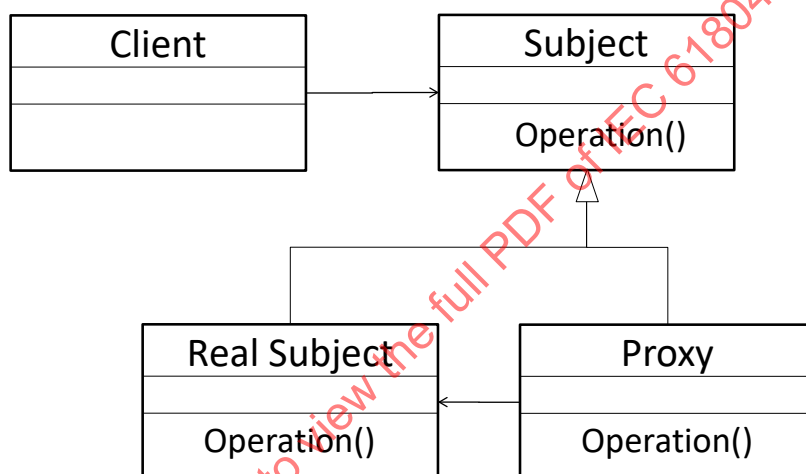
Annex C (informative)

Proxy concept and its use in FB applications

C.1 General proxy concept

The proxy concept for automation is based on the software proxy design pattern (Gamma et al, 1994). It represents the structural design pattern in the range of creational and behavioral patterns.

The main idea is that a client is interacting not with the real subject but with a deputy (i.e. the proxy) of this real subject. This is possible because the proxy provides the same interfaces as the real subject (Figure C.1).



IEC

Figure C.1 – Proxy model class diagram

NOTE The proxy (server) is defined in 3.1.60.

A proxy is a functional unit which is acting in the data channel between two automation devices. The proxy undertakes the roles of the partner pairwise, i.e. the active role for the receiving partner and the passive part for the sending partner. For example in a client-server interaction the proxy is the server related to the client partner and the client for the server partner. This means that the client does not interact with the server directly. This has for example the following implications:

- Synchronisation: Decoupling of the two partners because they are working asynchronously from each other.
- Timing: The age of the data is either unknown or additional means provide the necessary information.
- Availability: Changes of both partners can be done independently, because the proxy can occasionally represent the partner.
- Innovation: Stepwise and innovative upgrades of systems components can be managed because there is no direct influence of the bilateral interactions if the interfaces are stable.
- Privacy: Each partner builds with the proxy one separate communication segment with potential additional security means.

In general the proxy deputizes the function of the partner in terms of its interaction behaviour.

This pattern is frequently used for the interaction between FBs of one FB application which are located in different automation devices. An important example is the interaction between Input and Output FBs with control and Calculation FBs. Input and Output FBs are located in transmitters and actuators and control and calculation FBs often in controllers. Another example is the interaction between application FBs in the controller and supervisory functions in the supervisory station. The automation proxy concept includes both the communication and application functionality of the automation devices.

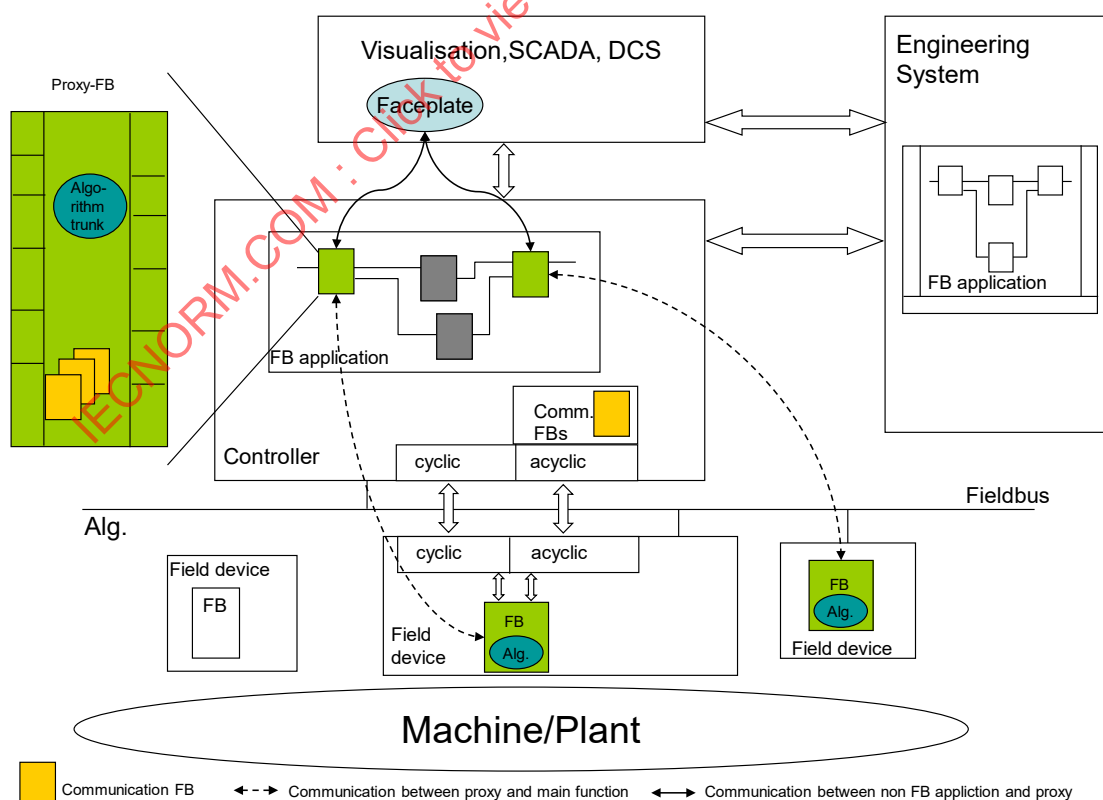
C.2 Use of the proxy concept in FB applications

A functional design of a plant application in terms of an FB application is a result of the planning phase of the engineering. This FB application has to be deployed into the device architecture. Input and Output FBs are naturally located in transmitters and actuators, and control and Calculation FBs in the controller. The signal channel between these FBs crosses then the device borders. This results in related communication services which map the signal channel to the communication services. The service types depends on the necessary quality of services, for example cyclic, acyclic or alarms.

The inputs and outputs for example are represented as proxy FBs in the controller FB application. They contains trunks of the input/output FB applications (e.g. abnormal behaviour, status generation) as well as the necessary communication service means in terms of communication FBs.

NOTE Communication FBs are FB representations of the ASEs of the related communication system.

Consequently the controller FB application contains the input and output functions but the FB internal algorithms (e.g. limit checking, mode of operation) are executed in the remote automation devices (Figure C.2).



IEC

Figure C.2 – Proxy integration in DCS

The faceplates in the supervisory station are mirrors of the data interface of FBs of the controller application. One typical solution for the interaction between the faceplates and the FBs are additional monitoring FBs. They take the role of proxies within the signal flow between both applications.

IECNORM.COM : Click to view the full PDF of IEC 61804-2:2018

Bibliography

IEC 60050-351:1998, *International Electrotechnical Vocabulary (IEV) – Part 351: Automatic control*

IEC 60584-1, *Thermocouples – Part 1: EMF specifications and tolerances*

IEC 61131-3, *Programmable controllers – Part 3: Programming languages*

IEC 61804 (all parts), *Function blocks (FB) for process control and electronic device description language (EDDL)*

IEC 61804-3, *Function Blocks (FB) for process control and Electronic Device Description Language (EDDL) – Part 3: EDDL syntax and semantics*

IEC 61804-5, *Function blocks (FB) for process control and electronic device description language (EDDL) – Part 5: EDDL Builtin library*

ISO/IEC 2022, *Information technology – Character code structure and extension techniques*

ISO/IEC 10646, *Information technology – Universal Coded Character Set (UCS)*

ISO/IEC 19501, *Information technology – Open Distributed Processing – Unified Modeling Language (UML) Version 1.4.2*

ISO 9899, *Information technology – Programming Languages – C*

ISO 15745-1, *Industrial automation systems and integration – Open systems application integration frameworks – Part 1: Generic reference description*

ISO/AFNOR *Dictionary of Computer Science* (1997)

REC-xml-20001006, *Extensible Markup Language (XML) 1.0 Second edition – W3C Recommendation 6 October 2000*

Erich Gamma, Richard Helm, Ralph Johnson und John Vlissides: *Design Patterns. Elements of Reusable Object-Oriented Software*. Addison-Wesley. ISBN 3-8273-2199-9. 1994

RFC 2828:2000, *Internet Security Glossary*

DIN 43710, *Measurement And Control – Electrical Temperature Sensors – Reference Tables Type U And Type L For Thermocouples*

SOMMAIRE

AVANT-PROPOS	79
INTRODUCTION.....	81
1 Domaine d'application	83
2 Références normatives	83
3 Termes, définitions, termes abrégés et conventions	84
3.1 Termes et définitions	84
3.2 Termes abrégés	93
3.3 Conventions pour les structures lexicales	94
4 Définition générale des blocs fonctionnels (FB) et modèle EDD	95
4.1 Structure d'appareil (modèle pour un appareil).....	95
4.1.1 Description de modèle pour un appareil	95
4.1.2 Type de bloc fonctionnel	99
4.1.3 Exécution des blocs fonctionnels	100
4.1.4 Référence entre les modèles décrits dans l'IEC 61499-1 et l'IEC 61804.....	101
4.1.5 Spécification UML du modèle pour un appareil	102
4.1.6 Classification des algorithmes.....	103
4.1.7 Description des algorithmes.....	104
4.1.8 Variables d'entrée et de sortie et définition des paramètres	104
4.1.9 Choix des variables et des paramètres	105
4.1.10 Mode, statut et diagnostic	105
4.2 Combinaisons de blocs	106
4.2.1 Canal de mesure	106
4.2.2 Canal d'activation	107
4.2.3 Application.....	107
4.3 Modèle EDD et EDDL	108
4.3.1 Présentation des EDD et de l'EDDL	108
4.3.2 Architecture EDD	108
4.3.3 Concepts d'EDD	108
4.3.4 Principes du processus de développement d'EDD.....	109
4.3.5 Interrelations entre la structure lexicale et les définitions formelles	110
4.3.6 Built-in	110
4.3.7 Profils	110
5 Définition détaillée des blocs	110
5.1 Généralités	110
5.2 Blocs fonctionnels d'application	111
5.2.1 Bloc fonctionnel à entrée analogique	111
5.2.2 Bloc fonctionnel à sortie analogique.....	112
5.2.3 Bloc fonctionnel à entrée discrète	114
5.2.4 Bloc fonctionnel d'activation sous tension/hors tension Bloc fonctionnel à sortie discrète	116
5.2.5 Bloc fonctionnel de calcul	117
5.2.6 Bloc fonctionnel de commande	118
5.3 Blocs fonctionnels composants	120
5.4 Bloc technique	120
5.4.1 Bloc technique de température	120
5.4.2 Bloc technique de pression	123
5.4.3 Bloc technique d'activation de modulation.....	126

5.4.4	Bloc technique d'activation sous tension/hors tension	128
5.5	Bloc appareil (ressource)	130
5.5.1	Identification	130
5.5.2	État d'un appareil.....	130
5.5.3	Message	132
5.5.4	Initialisation	132
5.6	Algorithmes communs à tous les blocs.....	132
5.6.1	Statut d'entrée de données/sortie de données	132
5.6.2	Validité	132
5.6.3	Initialisation de redémarrage.....	133
5.6.4	À sécurité intégrée.....	133
5.6.5	Initialisation du mode cascade distant.....	134
6	Environnement des blocs fonctionnels	134
7	Mapping avec la gestion de système	135
8	Mapping avec la communication	135
Annexe A (informative)	Description des paramètres	138
Annexe B (informative)	Niveaux de compatibilité.....	145
B.1	Généralités	145
B.2	Compatibilité.....	146
B.3	Incompatibilité	146
B.4	Coexistence	147
B.5	Interconnectabilité	147
B.6	Interexploitabilité	147
B.7	Interopérabilité.....	148
B.8	Interchangeabilité	148
Annexe C (informative)	Concept de mandataire et son utilisation dans les applications de blocs fonctionnels	149
C.1	Concept général de mandataire	149
C.2	Utilisation du concept de mandataire dans les applications de blocs fonctionnels	150
Bibliographie.....		152
Figure 1 – Position de l'IEC 61804-2 par rapport à d'autres normes et produits		81
Figure 2 – Structure des blocs fonctionnels déduite du processus (vue P&ID)		95
Figure 3 – La structure des blocs fonctionnels peut être distribuée entre des appareils (conformément à l'IEC 61499-1)		96
Figure 4 – Les blocs fonctionnels IEC 61804 peuvent être mis en œuvre dans différents appareils		97
Figure 5 – Composants généraux des appareils.....		97
Figure 6 – Types de blocs décrits dans l'IEC 61804 (toutes les parties)		98
Figure 7 – Vue d'ensemble des blocs IEC 61804 (représentation graphique non normative)		99
Figure 8 – Diagramme de classes UML du modèle pour un appareil		102
Figure 9 – Flux de signaux de processus de mesure		106
Figure 10 – Flux de signaux de processus d'activation.....		107
Figure 11 – Flux de signaux de processus d'application.....		108
Figure 12 – Processus de génération d'une EDD		109

Figure 13 – Bloc fonctionnel à entrée analogique	111
Figure 14 – Bloc fonctionnel à sortie analogique	113
Figure 15 – Bloc fonctionnel à entrée discrète	115
Figure 16 – Bloc fonctionnel à sortie discrète.....	116
Figure 17 – Bloc fonctionnel de calcul.....	118
Figure 18 – Bloc fonctionnel de commande.....	119
Figure 19 – Bloc technique de température.....	121
Figure 20 – Bloc technique de pression	124
Figure 21 – Bloc technique d'activation de modulation	126
Figure 22 – Bloc technique d'activation sous tension/hors tension	129
Figure 23 – Diagramme d'états de Harel.....	131
Figure 24 – Structure d'application du modèle de référence OSI ISO	135
Figure 25 – Relation client/serveur en termes de modèle de référence OSI	136
Figure 26 – Mapping des blocs fonctionnels IEC 61804 avec les Objets de processus d'application	136
Figure B.1 – Niveaux de compatibilité des appareils fonctionnels.....	145
Figure C.1 – Diagramme de classes de modèles de mandataires.....	149
Figure C.2 – Intégration du mandataire dans le système de commande réparti	151
Tableau 1 – Descriptions d'attribut de champ.....	94
Tableau 2 – Équivalences entre les éléments de modèles décrits dans l'IEC 61804 et l'IEC 61499	101
Tableau 3 – Variables et modèle de description des paramètres	104
Tableau 4 – Exemple de capteurs de température de type Sensor_Type	122
Tableau 5 – Table des états du statut d'un appareil	131
Tableau 6 – Table des transitions du statut d'un appareil.....	132
Tableau A.1 – Description des paramètres.....	138
Tableau B.1 – Caractéristiques de fonctionnalité	146

COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

**BLOCS FONCTIONNELS (FB) POUR LES PROCÉDÉS INDUSTRIELS ET
LANGAGE DE DESCRIPTION ELECTRONIQUE DE PRODUIT (EDDL) –****Partie 2: Spécification du concept de FB****AVANT-PROPOS**

- 1) La Commission Électrotechnique Internationale (IEC) est une organisation mondiale de normalisation composée de l'ensemble des comités électrotechniques nationaux (Comités nationaux de l'IEC). L'IEC a pour objet de favoriser la coopération internationale pour toutes les questions de normalisation dans les domaines de l'électricité et de l'électronique. À cet effet, l'IEC – entre autres activités – publie des Normes internationales, des Spécifications techniques, des Rapports techniques, des Spécifications accessibles au public (PAS) et des Guides (ci-après dénommés "Publication(s) de l'IEC"). Leur élaboration est confiée à des comités d'études, aux travaux desquels tout Comité national intéressé par le sujet traité peut participer. Les organisations internationales, gouvernementales et non gouvernementales, en liaison avec l'IEC, participent également aux travaux. L'IEC collabore étroitement avec l'Organisation Internationale de Normalisation (ISO), selon des conditions fixées par accord entre les deux organisations.
- 2) Les décisions ou accords officiels de l'IEC concernant les questions techniques représentent, dans la mesure du possible, un accord international sur les sujets étudiés, étant donné que les Comités nationaux de l'IEC intéressés sont représentés dans chaque comité d'études.
- 3) Les Publications de l'IEC se présentent sous la forme de recommandations internationales et sont agréées comme telles par les Comités nationaux de l'IEC. Tous les efforts raisonnables sont entrepris afin que l'IEC s'assure de l'exactitude du contenu technique de ses publications; l'IEC ne peut pas être tenue responsable de l'éventuelle mauvaise utilisation ou interprétation qui en est faite par un quelconque utilisateur final.
- 4) Dans le but d'encourager l'uniformité internationale, les Comités nationaux de l'IEC s'engagent, dans toute la mesure possible, à appliquer de façon transparente les Publications de l'IEC dans leurs publications nationales et régionales. Toutes divergences entre toutes Publications de l'IEC et toutes publications nationales ou régionales correspondantes doivent être indiquées en termes clairs dans ces dernières.
- 5) L'IEC elle-même ne fournit aucune attestation de conformité. Des organismes de certification indépendants fournissent des services d'évaluation de conformité et, dans certains secteurs, accèdent aux marques de conformité de l'IEC. L'IEC n'est responsable d'aucun des services effectués par les organismes de certification indépendants.
- 6) Tous les utilisateurs doivent s'assurer qu'ils sont en possession de la dernière édition de cette publication.
- 7) Aucune responsabilité ne doit être imputée à l'IEC, à ses administrateurs, employés, auxiliaires ou mandataires, y compris ses experts particuliers et les membres de ses comités d'études et des Comités nationaux de l'IEC, pour tout préjudice causé en cas de dommages corporels et matériels, ou de tout autre dommage de quelque nature que ce soit, directe ou indirecte, ou pour supporter les coûts (y compris les frais de justice) et les dépenses découlant de la publication ou de l'utilisation de cette Publication de l'IEC ou de toute autre Publication de l'IEC, ou au crédit qui lui est accordé.
- 8) L'attention est attirée sur les références normatives citées dans cette publication. L'utilisation de publications référencées est obligatoire pour une application correcte de la présente publication.
- 9) L'attention est attirée sur le fait que certains des éléments de la présente Publication de l'IEC peuvent faire l'objet de droits de brevet. L'IEC ne saurait être tenue pour responsable de ne pas avoir identifié de tels droits de brevets et de ne pas avoir signalé leur existence.

