

Function blocks (FB) for process control —

Part 1: Overview of system aspects

ICS 35.200; 35.240.50

National foreword

This Draft for Development reproduces verbatim IEC TS 61804-1:2003.

This publication is not to be regarded as a British Standard.

It is being issued in the Draft for Development series of publications and is of a provisional nature. The main purpose of this Draft for Development is the harmonization of different views, models and starting points of end-users, system providers and device manufacturers. It was the reference document leading the discussions during the specification and the guideline for BS EN 61804-2:2004. It should be applied on this provisional basis, so that information and experience of its practical application may be obtained.

Comments arising from the use of this Draft for Development are requested so that UK experience can be reported to the international organization responsible for the Technical Specification. A review of this publication will be initiated not later than 3 years after its publication by the international organization so that a decision can be taken on its status at the end of its 3-year life. Notification of the start of the review period will be made in an announcement in the appropriate issue of *Update Standards*.

According to the replies received by the end of the review period, the responsible BSI Committee will decide whether to support the conversion into an international standard, to extend the life of the Technical Specification for another 3 years or to withdraw it. Comments should be sent in writing to the Secretary of BSI Technical Committee AMT/7, Industrial communications: Process measurement and control, including Fieldbus, at British Standards House, 389 Chiswick High Road, London W4 4AL, giving the document reference and clause number and proposing, where possible, an appropriate revision of the text.

A list of organizations represented on this committee can be obtained on request to its secretary.

Cross-references

The British Standards which implement international or European publications referred to in this document may be found in the *BSI Catalogue* under the section entitled “International Standards Correspondence Index”, or by using the “Search” facility of the *BSI Electronic Catalogue* or of British Standards Online.

This Draft for Development was published under the authority of the Standards Policy and Strategy Committee on 18 August 2004

Summary of pages

This document comprises a front cover, an inside front cover, the IEC TS title page, pages 2 to 127 and a back cover.

The BSI copyright date displayed in this document indicates when the document was last issued.

Amendments issued since publication

Amd. No.	Date	Comments

© BSI 18 August 2004

ISBN 0 580 44292 6

TECHNICAL SPECIFICATION

IEC TS 61804-1

First edition
2003-10

Function blocks (FB) for process control –

Part 1: Overview of system aspects



Reference number
IEC/TS 61804-1:2003(E)

CONTENTS

FOREWORD	5
INTRODUCTION	7
1 Scope	9
2 Normative references	11
3 Terms, definitions and abbreviated terms	11
3.1 General definitions	11
3.2 Definitions based on IA/IM-channel	17
3.3 Abbreviated terms	18
4 Engineering requirements	18
4.1 General	18
4.2 Requirements for design phase	18
5 Compatibility levels	19
5.1 General	19
5.2 Incompatibility	21
5.3 Coexistence	21
5.4 Interconnectability	21
5.5 Interworkability	21
5.6 Interoperability	21
5.7 Interchangeability	21
6 Functional requirements	22
6.1 General	22
6.2 System (or channel or device) status	22
6.3 Validity index (VI)	24
6.4 Signal processing	24
6.5 Measurement information processing	24
6.6 Device diagnostics and test support	26
6.7 Local interfaces attachment	27
6.8 Device (and system and channel) management	27
7 FB application requirements	31
7.1 System overview	31
7.2 Overview of basic FB types	33
7.3 FB requirements	35
7.4 Initial sets of FBs derived from I&C	37
7.5 FB environment requirements	41
7.6 Communications requirements	43
8 Additional requirements	43
8.1 Cooperation with external applications	43
8.2 Additional characteristic requirements	47
8.3 Conformance requirements	49
9 Device descriptive language	49
9.1 Background	49
9.2 Basic requirements	49
9.3 General requirements	51

Annex A (informative) Life cycle of the system	53
Annex B (informative) FB functional requirements: the user's view	77
Annex C (informative) Relation between IEC 61804 series and IEC 61499-1	94
Annex D (informative) Mapping of an analogue input FB to IEC 61499	106
Annex E (informative) FB requirement overview support	119
Annex F (informative) AME requirements	121
 Bibliography	 127
 Figure A.1 – Life cycle from the process circuit to the FRDs	 55
Figure A.2 – PFD composed of two process elementary operations	56
Figure A.3 – Control functions explicitly represented on the extended P&ID	57
Figure A.4 – Extraction of control functions from the extended P&ID	58
Figure A.5 – Requirements for control functions and ABs	59
Figure A.6 – Structured documentation for the requirements of the control functions	59
Figure A.7 – The four sections of a folio	60
Figure A.8 – Selection of a folio from the CHD	61
Figure A.9– Folio for detailed requirements of the control function "to control the pumps"	62
Figure A.10 – Example of FRDs	62
Figure A.11 – Functional requirements diagrams independent of I&C system implementation	63
Figure A.12 – The need for a standardized FB language	64
Figure A.13 – Library of standardized EFBs	65
Figure A.14 – From FRDs down to the I&C system and devices	66
Figure A.15 – Design of an actuation and its AB from off-the-shelf actuation devices	67
Figure A.16 – Designing internal behaviour of actuation AB from actuation devices off the shelf	67
Figure A.17 – Networking of the internal blocks inside the AB and with upstream and downstream EFBs and ABs of the control function	68
Figure A.18 – Actuation AB graphic symbol and implicit internal description	69
Figure A.19 – OFF/ON value actuation AB using the IEC 61499-1 system model	69
Figure A.20 – OFF/ON value actuation AB using the IEC 61499-1 system model	70
Figure A.21 – Example of a safe switch over 2/3 AB	70
Figure A.22 – Switchover 2/3 AB using the IEC 61499-1 system model	71
Figure A.23 – From a FRD to a programming scheme using IEC 61499-1 system model	72
Figure A.24 – Distribution of the internal blocks of a pump AB	72
Figure A.25 – Distribution the EFBs and ABs of the control function "to control the pumps" into the I&C architecture	73
Figure A.26 – Implementation of the control function into the I&C devices	74
Figure A.27 – Distribution of the control function "to control the pumps" using the IEC 61499-1 model	74
Figure A.28 – From FRDs to programming schemes	75
Figure B.1– System properties	79
Figure B.2 – IA/IM channel validation	80

Figure B.3 – Reference functional architecture for distributed automation systems	83
Figure B.4 – User view of the major component functions of each IAM function	84
Figure B.5 – IM channel/CMM interaction: user vision.....	86
Figure B.6 – Distributed platform using fieldbus: physical composition.....	86
Figure B.7 – Intelligent transmitter reference model	88
Figure C.1 – Structure of IEC 61499-1 FBs	95
Figure C.2 – Type specific aspects of IEC 61499-1 FBs.....	95
Figure C.3 – Implementation specific aspects	96
Figure C.4 – Distributed application and distributed operating system.....	96
Figure C.5 – Basic concepts of process control FBs.....	98
Figure C.6 – Functional components in process control FBs	99
Figure C.7 – IEC 61499-1 graphical representation of a process control FB (example)	99
Figure C.8 – Full structure of an EFB (example)	100
Figure C.9 – Relationship between IEC 61499-1, IEC 61804 series and other standardization activities.....	101
Figure D.1 – Summary of the parameter of analogue input blocks.....	107
Figure D.2 – Simulation, mode and status diagram of analogue input block	107
Figure D.3 – Conditions of mode and status generation	108
Figure D.4 – State machine of the analogue input block.....	108
Figure D.5 – Example for the analogue input block parameter.	112
Figure D.6 – AI FB graphical representation.....	113
Figure D.7 – AI FB ECC.....	114
Figure F.1 – Naming and addressing approach	123
Table 1 – Functionality features	20
Table 2 – Example of initial FB set.....	37
Table 3 – Example of common list of EFBs	38
Table C.1 – Aspects overview of IEC 61499-1 FBs	97
Table C.2 – Reference between IEC 61499-1 and IEC 61804 components.....	105
Table D.1 – Conditions and results of the actual mode calculation	109
Table D.2 – Conditions and results of the status calculation from the output parameter	109
Table D.3 – Parameter attributes for the analogue input block	110
Table D.4 – Process parameter description.....	110
Table D.5 – Alarm parameter description	111
Table D.6 – Simulation.....	112
Table E.1 – Allocation of requirements to architecture components.....	119
Table E.2 – Overview of related requirements.....	120

INTERNATIONAL ELECTROTECHNICAL COMMISSION

FUNCTION BLOCKS (FB) FOR PROCESS CONTROL –

Part 1: Overview of system aspects

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 provides no marking procedure to indicate its approval and cannot be rendered responsible for any equipment declared to be in conformity with an IEC Publication.
- 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.

The main task of IEC technical committees is to prepare International Standards. In exceptional circumstances, a technical committee may propose the publication of a technical specification when

- the required support cannot be obtained for the publication of an International Standard, despite repeated efforts, or
- the subject is still under technical development or where, for any other reason, there is the future but no immediate possibility of an agreement on an International Standard.

Technical specifications are subject to review within three years of publication to decide whether they can be transformed into International Standards. IEC 61804-1, which is a Technical Specification, has been prepared by subcommittee 65C: Digital communications, of IEC technical committee 65: Industrial-process measurement and control.

This standard cancels and replaces IEC/PAS 61804-1 published in 2002.

