TECHNICAL REPORT

IEC TR 61499-3

First edition
2004-06

Function blocks for industrial-process measurement and control systems –

Part 3: Tutorial information

Reference number
IEC/TR 61499-3:2004(E)

Publication numbering

As from 1 January 1997 all IEC publications are issued with a designation in the 60000 series. For example, IEC 34-1 is now referred to as IEC 60034-1.

Consolidated editions

The IEC is now publishing consolidated versions of its publications. For example, edition numbers 1.0, 1.1 and 1.2 refer, respectively, to the base publication, the base publication incorporating amendment 1 and the base publication incorporating amendments 1 and 2.

Further information on IEC publications

The technical content of IEC publications is kept under constant review by the IEC, thus ensuring that the content reflects current technology. Information relating to this publication, including its validity, is available in the IEC Catalogue of publications (see below) in addition to new editions, amendments and corrigenda. Information on the subjects under consideration and work in progress undertaken by the technical committee which has prepared this publication, as well as the list of publications issued, is also available from the following:

- **IEC Web Site (www.iec.ch)**

- **Catalogue of IEC publications**

  The on-line catalogue on the IEC web site (www.iec.ch/searchpub) enables you to search by a variety of criteria including text searches, technical committees and date of publication. On-line information is also available on recently issued publications, withdrawn and replaced publications, as well as corrigenda.

- **IEC Just Published**

  This summary of recently issued publications (www.iec.ch/online_news/ justpub) is also available by email. Please contact the Customer Service Centre (see below) for further information.

- **Customer Service Centre**

  If you have any questions regarding this publication or need further assistance, please contact the Customer Service Centre:

  Email: custserv@iec.ch
  Tel: +41 22 919 02 11
  Fax: +41 22 919 03 00

TECHNICAL REPORT

IEC TR 61499-3

First edition
2004-06

Function blocks for industrial-process measurement and control systems –

Part 3: Tutorial information

International Electrotechnical Commission, 3, rue de Varembé, PO Box 131, CH-1211 Geneva 20, Switzerland
Telephone: +41 22 919 02 11 Telefax: +41 22 919 03 00 E-mail: inmail@iec.ch Web: www.iec.ch

Commission Electrotechnique Internationale
International Electrotechnical Commission
Международная Электротехническая Комиссия

PRICE CODE X

For price, see current catalogue

CONTENTS

INTERNATIONAL ELECTROTECHNICAL COMMISSION

____

FUNCTION BLOCKS FOR INDUSTRIAL-PROCESS MEASUREMENT AND CONTROL SYSTEMS –

Part 3: Tutorial information

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. 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 61499-3, which is a technical report, has been prepared by working group 6: Function blocks, of IEC technical committee 65: Industrial-process measurement and control systems.

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

Enquiry draft	Report on voting
65/308/DTR	65/321/RVC

Full information on the voting for the approval of this technical report 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 61499 consists of the following parts under the general title *Function blocks for industrial-process measurement and control systems:*

Part 1: Architecture

Part 2: Software tools requirements

Part 3: Tutorial information

Part 4: Communication requirements

NOTE Parts 1 and 2 are under consideration.

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

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

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

INTRODUCTION

The following gives a description of the contents of the various parts of IEC 61499.

a) Part 1 contains

- general requirements, including an introduction, scope, normative references, definitions, and reference models;
- rules for the declaration of function block types, and rules for the behaviour of instances of the types so declared;
- rules for the use of function blocks in the configuration of distributed industrial-process measurement and control systems (IPMCSs);
- rules for the use of function blocks in meeting the communication requirements of distributed IPMCSs;
- rules for the use of function blocks in the management of applications, resources and devices in distributed IPMCSs.

b) Part 2 defines requirements for software tools to support the following systems engineering tasks enumerated in 1.1 of IEC 61499-1:

- the specification of function block types;
- the functional specification of resource types and device types;
- the specification, analysis, and validation of distributed IPMCSs;
- the configuration, implementation, operation, and maintenance of distributed IPMCSs;
- the exchange of information among software tools.

c) Part 3 has the purpose of increasing the understanding, acceptance, and both generic and domain-specific applicability of IPMCS architectures and software tools meeting the requirements of the other parts, by providing

- answers to Frequently Asked Questions (FAQs) regarding IEC 61499;
- examples of the use of IEC 61499 constructs to solve frequently encountered problems in control and automation engineering.

d) Part 4 defines rules for the development of compliance profiles which specify the features of IEC 61499-1 and IEC 61499-2 to be implemented in order to promote the following attributes of IEC 61499-based systems, devices and software tools:

- interoperability of devices from multiple suppliers;
- portability of software between software tools of multiple suppliers; and
- configurability of devices from multiple vendors by software tools of multiple suppliers.

