

DIN EN 61804-4

ICS 35.060; 35.240.50

Einsprüche bis 2014-04-07

Entwurf

**Funktionsbausteine für die Prozessautomation und elektronische
Gerätebeschreibungssprache –
Teil 4: Interpretation von Gerätebeschreibungen
(IEC 65E/283/CD:2012), Text in Englisch**

Function blocks (FB) for process control and Electronic device description
language (EDDL) –
Part 4: EDD interpretation
(IEC 65E/283/CD:2012), Text in English

Anwendungswarnvermerk

Dieser Norm-Entwurf mit Erscheinungsdatum 2014-02-07 wird der Öffentlichkeit zur Prüfung und
Stellungnahme vorgelegt.

Weil die beabsichtigte Norm von der vorliegenden Fassung abweichen kann, ist die Anwendung dieses
Entwurfes besonders zu vereinbaren.

Stellungnahmen werden erbeten

- vorzugsweise online im Norm-Entwurfs-Portal des DIN unter www.entwuerfe.din.de bzw. für Norm-Entwürfe der DKE auch im Norm-Entwurfs-Portal der DKE unter www.entwuerfe.normenbibliothek.de, sofern dort wiedergegeben;
- oder als Datei per E-Mail an dke@din.de möglichst in Form einer Tabelle. Die Vorlage dieser Tabelle kann im Internet unter www.din.de/stellungnahme oder für Stellungnahmen zu Norm-Entwürfen der DKE unter www.dke.de/stellungnahme abgerufen werden;
- oder in Papierform an die DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik im DIN und VDE (Hausanschrift: Stresemannallee 15, 60596 Frankfurt am Main).

Die Empfänger dieses Norm-Entwurfs werden gebeten, mit ihren Kommentaren jegliche relevanten
Patentrechte, die sie kennen, mitzuteilen und unterstützende Dokumentationen zur Verfügung zu stellen.

Gesamtumfang 103 Seiten

DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik im DIN und VDE

Nationales Vorwort

Die englische Originalfassung des internationalen Dokuments IEC 65E/283/CD:2012 „Function blocks (FB) for process control and Electronic device description language (EDDL) – Part 4: EDD interpretation“ (CD, en: Committee Draft) ist unverändert in diesen Norm-Entwurf übernommen worden.

Das internationale Dokument wurde vom SC 65E „Devices and integration in enterprise systems“ der Internationalen Elektrotechnischen Kommission (IEC) erarbeitet und den nationalen Komitees zur Stellungnahme vorgelegt.

Die IEC und das Europäische Komitee für Elektrotechnische Normung (CENELEC) haben vereinbart, dass ein auf IEC-Ebene erarbeiteter Entwurf für eine Internationale Norm zeitgleich (parallel) bei IEC und CENELEC zur Umfrage (CDV-Stadium) und Abstimmung als FDIS (en: Final Draft International Standard) bzw. Schluss-Entwurf für eine Europäische Norm gestellt wird, um eine Beschleunigung und Straffung der Normungsarbeit zu erreichen. Dokumente, die bei CENELEC als Europäische Norm angenommen und ratifiziert werden, sind unverändert als Deutsche Normen zu übernehmen.

Da der Abstimmungszeitraum für einen FDIS bzw. Schluss-Entwurf prEN nur 2 Monate beträgt, und dann keine sachlichen Stellungnahmen mehr abgegeben werden können, sondern nur noch eine „JA/NEIN“-Entscheidung möglich ist, wobei eine „NEIN“-Entscheidung fundiert begründet werden muss, wird bereits der CD als DIN-Norm-Entwurf veröffentlicht, um die Stellungnahmen aus der Öffentlichkeit frühzeitig berücksichtigen zu können.

Für dieses Dokument ist das nationale Arbeitsgremium K 956 „Feldbus“ der DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik im DIN und VDE (www.dke.de) zuständig.

Für den Fall einer undatierten Verweisung im normativen Text (Verweisung auf eine Norm ohne Angabe des Ausgabedatums und ohne Hinweis auf eine Abschnittsnummer, eine Tabelle, ein Bild usw.) bezieht sich die Verweisung auf die jeweils neueste gültige Ausgabe der in Bezug genommenen Norm.

Für den Fall einer datierten Verweisung im normativen Text bezieht sich die Verweisung immer auf die in Bezug genommene Ausgabe der Norm.

Der Zusammenhang der zitierten Normen mit den entsprechenden Deutschen Normen ergibt sich, soweit ein Zusammenhang besteht, grundsätzlich über die Nummer der entsprechenden IEC-Publikation. Beispiel: IEC 60068 ist als EN 60068 als Europäische Norm durch CENELEC übernommen und als DIN EN 60068 ins Deutsche Normenwerk aufgenommen.

Das Präsidium des DIN hat mit Präsidialbeschluss 1/2004 festgelegt, dass DIN-Normen, deren Inhalt sich auf internationale Arbeitsergebnisse der Informationsverarbeitung gründet, unter bestimmten Bedingungen allein in englischer Sprache veröffentlicht werden dürfen. Diese Bedingungen sind für den vorliegenden Normentwurf erfüllt.

Da sich die Benutzer des vorliegenden Normentwurfs der englischen Sprache als Fachsprache bedienen, wird die Englische Fassung IEC 61804-4 veröffentlicht. Zu Abschnitt 3, der die Begriffe festlegt, wurde eine Übersetzung angefertigt und als informativer Nationaler Anhang NA des vorliegenden Normentwurfs hinzugefügt. Für viele der verwendeten Begriffe existieren keine gebräuchlichen deutschen Benennungen, da sich die deutschen Anwender in der Regel ebenfalls der englischen Benennungen bedienen. Dieser Normentwurf steht nicht in unmittelbarem Zusammenhang mit Rechtsvorschriften und ist nicht als Sicherheitsnorm anzusehen.

Es ist vorgesehen, auch bei der entsprechenden zukünftigen Deutschen Norm auf die deutsche Sprachfassung zu verzichten und diese in der englischsprachigen Fassung zu veröffentlichen.

Nationaler Anhang NA (informativ)

3.1 Begriffe

Für die Anwendung dieses Dokuments gelten die Begriffe nach IEC 61804-3 und die folgenden Begriffe.

3.1.1

EDD-Entwickler

(en: **EDD developer**)

Einzelperson oder Team, die/das eine EDD entwickelt

3.1.2

EDD-Anwendung

(en: **EDD application**)

Software oder Programme, welche die EDD zur Steuerung des Betriebs der Anwendung verwendet/verwenden

Anmerkung 1 zum Begriff: Zu diesen Anwendungen gehören Konfigurationstools, Kalibriergeräte, Gerätemanagementpakete und Gerätesimulatoren.