La Norme internationale IEC 61804-2 a été établie par le sous-comité 65E: Les dispositifs et leur intégration dans les systèmes de l'entreprise, du comité d'études 65 de l'IEC: Mesure, commande et automation dans les processus industriels.

Cette troisième édition annule et remplace la deuxième édition parue en 2006 et comprend des parties de l'IEC 61804-1, qui a été supprimée en janvier 2013. Cette édition constitue une révision technique.

Cette édition inclut les modifications techniques majeures suivantes par rapport à l'édition précédente:

- a) ajout de la commande mapping avec la communication à l'Article 8 ;

- b) déplacement et reformulation de la définition du niveau de compatibilité de l'IEC 62804-1 dans la nouvelle Annexe B et dans les termes et définitions ;
- c) ajout du concept de mandataire dans la nouvelle Annexe C.

Le texte de cette norme est issu des documents suivants:

FDIS	Rapport de vote
65E/567/FDIS	65E/576/RVD

Le rapport de vote indiqué dans le tableau ci-dessus donne toute information sur le vote ayant abouti à l'approbation de cette norme.

Cette publication a été rédigée selon les Directives ISO/IEC, Partie 2.

Une liste de toutes les parties de la série IEC 61804, publiées sous le titre général *Blocs Fonctionnels (FB) pour les procédés industriels et langage de description électronique de produit (EDDL)*, peut être consultée sur le site web de l'IEC.

Les futures normes de cette série porteront dorénavant le nouveau titre général cité ci-dessus. Le titre des normes existant déjà dans cette série sera mis à jour lors de la prochaine édition.

Le comité a décidé que le contenu de cette publication ne sera pas modifié avant la date de stabilité indiquée sur le site web de l'IEC sous "<http://webstore.iec.ch>" dans les données relatives à la publication recherchée. À cette date, la publication sera

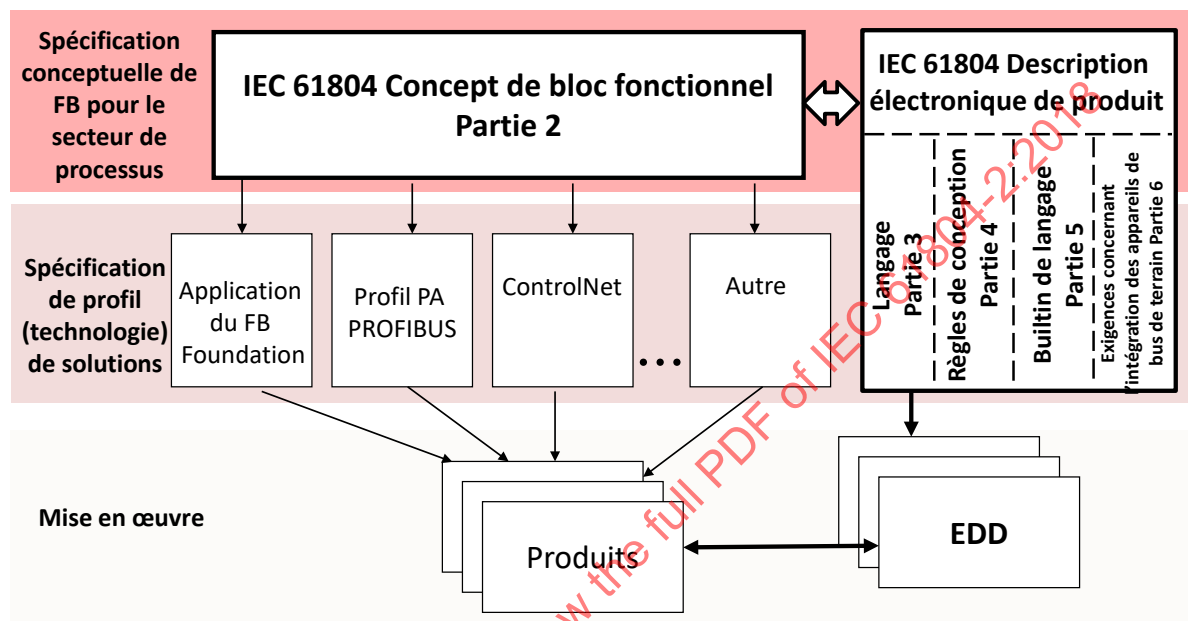
- reconduite,
- supprimée,
- remplacée par une édition révisée, ou
- amendée.

IMPORTANT – Le logo "colour inside" qui se trouve sur la page de couverture de cette publication indique qu'elle contient des couleurs qui sont considérées comme utiles à une bonne compréhension de son contenu. Les utilisateurs devraient, par conséquent, imprimer cette publication en utilisant une imprimante couleur.

INTRODUCTION

La présente partie de l'IEC 61804 fournit une spécification conceptuelle des blocs fonctionnels (FB), qui peut être mise en correspondance avec des systèmes de communication spécifiques et les définitions qui les accompagnent par les groupes industriels.

Le langage EDDL établit un lien entre la spécification conceptuelle des blocs fonctionnels du présent document et une mise en œuvre de produit. La Figure 1 représente ces aspects.



IEC

Figure 1 – Position de l'IEC 61804-2 par rapport à d'autres normes et produits

La Commission Électrotechnique Internationale (IEC) attire l'attention sur le fait qu'il est déclaré que la conformité avec les dispositions du présent document peut impliquer l'utilisation de brevets

Brevet U.S. n° 5,333 114

Brevet U.S. n° 5,485,400

Brevet U.S. n° 5,825,664

Brevet U.S. n° 5,909,368

Brevet U.S. en attente n° 08/916,178

Brevet australien n° 638507

Brevet canadien n° 2,066,743

Brevet européen n° 0495001

Validés dans les pays suivants:

RU – Brevet n° 0495001

France – Brevet n° 0495001

Allemagne – Brevet n° 69032954.7

Pays-Bas – Brevet n° 0495001

Japon – Brevet n° 3137643

L'IEC ne prend pas position quant à la preuve, à la validité et à la portée de ces droits de propriété. Le détenteur de ces droits de propriété a donné l'assurance à l'IEC qu'il consent à négocier des licences avec des demandeurs du monde entier, à des termes et conditions

raisonnables et non discriminatoires. À ce propos, la déclaration du détenteur des droits de propriété est enregistrée à l'IEC. Des informations peuvent être demandées à:

FieldComm Group Inc.,
9430 Research Boulevard, Suite 1-120,
Austin, Texas, USA 78759,
À l'attention de: M. le Président.

L'attention est d'autre part attirée sur le fait que certains des éléments du présent document peuvent faire l'objet de droits de propriété autres que ceux qui ont été mentionnés ci-dessus. L'IEC ne saurait être tenue pour responsable de l'identification de ces droits de propriété en tout ou partie.

L'ISO (www.iso.org/patents) et l'IEC (<http://patents.iec.ch>) maintiennent des bases de données, consultables en ligne, des droits de propriété pertinents à leurs normes. Les utilisateurs sont encouragés à consulter ces bases de données pour obtenir l'information la plus récente concernant les droits de propriété.

La série IEC 61804, dont le titre général est "Blocs fonctionnels (FB) pour les procédés industriels et le langage de description électronique de produit (EDDL)", comprend les parties suivantes:

Partie 2: Concept de FB

Partie 3: Langage de description électronique de produit (EDDL)

Partie 4: Règles de conception EDD

Partie 5: Bibliothèque de Builtin EDDL

Partie 6: Conformité aux exigences concernant l'intégration des appareils de bus de terrain dans les outils techniques pour appareils de terrain

BLOCS FONCTIONNELS (FB) POUR LES PROCÉDÉS INDUSTRIELS ET LANGAGE DE DESCRIPTION ELECTRONIQUE DE PRODUIT (EDDL) –

Partie 2: Spécification du concept de FB

1 Domaine d'application

La présente partie de l'IEC 61804 est applicable aux blocs fonctionnels (FB) pour les procédés industriels.

Le présent document spécifie les blocs fonctionnels sur la base des résultats d'un travail d'harmonisation concernant plusieurs éléments.

- a) Le modèle pour un appareil qui définit les composants d'un appareil conforme à l'IEC 61804-2.
- b) Les spécifications conceptuelles des blocs fonctionnels pour le mesurage, l'activation et le traitement. Ceci inclut des règles générales applicables aux caractéristiques essentielles à l'appui du processus de commande, tout en évitant les informations détaillées qui mettent un terme tant à l'innovation qu'à la spécialisation pour différents secteurs industriels.
- c) La technologie de description électronique de produit (EDD – electronic device description), qui permet, en utilisant les outils d'ingénierie, l'intégration de produits réels dans les systèmes tout au long du cycle de vie.

Le processus de normalisation des blocs fonctionnels a consisté à harmoniser la description des concepts des technologies existantes. Il a produit un niveau abstrait qui a permis de définir les caractéristiques communes d'une manière unique. Cette vision abstraite, appelée ici "spécification conceptuelle des blocs fonctionnels", est mise en correspondance avec des systèmes de communication spécifiques et les définitions qui les accompagnent par les groupes industriels.

NOTE Le présent document peut être mis en correspondance avec l'ISO 15745-1.

Des solutions sont actuellement commercialisées qui satisfont aux exigences du présent document et sont représentatives de la mise en œuvre de la spécification conceptuelle dans une technologie donnée. Il est nécessaire que les nouvelles technologies trouvent des solutions équivalentes (voir Figure 4).

2 Références normatives

Les documents suivants cités dans le texte constituent, pour tout ou partie de leur contenu, des exigences du présent document. Pour les références datées, seule l'édition citée s'applique. Pour les références non datées, la dernière édition du document de référence s'applique (y compris les éventuels amendements).

IEC 61158 (toutes les parties), *Réseaux de communication industriels – Spécifications des bus de terrain*

IEC 61499-1:2012, *Blocs fonctionnels – Partie 1: Architecture*

ISO/IEC 7498-1, *Technologies de l'information – Modèle de référence de base pour l'interconnexion de systèmes ouverts (OSI): Le modèle de base*

3 Termes, définitions, termes abrégés et conventions

3.1 Termes et définitions

Pour les besoins du présent document, les termes et définitions suivants s'appliquent.

L'ISO et l'IEC tiennent à jour des bases de données terminologiques destinées à être utilisées en normalisation, consultables aux adresses suivantes:

- IEC Electropedia: disponible à l'adresse <http://www.electropedia.org/>
- ISO Online browsing platform: disponible à l'adresse <http://www.iso.org/obp>

3.1.1

performance

niveau quantitatif ou qualitatif d'une propriété critique au moment considéré

[SOURCE: ISO 15686-1:2011, 3.15, modifié – Le second terme "performance d'utilisation" a été supprimé. Dans la définition, les mots "quantitatif ou" ont été ajoutés et le mot "critique" a été supprimé.]

3.1.2

sémantique

relations entre les éléments symboliques et leurs définitions, interprétation et utilisation

[SOURCE: IEC 61131-3:2013, 3.85, modifié – Les mots "d'un langage de programmation" ont été supprimés.]

3.1.3

algorithme

jeu fini de règles bien définies pour la solution d'un problème en un nombre fini *d'opérations*

3.1.4

application

unité fonctionnelle logicielle qui est spécifique à la solution d'un problème dans le domaine pour le mesurage et la commande dans les processus industriels

Note 1 à l'article: Une application peut être distribuée parmi des *ressources* existantes et peut communiquer avec d'autres applications.

3.1.5

bloc fonctionnel d'application

FB d'application

bloc fonctionnel qui ne comporte aucune entrée ou sortie du processus

3.1.6

attribut

propriété ou caractéristique d'une entité, par exemple, l'identificateur de version d'une spécification du type de bloc fonctionnel

Note 1 à l'article: La description formelle des attributs fait partie intégrante des profils de solution pour obtenir l'interopérabilité spécifique au domaine. L'IEC 61804 (toutes les parties) définit les règles générales qui permettent de définir les attributs et spécifie le langage EDDL de description des attributs, qui peuvent être décrits dans les profils de solution.

[SOURCE: IEC 61499-1:2012, 3.6, modifié – Une note à l'article a été ajoutée.]

3.1.7

Builtin

sous-programme prédéfini pour la communication et l'affichage exécuté par l'application EDD

3.1.8**coexistence**

aptitude de deux appareils ou plus à fonctionner indépendamment les uns des autres dans le même réseau en respectant les règles communes applicables au partage d'un même support

3.1.9**compatibilité**

aptitude d'un appareil à fournir l'ensemble des fonctions et des données exigées par une application pour un rôle spécifique dans le processus physique

Note 1 à l'article: La fonction comprend les fonctions d'application et de communication, y compris le comportement dynamique.

Note 2 à l'article: Les données comprennent le format de trame et l'ordre des communications ainsi que les définitions des types de données jusqu'à la description sémantique des fonctions.

3.1.10**bloc fonctionnel composant****FB composant**

instance de bloc fonctionnel qui est utilisée dans la spécification d'un algorithme d'un type de bloc fonctionnel composé

Note 1 à l'article: Un bloc fonctionnel composant peut être un type de bloc fonctionnel simple ou un type de bloc fonctionnel composé.

3.1.11**type de bloc fonctionnel composé**

type de bloc fonctionnel dont les algorithmes sont entièrement exprimés en termes de blocs fonctionnels composants et de variables interconnectés

[SOURCE: IEC 61499-1:2012, 3.16, modifié – Les mots "et le contrôle de leur exécution" et "événements" ont été supprimés.]

3.1.12**configuration**

<d'un système ou d'un équipement> action consistant à sélectionner des unités fonctionnelles, affecter leurs emplacements et définir leurs interconnexions

[SOURCE: IEC 61499-1:2012, 3.18]

3.1.13**données**

représentation de faits, concepts ou instructions d'une manière formalisée adaptée à la communication, à l'interprétation ou au traitement par des êtres humains ou par des moyens automatiques

[SOURCE: IEC 61499-1:2012, 3.23, modifié – La définition a été reformulée.]

3.1.14**connexion de données**

association entre des unités fonctionnelles pour l'acheminement de données

[SOURCE: IEC 61499-1:2012, 3.24, modifié – Les mots "deux blocs fonctionnels" ont été remplacés par "des unités fonctionnelles".]

3.1.15**entrée de données**

interface d'un bloc fonctionnel qui reçoit des données d'une connexion de données

[SOURCE: IEC 61499-1:2012, 3.25]

3.1.16

sortie de données

interface d'un bloc fonctionnel qui fournit des données à une connexion de données

[SOURCE: IEC 61499-1:2012, 3.26]

3.1.17

type de données

jeu de valeurs accompagné d'un jeu d'opérations autorisées

[SOURCE: IEC 61499-1:2012, 3.27]

3.1.18

équipement

entité physique indépendante capable d'accomplir une ou plusieurs fonctions spécifiées dans un contexte particulier et délimitée par ses interfaces

[SOURCE: IEC 61499-1:2012, 3.29, modifié – La note à l'article a été supprimée.]

3.1.19

bloc appareil

bloc fonctionnel ne comportant ni entrée ni sortie

3.1.20

application de gestion d'équipement

application dont la fonction principale est la gestion de plusieurs ressources au sein d'un équipement

[SOURCE: IEC 61499-1:2012, 3.30]

3.1.21

application EDD

programme utilisant l'EDD, ou une forme traduite quelconque, qui produit une fonctionnalité telle qu'une représentation de communication, une représentation de données, une représentation graphique, etc.

Note 1 à l'article: L'abréviation "EDD" est dérivée du terme anglais développé correspondant "electronic device description".

3.1.22

processeur EDDL

processeur ou programme, qui traduit l'EDD en forme exécutable qui peut être traitée par une application EDD

Note 1 à l'article: L'abréviation "EDDL" est dérivée du terme anglais développé correspondant "electronic device description language".

3.1.23

profil EDDL

sélection des éléments pris en charge de la structure lexicale EDDL comprenant les définitions de syntaxe pour une pluralité de consortiums spécifiques

Note 1 à l'article: L'abréviation "EDDL" est dérivée du terme anglais développé correspondant "electronic device description language".

3.1.24

langage de description électronique de produit

EDDL

méthodologie pour décrire un ou des paramètres d'un composant de système mettant en œuvre des automatismes

Note 1 à l'article: L'abréviation "EDDL" est dérivée du terme anglais développé correspondant "electronic device description language".

3.1.25

description électronique de produit

EDD

collection de données contenant le ou les paramètres de l'appareil, leurs dépendances, leur représentation graphique et une description des ensembles de données qui sont transférés

Note 1 à l'article: La description électronique de produit est créée en utilisant le langage de description électronique de produit (EDDL).

Note 2 à l'article: L'abréviation "EDD" est dérivée du terme anglais développé correspondant "electronic device description".

3.1.26

source de description électronique de produit

EDDS

fichier ASCII contenant une description spécifique d'appareil

Note 1 à l'article: L'abréviation "EDDS" est dérivée du terme anglais développé correspondant "electronic device description source".

3.1.27

technologie de description électronique de produit

EDDT

technologie qui comprend le processus de développement d'une EDD, l'usage d'une EDD et la chaîne d'outils impliquée

Note 1 à l'article: L'abréviation "EDDT" est dérivée du terme anglais développé correspondant "electronic device description technology".

3.1.28

compilateur de langage de description électronique de produit

outil qui traduit la source d'EDD dans un format interne qui est utilisé par l'interpréteur d'EDD

3.1.29

interpréteur de description électronique de produit

EDDI

outil qui utilise la source d'EDD ou un format interne qui est donné par le compilateur EDDL pour fournir les informations d'EDD à l'utilisateur d'EDD

Note 1 à l'article: L'abréviation "EDDI" est dérivée du terme anglais développé correspondant "electronic device description interpreter".

