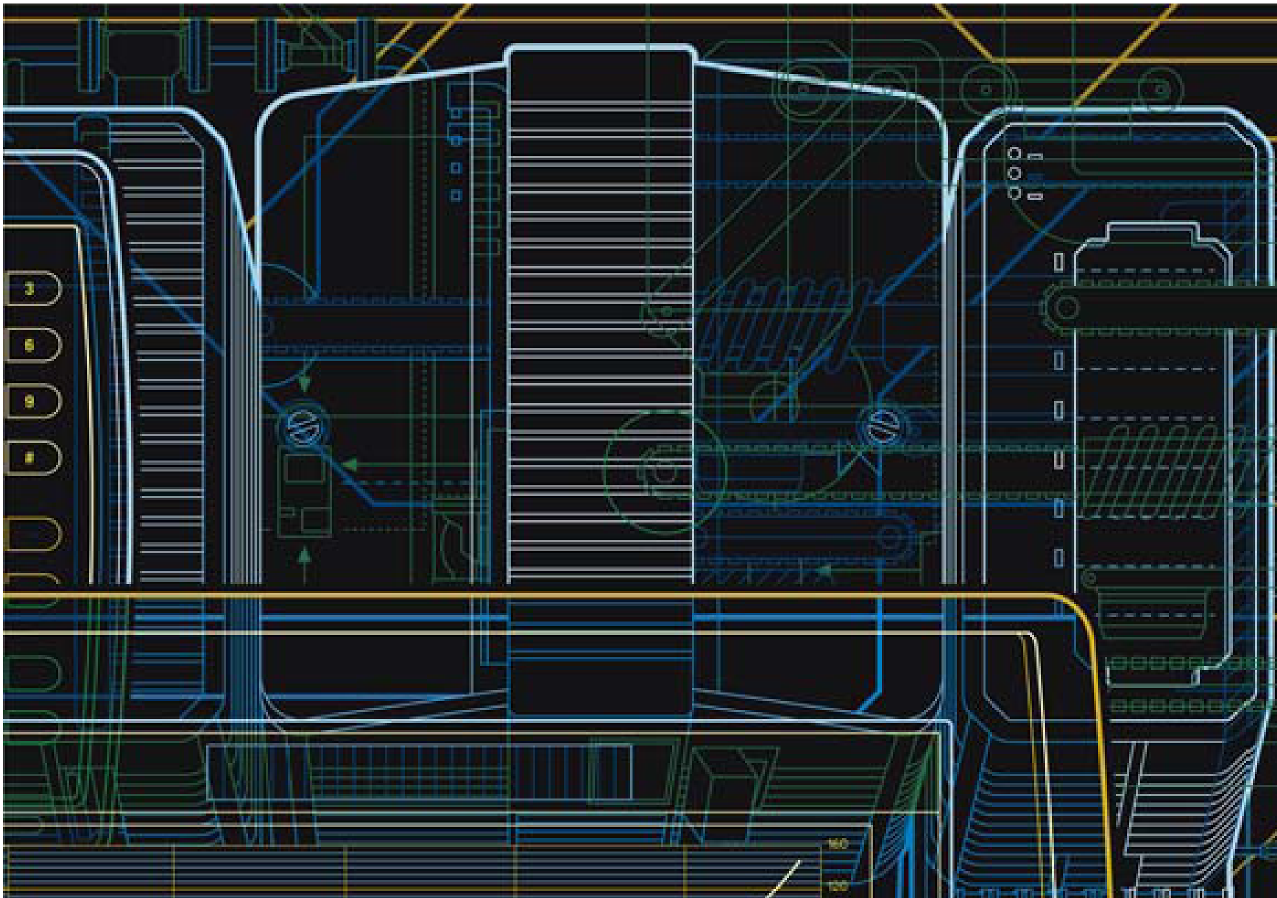


Compact Control Builder AC 800M

Product Guide

Version 5.1.1

Power and productivity
for a better world™

Compact Control Builder AC 800M

Product Guide

Version 5.1.1

NOTICE

TRADEMARKS

Release: April 2013
Document number: 3BSE041586-511

Table of Contents

About This Book

Section 1 - Key Benefits

Section 2 - Product Description

Appendix B - Performance and Capacity

Index

About This Book

Intended Use of This Book

Target Group

This Product Guide is intended for sales representatives to provide information on Compact Control Builder AC 800M and OPC Server for AC 800M.

Compact Control Builder Release Notes (*3BSE033044**) contains additional information.

Purpose, Scope and Intended Use

This book provides details on Compact Control Builder AC 800M and OPC Server for AC 800M.

Section 1, Key Benefits describes key benefits of the Compact Control Builder AC 800M and OPC Server for AC 800M.

Section 2, Product Description describes the Compact Control Builder AC 800M product and some of the components included when purchasing the Compact Control Builder AC 800M.

Section 3, Technical Data and Performance describes hardware and software requirements for operating Compact Control Builder AC 800M.

Section 4, Ordering and Licensing describes the ordering procedure, price list structure and licenses for purchasing the Compact Control Builder AC 800M and OPC Server for AC 800M.

Appendix A, Supported Hardware and I/O Families describes the hardware modules and I/O families supported by Compact Control Builder AC 800M.

Appendix B, Performance and Capacity describes performance and technical data for Control Software and Control Builder key functions.

New in this Release

The Compact Control Builder version 5.1.1 contains new and improved functionality compared to version 5.1.0:

Engineering Environment

- Diagram Editor with Function Diagram - Diagram is a new graphical language that graphically interconnects functions, function blocks, control modules and embedded ST and SFC code blocks on the same page.
- Search and Navigation - Communication variable references from other projects within the same system do now appear as references, if the projects are downloaded.
- Task Analysis Tool - The task analysis tool in Control Builder has been made more easy to use. The meaning of the different messages is now more precise. Time stamps have been added to the warning and error messages in the summery view of the Task Analysis tool, this makes it easier to relate between the messages and the graph.
- Display of Unit Specific System Alarm and Event Generation - There is a new menu entry in Control Builder on hardware types called Unit Specific System Alarm and Events. It brings up a new user interface that shows what system alarms and events a unit can generate. The information is shown in Control Builder rather than being printed in a user manual.
- Heap Utilization Dialog Available in Online Mode - The controller heap utilization dialog in Control Builder is now available also in online mode.
- Additional Caution Dialog at Download with Init Restart - An additional caution dialog is shown before download of changes in case the user or the system suggested an init restart (i.e. loss of all cold retained values).
- Compiler Warning in case Data Type used for IAC has been changed
  If the user changes a structured data type used for communication variables, then the receiving controller will get ISP until both server and client have been downloaded to. With this new version, the user gets to know in advance in case

a download of changes includes changed data types that will cause ISP on any communication variable.

Control and I/O

- CI873 EtherNet/IP

  The CI873 EtherNet/IP has been improved. It now supports master redundancy by using two CI873s. The redundancy does not require the slaves to have the EtherNet/IP redundancy connection method implemented. There is now also support for native EtherNet/IP devices. CI873 supports logical segment Class 1 connection for reading and writing data to EtherNet/IP devices, and it originates Class1 for tag reading and Class 3 for tag writing to Allen Bradley Logix 5000 series PLCs.

- CI868 Enhancements
  - IEC 61850 Wizard Tool improvements
  - IEC 61850 MMS Client functionality for CI868 Module
- Self-defined UDP and TCP Communication

  The new UDPCommLib and TCPCommLib contains function block types that are used for self-defined UDP and TCP communication. These function blocks are used when the controller needs to communicate with external equipment. The used protocol is UDP and TCP, running on Ethernet. The function is similar to the already existing self-defined serial communication, but on Ethernet. It uses the inbuilt CN1 and CN2 on the PM8xx CPU. Redundancy is handled by RNRP.

- Support for DI818, DO818, DI828, DO828

  This version supports four new S800 I/O modules on modulebus and PROFIBUS via CI801/CI840.

  - DI818 - 32 channel digital input for 24VDC
  - DO818 - 32 channel digital output for 24VDC
  - DI828 - 16 channel digital input for High Voltage AC/DC
  - DO828 -16 channel relay output for High Voltage AC/DC

- More memory with PM851A, PM856A, PM860A - These renewed CPUs have with this version more available memory:
  - PM851A - Now 12 Mbyte total RAM
  - PM856A and PM860A - Now 16 Mbyte total RAM
- Backup Media in PM891- Memory cards larger than 2GB can now be used in PM891, by the introduction of support for SDHC and SDXC cards formatted as FAT32.
- Enhanced Integration of ABB Devices - This version brings new ways of integrating ABB Drives and Motor Starters into AC 800M.
  - Support for ABB standard drive ACS880 with FENA-11 and PROFINE

    The new hardware library - ABBDrvFenaCI871HwLib - provides PROFINET connectivity to the ACS880 drive via the communication adapter FENA-11.
  - Support for ABB standard drives with FPBA-01 and PROFIBUS

    The new hardware library - ABBDrvFpbaCI854HwLib - provides PROFIBUS connectivity to ABB drives via the communication adapter FPBA-01.
  - New motor starters for MNS iS on PROFINET

    The MNS iS hardware library - ABBMNSiSCI871HwLib - offers support for the two new motor starter types - Sace Circuit Breakers (Sace CBR) and DC Feeder (DC MFeed).
- Improved Analog Control
  - External Reset Feedback - The PidCC and PidAdvancedCC control modules have a new mode called External Reset Feedback. The controller follows an auxiliary value in this mode.
  - Disable PD part - It is now possible to disable the PD part at windup situations in the PidCC and PidAdvancedCC control modules. This can be useful if an override controller shall not take any action until its epsilon changes sign.

- Auto tuning flag - The PidCC and PidAdvancedCC control modules have a new parameter indicating to the outside that auto tuning is currently being performed.
- Epsilon available as a parameter - The PidCC and PidAdvancedCC control modules have a new parameter indicating the value of Sp-Pv.
- Gain Scheduling based on Epsilon - Gain scheduling based on the value of epsilon is added to the PidAdvancedCC control module.
- Continuous Moving Average function - New control module, TimeAverageCC, that determines the moving average of an analog input over a specified number of samples.

• Functions to convert an IP Address to/ from String/Dword

There are two new firmware functions to be used for conversion between dword and a string value containing an IP address.

- DWordToIPString converts a dword to a string data type in an IP version 4 address formats.
- IPStringToDWord converts a string data type in an IP version 4 address formats to a dword.

Section 1 Key Benefits

This section is focused on getting you acquainted with the key benefits for the Compact Control Builder AC 800M software products.

Compact Control Builder AC 800M

Compact Control Builder AC 800M aims to meet the customers need for a modern industrial PLC solution, capable of handling mid-sized to large applications. Its primary target market is the process automation area, where PLC products are used, however, it can also be used for other application areas.

The Compact Control Builder software product contains the following components:

- Compact Control Builder AC 800M
- OPC Server for AC 800M
- Base Software for SoftControl

These products are delivered out of the box and easy to install, run and maintain. For more information about the Compact Control Builder software product offering, see Price Lists Structure on page 49.

Compact Control Builder AC 800M

Compact Control Builder AC 800M adds the following key benefits to the PLC market:

- Programming tool for AC 800M PLCs
  - Contains a compiler, programming editors, standard libraries for developing PLC applications and standard hardware types (units) in libraries for hardware configuring.
- Programming environment
  - Testing the application off-line.
  - Download to PLC via serial communication or Ethernet.
  - Online change on applications.
  - Cold retain of data (kept at cold start).
  - Backup/restore of projects.
- Support for all IEC 61131-3 languages
  - Function Block Diagram (FBD), Structured Text (ST), Instruction List (IL), Ladder Diagram (LD), Sequential Function Chart (SFC).
- Extensions to IEC 61131-3 languages
  - Function Diagram (FD) and Control Module Diagram (CMD).
- Create/Change/Insert Libraries
  - Creating self-defined libraries containing data types, function block types etc. which can be connected to any project.
  - Creating self-defined libraries with hardware types.
  - When no available hardware library is sufficient, the Device Import Wizard can be used to import a customized hardware type from a device capability description file.

    You can import PROFIBUS GSD-files with hardware types for CI854, and not for CI851. (However, when you upgrade a previous system offering, any included hardware types for CI851 will be upgraded as well.)

You can also import PROFINET GSD files for CI871, and DeviceNet and EtherNet/IP EDS files for CI873.

- Various functions and type solutions for simple logic control, device control, loop control, alarm handling etc. packaged as standard libraries.
- The open library structures provide easy access to set-up and connect type solutions into self-defined libraries and/or applications before programming.

- Multi-user engineering
  - Project files can be distributed on Compact Control Builder stations (up to 32 stations).
- Redundancy functions
  - AC 800M CPU redundancy (using PM861, PM864, PM866, or PM891).
  - Redundant Control Network on MMS and TCP/IP, using Redundant Network Routing Protocol (RNRP).
  - Master and line redundancy (PROFIBUS DP-V1) for AC 800M (CI854 interface module).
  - Redundant optical ModuleBus.
- Clock synchronization
  - 1 millisecond clock synchronization accuracy between PLC nodes in control network.
  - Generating Sequence-Of-Events (SOE), using time stamps for digital I/O with high accuracy.
  - System alarm and system event functions.
- ABB Drives support
  - ABB Standard Drives.
  - ABB Application Drives.

- Interfacing with Satt I/O
  - CI865 unit for Satt I/O system (Rack I/O and Series 200 I/O) with the AC 800M PLC platform.
  - 200-RACN ControlNet I/O adapter for rack-based I/O boards.
  - 200-ACN unit for 200 I/O units via Satt ControlNet.
- Compact Flash (CF) and Secure Digital (SD/SDHC)
  - Store a compiled PLCs configuration, that can be used at restart of the PLC.

OPC Server for AC 800M

OPC server for AC 800M is a stand-alone product that support both Data Access and Alarm/Event traffic from PLCs.

- Stand-alone OPC Server, fully OPC compliant.
  - OPC Server DA that handles run-time data.
  - OPC Server AE that handles alarm and event from the control system, via the OPC Server to the OPC client.
  - OPC Server Online help.

SoftController

- Testing tool for running applications offline.
  - SoftController provides reduced engineering and test costs.
  - It is a simulation tool that runs with Base Software for SoftControl and is automatically installed together with the Compact Control Builder.

Section 2 Product Description

This section describes the Compact Control Builder AC 800M product and some of the components included when purchasing the Compact Control Builder AC 800M.

The Compact Control Builder is used to configure the AC 800M hardware. The OPC server is used to connect the AC 800M to a HMI or SCADA system.

Compact Control Builder offers amongst other things multi-user engineering and support for redundancy functions (CPU redundancy, RNRP, master and line redundancy with CI854). More information can be found in Compact Control Builder AC 800M Functions on page 22.

The OPC Server runs stand-alone and is fully OPC Data Access and Alarm/Event OPC compliant.