FUNCTION BLOCKS FOR INDUSTRIAL-PROCESS MEASUREMENT AND CONTROL SYSTEMS –

Part 3: Tutorial information

1 Scope

This part of IEC 61499, which is a technical report, is intended to provide a simple shorthand for common functionality in a broad number of "application domains" and, to that extent, may be considered a "language". It should be noted that IEC 61499 is not a programming methodology *per se*.

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 61131-3:2003, *Programmable controllers – Part 3: Programming languages*

IEC 61804-1:2003, *Function blocks (FB) for process control – Part 1: Overview of system aspects*

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

3 Frequently asked questions (FAQ)

3.1 General questions

What is this clause about?

This clause is a compilation of responses to FAQ about the various parts of IEC 61499.

What good is IEC 61499?

IEC 61499-compliant distributed industrial-process measurement and control systems (IPMCSs), devices, and their associated life-cycle support systems will be able to deliver a number of significant benefits to their owners and system integrators, including:

a) IEC 61499-compliant life-cycle support systems will be able to **reduce engineering costs** through integrated facilities for configuration, programming and data management. Additional engineering cost savings will result from the **ease of system integration** provided by IEC 61499's simple yet complete model of distributed systems. This model provides hardware- and operating-system-independent representations for all functions of the system, including control and information processing as well as communications and process interfaces.

b) Engineers and technicians will be able to **reduce system implementation time** by applying a common set of concepts and skills to all elements of the system. Further reductions in implementation time will result from the elimination of software patches and "glueware" formerly required to integrate incompatible system elements and software tools.

   1) The elimination of patchware and glueware, the availability of interoperable software tool sets, the portability of engineering skills, and the ease of integration of system elements, will yield **higher reliability and maintainability** over the system life cycle.

2) IEC 61499 provides an abstract, implementation-independent means of representing system functions. This common "target" will lead to **simplified migration from existing systems** to IEC 61499-compliant systems, and from older to **newer technological platforms** (operating systems, communications, etc.) in IEC 61499-compliant systems.

3) Economies of scale from uniform application of common software and firmware technologies will provide **lower hardware cost per function**, since the most significant cost items in modern control hardware are its firmware and supporting software.

Is IEC 61499 a "stand-alone" document?

No. In order to realize the benefits listed above, distributed industrial-process measurement and control systems (IPMCSs) will need

- **compliance profiles** following the rules given in IEC 61499-4, including full definitions of the communication protocols utilized;
- **standard programming languages** such as those defined in IEC 61131-3 for the specification of algorithms in basic function block types;
- **libraries** of standardized and customized function block types, resource types and device types and guidelines for their application in specific domains.

How can IEC 61499 function blocks be distinguished from IEC 61131-3 function blocks?

Unfortunately, the term was already used differently in IEC 61131-3 and in the process control domain as reflected in IEC 61804. IEC 61499 defines the term most generically in terms of a distributed, event-driven architecture, of which the centralized, scanned architecture underlying IEC 61131-3 and the distributed, scanned architecture underlying IEC 61804 may be considered special cases. IEC 61131-3 and IEC 61804 may, in future, choose to specify their own **compliance profiles** following the rules of IEC 61499-4 in order to provide full harmonization of these standards.

3.2 Object orientation

Why use such a heavily object-oriented model?

The degree of object orientation used in IEC 61499 is necessary in order to achieve the required level of distributability of encapsulated, reusable software modules (function blocks).

The benefits of using object-oriented development, and the extent to which IEC 61499 realizes these benefits, are discussed in Annex B of this document.

Is this not very expensive to implement?

Not necessarily; a full object-oriented implementation is not required by the IEC 61499 model – only that the externally visible behaviour of function blocks and compliant devices conform to the requirements of IEC 61499 and possibly of compliance profiles as defined in IEC 61499-4.

Why not use a general-purpose distributed object model, like DCOM or CORBA?

Implementation of the features specified for these information-technology models would be too expensive, and their performance would almost always be too slow for use in a distributed real-time industrial-process measurement and control system (IPMCS).

Additionally, there is no standard, easily understood graphical model for representing the interconnections of events and data among these kinds of objects in distributed applications.

Are data connections and event connections a kind of object?

The **declarations** of data and event connections contained in **library elements** can be considered objects to be manipulated by software tools, as shown in Annex C of IEC 61499-1. Additionally, data and event connections are regarded as **managed objects** within **resources**, as shown in Annex C of IEC 61499-1.