3.1.30

entité

objet particulier, tel qu'une personne, un lieu, un processus, un objet, un concept, une association ou un événement

[SOURCE: IEC 61499-1:2012, 3.31]

3.1.31

événement

occurrence instantanée qui est significative pour la programmation de l'exécution d'un algorithme

Note 1 à l'article: L'exécution d'un algorithme peut utiliser des variables associées à un événement.

[SOURCE: IEC 61499-1:2012, 3.32]

3.1.32

exception

événement qui entraîne la suspension d'une exécution normale

[SOURCE: IEC 61499-1:2012, 3.36]

3.1.33

fonction

but prévu d'une entité ou son action caractéristique

[SOURCE: IEC 61499-1:2012, 3.44, modifié – Le mot "spécifique" a été remplacé par "prévu".]

3.1.34

unité fonctionnelle

entité d'équipement matériel et/ou logiciel capable d'accomplir une action spécifiée

[SOURCE: IEC 61499-1:2012, 3.48]

3.1.35

bloc fonctionnel

instance de bloc fonctionnel

unité fonctionnelle logicielle consistant en une copie individuelle nommée d'une structure de données et des opérations associées spécifiées par un type de bloc fonctionnel correspondant

Note 1 à l'article: Les opérations typiques d'un FB comprennent la modification des valeurs des données dans sa structure de données associée.

[SOURCE: IEC 61499-1:2012, 3.45, modifié – La définition a été reformulée et la seconde note à l'article a été supprimée.]

3.1.36

réseau de blocs fonctionnels

FBD

réseau dont les nœuds sont des instances de blocs fonctionnels, des variables, des libellés et des événements

Note 1 à l'article: Il ne s'agit pas du diagramme de bloc fonctionnel défini dans l'IEC 61131-3.

Note 2 à l'article: L'abréviation "FBD" est dérivée du terme anglais développé correspondant "function block diagram".

3.1.37

matériel

équipement physique, par opposition aux programmes, procédures, règles et documentation associée

[SOURCE: IEC 61499-1:2012, 3.49]

3.1.38

mise en œuvre

phase de développement dans laquelle le matériel et le logiciel d'un système deviennent opérationnels

[SOURCE: IEC 61499-1:2012, 3.51]

3.1.39**incompatibilité**

inaptitude des appareils à assurer les fonctions et les performances exigées par leur rôle dans une application distribuée

3.1.40**variable d'entrée**

variable dont la valeur est donnée par une entrée de données et qui peut être utilisée dans une ou plusieurs opérations d'un FB

Note 1 à l'article: Un paramètre d'entrée d'un bloc fonctionnel, tel que défini dans l'IEC 61131-3, est une variable d'entrée.

[SOURCE: IEC 61499-1:2012, 3.54]

3.1.41**instance**

unité fonctionnelle consistant en une entité nommée individuelle avec les attributs d'un type défini

[SOURCE: IEC 61499-1:2012, 3.55]

3.1.42**nom d'instance**

identificateur associé à une instance et la désignant

[SOURCE: IEC 61499-1:2012, 3.56]

3.1.43**instanciation**

création d'une instance d'un type spécifié

[SOURCE: IEC 61499-1:2012, 3.57]

3.1.44**interchangeabilité**

aptitude d'un appareil à remplir exactement le même rôle qu'un autre appareil dans le processus physique et l'application distribuée, tel qu'il est exigé par la conception du processus

3.1.45**interconnectivité**

aptitude de deux appareils ou plus à fonctionner ensemble à l'aide des mêmes protocoles de communication et de la même interface de communication

3.1.46**interface**

frontière commune entre deux unités fonctionnelles, définie par des caractéristiques fonctionnelles, des caractéristiques de signal, ou d'autres caractéristiques appropriées

[SOURCE: IEC 60050-351:2013, 351-42-25, modifié – Le mot "commune" a été ajouté à la définition, et les notes à l'article ont été supprimées.]

3.1.47**variable interne**

variable dont la valeur est utilisée ou modifiée par une ou plusieurs opérations d'un bloc fonctionnel, mais n'est pas fournie par une entrée de données ou à une sortie de données

[SOURCE: IEC 61499-1:2012, 3.60]

3.1.48
interopérabilité

aptitude de deux appareils ou plus à fonctionner ensemble dans une ou plusieurs applications distribuées avec les connaissances partagées des types de données et la sémantique des données transmises

3.1.49
interexploitabilité

aptitude de deux appareils ou plus à prendre en charge le transfert de données entre des appareils ayant des connaissances partagées des types de données des données transmises

Note 1 à l'article: Le contenu des données est inconnu, par exemple, chaîne de texte pouvant être affichée mais non interprétée par l'appareil récepteur lui-même, ou valeur dont l'unité n'est pas connue.

3.1.50
invocation

processus de lancement de l'exécution de la séquence d'opérations spécifiées dans un algorithme

[SOURCE: IEC 61499-1:2012, 3.61]

3.1.51
bloc fonctionnel de gestion

bloc fonctionnel dont la fonction principale est la gestion d'applications dans les limites d'une ressource

[SOURCE: IEC 61499-1:2012, 3.64]

3.1.52
ressource de gestion

ressource dont la fonction principale est la gestion d'autres ressources

[SOURCE: IEC 61499-1:2012, 3.65]

3.1.53
mapping

ensemble de valeurs ayant une correspondance définie avec les grandeurs ou valeurs d'un autre ensemble

[SOURCE: IEC 61499-1:2012, 3.66, modifié – Les mots "caractéristiques et attributs" ont été remplacés par "valeurs", et "membres" par "grandeurs ou valeurs".]

3.1.54
modèle

représentation d'un processus, équipement ou concept du monde réel

3.1.55
opération

action bien définie qui, lorsqu'elle est appliquée à n'importe quelle combinaison admissible d'entités connues, produit une nouvelle entité

[SOURCE: IEC 61499-1:2012, 3.73]

3.1.56**variable de sortie**

variable dont la valeur est établie par une ou plusieurs opérations d'un bloc fonctionnel et est fournie à une sortie de données

Note 1 à l'article: Un paramètre de sortie d'un bloc fonctionnel, comme défini dans l'IEC 61131-3, est une variable de sortie.

[SOURCE: IEC 61499-1:2012, 3.74]

3.1.57**paramètre**

variable à laquelle est donnée une valeur constante pour une application spécifiée et qui peut représenter l'application

[SOURCE: IEC 61499-1:2012, 3.75]

3.1.58**préprocesseur**

partie de l'environnement d'exécution qui transforme les informations fournies dans une EDD

3.1.59**directive de préprocesseur**

description de conditions pour filtrer le code d'EDD avant compilation ou interprétation

Note 1 à l'article: Par exemple, une directive de préprocesseur permettant de définir des noms pour des constantes ou d'écrire des macros pour rendre le code plus facile à lire.

3.1.60**mandataire**

processus par ordinateur de liaison d'un protocole entre les systèmes informatiques du client et du serveur, en apparaissant comme le serveur pour le client et comme le client pour le serveur

[SOURCE: RFC 2828:2000]

3.1.61**ressource**

unité fonctionnelle contenue dans un équipement qui a un contrôle indépendant de son fonctionnement et qui fournit divers services à des applications, y compris la programmation et l'exécution d'algorithmes

Note 1 à l'article: Le terme RESOURCE défini dans l'IEC 61131-3 est un élément de langage de programmation correspondant à la ressource définie ci-dessus.

Note 2 à l'article: Un équipement contient une ou plusieurs ressources.

[SOURCE: IEC 61499-1:2012, 3.80, modifié – Les mots "contenue dans un équipement" ont été ajoutés.]

3.1.62**application de gestion de ressource**

application dont la fonction principale est la gestion d'une seule ressource

[SOURCE: IEC 61499-1:2012, 3.81]

3.1.63**service**

capacité fonctionnelle d'une ressource qui peut être modélisée par une séquence de primitives de service

[SOURCE: IEC 61499-1:2012, 3.87]

3.1.64

logiciel

création intellectuelle comprenant les programmes, procédures, règles et toute documentation associée relatifs à l'exploitation d'un système

[SOURCE: IEC 61499-1:2012, 3.92, modifié – Le mot "configurations" a été supprimé.]

3.1.65

système

ensemble d'éléments reliés entre eux, considéré comme un tout dans un contexte défini et séparé de son environnement

Note 1 à l'article: De tels éléments peuvent être tant des objets matériels que des concepts et les résultats de ceux-ci (par exemple, des formes d'organisation, des méthodes mathématiques et des langages de programmation).

Note 2 à l'article: Le système est considéré comme séparé de l'environnement et des autres systèmes extérieurs par une surface imaginaire qui peut couper les liaisons entre eux et le système pris en considération.

[SOURCE: IEC 60050-351:2013, 351-42-08, modifié – Suppression des notes à l'article 1, 2, 4 et 5 et ajout d'une nouvelle note 1 à l'article]

3.1.66

bloc technique

bloc fonctionnel qui comporte au moins une entrée ou une sortie du processus

3.1.67

dictionnaire de texte

collection de textes multilingues ou autres dans l'EDD

Note 1 à l'article: Les références dans une EDD sont utilisées pour sélectionner un dictionnaire de texte approprié.

3.1.68

type

élément logiciel qui spécifie les attributs communs partagés par toutes les instances du type

[SOURCE: IEC 61499-1:2012, 3.99]

3.1.69

nom de type

identificateur associé à un type et le désignant

[SOURCE: IEC 61499-1:2012, 3.100]

3.1.70

variable

entité logicielle qui peut prendre différentes valeurs, une à la fois

Note 1 à l'article: Les valeurs d'une variable sont habituellement limitées à un certain *type de données*.

Note 2 à l'article: Les variables sont décrites comme étant des variables d'entrée, variables de sortie et variables internes.

[SOURCE: IEC 61499-1:2012, 3.102, modifié – Dans la note 2, les mots "variables internes et variables temporaires" ont été remplacés par "et variables internes".]

3.2 Termes abrégés

ADU	Analog digital unit (Unité analogique-numérique)
AFB	Application function block (Bloc fonctionnel d'application)
ANSI	American National Standard Institute (Institut national américain de normalisation)
ANSI C	American National Standard Institute for the programming language C (Institut national américain de normalisation pour le langage de programmation C) (voir ISO 9899-1990)
AP	Application process (Processus d'application)
ASCII	American Standard Code for Information Interchange (Code américain normalisé pour l'échange d'information) (voir ISO/IEC 10646)
ASN.1	Abstract Lexical Structure Notation 1 (Notation de structure lexicale abstraite 1)
BNF	Backus Naur Format (Format de Backus-Naur)
CFB	Component function block (Bloc fonctionnel composant)
DAU	Digital analog unit (Unité numérique-analogique)
DD	Device description (Description de produit)
DTD	Data type definition (Définition de type de données)
EDD	Electronic Device Description (Description électronique de produit)
EDDI	Electronic Device Description Interpreter (Interpréteur de description électronique de produit)
EDDL	Electronic Device Description Language (Langage de description électronique de produit)
EUC	Extended Unit Code (Code d'unité étendu) (voir ISO/IEC 2022:1994)
FB	Function bloc (Bloc fonctionnel)
FBD	Function block diagram (Réseau de blocs fonctionnels)
FMS	Fieldbus message specification (Spécification de message de bus de terrain)
IHM	Interface homme-machine
HTML	Hypertext Mark-up Language (Langage de balisage hypertexte)
E/S	Entrée/Sortie
IAM	Intelligent actuation and measurement (Activation et mesurage intelligents)
ID	Identificateur
mA	Milliampère
NOAH	Network oriented application harmonisation (Harmonisation d'applications orientées réseau)
OSI	Open systems interconnection (Interconnexion de systèmes ouverts)
P&ID	Piping and instrument diagram (Schéma de tuyauterie et d'instrumentation)
PDU	Protocol data unit (Unité de données du protocole)
SM	System management (Gestion de système)
TB	Technology block (Bloc technique)
UML	Unified Modelling Language (Langage de modélisation unifié)
wao	Write as one (Écrire en un bloc)
XML	Extended Mark-up Language (Langage de balisage étendu)

3.3 Conventions pour les structures lexicales

Le langage EDDL est généralement décrit à l'aide de structures lexicales dans lesquelles les éléments et la présence de champs sont spécifiés. Une forme générale des structures lexicales est présentée ci-dessous.

ABC champ1, champ2

ABC est un élément lexical. Cet élément doit être codé dans une syntaxe concrète. Le codage de cet élément sous l'appellation "ABC" n'est pas exigé. Il est également possible de coder cet élément, par exemple avec un numéro de balise (Tag).

Champ1 et champ2 sont des champs de l'élément lexical ABC. Chaque champ est obligatoire et peut avoir plus d'un attribut. Si un champ a des attributs, la présence des attributs est spécifiée dans une table. Une virgule sépare champ1 et champ2. La virgule est un élément lexical qui n'est pas codé de manière explicite.

Si un champ a des attributs additionnels, les attributs sont définis dans une table. La disposition de la table et les qualificatifs d'usage possibles sont présentés dans le Tableau 1.

Tableau 1 – Descriptions d'attribut de champ

Usage	Attribut	Description
m	yyy	La présence de cet attribut est obligatoire
o	xxx	La présence de cet attribut est facultative
s	z1	La présence de cet attribut est configurable avec d'autres attributs, qui sont également marqués avec "s" dans la colonne Usage. Un seul des attributs configurables (z1 ou z2) est présent
s	z2	La présence de cet attribut est configurable avec d'autres attributs, qui sont également marqués avec "s" dans la colonne Usage. Un seul des attributs configurables (z1 ou z2) est présent
c	uuu	La présence de cet attribut est conditionnelle, et il est présent uniquement si la condition est vraie

Les caractères dans la colonne Usage ont les définitions suivantes.

- m: cet attribut est obligatoire et doit être présent;
- o: cet attribut est facultatif et n'est pas nécessairement présent;
- s: cet attribut est une sélection; un et un seul des champs marqués avec "s" (z1 ou z2) doit être présent;
- c: cet attribut est conditionnel; la condition est décrite dans la colonne Description.

Dans la colonne Attribut, s'il existe plusieurs attributs qui ont le même usage, ceux-ci sont triés par ordre alphabétique.

ABC champ1+

Le signe plus (+) après champ1 est utilisé pour indiquer que champ1 est utilisé au moins une fois. Celui-ci peut être utilisé plus d'une fois.

ABC champ2*

L'astérisque (*) derrière champ2 est utilisé pour indiquer que champ2 est facultatif et n'est pas nécessairement utilisé. S'il est utilisé, il peut être utilisé plus d'une fois.

ABC [champ1, champ2] +

Les éléments champ1 et champ2 entre les crochets [] sont une liste non triée. Le signe plus (+) derrière le crochet final indique que champ1 et champ2 sont utilisés au moins une fois et peuvent être utilisés plus d'une fois en tant que groupe.

ABC champ1, (champ2, champ3) <exp>

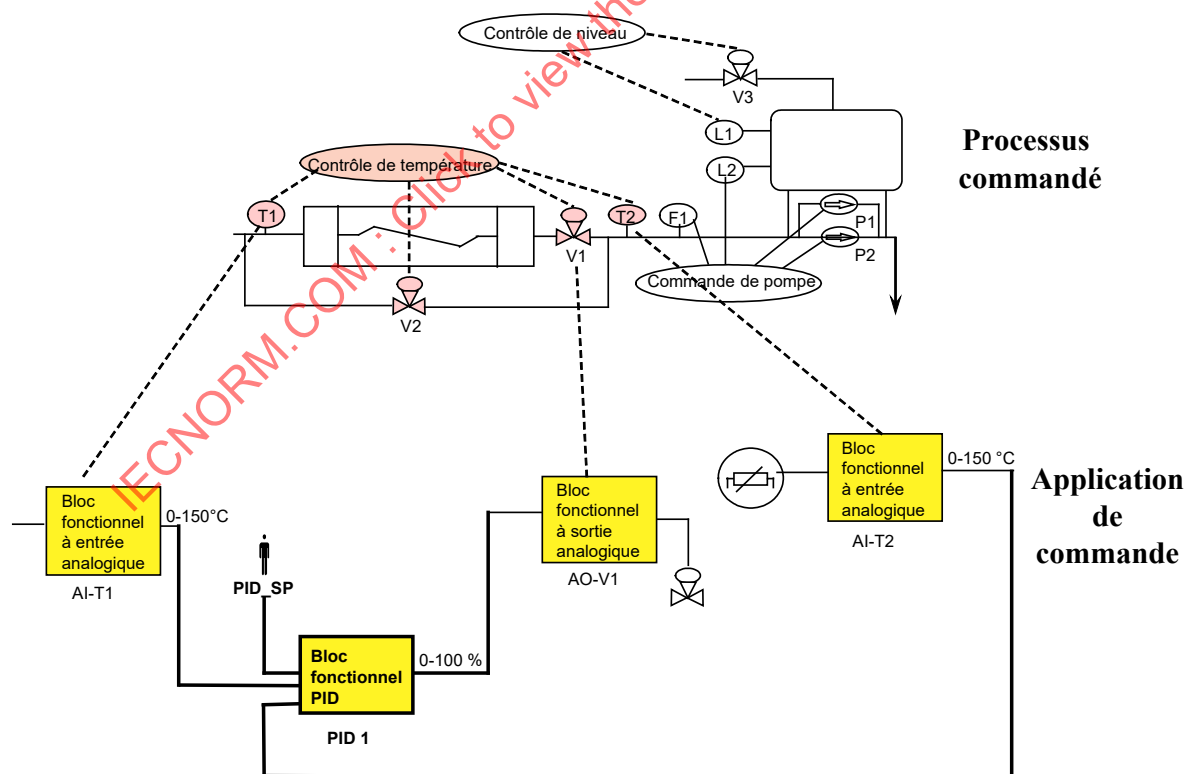
<exp> indique que champ2 et champ3 sont utilisés conjointement avec l'expression conditionnelle (des constructions conditionnelles sont spécifiées dans le Tableau 1). L'expression <exp> désigne uniquement les champs entre parenthèses (). L'utilisation d'expressions conditionnelles est facultative.