Compact Control Builder AC 800M supports the following CPUs:

- PM851/PM851A
- PM856/PM856A
- PM860/PM860A
- PM861/PM861A
- PM864/PM864A
- PM866
- PM891
- SoftController running on PC

Software Overview

The software delivered on the DVD is divided in two parts - the Compact Control Builder AC 800M and OPC Server for AC 800M. While installing Compact Control Builder additional components and services will be installed in the background.

- Compact Control Builder AC 800M
  - Base Software for SoftControl
  - RNRP
  - User Documentation
- OPC Server for AC 800M

Compact Control Builder AC 800M

Compact Control Builder AC 800M is a programming tool for creating PLC based control solutions when using the AC 800M as hardware. It works on Windows 7 or Windows Server 2008 platform.

Overview

Firmware and applications can be downloaded to PLCs using Ethernet or via a direct serial link. Ensure that the IP address of the PLC is configured in Control Builder, communication is set up, and the cables are connected at both ends. An OPC Server for AC 800M can be installed on the same PC as Control Builder (Figure 1) or be installed on a separate PC, typically together with Human Machine Interface (HMI) software.

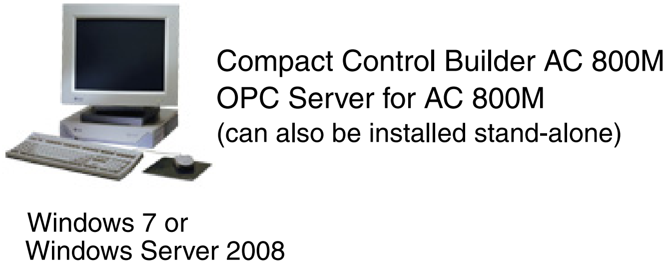


Figure 1. Control Builder and supporting software.

Download from Programming Station

PLC firmware and control applications can be downloaded from a standard PC to PLCs using Ethernet or via a direct serial link (using TK212A cable).

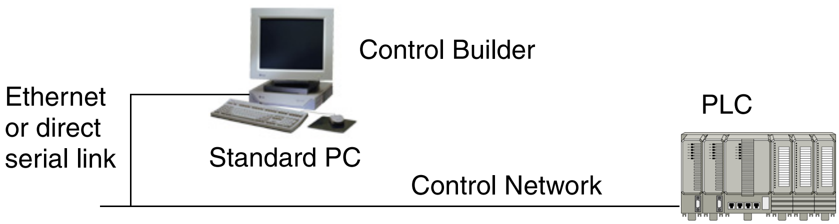


Figure 2. Downloading firmware and/or applications.

PLC Communication

PLCs, programming stations and operator stations communicate with each other through the control network. The control network is used to communicate between Control Builder stations and the PLCs, between HMI and PLCs and also for communication between the PLCs.

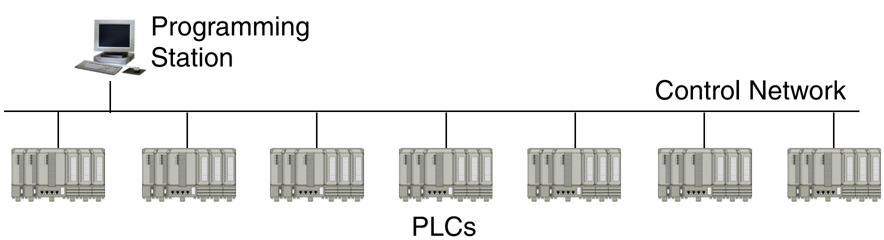


Figure 3. PLC communication in control network.

Compact Control Builder AC 800M Functions

Compact Control Builder supports a number of functions:

- Support for IEC 61131-3 Languages on page 22.
- Testing the Application on page 24.
- Downloading to a PLC on page 24.
- Multi-user Engineering on page 25.
- Alarm and Events Handling on page 25.
- I/O Connectivity and Communication on page 26.
- Supported ABB I/O Systems and Families on page 33.
- Serial Communication Protocols on page 34.
- Control Network on page 37.
- Clock Synchronization on page 38.
- Redundancy on page 38.
- Backup Media on page 40.
- Online Help and Manuals on page 41.
- Additional Software on page 42.

Support for IEC 61131-3 Languages

The IEC 61131-3 standard defines five of the most commonly used programming languages on the market. These are Function Block Diagram (FBD), Structured Text (ST), Instruction List (IL), Ladder Diagram (LD) and Sequential Function Chart (SFC). In addition to these, Control Builder supports creation of logic using Diagrams (which use the Function Diagram (FD) language) and Control Module Diagrams.

Depending on previous experience, programmers often have their own personal preference for a certain language. All the languages have advantages and disadvantages, and no single one of them is suitable for all control tasks.

Table 1. Compact Control Builder programming languages.

Language	Function
Function Block Diagram (FBD)	A graphical language for depicting signal and data flows through function blocks and functions. Function blocks and variables are interconnected graphically, which makes the resulting control diagrams easy to read.
Structured Text (ST)	A high-level programming language. ST is highly structured and has a comprehensive range of constructs for assignments, function/function block calls, expressions, conditional statements, iterations, etc. It is easy to write advanced, compact, but clear ST code, due to its logical and structured layout.
Instruction List (IL)	A traditional PLC language. It has a structure similar to simple machine assembler code.
Ladder Diagram (LD)	Ladder diagram (LD) is a graphical language based on relay ladder logic.
Sequential Function Chart (SFC)	Sequential function chart (SFC) is a graphical language for depicting the sequential behavior of a control program.
Function Diagram (FD)	Function Diagram (FD) is a graphical language that allows mixing of functions, function blocks, control modules, and diagrams in one code block and create graphical connections between them.

Testing the Application

The Compact Control Builder provides two ways for testing an application, Test mode and simulating an application with the SoftController.

Test Mode

Test mode is normally used for testing smaller parts of an application and without performing a download to the PLC. In Test Mode, Compact Control Builder compiles and executes the code in the local PC similar to the execution on PLC.

SoftController

The Base Software for SoftControl is a software product that comes with the Compact Control Builder installation. It is used for simulating a complete application (with a complete hardware configuration done). But, instead of downloading the application to a PLC, it can be downloaded to the SoftController, thus no need for a real PLC and I/O.

Downloading to a PLC

Firmware

Firmware is the software that provides the basic functionality of the AC 800M PLC. It contains functions like operating system, real-time clock, communication etc. The firmware is stored in electrically erasable programmable read-only memory (EEPROM).

The firmware is pre-installed in some of the hardware. The firmware can also be downloaded from Compact Control Builder to CPUs and communication modules either through Ethernet or through Serial Cable. If Ethernet is used as media, the IP address of the PLC must be set before any download. This is carried out with the IP Configuration tool, see also IP Configuration Tool on page 42.

Ensure that the application program in the PLC is removed before downloading the new firmware to the PLC.

After the firmware is updated, the application program has to be downloaded again and a cold start of the CPU must be performed.

Applications

Applications can be downloaded to the PLC via Ethernet or direct via a serial connection (TK212A cable). An application can be distributed between several PLCs. Parts of the application are then downloaded to different PLCs.

Multi-user Engineering

Compact Control Builder supports multi-user engineering with a maximum of 32 separate Control Builder PCs. In a multi-user configuration all Control Builder PCs and the OPC Server must have access to the common project file(s). This means that a common Project folder must be created on a shared network server.

Alarm and Events Handling

Compact Control Builder handles alarm and events generated internally in the system, a PLC or other hardware unit or in applications.

Alarm and event information is communicated throughout the control network via OPC servers, that is, a number of OPC Server for AC 800M.

Alarm and event handling supports the following.

- Disabling and enabling of alarms
- Acknowledgement and cancellation of alarms
- Filtering of alarms and events
- Printing of alarm and event lists on local printer
- System events and alarms

System events and alarms created in PLC can be read and accessed by operators through HMI. The time stamps and attributes are also created in PLCs. The event or alarm has its origin attached to it.

OPC Server

Alarms and events are collected and forwarded by the Alarm and Event (AE) part of the OPC server, see also OPC Server Alarm and Event (AE) Part on page 44. PLCs then gain access to alarms and events from other PLCs by reading data from the OPC server. Alarm and event information can also be read by other OPC clients.

I/O Connectivity and Communication

Control Builder supports a number of fieldbuses and I/O systems. PLCs can be connected to fieldbuses and other I/O systems using adapters and I/O units belonging to ABB I/O families.

I/O Connectivity

- **ModuleBus**

  ModuleBus is an integrated master unit for S800 I/O. I/O units connected to ModuleBus are divided into clusters. 12 I/O units can be directly connected to the ModuleBus on the AC 800M, while the remaining I/O units have to be connected via I/O-clusters. Up to 7 I/O-clusters can be connected to the ModuleBus. PM851 only allows up to 24 S800 I/O units on ModuleBus (12 local and 12 on cluster 1).

- **PROFIBUS DP**

  Control Builder supports the fieldbus system PROFIBUS DP. It can be connected to PLCs via the CI854 interface module, offering master and built-in line redundancy.
  Applications access the built-in fieldbus functions through corresponding I/O modules.

- **PROFINET IO**

  PROFINET is a manufacturer-independent Fieldbus standard for applications in manufacturing and process automation. PROFINET technology is described in fixed terms in IEC 61158 and IEC 61784 as an international standard.

  PROFINET IO uses Ethernet communication to integrate simple distributed I/O and time-critical applications.

  PROFINET IO describes a device model oriented to the PROFIBUS framework, which consists of places of insertion (slots) and groups of I/O channels (subslots). The technical characteristics of the field devices are described by the General Station Description (GSD) on an XML basis. The PROFINET IO engineering is performed in a way familiar to PROFIBUS. The distributed field devices are assigned to the PLCs during configuration.

The PROFINET IO is interfaced to the IEC 61131 PLC AC 800M, using the PROFINET IO module CI871.

- **DriveBus**

  The CI858 unit is the communication interface for the DriveBus protocol. ABB Drives and Special I/O units communicate with the AC 800M PLC via the CI858 unit. The CI858 Drive channel can be used to connect up to 24 drives.

- **S100 I/O**

  The CI856 is the AC 800M communication interface for the S100 I/O system The CI856 unit handles the I/O configuration and I/O scanning of up to five S100 I/O racks where each I/O rack can hold up to 20 I/O boards.

- **Satt I/O**

  The CI865 unit is the AC 800M communication interface for Satt I/O. The CI865 unit makes it possible to use older Satt I/O system (Rack I/O and Series 200 I/O) with the PLC.

- **INSUM**

  INSUM (INtegrated System for User-optimized Motor control) is a system for motor and switch gear control and protection from ABB. PLCs can be integrated with INSUM by means of a TCP/IP gateway and a CI857 interface module (Figure 4).

INSUM and Control Network must use separate physical networks.

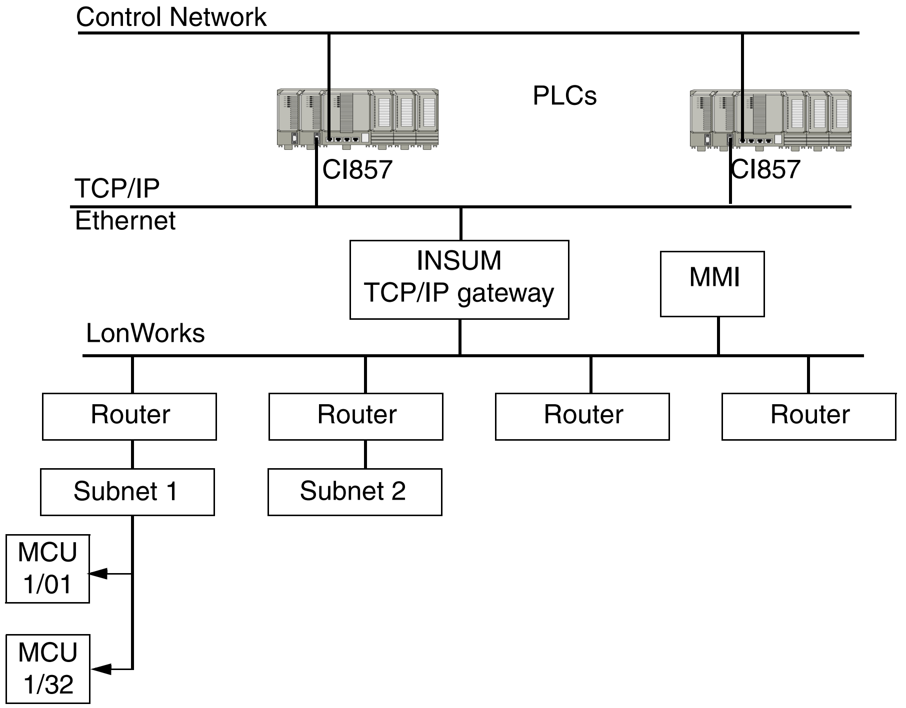


Figure 4. INSUM integration with PLCs.

The TCP/IP gateway connects PLCs to the Local Operating Network (LON) fieldbus. Motor Control Units (MCUs) are grouped into sub-networks accessed through a number of routers.

INSUM applications handle motor and switch gear control. They can also be set to send alarm and event information to a PLC through the TCP/IP gateway.

The INSUM operator station gives direct access to INSUM functions. PLCs also have access to INSUM functions through the function blocks in the INSUM library.

- **IEC 61850**

  The IEC 61850 for Substation Automation System (SAS) defines communication between intelligent Electronic Devices (IED) in the substation and other related equipment. The IEC 61850 standard itself defines the superset of what an IEC 61850 compliant implementation might contain.

- **Advant Fieldbus 100**

  Advant Fieldbus 100 (AF 100) is a high performance fieldbus, which is used for:

  – Communication between Advant Controllers.
  – Communication between Advant Controllers and S800 I/O Stations, AC 800M PLCs, AdvaSoft for Windows, and the equipments developed and sold by other ABB companies.

  The CI869 communication interface that is attached to the AC 800M PLC provides connectivity to other AC 800M, AC 160 or connectivity server over AF 100. An AC 800M PLC with the communication interface CI869 behaves as an AF 100 station, receiving data from other AF 100 stations/devices. The CI869 has integrated Twisted Pair modems.

- **EtherNet/IP and DeviceNet**

  The Industrial Ethernet Protocol (EtherNet/IP) is an application layer protocol built on the standard TCP/IP protocol suite used to communicate with high-level industrial devices.

  DeviceNet is an application layer protocol built on the standard Controller Area Network (CAN). It is used to communicate with low-level industrial devices.

  DeviceNet and EtherNet/IP are based on Common Industrial Protocol (CIP) and share all the common aspects of CIP.

  The following are the software components implemented in EtherNet/IP:

  – CI873 EtherNet/IP Hardware Library (CI873EthernetIPHWLib).

  – Device Import Wizard (DIW) to import the EDS files into Control Builder.

  The CI873EthernetIPHWLib integrated with AC 800M provides CEX based Communication interface along with three components of CI873 protocol for the Control Builder, PLC, and CEX module CI873.

  The CI873EthernetIPHWLib provides the following functionalities:

  – Configuring CI873 as EtherNet/IP scanner.

  – Class 1 connection to LD 800DN for I/O communication with DeviceNet devices.