3.1.3

Endanwender

(en: **end user**)

Person, die das Feldgerät und/oder die EDDL-Anwendung benutzt

CONTENTS

FOREWORD	7
INTRODUCTION	2
1 Scope	3
2 References.....	3
3 Terms, definitions, abbreviated terms and acronyms.....	3
3.1 Terms and definitions	3
3.2 Abbreviated terms and acronyms	5
3.3 Conventions	5
4 User interface support	5
4.1 Overview	5
4.2 Menu conventions for mobile device applications	5
4.3 Menu conventions for PC-based applications	6
4.3.1 Overview	6
4.3.2 Diagnostic views	6
4.3.3 Process variable views	6
4.3.4 Device feature views for configuration	6
4.3.5 Maintenance feature views.....	7
4.3.6 Offline configuration.....	7
4.3.7 Void (Overview defined identifiers).....	7
4.3.8 Example of EDD menu structure	8
4.3.9 User interface	12
4.4 Menu conventions for all applications	16
4.4.1 Overview	16
4.4.2 Upload and download	16
4.5 User interface extensions	17
4.5.1 Overview	17
4.5.2 Context Sensitive User Interface	17
4.5.3 Note: Communication is influence by VALIDITY (see 10).Containers	18
4.5.4 Contained items.....	20
4.6 Layout rules	23
4.6.1 Overview	23
4.6.2 Layout rules for WIDTH and HEIGHT	23
4.6.3 Layout examples.....	26
4.7 Default menu styles	34
5 Additional user interface elements	35
5.1 Overview	35
5.2 Graph and chart	36
5.2.1 Common attributes.....	36
5.2.2 CHART	37
5.2.3 GRAPH.....	41
5.2.4 Axis	50
5.3 IMAGE	51
5.4 GRID.....	52
6 EDDL data description.....	53
6.1 Variables.....	53
6.1.1 Variable of type BIT_ENUMERATED	53

6.1.2	VARIABLE CLASSES	54
6.1.3	VARIABLE ACTIONS	54
6.2	EDDL application stored device data	55
6.2.1	Overview	55
6.2.2	File	55
6.2.3	List	57
6.2.4	Example of an EDD that inserts, replaces, or compares radar signals	61
7	EDDL builtin library	64
7.1	Builtin categories	64
7.1.1	Overview	64
7.1.2	Usage of builtins	64
7.1.3	User Interface Builtin	64
7.1.4	Communication Builtin	65
7.2	User interface built-ins	66
7.2.1	MenuDisplay	66
8	Modular Devices	69
8.1	Overview	69
8.2	Definitions	69
8.3	EDD Identification	69
8.4	Instance Object Model	70
8.5	Offline Configuration	70
8.6	Online Configuration	70
8.7	Simple modular device example	70
8.7.1	Separate EDD file example with direct EDD referencing	71
8.7.2	Separate EDD file example with classification EDD referencing and interfaces	73
8.7.3	One EDD file example	75
8.7.4	Combination of single and separate modular device example	77
8.8	COMPONENT_RELATION	77
8.8.1	NEXT_COMPONENT usage	77
8.8.2	REQUIRED_RANGES and ADDRESSING usage	77
8.9	Upload and Download for modular devices	77
8.10	Diagnostic	78
8.11	Reading Modular Device Topology	79
8.11.1	Scan	79
8.11.2	Detect Module Type	80
8.12	Configuration Check	80
9	Offline Configuration	81
9.1	Objective	81
9.2	Overview	81
9.2.1	Templates	82
9.2.2	Configuration Database	82
9.2.3	Upload Data Sets	82
9.2.4	Online Parameterization Process	82
9.2.5	Offline Parameterization Process	83
9.2.6	Download Process	83
9.2.7	Upload Process	83
9.2.8	Equalization Process	83
9.3	Online Parameterization	83

9.4	Offline Parameterization	83
9.5	Configuration Database	84
9.6	Templates	84
9.7	Download, Upload, and Equalization	85
9.7.1	Download Process	85
9.7.2	Upload Process	86
9.7.3	Equalization Process	87
9.7.4	Download / Upload Menus	87
9.7.5	Download / Upload Action Methods	88
9.7.6	Error Recovery	89
10	EDDL communication description	89
10.1	COMMAND	89
10.1.1	OPERATION	89
10.1.2	TRANSACTION	90
10.1.3	Command Addressing	94
Figure 1	– EDD Example of root menus	12
Figure 2	– Example of an EDD application for diagnostics	13
Figure 3	– Example of an EDD application for process variables	14
Figure 4	– Example of an EDD application for primary variables	14
Figure 5	– Example of an EDD application for process-related device features	15
Figure 6	– Example of an EDD application for device features	15
Figure 7	– Example of an EDD application for device features	16
Figure 8	– Usage of COLLECTION MEMBERS in MENUs of STYLE GROUP	20
Figure 9	– Displaying single bits of BIT_ENUMERATED	21
Figure 10	– Displaying multiple bits of BIT_ENUMERATED	21
Figure 11	– Example of an EDD application for a variable of type BIT_ENUMERATED	22
Figure 12	– Layout for protruding elements	24
Figure 13	– Layout for partially filled rows	25
Figure 14	– Layout for oversized elements	25
Figure 15	– Layout for columns in stacked group	26
Figure 16	– Layout for columns with GRAPHS in stacked group	26
Figure 17	– Example of an EDD for an overview menu	27
Figure 18	– Example of an EDD application for an overview window	27
Figure 19	– Example of an EDD using COLUMNBREAK	27
Figure 20	– Example of an EDD application for an overview window	28
Figure 21	– EDD Example for an overview window	28
Figure 22	– Example of an EDD application for an overview window	29
Figure 23	– Example of an EDD for in-line graphs and charts	29
Figure 24	– Example of an EDD application for an in-line graph	30
Figure 25	– Example of an EDD for full-width graphs and charts	30
Figure 26	– Example of an EDD application for a full-width graph	31
Figure 27	– Example of an EDD for nested containers	32
Figure 28	– Example of an EDD application for nested containers	32
Figure 29	– Example of an EDD for EDIT_DISPLAYS	33

Figure 30 – Example of an EDD application for EDIT_DISPLAYS	33
Figure 31 – Example of an EDD for images	34
Figure 32 – Example of an EDD application for images	34
Figure 33 – HEIGHT and WIDTH attributes for CHART and GRAPH	36
Figure 34 – EMPHASIS attribute to differentiate one or more SOURCES or WAVEFORMS	37
Figure 35 – Example of a chart with one curve in a dialog	39
Figure 36 – Graph and the visual elements	42
Figure 37 – Multiple used axes	46
Figure 38 – EDD with device supported zooming and scrolling	50
Figure 39 – EDD example of an IMAGE	51
Figure 40 – EDD example of an IMAGE with LINK attribute	51
Figure 41 – EDD example of a GRID	53
Figure 42 – Result of the EDD example	53
Figure 43 – Wrong usage of a BIT_ENUMERATED	54
Figure 44 – Usage of ENUMERATED instead of BIT_ENUMERATED	54
Figure 45 – Example of a file declaration	56
Figure 46 – Example of comparing valve signatures	57
Figure 47 – Example of more complex file declaration	58
Figure 48 – Example of reviewing the stored radar signals	59
Figure 49 – Example of a wizard	68
Figure 50 – The different relations of module	70
Figure 51 – Components and possible configuration of the modular devices	71
Figure 52 – Separate EDD file example with direct EDD referencing	72
Figure 53 – EDD example for module1	72
Figure 54 – EDD example for module2	73
Figure 55 – EDD example for modular device	74
Figure 56 – EDD example for module1	74
Figure 57 – EDD example for module2	75
Figure 58 – EDD example for module2	76
Figure 59 – NEXT_COMPONENT usage	77
Figure 60 – REQUIRED_RANGES usage	77
Figure 61 – Upload/Download order of a modular device. Arrows shows the hierarchy, numbers shows ordering	78
Figure 65 – offline configuration process	82
Figure 66 – Example of TEMPLATE usage	85
Figure 67 – download process	86
Figure 68 – upload process	87
Figure 69 – single item mask	91
Figure 70 – mapping example with a single item mask	91
Figure 71 – multiple item mask	91
Figure 72 – mapping example with a multiple item mask	92
Figure 73 – INFO qualifier	92
Figure 74 – INDEX qualifier	93

Figure 75 – INFO and INDEX qualifier.....93

Table 1 – List of defined menu identifiers for mobile devices 5

Table 2 – List of defined menu identifiers for PC-based devices 6

Table 3 – List of defined menu identifiers 35

Table 4 – Image formats 52

Table 5 – Usage of user interface and communication builtins in EDDs 64

Table 6 – User Interface Builtins 64

Table 7 – Communication and Communication Related Builtins 65

INTERNATIONAL ELECTROTECHNICAL COMMISSION

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

Part 4: EDD interpretation

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.