4 Définition générale des blocs fonctionnels (FB) et modèle EDD**4.1 Structure d'appareil (modèle pour un appareil)****4.1.1 Description de modèle pour un appareil**

Les blocs fonctionnels sont des encapsulations de variables et de leurs algorithmes de traitement. Les variables et les algorithmes sont les éléments exigés par la conception du processus et son système de commande.

NOTE 1 Les blocs fonctionnels peuvent être déduits du diagramme de la Figure 2.

Les blocs fonctionnels exécutent l'application (mesurage, activation, commande et surveillance) par la connexion de leurs entrées et de leurs sorties de données.

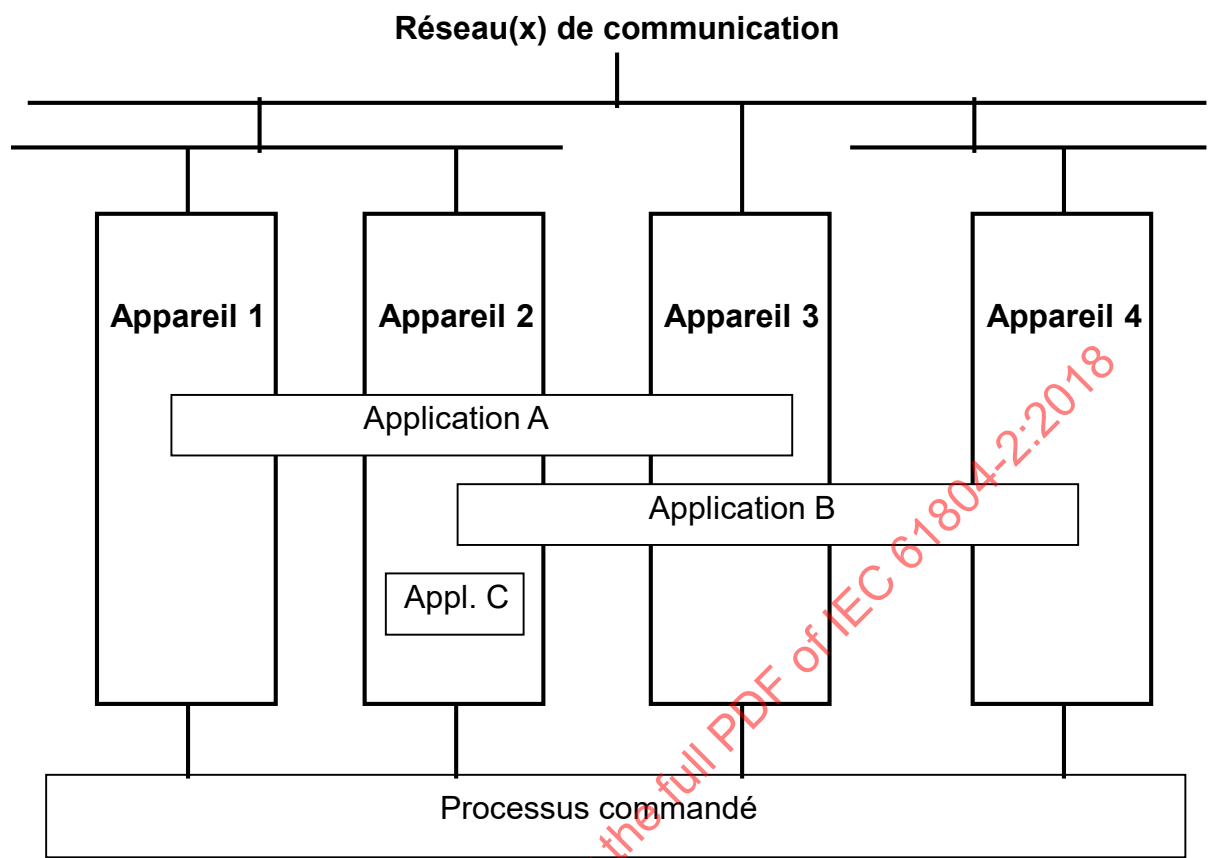


IEC

Figure 2 – Structure des blocs fonctionnels déduite du processus (vue P&ID)

Les appareils sont connectés via un réseau de communication ou une hiérarchie de réseaux de communication.

NOTE 2 L'application peut être distribuée parmi plusieurs appareils. Voir, par exemple, Figure 3.



IEC

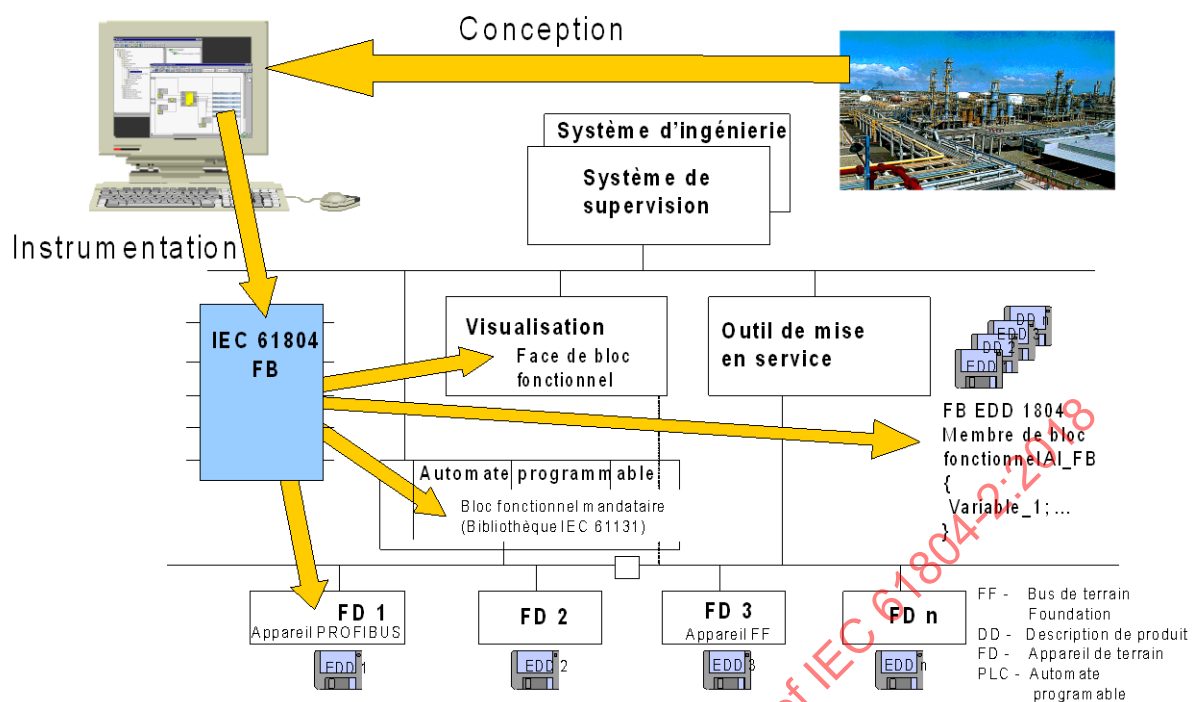
Figure 3 – La structure des blocs fonctionnels peut être distribuée entre des appareils (conformément à l'IEC 61499-1)

Les blocs fonctionnels issus de la conception du système de commande sont des représentations abstraites.

NOTE 3 Ils peuvent être mis en œuvre de différentes manières dans différents types d'appareils, voir Figure 4. Les blocs fonctionnels peuvent être mis en œuvre, par exemple, dans les appareils de terrain, les automates programmables, les postes de visualisation et les descriptions de produits.

De plus, d'autres applications, telles que l'ingénierie système et les systèmes de supervision, doivent s'associer ou interagir avec les blocs fonctionnels.

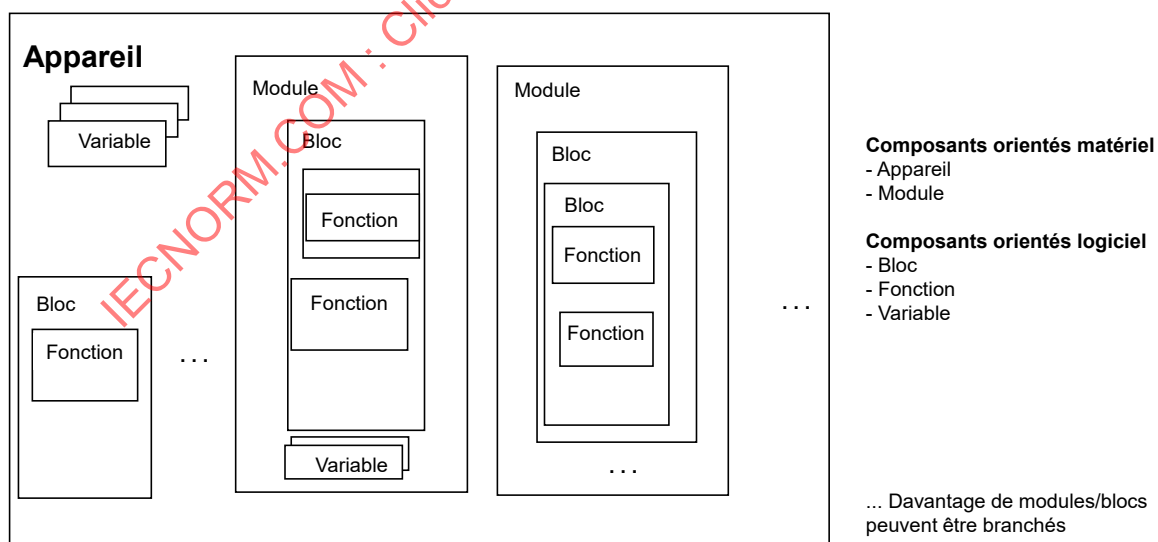
NOTE 4 Les algorithmes définis pour un bloc fonctionnel dans le modèle conceptuel ne sont pas nécessairement mis en correspondance au cas par cas avec l'appareil; ils peuvent être mis en correspondance avec l'appareil, un mandataire ou un poste de supervision si la technologie actuelle ne les résout pas dans l'appareil.



IEC

Figure 4 – Les blocs fonctionnels IEC 61804 peuvent être mis en œuvre dans différents appareils

Pour les besoins du présent document, les appareils mettent en œuvre des algorithmes issus de la conception du processus commandé en termes de blocs fonctionnels. Les appareils sont modulaires avec les matériels et les logiciels; voir, par exemple, Figure 5. Les composants des appareils sont les modules, les blocs, les variables et les algorithmes. Des relations définies existantes entre les composants sont spécifiées dans le diagramme de classes UML ci-dessous, voir Figure 8.

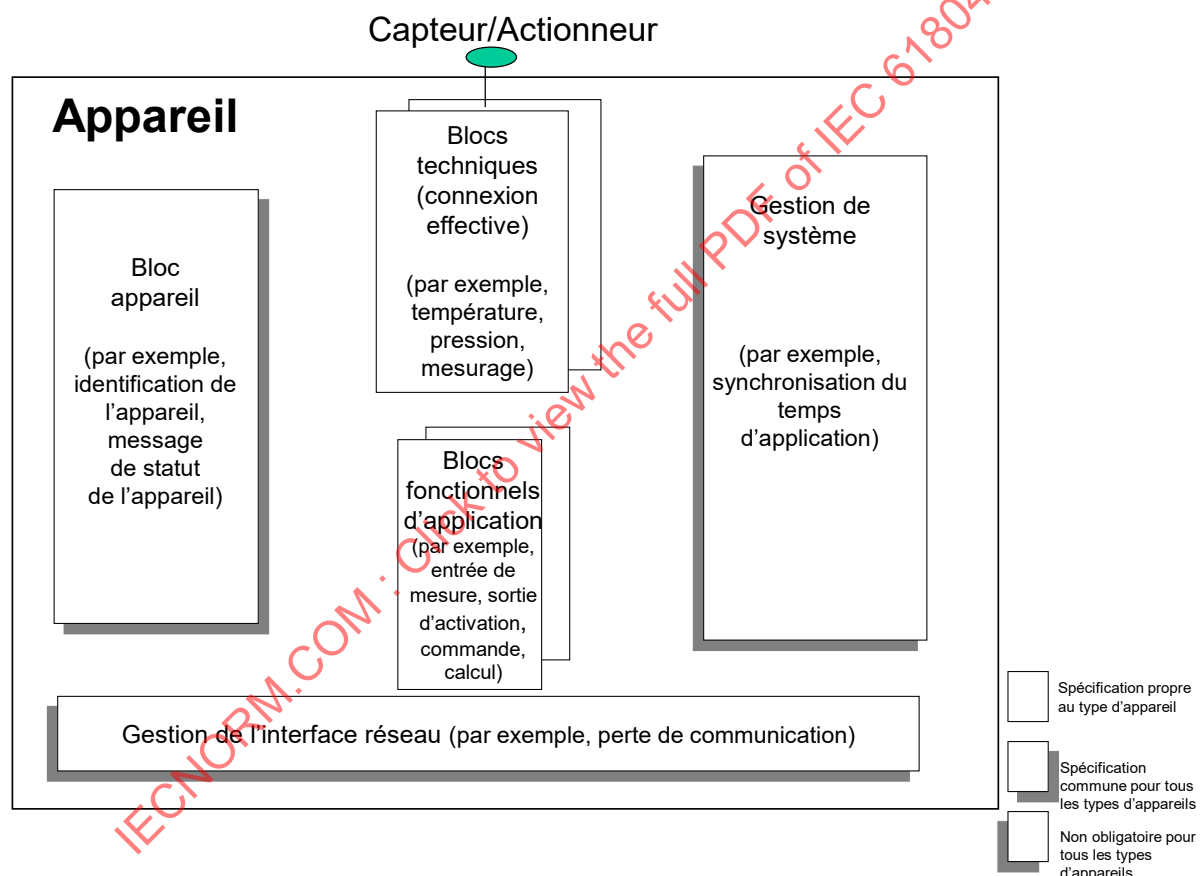


IEC

Figure 5 – Composants généraux des appareils

Pour les besoins du présent document, il existe différents types de blocs (voir Figure 6), qui encapsulent la fonctionnalité spécifique des appareils qui exécutent une application d'automatisation. Le bloc technique représente la connexion effective d'un appareil. Il contient les principes de mesure ou d'activation d'un appareil. Le bloc technique est composé de parties d'acquisition ou de production et de parties de transformation. Le bloc fonctionnel d'application (dénommé ci-après FB) contient le traitement de signaux associé à l'application, tel que la mise à l'échelle, la détection ou la commande d'alarme et le calcul d'alarme. Les blocs fonctionnels composants peuvent effectuer un traitement mathématique et logique associé à des procédures de gestion d'exception supplémentaires spécifiques, telles que les valeurs de paramètres non autorisés. Ces blocs doivent être encapsulés dans les blocs fonctionnels composés.

Le bloc appareil représente la ressource de l'appareil qui contient les informations et les fonctions concernant l'appareil proprement dit, ainsi que son système d'exploitation et le matériel qui lui est associé. L'appareil doit avoir une interface avec le système de communication et peut comprendre des fonctionnalités de gestion de système.



IEC

Figure 6 – Types de blocs décrits dans l'IEC 61804 (toutes les parties)

Tous les appareils qui relèvent du domaine d'application du présent document doivent avoir la même structure logique, voir Figure 6. Le nombre et les types de blocs, qui sont instanciés dans un appareil, sont spécifiques à ce dernier et au fabricant. L'appareil doit comporter au moins un bloc appareil, un bloc fonctionnel d'application et un système de gestion d'interface de réseau.

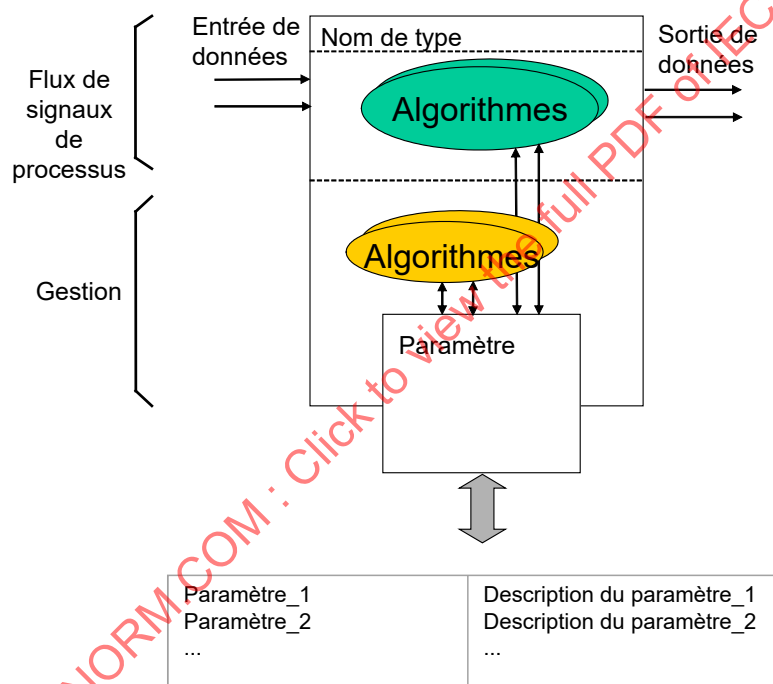
Il existe une chaîne de flux de données qui part de la détection de signal et transite par le bloc technique et les blocs fonctionnels et inversement. Les signaux entre les parties de la chaîne sont internes aux blocs ou visibles en tant que liaisons entre les blocs. La chaîne logique des blocs techniques et fonctionnels est appelée "canal". Ce concept est clarifié en 4.2.1 et 4.2.2.

4.1.2 Type de bloc fonctionnel

Les blocs fonctionnels sont des unités fonctionnelles de logiciels, qui encapsulent des variables et des algorithmes. Un type de bloc fonctionnel est défini par son comportement. Un bloc fonctionnel contient un ou plusieurs algorithmes. La description d'un bloc fonctionnel se présente sous la forme d'une liste d'algorithmes encapsulés dans le bloc fonctionnel, avec les entrées et les sorties de données et les paramètres associés. Il existe des algorithmes qui sont associés au flux de signaux de processus et d'autres qui sont associés à d'autres algorithmes spécifiques aux blocs. Ces autres algorithmes sont appelés "algorithmes de gestion". Les paramètres sont associés au flux de signaux de processus et à leur gestion.