- System command to change the Run/Idle state of LD 800DN.
- LD 800DN Scanner diagnostics.
- Status supervision of devices.
- Hot swap of CI873, LD 800DN and DeviceNet devices.
- Logging of CI873 messages.
- CI873 Scanner diagnostics.
- CI873 Firmware Upgrade.

The Device Import Wizard (DIW) is an integrated component of the Control Builder. The DIW converts the device description files – EDS files of DeviceNet devices – into Hardware Definition (HWD) files. These unit types can be instantiated in the Hardware tree of Control Builder.

Communication

- **IAC**

  Inter Application Communication (IAC) is defined as the variable communication between applications that use a special category of variables called communication variables. The applications can reside in the same PLC or in a different PLC in the project. IAC is possible within an application also. IAC is supported by the MMS protocol, and it uses an IP based resolution for communication between applications.

  IAC is based on the name of the communication variables and the IP address of the controllers to which the applications are downloaded.

- **MMS**

  The MMS protocol defines communication messages transferred between PLCs as well as between engineering stations (such as Compact Control Builder) and the PLC (e.g. downloading an application or reading/writing variables).

- **MasterBus 300**

  The MB 300 supports both network redundancy and clock synchronization (with the accuracy offered by MB 300).

Note that MasterBus 300 and Control Network must use separate physical networks.

- **MODBUS TCP**

  MODBUS is an open industry standard widely spread due to its ease of use. It is a request response protocol and offers services specified by function codes. MODBUS TCP combines the MODBUS RTU with standard Ethernet and universal networking standard TCP. It is an application-layer messaging protocol, positioned at level 7 of the OSI model.

  MODBUS TCP communicates via the CI867 communication interface unit. CI867 is a dual channel Ethernet unit; Ch1 and Ch2. Ch1 supports full duplex with 100 Mbps speed and Ch2 supports half duplex with 10 Mbps speed.

  Both master and slave functionality are supported. A maximum of 70 slave and 8 master units per CI867 (on Ch1 and Ch2 together) can be used.

  Function blocks are used for master communication and access variables is used for slave communication.

  A number of MODBUS TCP commands are supported. Protocol functions are accessible through function blocks.

  Table 2 describes the protocol commands that are supported by MODBUS TCP.

Table 2. Supported MODBUS TCP protocol commands

Protocol	Description	Protocol	Description
FC 1	Read coils	FC 8	Diagnostic
FC 2	Read input discreet	FC 15	Force multiple coils
FC 3	Read multiple registers	FC 16	Write multiple registers
FC 4	Read input register	FC 20(1)	Read file record
FC 5	Write coil	FC 21(1)	Write file record
FC 6	Write single register	FC 23	Read Write file record
FC 7	Read exception status		

(1) Supported in Master only.

- **SattBus**

Compact Control Builder supports SattBus on Ethernet only!

SattBus is a network standard for PLC communication. SattBus can be used as a low-cost fieldbus for collection of small amounts of data under hard conditions.

- **Self-defined UDP Communication**

The UDP hardware library (UDPHwLib) contains the UDPProtocol hardware type that is used for self-defined UDP communication. The following function block types are available:

– UDPConnect

– UDPWrite

– UDPRead

- **Self-defined TCP Communication**

The TCP hardware library (TCPHwLib) contains the TCPProtocol hardware type that is used for self-defined TCP communication. The following function block types are available:

- TCPClientConnect
- TCPServerConnect
- TCPWrite
- TCPRead

Supported ABB I/O Systems and Families

Control Builder supports the following common ABB I/O systems and families.

- S800 I/O, a distributed modular I/O system for communication via ModuleBus and PROFIBUS DP.
- S900 I/O, a remote I/O system (for hazardous areas) that can be connected to PLCs via PROFIBUS DP.
- S200 I/O and S200L I/O, two compatible, modular I/O systems. S200 I/O modules can be connected via CI856 or PROFIBUS DP to PLCs.
- S100 I/O, a rack-based I/O system that can be connected to PLC using the CI856 interface module.
- Satt I/O, makes it possible to use Satt Rack I/O (an older Satt I/O system) connected to PLC using the CI865 communication interface.

Serial Communication Protocols

Control Builder supports a number of serial communication protocols for Compact Control Builder products and third party HMI. These protocols can be used for communication between PLCs, as well as with other devices.

ModBus RTU

ModBus is a wide-spread communication protocol that can be used on a variety of media, such as wire, fiber optics, radio and telephony. ModBus is an asynchronous serial master/slave protocol that is executed in half-duplex.

The Compact Control Builder software only supports ModBus RTU **master** functionality.

ModBus RTU protocol functions are accessible through function blocks. The following protocol commands are supported:

Table 3. Supported ModBus protocol commands

Protocol	Description	Protocol	Description
FC1	Read coil status	FC6	Preset single register
FC2	Read input status	FC7	Read exception status
FC3	Read holding registers	FC8[1]	Diagnostic request
FC4	Read input registers	FC15	Force multiple coils
FC5	Force single coil FC16	FC16	Preset multiple registers

(1) Some slaves do not understand FC8. To avoid problems, set Poll Time to zero (0).

COMLI

COMLI is a protocol for data transmission between PLCs from ABB. It is designed for asynchronous master/slave communication in half-duplex. COMLI can be used for serial communication.

The Compact Control Builder software supports COMLI master and slave functionality.

The following COMLI services are supported:

Table 4. Supported COMLI services

Message Type	Description	Limitation
0	Transfer I/O bits or a register	Bit 0 to 37777 (octal) and register 0 to 3071 (decimal)
2	Request several I/O bits or registers	Bit 0 to 37777 (octal) and register 0 to 3071 (decimal)
3	Transfer individual I/O bits	Bit 0 to 37777
4	Request individual I/O bits	Bit 0 to 37777
<	Request high registers	Registers 0 to 65535 (decimal)
=	Transfer high registers	Registers 0 to 65535 (decimal)
J	Transfer date and time	Clock synchronization of COMLI slave

Siemens 3964R

Siemens 3964R is a standard serial, point-to-point master/slave protocol. It can be used on any RS-232C or RS-485 channel. It is suitable for communicating with PLCs and devices with Siemens 3964R support. Communication requires installation of the RK512 interpreter in the slave system.

Compact Control Builder software supports only the Siemens 3964R **master** protocol, thus no support for slave protocols.

The following Siemens 3964R services are supported:

Table 5. Supported Siemens 3964R Services

Service	Direction	Comment
“E” message, data type D	AC 800M to Siemens PLC	Request for data, register
“E” message, data type E, A, M	AC 800M to Siemens PLC	Request for data, byte

Table 5. Supported Siemens 3964R Services

Service	Direction	Comment
"E" message, data type E, A, M	AC 800M to Siemens PLC	Request for data, bit
"E" message, data type D, E; A, M	Siemens PLC to AC 800M	Answer to request for data
"A" message, data type D	AC 800M to Siemens PLC	Transfer of data, register
"A" message, data type D	AC 800M to Siemens PLC	Transfer of data, bit
"A" message, data type D	Siemens PLC to AC 800M	Answer to transfer of data

Modem Communication

There are two types of modem that can be used with Control Builder:

- Short-distance modems using PPP, COMLI, Siemens 3964R, ModBus RTU or PROFIBUS DP.
- Dial-up modems using public telephone communications, COMLI is the only protocol for which dial-up modem communication is supported.

If COMLI is not used, it is still possible to set up serial modem communication using a phone line. In this case, the communication can be between Control Builder and a PLC, or between an external system and a PLC (using AutoConnect).

There are two main reasons for using modem communication:

1. A need for increasing the maximum length of RS-232C, RS-485 and Ethernet twisted-pair connections.
2. A need for using fiber-optic communication, to eliminate either electromagnetic interference or the risk of intrusion.

Self-defined Serial Protocol

Function blocks in SerialCommLib allow implementation of a personal character oriented protocol on a serial port. It supports writing an application that both controls the characters sent and checks that the correct answer is received by using various checksum algorithms. The serial protocol can only be executed in half duplex. Accordingly it can not send and receive simultaneously. The following function block types are available:

- SerialConnect
- SerialSetup
- SerialWriteWait
- SerialListenReply
- SerialWrite
- SerialListen

A maximum 140 characters is supported. ASCII telegrams are recommended, since binary telegrams are difficult to implement.

Control Network

The recommended alternative for communication with PLCs and other devices, is Control Network, a private IP domain designed for industrial applications. Control Network is based on MMS via Ethernet or PPP on RS-232C.

Routing and redundancy functions are handled by the Redundant Network Routing Protocol (RNRP), an ABB protocol for handling redundancy and for routing between nodes in a control network, see Redundancy on page 38.

Clock Synchronization

In cases where all PLCs must use the same time, for example when time stamps are useful, clock synchronization is needed. AC 800M supports clock synchronization by five different protocols – CNCP, SNTP, SNTP on CI, MB 300 Clock Sync and MMS Time Service.

CNCP is the normal protocol for clock synchronization on the Control Network. An AC 800M PLC selected as Clock Master multicasts synchronization messages on the network. CNCP is used if relative accuracy is needed, that is, the clocks between all AC 800M PLCs are synchronized with an accuracy of <1ms.

In addition SNTP is used if absolute accuracy of <1ms is needed. SNTP is a standardized protocol that typically is used by AC 800M PLCs that need to be synchronized from an external time server (for example a GPS receiver) which is connected to the Control Network.

SNTP on CI is a protocol that is used by AC 800M PLCs that have communication interfaces that can handle clock synchronization independently (for example, the CI869 that communicates with AF 100).

The AC 800M OPC Server supports the MMS Time Service for small systems where no AC 800M is used for backward compatibility with older products.

MB 300 Clock Sync is a protocol for clock synchronization of Advant/Master products on a MasterBus 300 network.

Redundancy

Control Builder supports the following redundancy functions:

- CPU redundancy for PLC (PM861, PM864, PM866, and PM891)
- Network redundancy (RNRP)
- Line redundancy (CI854)
- Master redundancy (CI854A)
- Optical ModuleBus redundancy
- CEX-Bus redundancy (using BC810)

CPU Redundancy

PLCs with PM861, PM864, PM866 and PM891 processors can be configured for CPU redundancy. Two CPU modules are then run in parallel, one as primary and one as secondary. If the primary CPU fails, the secondary CPU automatically takes over.

It is also possible to run a PLC in single CPU mode with PM861, PM864, PM866, or PM891.

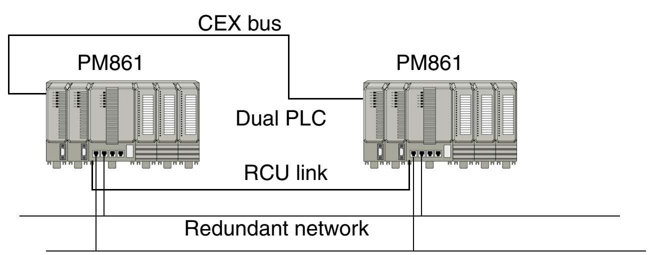


Figure 5. Example of a redundant CPU configuration.

Network Redundancy

Network redundancy is based on the Redundant Network Routing Protocol (RNRP). This protocol is an ABB protocol for handling redundancy functions and routing between nodes in a control network. The protocol is designed for rapid detection of network failure and instant switching to alternative paths.

The maximum number of RNRP nodes in a network area is limited to 50 nodes.

Network redundancy requires two independent IP networks, one primary and one secondary. Whenever the maximum number of lost messages is exceeded, the traffic is switched to the secondary network.

All devices with network redundancy must be connected to both networks. The node number must be identical in both networks.

Network redundancy can be implemented in part of the network. Nodes with one connection only must be connected to the primary network.

Line Redundancy

Line redundancy support is provided by PROFIBUS DP communication, through dual ports on the CI854 interface module. Line redundancy may be achieved for other communication by adding extra equipment.

Backup Media

The AC 800M PLCs contain a card slot located at the front of the PLC. This card slot supports backup media cards. It is possible to restore the saved configuration data and firmware data from the backup media card to the PLC.

The supported backup media cards for AC 800M PLCs are:

- Compact Flash (CF) card (supported in all AC 800M PLCs except PM891).
- Secure Digital (SD/SDHC) card (supported only in PM891).

The CF/SD memory card helps to store a compiled PLC configuration to the card and then install it into the PLC by inserting the CF/SD card. This makes it easy to distribute new software upgrades to PLCs in different locations which are not networked. The control software is installed without requiring any tool.

Compact Flash

Before downloading the application to CF card, an external Compact Flash Writer must be connected to the USB port of the Control Builder PC, if the PC does not have a built-in card reader. See also Compact Flash Requirements on page 46.

Secure Digital

Before downloading the application to SD card, an external Secure Digital Writer must be connected to the USB port of the Control Builder PC, if the PC does not have a built-in card reader. See also Secure Digital Requirements on page 46.

Cold Retain Values

The cold retain values saved by Compact Flash/ Secure Digital can either be saved cyclic via settings in the hardware editor or from the code via the function block (SaveColdRetain) located in BasicLib. Either way, these values are only saved on files located on the CF/SD card. These settings do not apply for the cold retain values saved by Control Builder or OPC Server during a download.

Cold Retain Values from a Redundant CPU Configuration

If you have a redundant CPU configuration; you cannot save cold retain values cyclic or by the function block.

However, you can always save cold retain values via the Tool menu in Control Builder so that your cold retain values will be part of the application and gets loaded to the CF or SD card.

Online Help and Manuals

Online Help

Control Builder has an extensive online help system with context-sensitive (F1) help for objects displayed in the Project Explorer. Online help can also be displayed by clicking **Help** in dialog boxes or selecting it under the **Help** menu.

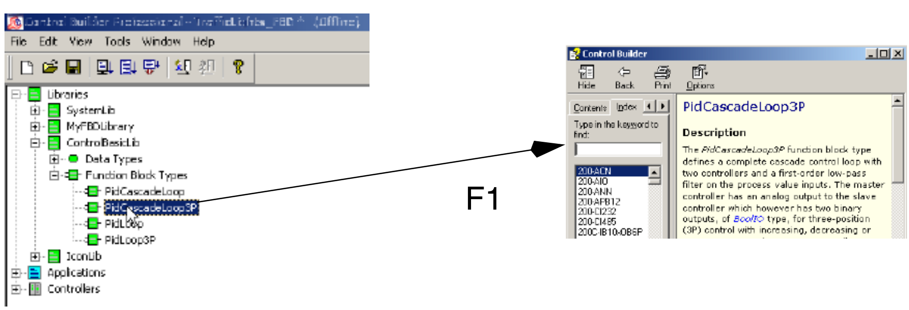


Figure 6. Context-sensitive (F1) help

Customized help can be added for self-defined libraries, applications and components of externally added applications, as well as for non-standard hardware.

Added customized files for user-defined libraries with data types, function block types and control module types as well as for applications are displayed under **User Help** on the **Help** menu.

Context-sensitive help on user-defined libraries with hardware and non-standard hardware is available if a help file (HTML or WinHelp file with any file name) is added to the library or to the hardware type.