The text of this technical specification is based on the following documents:

Enquiry draft	Report on voting
65C/296/DTS	65C/310A/RVC

Full information on the voting for the approval of this technical specification 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.

IEC 61804 consists of the following parts under the general title *Function blocks (FB) for process control*

Part 1: *Overview of system aspects*

Part 2: *Specification of FB concept and electronic device description language (EDDL)*

The committee has decided that the contents of this publication will remain unchanged until 2007. At this date, the publication will be

- transformed into an International standard;
- reconfirmed;
- withdrawn;
- replaced by a revised edition, or
- amended.

INTRODUCTION

This Technical Specification is an end-user driven specification of the requirements of distributed process control systems based on Function Blocks (FB). This Technical Specification and its associated FB standard (IEC 61804-2) originate from the power-plant industrial sector. It is validated by applications in oil and gas, petrochemicals, pharmaceuticals and fine chemicals, pulp and paper, food and beverage, waste water treatment plants, steel milling and others. There will be other general requirement standards and associated specifications for other industrial sectors.

Present and future digital process control systems need to fulfil the following requirements:

- increase security and safety;
- reduce time to market;
- be supportable with available tools;
- reduce costs of development and support;
- minimize training costs;
- support integration of distributed control applications
- support integrated methodology for implementation;
- have increased maintainability, modifiability, agility, upgradeability, flexibility, ability to validate, accessibility, availability, compatibility of support tools, multi-vendor device/application compatibility, re-usability of knowledge and designs, re-usability of software components;
- be made up of digital devices that are compatible, interworkable, interconnectable interoperable and interchangeable with each other.

Process control systems are required to fulfil these requirements in terms of their architecture and their operation during all the phases of the life cycle. The accepted basic concept for the design process control system is to describe all necessary implementation-specific functions with FB. A FB is an encapsulation of data and algorithms to provide a specific function, which can be self-standing. Process control systems can involve many instances of many different FBs operating in an environment providing common services (for example, communications) and interfaces to other applications. See Figure 1.

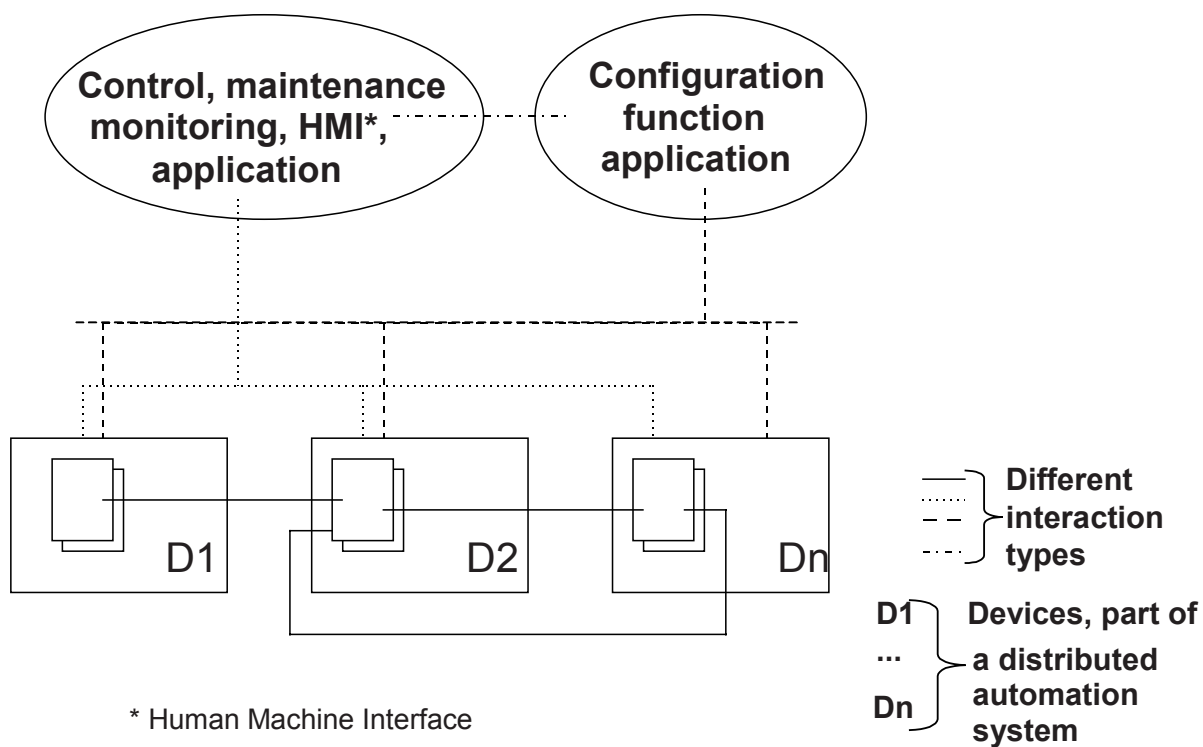


Figure 1 – Interactions of applications

FUNCTION BLOCKS (FB) FOR PROCESS CONTROL –

Part 1: Overview of system aspects

1 Scope

This part of IEC 61804 is a Technical Specification which provides *guidelines* for suppliers to meet evolving requirements for digital process control systems through which users can be assured of the compatibility, the interworkability, the interconnectability, the interoperability and the interchangeability of the devices they choose. This part gives the overall requirements. For better understanding, this part gives background information and examples in annexes.

This Technical Specification defines the requirements for FBs to provide control and to facilitate maintenance and technical management as applications which interact with actuators and measurement devices:

- control covers functions necessary to bring and hold the process at the desired behaviour;
- maintenance covers functions to acquire information about the state of the process equipment and the state of automation devices including their adjustments, for example, calibrate a sensor that has drifted;
- technical management deals with information for the optimization of the process.

It is concerned primarily with the economics of the process and plant equipment. This relates in particular to the evaluation of performance and reliability of specific items of plant or equipment for all the installed period and comparison of performance and reliability of items from different suppliers performing an identical function in the same operating environment. An example of performance is the number of cycles achieved before failure of two valves from different suppliers. This allows for the production of detailed and valid statistical analysis to support management decisions and plant equipment modifications.

A prerequisite for designing, implementing and operating a FB-based process control system is that the tools, the devices and other components follow the same architecture based on a common specification. The architecture is required to define the components of the systems, for example FB, device, data, data connections and more as well as relations between these components. The IEC 61499 series generic FB model on which this Technical Specification is related is able to provide these basic components for FBs for process control. One add-on to the IEC 61499 series is the specification of parameters and functions of FBs that are implementable in devices.

The architecture and the range of FBs that have to be specified are described in 7.4 contains a minimum set of FBs that will be required for the process industries. These are presented in TWO different clauses. One deals with “rich” FBs covering complex but common functions such as control loop (for example proportional, integral, differential – PID) required by the majority of the process industries. Another covers a set of elementary FBs (EFB) such as Boolean functions required to compose very specific and unique functionality.

FBs are used during the complete life cycle of process control systems but viewed from different aspects. This is covered in detail in Annex A. The process design starts with the Piping and Instrumentation Diagram (P&ID) which gives the requirements of the process and instrumentation from a purely functional point of view. From the P&ID, the desired behaviour of the process control system is extracted into a functional requirements diagram (FRD) without considering the detailed behaviour of the underlying devices. The bricks making up the FRD are application blocks (AB), the representation of the data and algorithms in the

design phase. After discussion between the process and automation engineers (end-user and system integrator), the FRDs are turned into detailed designs for the application via several design using devices available on the market together with interconnections and configurations of these devices. In this way a PID loop shown in via bubbles on a P&ID will be transformed into implementable FBs in specific field and/or control-room devices. It should be noted that many parts of the process industries, in particular those with many similar and relatively simple processes (for example, the water industry), do not use the concept or term FRD. They go directly from P&IDs to the implementable FBs and will use a variety of names to describe the process and the resulting design documents. The FRD approach is used here since it represents the most formal view of the design cycle and illustrates the use of FBs at the earliest of phases in the life cycle. Clause 4 summarizes the requirements from this life-cycle point of view.

This Technical Specification specifies a system (an industrial-process measurement and control system based on distributed FB application). A system is described stepwise in terms of architecture, models and the life cycle. The architecture is the "road map" which names the components and presents the structure of the system. The models describe the details of the components, i.e. their functions in the system. The life cycle makes visible how the components work together during their use in different phases of the lifetime, i.e. it makes the operation visible.

Figure 2 shows the different influences, basic specifications and technology support on IEC 61804 from the top-down and bottom-up point of view.