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.

IEC takes no position concerning the evidence, validity and scope of this patent right.

The main task of IEC technical committees is to prepare International Standards. However, a technical committee may propose the publication of a technical report when it has collected data of a different kind from that which is normally published as an International Standard, for example, “state of the art”.

IEC 61804-4, which is an International Standard, has been prepared by subcommittee 65E: Devices and integration in enterprise systems.

The text of this International Standard is based on the following documents:

Enquiry draft	Report on voting
65C/xxx/CDV	65C/xxx/RVC

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

This IS was developed using material from FDI Cooperation LLC (Fieldbus Foundation^{TM1}, HART^{®1} Communication Foundation (HCF), PROFIBUS^{TM1} Nutzerorganisation e.V. (PNO)), OPC Foundation (OPCF) and FDT Group.

The committee has decided that the contents of this publication will remain unchanged until the maintenance result date² 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.

The National Committees are requested to note that for this publication the stability date is 2017

THIS TEXT IS INCLUDED FOR THE INFORMATION OF THE NATIONAL COMMITTEES AND WILL BE DELETED AT THE PUBLICATION STAGE.

A bilingual version of this publication may be issued at a later date.

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

¹ HART is the registered trademark of the HART Communication Foundation. Fieldbus Foundation is the trademark of the Fieldbus Foundation. PROFIBUS is the trademark of the PROFIBUS Nutzerorganisation e.V. This information is given for the convenience of users of this document and does not constitute an endorsement by IEC of the product named. Equivalent products may be used if they can be shown to lead to the same results.

² The National Committees are requested to note that for this publication the maintenance result date is 2047.

1 INTRODUCTION

2 This document

- 3 • contains an overview of the use of EDDL;
- 4 • provides examples demonstrating the use of the EDDL constructs;
- 5 • shows how the use cases are fulfilled; and
- 6 • shows the proper EDD application interpretation for each example.

7 This document is not an EDDL tutorial and is not intended to replace the EDDL specification.

8 Instructions are provided for the EDD application, which describe what is to be performed

9 without prescribing the technology used in the host implementation. For example, the FILE

10 construct describes data that is to be stored by the EDD application on behalf of the EDD.

11 The FILE construct does not specify how the data is to be stored. The EDD application can

12 use a database, a flat file, or any other implementation it chooses.

13 Headings named 'Void' are used to retain the numbering of previous IEC/TR 61804-4:2006.

14

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

Part 4: EDD interpretation

1 Scope

This International Standard (IS) specifies EDD interpretation for EDD applications and EDDs to support EDD interoperability. This document is intended to ensure that field device developers use the EDDL constructs consistently and that the EDD applications have the same interpretations of the EDD. It supplements the EDDL specification to promote EDDL application interoperability and improve EDD portability between EDDL applications.

2 References

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies. IEC 60050-351, *International Electrotechnical Vocabulary (IEV) – Part 351: Automatic control*

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

IEC 61784-2, *Digital data communications for measurement and control Profile – Part 2: Additional Fieldbus profiles for real-time networks based on ISO/IEC 8802-3*

IEC 61804-2:2004, *Function blocks (FB) for process control – Part 2: Specification of FB concept and Electronic Device Description Language (EDDL)*

IEC 61804-2, *Function blocks (FB) for process control and Electronic device description language (EDDL) – Part 2: Specification of FB concept*

IEC 61804-3 Ed 3.0, *Function blocks (FB) for process control and Electronic device description language (EDDL) – Part 3: EDDL syntax and semantics*

ISO/IEC 10918 series, *Information technology - Digital compression and coding of continuous-tone still images*

3 Terms, definitions, abbreviated terms and acronyms

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in IEC 61804-3 and the following apply.

3.1.1

EDD developer

individual or team that develops an EDD

3.1.2

EDD application

software or programmes utilizing the EDD to guide the operation of the application

57 Note 1 to entry: These applications include configuration tools, calibrators, instrument management packages,
58 and instrument simulators

59 **3.1.3**

60 **end user**

61 individual using the field device and/or the EDDL application

3.2 Abbreviated terms and acronyms

CP	Communication Profile
CPF	Communication Profile Family
EDD	Electronic Device Description
EDDL	Electronic Device Description Language
FF	Foundation Fieldbus
PC	Personal computer

3.3 Conventions

There exists some differences using and interpreting EDDL based on the used communication network and device model. The different communication networks used within this document are:

- HART (according to IEC 61784-1 CPF9)
- FF (according to IEC 61784-1 CPF1)
- PROFIBUS (according to IEC 61784-1 CPF3)
- PROFINET (according to IEC 61784-2 CPF3)

4 User interface support

4.1 Overview

Most EDD applications can be characterized as either a PC application or a hand-held application. Due to the relatively small screen of a hand-held device, hand-held applications can only display a small amount of information at any given time. On the other hand, PC applications can provide a much more beneficial user interface, largely due to their larger screen size.

To support the capabilities of PC applications, the MENU construct has been extended in IEC 61804-3 compared to IEC 61804-2:2004. Due to the differences in the user interfaces of PC applications and hand-held applications, it is expected that many devices will define two MENU hierarchies – one for hand-held applications and the other for PC applications. Some MENUs may be used in both hierarchies. Therefore, the entire hierarchy need not be specified twice.

Different menu structures for different classes of applications are possible. This specification may be used to create menu structures in an EDD that are interpreted by applications in an unambiguous way. To promote interoperability across applications, it is highly recommended that all EDD applications follow this specification.

4.2 Menu conventions for mobile device applications

EDD applications use specific menus in the EDDs to show the user interface of the device. In addition user interface items can be described for mobile devices and PCs at once. For mobile devices strings can have beside language code a specific country code zz to specify shorter strings or lower resolution images (see Table 1).

Table 1 – List of defined menu identifiers for mobile devices

Menu Identifier	Default STYLE	Short description
root_menu	TABLE	Handheld root menu for HART devices. This is mandatory for all HART EDDs.

Menu_Top*	MENU	Handheld root menu for FF devices e.g. Menu_Top_TB
-----------	------	--

4.3 Menu conventions for PC-based applications

4.3.1 Overview

EDD applications use special menus in the EDDs to show the user interface of the device. Such menus are defined for diagnostic, process variables, device features, and offline configuration. Table 2 defines identifiers for the different root menus. Other submenus can be underneath these root menus.

Table 2 – List of defined menu identifiers for PC-based devices

Menu Identifier	Default STYLE	Usage	Short description
device_root_menu	MENU	online	Device feature views for set up
diagnostic_root_menu	MENU	online	Diagnostic views
maintenance_root_menu	MENU	online	Maintenance feature views
offline_root_menu	TABLE	offline	Offline configuration
process_variables_root_menu	MENU	online	Process variable views

If none of those MENUs exists for HART EDDs (e.g. for legacy EDDs) the EDD application shall use root_menu. HART EDD application shall display offline_root_menu as STYLE WINDOW.