Leur représentation graphique n'est pas normative, voir Figure 7. En d'autres termes, les entrées et les sorties de données représentent l'objectif du flux de signaux de processus (définition conceptuelle) et non les données spécifiques qui acheminent les valeurs correspondantes.

Le tableau des paramètres spécifie toutes les entrées et sorties de données ainsi que tous les paramètres du bloc fonctionnel qui sont accessibles et nécessaires.



IEC

**Figure 7 – Vue d'ensemble des blocs IEC 61804
(représentation graphique non normative)**

Le bloc fonctionnel est résumé par les composants suivants:

- les entrées de données¹ qui prennent en charge le statut² et qui sont associées au flux de signaux de processus uniquement;
- les sorties de données¹ qui prennent en charge le statut² et qui sont associées au flux de signaux de processus uniquement;
- les paramètres² associés au flux de signaux de processus et à leur gestion;

¹ La décision consistant à déterminer quelles données d'un FB constituent une entrée de données, une sortie de données ou un paramètre dépend de mises en œuvre spécifiques.

² Pour des raisons de cohérence, entrée de données/sortie de données et statut constituent une seule structure, de sorte qu'ils soient interdépendants.

- d) les paramètres² qui influencent les fonctions;
- e) les paramètres² qui notifient et rendent visible le comportement interne;
- f) les paramètres² qui sélectionnent des fonctions dans le flux de signaux;
- g) les variables internes avec mémoire pour la prise en charge de l'initialisation, par exemple;
- h) les algorithmes mathématiques/logiques.

Seuls les entrées de données et les paramètres rendent possible l'influence du comportement du bloc fonctionnel. Les entrées de données et les paramètres sont utilisés des différentes manières suivantes:

- i) les données sont utilisées comme entrées ou sorties de fonctions (par exemple, point de consigne pour les fonctions de mise à l'échelle);
- j) les données sont utilisées comme paramètre des fonctions (par exemple, limites pour les alarmes et les avertissements);
- k) les modifications de données de paramètres sont interprétées comme des événements qui permutent les transitions d'automates d'états (par exemple, lancer, interrompre, reprendre le mode d'exploitation des appareils);
- l) les modifications de données de paramètres sont interprétées comme des événements qui lancent les transactions de séquences d'algorithmes (par exemple, lancement des procédures d'étalonnage).

Les noms de données et leur description doivent être vérifiés pour comprendre l'objet des données.

4.1.3 Exécution des blocs fonctionnels

Les appareils comportent différentes méthodes de commande d'exécution. La commande d'exécution des algorithmes de blocs fonctionnels est une caractéristique de chaque appareil. Les appareils et un système distribué peuvent comporter différentes méthodes d'exécution.

NOTE Par exemple, les combinaisons des méthodes de commande d'exécution suivantes sont possibles et d'autres peuvent être ajoutées:

- a) mode libre;
- b) calendrier d'exécution interne à l'appareil (synchronisation temporelle);
- c) mécanisme de déclenchement d'événements internes à l'appareil;
- d) modifications de données de paramètres interprétés comme des événements (voir 4.1.2);
- e) synchronisation temporelle dans l'ensemble du système (synchronisation temporelle au sein du système de communication);
- f) mécanisme de déclenchement du service de communication;
- g) mécanisme de déclenchement d'événements dans l'ensemble du système (par exemple, IEC 61499-1);
- h) commande d'exécution distribuée;

La commande d'exécution des blocs fonctionnels au sein d'un appareil ne constitue qu'un seul aspect de la commande d'exécution d'application globale. La commande d'exécution globale est déterminée, par exemple:

- i) ordre des séquences (de manière séquentielle ou parallèle):
 - 1) ordre d'exécution des blocs suivant le flux de signaux;
 - 2) schéma de tuyauterie des données à exécution parallèle;
 - 3) traitement de la perte de communication entre les appareils;
- j) synchronisation:
 - 1) synchronisation temporelle entre les appareils;
 - 2) établissement d'un calendrier d'exécution échelonné;
- k) contraintes de temps; les éléments suivants sont pris en considération:
 - 1) temps d'exécution des blocs;
 - 2) retard du temps de communication;

- 3) fréquence de mesure;
- 4) temps d'activation;
- 5) choix des algorithmes de blocs;
- 6) retard provenant du comportement de communication;
- l) temps d'exécution des blocs:
 - 1) retard du temps de communication;
 - 2) fréquence de mesure;
 - 3) temps d'activation;
 - 4) choix des algorithmes de blocs;
- m) effet du traitement des exceptions:
 - 1) erreur relative à l'horloge;
 - 2) erreur relative à l'appareil;
 - 3) erreur de communication.

La décision déterminant quelle technologie satisfait aux exigences doit reposer sur une vérification détaillée d'au moins l'ensemble de ces aspects. Le choix de la méthode de commande d'exécution dépend également du niveau technique utilisé pour construire les appareils. La méthode de commande d'exécution des blocs fonctionnels est donc également limitée par les niveaux techniques disponibles dans le bus de terrain utilisé par le système.

4.1.4 Référence entre les modèles décrits dans l'IEC 61499-1 et l'IEC 61804

Les relations avec l'IEC 61499-1 sont données dans le Tableau 2.

Tableau 2 – Équivalences entre les éléments de modèles décrits dans l'IEC 61804 et l'IEC 61499

Élément de modèle de l'IEC 61804	Élément de modèle de l'IEC 61499
Référence des types de blocs	
Bloc fonctionnel d'application	Bloc fonctionnel d'application
Bloc technique	Bloc technique
Bloc appareil (ressource)	Bloc appareil (ressource)
Référence des éléments de blocs fonctionnels	
Bloc composant	Bloc composant
Nom de type	Nom de type
Entrée de données ^a	Entrée de données ^a
Sortie de données ^a	Sortie de données ^a
Algorithmes	Algorithmes
Paramètre	Paramètre
Variable interne	Variable interne
Relations de principe entre les éléments d'EDDL et les éléments de syntaxe de transfert décrits dans l'IEC 61499 ^b	
BLOCK_A, BLOCK_B	FUNCTION BLOCK
VARIABLE et CLASS INPUT	VAR_INPUT, END_VAR
VARIABLE et CLASS OUTPUT	VAR_OUTPUT, END_VAR
-	ALGORITHM
VARIABLE et CLASS CONTAINED	-
VARIABLE	VAR, END_VAR
^a Les entrées et les sorties de données représentent les points d'origine et de destination du flux de signaux de processus (définition conceptuelle), et non les variables spécifiques qui acheminent les données correspondantes.	
^b Il ne s'agit pas d'une référence syntaxique exacte. Elle a pour but de présenter les relations générales.	

Un bloc fonctionnel de type IEC 61804 est un bloc fonctionnel de type IEC 61499-1 sans commande d'exécution et n'a, par conséquent, aucune entrée et aucune sortie d'événement. La commande d'exécution des algorithmes des blocs fonctionnels de type IEC 61804 est masquée, voir 4.1.3.

4.1.5 Spécification UML du modèle pour un appareil

Les définitions du modèle pour un appareil en 4.1.1, à la Figure 5 et à la Figure 6, sont générales. Pour lever l'ambiguïté, le modèle est décrit comme un diagramme de classes UML (voir ISO/IEC 19501-1). Les composants sont transformés en éléments de langage UML à la Figure 8.

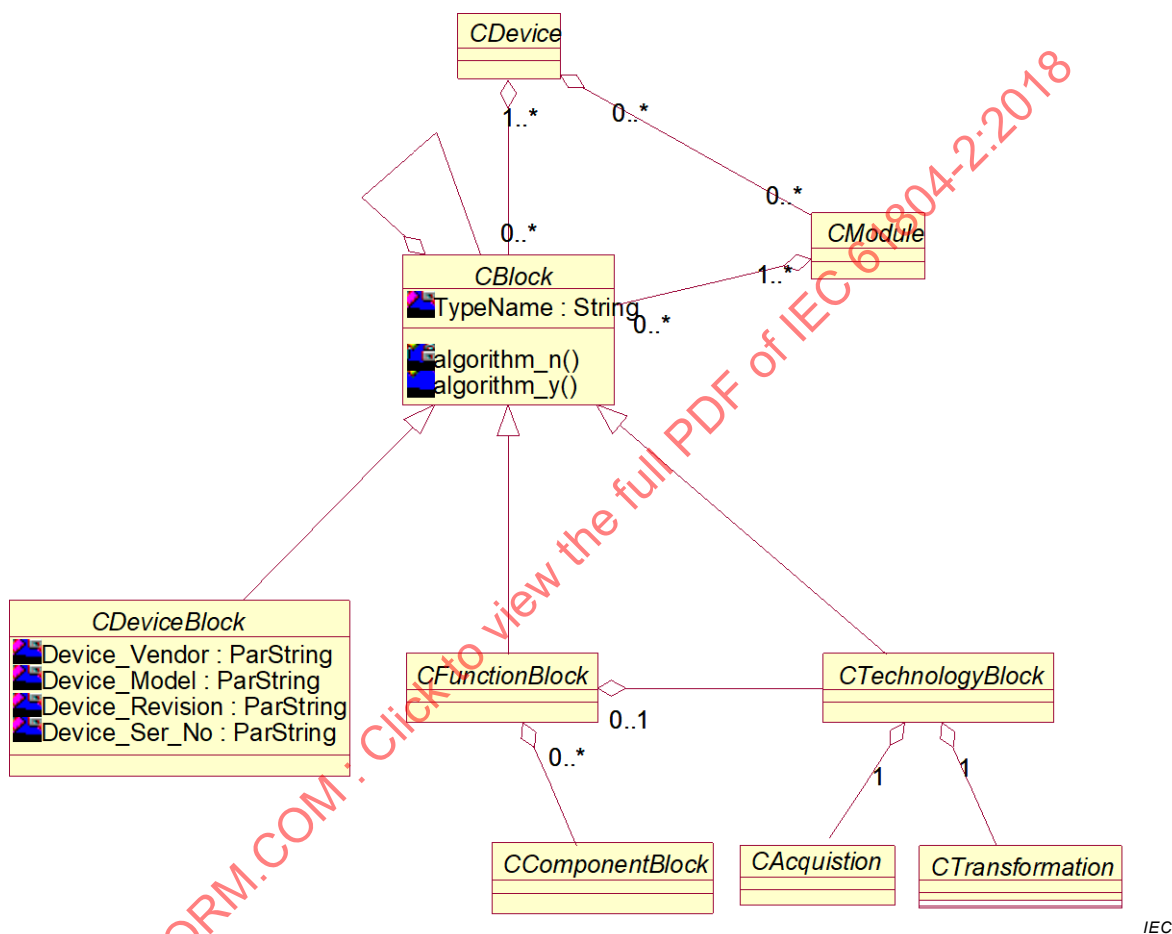


Figure 8 – Diagramme de classes UML du modèle pour un appareil

Les principales étapes suivantes permettent de convertir le modèle pour un appareil en un diagramme de classes UML:

- l'appareil devient la classe CDevice;
- le module devient la classe CModule;
- le bloc appareil, le bloc fonctionnel, le bloc fonctionnel composant et le bloc technique deviennent respectivement CDeviceBlock, CFunctionBlock, CComponentFunctionBlock et CTechnologyBlock;
- les types de blocs sont du type Bloc qui devient CBlock;
- un appareil contient au minimum un bloc;
- un appareil peut contenir des modules;
- un module contient au minimum un bloc;
- les blocs peuvent être constitués d'autres blocs, c'est-à-dire peuvent être du type bloc fonctionnel composé;

- i) un bloc contient au minimum aucun ou plusieurs paramètres;
- j) un bloc doit comporter des algorithmes qui peuvent être uniquement internes ou visibles de l'extérieur (c'est-à-dire à caractère privé ou public);
- k) un bloc appareil contient les attributs Device_Vendor, Device_Model, Device_Revision et Device_Ser_No qui sont des paramètres;
- l) le bloc fonctionnel, le bloc fonctionnel composant et le bloc technique contiennent l'attribut TypeName.

NOTE La classe CBlock peut être référencée par rapport à la déclaration de base du type de bloc fonctionnel de l'IEC 61499-1. Le type de bloc IEC 61804 n'est pas associé à la classe ECCDeclaration.

4.1.6 Classification des algorithmes

La liste suivante fournit des algorithmes communs à utiliser dans les blocs fonctionnels d'application, les blocs transducteurs et les blocs appareil.

a) Algorithmes de signaux de processus:

1) Acquisition de mesure:

- i) connexion des capteurs;
- ii) plage/étalonnage des capteurs;
- iii) conversion analogique-numérique;
- iv) estimation de statut.

2) Transformation de mesure:

- i) linéarisation;
- ii) filtrage;
- iii) compensation;
- iv) mise à l'échelle.

3) Application de mesure:

- i) limite;
- ii) unité;
- iii) mise à l'échelle;
- iv) linéarisation;
- v) simulation.

4) Disposition d'activation:

- i) amplification;
- ii) conversion;
- iii) estimation de statut.

5) Acquisition d'activation:

Se reporter à l'acquisition de mesure pour la relecture de la valeur de sortie de l'actionneur.

6) Transformation d'activation:

- i) mise à l'échelle;
- ii) compensation;
- iii) limites de transition ou d'activité.

7) Application d'activation:

- i) limite;
- ii) unité;
- iii) mise à l'échelle;

- iv) linéarisation;
- v) simulation.
- b) Gestion:
 - 1) estimation du statut de l'appareil;
 - 2) essai;
 - 3) diagnostic;
 - 4) mode de fonctionnement.

4.1.7 Description des algorithmes

La description des algorithmes s'effectue de manière individuelle pour chaque algorithme dans la langue appropriée, par exemple, anglais simple, Diagramme d'états de Harel ou l'un des langages décrits dans l'IEC 61131-3 (par exemple, FBD (réseau de blocs fonctionnels) ou le langage ST (texte structuré)).

La description des profils a pour objet de définir un ensemble général de règles qui permettent l'identification d'un appareil, ainsi que la classification et la spécification des algorithmes pris en charge par l'appareil.

4.1.8 Variables d'entrée et de sortie et définition des paramètres

Pour la description des paramètres de blocs, le Tableau 3 doit être utilisé. Ce tableau propose un modèle de description de l'interface avec un bloc. Il est comparable à un dictionnaire ou à une base de données.

Tableau 3 – Variables et modèle de description des paramètres

Nom du paramètre	Description	Type de données	Accès utilisateur en lecture/écriture	Classe m/o/c
Classe de bloc				

Nom du paramètre:

Identificateur des variables/paramètres auxquels le bloc fonctionnel permet l'accès. Le nom est valide dans le cadre de cette spécification, mais n'est pas normatif pour les produits commercialisés. La décision consistant à déterminer si une donnée est une entrée, une sortie ou un paramètre dépend de l'application.

Description:

Texte informatif, qui décrit l'objet de la variable/du paramètre.

Type de données:

Les types de données suivants sont conceptuels, c'est-à-dire qu'ils identifient le type de signal et non le type de données qui peut être mis en œuvre. Le profil technique permet de les mettre en correspondance avec les données prises en charge dans les catégories suivantes:

- a) numérique (par exemple, virgule flottante, nombre réel, nombre réel long, entier);
- b) énuméré;
- c) booléen;
- d) chaîne (par exemple, chaîne visible, chaîne d'octets);

- e) matrice;
- f) structure.

Accès utilisateur en lecture/écriture

Indique si un appareil distant est en mesure de modifier la variable/le paramètre ou non.

Classe m/o/c

Indique si la variable/le paramètre doivent être pris en charge au sein du bloc ou non; les états sont les suivants: obligatoire (m), facultatif (o), conditionnel (c).

Les attributs de paramètres supplémentaires qui doivent être spécifiés lors du mapping des blocs décrits dans l'IEC 61804 avec d'autres spécifications de blocs fonctionnels sont les suivants:

- g) La classe de récupération après une panne d'alimentation doit avoir la valeur N ou D suivante:

N indique un paramètre non volatil qui doit être conservé pendant un cycle d'alimentation, mais qui ne relève pas du code d'actualisation statique;

D indique un paramètre dynamique calculé par le processus ou le bloc, ou lu à partir d'un autre bloc.

- h) Valeur par défaut:

indique la valeur assignée à un paramètre dans le processus d'initialisation pour un bloc non configuré.

4.1.9 Choix des variables et des paramètres

Les variables, paramètres et algorithmes inclus dans un bloc sont les éléments importants pour l'algorithme et l'appareil. Les blocs fonctionnels incluent au minimum les variables et les paramètres définis dans le schéma P&ID. Les noms des paramètres et des variables ne sont pas normatifs.

4.1.10 Mode, statut et diagnostic

Ces paramètres gèrent et indiquent les performances des canaux. Ils peuvent être consignés dans un rapport; toutefois, les mécanismes de compte-rendu dépendent de la technologie utilisée. Les valeurs consignées peuvent également inclure des éléments supplémentaires, tels que des horodatages, des éléments prioritaires, l'indication des raisons éventuelles.

Le mode décrit l'état de fonctionnement d'un canal ou d'un bloc fonctionnel et influence le flux de signaux au sein de ce canal. Les exemples de modes sont les suivants: manuel, automatique, remplacement local, hors service.

Le statut est un aspect caractéristique d'un canal qui peut accompagner les informations transmises au sein du canal, c'est-à-dire les entrées et les sorties de données de bloc fonctionnel.