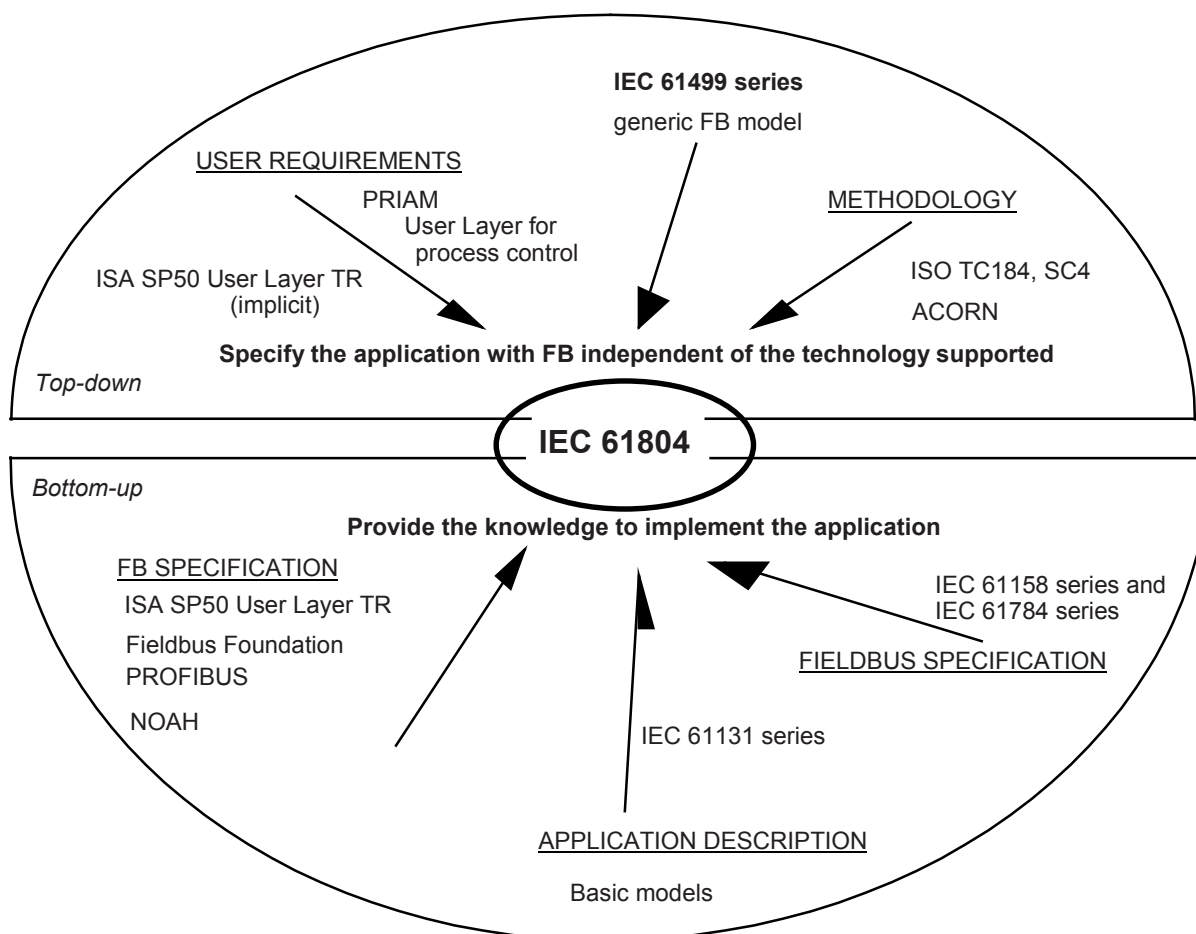


Figure 2 – Influences on IEC 61804

The influences are international standards and projects (PROFIBUS and Fieldbus Foundation¹), which relate to the same area as IEC 61804. These standards are either technology-independent ones supporting the top-down approach or dedicated to a certain technology, for example, programmable controller or fieldbus. Both together will build the basis of the standard specified by IEC 61804.

The main purpose of this part is the harmonization of different views, models and starting points of end-users, system providers and device manufacturers. It will be the reference document leading the discussions during the specification and the guideline for the readers of IEC 61804-2.

2 Normative references

The following referenced documents are indispensable for the application 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 61131-3:1992, *Programmable controllers – Part 3: Programming languages*

IEC 61499-1:2000, *Function blocks for industrial-process measurement and control systems – Part 1: Architecture*

IEC 61512-1:1997, *Batch control – Part 1: Models and terminology*

IEC 61784:2003, *Digital data communications for measurement and control – Part 1: Profile sets for continuous and discrete manufacturing – Fieldbus relative to use in industrial control systems*

IEC 61158 (all parts), *Digital data communication for measurement and control – Fieldbus for use in industrial control systems*

IEC 61804-2, *Function blocks for process control – Part 2: Specification of FB concept and Electronic Device Description Language (EDDL)* ²

EN 50170:1996, *General purpose field communication system*

3 Terms, definitions and abbreviated terms

3.1 General definitions

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

3.1.1

interface

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

[IEC 60050-351:1998, 11-19]

¹ PROFIBUS is the registered trade mark of PROFIBUS International (PI). PI is a non-profit trade organization to support the fieldbus PROFIBUS. This information is given for the convenience of users of this Technical Specification and does not constitute an endorsement by CENELEC of the trademark holder or of any of its products. Compliance to this profile does not require use of the trade name PROFIBUS. Use of the trade name PROFIBUS requires permission of the trade-name holder.

Foundation Fieldbus is the trade name of the consortium Fieldbus Foundation. This information is given for the convenience of users of this Technical Specification and does not constitute an endorsement by CENELEC of the product named. Equivalent products may be used if they can be shown to lead to the same results.

² To be published.

3.1.2

system

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

[IEC 60050-351:1998, 11-01]

NOTE 1 Such element may be material objects and concepts as well as the results thereof (for example forms of organization, mathematical methods, and programming languages).

NOTE 2 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 system considered.

3.1.3

data type

set of values together with a set of permitted *operations*

[ISO 2382 series]

3.1.4

data connection

association between two *function blocks* for the conveyance of *data*

[IEC/PAS 61499-1:2000, 1.3.2.22]

3.1.5

data

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

[ISO modified³]

3.1.6

functional unit

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

[ISO 2382 series]

3.1.7

hardware

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

[ISO modified³]

3.1.8

mapping

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

[ISO 2382 series]

3.1.9

parameter

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

[ISO 2382 series]

³ The notation [ISO modified] following a definition indicates that the definition is taken from the "ISO/AFNOR Dictionary of Computer Science" and has been modified.

3.1.10**algorithm**

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

[IEC/PAS 61499-1:2000, 1.3.2.5]

3.1.11**application**

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

[IEC/PAS 61499-1:2000, 1.3.2.6]

NOTE An application may be distributed among resources, and may communicate with other applications.

3.1.12**application block**

design pattern which is used in an FRD to represent one or multiple (rich) functions blocks

3.1.13**attribute**

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

[IEC/PAS 61499-1:2000, 1.3.2.7]

NOTE The formal description of attributes is to specify interoperability. IEC 61499-1 does not specify certain attributes like FB type information. IEC 61499-1 gives the general rules to define the attributes and IEC 61804-2 specifies the attributes for process control as other groups may specify their own. Rules should be able to prevent non-unique attribute names.

3.1.14**configuration (of a system or device)**

step in system design: selecting *functional units*, assigning their locations and defining their interconnections

[IEC/PAS 61499-1:2000, 1.3.2.17]

3.1.15**device**

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

[IEC/PAS 61499-1:2000, 1.3.2.26]

3.1.16**device management application**

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

[IEC/PAS 61499-1:2000, 1.3.2.27]

3.1.17**elementary FB**

FB or *function* which provide logico-mathematical functionality with exception handling

NOTE There can be a difference between an EFB used in the FRD and the FB in the operating application

3.1.18**entity**

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

[IEC/PAS 61499-1:2000, 1.3.2.28]

3.1.19

event

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

[IEC/PAS 61499-1:2000, 1.3.2.29]

NOTE The execution of an algorithm may make use of variables associated with an event.

3.1.20

exception

event that causes suspension of normal *execution*

[IEC/PAS 61499-1:2000, 1.3.2.35]

3.1.21

execution

process of carrying out a sequence of *operations* specified by an *algorithm*

NOTE The sequence of operations to be executed may vary from one *invocation* of a *function block* instance to another, depending on the rules specified by the FB *algorithm* and the current values of *variables* in the FB data structure.

[IEC/PAS 61499-1:2000, 1.3.2.36]

3.1.22

function

specific purpose of an *entity* or its characteristic action

[IEC/PAS 61499-1:2000, 1.3.2.42]

3.1.23

FB (FB instance)

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

NOTE Typical operations of a FB include modification of the values of the data in its associated data structure.

[IEC/PAS 61499-1:2000, 1.3.2.43]

3.1.24

implementation

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

[ISO modified³]

3.1.25

input variable

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

NOTE An input parameter of a FB, as defined in IEC 61131-3, is an input variable.

[IEC/PAS 61499-1:2000, 1.3.2.48]

3.1.26

instance

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

[IEC/PAS 61499-1:2000, 1.3.2.49]

3.1.27**instance name**

identifier associated with and designating an *instance*

[IEC/PAS 61499-1:2000, 1.3.2.50]

3.1.28**instantiation**

creation of an *instance* of a specified *type*

[IEC/PAS 61499-1:2000, 1.3.2.51]

3.1.29**internal operations (of a FB)**

operations associated with an *algorithm* of a *FB*, with its *execution* control, or with the functional capabilities of the associated *resource*

[IEC/PAS 61499-1:2000, 1.3.2.52]

3.1.30**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*.

[IEC/PAS 61499-1:2000, 1.3.2.53]

3.1.31**invocation**

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

[IEC 61131-3, modified]

3.1.32**management FB**

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

[IEC/PAS 61499-1, 1.3.2.56]

3.1.33**management resource**

resource whose primary *function* is the management of other *resources*

[IEC/PAS 61499-1:2000, 1.3.2.57]

3.1.34**model**

representation of a real world process, *device*, or concept

[IEC/PAS 61499-1:2000, 1.3.2.58]

3.1.35**operation**

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

3.1.36

output variable

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

NOTE An output parameter of a FB, as defined in IEC 61131-3, is an output variable.