If none of those MENUs exists for FF EDDs (e.g. for legacy EDDs) the EDD application shall generate menus structure from the BLOCK_A PARAMETERS attribute.

If none of those MENUs exists for PROFIBUS and PROFINET EDDs (e.g. for legacy EDDs) the EDD application shall use Table_Main_Maintenance or Table_Main_Specialist as entry point for the offline configuration and is represented as STYLE TABLE and Menu_Main_Maintenance or Menu_Main_Specialist as entry point for the online configuration and is represented as STYLE MENU.

The Usage column in Table 2 defines how the EDD application has online or offline access to the device parameter (see Clause 9).

PROFIBUS and PROFINET allows in submenus to define different parameter access by defining ACCESS ONLINE or ACCESS OFFLINE.

4.3.2 Diagnostic views

The diagnostic_root_menu includes views that show the device state, detailed diagnostic information and may include graphical views that show, for example, a valve signature.

4.3.3 Process variable views

The process_variable_root_menu includes views that show process measurements and set points with their quality and important information for process operators, for example, ranges.

4.3.4 Device feature views for configuration

The device_root_menu includes features for device. These features can be split into process-related and device-specific features. This structure is not required if the number of features is too small for splitting. Submenus that represent any structuring are allowed on the device

128 feature menu. In case of such submenus, the menus that are underneath can be split into
129 process-related and device-specific features.

130 **4.3.5 Maintenance feature views**

131 The maintenance_root_menu includes features for maintaining the device while the runtime
132 phase.

133 **4.3.6 Offline configuration**

134 This menu hierarchy includes parameters, variables, and methods for offline configuration. It
135 contains, in particular, all of the application-specific parameters of the device and may also
136 contain important read-only and writeable variables. For more information about application-
137 specific parameters, see 4.4. The menu can have offline methods, for example, configuration
138 assistants.

139 **4.3.7 Void (Overview defined identifiers)**

140

4.3.8 Example of EDD menu structure

EDD example in Figure 1 includes additional menus that can be added to an EDD for PC applications. These menus are additional to the existing menus for the applications. This example is specific for a device with an interface according to IEC 61784-1 CPF9 (HART). Examples for devices with an interface according IEC 61784-1 CPF1 (Foundation Fieldbus) and IEC 61784-1 CP 3/1 or CP 3/2 (PROFIBUS) would be very similar.

```

MENU diagnostic_root_menu
{
    LABEL "Diagnostics";
    STYLE MENU;
    ITEMS
    {
        status_window,          /* menu: style=window */
        self_test               /* method */
    }
}

MENU status_window
{
    LABEL "Status";
    STYLE WINDOW;
    ITEMS
    {
        standard_diagnostics_page, /* menu: style=page */
        devspec_diagnostics_page  /* menu: style=page */
    }
}

MENU standard_diagnostics_page
{
    LABEL "Standard";
    STYLE PAGE;
    ITEMS
    {
        device_status           /* variable */
    }
}

MENU devspec_diagnostics_page
{
    LABEL "Device Specific";
    STYLE PAGE;
    ITEMS
    {
        xmtr_specific_status_1, /* variable */
        xmtr_specific_status_2 /* variable */
    }
}

METHOD self_test
{
    LABEL "Self Test";
    DEFINITION
    {
        /* elided */
    }
}

MENU maintenance_root_menu
{
    LABEL "Maintenance";
    STYLE MENU;
    ITEMS
    {
        device_mode_dialog, /* menu: style=dialog */
        teach_in            /* method */
    }
}

```

```

209 MENU device_mode_dialog
210 {
211     LABEL "Device Mode";
212     STYLE DIALOG;
213     ITEMS
214     {
215         mode_page                      /* menu: style=page */
216     }
217 }
218
219 MENU mode_page
220 {
221     LABEL "Process Variables";
222     STYLE PAGE;
223     ITEMS
224     {
225         transducer_group,              /* menu: style=group */
226         function_group                 /* menu: style=group */
227     }
228 }
229
230 MENU transducer_group
231 {
232     LABEL "Transducer";
233     STYLE GROUP;
234     ITEMS
235     {
236         trans_target_mode,             /* variable */
237         trans_actual_mode              /* variable */
238     }
239 }
240
241 MENU function_group
242 {
243     LABEL "Function";
244     STYLE GROUP;
245     ITEMS
246     {
247         func_target_mode,              /* variable */
248         func_actual_mode               /* variable */
249     }
250 }
251
252 METHOD teach_in
253 {
254     LABEL "Teach-in";
255     DEFINITION
256     {
257         /* elided */
258     }
259 }
260
261 MENU process_variables_root_menu
262 {
263     LABEL "Process Variables";
264     STYLE MENU;
265     ITEMS
266     {
267         overview_window,               /* menu: style=window */
268         primary_vars_window            /* menu: style=window */
269     }
270 }
271
272 MENU overview_window
273 {
274     LABEL "Overview";
275     STYLE WINDOW;
276     ITEMS
277     {
278         process_vars_page              /* menu: style=page */
279     }
280 }

```

```
281 MENU process_vars_page
282 {
283     LABEL "Process Variables";
284     STYLE PAGE;
285     ITEMS
286     {
287         pressure_group,          /* menu: style=group */
288         temperature_group        /* menu: style=group */
289     }
290 }
291
292
293 MENU pressure_group
294 {
295     LABEL "Pressure";
296     STYLE GROUP;
297     ITEMS
298     {
299         pv_digital_value,        /* variable */
300         pv_upper_range_value,    /* variable */
301         pv_lower_range_value     /* variable */
302     }
303 }
304
305 MENU temperature_group
306 {
307     LABEL "Temperature";
308     STYLE GROUP;
309     ITEMS
310     {
311         sv_digital_value,        /* variable */
312         sv_upper_range_value,    /* variable */
313         sv_lower_range_value     /* variable */
314     }
315 }
316
317 MENU primary_vars_window
318 {
319     LABEL "Primary Variables";
320     STYLE WINDOW;
321     ITEMS
322     {
323         pressure_chart_page,     /* menu: style=page */
324         temperature_chart_page   /* menu: style=page */
325     }
326 }
327
328 MENU pressure_chart_page
329 {
330     LABEL "Pressure";
331     STYLE PAGE;
332     ITEMS
333     {
334         pressure_chart           /* chart */
335     }
336 }
337
338 CHART pressure_chart
339 {
340     /* elided */
341 }
342
343 MENU temperature_chart_page
344 {
345     LABEL "Temperature";
346     STYLE PAGE;
347     ITEMS
348     {
349         temperature_chart        /* chart */
350     }
351 }
352
353 CHART temperature_chart
354 {
355     /* elided */
356 }
```

```

357 MENU device_root_menu
358 {
359     LABEL "Device";
360     STYLE MENU;
361     ITEMS
362     {
363         process_related_window,          /* menu: style=window */
364         device_specific_window,          /* menu: style=window */
365         master_reset                      /* method */
366     }
367 }
368
369 MENU process_related_window
370 {
371     LABEL "Process Related";
372     STYLE WINDOW;
373     ITEMS
374     {
375         identification_page,             /* menu: style=page */
376         output_info_page                 /* menu: style=page */
377     }
378 }
379
380 MENU identification_page
381 {
382     LABEL "Identification";
383     STYLE PAGE;
384     ITEMS
385     {
386         tag,                             /* variable */
387         manufacturer,                    /* variable */
388         device_type,                     /* variable */
389         device_revision,                 /* variable */
390         descriptor,                      /* variable */
391         message                          /* variable */
392     }
393 }
394
395 MENU output_info_page
396 {
397     LABEL "Output Information";
398     STYLE PAGE;
399     ITEMS
400     {
401         range_values_group,              /* menu: style=group */
402         sensor_limits_group              /* menu: style=group */
403     }
404 }
405
406 MENU range_values_group
407 {
408     LABEL "Range Values";
409     STYLE GROUP;
410     ITEMS
411     {
412         pv_units,                        /* variable */
413         pv_urv,                          /* variable */
414         pv_lrv                           /* variable */
415     }
416 }
417
418 MENU sensor_limits_group
419 {
420     LABEL "Sensor Limits";
421     STYLE GROUP;
422     ITEMS
423     {
424         sensor_units,                    /* variable */
425         upper_sensor_limit,               /* variable */
426         lower_sensor_limit                /* variable */
427     }
428 }
429
430