Why are there no GLOBAL or EXTERNAL variables?

All variables are encapsulated. There is no guarantee that there will be an implicit global distribution mechanism available. When such mechanisms are available, they can always be mapped to service interface function blocks.

How can "contained parameters" be accessed?

External access to internal variables of function blocks is contrary to the principles of good software design. Nonetheless, to accommodate exceptional cases and previous practice, the READ and WRITE services and associated access paths to internal variables are defined for management function blocks. However, this practice may substantially degrade system performance, reliability, maintainability and safety, especially if used instead of the standard PUBLISH/SUBSCRIBE or CLIENT/SERVER services for high-rate, periodic access to variables.

Why use function blocks to model device or resource management applications?

The benefits of this approach are

- a consistent model of all applications in the system, including management applications;
- a consistent means of encapsulating and reusing all functions, including management functions;
- reuse of existing data types;
- use of existing, standardized means for the definition of required new data types and specification of management messages.

3.3 The event-driven model

Why an event-driven model?

Any execution control strategy (cyclic, time-scheduled, etc.) can be represented in terms of an event-driven model, but the converse is not necessarily true. IEC 61499 opts for the more general model in order to provide maximum flexibility and descriptive power to compliant standards and systems.

How can a change in a data value generate an event?

`E_R_TRIG` and `E_F_TRIG` function block types, defined in Annex A of IEC 61499-1, propagate an input event when the value of a Boolean input rises or falls, respectively, between successive occurrences of the input event. Instances of this type may be combined with other function blocks to produce rising- and falling-edge triggers, threshold detection events, etc.

What are event types? What are they used for?

An **event type** is an identifier associated with an event input or an event output of a function block type, assigned as part of the event input or output declaration, as described in 2.2.1.1 of IEC 61499-1. It can be used by software tools to assure that event connections are not improperly mixed, for instance, to assure that an event output that is intended to be used for initialization is not connected to an event input that is intended to be used for alarm processing.

Event types cannot be detected by execution control charts (ECCs), which control the execution of algorithms in basic function block types as defined in 2.2 of IEC 61499-1, so the type of event cannot be used to influence the processing of events in such function block types.

IEC 61499 does not define any standard event types other than the default type EVENT.

How can this model accommodate sampled-data systems?

The major issues in sampled-data systems such as those used in motion, robotic and continuous process control are

- how to achieve synchronized sampling of process or machine inputs;
- how to assure that all data has arrived in time for processing by control algorithms;
- how to assure that all outputs are present and ready for sampling before beginning the next cycle of sampling and execution.

Solutions to these problems typically require specialized communication and operating system services, which can be represented in terms of the IEC 61499 model by **service interface function blocks**. The inputs and outputs to be sampled can likewise be represented by service interface function blocks, while the algorithms to be performed will typically be encapsulated in basic function blocks. The relationships between the system services, input and output sampling, and algorithm execution can then be expressed as event connections and data connections.

What happens if a critical event is lost by the communication subsystem?

This issue is common to all distributed control systems and its solutions are well known; for example, via periodic communication, missing event detection and/or positive acknowledgement protocols. The IEC 61499 model provides for notification of abnormal operation via the IND-, CNF- and INITO- service primitives and STATUS output of service interface function blocks.

Is there any way to distinguish between "process events" and "execution scheduling events"?

These events can be distinguished from each other by the event type mechanism. They can be used by software tools to show only the event types of interest and to restrict connections to be only among compatible event types.

How can a function block respond to faults and exceptions?

In the IEC 61499 model, faults and exceptions are modelled as event outputs and associated data outputs of service interface function blocks. These outputs can be connected to appropriate event inputs and associated data inputs of any function blocks, which must respond to the fault or exception, for example, by changing their operational modes.

Where can event handling function blocks (E_CYCLE, E_RESTART, etc.) be instantiated?

The standard does not restrict the context for instantiation of event handling function blocks or service interface function blocks in general. They can, in principle, be used wherever any other function block type may be instantiated, for instance, as components of composite function blocks or as part of a resource configuration.

Are there any mechanisms to prioritize execution of algorithms or events ?

There is just some description of priority attributes for algorithms in Annex H of IEC 61499-1; however, this description is not normative.

The rules for algorithm invocation given in 2.2.2.2 of IEC 61499-1 provide a substantial degree of control over prioritization by defining specifically the processing order of transitions and actions of an Execution Control Chart (ECC).