Online Manuals

User manuals are available from Control Builder AC 800M, in Adobe Acrobat PDF format.

Additional Software

Compact Control Builder AC 800M also contains a number of additional tools and products:

- IP Configuration tool
- Serial Firmware Upgrade tool
- RNRP tool

IP Configuration Tool

The IP Configuration tool is used to set PLC IP addresses via a direct serial channel. The initial IP address must be set before downloading firmware and applications to the PLC.

Serial Firmware Upgrade Tool

The Serial Firmware Upgrade tool is used to upgrade PLC CPU firmware via a direct serial channel.

Serial Firmware Upgrade Tool cannot be used for firmware upgrade of PM891. The firmware upgrade of PM891 can be done using an SD card or from the Remote System dialog in Control Builder.

RNRP Tool

Wizard for setting up routing between two PC stations on a redundant network.

OPC Server for AC 800M

OPC Server for AC 800M gives OPC clients access to PLC data they subscribe to. The OPC server can also be used to transfer alarm and event information. It consists of two parts:

- Data Access (DA) part
- Alarm and Event (AE) part

The OPC server exposes data to the clients (DA part) and supports the transfer of alarm and event information from attached PLCs to subscribing OPC clients (AE part).

OPC Server Data Access (DA) Part

The Data Access (DA) part of the OPC server gives all OPC clients access to run-time data in PLCs.

The OPC server exposes the following data to OPC clients.

- Variables and parameters used in applications, programs, diagrams, control modules, function blocks, data structures, etc.
- Hardware configurations
- Access variables

It can also be used to store cold retain data.

The OPC server detects the following events and updates data on each.

- A new version of an application and/or a PLC configuration is downloaded.
- A new application (an application that did not previously exist) is downloaded.
- An application is deleted from a PLC.
- One application or several new ones and a PLC configuration are downloaded to a previously empty PLC.

The DA part of OPC Server for AC 800M supports the OPC Data Access 1.0a and OPC Data Access 2.05 standards.

OPC Server Alarm and Event (AE) Part

The Alarm and Event (AE) part of the OPC server subscribes to alarms and events generated by PLCs and other devices in the control network. All these alarms and events are then stored and made accessible to OPC clients.

The AE part of the OPC server also collects acknowledgements and cancellations of alarms from OPC clients and forwards them to the PLC or device in question. Clients may also disable or enable alarm conditions in PLCs or devices through the OPC server.

The AE part of OPC Server for AC 800M supports the OPC Alarm and Events 1.02 standard.

Section 3 Technical Data and Performance

This section describes prerequisites and requirements that must be fulfilled, in order for Compact Control Builder AC 800M and OPC Server for AC 800M, to function properly. It also contains a list of functions that, compared to 800xA System with Control Builder Professional, are not included in Compact Control Builder AC 800M.

For information about hardware and I/O, see Appendix A, Supported Hardware and I/O Families.

Type solutions for simple logic control, device control, loop control, alarm handling etc. are located in standard libraries. An overview of all standard libraries are described in the manual Extended Control Software, Binary and Analog Handling.

General

The PLC hardware to be used for Compact Control Builder is AC 800M only.

AC 800M High Integrity controllers are not supported, thus SIL (Safety Integrity Level) applications cannot be handled in Compact Control Builder AC 800M.

Firmware can be downloaded to PLC using Ethernet or via a direct serial link. Serial communication between Compact Control Builder and PLC is done by using the TK212A cable.

Compact Control Builder AC 800M Performance

A project in Compact Control Builder can handle up to 1024 applications. Each application can handle 64 programs and 128 diagrams at the most. A maximum of 32 Control Builder PCs can be used together in multi-user environment and up to 32 PLCs can be created and handled within a project.

OPC Server Performance

An OPC Server can handle up to 24 PLCs, while a PLC can handle up to 3 OPC Servers.

Compact Flash Requirements

Compact Flash Writer

- It is typically an external device, and not an onboard PC function.

Compact Flash Card

The following are the specifications required for the CF card used in AC 800M PLCs (PM8xx, except PM891):

- Formatted according to FAT16 or FAT32.
- Minimum read speed – 8MB/second.
- Minimum write speed – 6MB/second.
- Same (or better) ambient temperature operative range compared to the PM8xx that uses the card.

Secure Digital Requirements

Secure Digital Writer

- It is typically an external device, and not an onboard PC function.

Secure Digital Card

The following are the specifications required for the SD/SDHC card used in AC 800M PLC (PM891):

- Formatted according to FAT 16 or FAT32.
- Minimum read speed – 8MB/second.

- Minimum write speed – 6MB/second.
- Same (or better) ambient temperature operative range compared to the PM891 that uses the card.

Prerequisites and Requirements

Compact Control Builder AC 800M

The following software requirement must be fulfilled in order for Compact Control Builder AC 800M to function properly. Using other software than recommended may affect performance.

Table 6. Compact Control Builder AC 800M software requirements

Software	Requirement
Operating system	• Windows Server 2008: – R2 with Service Pack 1 – 32-bit (x86) R1 with Service Pack 2 • Windows 7 with Service Pack 1: – 64-bit (x64) – 32-bit (x86)
Printing project documentation	Microsoft Word 2010 or later
Reading online manuals	Adobe Reader version 9.0 or later

OPC Server

OPC Server for AC 800M requires the following operating system:

- Windows Server 2008:
  R2 with Service Pack 1
  32-bit (x86) R1 with Service Pack 2
- Windows 7 with Service Pack 1:
  64-bit (x64)
  32-bit (x86)

Not Supported Functions

Compact Control Builder AC 800M is similar to the 800xA System and Control Builder Professional, with a few exceptions. The Control Builder Professional in 800xA adds the following functions, to the set of functions available in Compact Control Builder:

- Online Upgrade
- Load Evaluate Go
- Batch Handling
- Audit Trail
- SFC Viewer
- High Integrity Controller for SIL applications
- CI860 for FF HSE, and CI862 for TRIO I/O
- Information routing via HART protocol

Section 4 Ordering and Licensing

This section is intended for sales representatives. It merely presents internal identity numbers for ABB price books and price lists. If you are **not** *involved in selling Compact Products 800, please disregard this section completely.*

Ordering Procedure

One purpose of the Product Guide is to support the sales representatives when ordering Compact Products 800. The price lists used can all be found in the price book of the Compact Products 800. The price book includes Compact HMI 800, Compact Control Builder AC 800M, S800 I/O, AC 800M and Panel 800.

Price Lists Structure

The Compact Products 800 offering and related price lists are organized in a price book. This price book consists of the price lists as described in Table 7.

Price Book: 3BSE045561 includes the following Price List:

Table 7. Price List

Price List	Article No.
Panel 800, Version 6	3BSE070940
Panel 800, Version 5	3BSE043387
AC 800M used for Compact Control	3BSE058196
Compact Control Builder AC 800M 5.1	3BSE058194
S800 I/O used for Compact Control	3BSE058195
Compact HMI 5.1	3BSE064096
Compact HMI 5.1 Expansion	3BSE064097
Compact HMI 5.0 Expansion	3BSE054250
Extended Warranty Time - S800 I/O, S900 I/O, Fieldbus and AC 800M	3BSE049908

Compact Control Builder AC 800M, 3BSE058194

Table 8 describes the items in the price list for Compact Control Builder AC 800M.

Table 8. Items in the price list for Compact Control Builder

Item No.	Description	Article No.
A020	Automation Sentinel Upgrade. *Automation Sentinel agreement number or SoftCare id shall be given at ordering.*	3BSE047992R1
A040	Media Box with Compact Control Builder AC 800M and OPC server for AC 800M version 5.1 This item can be ordered by users with a valid Automation Sentinel agreement for Compact Control Builder AC 800M or OPC Server for AC 800M. It includes media and documentation for Compact Control Builder AC 800M and OPC server for AC 800M. No license is included.	3BSE046066R51

Table 8. Items in the price list for Compact Control Builder (Continued)

Compact Control Builder AC 800M		
A110	Compact Control Builder AC 800M 5.1 Product Box including: - licenses for one Compact Control Builder AC 800M, one OPC Server for AC 800M, and one SoftController. - DVD with software for Compact Control Builder AC 800M, OPC Server for AC 800M, and SoftController. - firmware for AC 800M and its communication units. - manuals as pdf-files. - a Getting Started manual. 1 year Automation Sentinel Subscription included.	3BSE040360R51
A120	OPC Server for AC 800M 5.1 License for one OPC Server for AC 800M. 1 year Automation Sentinel Subscription included.	3BSE039915R51
User Documentation		
H130	Compact 800 Engineering, Compact Control Builder, AC 800M 5.1.1, Getting Started	3BSE041584-511
H140	Compact 800 Engineering, Compact Control Builder, AC 800M 5.1.1, Configuration	3BSE040935-511
H150	Compact 800 Engineering, Compact Control Builder, AC 800M 5.1.1, Planning	3BSE044222-511
H160	Compact 800 Engineering, Compact Control Builder, AC 800M 5.1.1, Binary and Analog Handling	3BSE041488-511

The other price lists in the price book contain selected products that work together with the AC 800M for Compact Control.

Licensing

The software license is delivered as part of selected product package (A110 or A120), see Table 8 for details. The product also includes one year subscription on ABBs Sentinel software maintenance program. This gives the user free upgrades of the software during this period. The period starts at the date of shipping from the factory. The Sentinel agreement can be extended by purchasing the sentinel extension pricelist items.

Software updates

To get access to software upgrades the license owner needs to register as an owner of the license. Information about how to register is delivered together with the software.

After registration software updates and product information accessible on the internet.

Upgrades

To upgrade previous versions of Control Builder to the latest version users need to purchase a subscription on ABBs Sentinel software maintenance program.

SoftCare subscriptions for System Baseline 2 Control Builder M Basic, Standard or Professional products include the right to sign up for sentinel for a discounted fee.

Each new license for Compact Control Builder AC 800M includes Sentinel subscription for one year from day of shipping from the factory. The Sentinel includes the right to get software upgrades (downloaded) during this period. Media is not included but can be purchased at a nominal fee.

Ordering Example

A system integrator gets an order for a control solution where the end customer requires two PC based HMI and three AC 800M PLCs. The PLCs are configured by two engineers and the end user does not need any PLC configuration functionality. Below are the required items.

System integrator:

- Two Compact Control Builder AC 800M (license is bought by, and kept by the system integrator)

End user:

- Three AC 800M PLCs for Compact Control
- One Compact HMI 800 license including one Server Workplace and one Client Workplace (AC 800M OPC server is included in the HMI server workplace)

Price List Items

1. From the Compact Control Builder AC 800M price list (3BSE058194), order the following items:
   – Two items A110 (Compact Control Builder AC 800M)
2. The AC 800M PLC items (CPUs, communication interfaces, accessories etc.) can be found in the price list, 3BSE058196.
3. From the Compact HMI price list (3BSE064096), order the following items:
   – One item A110 (Compact HMI 800)
   – One of the items B110, B120 or B130 (depending on number of signals)
   – One Compact HMI Operator Workplace Client (item D110-D160, dependent of the size of the server)

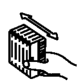
Appendix A Supported Hardware and I/O Families

For some hardware units a certain product revision is required, as described in Release Notes.

PLCs

AC 800M

The AC 800M modules supported are shown in the following table.

The symbol  on the front of a CEX bus unit indicates support for online replacement.

All communication interface units support firmware download by the Control Builder except CI858, which is upgraded with an external tool.

Supported AC 800M modules are shown in the Table 9.

Table 9. Supported Modules

Unit	Description	Online Replace-ment	Redun-dancy	Online upgrade (only valid for Control Builder Professional in 800xA)	
				Non-redun-dant	Redun-dant
PM851	PLC unit PM851 is a 32-bit, Single Board Computer, which directly connects to the S800 I/O system via ModuleBus (one electrical and one optical) and one communication interface. PM851 supports a maximum of one CEX bus module.	No	No	No	N/A
PM851A	This is a replacement for PM851 having 12 Mbyte RAM in total.	No	No	No	N/A
PM856	PLC unit PM856 is a 32-bit, Single Board Computer, which directly connects to the S800 I/O system via ModuleBus. PM856 supports a maximum of twelve CEX bus modules.	No	No	No	N/A
PM856A	This is a replacement for PM856 having 16 Mbyte RAM in total.	No	No	No	N/A
PM860	PLC unit PM860 is a 32-bit, Single Board Computer, which directly connects to the S800 I/O system via ModuleBus. PM860 is twice as fast as PM856 in executing an application program. PM860 supports a maximum of twelve CEX bus modules.	No	No	No	N/A

Table 9. Supported Modules (Continued)

Unit	Description	Online Replace-ment	Redun-dancy	Online upgrade (only valid for Control Builder Professional in 800xA)	
				Non-redun-dant	Redun-dant
PM860A	This is a replacement for PM860 having 16 Mbyte RAM in total.	No	No	No	N/A
PM861	PLC unit PM861 (Redundant and Singular) is a 32-bit, Single Board Computer, which directly connects to the S800 I/O system via ModuleBus. The unit has one optional Redundancy Control Link for redundant configuration. PM861 supports a maximum of twelve CEX bus modules.	Yes[1]	Yes	No	Yes
PM861A	This is a replacement for PM861 and can use redundant communication units.	Yes[1]	Yes	No	Yes
PM864	PLC unit PM864 (Redundant and Singular) is a 32-bit, Single Board Computer, which directly connects to the S800 I/O system via ModuleBus. PM864 is 50% faster than PM861 in executing an application program. PM864 supports a maximum of twelve CEX bus modules.	Yes[1]	Yes	No	Yes
PM864A	This is a replacement for PM864 and can use redundant communication units.	Yes[1]	Yes	No	Yes

Table 9. Supported Modules (Continued)

Unit	Description	Online Replace-ment	Redun-dancy	Online upgrade (only valid for Control Builder Professional in 800xA)	
				Non-redun-dant	Redun-dant
PM866	PLC unit PM866 (Redundant and Singular) is a high-performance, 32-bit, Single Board Computer, which directly connects to the S800 I/O system via ModuleBus. The unit has one optional Redundancy Control Link for redundant configuration. The PM866 processor unit has performance data which is approximately 1.4 times the performance of PM864.	Yes[1]	Yes	No	Yes
PM891	PLC unit PM891 (Redundant and Singular) is a high performance PLC, with four times higher memory than PM866, and about two times faster performance than PM866. PM891 is capable of handling applications with high requirements. PM891 connects to the S800 I/O system through the optical Modulebus. It can act as a stand-alone Process Controller, or as a PLC performing local control tasks in a control network.	Yes[1]	Yes	No	Yes
BC810	CEX-bus interconnection unit.	Yes	N/A	N/A	N/A
CI853	The CI853 is the RS-232C serial communication interface unit for the AC 800M. Two possible settings of the serial ports on the CI853 unit are not valid and must not be used. These are 7 data bits, no parity, 1 stop bit or 8 data bits, parity, 2 stop bits.	Yes	No	Yes[2]	N/A

Table 9. Supported Modules (Continued)