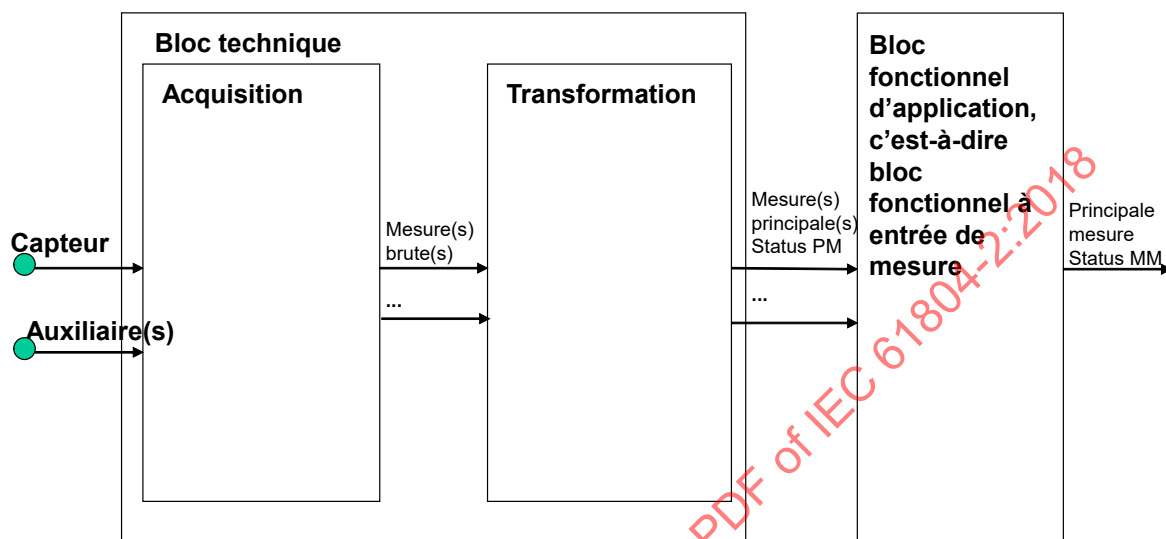
L'état d'un appareil décrit l'état de fonctionnement de ce dernier et interagit avec les blocs techniques et d'application qui lui sont associés; cet état est maintenu au sein de l'appareil par le bloc appareil.

Le diagnostic se présente sous la forme d'un rapport réalisable à partir des algorithmes qui évaluent les performances internes d'un canal ou d'un appareil. Les résultats de ces évaluations internes peuvent servir à établir les informations génériques concernant le statut de mesure, de commande et d'activation.

4.2 Combinaisons de blocs

4.2.1 Canal de mesure

Les blocs fonctionnels techniques et d'application fournissent une chaîne fonctionnelle pour le flux des signaux de processus. Combinés, ils comportent un canal de mesure (voir Figure 9) ou un canal d'activation (voir Figure 10).



IEC

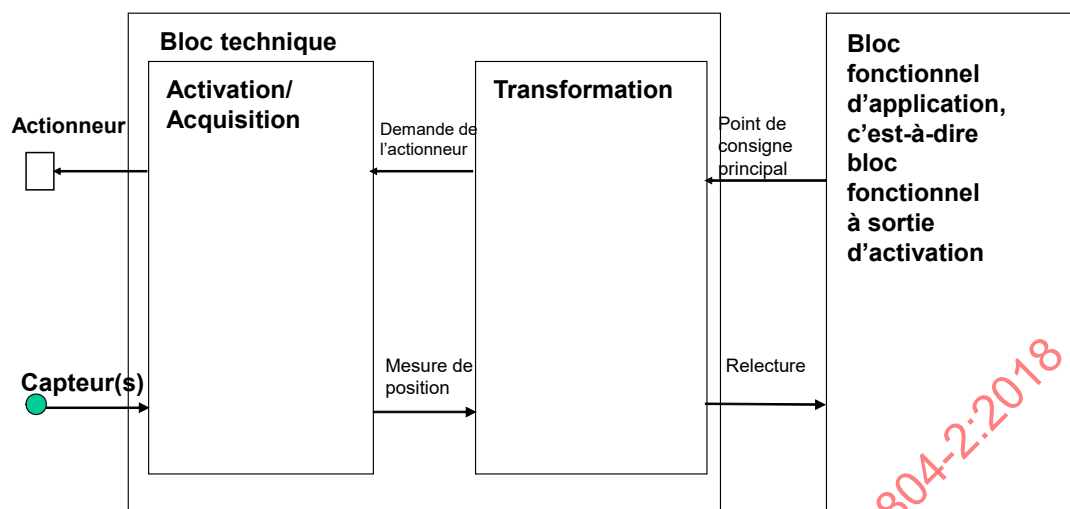
Figure 9 – Flux de signaux de processus de mesure

Un mesurage peut être accompagné de mesurages auxiliaires supplémentaires facultatifs, à des fins telles que la compensation. Le bloc technique fournit une valeur mesurée principale et son statut d'accompagnement. De plus, le bloc technique peut fournir d'autres données de sortie, par exemple, informations de diagnostic ou de validation.

NOTE Des entrées capteur supplémentaires peuvent également être utilisées et transférées par un bloc technique.

Le bloc fonctionnel d'application utilise les données de sortie du bloc technique et d'autres données internes pour générer la mesure principale et son statut d'accompagnement. Le statut est réalisé par chaque fonction dans le flux de signaux en partant du ou des capteurs jusqu'à la dernière fonction dans le bloc fonctionnel d'application. Les informations provenant d'un bloc technique sont proposées à deux blocs fonctionnels d'application ou plus. Un canal de mesure doit comporter au moins un bloc fonctionnel d'application. Des canaux sans bloc technique sont possibles.

4.2.2 Canal d'activation



IEC

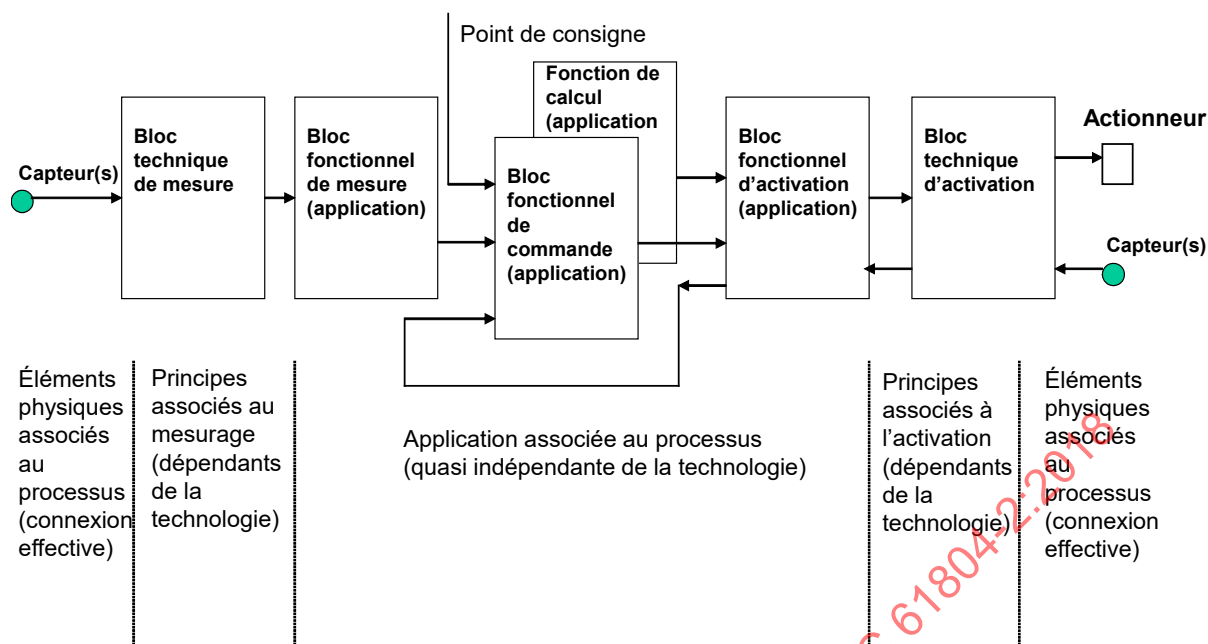
Figure 10 – Flux de signaux de processus d'activation

Le canal d'activation se déroule à l'extérieur de la fonction du flux de signaux d'activation et des fonctions de mesure supplémentaires pour le mesurage de la position actuelle de l'actionneur. En l'absence de capteur pour le mesurage de la position, la transformation utilise alors la demande de l'actionneur pour déterminer la valeur de relecture. Éventuellement, les valeurs de statut peuvent accompagner les deux directions de flux de signaux et comprendre les informations concernant les entités impliquées. Le statut d'accompagnement du point de consigne principal achemine les informations qui permettent au bloc technique d'adopter la position à sécurité intégrée, lorsque ledit point n'est pas correct. Le statut d'accompagnement de la relecture achemine les informations lorsque la valeur mesurée est correcte ou non. Un canal d'activation doit comporter au moins un bloc fonctionnel d'application. Des canaux sans bloc technique sont possibles.

4.2.3 Application

Une application complète est prise en charge par les combinaisons de canaux de mesure et d'activation, ainsi que les blocs fonctionnels de commande et de calcul (voir Figure 11). Les blocs techniques dépendent de la technologie tandis que les autres blocs fonctionnels en sont indépendants. Une application peut être mise en œuvre de nombreuses façons différentes, selon la technologie employée au sein des appareils. L'application peut être exécutée par des mises en œuvre qui n'utilisent que des appareils de mesure et d'activation (c'est-à-dire des appareils complexes capables d'effectuer des opérations de mesure, de commande et d'activation), ou l'application peut être constituée à partir d'appareils de mesure et d'activation, ainsi que d'appareils régulateurs et d'autres composants système.

NOTE Un régulateur peut, par exemple, être intégré à l'application en tant que bloc fonctionnel de calcul, ou un appareil d'activation peut utiliser des parties de fonctions programmables d'appareils régulateurs en termes de blocs fonctionnels de calcul.



IEC

Figure 11 – Flux de signaux de processus d'application

4.3 Modèle EDD et EDDL

4.3.1 Présentation des EDD et de l'EDDL

Une description électronique de produit (EDD) contient tous les paramètres d'un appareil d'un système d'automatisation. La technologie utilisée pour décrire une EDD est appelée langage de description électronique de produit (EDDL). L'EDDL comporte un ensemble d'éléments de langage basiques évolutifs pour gérer des appareils simples, complexes ou modulaires. L'EDDL est un langage descriptif basé sur un format ASCII avec une séparation nette entre les données et le programme.

NOTE Les données dans un champ de texte, qui sont marquées avec un code de pays, par exemple, japonais, peuvent utiliser un code multioctets.

4.3.2 Architecture EDD

Du point de vue du modèle ISO/OSI (ISO/IEC 7498-1), une EDD est au-dessus de la Couche 7. Cependant, l'application EDD utilise le système de communication pour transférer ses informations. Une EDD contient des constructions qui prennent en charge le mapping avec un système de communication de soutien.

Le fabricant de l'appareil définit les objets qui formeront la représentation logique des objets dans une application EDD. Pour cette raison, l'EDDL comprend des éléments de langage qui mettent en correspondance les données de l'EDD avec la représentation de données du système de communication de sorte qu'il n'est pas nécessaire pour l'utilisateur d'une EDD de connaître l'emplacement physique (adresse) d'un objet appareil.

Une EDD décrit la gestion des informations à afficher à l'utilisateur. La représentation spécifique d'une telle visualisation ne fait pas partie des définitions d'EDD ou d'EDDL.

4.3.3 Concepts d'EDD

Le fabricant d'un appareil ou d'un composant d'un système d'automatisation décrit les propriétés de l'appareil en utilisant l'EDDL. L'EDD résultant contient des informations telles que:

- la description des paramètres de l'appareil

- la description des dépendances des paramètres
- le regroupement logique des paramètres de l'appareil
- la sélection et l'exécution des fonctions de l'appareil prises en charge
- la description des ensembles des données transférées

Suivant l'utilisation exigée, l'EDD peut être physiquement située:

- dans un appareil
- dans un support de stockage de données externe tel qu'un disque compact, une disquette ou un serveur
- partiellement distribué dans l'appareil et un support de stockage externe

Une EDD prend en charge les chaînes de texte (termes communs, phrases, etc.) en plusieurs langues (anglais, allemand, français, etc.). Les chaînes de texte peuvent être stockées dans des dictionnaires séparés. Il peut y avoir plus d'un dictionnaire pour une EDD.

Une mise en œuvre d'EDD comprend des informations nécessaires concernant l'appareil cible, par exemple, le fabricant, le type d'appareil, la révision. Cela est utilisé pour mettre en correspondance une EDD spécifique avec un appareil spécifique.

4.3.4 Principes du processus de développement d'EDD

4.3.4.1 Généralités

Le processus de création d'une EDD comporte trois étapes: génération de source d'EDD, prétraitement de l'EDD et compilation de l'EDD.

4.3.4.2 Génération de source d'EDD

Le processus de génération d'une EDD est représenté à la Figure 12.

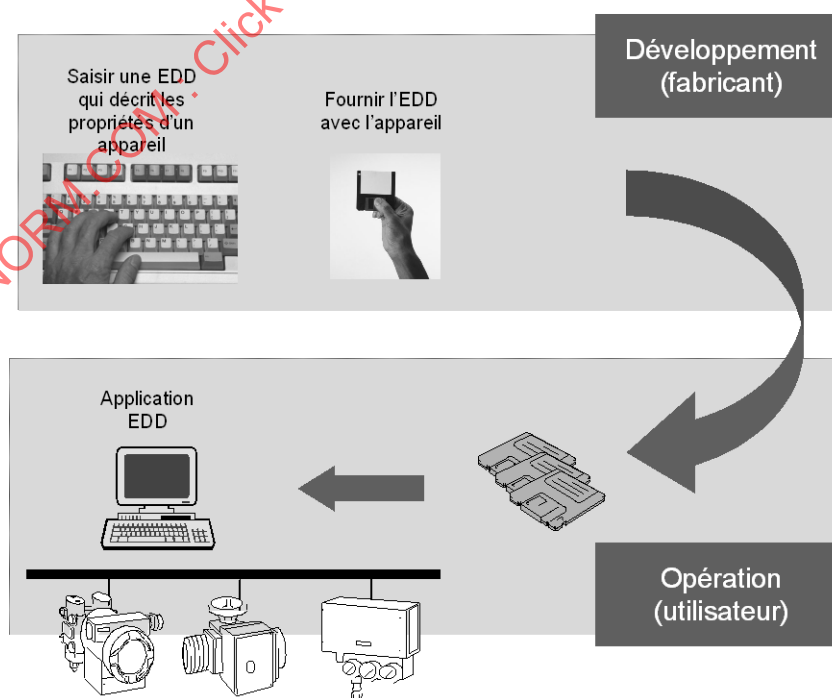


Figure 12 – Processus de génération d'une EDD

Le fabricant d'un appareil écrit une EDD appropriée pour l'appareil et fournit les deux (l'EDD et l'appareil). Si le système d'automatisation prend en charge la méthode EDD, une intégration de système peut être effectuée par l'utilisateur.

Les EDD pour les appareils peuvent être embarquées dans la mémoire de l'appareil ou fournies en utilisant des supports de stockage séparés ou téléchargés depuis un serveur réseau approprié. L'EDD est "interprétée" ou "parcourue" par une application EDD. Les EDD sont normalement stockées en tant que fichiers sources ou fichiers prétraités.

4.3.4.3 Prétraitement de l'EDD

Dans l'étape de prétraitement, un préprocesseur d'EDD génère une représentation de l'EDD cohérente et adaptée pour une compilation finale.

Le prétraitement prend en charge, par exemple, la substitution des définitions et l'inclusion de texte externe. La sortie du préprocesseur est une EDD complète sans aucune directive de préprocesseur.

4.3.4.4 Compilation de l'EDD

L'étape de compilation d'EDD produit une représentation interne d'application EDD à partir d'une EDD prétraitée destinée à être utilisée dans l'application EDD.

4.3.5 Interrelations entre la structure lexicale et les définitions formelles

La structure lexicale de l'EDDL et ses éléments sont décrits dans l'IEC 61804-3, ainsi que les définitions et la syntaxe formelles pour chaque élément de l'EDDL. La structure lexicale et sa description formelle utilisent le même nom.

NOTE À la place des définitions et de la syntaxe formelles spécifiées dans l'IEC 61804-3, une autre spécification peut être développée en tant qu'option additionnelle.

4.3.6 Builtin

Les Builtin sont des sous-programmes prédéfinis qui sont exécutés dans l'application EDD.

EXEMPLE Un terminal portable est un appareil simple ayant un petit écran et des fonctions de curseur limitées. Pour ce type d'appareil, un Builtin peut être spécifié pour produire une entrée d'affichage en utilisant uniquement des actions de curseur haut/bas gauche/droite.

La bibliothèque de Builtin est définie dans l'IEC 61804-5.

4.3.7 Profils

EDDL est une spécification harmonisée de concepts EDD hérités existants. Les applications EDD concrètes utilisent un sous-ensemble de la spécification EDDL. La sélection d'éléments EDDL et de Builtin est effectuée dans les profils définis dans l'IEC 61804-3.

En plus des profils de l'EDD, les consortiums de mise en œuvre publient également des "profils d'appareils" (Device Profiles), qui sont utilisés pour prendre en charge l'interchangeabilité d'appareils conformes. Ces profils d'appareils peuvent être décrits en utilisant l'EDDL spécifié dans le présent document.

5 Définition détaillée des blocs

5.1 Généralités

La présente sélection de blocs n'est pas destinée à être exhaustive. Elle constitue une sélection de blocs de mesure et d'activation très courants.