[IEC/PAS 61499-1:2000, 1.3.2.60]

3.1.37

resource

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

NOTE 1 The resource defined in IEC 61131-3 is a programming language element corresponding to the resource defined above.

NOTE 2 A device contains one or more resources.

[IEC/PAS 61499-1:2000, modified]

3.1.38

resource management application

application whose primary function is the management of a single *resource*

[IEC/PAS 61499-1:2000, 1.3.2.66]

3.1.39

scheduling function

function which selects *algorithms* or *operations* for *execution*, and initiates and terminates such execution

[IEC/PAS 61499-1:2000, 1.3.2.70]

3.1.40

service

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

[ISO/IEC 7498-1, modified]

3.1.41

software

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

[ISO modified³]

3.1.42

transaction

unit of *service* in which a request and possibly *data* are conveyed from a requester to a responder, and in which a response and possibly data may also be conveyed from the responder back to the requester

[IEC/PAS 61499-1:2000, 1.3.2.79]

3.1.43

type

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

[IEC/PAS 61499-1:2000, 1.3.2.80]

3.1.44**type name**

identifier associated with and designating a *type*

[IEC/PAS 61499-1:2000, 1.3.2.81]

3.1.45**variable**

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

NOTE 1 The values of a variable are usually restricted to a certain data type.

NOTE 2 Variables may be classified as input variables, output variables, and internal variables.

[ISO modified³]

3.2 Definitions based on IA/IM-channel**3.2.1****actuation (measurement) channel**

sum of all the items necessary to perform each actuation (measurement) as users need it. The physical composition extends from the attachment to the process, to the valve, motor, actuator (sensor, transmitter), the network, and the complementary processing in the computers

NOTE The expression IA/IM-channel means intelligent actuation/measurement solution of all the requirements for each actuation/measurement necessary. Intelligent here means provided with all the functionalities as users need them.

3.2.2**system (or channel or device) status**

actual health (or condition) of the related item (system or channel or device). In other words it is defined at the several levels of system distribution: the system as a whole, each IA/IM-channel of the system, each device composing the channel

NOTE A detailed explanation is given in 6.2.

3.2.3**validity index (VI)**

qualifier of the information to which it is added. It can be seen as a quality index

NOTE A detailed explanation is given in 6.3.

3.2.4**measurement uncertainty**

parameter associated with the actual result of a measurement, which characterizes the dispersion of the values that could reasonably be attributed to the measured

NOTE 1 The word "uncertainty" means "doubt", and thus, in its broadest sense, "uncertainty of measurement" means the extent of doubt about the exactness or accuracy of the result of a measurement.

NOTE 2 The uncertainty may be, for example, a standard deviation or the width of a confidence interval.

NOTE 3 The uncertainty can be expressed with data which can be treated mathematically, so that the uncertainty of an indirect measurement can be calculated if the uncertainty of the several component direct measurements are known.

3.2.5**profile**

set of one or more base standards and/or ISPs, and, where applicable, the identification of chosen classes, conforming subsets, options and parameters of those base standards, or ISPs necessary to accomplish a particular function

[ISO/IEC 10000-1]

NOTE ISPs may contain normative references to specifications other than International Standards.

3.3 Abbreviated terms

AB	Application Block
AME	Application Management Entity
CHD	Control Hierarchy Diagram
CMM	Control, Maintenance and Technical Management
DCS	Distributed Control System
DFBAP	Distributed FB Application Process
EFB	Elementary FB
FB	Function Block
FRD	Functional Requirement Diagram
HMI	Human Machine Interface
I&C	Instrumentation and Control
IA/IM-channel	Intelligent Actuation/Intelligent Measurement-channel
ISP	International Standard Profile
MCC	Motor Control Centre
MIB	Management Information Base
MGT	Management
PFD	Process Flow Diagram
PID	Proportional, Integral, Derivative
PRIAM	Prenormative Requirements for Intelligent Actuation and Measurement
P&ID	Piping & Instrumentation Diagram
SCADA	Supervision Configuration And Data Acquisition
SM	System Management
ST	Structured Text
VI	Validity Index

4 Engineering requirements

4.1 General

This clause expects from the reader certain knowledge of engineering of a distributed FB system. See Annex A for background information. It is designed to give the reader an overview of the life of a FB-oriented control system from conception through design and engineering and onto operation support and maintenance. Each of these phases has different environments for the actual FB entities and their own specific requirements.

4.2 Requirements for design phase

- a) To identify a FB as part of a particular functional requirement diagram (FRD) or a certain application blocks (AB) in a distributed field device system, a FB is required to be able to carry an identification of a particular FRD block.

NOTE This is required to be a parameter and may be called STRATEGY. On the basis of this parameter an engineering system may be able to identify the distributed FBs which are combined in a FRD as one FB for reverse engineering purposes.

- b) To identify a FB type a type name is required. This type name is required to be unique within a library but can have multiple instances. An engineering tool may navigate by this information to an online help file.

- c) To identify a device that hosts one or more FBs a device type identifier is required that allows a link of a device description type to this device type. This device type identifier is required to be based on a profile and not on a vendor-specific type to support interchangeability during design phase.
- d) Graphical representation of a device type is required to be referenced within device description as an option. That icon is not used for FB chart nor for P&ID, it is only usable within a topological view of field devices. There is no requirement of this representation within this standard.
- e) To identify the elements of control hierarchy diagram (CHD), which are comparable with IEC 61512-1 BATCH processes, some parameters are required that are defined according to this batch standard.

Example:

FBs carry these parameters. There is no algorithm necessary within a FB. The EFBs do not carry these parameters.

1. Batch ID (BATCH_ID)
 2. No. of recipe unit procedure or of unit
 3. No. of recipe operation
 4. No. of recipe phase
- f) FB invocation is required to be supported by appropriate scheduling mechanisms in cyclic manner.
 - g) There is no requirement for field devices and their FBs based on ABs.
 - h) EFBs are required to be defined and gathered within a library. An EFB is a repetitive logic-mathematics treatment, which is embedded into a FB or a function as defined in the IEC 61131 series. An EFB is a processing module restricted to the process control domain. An EFB cannot be split. All FBs (not functions) are required to carry a Library Name. The combination of a FB name and a library name is a unique identifier of a FB. That means to identify this FB type when instantiated in a device mixed with FBs of branch or application-specific libraries it can be identified as an IEC EFB. This is necessary for version control, link to online help and so on. A formal internal description of the behaviour of the EFB is required to be specified in an IEC 61131-3 language, for example, structured text (ST), ladder diagram or instruction list. Missing elements within ST have to be programmed in several statements and may lead to an additional parameter, for instance, exception handling results.
 - i) For human machine interface, it is necessary to select the dynamic parameter of an EFB, which builds the information base to visualize the process, to allow operation by changing a parameter and to store data. For this selection, there is no syntax defined in the IEC 61131 series. As an addition to IEC 61131 for process control, attributes within the comment field of definitions within ST are defined.
 - j) For maintenance and technical management tools, it is necessary to specify or predefine attributes to the parameter and FBs of EFB. For this selection there is no syntax defined in the IEC 61131 series. As an addition to IEC 61131 for process control, attributes within the comment field of definitions within ST are defined.

5 Compatibility levels

5.1 General

There are certain levels of compatibility and according levels of cooperation between FB-based devices. The levels are dependent on well-defined communication and application features. See Figure 3 and Table 1.

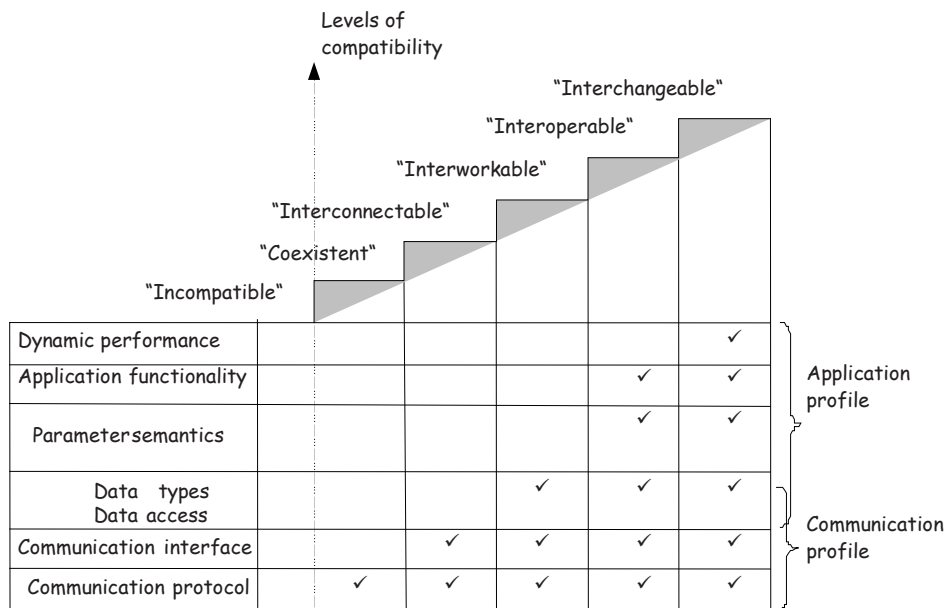


Figure 3 – Levels of functional device compatibility

The following main features are used for the definition of compatibility levels.