Unit	Description	Online Replace-ment	Redun-dancy	Online upgrade (only valid for Control Builder Professional in 800xA)	
				Non-redun-dant	Redun-dant
CI854	The CI854 unit is the communication interface for PROFIBUS DP/V1 for the AC 800M with redundant PROFIBUS lines and DP/V1 communication. It is a master unit and you can connect up to 124 slaves to the master. However, you cannot connect more than 32 units in one segment.	No	No	Yes [3]	N/A
CI854A	The CI854A unit is the communication interface for PROFIBUS DP/V1 for the AC 800M with redundant PROFIBUS lines and DP/V1 communication. It is a master unit and you can connect up to 124 slaves to the master. However, you cannot connect more than 32 units in one segment.	Yes	Yes	Yes [3]	Yes [4]
CI855	The CI855 unit is the communication interface for MasterBus 300 for the AC 800M. CI855 houses two Ethernet ports to support MasterBus 300 Network redundancy.	Yes	No	Yes [2]	N/A
CI856	The CI856 is a communication interface for the S100 I/O system for the AC 800M. Up to five S100 I/O racks can be connected to one CI856 where each I/O rack can hold up to 20 I/O boards.	Yes	No	Yes [3]	N/A
CI857	The CI857 unit is the communication interface for INSUM for the AC 800M.	Yes	No	Yes [5]	N/A

Table 9. Supported Modules (Continued)

Unit	Description	Online Replace-ment	Redun-dancy	Online upgrade (only valid for Control Builder Professional in 800xA)	
				Non-redun-dant	Redun-dant
CI858	The CI858 unit is the communication interface for ABB Drives using DDCS protocol for the AC 800M.	Yes	No	Yes[3]	N/A
CI865	The CI865 is the communication interface to Satt I/O on ControlNet for AC 800M.	Yes	No	Yes[3]	N/A
CI867	The CI867 unit is the MODBUS TCP communication interface for the AC 800M. CI867 houses two Ethernet ports. One port supports full duplex with 100 Mbps speed and one port supports half duplex with 10 Mbps speed.	Yes	Yes[6]	Yes[2]	Yes[2]
CI868	The CI868 unit is the IEC 61850 communication interface for the AC 800M.	Yes	No	Yes[2,7]	N/A
CI869	The CI869 is the AF 100 communication interface for AC 800M.	Yes	Yes	Yes[2]	Yes[2]
CI871	The CI871 is the PROFINET IO communication interface for the AC 800M.	Yes	No	Yes[2,3]	N/A

Table 9. Supported Modules (Continued)

Unit	Description	Online Replace-ment	Redun-dancy	Online upgrade (only valid for Control Builder Professional in 800xA)	
				Non-redun-dant	Redun-dant
CI873	The CI873 is the EtherNet/IP and DeviceNet communication interface for AC 800M.	Yes	Yes	Yes [2]	Yes [4]

NOTES:

1. Online replacement is only supported in a redundant configuration, the unit to replace MUST NOT be energized.
2. During an online upgrade, the communication between the communication interface and the connected sub units are interrupted.
3. During an online upgrade, the communication interface sets the outputs of connected I/O units to values specified by OSP control (Output Set as Predetermined).
4. Full support of online upgrade. One of the redundant communication interface units is always active during the online upgrade process.
5. During an online upgrade, CI857 is disconnected from INSUM Gateway and the connected INSUM devices keep on running with the values they have just before the switch.
6. Module redundancy only. It is not possible to get media redundancy by enabling the second Ethernet port (Ch2).
7. For CI868 Firmware Upgrade scenarios applicable during Control Builder project migration from earlier versions, refer to *AC 800M IEC 61850 Engineering and Configuration (9ARD171385*)* Manual.

Adapters for I/O Types

Table 10 shows the supported adapters for I/O types.

Table 10. Adapters for I/O Types

Adapter	Can be connected to	HART[1]	SOE[2]
TB820	PM851 and PM851A PM856 and PM856A PM860 and PM860A PM861 and PM861A (Single PLC only) PM864 and PM864A (Single PLC only) PM866 (Single PLC only) PM891 (Single PLC only)	Yes Yes Yes Yes Yes Yes Yes	Yes Yes Yes Yes Yes Yes Yes
TB840 TB840A	PM851 and PM851A PM856 and PM856A PM860 and PM860A PM861 and PM861A PM864 and PM864A PM866 PM891	Yes Yes Yes Yes Yes Yes Yes	Yes Yes Yes Yes Yes Yes Yes
DSBC 173A	CI856	No	Yes
DSBC 174	CI856	No	Yes
DSBC 176	CI856	No	Yes
CI801	CI854 and CI854A	Yes	No
CI830[3]	CI854 and CI854A	No	No
CI840	CI854 and CI854A	Yes	No
CI840A	CI854 and CI854A	Yes	No

Table 10. Adapters for I/O Types (Continued)

Adapter	Can be connected to	HART[1]	SOE[2]
CI920	CI854 and CI854A	Yes	No
CI920A	CI854 and CI854A	Yes	No
200-APB12	CI854 and CI854A	No	No
200-ACN	CI865	No	No
200-RACN	CI865	No	No
RPBA-01	CI854 and CI854A	No	No
NPBA-12	CI854 and CI854A	No	No
FPBA-01	CI854 and CI854A	No	No
RETA-02	CI871	No	Yes
FENA-11	CI871	No	Yes
MNS *i*S	CI871	No	Yes
LD800 DN	CI873	No	No

NOTES:
1. Only valid for Control Builder Professional in 800xA.
2. OPC Server for AC 800M must be used for alarms and events.
3. CI830 is replaced by CI801 at new installations.

Table 11 provides a description of the supported adapters.

Table 11. Adapter Description

Adapter	Description
TB820	ModuleBus Modem
TB840 TB840A	ModuleBus Modem, primarily for redundant ModuleBus.
DSBC 173A	The DSBC 173A unit is the bus extender slave inserted in the last position of a S100 I/O rack.

Table 11. Adapter Description (Continued)

Adapter	Description
DSBC 174	The DSBC 174 unit is the bus extender slave inserted in the last position of a S100 I/O rack.
DSBC 176	The DSBC 176 unit is the bus extender slave inserted in the last position of a S100 I/O rack.
CI801	The CI801 is a remote PROFIBUS DP-V1 adapter for S800 I/O units. The CI801 does not support redundancy. The CI801 can handle up to 24 S800 I/O-units. 12 I/O-units can be directly connected to the ModuleBus on the CI801, while the remaining I/O-units have to be connected via I/O-clusters. Up to 7 I/O-clusters can be connected to one CI801, and the numbering of I/O-units connected to a cluster will start with 101 for cluster 1, 201 for cluster 2 and so on.
CI840 CI840A	The CI840(A) is a remote PROFIBUS DP-V1 adapter for S800 I/O units, with redundancy capabilities. CI840 supports redundant I/O modules. The CI840(A) can handle up to 24 S800 I/O-units. 12 I/O-units can be directly connected to the ModuleBus on the CI840, while the remaining I/O-units have to be connected via I/O-clusters. Up to 7 I/O-clusters can be connected to one CI840(A), and the numbering of I/O-units connected to a cluster will start with 101 for cluster 1, 201 for cluster 2 and so on.
CI920 CI920A	The CI920(A) is a remote PROFIBUS DP-V1 adapter for S900 I/O units.
200-APB12	he 200-APB12 unit is a remote PROFIBUS DP slave I/O adapter for S200 I/O and S200L I/O units. 200-APB12 is connected to the PLC via a PROFIBUS DP/V0 master unit on the PLC system bus. A 200-APB12 unit can have up to eight S200 I/O units. The number of 200-APB12 slaves are, by the DIP switches, limited to 99.
200-ACN	The 200-ACN is a remote ControlNet I/O adapter for Series 200 I/O units. 200-ACN is connected to a PLC via a CI865 communication interface on the PLC system bus. 200-ACN units are used as nodes on the Satt ControlNet fieldbus. Each 200-ACN unit can handle up to eight Series 200 I/O units.
200-RACN	The 200-RACN unit is a remote ControlNet adapter for rack based I/O units. 200-RACN is connected to a PLC via a CI865 communication interface on the PLC system bus. One or several adapter 200-RACN units are used as nodes. A maximum of eight I/O-racks are supported on the Satt ControlNet fieldbus.

Table 11. Adapter Description (Continued)

Adapter	Description
RPBA-01 NPBA-12 FPBA-01	These PROFIBUS-DP adapter units are an optional device for ABB drives which enables the connection of the drive to a PROFIBUS system. The drive is considered as a slave in the PROFIBUS network. It is possible to: • give control commands to the drive (Start, Stop, Run enable, etc.) • feed a motor speed or torque reference to the drive • give a process actual value or a process reference to the PID controller of the drive • read status information and actual values from the drive • change drive parameter values • reset a drive fault.
RETA-02 FENA-11	These Ethernet Adapter modules are an optional device for ABB drives, which enables the connection of the drive to a PROFINET IO (PNIO) network. The drive is considered as a PNIO device on the PROFINET IO network, and it is compatible with all PNIO controller stations that support PROFINET IO and sub-slots. Through the Ethernet Adapter module, it is possible to: • give control commands • give control commands to the drive (Start, Stop, Run enable, etc.) • feed a motor speed or torque reference to the drive • give a process actual value or a process reference to the PID controller of the drive • read status information and actual values from the drive • change drive parameter values • reset a drive fault.
MNS *i*S	MNS *i*S is a motor control center solution that can be used in PROFINET IO network. MNS *i*S delivers all the functions for control, protection, and monitoring of motors and motor starters using software and hardware modules for the specific tasks. M*Link*, one of the interface modules in MNS *i*S, serves as the serial gateway interface to higher level systems which communicate to all modules through PROFINET IO.
LD800 DN	The LD 800DN adapter, which functions as a gateway to connect control level networks with device level networks, provides a router or bridge functionality to connect EtherNet/IP to DeviceNet. The LD 800DN provides centralized data storage for data that is shared between the DeviceNet and Ethernet/IP networks.

The following adapters are supported (Table 12), but only for migration purposes, NOT at new installations.

Table 12. Supported Adapters for Migration

Adapter	Description
CI830	The unit CI830 is a remote PROFIBUS DP-V0 I/O adapter for units. CI830 is connected to a PLC via a PROFIBUS DP-V0 master unit on the PLC system bus. The CI830 can handle up to 24 S800 I/O-units. 12 I/O-units can be directly connected to the ModuleBus on the CI830, while the remaining I/O-units have to be connected via I/O-clusters. Up to 7 I/O-clusters can be connected to one CI830, and the numbering of I/O-units connected to a cluster will start with 101 for cluster 1, 201 for cluster 2 and so on. CI830 is replaced by CI801 at new installations. CI830 does not have full support for all S800 I/O-types.

I/O Families

All I/O units may be replaced in a running system. Table 13 shows the different I/O families.

Table 13. I/O Families

I/O Family	Connects To
S800 I/O	PM851, PM851A, PM856, PM856A, PM860, PM860A, PM861, PM861A, PM864, PM864A, PM866, PM891 TB820, TB840, TB840A CI801, CI830, CI840, CI840A
S900 I/O	CI920, CI920A
ABB Standard Drives	PM851, PM851A, PM856, PM856A, PM860, PM860A, PM861, PM861A, PM864, PM864A, PM866, PM891 TB820, CI801, CI830, CI858, RPBA-01, NPBA-12, FPBA-01, RETA-02, FENA-11

Table 13. I/O Families (Continued)

I/O Family	Connects To
ABB Engineered Drives	PM851, PM851A, PM856, PM856A, PM860, PM860A, PM861, PM861A, PM864, PM864A, PM866, PM891 TB820, CI858, RPBA-01, NPBA-12, FPBA-01, RETA-02, FENA-11
S100 I/O	CI856
S200 I/O, S200L I/O and I/O 200C	200-APB12, 200-ACN
Satt Rack I/O	200-RACN

S800 I/O

Table 14 shows the different S800 I/Os.

Table 14. S800 I/O

Name	Description
AI801	Analog input unit, 8 inputs
AI810	Analog input unit, 8 inputs
AI815	Analog input unit, 8 inputs
AI820	Analog input unit, 4 differential inputs
AI825	Analog input unit, galvanic isolated analog input unit, 4 channels
AI830[1]	Analog input unit, 8 RTD inputs
AI835[2]	Analog input unit, 8 TC inputs
AI843	Analog input unit, 8 TC inputs, redundant possibilities
AI845	Analog input unit, 8 inputs, redundant possibilities, HART
AI890	Analog input unit, 8 inputs, Intrinsic Safety interface
AI893	Analog input unit, 8 RTD/TC inputs, Intrinsic Safety interface
AI895	Analog input unit, 8 inputs, Intrinsic Safety interface, HART
AO801	Analog output unit, 8 outputs

Table 14. S800 I/O (Continued)

Name	Description
AO810[3]	Analog output unit, 8 outputs
AO815	Analog output unit, 8 outputs
AO820	Analog output unit, 4 outputs
AO845	Analog output unit, 8 outputs, redundant possibilities, HART
AO890	Analog output unit, 8 outputs, Intrinsic Safety interface.
AO895	Analog output unit, 8 outputs, Intrinsic Safety interface, HART
DI801	Digital input unit, 16 inputs
DI802	Digital input unit, 8 inputs
DI803	Digital input unit, 8 inputs
DI810	Digital input unit, 16 inputs
DI811	Digital input unit, 16 inputs
DI814	Digital input unit, 16 inputs
DI818	Digital input unit, 32 inputs
DI820	Digital input unit, 8 inputs
DI821	Digital input unit, 8 inputs
DI825[4]	Digital input unit, 8 channels with event recording (SoE, Sequence of events)
DI828	Digital input unit, 16 inputs
DI830[4]	Digital input unit, 16 inputs with event recording (SoE, Sequence of events)
DI831[4]	Digital input unit, 16 inputs with event recording (SoE, Sequence of events)
DI840[4]	Digital input unit 16 inputs, redundant possibilities with event recording (SoE, Sequence of events)
DI885[4]	Digital input unit, 8 inputs

Table 14. S800 I/O (Continued)

Name	Description
DI890	Digital input unit, 8 inputs, Intrinsic Safety interface
DO801	Digital output unit, 16 outputs
DO802	Digital output unit, 8 outputs
DO810	Digital output unit, 16 outputs
DO814	Digital output unit, 16 outputs
DO815	Digital output unit, 8 outputs
DO818	Digital output unit, 32 outputs
DO820	Digital output unit, 8 outputs
DO821	Digital output unit, 8 outputs
DO828	Digital output unit, 16 outputs
DO840	Digital output unit 16 outputs, redundant possibilities
DO890	Digital output unit, 8 outputs, Intrinsic Safety interface
DP820	Digital pulse counter
DP840	Pulse/Frequency input, 8 inputs, redundant possibilities, supported in CI830 but without redundancy

NOTES:
1. AI830/AI830A.
2. AI835/AI835A.
3. AO810/AO810V2.
4. No support in CI801 and CI840.

S900 I/O

Table 15 shows the different S900 I/Os.

