

# 54

## Communication Aspects of IEC 61499 Architecture

---

Valeriy Vyatkin  
*University of Auckland*

Mário de Sousa  
*University of Porto*

Alois Zoitl  
*Vienna University  
of Technology*

54.1	Introduction .....	54-1
54.2	Illustrative Example .....	54-3
54.3	Logic Encapsulated in Basic FB.....	54-7
54.4	Extension.....	54-7
54.5	Distribution .....	54-9
54.6	Communication FBs .....	54-10
54.7	Communication Using Services of Internet Protocol Suite .....	54-10
54.8	Adding Distribution and Communication to the Sample System.....	54-13
54.9	Internals of Communication FBs: Modbus.....	54-15
54.10	Communication via the CIP.....	54-17
54.11	Impact of Communication Semantics on Application Behavior.....	54-19
54.12	Failures in Distributed Applications .....	54-20
54.13	Conclusion .....	54-20
	References.....	54-21

### 54.1 Introduction

---

The IEC 61499 standard [1] defines a system architecture for industrial measurement and control systems based on the concept of *function block*. A function block (FB) is a software unit (or, more generally, an *intellectual property capsule*) that encapsulates some behavior and facilitates its re-use.

The International Electrotechnical Commission (IEC) initiated the IEC 61499 project to address the limitations of the legacy PLC programming languages looking toward the realities of implementing real-time multiagent systems. Unlike previous standardization efforts, this is not a retrospective recognition of practices, but an attempt to guide future developments toward an open standard that allows genuine vendor interoperability.

FBs provide a pathway to integrate established automation programming languages, such as ladder logic, instruction list, or structured text, into modern component architecture. However, their application extends past simple replacement of legacy systems because of the inherent support for distributed applications and ability to provide a platform for modeling and simulating with well-defined interfaces. IEC 61499 defines three classes of FBs—basic FBs, composite FBs, and service interface FBs—for capturing and hiding platform-dependent functions of devices. Each FB has a set of input and output variables. The input variables are read by the internal algorithm when it is executed, while the results from the algorithm are written to the outputs.

AQ1  
AQ2