Table 1 – Functionality features

Feature	Description
Communication profile	
Communication protocol	This feature is defined by all protocols of layer 1 to 7 of the 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 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 profile	
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, 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. The IEC 61804 standard 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.

5.2 Incompatibility

Inability of two or more devices to work together in the same distributed application.

NOTE Incompatibility can result from differences in application functionality, data semantic, data types, communications interface, or even communications protocols used by the affected devices. Incompatible devices may even interfere with, or prevent, each other's proper communication or functioning (possibly even destructively), if placed in the same distributed application network.

5.3 Coexistence

Ability of two or more devices, regardless of manufacturer, to operate independently of one another in the same communications network, or to operate together using some or all of the same communications protocols, without interfering with the functioning of other devices on the network.

NOTE 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

5.4 Interconnectability

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

NOTE The devices allow data exchange without agreements about the data types. A data type conversion may 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

5.5 Interworkability

Ability of two or more devices, regardless of manufacturer, to support the transfer of device parameters between devices having the same data types of the data inputs, data outputs and parameters.

NOTE If a device is replaced with a similar one of a different manufacture, it can be necessary to reprogram the application. The distributed application must be designed to accommodate the unique functionality and dynamic responses of the interworkable devices used in the implementation.

5.6 Interoperability

Ability of two or more devices, regardless of manufacturer, to work together in one or more distributed applications. 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.

NOTE 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.

5.7 Interchangeability

Ability of two or more devices, regardless of manufacturer, to work in one or more distributed applications using the same communications protocol and interface, with the data and functionality of each device so defined that, if any device is replaced with another of the interchangeable devices, any distributed applications involving the replaced device will continue to operate as before the replacement, including identical dynamic responses of the distributed applications.

6 Functional requirements

6.1 General

The functional requirements, which are expressed in the following subclauses, are defined at an abstraction level suitable to be well understood by both end-users and vendors.

This abstraction level differs from the FB abstraction level, which is directly used by vendors and system integrators.

The correspondence between the functional requirements defined in this part and the FB concept defined in IEC 61804-2 is not necessarily one-to-one. It may be one-to-one, one-to-many or many-to-one.

Annex B is required to be read for a more complete understanding and use of the following requirements.

The description of the device functional requirements explicitly addresses this functionality of a measurement channel which is located in the field device (for example, sensor, transmitter, etc.). It is required that similar requirements are valid for the field device part of the actuation channel (valve, actuator, positioner, etc.).

6.2 System (or channel or device) status

In principle, there are three main values of the status (as shown in Figure 4), whatever the qualified item:

- a) validated capability;
- b) acceptable degradation(s);
- c) out of service (unacceptable degradation).

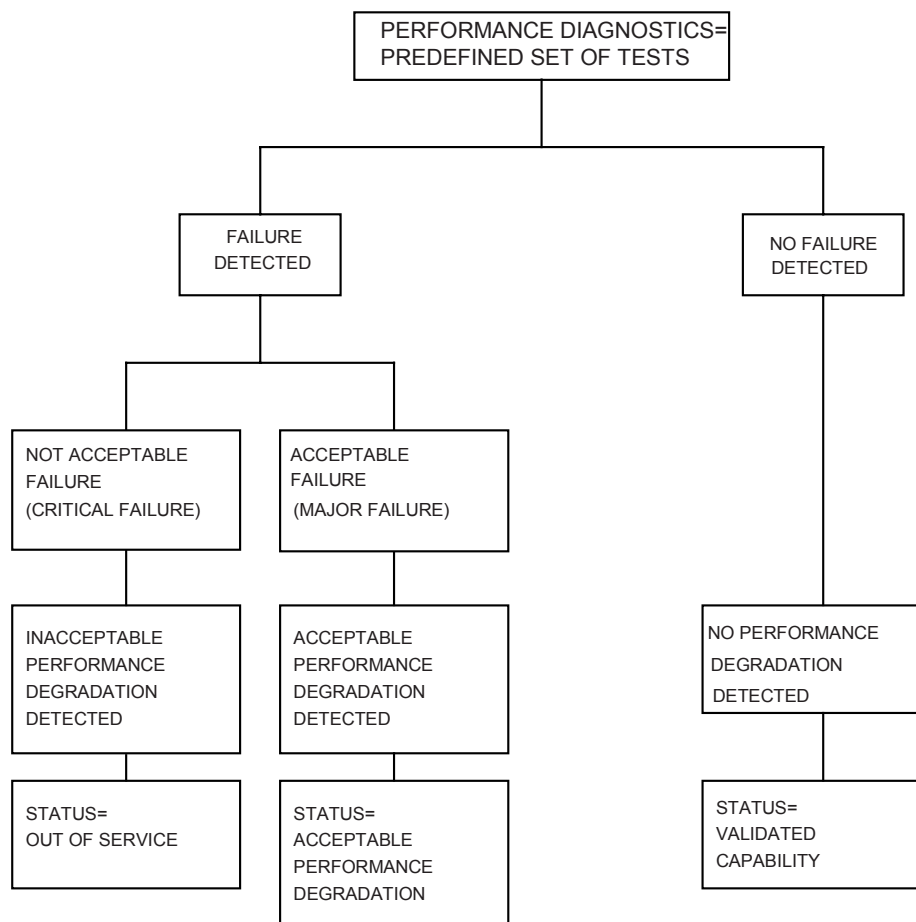


Figure 4 – Device (or channel or system) status

The degradation(s) are defined as (a number of) performance degradations which are considered acceptable by the user of the system (or device) according to his specific needs and the supporting diagnostics.

For practical purpose users need two kinds of information on this status: only one synthesized status and several detailed statuses, each one as required for each task of the several operators.

a) Synthesized status

The synthesized status expresses the actual degree of capability of the item (device or system or channel) to perform the required functions. This information is intended to support immediate actions specified by the users for the control, maintenance and technical management of the systems.

b) Detailed status

The detailed status information makes explicit the detailed diagnostic information needed for the maintenance and technical management of the system (or device). In the user vision at least three needs are required to be covered, as follows:

- 1) detailed information to document the behaviour of each replaceable component as needed to guide the intervention on the component;
- 2) information which helps diagnose the faulty part and provides all the details which are useful to properly repair that part (this diagnosis usually requires more details to be documented);

- 3) as 1) and 2), but here the details are those needed to properly judge the behaviour of each component for technical management purpose.

6.3 Validity index (VI)

The quality is a relative quality, judged on the basis of achievement of the predefined requirements for the properties of the channel generating the information. A basic set of quality criteria to be considered is

- a) accuracy and uncertainty criteria (see ISO Guide to the expression of uncertainty in measurement),
- b) timeliness criteria,
- c) good/bad criteria,

In practical implementations, the criteria used to generate the VI have to be explicitly stated.

The VI is intended to allow immediate and correct use of the qualified information.

The VI is generated in real time taking into account the results of the actual diagnostic and validation of the whole channel that has produced the information. Therefore the VI explicitly makes the user of the information aware of the actual situation (deterministic VI) resulting at varying operating conditions, normal and abnormal, after an error detection and correction.

6.4 Signal processing

Functions producing raw measurements and data as inputs for measurement information processing and device management.

- a) Sensor limit thresholds (diagnostics dedicated to fault management support).
- b) Sensor ranging.
- c) A/D converter control.
- d) Data quality determination (VI associated to the raw measurement after the A/D converter due to the results of hardware diagnostics).
- e) Influencing quantity compensation (hardware).
- f) Process attachment and sensor diagnostics (run time auto tests).
- g) Support for maintenance tests on demand.

6.5 Measurement information processing

6.5.1 General

This processes information provided by the measurement signal processing.

The main purpose of the information processing component is to provide the higher levels of the measurement channel with the processed measurement closely associated with its VI. The complete semantics of the measurement and validation information delivered by the measurement device is required to be known. (All the functional units that have taken part in its elaboration are required to be known.)

Validation is an essential part of the measurement processing and determines the degree of credibility of the measurement delivered by each step in the measurement processing. The validation of the information achieved incrementally at each step of elaboration is a fundamental part of the measurement itself in order to allow a correct use of the measurement information to which it refers.

Another important aspect is that, through the convenient use of the VI and of the device status, the device can work under degraded conditions, leaving to higher levels of management the decision on the use of the data. In this sense the fault tolerant behaviour of the transmitter consists of the diagnosis of the fault and in the indication on how much this impairs the measurement (VI) and the device capabilities (device status).

The functions that have to be covered by the measurement information processing are given hereafter.

6.5.2 Fault-tolerant behaviour processing

Fault identification and confining. Information contribution to the processing of the corresponding measurement VI.

6.5.3 VI processing

a) Measurement validation

This processing has to be accomplished along with each phase of the measurement processing and not separately, in such a way to guarantee mutual consistency between data and VI.