Table 15. S900 I/O

Name	Description
AI910N/S	Analog input unit, 4 inputs, 4-20 mA
AI920N/S	Analog input unit, 4 inputs, 4-20 mA, isolated
AI921N/S	Analog input unit, 4 inputs
AI930N/S	Analog input unit, 4 inputs, 4-20 mA, HART
AI931N/S	Analog input unit, 4 inputs, 0/4-20 mA, HART
AI950N/S	Analog input unit, 4 inputs, temperature sensor
AO910N/S	Analog output unit, 4 outputs, 4-20 mA
AO920N/S	Analog output unit, 4 outputs, 4-20 mA, isolated
AO930N/S	Analog output unit, 4 outputs, 4-20 mA, HART
DI920N	Digital input unit, 4 inputs, (NAMUR), isolated
DO910N/S	Digital output unit, 4 outputs, (for solenoid valves)
DO930N/S	Digital output unit, 4/6 outputs, dry contacts (relay)
DO940N/S	Digital output unit, 8 outputs
DO980N/S	Digital output unit, 16 outputs
DP910N/S	Frequency input and pulse counter, 2 inputs
DX910N/S	Bidirectional unit, 8 channels, (programmable) for digital input, 8 inputs, NAMUR/dry contacts or digital output, 8 outputs, for low power valves

S100 I/O

Table 16 shows the different S100 I/Os.

Table 16. S100 I/O

Name	Description
DSBC 173A/174 DSDC 176	Bus extender slave
DSAI 130 DSAI 130A	Analog input board, 16 inputs
DSAI 130D	Analog input board, 16 inputs with 4 sets of filter times
DSAI 133 DSAI 133A	Analog input board, 32 inputs
DSDI 110, DSDI 110A DSDI110AV1	Digital input board, 14 inputs, 24V
DSDI 115	Digital input board, 32 channels, 24 V
DSDI116	Digital input board, 32 channels, 24 V non-isolated
DSDI 120, DSDI 120A DSDI 120AV1	Digital input board, 32 inputs, 48 V
DSDI 125	Digital input board, 32 channels, 48 V
DSDI 126	Digital input board, 32 channels, 48 V non-isolated
DSDO 110	Digital output board, 32 outputs
DSDO 115	Digital output board, 32 outputs
DSDO 115A	Digital output board, 32 outputs, OSP control
DSDO 130	Digital output board, 16 relay outputs 24 - 240 VAC/VDC
DSDO 131	Digital output board, 16 relay outputs 24 - 240 VAC/VDC
DSAO 110	Analog output board, 4 outputs
DSAO 120	Analog output board, 8 outputs
DSAO 120A	Analog output board, 8 outputs, OSP control

Table 16. S100 I/O (Continued)

Name	Description
DSAO 130	Analog output board, 16 outputs
DSAO 130A	Analog output board, 16 outputs, OSP control
DSAX 110 DSAX 110A	Analog input/output board, 8 inputs 8 outputs
DSDP 010	Absolute binary decoder with hardware strobe, 2 channels
DSDP 140B	Positioning control board for one positioning loop
DSDP 160	Loop transducer interface board, 4 channels
DSDP 161	Loop transducer interface board, 4 channels
DSDP 170	Pulse counter board, 4 channels

S200 I/O

Table 17 shows the different S200 I/Os.

Table 17. S200 I/O

Name	Description
200-DUTB	Dummy I/O unit
200-IA8	Digital input unit, 8 inputs
200-IB10xOB6	Digital combined unit, 10 inputs and 6 outputs
200-IB16	Digital input unit, 16 inputs
200-IB16xOB16P	Digitally combined unit, 16 inputs and 16 outputs
200-IB32	Digital input unit, 32 inputs
200-IE4xOE2	Analog combined unit, 4 inputs and 2 outputs
200-IE8	Analog input unit, 8 inputs
200-IF4I	Analog input unit, 4 inputs
200-IM8	Digital input unit, 8 inputs

Table 17. S200 I/O (Continued)

Name	Description
200-IP2	Pulse counter board, 2 x 4 inputs
200-IP4	Pulse counter board, 4 x 2 inputs
200-IR8	Analog input unit, 8 inputs
200-IR8R	Analog input unit, 8 inputs
200-IT8	Analog input unit, 8 inputs
200-OA8	Digital output unit, 8 outputs
200-OB16	Digital output unit, 16 outputs
200-OB16P	Digital output unit, 16 outputs
200-OB32P	Digital output unit, 2 x 16 outputs
200-OB8EP	Digital output unit, 8 outputs
200-OE4	Analog output unit, 4 outputs
200-OF4I	Analog output unit, 4 outputs
200-OM8	Digital output unit, 8 outputs
200-OW8	Digital output unit, 8 outputs

S200L I/O and I/O 200C

Table 18 shows the different S200L I/Os and Table 19 shows I/O 200C.

Table 18. S200L I/O

Name	Description
AI210	Analog input unit, 8 inputs
AO210	Analog output unit, 4 outputs
AX210	Analog combined unit, 4 inputs and 2 outputs
DI210	Digital input unit, 16 inputs

Table 18. S200L I/O (Continued)

Name	Description
DO210	Digital output unit, 16 outputs
DX210	Digital combined unit, 10 inputs and 6 outputs

Table 19. I/O 200C

Name	Description
200C-IB10xOB6P	Digital combined unit, 10 inputs and 6 outputs
200C-IB16	Digital input unit, 16 inputs
200C-IE4xOE2	Analog combined unit, 4 inputs and 2 outputs
200C-IE8	Analog input unit, 8 inputs
200C-OB16P	Digital output unit, 16 outputs
200C-OE4	Analog output unit, 4 outputs

Satt Rack I/O

Table 20 shows the different Satt Rack I/Os.

Table 20. Satt Rack I/Os

Name	Description
IAPG	Digital input board with 16 inputs
IDLD	Digital input board with 16 inputs
IDP	Digital input board with 32 inputs
IDPG	Digital input board with 32 inputs
IDN	Digital input board with 32 inputs
IDI	Digital input board with 32 inputs
PTC	Digital input board with 32 inputs
ORG	Digital output board with 16 outputs
ORGH	Digital output board with 16 outputs
OATG	Digital output board with 16 outputs
ODP2	Digital output board with 16 outputs
ODPG2	Digital output board with 16 outputs
ORM	Digital output board with 16 outputs
ODP.5	Digital output board with 32 outputs
ODP.8	Digital output board with 32 outputs
ODPG.8	Digital output board with 32 outputs
ODPL.5	Digital output board with 32 outputs
ODPLD	Digital output board with 32 outputs
ODN.2	Digital output board with 32 outputs
ODLD.5	Digital output board with 32 outputs
ODSG	Digital output board with 32 optocoupled outputs, short circuit proof

Table 20. Satt Rack I/Os (Continued)

Name	Description
IBA	Analog input board with 8 inputs
IRA	Analog input board with 8 inputs
ICA	Analog input board with 8 inputs
IVA	Analog input board with 8 inputs
IVAPOT	Analog input board with 8 inputs
OCVA	Analog output board with 2 outputs
OCAHG	Analog output board with 4 outputs
OCAH	Analog output board with 4 outputs
OCAH with handstation	Analog output board with 4 outputs
IPA4	Input pulse analyzer board with 4 inputs, 8 bit counters

Drives System

There are two types of drives systems, ABB standard and ABB engineered.

ABB Standard Drives

Table 21 shows the ABB standard drives.

Table 21. ABB Standard Drives

Name	Application
ACS400	Standard drive
ACS600	Crane application
ACS600	Pump and fan application
ACS600	Standard application
ACS800	Crane application

Table 21. ABB Standard Drives (Continued)

Name	Application
ACS800	Pump and fan application
ACS800	Standard application
DCS400	Standard drive
DCS500	Standard drive

ABB Engineered Drives

Table 22 shows the ABB engineered drives.

Table 22. ABB Engineered Drives

Name	Application
ACS600	IGBT supply (ISU) application
ACS600	System application
ACS600AD	Asynchronous drive
ACS600C	Cycle converter drive
ACS600SD	Synchronous drive
ACS800	IGBT supply (ISU) application
ACS800	System application
ACS1000	Standard drive
DCS600	System application

Appendix B Performance and Capacity

This section presents performance and technical data for Control Software and Control Builder key functions, configuration, and items.

Late changes might affect performance and/or functionality. For information on late changes and restrictions on the use of the product, please refer to the Release Notes.

Memory and Execution Performance

Memory Size

Figure 7 shows the memory organization. The total physical memory less the executing firmware is called “Memory size” by the “SystemDiagnostics” function block. This amount of memory is sometimes also called the “heap”.

The memory usage is also displayed in the Control Builder Heap Utilization dialog which can be displayed for each controller. The available memory is called “Non-Used Heap” and the rest is called “Used Shared Heap”.

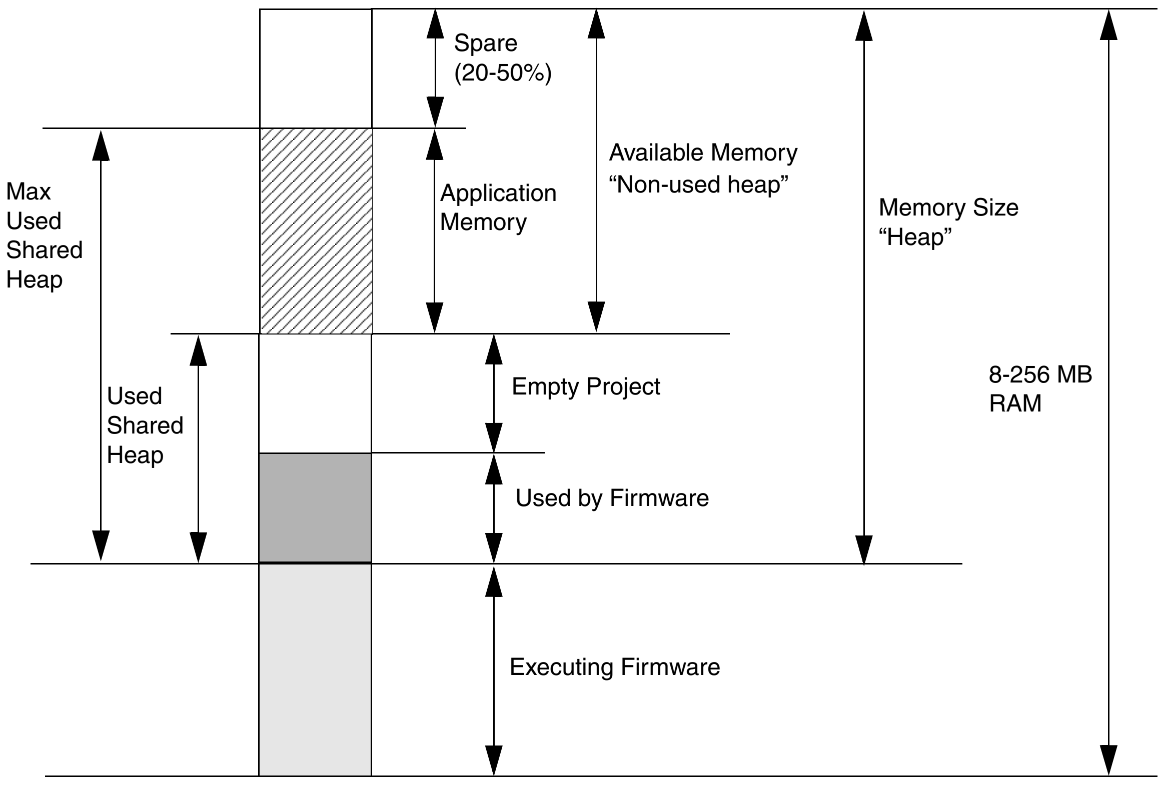


Figure 7. The Memory Organization

Available Memory

The amount of free memory in the controller decreases when the controller has started up, and an empty project has been downloaded from Control Builder M.

The remaining memory is what can be used for application code, and is hereafter referred as to “Available memory”.

The measurement results in Table 23 are made with IAC, but without any configured communication protocols and CEX units. Memory consumptions for used protocols and CEX units have to be added, according to Table 24.

Table 23. Available RAM Memory and Performance in AC 800M Controller (without Protocol Handlers)

Controller	Execution Performance Factor	Total RAM (kbytes)	Firmware and an Empty Project (kbytes)	Available Memory (kbytes)
PM851	0.5	8192	5909	2282
PM851A	0.5	12288	5916	6372
PM856	0.5	8192	5909	2282
PM856A	0.5	16384	5928	10456
PM860	1.0	8192	5909	2282
PM860A	1.0	16384	5927	10457
PM861	1.0	16384	9063	7320
PM861A	1.0	16384	9063	7320
PM864	1.5	32767	9104	23663
PM864A	1.5	32767	9104	23663
PM866	2.1	65535	14133	51402
PM891	4.5	268032	68798	199233

Table 24. Memory Consumptions of Protocols and CEX Units

Protocol/ CEX Unit	PM864		PM866		PM891	
	First Unit (kbytes)	Next Unit (kbytes)	First Unit (kbytes)	Next Unit (kbytes)	First Unit (kbytes)	Next Unit (kbytes)
MODBUS RTU	74	13	74	16	63	5
COMLI	69	16	68	19	57	3
S3964R	61	14	60	15	52	4
SerialLib	59	18	59	18	47	6
IAC	176		175		163	
UDP	36		35		35	
TCP	45		44		44	
CI853	4	4	4	3	4	4
CI854	233	30	234	29	150	12
CI855	96	11	96	11	88	3
CI856	96	10	95	12	89	2
CI857	177	13	178	12	169	4
CI858	59	19	60	18	47	9
CI865	126	73	126	86	125	74
CI867	164	36	164	36	163	34
CI868	197	63	197	64	127	4
CI869	183	64	183	63	124	3
CI871	192	25	191	26	118	16
CI872	232	70	232	69	173	10
CI873	223	102	223	100	164	42

Execution Performance

Cyclic CPU load is calculated as a percentage using the following formula.

Cyclic CPU load (%) = 100(Total execution time / Total interval time)*

Depending on the amount of code and requested task interval times, applications may demand up to 70% of CPU capacity (never more)[1]; the execution of IEC 61131-3 code is called *Cyclic Load*. Should an application require more than 70% of CPU capacity, the task scheduler automatically increases the task interval times to re-establish a 70% load.

Load balancing can be disabled (see the manual *Compact Control Builder Configuration, 3BSE040935*)*.*

It is important to consider CPU load if communication handling is vital to the application. Running at the maximum cyclic load will result in poor capacity and response times for peer-to-peer and OPC Server communication.

Communication handling has the lowest priority in a controller. It is therefore important to consider controller CPU load if the communication handling is vital to the application. Running close to 100% total load will result in poor capacity and response times for peer-to-peer and (OPC Server for AC 800M) communication. It is recommended that peak total load will be kept below 100%.

Among the communication protocols, the IAC MMS protocol will be the last to be affected if there is a communication data starvation.