```

```
431 MENU device_specific_window
432 {
433     LABEL "Device Specific";
434     STYLE WINDOW;
435     ITEMS
436     {
437         identification_page,      /* menu: style=page */
438         calibration_page          /* menu: style=page */
439     }
440 }
441
442 MENU calibration_page
443 {
444     LABEL "Calibration";
445     STYLE PAGE;
446     ITEMS
447     {
448         sensor_limits_group,      /* menu: style=group */
449         sensor_trim_group         /* menu: style=group */
450     }
451 }
452
453 MENU sensor_trim_group
454 {
455     LABEL "Sensor Trim";
456     STYLE GROUP;
457     ITEMS
458     {
459         upper_sensor_trim_point,  /* variable */
460         lower_sensor_trim_point,  /* variable */
461         sensor_trim               /* method */
462     }
463 }
464
465 METHOD master_reset
466 {
467     LABEL "Master Reset";
468     DEFINITION
469     {
470         /* elided */
471     }
472 }
473 }
```

Figure 1 – EDD Example of root menus

4.3.9 User interface

4.3.9.1 Diagnostics

Figure 2 shows an example of an EDD application for diagnostics.

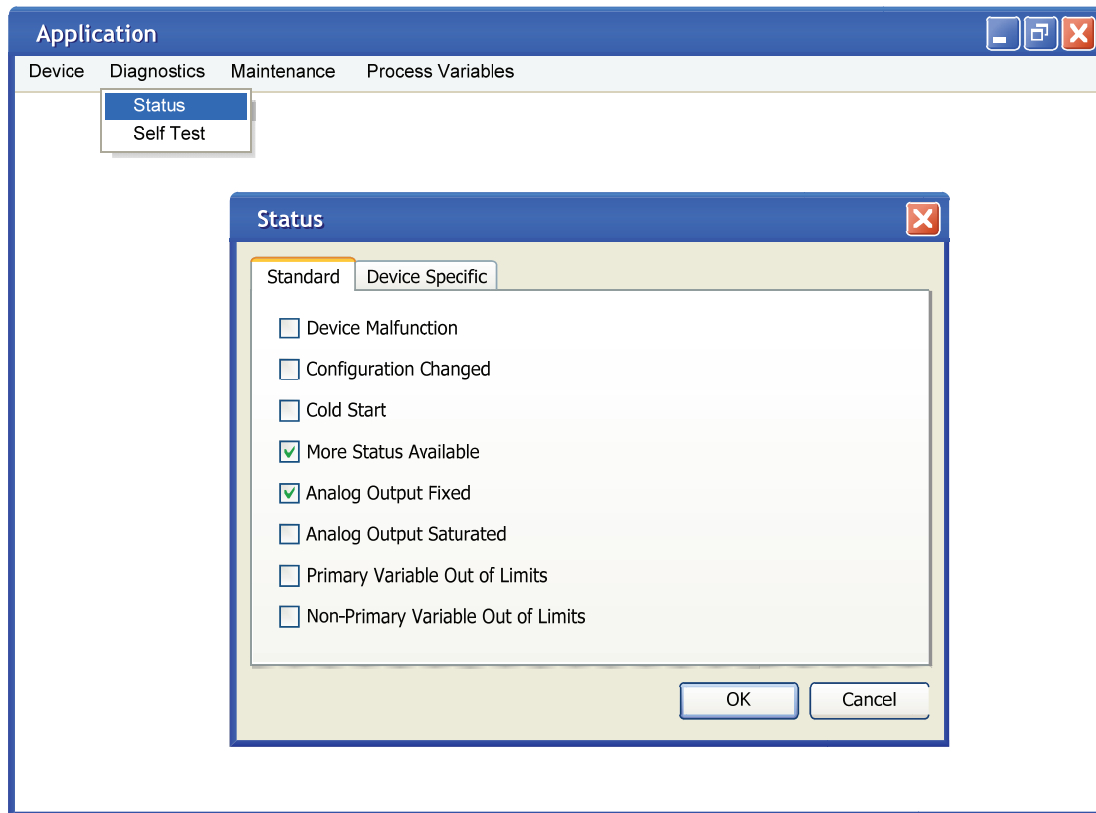


Figure 2 – Example of an EDD application for diagnostics

4.3.9.2 Process variables

Figure 3 and Figure 4 show examples of EDD applications for process variables.