It may include treatment of time redundancy based on comparison among a number of successive samplings of the same quantity successively taken within the sampling period defined for the application.

b) Uncertainty processing

The parameter (uncertainty) may be, for example, a standard deviation (or a given multiple of it), or the half-width of an interval having a stated level of confidence. Uncertainty of measurement comprises, in general, many components. Some of these components may be evaluated from the statistical distribution of the results of series of measurements and can be characterized by experimental standard deviations. The other components, which also can be characterized by standard deviations, are evaluated from assumed probability distributions based on experience or other information. It is understood that the result of the measurement is the best estimate of the value of the measurand, and that all components of uncertainty, including those arising from systematic effects, such as components associated with corrections and reference standards, (that) contribute to the dispersion.

6.5.4 Measurement channel processing functions

- a) Scaling
- b) Linearization
- c) Influencing quantity compensation (software)
- d) Filtering
- e) Dumping
- f) Engineering units conversion
- g) Limit thresholds (regarding process alarm application) and all relevant data treatment
- h) Transformed measurement processing
- i) Measurement time stamping
- j) Measurement safe values (generally by replicating the last valid value for an assigned number of samplings)
- k) Measurement trend processing

NOTE There may exist additional functions.

6.6 Device diagnostics and test support

6.6.1 General

The following subclauses concern functions producing information related to device status.

6.6.2 Device diagnostics

- a) Power-on self-tests
- b) Run-time self-tests

6.6.3 Maintenance-related processing signals and counts all events useful to perform device degradation analysis

The target is to allow the implementation of algorithms in the maintenance systems to foresee the degradation of a device or of a part of it. The manufacturer has to define which parameters are significant and include in the device the necessary FBs that recognize these situations and report their occurrence to the maintenance system and to the technical management system.

Examples of these parameters that have been found useful include

- a) the time of operation spent outside nominal working conditions (defined through for example temperature limits, pressure limits, etc.),
- b) the number of abnormal events (such as pressure spikes for a P, DP transmitter),
- c) the number of electrical shocks of defined characteristics.

NOTE There may exist additional parameters.

6.6.4 Support for tests on demand (remotely or locally issued)

This support, as needed for actuation and measurement remote or local tests (for example calibration), is divided in the two following categories:

- intrusive tests: the execution of these tests interferes with the normal operation of the device/channel (for example it requires that the attachment with the process is switched off);
- non-intrusive tests: the execution of these tests does not interfere with the normal operation of the device/channel.

According to the requested tests, the necessary authorization is needed, and the device/channel have to be put in the appropriate operating mode.

The required functionalities are

- a) intrusive device tests execution:
 - 1) support for calibration procedures (both for primary and transformed measurements)
The aim of this function is to give the maximum help to the maintenance operator responsible for device calibration. As he will perform calibrations on many different devices of many different manufacturers the help he needs is achieved through
 - use of standard terminology where already available,
 - application of a standard procedure when applicable,
 - direct availability of calibration reports and results in electronic form in order to be transferred to the maintenance system through the network or electronic support.
 - 2) measurement processing tests against reference inputs;
 - 3) process attachment intrusive diagnostics tests;

- b) non-intrusive device tests execution:
 - 1) process attachment non-intrusive diagnostics tests;
 - 2) measurement processing tests against reference inputs (if executable at run time during time periods agreed upon by the control operator; for example, test of a redundant part at a time);
- c) tests results retrieval:
 - 1) retrieval of device tests results (information about self-diagnostics, test results and maintenance interventions);
 - 2) direct availability of actual information about self-diagnostics, test results and maintenance interventions in electronic form in order to be transferred to the maintenance system through the network or electronic support.

6.7 Local interfaces attachment

Availability of a local access, through a suitable terminal, to

- a) access rights management function (see 6.8.10),
- b) pre-defined list of information,
- c) pre-defined command inputs,
- d) required local access to support field operator interventions.

6.8 Device (and system and channel) management

6.8.1 General

The following functions are described as the expected results of the cooperative actions of the management functions of the device and those of the system manager; of course, the contribution of the device agent manager belongs to the device functional units.

In this perspective, a channel is a subsystem.

6.8.2 Device identification function

- a) Vendor related device information retrieval. Information as needed by the several device-users interacting with the device along its life cycle (for example, device components part number, materials, software and hardware versions, working ranges, etc.).
- b) User-related device information retrieval (complementary information usually written during commissioning, for example, device tag, installation date, etc.).

The above information is expected to be written in the device in the relevant phases of the life cycle by the vendor operator and the commissioning operator using the following functionality:

- a) vendor-related device information modification;
- b) user-related device information modification.

6.8.3 Configuration function (and help for system configuration)

When the transmitter is programmable, the software for the specific functions in the processing and acquisition parts of the transmitter has to be downloaded from the configuration system. For instance, if a DP transmitter is used as a level or flow transmitter a download of the specific software is needed. Of course, this logical action may have several practical implementations (for example, EPROM replacement, selection of pre-installed functions, etc.).

Configuration will include all the software components needed for supporting the different tasks assigned to the transmitter such as measurement, diagnostics, test routines, etc. A specific user requirement is the availability of reports on demand, which allow the operator to check the actual configuration (consistency check versus a specified configuration). Along the device life cycle the different users will take advantage of the following functionality:

- a) vendor configuration modification;
- b) user configuration modification (selection of needed functions between the available ones);
- c) configuration retrieval;
- d) read available functions;
- e) read vendor configuration revision;
- f) read user configuration revision;
- g) read selected (active) functions.

6.8.4 Parameterization function

These functions consist in setting parameters to completely define generic functions in order to satisfy the plant application constraints.

The processing functions parameterization depends on the functions selected/downloaded during configuration; parameters are required to be loaded in non-volatile parameter support and their integrity is required to be continuously checked.

Parameters to be set are, for example, the range, the offset and the engineering units of the measured variable, warning and alarm levels, sampling frequency, filters time constants, etc. A specific user requirement is the availability of reports on demand, which allow the operator to check the actual parameterization (consistency check versus a specified parameterization).

Device parameters are divided into the following two categories:

- intrusive parameters: the modification of these parameters interferes with the normal operation of the device;
- non-intrusive parameters: the modification of these parameters does not interfere with the normal operation of the device.

According to the requested parameter modification request, the necessary authorization is needed, and the device/channel has to be put in the appropriate operating mode.

The required functionality is

- a) intrusive device parameterization modification (single parameter/for defined groups of parameters),
- b) non-intrusive related device parameterization modification (single parameter/for defined groups of parameters),
- c) device parameterization retrieval.

6.8.5 Measurement management function

This function is part of the device management. Its relationship with the system management is clarified hereafter.

This function produces proper execution events as needed for each kind of measurement execution timing. The needed synchronization among the several device managers is produced by the system manager, by means of suitable network mechanisms.

It produces the following services according to the needs.

- a) Execution event for cyclic asynchronous measurements (i.e. not synchronized with a system time reference)
This is the execution event needed for the measurement to be cyclically acquired asynchronously with respect to its use and to be cyclically made available to all its users within the times necessary to guarantee data consistency for the specific application
- b) Execution event for cyclic synchronous measurements (i.e. synchronized with a system time reference)
This is the execution event needed for the measurement to be cyclically acquired synchronously with respect to its use following the reception of sync. Command and to be made available to all its users within the times needed for the specific application
- c) Cyclic measurements with asynchronous start and stop
This is the execution event needed for the measurement to be cyclically acquired asynchronously with respect to its use and to be cyclically made available to all its users within the times necessary to guarantee data consistency for the specific application
Acquisition and distribution of measurement are not continuous but are started and stopped following a specific command or an asynchronous event
- d) Execution event for measurements on demand
The measurement is acquired following a specific command issued by the user of the measurement. The measurement will be made available after a command of data request issued by the user of the measurement
- e) Execution events for detection of logic values variations
- f) Event notification of value modification of binary data

6.8.6 System time management function

The system time management function is aimed at coordinating the several time managers of each device.

In some applications it is, for example, required to time-stamp measurements and/or events with an absolute time information. A system (master) clock is required and all the devices are required to have an internal clock which is required to be maintained synchronized with the master clock. The accuracy required by the specific application is required to be guaranteed through a well-defined combination of clocks accuracy and a synchronization operating through the network.

It coordinates the managers of the several devices, as needed, to share

- a) absolute time, with the requested time accuracy (typically 1 ms),
- b) relative time, with the requested time accuracy (typically at least 1 ms),
- c) time synchronization command, sent by the master clock at a defined rate as needed to guarantee the requested time accuracy.

6.8.7 Timeliness verification support

This function checks that data transmitted, or received, via the communication interface are always within the specified timeliness.

This processing contributes to the processing of the data validation.

6.8.8 Device failure management

The device failure management is aimed at supporting proper management of device failures.

a) Detection of failures

It takes into proper account all the detailed diagnostic information made available by the device diagnostics and test support. It uses this information to carry out activities of failure detection and to generate the device statuses (device detailed status and device synthesized status) as needed for specified internal actions and to be treated by the measurement channel status management function. This latter function will process all this information to provide the different users with the information required for the specific application (it is an end-user adaptation).

b) Device decisions (fault-tolerant procedures)

It carries out internal procedures aimed to confine detected faults (to prevent local failures from causing system failures) following requests received from a higher level of management or to be automatically activated inside the device.

It includes procedures to treat possible redundancy of sensors and attachment to the process.