CPU load is also influenced by other factors, such as Modulebus scan interval and the number of modules on Modulebus (AC 800M), or the scanning of ABB Drives.

The PM860/PM860A and PM861/PM861A processor units have the same internal design and the same performance when execution application program.

The PM851/PM851A, PM856/PM856A and PM860/PM860A processor units have the same internal design. They differ only in performance when executing an application program. The execution time in PM851/PM851A and PM856/PM856A is approximately two times the execution time in PM860/PM860A.

1. This is **not** true if load balancing is set to false. The controller will run until it is forced to stop.

The PM864/PM864A processor unit, in single configuration, has performance data which theoretically peaks at twice the performance compared to the PM860/PM860A. The useful sustained performance improvement is, however, a bit lower and dependent on the actual application program but can be expected to be 10 to 50% compared to PM860/PM860A. The difference in execution performance is dependent on how much CEX bus accesses, and how much communication is running in the controller (both communication running as CEX bus interfaces and communication running on the built in ports on the CPU i.e. ModuleBus Ethernet and RS-232). CEX bus access and communication decreases execution performance.

In redundant configuration the execution performance is lower than in single configuration (reduction is typically less than 10%). Switch over time from primary controller to backup controller, in redundant configuration, is less than 10 ms.

The PM866 processor unit has performance data which is approximately 1.4 times the performance of PM864/PM864A.

The PM891 processor unit has performance data which is approximately 2 times the performance of PM866.

Spare Memory Needed for Online Changes

As a general rule, an application should never exceed half the size of the available memory. The reason for this is the manner in which applications are updated online.

1. The modifications (the difference between the old and the updated application) are downloaded to the controller memory.
2. A new version of the application is created in controller memory, based on the old application and the modifications.
3. The controller switches from the old to the new application.
4. The old application is deleted.

This technique handles all updates in a controlled and efficient way. Free memory equal to the size of the largest application is required.

If an application comes close to this limit, it should be divided into two parts so that they can be updated separately.

One Application in the Controller

There must be spare memory in the available memory in order to be able to make on-line changes shown in the Figure 7. The amount of spare memory must be at least 20% of available memory, and may require up to 50%.

A minimum of 20% spare available memory may be sufficient, depending on a number of factors, such as the complexity of the application and the number of defined alarms.

The function block “SystemDiagnostics” reports used memory based on the memory size, not on the available memory, but the dialog “Heap Utilization” will show the available memory as “Non-Used Heap”.

The function block *SystemDiagnostics* also presents another figure: the “Maximum used memory”. This figure is presented in actual bytes, and as a percentage of the memory size. This figure is far more useful to look at when determining how close you are to being unable to make on-line changes. Several on-line changes must be made in order to catch the maximum memory need in the controller.

It is still possible to make on-line changes as long as the maximum used memory value is less than 100%.

More than One Application in the Controller

Less spare memory is needed when there is more than one application in the controller.

The on-line changes are done to one application at the time. This means that if changes are done to more than one application in the controller, these changes will not take effect in a synchronized way.

Example: One application requires 50% used memory and 70% maximum used memory. If you split this application into two equally smaller applications, it will still require 50% used memory, but only 60% maximum used memory, since the extra memory needed for the on-line changes will be half.

Comparing Memory Allocations Made with Different Versions

From the discussions above, you can see that the “used memory” value provided by the *SystemDiagnostics* function block cannot be used to compare different versions.

The amount of available memory in the controller varies between versions for a number of reasons, one being the number of functions implemented in the firmware.

Memory Consumption and Execution Times

Memory is reserved for each function block type defined. When another instance is created, the amount of memory reserved for the instance is very small in relation to the type. This means that the memory consumed by the type itself is of great importance.

The following tables show memory consumption and execution time for AC 800M PM864/PM866/PM891 controllers, for a number of common function blocks and control modules.

In the tables the *First Object* column shows the required memory for the object type and one function block or control module and *Next Object* column shows the required memory for every further function block or control module.

Table 25. AC 800M Memory Consumption for Function Blocks and Control Modules

Library	Object	PM864		PM866		PM891	
		First Object (kbyte)	Next Object (kbyte)	First Object (kbyte)	Next Object (kbyte)	First Object (kbyte)	Next Object (kbyte)
Function Blocks							
SignalLib	SignalInBool	23.2	3.7	23.5	4	23.8	4.2
SignalLib	SignalOutBool	20.7	2.8	28.33	4.27	26.18	4.24
SignalLib	SignalSimpleInReal	19.5	3.2	28.62	3.97	25.7	3.85
SignalLib	SignalInReal	55	9.7	62.39	12.42	61.1	10.54
SignalLib	SignalSimpleOut Real	16.5	2.8	22.3	3.46	22.69	3.57

Table 25. AC 800M Memory Consumption for Function Blocks and Control Modules (Continued)

Library	Object	PM864		PM866		PM891	
		First Object (kbyte)	Next Object (kbyte)	First Object (kbyte)	Next Object (kbyte)	First Object (kbyte)	Next Object (kbyte)
SignalLib	SignalOutReal	52.8	8.8	58.3	9.39	57.53	9.4
AlarmEventLib	AlarmCondBasic	10.9	1.2	22.7	1.6	12.24	1.59
AlarmEventLib	AlarmCond	11.8	2	15.39	1.96	13.27	1.99
ProcessObjectExtLib	Uni	56.6	8.3	68.5	8.8	50.34	8.35
ProcessObjectExtLib	Bi	56.6	12.4	49.51	5.53	47.21	5.56
ProcessObjectExtLib	MotorUni	57.9	10.4	50.24	6.66	47.96	6.6
ProcessObjectExtLib	MotorBi	67.7	14.4	54.9	7.6	52.68	7.65
ProcessObjectExtLib	ValveUni	47.6	7.6	42.2	4.2	39.98	4.05
ProcessObjectInsum Lib	McuExtended	116.5	28.1	126.6	28.3	121.3	19.42
ProcessObjectDrive Lib	ACStdDrive	88.5	16.1	86.8	16.95	91.95	16.88
ControlSimpleLib	PidSimpleReal	17.1	1.3	27.51	2	21.13	2
ControlBasicLib	PidLoop	63.4	5.7	64.18	5.8	62.42	5.83
ControlBasicLib	PidLoop3P	67.7	5.7	67.5	6.67	65.05	6.84
ControlBasicLib	PidCascadeLoop	76.5	11.5	75.89	12.78	74.33	12.78
ControlBasicLib	PidCascadeLoop3P	80.8	11.5	79.9	12.42	77.53	12.89
SignalBasicLib	SignalBasicBool	13.5	0.5	22.2	1.17	17.01	1.1
SignalBasicLib	SignalBasicInBool	14.5	0.1	20.43	1.25	18.07	1.2
SignalBasicLib	SignalBasicInReal	19.8	0.8	25.22	1.92	22.59	1.82
SignalBasicLib	SignalBasicOutBool	15.4	0.7	20.78	1.27	18.35	1.27
SignalBasicLib	SignalBasicOutReal	16.6	1	21.81	1.5	19.28	1.56

Table 25. AC 800M Memory Consumption for Function Blocks and Control Modules (Continued)

Library	Object	PM864		PM866		PM891	
		First Object (kbyte)	Next Object (kbyte)	First Object (kbyte)	Next Object (kbyte)	First Object (kbyte)	Next Object (kbyte)
SignalBasicLib	SignalBasicReal	15.7	0.2	20.89	1.33	18.39	1.38
SupervisionBasicLib	SDBool	34.1	4.8	41.2	5.26	36.89	5.35
SupervisionBasicLib	SDInBool	35.3	4.4	39.99	5.85	37.68	5.83
SupervisionBasicLib	SDInReal	53.6	12.1	58.43	12.85	55.64	12.81
SupervisionBasicLib	SDLevel	33.7	5.4	38.82	5.6	36.56	5.58
SupervisionBasicLib	SDOutBool	38.3	5.9	42.93	7.17	40.55	7.35
SupervisionBasicLib	SDReal	48.7	12	53.79	12.19	51.17	12.25
SupervisionBasicLib	SDValve	41	5.6	45.49	6.4	43.24	6.3
SupervisionBasicLib	StatusRead	23.7	3.7	28.93	3.71	26.58	3.69

Table 25. AC 800M Memory Consumption for Function Blocks and Control Modules (Continued)

Library	Object	PM864		PM866		PM891	
		First Object (kbyte)	Next Object (kbyte)	First Object (kbyte)	Next Object (kbyte)	First Object (kbyte)	Next Object (kbyte)
Control Modules							
SignalLib	SignalInBoolM	25.6	4.3	26.5	4	26.6	4.3
SignalLib	SignalOutBoolM	26.86	3.7	32.75	4.17	29.56	4.14
SignalLib	SignalInRealM	66.78	9.8	72.7	10	68.92	10
SignalLib	SignalOutRealM	61.14	9.7	67.17	10.13	65.18	10.07
AlarmEventLib	AlarmCondBasicM	6.9	1	12.9	1.01	12.21	0.91
AlarmEventLib	AlarmCondM	7.6	1.6	13.69	1.1	12.9	1.07
ProcessObjectExtLib	UniM	55.85	8.9	62.35	8.4	61.68	10.37
ProcessObjectExtLib	BiM	64.48	12.8	70.7	12.6	71.46	12.56
ProcessObjectExtLib	MotorUniM	77.1	11.4	69.9	10.8	76.1	6.54
ProcessObjectExtLib	MotorBiM	87.5	14.1	80.44	14.5	81.3	14.3
ProcessObjectExtLib	ValveUniM	67.9	8.3	61.05	8.6	61.72	8.7
ProcessObjectInsum Lib	McuExtendedM	127.8	27.7	120.9	27.7	122.09	27.82
ProcessObjectDrive Lib	ACStdDriveM	100.8	16.2	93.7	42.4	93.86	16.61
ControlStandardLib	AnalogInCC	37.2	3.7	30.4	3.6	30.07	3.67
ControlStandardLib	AnalogOutCC	34.2	3.8	27.25	3.8	27.19	3.77
ControlStandardLib	Level2CC	22.89	4.7	30.5	17	31	4.62
ControlStandardLib	Level4CC	44.5	7.1	37.9	6.4	38.51	6.4
ControlStandardLib	Level6CC	53.1	8.2	45.8	8.2	46.31	23.7
ControlStandardLib	ThreePosCC	22.53	4.3	30.4	4.19	29.5	4.35
ControlStandardLib	PidSimpleCC	29.5	2.4	22.1	2.5	22.23	2.35
ControlStandardLib	PidCC	112.2	15	109.6	14.9	103.1	14.9

Table 25. AC 800M Memory Consumption for Function Blocks and Control Modules (Continued)

Library	Object	PM864		PM866		PM891	
		First Object (kbyte)	Next Object (kbyte)	First Object (kbyte)	Next Object (kbyte)	First Object (kbyte)	Next Object (kbyte)
ControlAdvancedLib	PidAdvancedCC	225.3	23.2	224.7	24.4	206.6	25.23
ControlSolutionLib	SingleLoop	199.1	31.1	196.4	33.12	186.4	33.07
ControlSolutionLib	CascadeLoop	232.4	58.9	225.3	58.9	215.03	58.9
ControlSolutionLib	OverrideLoop	310.2	105.2	295.8	109.5	284.5	109.5
ControlSolutionLib	FeedForwardLoop	208.8	42.6	221.6	45.9	211.1	45.8
ControlSolutionLib	MidRangeLoop	204.3	44	220.5	46.5	210.1	46.44

Table 26. AC 800M Execution Time for Function Blocks and Control Modules

Library	Object	PM864 (μs)	PM866 (μs)	PM891 (μs)
Function Blocks				
AlarmEventLib	AlarmCondBasic	20	15	9
AlarmEventLib	AlarmCond	35	25	14
AlarmEventLib	ProcessObjectAE	22	16	9
AlarmEventLib	SimpleEventDetector	32	23	14
AlarmEventLib	SignalAE	39	28	16
BasicLib	CTD	8	6	3
BasicLib	CTU	8	6	3
BasicLib	CTUD	11	8	4
BasicLib	ErrorHandler	9	7	3
BasicLib	F_Trig	5	4	2
BasicLib	ForcedSignals	30	20	12

Table 26. AC 800M Execution Time for Function Blocks and Control Modules (Continued)

Library	Object	PM864 (μs)	PM866 (μs)	PM891 (μs)
BasicLib	PulseGenerator	9	6	3
BasicLib	R_Trig	5	4	2
BasicLib	RS	6	4	2
BasicLib	SR	5	4	2
BasicLib	SystemDiagnostics	622	443	185
BasicLib	TimerD	46	34	14
BasicLib	TimerU	35	25	9
BasicLib	TOf	9	6	3
BasicLib	TOn	8	6	3
BasicLib	TP	8	6	3
ControlBasicLib	PidLoop	284	203	102
ControlBasicLib	PidLoop3P	311	245	99
ControlBasicLib	PidCascadeLoop	580	378	155
ControlBasicLib	PidCascadeLoop3P	539	371	145
ControlSimplelib	PidSimpleReal	61	44	16
ProcessObjBasiclib	BiSimple	193	140	78
ProcessObjBasiclib	UniSimple	113	83	50
ProcessObjDriveLib	ACStdDrive	570	422	216
ProcessObjExtLib	Bi	313	229	135
ProcessObjExtLib	MotorBi	405	285	181
ProcessObjExtLib	MotorUni	306	242	139
ProcessObjExtLib	Uni	182	144	82
ProcessObjExtLib	ValveUni	197	131	79

Table 26. AC 800M Execution Time for Function Blocks and Control Modules (Continued)

Library	Object	PM864 (μs)	PM866 (μs)	PM891 (μs)
ProcessObjInsumLib	McuExtended	503	365	224
SignalBasicLib	SignalBasicBool	6	5	3
SignalBasicLib	SignalBasicInBool	7	5	3
SignalBasicLib	SignalBasicInReal	50	36	18
SignalBasicLib	SignalBasicOutBool	7	6	3
SignalBasicLib	SignalBasicOutReal	13	10	4
SignalBasicLib	SignalBasicReal	16	12	5
SignalLib	SignalBool	57	41	24
SignalLib	SignalInBool	50	34	18
SignalLib	SignalInReal	206	143	85
SignalLib	SignalOutBool	48	34	21
SignalLib	SignalOutReal	127	89	51
SignalLib	SignalReal	152	119	60
SignalLib	SignalSimpleInReal	86	64	33
SignalLib	SignalSimpleOut Real	43	30	17
SupervisionBasicLib	SDBool	126	85	48
SupervisionBasicLib	SDInBool	94	67	36
SupervisionBasicLib	SDInReal	261	186	92
SupervisionBasicLib	SDLevel	78	57	31
SupervisionBasicLib	SDOutBool	122	88	45
SupervisionBasicLib	SDReal	249	169	80
SupervisionBasicLib	SDValve	165	117	68
SupervisionBasicLib	StatusRead	35	26	14

Table 26. AC 800M Execution Time for Function Blocks and Control Modules (Continued)