5.2 Blocs fonctionnels d'application

5.2.1 Bloc fonctionnel à entrée analogique

5.2.1.1 Vue d'ensemble d'un bloc fonctionnel à entrée analogique

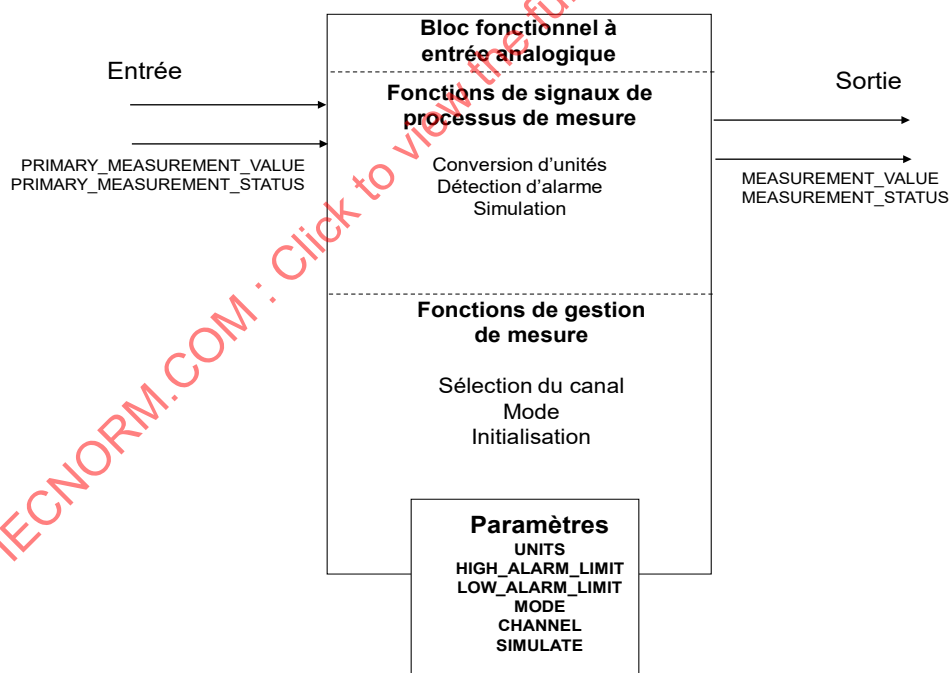
La Figure 13 présente le bloc fonctionnel à entrée analogique.

La fonction de signaux de processus de mesure doit permettre de convertir le ou les signaux d'un bloc technique en unités appropriées pour le mesurage principal exigé pour une application. Le résultat est la MEASUREMENT_VALUE.

NOTE Par exemple, conversion de pouces d'eau en litres par minute. De même, ce bloc peut être utilisé pour notifier à l'opérateur que le mesurage principal a détecté une alarme puissante ou faible. La capacité peut être prévue pour simuler le mesurage de processus lors du contrôle et de l'essai du système.

Chaque signal de processus implique davantage d'informations que la seule valeur du signal; les paramètres de gestion sont généralement exigés. Chaque mesurage a un statut qui indique la qualité de la valeur de mesure.

Le statut fourni par le bloc technique est attribué au bloc fonctionnel de mesure (entrée) par le PRIMARY_MEASUREMENT_STATUS. Le statut est un élément d'information associé à chaque mesurage afin d'aider l'utilisateur dans l'évaluation du caractère utile des données de mesure (généralement les fonctions de commande). Par exemple, il peut s'agir d'une valeur booléenne (valide/non-valide), une valeur continue (incertitude de mesure), une valeur discrète ou une combinaison de ces valeurs (voir 5.6.1).



IEC

NOTE Pour la description des paramètres, voir l'Annexe A.

Figure 13 – Bloc fonctionnel à entrée analogique

5.2.1.2 Conversion d'unités

Cet algorithme convertit le signal d'un bloc technique en une valeur compréhensible. Cette fonction peut être utilisée directement par l'opérateur.

L'utilisateur se sert des UNITS pour sélectionner les unités techniques dans lesquelles la MEASUREMENT_VALUE doit être affichée; par exemple, bar ou mbar.

NOTE Cet algorithme peut également fournir des informations concernant l'état de fonctionnement du canal et de l'appareil afin de faciliter le diagnostic des activités de gestion.

5.2.1.3 Détection d'alarme

Le bloc fonctionnel doit fournir la détection d'alarme facultative interne.

EXEMPLE Alarme faible, alarme puissante, écart, mise à jour.

Lorsqu'elles sont appliquées, les valeurs LOW_ALARM_LIMIT et HIGH_ALARM_LIMIT doivent être comparées avec la MEASUREMENT_VALUE du bloc fonctionnel. Les résultats sont la notification d'une alarme puissante et d'une alarme faible, par exemple, à un opérateur.

NOTE La méthode de notification des alarmes détectées dépend de la technologie; par conséquent, elle n'est pas décrite dans le présent document ou présentée dans la figure correspondante.

5.2.1.4 Simulation

Cet algorithme doit être utilisé pour simuler la valeur MEASUREMENT_VALUE par rapport à une valeur assignée en utilisant le paramètre SIMULATE. Cette opération est habituellement effectuée lors de la mise en service, au cours des phases d'ajustement ou à des fins d'essai, et permet le découplage provisoire de l'application en cours d'exécution du processus concerné.

5.2.1.5 Sélection du canal

Un bloc technique est utilisé pour les données principales d'éléments finaux. Des numéros de canal (CHANNEL) sont définis pour l'appareil de mesure lorsque deux blocs techniques ou plus sont utilisés.

5.2.1.6 Mode

L'algorithme de mode détermine la source de sortie d'un bloc fonctionnel à entrée de mesure sur la base de la valeur de paramètre MODE. En mode automatique, l'algorithme de mesure détermine la sortie. Lorsque le mode est manuel, la sortie du bloc fonctionnel est définie par une source différente; par exemple, elle peut être définie par l'opérateur.

5.2.1.7 Initialisation

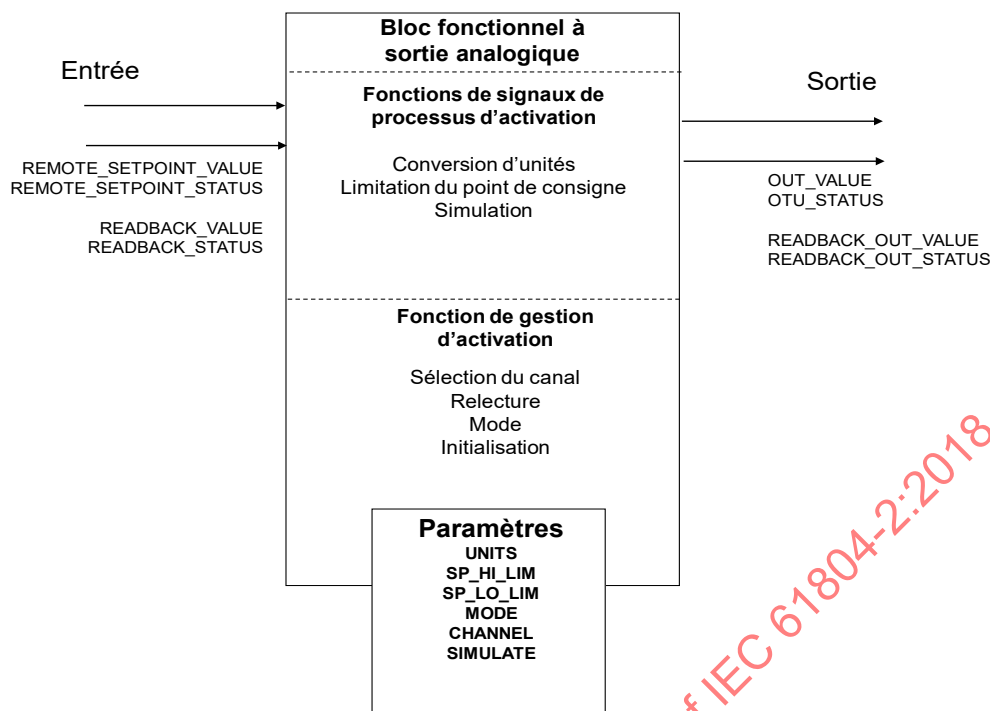
L'algorithme d'initialisation est appliqué à ce bloc et décrit en 5.6.3.

5.2.2 Bloc fonctionnel à sortie analogique

5.2.2.1 Vue d'ensemble d'un bloc fonctionnel à sortie analogique

La Figure 14 présente le bloc fonctionnel à sortie analogique.

L'algorithme de signal de processus d'activation convertit la REMOTE_SETPOINT_VALUE en une valeur utile (OUT_VALUE) pour le matériel spécifié par la sélection du canal relatif au bloc technique. La valeur de retour (transmise par l'actionneur) est fournie comme la READBACK_VALUE. Si le bloc fonctionnel à sortie analogique fait partie d'une chaîne en cascade, alors la READBACK_OUT_VALUE fournit la valeur réelle au bloc fonctionnel en amont. Tous ces paramètres d'entrée et de sortie doivent être accompagnés de leur statut (voir 5.6.1).



IEC

NOTE Pour la description des paramètres, voir l'Annexe A.

Figure 14 – Bloc fonctionnel à sortie analogique

5.2.2.2 Conversion d'unités

Cet algorithme convertit la REMOTE_SETPOINT_VALUE en une valeur que l'actionneur peut utiliser. Les UNITS de la valeur principale de point de consigne REMOTE_SETPOINT_VALUE définissent les unités du point de consigne. La READBACK_VALUE (c'est-à-dire la valeur fournie réelle ou la valeur finale demandée) est également contenue dans les unités du point de consigne.

5.2.2.3 Limitation du point de consigne

La REMOTE_SETPOINT_VALUE fournie au bloc fonctionnel est restreinte aux limites de plages inférieure (SP_LO_LIM) et supérieure (SP_HI_LIM) du point de consigne.

5.2.2.4 Simulation

Cet algorithme permet d'imposer les READBACK_VALUE et READBACK_STATUS aux valeurs assignées par le biais du paramètre SIMULATE. La simulation peut être utilisée, par exemple, pour simuler les anomalies de blocs techniques. En mode simulation, le bloc technique ignore la ou les valeurs de sortie du bloc fonctionnel à activation analogique et maintient la dernière valeur. Cette opération est habituellement effectuée lors de la mise en service, au cours des phases d'ajustement ou à des fins d'essai, et permet le découplage provisoire de l'application en cours d'exécution du processus concerné.

5.2.2.5 Sélection du canal

Un bloc technique est utilisé pour les données principales d'éléments finaux. Des numéros de canal (CHANNEL) sont définis pour l'appareil d'activation de modulation (Modulation Actuator Device) lorsque deux blocs techniques ou plus sont utilisés.

5.2.2.6 Relecture

Cet algorithme donne des informations concernant la valeur fournie réelle de l'actionneur dans le processus.

Les informations READBACK_STATUS sont fournies pour refléter l'état de la valeur d'activation. Il peut s'agir d'une valeur booléenne (valide/non-valide), une valeur continue (incertitude de mesure), une valeur discrète ou une combinaison de ces valeurs.

5.2.2.7 À sécurité intégrée

L'algorithme de sécurité intégrée est décrit en 5.6.4.

5.2.2.8 Mode

L'algorithme de mode détermine la source de sortie du bloc fonctionnel d'activation de modulation sur la base de la valeur de paramètre MODE. En mode automatique, la sortie est déterminée par l'algorithme d'activation de modulation. Lorsque le mode est manuel, la sortie du bloc fonctionnel est définie par une source différente; par exemple, elle peut être définie par l'opérateur.

5.2.2.9 Initialisation

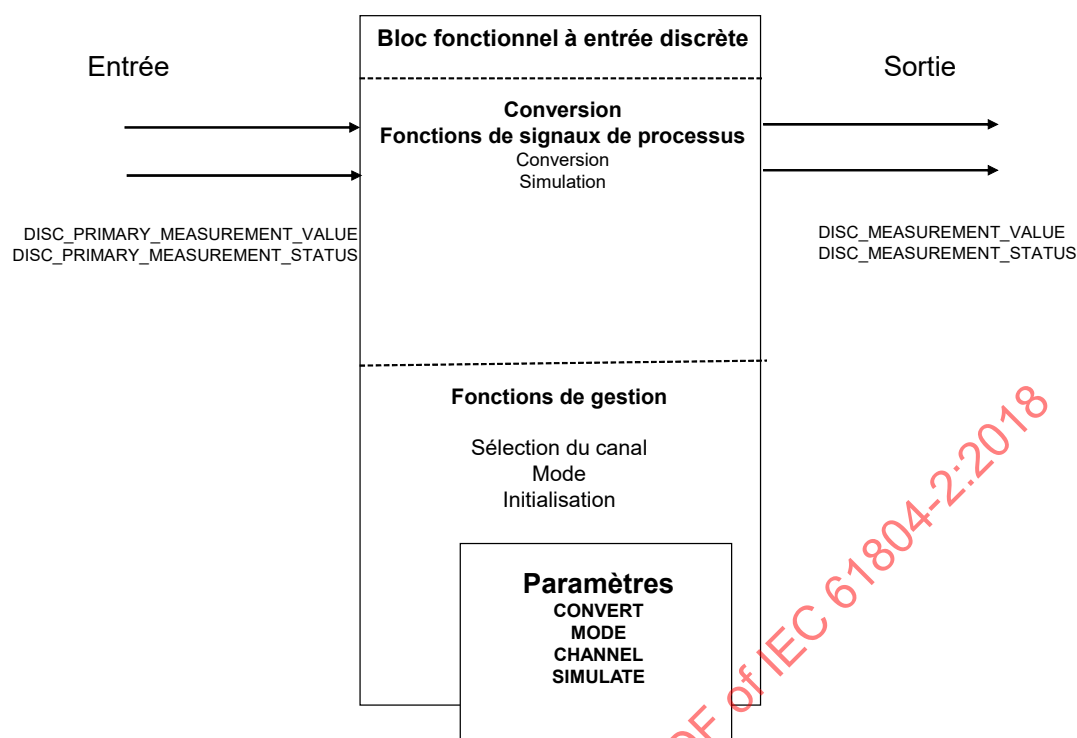
L'algorithme d'initialisation est appliqué à ce bloc et décrit en 5.6.3.

5.2.3 Bloc fonctionnel à entrée discrète

5.2.3.1 Vue d'ensemble de l'entrée discrète

La Figure 15 présente le bloc fonctionnel à entrée discrète.

Les entrées discrètes représentent, par exemple, les détecteurs de proximité inductifs, optiques, capacitifs, ultrasoniques. Lorsque l'entrée numérique change d'état, la sortie discrète change également d'état.



IEC

NOTE Pour la description des paramètres, voir l'Annexe A

Figure 15 – Bloc fonctionnel à entrée discrète

5.2.3.2 Conversion

Cet algorithme convertit la mesure booléenne ou discrète en un signal logique.

Le résultat est la `DISC_MEASUREMENT_VALUE` accompagnée du `DISC_MEASUREMENT_STATUS`.

5.2.3.3 Sélection du canal

Un bloc technique est utilisé pour les données principales d'éléments finaux. Des numéros de canal (`CHANNEL`) sont définis pour l'appareil de détection discrète lorsque deux blocs techniques ou plus sont utilisés.

5.2.3.4 Simulation

Cet algorithme permet d'imposer la valeur discrète principale à une valeur assignée par le biais du paramètre `SIMULATE`. Cette opération est habituellement effectuée lors de la mise en service, au cours des phases d'ajustement ou à des fins d'essai, et permet le découplage provisoire de l'application en cours d'exécution du processus concerné.

5.2.3.5 Mode

L'algorithme de mode détermine la source de la sortie du bloc fonctionnel à entrée de mesure (mesure principale discrète) sur la base de la valeur de paramètre `MODE`. En mode automatique, l'algorithme de mesure discrète détermine la sortie. Lorsque le mode est manuel, la sortie du bloc fonctionnel est définie par une source différente; par exemple, elle peut être définie par l'opérateur.

5.2.3.6 Initialisation

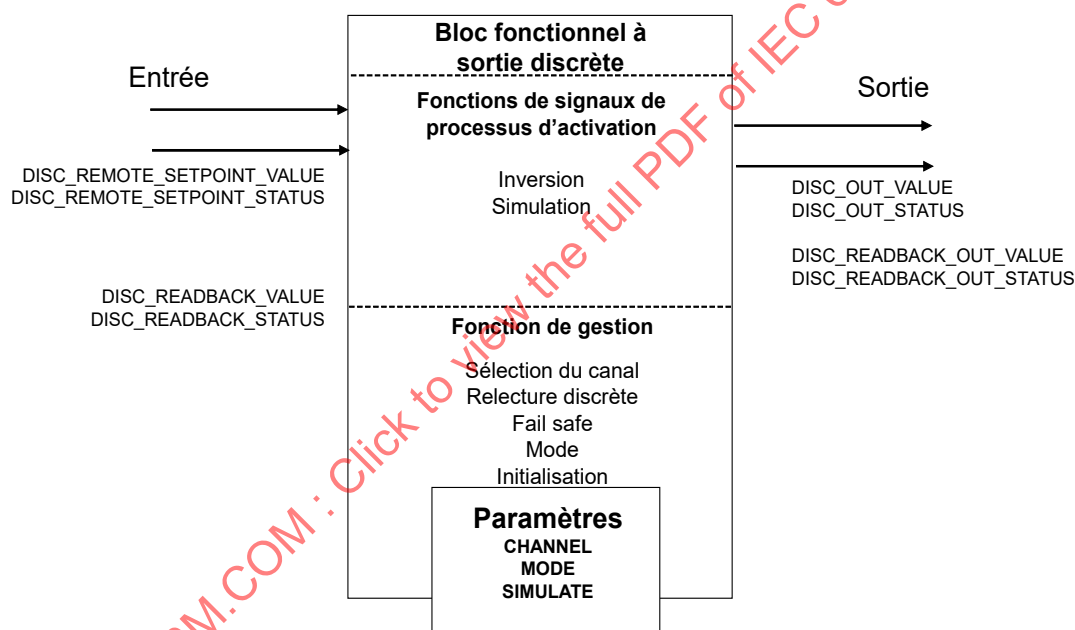
L'algorithme d'initialisation est appliqué à ce bloc et décrit en 5.6.3.

5.2.4 Bloc fonctionnel d'activation sous tension/hors tension Bloc fonctionnel à sortie discrète

5.2.4.1 Vue d'ensemble du bloc fonctionnel d'activation sous tension/hors tension Bloc fonctionnel à sortie discrète

La Figure 16 présente le bloc fonctionnel à sortie discrète.