6.8.9 Operation management function (extended for access rights management)

This is a system management function aimed at managing the transitions among the several operating modes. In particular, it manages the qualification transfer procedures between the different human or automatic operators. This is to support procedures of intervention in the field devices in order to realize intervention plans that are agreed among the human organizations. Of course, the device management is required to cooperate with the system management to achieve the result.

All the paper procedures that are used in a plant in order to achieve this function may now be simplified because of the new communication means between the devices and the intelligence embedded in the devices themselves.

It is required that for each needed interaction between the operators and the distributed system the device management enables the needed device functions and disables all the others. A management procedure has to be established in order to give the different operators the qualifications to access functions or devices as defined on the basis of agreed needs. This is intended to prevent unauthorized action and possible errors (see qualification access).

Typical operating modes, defined according to the several needed interactions, to be supported by the system management are

- a) control remote auto (associated operator: control device),
- b) control remote manual (associated operator: control remote operator),
- c) control local (manual) (associated operator: control local operator),
- d) maintenance remote (associated operator: maintenance remote operator),
- e) maintenance local (associated operator: maintenance local operator),
- f) commissioning remote (associated operator: commissioning remote operator),
- g) commissioning local (associated operator: commissioning local operator),
- h) parameterization (associated operator: parameterization operator),
- i) self (associated operator: none; the device has lost the fieldbus connection).

NOTE There may exist additional operating modes.

6.8.10 Access rights management function

The access right management function is the qualification check on every access to normal device functions.

In each operating mode the devices themselves qualify which operators are allowed to perform on them certain pre-defined actions.

It is obvious, for example, that it is extremely important to disable the write access to the device for configuration, parameterization and tests on demand purposes when the measurement processing functions is required to be active (such as in control operating modes).

This device requirement is the complement of a corresponding requirement at the system level.

7 FB application requirements

7.1 System overview

The preceding clauses describe what are the requirements to a system from the end-user point of view. This clause focuses on the architectural point of view of a system.

Process control systems are combinations of applications controlling the process (the FB based applications), applications carrying out maintenance, monitoring, HMI, commissioning functions and applications for the configuration functions of the application software and hardware in a device, also known as system management functions see Figure 5. A device performing the process control in terms of FB application has to provide all functionality to interact with the FB devices each other and the external applications in an interoperable way. Therefore the components of the system and the devices, including their relationships, have to be defined. This subclause covers only the FB-based devices. The device and system architecture is based on the definitions in IEC 61499-1.

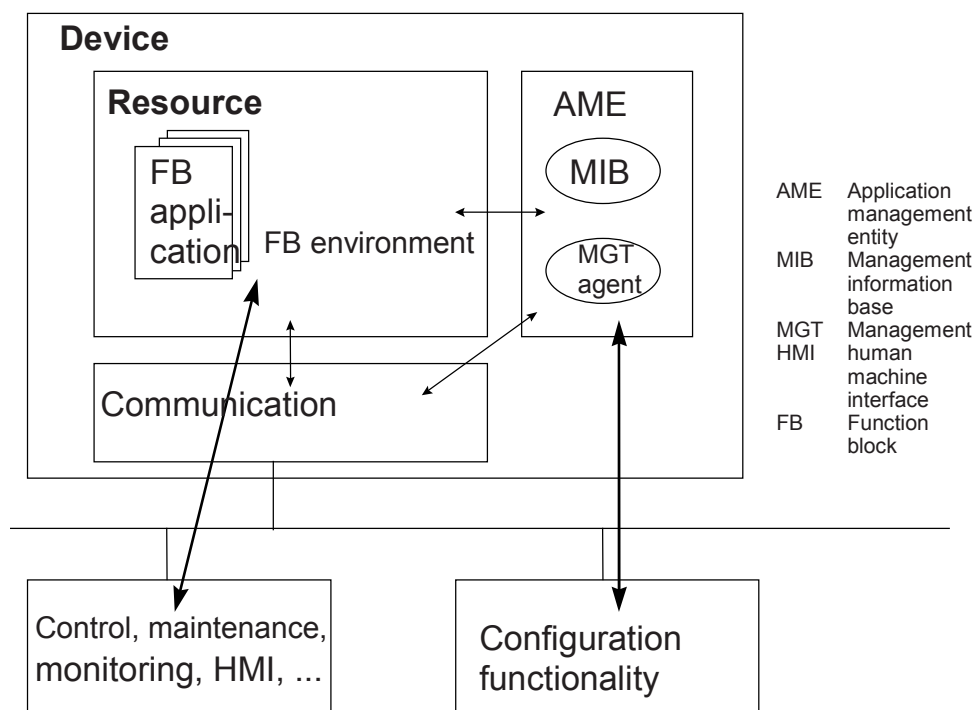


Figure 5 – Components of a FB device

In a real device or resource, there are only data and function (program) codes as well as interfaces to the communication, process and HMI. Only via the communication system, by a configuration or design tool, these data and program code are seen as FBs, management agents or data connections. The specification of the device model is an external view to the device implementation. The device model describes details of the overall system architecture. The device model is an abstraction of the real device.

- The basic components of a device performing FB applications are the FBs (see Figure 6) .
- FBs are composed of data and event inputs and outputs, internal data, contained data and algorithms. They have the following general requirements.

NOTE The term FB is used as a synonym for blocks in general (function, device and technology blocks) as well as for FB in particular.

Further description is given in 7.3.

a) FB environment

FBs are embedded in an operating environment shown as FB environment. The FB environment performs special process control functions, which are a priori available application portions without any programming design activities, for example alarms. These FB environment functions are used by external applications independent of, i.e. parallel to, the FB application (see 7.5).

NOTE Common functions and data of the FB environment can be encapsulated in FBs too, for example trending.

b) Application management entity (AME)

Many process control devices need adoption to the used hardware configuration from the functional point of view. For instance, the available FBs in a modular device have to be configured or the model version with or without an additional input for measurement compensation. This causes the addition or deletion of FBs. These functions will be initiated by the so-called AME, i.e. addition functions in a device modifying their application software. The IEC 61804 series does not define the AME. This shall be done within the framework of the IEC 61158 series. Annex F describes the management requirements .

NOTE The AME contains the so-called system and network management functionality.

c) Communication

The data transfer between FBs, the interactions between FBs/FB environment and maintenance, HMI monitoring and the interaction between FB environment/AME and configuration functions have to be supported by the communication services and protocols. In the IEC framework, the communication system is within the scope of IEC 61158 series and IEC 61784. The specification of the FBs and the FB environment is independent of the communication system. A mapping sublayer between the FB environment and communication system will adapt the application and the communication.

Parameters, blocks, objects, and functions in the FB application process and the system and network management application process (i.e. AME) are required to map efficiently to the underlying communications protocols and services. Communications services and protocols used with the FB application process and the system and network management application process are required to be specifically designed for use with distributed application processes, and are required to provide the services required by those applications.

Three different types of communication requirements are required to be considered:

1) Time-critical communications

Communications services and protocols used with the FB application process are required to support the unique requirements of time-critical communications, including

- i) deterministic transfers,
- ii) spatial consistency of transfers,
- iii) temporal consistency of transfers.

2) Demand communications

Communications services and protocols used with the FB application process are required to support the unique requirements of time-available communications, including segmented transfers of large data blocks. Communications services and protocols used with the FB application process are required to also support the unique requirements of “report by exception” communications.

3) Event communication

Communications services and protocols used with the FB application process are required to support the unique requirements for communicating events, and in particular be designed to

- i) minimize loading during quiescent periods,
- ii) prevent overloading during high-activity periods.

d) Resources

A resource is considered to be a logical subdivision within the software (and possibly hardware) structure of a device. Resources have independent control of their operation. The definition of a resource may be modified without affecting other resources within a device. A resource accepts and processes data and/or events from the process and/or communication interfaces and returns data and/or events to the process and/or communication interfaces, as specified by the applications utilizing the resource. An interoperable network view of applications is provided through device resources. Each resource specifies the network visible aspects of one or more local applications (or parts of distributed applications).

7.2 Overview of basic FB types

7.2.1 General

The basic FB types that may be accessed through their associated resources are listed below (see Figure 6). They are

- resource block;
- technology block;
- FB;
- view block;
- trend block and
- alert block.