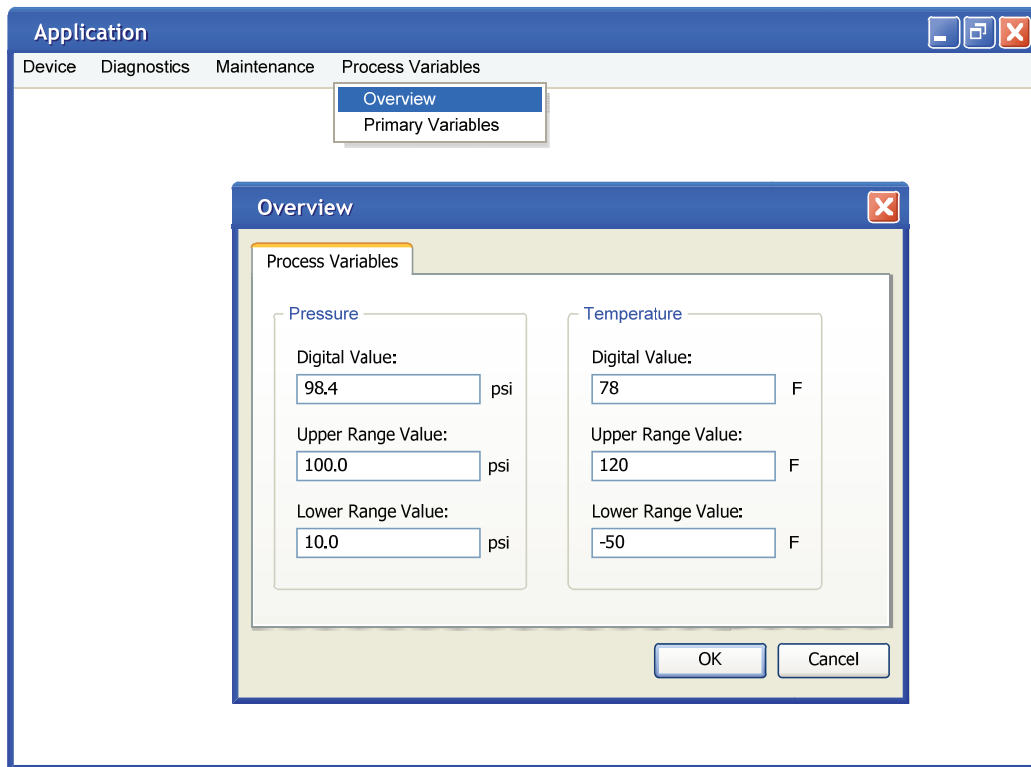


Figure 3 – Example of an EDD application for process variables

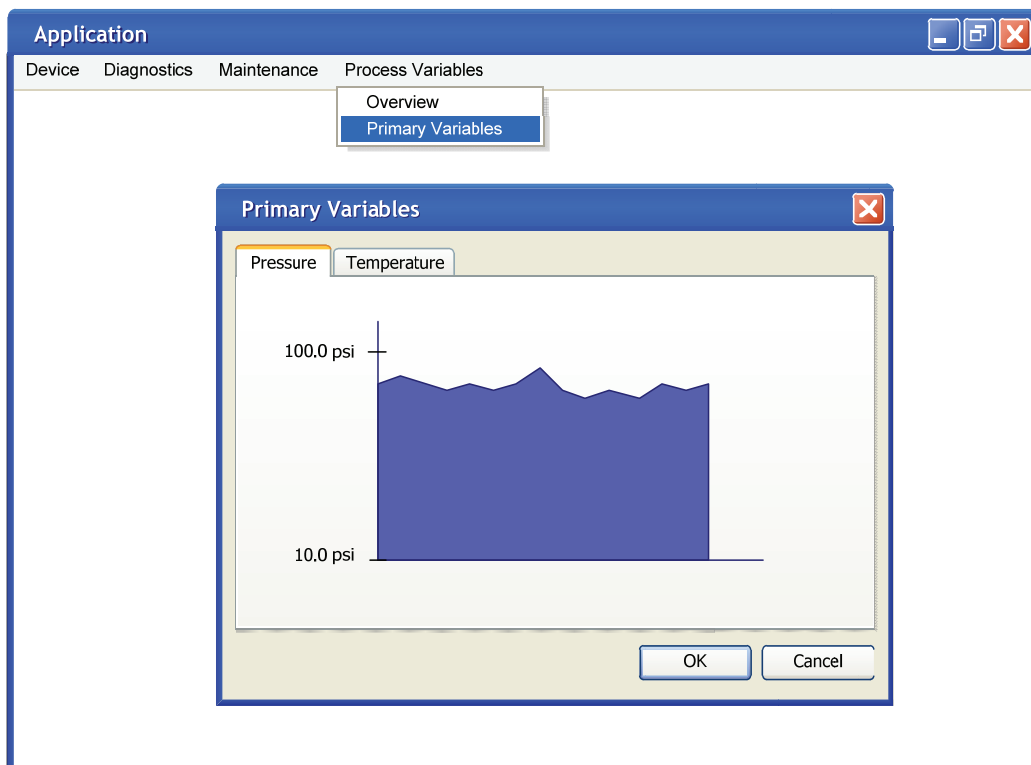


Figure 4 – Example of an EDD application for primary variables

489 4.3.9.3 Device features

490 Figure 5, Figure 6 and Figure 7 show examples of EDD applications for device features.

The screenshot shows a software interface with a main window titled 'Application'. It has a menu bar with 'Device', 'Diagnostics', 'Maintenance', and 'Process Variables'. A dropdown menu is open under 'Diagnostics', showing 'Process Related', 'Device Specific', and 'Master Reset'. The 'Process Related' dialog box is open, showing two tabs: 'Identification' and 'Output Information'. The 'Output Information' tab is active, displaying 'Range Values' and 'Sensor Limits' sections. Both sections have a 'Units' dropdown set to 'psi'. The 'Range Values' section has 'Upper Range Value' set to 100.0 and 'Lower Range Value' set to 10.0. The 'Sensor Limits' section has 'Upper Sensor Limit' set to 250 and 'Lower Sensor Limit' set to 0. 'OK' and 'Cancel' buttons are at the bottom.

491
492 Figure 5 – Example of an EDD application for process-related device features

The screenshot shows the same 'Application' window, but the 'Device Specific' dialog box is open. The 'Diagnostics' menu is still open. The 'Device Specific' dialog box has two tabs: 'Identification' and 'Calibration'. The 'Calibration' tab is active, displaying 'Sensor Limits' and 'Sensor Trim' sections. Both sections have a 'Units' dropdown set to 'psi'. The 'Sensor Limits' section has 'Upper Sensor Limit' set to 250 and 'Lower Sensor Limit' set to 0. The 'Sensor Trim' section has 'Upper Trim Point' set to 250 and 'Lower Trim Point' set to 0, with a 'Trim Sensor' button below. 'OK' and 'Cancel' buttons are at the bottom.

494
495 Figure 6 – Example of an EDD application for device features

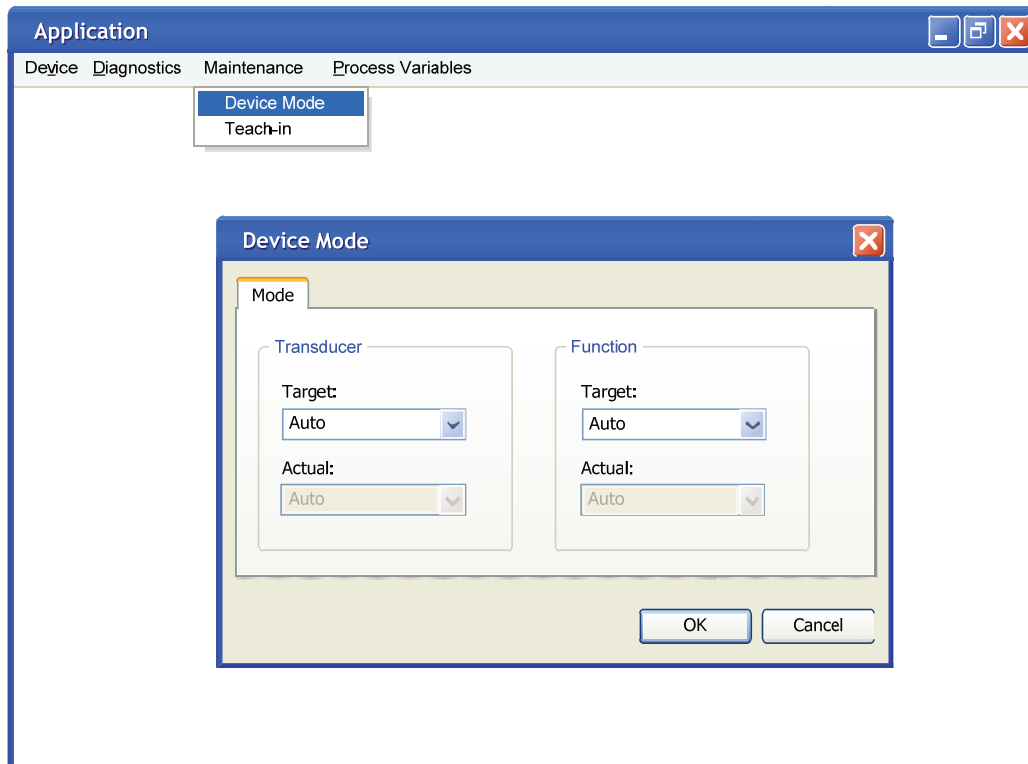


Figure 7 – Example of an EDD application for device features

4.4 Menu conventions for all applications

4.4.1 Overview

This menu is used by hand-held and PC applications.