L'algorithme de signal de processus d'activation convertit la valeur DISC_REMOTE_SETPOINT_VALUE en une valeur utile (DISC_OUT_VALUE) pour le matériel propre à la sélection du canal par rapport au bloc technique. La DISC_READBACK_VALUE définit la valeur cible de l'élément final. Si le bloc fonctionnel à sortie discrète fait partie d'une chaîne en cascade, la DISC_READBACK_OUT_VALUE fournit la valeur réelle au bloc fonctionnel en amont. Tous ces paramètres d'entrée et de sortie doivent être accompagnés de leur statut (voir 5.6.1).



IEC

NOTE Pour la description des paramètres, voir l'Annexe A.

Figure 16 – Bloc fonctionnel à sortie discrète

5.2.4.2 Inversion

Il est parfois nécessaire d'inverser logiquement la DISC_REMOTE_SETPOINT_VALUE avant de la retransmettre à la demande d'activation discrète. Cet algorithme effectue cette opération.

5.2.4.3 Simulation

Cet algorithme permet d'imposer les DISC_READBACK_VALUE et DISC_READBACK_STATUS aux valeurs assignées par le biais du paramétrage SIMULATE. La simulation peut être utilisée, par exemple, pour simuler les anomalies de blocs techniques. En mode simulation, le bloc technique ignore la valeur DISC_OUT_VALUE et maintient la dernière valeur. Cette opération est habituellement effectuée lors de la mise en service, au cours des phases d'ajustement ou à des fins d'essai, et permet le découplage provisoire de l'application en cours d'exécution du processus concerné.

5.2.4.4 Sélection du canal

Un bloc technique est utilisé pour les données principales d'éléments finaux. Des numéros de canal (CHANNEL) sont définis pour l'appareil d'activation de modulation (Modulation Actuator Device) lorsque deux blocs techniques ou plus sont utilisés.

5.2.4.5 À sécurité intégrée

L'algorithme de sécurité intégrée est décrit en 5.6.4.

5.2.4.6 Mode

L'algorithme de mode détermine la source de sortie du bloc fonctionnel d'activation sous tension/hors tension sur la base de la valeur de paramètre MODE. En mode automatique, l'algorithme d'activation sous tension/hors tension détermine la sortie. Lorsque le mode est manuel, la sortie du bloc fonctionnel est définie par une source différente; par exemple, elle peut être définie par l'opérateur.

5.2.4.7 Initialisation

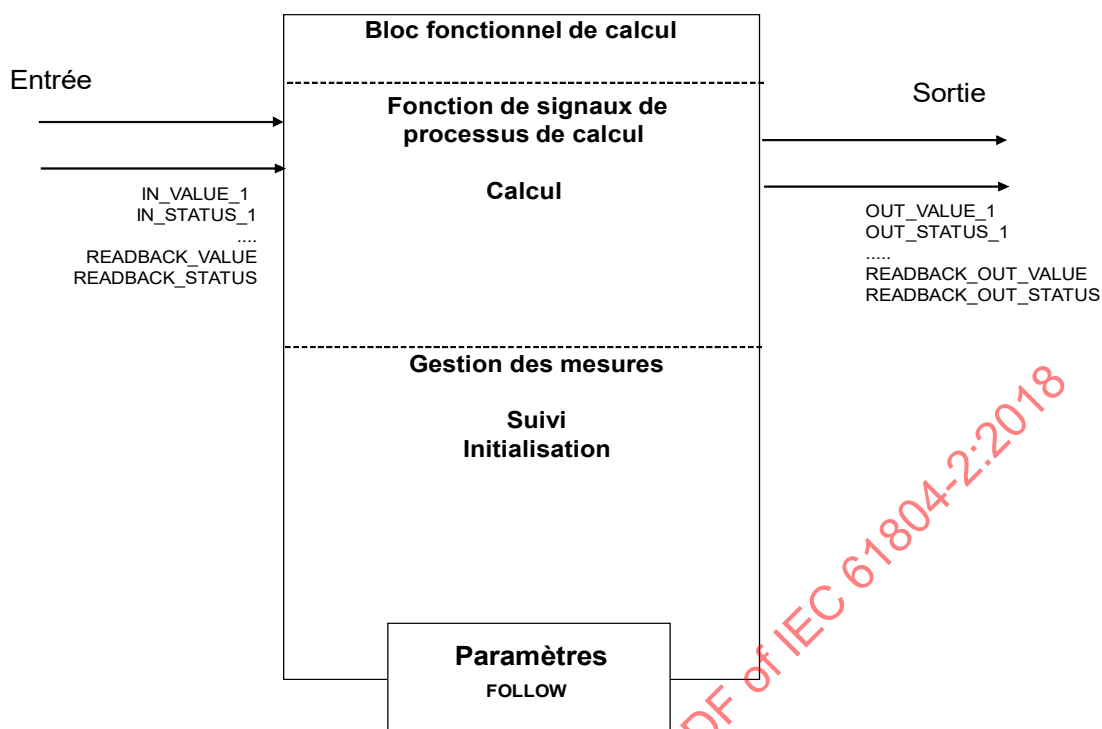
L'algorithme d'initialisation est appliqué à ce bloc et décrit en 5.6.3.

5.2.5 Bloc fonctionnel de calcul

5.2.5.1 Vue d'ensemble du bloc fonctionnel de calcul

La Figure 17 présente le bloc fonctionnel de calcul.

Le bloc fonctionnel de calcul influe sur le ou les signaux d'entrée (IN_VALUE_x) d'autres blocs fonctionnels afin de fournir une valeur d'application (OUT_VALUE_x). Lorsque le bloc fonctionnel de calcul fait partie d'une chaîne en cascade, la READBACK_OUT_VALUE fournit la valeur réelle au bloc fonctionnel en amont et la READBACK_VALUE est fournie par un bloc fonctionnel en aval. Tous ces paramètres d'entrée et de sortie doivent être accompagnés de leur statut (voir 5.6.1).



IEC

NOTE Pour la description des paramètres, voir l'Annexe A.

Figure 17 – Bloc fonctionnel de calcul

5.2.5.2 Calcul

Cet algorithme détermine le ou les signaux de sortie sur la base d'un algorithme prédéfini et de la ou des entrées du bloc fonctionnel. Les exemples de fonctions de calcul sont le filtrage, le report, la sélection des entrées.

5.2.5.3 Suivi

L'algorithme suivi permet de régler la sortie du bloc fonctionnel sur une valeur d'entrée lorsque le paramètre FOLLOW est actif, c'est-à-dire de valeur non nulle. Par exemple, cet algorithme peut être utilisé pour initialiser un bloc ou pour imposer les résultats de calcul à une valeur spécifique.

Ceci ne s'applique pas à tous les blocs.

5.2.5.4 Initialisation

L'algorithme d'initialisation est appliqué à ce bloc et est décrit en 5.6.3.

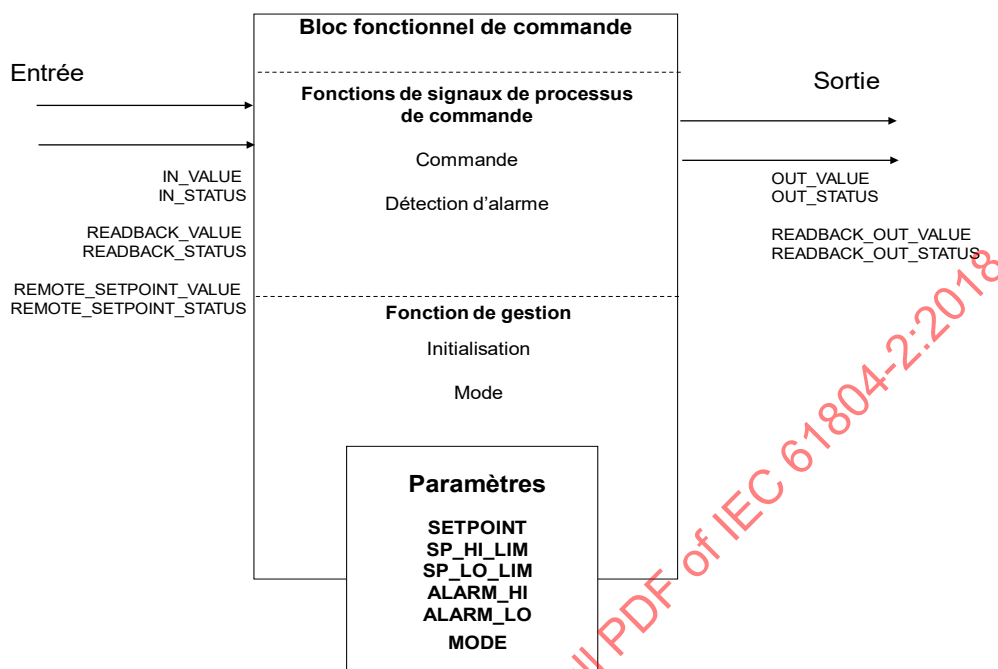
5.2.6 Bloc fonctionnel de commande

5.2.6.1 Vue d'ensemble du bloc fonctionnel de commande

La Figure 18 présente le bloc fonctionnel de commande.

Le bloc fonctionnel de commande maintient une entrée de processus (IN_VALUE) à la valeur du point de consigne (SETPOINT) par la régulation d'une ou de plusieurs sorties d'activation de processus. Le mesurage d'entrée de processus est assuré par un bloc fonctionnel approprié par la connexion d'entrée principale. La sortie principale du bloc fonctionnel de commande régule le processus grâce à un bloc fonctionnel d'activation approprié. Le

SETPOINT définit la valeur cible du mesurage de processus en mode "automatique". La valeur et le statut de relecture fournis par le bloc d'activation en aval peuvent être utilisés lors de l'initialisation et pour modifier l'action de commande lorsque la sortie est limitée par une condition en aval.



IEC

Figure 18 – Bloc fonctionnel de commande

5.2.6.2 Commande

Cet algorithme détermine la valeur de sortie de bloc fonctionnel nécessaire pour acheminer la valeur d'entrée principale vers la valeur cible spécifiée par le paramètre SETPOINT. Les modifications de la valeur SETPOINT sont limitées à la plage spécifiée par les limites SP_HI et SP_LO. Les actions peuvent être modifiées lorsqu'une entrée de relecture du bloc aval indique qu'une condition également aval limite l'ajustement de la sortie du bloc.

5.2.6.3 Détection d'alarme

La détection d'alarme est facultative. Lorsqu'elles sont appliquées, les valeurs LOW_ALARM_LIMIT et HIGH_ALARM_LIMIT doivent être comparées avec la valeur principale de mesure de la commande du bloc. Les résultats sont la notification d'une alarme puissante et d'une alarme faible, par exemple, à un opérateur.

NOTE La méthode de notification des alarmes détectées dépend de la technologie; par conséquent, elle n'est pas décrite dans le présent document ou présentée dans la figure correspondante.

5.2.6.4 MODE

L'algorithme de mode détermine la source de sortie du bloc de commande sur la base de la valeur de paramètre MODE. En mode automatique, la sortie est déterminée par l'algorithme de commande et le SETPOINT est spécifié par l'opérateur. Lorsque le mode est Manuel, la sortie du bloc est définie par une source différente; par exemple, elle peut être définie par l'opérateur.

En mode distant, la sortie est déterminée par l'algorithme de commande et le point de consigne est déterminé par l'entrée REMOTE_SETPOINT d'un autre bloc fonctionnel.

5.2.6.5 Initialisation

Lorsque le statut de retour indique le blocage du trajet vers l'entrée de processus, la sortie du bloc fonctionnel est définie sur la base de la valeur de relecture afin de permettre un transfert sans perturbation lorsque le mode aval passe en mode distant.

5.3 Blocs fonctionnels composants

Une application de procédé industriel est constituée des blocs fonctionnels d'application définis ci-dessus. De plus, l'application peut inclure les blocs fonctionnels composants combinés d'une manière spécifique à l'application et encapsulés par des blocs fonctionnels de type composé, comme décrit en 4.1.1. Le traitement des exceptions et des statuts est spécifique à la technologie et fait partie intégrante des définitions propres aux blocs fonctionnels composants.

5.4 Bloc technique

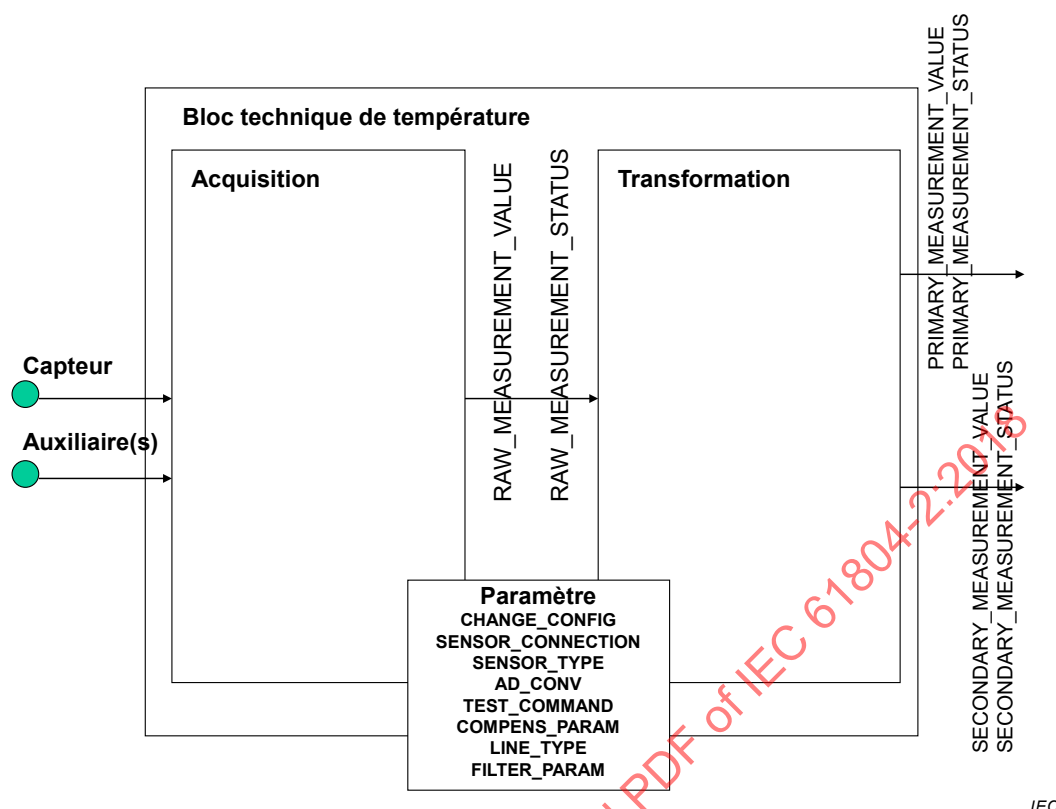
5.4.1 Bloc technique de température

5.4.1.1 Vue d'ensemble du bloc technique de température

Les algorithmes du bloc technique de température sont résumés ci-dessous:

- a) connexion des capteurs
- b) plage/mise à l'échelle des canaux
- c) conversion analogique-numérique
- d) essai
- e) diagnostic
- f) compensation de jonction froide
- g) linéarisation
- h) filtrage
- i) initialisation

Les algorithmes sont encapsulés dans la partie Acquisition et Transformation du bloc technique (voir Figure 19).



IEC

Figure 19 – Bloc technique de température

5.4.1.2 Fonctions d'acquisition de la température

5.4.1.2.1 Connexion des capteurs

Le signal de processus est relié directement au module d'interface.

Il est possible de connecter la résistance thermique au moyen de 2, 3 ou 4 fils. La compensation est choisie par le paramètre SENSOR_CONNECTION.

Cet algorithme vérifie la liaison des capteurs et fait état d'une anomalie en cas de court-circuit ou de circuit ouvert. La vérification du câblage est activée/désactivée via la configuration (CHAN_CONFIG).

5.4.1.2.2 Plage des canaux

Cet algorithme sélectionne le type de capteur connecté à l'appareil. Selon la configuration (SENSOR_TYPE), il est nécessaire de différencier:

- la plage électrique (± 10 V, 0 V à 10 V, 0 V à 5 V, 1 V à 5 V, 0 mA à 20 mA ou 4 mA à 20 mA),
- le couple thermoélectrique,
- les sondes de température.

Le Tableau 4 donne un exemple de plusieurs types de capteurs.

Tableau 4 – Exemple de capteurs de température de type Sensor_Type

Symbole	Description
Type B	Platine, 30 % Rhodium/ Platine, 6% Rhodium
Type C	Tungstène, 5 % Rhénium/Tungstène, 26 % Rhénium
Type D	Tungstène, 3 % Rhénium/Tungstène, 25 % Rhénium
Type E	Chromel/Constantan
Type G	Tungstène/Tungstène, 26 % Rhénium
Type J	Fer/Constantan
Type K	Chromel/Alumel
Type L	Platinel 5355/Platinel 7674
Type N	Nicrosil/Nisil
Type R	Platine, 13 % Rhodium/Platine
Type S	Platine, 10 % Rhodium/Platine
Type T	Cuivre/Constantan
Pt50	Platine 50 Ω
Pt100	Platine 100 Ω
Pt200	Platine 200 Ω
Pt500	Platine 500 Ω
Pt1000	Platine 1 000 Ω
Ni10	Nickel 10 Ω
Ni50	Nickel 50 Ω
Ni100	Nickel 100 Ω
Ni120	Nickel 120 Ω
Cu10	Cuivre 10 Ω
Cu25	Cuivre 25 Ω
Cu100	Cuivre 100 Ω
NOTE La plage de températures peut être la plage par défaut du couple thermoélectrique ou de la sonde de température sélectionné(e) définie en dizaines de degrés (par exemple, –600 à +11 000 dizaines de °C pour une sonde 1 000 en nickel).	

5.4.1.2.3 Conversion analogique-numérique

Numérisation du signal analogique de mesure d'entrée, selon le paramètre défini lors de la configuration (ADCONV).

5.4.1.2.4 Essai

De nombreuses stratégies d'essai sont possibles, par exemple, commutation de l'entrée du capteur sur un signal de référence et vérification de la sortie du bloc technique par rapport à la valeur prévue, afin d'évaluer le fonctionnement correct.

Les résultats d'essai contribuent alors au traitement des informations de statut.