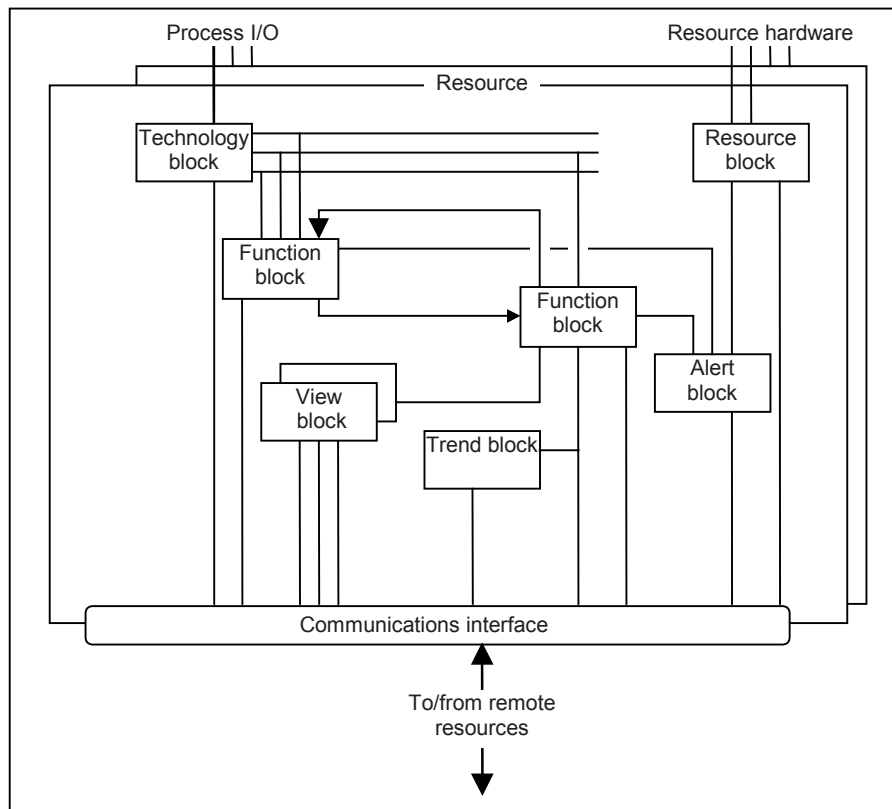


Figure 6 – Device structure for a FB application for the process industry

7.2.2 Resource block

The characteristics of the physical subcomponent associated with a resource may be described by a set of parameters which are included in a resource block. The resource block may also contain variables that are common to FBs and technology blocks for example set fail-safe. These variables are defined in the device block (see 7.4.4). The terms device block and resource block are used interchangeably.

7.2.3 Technology block

Technology blocks insulate FBs from the specifics of I/O devices, such as sensors, actuators, and switches. Technology blocks control access to I/O devices through a device independent interface defined for use by FBs. Technology blocks also perform functions, such as calibration and linearization, on I/O data to convert them to a device-independent representation. Their interface to FBs is defined as one or more implementation independent I/O channels.

7.2.4 FB (function block)

The FB is the primary means of defining monitoring and control in a FB application. FBs represent the basic automation functions performed by an application, which is as independent as possible of the specifics of I/O devices and the network. Each FB processes input variable and technology block input according to a specified algorithm and an internal set of contained variables. They produce output variable and output to technology blocks.

Based on the processing algorithm, a desired monitoring, calculation or control function may be provided. The results from FB execution can be reflected in parameters which are included in the function block for operation or diagnostic information. In addition, processing results may be reflected in the output to a technology block or to one or more output variables that may be linked to other FBs (see 7.3).

7.2.5 View block

The FB environment includes data structure definitions, called view blocks, to allow access to related block parameters as a group. View blocks facilitate fast operator display response when viewing FB data.

For each FB type, view blocks are defined for each of the following block parameter groupings:

- a) operations dynamic parameters;
- b) operations static parameters;
- c) all dynamic parameters;
- d) other static parameters.

7.2.6 Trend block

FBs are required to include data structure definitions, called trend blocks, to allow access to a sequential set of time-stamped samples of a single block parameter as a group. Trend blocks eliminate the communications and system processor overhead required for scanning parameters at a fast rate for trending.

Trend blocks definitions are required to include the definition of standard sample types and their associated sampling functions (for example, spot sample, integrated average, minimum, maximum, etc.).

7.2.7 Alert block

An alarm is the detection of a block when it leaves a particular state and when it returns to that state. The time at which the alarm state was detected is included as a time stamp in the alert message. Also, the priority is included to indicate whether this is an advisory or critical alarm.

The FB environment is required to include data structure definitions, called alert blocks, and associated resource and FB functions to allow the controlled transfer of alarm and event information within the system. Alert blocks predictably and efficiently route alarms and events to a selected destination (or destinations) within the system.

Alert block definitions include the definition of standard exchange protocols for initiating, sending, and confirming alert block reports, acknowledging alerts, interpreting standard and custom reason codes, and configuring alert functions such as alert key, alert priority, alert auto-acknowledge, etc.

Alerts are used by resource, technology and FBs to communicate notification messages when alarms are detected.

7.3 FB requirements

7.3.1 FB type specifications

The process FBs are required to include

- a) standardized data structures,
- b) standardized semantic meaning,
- c) common behaviours associated with standardized data,
- d) standardized definitions for basic input, output, and control functions.

7.3.2 FB normal and abnormal operation requirements

A FB performs normal and abnormal operation (see C.2.6). The normal operation of a FB is carried out under positive conditions of the process and automation devices during stable operation (operating point). The initialization, the warm or cold start as well as more advanced operations like going in a safe position are additional functions of the process control FBs. These operations are fixed parts of the FBs that are default linked to the associated events. These operations are known as abnormal operation or exception handling.

7.3.3 FB functional requirements

FB functional requirements valid for both discrete and continuous control are

- a) the instances name of FBs will be defined as the block tag. Tags are unambiguous within the scope of a system at one plant site. Tag names may have syntax rules that are not compatible with control language rules (for example, length, special characters);
- b) a status which is hard combined with the main process signal flow through the FB application, i.e. the FB inputs and outputs, not the contained parameter, are required to have status;
- c) FBs intended for use in cascade control structures are required to include back calculation parameters for inputs and outputs to carry defined standard cascade initialization, output limitation, and input or output value status information;
- d) FB functional specifications are required to include defined standard “bumpless” control-mode shedding and recovery actions on detection of bad input, output, or transfer values, or on detection of initialization requests or control limits from back calculation parameters;
- e) FB functional specifications are required to include defined standard fail-safe actions on detection of bad input, output, or transfer values, or a FB application process resource fail-safe command. When a block executes its fail-safe action, it is required to also send an alert message via an alert block. The fail-safe alert message is required to include a reason code that identifies what triggered the fail-safe action;
- f) FB functional specifications are required to include defined standard setpoint and output tracking behaviour. FB functional specifications are required to include defined standard bias and ratio set-point ramping behaviour;
- g) FBs are required to include parameters to simulate inputs and/or outputs and their status (“simulate parameters”) while viewing the actual inputs and/or outputs and their status, along with a parameter to enable and disable each simulation parameter. The FB specifications are required to include defined standard behaviour for the simulate parameters and for the simulate enable/disable parameters. The defined standard enable/disable behaviours are required to accommodate a mandatory simulation disable hardware jumper or switch;
- h) FBs are required to include parameters and standardized functional definitions for process measurement (PV) and deviation (DV) alarms;
- i) FBs are required to have mode, which controls the FB internal flow of information from inputs to outputs and the variations of the algorithms of a FB;
- j) FBs are required to include standardized functional definitions for initialization and restart for each block. Behaviour is required to be defined for the following operating circumstances:
 - 1) new device;
 - 2) cold restart (extended power failure);
 - 3) warm restart (short power failure);
 - 4) return from device fail-safe.

7.4 Initial sets of FBs derived from I&C

7.4.1 Minimum set of FBs derived from ABs (rich FBs)

The following examples give the minimum set of FBs for process automation (see Table 2) required to ensure integrity of these lower level controls and to ensure that a large section of the experience embodied in the last 10 years of fieldbus standardization effort is not wasted. Of course, the minimum set allows extensions.

Table 2 – Example of initial FB set

FB name	Description	Why included in the minimum list
Loop control (Proportional Integral Derivative – PID)	PID with configuration for proportional on error, proportional on measurement change, velocity and position outputs Interactive and non-interactive algorithms include simple set-point ramp option including ratio bias on the set-point input Include error squared and non-linear option	To ensure all PIDs operate in the same way given the same configuration and tuning parameters. This will help to ensure maximum loops are well tuned. Will promote understanding of need for different tuning sets. Will promote standard tuning tools and methods
Selector for control outputs	Multi-input high/low selector, with initialization and status passing	To allow common override and constraint control implementation with secure initialization and rest wind-up prevention
Selector for measurements	Highest, lowest, average and middle of three measurement selectors	For two out of three voting and extra security on unreliable measurements (for example, analytical)
Splitter	For split range controls	Support of back calculation for initialization and tracking for bump-less transfer, wind-up prevention
Analogue input	Scaling linearization quality checking	To ensure that scaling and conversion work is kept to a minimum and performed in the same way. To allow all manipulations in a control scheme to be done in floating-point engineering units in all implementations
Analogue output	With interlocks, check-back read-back linearization for non-linear valves	To ensure that common nomenclature, fail-safe operation, constraint controls and output checks are performed in the same standard way in all implementations
Output fan-out	Fan-out of controller outputs with auto manual switch	To ensure that common nomenclature, fail-safe operation, constraint controls and output checks are performed in the same standard way in all implementations
Discrete variable/pulse input	Single and multiple status inputs	To ensure common nomenclature, fail-safe operation, checks, counts are performed in the same way in all implementations
Discrete variable output	Single and multiple digital control outputs	To ensure common nomenclature, fail safe operation, checks, counts are performed in the same way in all implementations
Lead lag	First order lag and lead function	To allow common feed-forward and constraint control implementation with secure initialization
Dead time	Table driven dead time	To allow common predictor, feed-forward and constraint control implementation with secure initialization
Control/incremental summer	To sum outputs from incremental algorithms or to combine controller outputs	To allow common feed-forward and constraint control implementation with secure initialization
Control ratio	Ratio outputs of controller with initialization	To allow common feed-forward and constraint control implementation with secure initialization
Input switch	To switch inputs on event	To allow common predictor, feed-forward and constraint control implementation with secure initialization