4.4.2 Upload and download

There are two kinds of parameters in a device:

- application-specific parameters; and
- device-specific parameters.

When a device is replaced at a particular plant location, the application-specific parameters from the old device should be downloaded into the new device. However, the device-specific parameters are never downloaded because they are specific to the physical device.

For loading the application-specific parameter two menus should be used.

- The “download_variables” menu specifies the order and the total set of the application-specific parameters that should be read from the device.
- The “upload_variables” menu specifies the order and the total set of the application-specific parameters that should be loaded to the device.

Both the upload and download menus should contain all of the application-specific parameters for the device. If the device has special needs in terms of handling the data, errors, timing of the commands, and so on, the upload and download menus may contain methods in addition to, or instead of, variables.

Hart and Profibus supports the menu called upload_variables and download_variables. The definition of these menus for both Hart and Profibus is consistent with the definition of the

upload and download menus described above. Therefore, these well-known names for the upload and download menus are used. If upload and download menus are not included in the EDD, then this capability, if supported by the device, is EDD application specific.

The ordering in the off-line configuration menu (`offline_root_menu`) is user-aspect-orientated in contrast to the upload/download menus that are ordered from a communications point of view.

4.5 User interface extensions

4.5.1 Overview

The user interface extensions are based on a simple user interface model. The model consists of two concepts:

- containers; and
- contained items.

Containers are so named because they contain other user interface elements. Containers may include: menus, windows, dialog boxes, tables, pages, groups, and other containers. Containers correspond to a MENU. They are distinguished from one another via the STYLE attribute of the MENU. This STYLE attribute indicates how the MENU will be displayed. EDD constructs for these user interface components are the same as a MENU except for their display.

Contained items are variables, methods, edit displays, graphs, charts, images, static text, static string, COLUMNBREAK, ROWBREAK, etc.

4.5.2 Context Sensitive User Interface

4.5.2.1 Overview

To control appearance of user interface elements EDDL has different possibilities:

- attribute VISIBILITY
- conditional expressions on MENU ITEM
- attribute VALIDITY

4.5.2.2 VISIBILITY

The attribute VISIBILITY of the basic constructs (e.g. MENU, VARIABLE, IMAGE, GRAPH, CHART, GRID) allows to control the appearance of user interface elements. The values TRUE or FALSE of VISIBILITY are meant to be evaluated dynamically depending on VARIABLE values.

VISIBILITY does not affect the screen layout. In the screen layout, the same space is reserved as if the element is shown.

4.5.2.3 Conditional Expressions on MENU ITEM

Conditional expressions in the MENU ITEM list allow to control the screen layout. In the screen layout, no space is reserved if an element is not shown.

4.5.2.4 VALIDITY

The attribute VALIDITY of the basic constructs e.g. MENU, VARIABLE, IMAGE, GRAPH, CHART, GRID allows to control the appearance of user interface elements. The values TRUE

559 or FALSE of VALIDITY are meant to be evaluated dynamically dependent of VARIABLE
560 values.

561 VALIDITY does affect the screen layout. In the screen layout, no space is reserved if the
562 element is not valid.

563 NOTE Communication is influence by VALIDITY (see Clause 10).

564 **4.5.3 Containers**

565 **4.5.3.1 Overview**

566 Subclause 0 describes the general layout of containers.

567 **4.5.3.2 Menu**

568 A menu takes the form of a drop-down menu, a pop-up menu or a navigation bar. A menu
569 may contain windows, dialog boxes, tables, methods, and separators.

570 **4.5.3.3 Window**

571 A menu of STYLE WINDOW takes the form of a modeless window. It may contain dialog
572 boxes, tables, pages, groups, methods, variables, edit displays, graphs, charts, images,
573 static text, column/row breaks, other windows, etc. Windows may contain menus of STYLE
574 MENU.

575 When a container contains a menu STYLE WINDOW, the container should render this
576 window as either a button or a hyperlink. The LABEL of the corresponding MENU should
577 appear on the button or as the hyperlink text. When the button or hyperlink is pressed, the
578 corresponding window should be opened on top of the calling dialog or window.

579 **4.5.3.4 Dialog box**

580 A menu STYLE DIALOG box takes the form of a modal window. It may contain windows,
581 tables, pages, groups, methods, variables, edit displays, graphs, charts, images, static text,
582 column/row breaks, other dialog, etc. Dialog may contain menus STYLE MENU. If a dialog
583 contains a subordinated window, the EDD application shall manage the window as a modal
584 subdialog.

585 When a container contains a menu STYLE DIALOG, the dialog should be rendered by the
586 container as either a button or a hyperlink. The LABEL of the corresponding MENU should
587 appear on the button or as the hyperlink text. When the button or hyperlink is pressed, the
588 corresponding dialog should be opened on top of the calling dialog or window. If a dialog is
589 called from a window, no interactions are possible unless the dialog is closed.

590 **4.5.3.5 Table**

591 A menu of STYLE TABLE may contain variables, methods and other menus. The submenus
592 build a hierarchy, which may show the device to the user in a compact way, for example, a
593 table in which each row is a variable, method or menu.

594 The STYLE attribute (STYLE = TABLE) should be defined only on the root menu of a
595 hierarchy. The EDD application also uses the style of the root for its items unless another
596 style is defined.

597 **4.5.3.6 Page**

598 The menu of STYLE PAGE within a container form a tabbed dialog. It may contain windows,
599 dialog boxes, tables, groups, methods, variables, edit displays, graphs, charts, images, static

600 text, COLUMNBREAKs and ROWBREAKs separators.. Pages may not contain menus of
601 STYLE MENU or other pages.

602 The EDD application shall ignore ROWBREAK, COLUMNBREAK, and SEPARATOR between
603 pages.

604 4.5.3.7 Group

605 A menu of STYLE GROUP takes the form of a group box. It may contain windows, dialog
606 boxes, methods, variables, edit displays, graphs, charts, images, static text, column/row
607 breaks, and other groups. In order to maintain a clear arrangement on the user interface, the
608 following restricting rules apply.

609 • The nesting level of GROUP declarations is restricted to two. In other words, a GROUP
610 may contain further GROUPs but these may not contain further GROUPs.
611 When a second layer groupbox contains a group box, this group box shall be rendered as
612 either a button or a hyperlink. The LABEL shall appear on the button or as the hyperlink
613 text. When the button or hyperlink is pressed, the corresponding dialog box shall be
614 opened on top of the calling dialog or window.

615 • Groups shall not contain menus of STYLE PAGE and menu of STYLE TABLE.

616 Within a GROUP, usually logically related parameters and methods are grouped. The title of
617 the GROUP indicates this relation, for example, "AnalogInput 1". Often the same string is
618 also used in the labels of the GROUP items, for example, "AnalogInput1_StaticRevision" or
619 "Analog Input1_Blockmode". In order to avoid this duplication of information, the EDD-editor
620 may assemble the items of a GROUP in a COLLECTION in order to override the original
621 labels. If the parameters are to be referred to in a GROUP the parameters can be referenced
622 using the COLLECTION; in other cases, especially without additional semantic context, the
623 parameters can be referenced direct.