How can IEC 61131-3 tasks with priorities and cycling time be converted to IEC 61499?

The E_CYCLE function block is intended to be used mainly for the control of cyclic execution like the cyclic tasks of IEC 61131-3. See the previous question for a discussion of priorities.

3.4 Engineering methodologies

How can I use IEC 61499 to design and implement state-machine control?

There are various formal and informal methodologies for the design of state-machine control systems. A general outline of these methods and how IEC 61499 models and software tools can be used is as follows.

a) Define the desired sequence of operations for the controlled machine or process, as well as the abnormal sequences which may occur, if possible. This may be done informally in ordinary language, or by using more formal notations such as Petri nets or IEC 61131-3 Sequential Function Charts (SFCs).

b) Define the actuators that are to be used to implement the desired behaviour, the sensors that are to be used to determine the actual state of the controlled machine or process, and their interfaces to the state-machine controller(s). IEC 61499 service interface function block types may be used for this purpose; such type definitions will typically be provided for this purpose by the supplier of the IEC 61499-compliant sensor and actuator devices.

c) Model the behaviour of the machine or process in response to commands (events plus data) given to the actuators, and the resulting, observable time-dependent outputs (events plus data) from the sensors. This modelling may be done formally using notations such as Petri nets, or informally using simulation models (which may be implemented with appropriate IEC 61499 models).

d) Define the appropriate state-machine controllers, typically as the ECCs and algorithms of basic function block types. Methodologies for doing this step may be informal, based on engineering experience, or more formal, using for example Petri-net theory. The best method is to reuse existing, proven state-machine controllers from an IEC 61499 function block library!

e) Validate the proposed state-machine controllers versus the model of the controlled machine or process. In order to catch possible design or specification errors, this would usually be done using simulation; in addition, more formal validation methods may be used if available.

f) Finally, replace the simulated sensor and actuator interfaces with the service interfaces to the actual machine or process to be controlled. This installation should normally be carried out piecewise for testing purposes.

What is an ECC? Why should I use it and when?

An ECC is a specialized state machine to enable multiple events to trigger multiple algorithms in a basic function block type, possibly dependent on some internal state. It should be used when maximum flexibility in algorithm selection and scheduling or when a high-performance, event-driven state machine is needed. Otherwise, the mechanisms in Annex D of IEC 61499-1 for converting IEC 61131-3 function blocks to IEC 61499-style blocks can be used.

Why can a composite function block not have internal variables?

Internal variables are not required in composite function blocks, since all possible sources of data are accounted for as input or output variables of the composite function block or of its component function blocks.

Are extensible user-defined function blocks permitted?

IEC 61499 follows the usage of IEC 61131-3 and does not define a specific syntax for specification of extensible inputs and outputs of user-defined function blocks. The use of the ellipsis (...) and accompanying textual descriptions is considered adequate for the specification of extensible inputs and outputs of standard and service interface function block types.

Can alternative graphical representations be used?

Software tools can use alternative graphical, textual or tabular representations, as long as the accompanying documentation specifies an unambiguous mapping to the graphical elements and associated textual syntax defined in IEC 61499-1.

How can "trend" and "view" objects be created?

In some process control terminologies, a view is a collection of data values from various sources, arranged for remote access. This function is performed by standard IEC 61499 communication function blocks.

NOTE This usage of the term "view" differs from the well-known Model/View/Controller (MVC) model for user interface applications.

A trend is a sequence of data values from the same source. The function of collecting trend data can be implemented as an algorithm of a basic function block, and remote access to the data so collected can be provided by standard communication function blocks. See 4.1.1 and 4.1.2 for further information.

Why and when should I use the WITH construct?

When you are designing function block types, you use the WITH construct to specify an association between an event input and a set of input variables, or between an event output and a set of output variables. Subclause 2.2.2.2 of IEC 61499-1 states that this association means that the associated input variables are to be sampled at a particular transition of the ECC state machine; no specific semantics are given for the WITH associations of event outputs and output variables. It is recommended, but not required, that the following interpretations be placed on these semantics.

IEC 61499-1 states that the WITH construct is used to determine

- which input variables to sample when an event occurs at the associated event input of an instance of the type;
- which output events are used to indicate when values of associated output variables change.

In either case, it is expected that this information will be used by software tools to assist the user to ensure that

- the data used by an algorithm in one function block is consistent with the data produced by an algorithm in another function block and delivered over one or more data connections associated with one or more event connections;
- the messages transmitted over communication connections between resources in a distributed application carry consistent data and events between the function block instances in the application in the way intended by the application designer.