624 Figure 8 shows an EDD example with COLLECTION MEMBERS within a GROUP style
625 MENU.

```

626 VARIABLE ail_strev
627 {
628     LABEL "Analog Input 1: Static Revision";
629     TYPE INTEGER(4);
630 }
631
632 VARIABLE ail_bloMo
633 {
634     LABEL "Analog Input 1: Block Mode";
635     TYPE INTEGER(1);
636 }
637
638 MENU ail_group_double /* Leads to double display of */
639 "Analog_Input1" label
640 {
641     LABEL "Analog Input 1";
642     STYLE GROUP;
643     ITEMS
644     {
645         ail_strev,
646         ail_bloMo
647     }
648 }
649
650 COLLECTION ail_groupcol
651 {
652     MEMBERS
653     {
654         strev,ail_strev,"Static Revision";
655         bloMo,ail_bloMo,"Block Mode";
656     }
657 }
```

```

658 MENU ail_group_single
659 label /* Leads to single display of */ "Analog_Input1"
660 {
661     LABEL "Analog Input 1";
662     STYLE GROUP;
663     ITEMS
664     {
665         ail_groupcol.strev,
666         ail_groupcol.blomo
667     }
668 }
669

```

Figure 8 – Usage of COLLECTION MEMBERS in MENUs of STYLE GROUP

4.5.4 Contained items

4.5.4.1 Overview

Subclause 4.5.4 describes how a container renders contained items. All containers render contained items in the same way. Contained items, such as methods, variables and others, have a LABEL attribute. In order not to disturb the layout of the EDD application, for example, with oversized buttons for methods, the EDD developer should not use very long strings for LABELs. For this reason, some EDD applications may truncate these labels.

4.5.4.2 Methods

A method should be rendered as a button or a hyperlink. The LABEL of the corresponding METHOD should appear on the button or as the hyperlink text. When the button or hyperlink is pressed, the corresponding METHOD should be executed.

4.5.4.3 Variables

4.5.4.3.1 General

The label, value and, if defined, the units of the variable should be displayed in a manner consistent with the definition of the corresponding VARIABLE.

The general handling of variables is as follows.

- HANDLING = READ or menu item attribute is READ_ONLY: the value of the variable should not be editable.
- CLASS = DYNAMIC: the value should be updated continuously with current read values from the device.

4.5.4.3.2 Variable of type BIT_ENUMERATED

Multiple bits can be packaged in a single-bit enumerated variable. Each bit could have a distinct meaning. It is possible to display each bit separately and, in addition, to display the whole BIT_ENUMERATED variable.

In this case the variable reference should be extended with a mask in square brackets and the LABEL of the variable is not shown.

Figure 9 shows an EDD example for displaying single bits of a BIT_ENUMERATED variable.

```

698 VARIABLE var1
699 {
700     LABEL "Var 1";
701     TYPE BIT_ENUMERATED
702     {
703         { 1, "Bit 0"},
704         { 2, "Bit 1"},
705         { 4, "Bit 2"}
706     }
707 }
708
709 MENU diagnostic_info
710 {
711     LABEL "Diagnostic";
712     ITEMS
713     {
714         var1[2],          // Bit 2
715         var1[4]           // Bit 3
716     }
717 }

```

Figure 9 – Displaying single bits of BIT_ENUMERATED

BIT_ENUMERATED variables are always displayed as checkboxes, independent of one bit or all bits are shown.

Figure 10 shows an EDD example that shows the differences between the full display of a BIT_ENUMERATED variable and how it could be displayed if all bits are explicitly addressed.

```

724 VARIABLE var1
725 {
726     LABEL "Var 1";
727     TYPE BIT_ENUMERATED
728     {
729         { 1, "Bit 0"},
730         { 2, "Bit 1"},
731         { 4, "Bit 2"}
732     }
733 }
734
735 MENU var1_group
736 {
737     LABEL var1.LABEL;
738     STYLE GROUP;
739     ITEMS
740     {
741         var1[1],          // Bit 0
742         var1[2],          // Bit 1
743         var1[4]           // Bit 2
744     }
745 }
746
747 MENU diagnostic_info
748 {
749     LABEL "Diagnostic";
750     STYLE PAGE;
751     ITEMS
752     {
753         var1,              // all bit should be shown
754         COLUMNBREAK,
755         var1[1],           // Bit 0
756         var1[2],           // Bit 1
757         var1[4],           // Bit 2
758         COLUMNBREAK,
759         var1_group
760     }
761 }

```

Figure 10 – Displaying multiple bits of BIT_ENUMERATED

Figure 11 shows the result of Figure 10 EDD example in an EDD application.

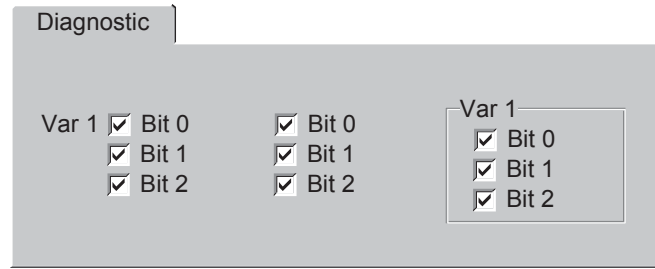


Figure 11 – Example of an EDD application for a variable of type BIT_ENUMERATED

4.5.4.3.3 Variable of type INDEX

When a VARIABLE of type INDEX is presented to the user, the text associated with the indices of the array should be displayed. If the variable is editable, it should be presented as a combo box.

4.5.4.4 Records and Collections

RECORDs and COLLECTIONs shall be displayed as group boxes (MENU with STYLE GROUP). COLLECTIONs can contain COLLECTIONs. This is handled like nested group boxes (see 4.5.3.7). The rules about nesting groupboxes apply also in this case.

4.5.4.5 Arrays and Lists

Value arrays, reference arrays and LISTs shall be rendered as a scrollable area within the container. The item LABEL shall always be visible.

Note: HART does not support arrays and lists references in containers.

4.5.4.6 Edit displays

When a container contains an edit display, the container should render the edit display as either a button or a hyperlink. The LABEL of the corresponding EDIT_DISPLAY should appear on the button or as the hyperlink text. When the button or hyperlink is pressed, the corresponding dialog box should be opened.

4.5.4.7 Graphs

A graph should be displayed in a manner that is consistent with the definition of the corresponding GRAPH. Refer to the layout rules defined in 4.6 and IEC 61804-3, 7.12.2.2 for information on the affect of the WIDTH attribute on the layout of the GRAPH.

4.5.4.8 Charts

A chart should be displayed in a manner that is consistent with the definition of the corresponding CHART. Refer to the layout rules defined in 4.6 and IEC 61804-3, 7.5.2.3 for information on the affect of the WIDTH attribute on the layout of the CHART.

4.5.4.9 Images

The images that are referenced by the MENU should be displayed. The image width will be adapted on the dialog, window, page or group width. With the menu item attribute INLINE the width of the image will be adapted on the width of the items in the same column. If an image is used without an INLINE attribute, the EDD application makes an implicit row break.

4.5.4.10 Static text

The text that is referenced by the MENU should be displayed. The text will be wrapped in multiple lines, if the text is longer than the width of dialog, window, page, group, or column.