Library	Object	PM864 (μs)	PM866 (μs)	PM891 (μs)
Control Modules				
AlarmEventLib	AlarmCondBasicM	30	21	13
AlarmEventLib	AlarmCondM	24	16	9
BasicLib	CVAckISP	45	32	18
BasicLib	ErrorHandlerM	17	11	5
BasicLib	ForcedSignalsM	35	26	15
ControlAdvancedLib	PidAdvancedCC	885	634	288
ControlSolutionLib	CascadeLoop	1779	1279	658
ControlSolutionLib	FeedforwardLoop	1484	1076	501
ControlSolutionLib	MidrangeLoop	1367	978	472
ControlSolutionLib	OverrideLoop	3405	2398	1237
ControlSolutionLib	SingleLoop	1063	767	383
ControlSolutionLib	AnalogInCC	107	79	43
ControlSolutionLib	AnalogOutCC	88	62	32
ControlSolutionLib	Level2CC	63	45	25
ControlSolutionLib	Level4CC	95	67	39
ControlSolutionLib	Level6CC	120	86	52
ControlSolutionLib	PidCC	489	359	173
ControlSolutionLib	PidSimpleCC	106	69	32
ControlSolutionLib	ThreePosCC	153	110	52
ProcessObjBasicLib	BiSimpleM	183	142	77
ProcessObjBasicLib	UniSimpleM	121	88	51
ProcessObjDriveLib	ACStdDriveM	598	417	222

Table 26. AC 800M Execution Time for Function Blocks and Control Modules (Continued)

Library	Object	PM864 (μs)	PM866 (μs)	PM891 (μs)
ProcessObjExtLib	BiM	346	249	138
ProcessObjExtLib	MotorBiM	403	295	161
ProcessObjExtLib	MotorUniM	315	223	137
ProcessObjExtLib	UniM	215	152	94
ProcessObjExtLib	ValveUniM	188	157	91
ProcessObjInsumLib	McuExtendedM	535	370	227
SignalLib	SDLevelAnd4	28	19	11
SignalLib	SDLevelBranch4	21	14	7
SignalLib	SDLevelM	102	74	44
SignalLib	SDLevelOr4	23	17	8
SignalLib	SignalBoolCalcInM	71	50	29
SignalLib	SignalBoolCalcOutM	92	65	36
SignalLib	SignalInBoolM	63	43	23
SignalLib	SignalInRealM	245	177	100
SignalLib	SignalOutBoolM	66	53	31
SignalLib	SignalOutRealM	219	145	86
SignalLib	SignalRealCalcInM	222	161	91
SignalLib	SignalRealCalcOutM	234	183	105
SignalLib	Vote1oo1Q	228	153	77
SignalLib	VoteBranch4	41	29	14
SignalLib	VotedAnd4	37	27	14
SignalLib	VotedBranch4	25	18	8

Table 26. AC 800M Execution Time for Function Blocks and Control Modules (Continued)

Library	Object	PM864 (µs)	PM866 (µs)	PM891 (µs)
SignalLib	VoteXoo2D	259	184	83
SignalLib	VoteXoo3Q	285	202	88
SignalLib	VoteXoo8	299	216	105

Table 27. Execution Time for a Number of Standard Operations and Function Calls

Operation/Function	Data Type	PM864 (µs)	PM866 (µs)	PM891 (µs)
a:= b or c	bool	0.7	0.5	0.3
a:= b and c	bool	0.7	0.5	0.2
a:= b xor c	bool	0.7	0.5	0.2
a := b + c	dint	0.7	0.5	0.2
a := b + c	real	1.9	1.4	0.2
a := b - c	dint	0.7	0.5	0.2
a := b - c	real	1.7	1.2	0.2
a := b * c	dint	0.7	0.5	0.3
a := b * c	real	1.8	1.3	0.2
a := b / c	dint	1.0	0.7	0.5
a := b / c	real	4.0	2.9	0.3
a:= b <> c	dint	0.8	0.6	0.3
a:= b <> c	real	1.8	1.3	0.3
a := b	string[140]	8.4	6.3	2.7
a := b + c	string[10]	17.5	11.9	3.4
a := b + c	string[40]	17.6	13.0	2.6

Table 27. Execution Time for a Number of Standard Operations and Function Calls

Operation/Function	Data Type	PM864 (µs)	PM866 (µs)	PM891 (µs)
a := b + c	string[140]	27.8	20.1	5.5
a := AddSuffix (b + c)	string[10]	4.0	3.3	1.4
a := AddSuffix (b + c)	string[40]	7.6	5.3	2.4
a := AddSuffix (b + c)	string[140]	16.9	11.6	4.2
a := real_to_dint(b)	dint	15.6	11.2	0.9
a := dint_to_real(b)	real	1.8	1.3	0.4
a := real_to_time(b)	time	22.0	15.6	1.5
a := time_to_real(b)	real	6.5	4.5	0.8

Hardware and I/O

Modulebus Response Time and Load

Modulebus scanning has a considerable influence on CPU load, since I/O copying on Modulebus is handled by the controller CPU.
The scan time increases as modules are added, and at a certain point Modulebus scanning will start to seriously influence CPU load.

The Modulebus scan cycle time can be set in Control Builder. The cycle time must be set to suit the module requiring the shortest scan interval. A solution to this problem is to connect I/O variables requiring shorter scan intervals via the CI854A PROFIBUS adapter.

In AC 800M, Modulebus scanning has the highest priority. The cyclic load presented for IEC 61131-3 applications includes extra load caused by Modulebus interrupts.

Calculation of Scan Time on the Modulebus and CPU Load

The following definitions are used in the calculations:

1. Amount of modules:

- n_1 = amount of Drives and DP, DI, DO, AI and AO modules (except AI880A, DI880 and DO880)
- n_2 = amount of AI880A, DI880 and DO880 modules

For the modules below, the following number of modules should be accounted:

- AO845 (redundant) = 2
- DO840 (redundant) = 2
- DO880 (redundant) = 2
- DP820 = 4
- DP840 (single) = 8
- DP840 (redundant) = 9
- ABB Engineered Drives = 3
- ABB Standard Drives = 2

For other redundant modules, only one should be accounted.

2. Scan time for different modules:

   t_1 = 0.5 ms (scan time for n_1)

   t_2 = 1.3 ms (scan time for n_2)

3. Load caused by n_1 module types:

   L_1 = 20%

4. Load caused by n_2 module types:

   L_2 = 8%

For PM866, the values will be approximately half, that is, L_1 = 10% and L_2 = 4%.

For PM891, the values will be approximately one third, that is, L_1 = 7% and L_2 = 3%.

Calculation of Fastest Possible Scan Time

The fastest possible scan time is $n_1 * t_1 + n_2 * t_2$.

Example:
It can never take less than 10 * 0.5 = 5.0 ms to scan 10 I/O modules.

Calculation of the Modulebus CPU Load

The Modulebus scanning causes the following CPU load if the chosen scan cycle time is less or equal to the fastest possible scan time:

$$Load_{(fastest)} = (n_1 / (n_1 + n_2) * L_1 + (n_2 / (n_1 + n_2) * L_2)$$

The following CPU load is caused for other scan cycle times:

$$Load_{(chosen)} = \text{Fastest Possible Scan Time / Chosen Scan time} * Load_{(fastest)}$$

The formulas are valid for all AC 800M processor unit types.

Example Scan Time and CPU Load

Assume that following units are used:

1 AI810: 0.5*1 = 0.5 ms
1 redundant DO880: 1.3*2= 2.4 ms
1 redundant DP840: 0.5*9 = 4.5 ms

This gives a scan cycle time of 8 ms (resolution = 1 ms).

CPU Load will be: (10/12)*20 + (2/12)*8 = 18%

Updating Rate of Data to an Application

The rate in milliseconds at which all channels of an I/O module are updated in the controller to be used in the IEC 61131-3 application, as a function of the scan time in milliseconds is as follows:

- For AI, AO and AI843 (except AI880A and other temperature measuring I/O than AI843) the updating time is equal to number of channels divided by two multiplied by the scan time.
- For temperature measuring I/O (except for AI843) the updating time is equal to number of channels multiplied by the scan time.
- For Standard Drives the updating time is equal to scan time.
- For Engineered Drives the updating time is equal to scan time multiplied by 12.
- For DI, DO, DP the updating time is equal to scan time.

ModuleBus Scanning of ABB Drives

Scanning of ABB Drives on Modulebus also influences CPU load.

Modulebus Scanning of ABB Engineered Drives (AC 800M)

Scanning of an engineered Drive is distributed over 3 * 12 scan cycles. Three channels (DDS pairs) are scanned in each scan cycle. The first two are always channels 1 and 2 (i.e. DDS pairs 10/11 and 12/13); the third will be different for each scan cycle.

Table 28. Scan cycles for ABB Engineered Drives DDS Pair 3

Scan Cycle	DDS Pair 3
1, 5, 9	14/15
2, 6, 10	16/17
3, 7 11	18/19
4	20/21
8	22/23
12	24/25

To scan the three DDS pairs each cycle takes 3 * 0.5 = 1.5 ms. It is not possible to have a scan interval less then 2 ms (=PA controller) / 5 ms =HI controller) for the Modulebus scanner. Thus, for one drive the scan time will be 2 ms.

Example

For four drives, the scan time will be 1.5 ms * 4 = 6.0 ms for the DDS pairs 10/11 and 12/13, and the scan time for the remaining of the DDS pairs will be 1.5 ms * 4 * 12 = 72.0 ms.

ModuleBus Scanning of ABB Standard Drives (AC 800M)

For ABB Standard Drives, all data sets (DDS 1/2 and DDS 3/4) are scanned in each scan cycle. It takes 2 * 0.5 = 1.0 ms to scan a single Standard Drive.

Example

For four ABB Standard Drives the scan time will be 1.0 ms * 4 = 4.0 ms.

Dynamic Data Exchange S800 I/O Connected via CI854A

The transportation of dynamic data between PROFIBUS-DP/DPV1 master and the S800 I/O modules shown in the Figure 8.

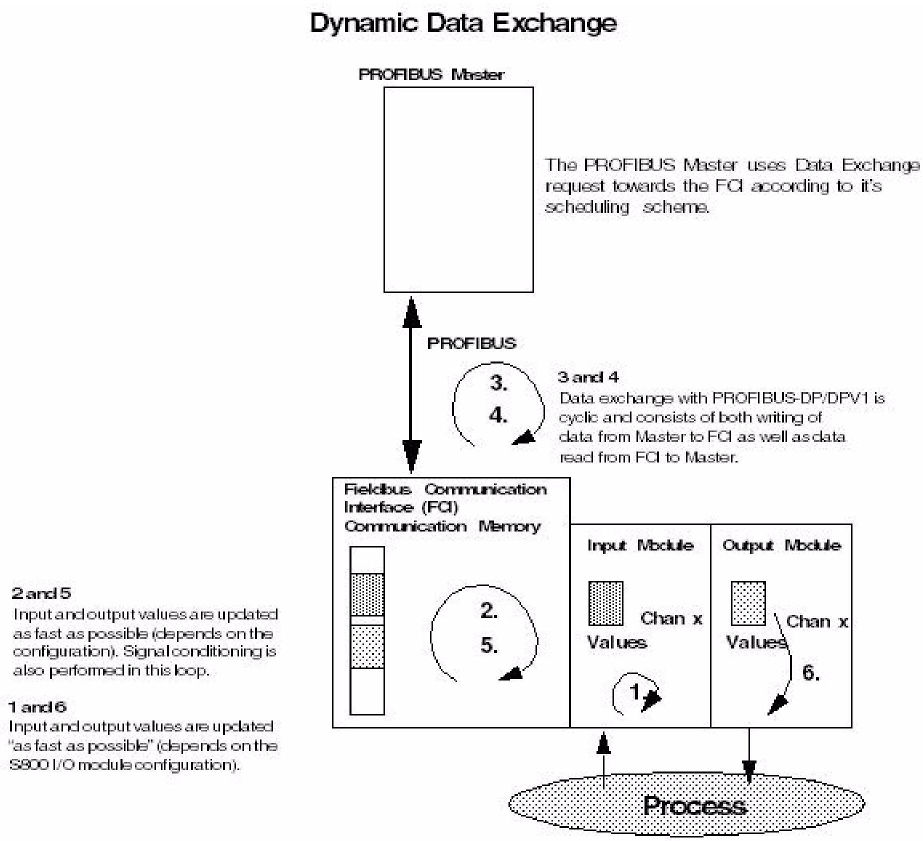


Figure 8. Transportation of dynamic data between PROFIBUS-DP/DPV1 master and S800 I/O modules.

The transportation of dynamic data between PROFIBUS-DP/DPV1 and the ModuleBus is the main task for the Field Communication Interface FCI. The FCI has a dedicated memory area where it sends the output values and reads the input values.

The CPU in the FCI performs the rest of the data transportation. It reads output values from the memory and writes to the I/O Modules via the ModuleBus and vice versa.

Data Scanning Principles

The data transfer between PROFIBUS-DP/DPV1 and the ModuleBus (3 and 4 in the figure) is not synchronized. Read and write operations are performed from and to a dual port memory in the FCI.

The ModuleBus data is scanned (read or written) (2 and 5 in the figure) cyclically, depending on the I/O module configuration. On one scan all digital modules, 1/4 of the analog modules and 1/10 of the slow analog modules (modules for temperature measurement) are scanned. It takes 4 scans to read all analog modules and 10 scans to read all slow analog modules.

At en typical configuration with 3 AI, 2 AO, 3 DI and 2 DO modules the data scan time will be 18 ms.

For calculation of the ModuleBus data scanning in the FCI, see S800 I/O User's Guide Fieldbus Communication Interface PROFIBUS-DP/DPV1 Section 3 Configuration and Chapter Data Scanning.

The internal data scanning (1 and 6 in the figure) on the I/O modules is not synchronized with the ModuleBus data scanning.

Typical data scanning on S800 I/O modules (filter times not included):

Digital modules 1ms.

Analog modules 10ms.

Slow analog modules 1s.

Data scanning on S800 I/O modules see, S800 User's Guide Module and termination Units Appendix A Specifications.

Calculation of Signal Delay

Signal delay from process to controller and vice versa can be calculated according to following:

Signal delay = Controller scan time + Profibus scan time + FCI scan time + Module scan time + Filter time.

For example:

Signal delay digital signal = Controller scan time + Profibus scan time + FCI scan time + Module scan time + Filter time.

Signal delay analog signal = Controller scan time + Profibus scan time + 4 * FCI scan time + Module scan time + Filter time.

Signal delay slow analog signal = Controller scan time + Profibus scan time + 10 * FCI scan time + Module scan time + Filter time.

S100 I/O Response Time and Load

The response time is the time it takes for a signal to go from the input terminals on a S100 I/O board to the double port memory on the CI856 unit or vice versa for output signals. The delay caused by the filtering of the input signals is not included.

The S100 I/O response time is the sum of the following times:
`Conversion Time + Internal Scan Time + Scan Interval CI856`

- `Conversion Time` = 0.1 ms for DSAI 130/130A. For other I/O boards it can be ignored.
- `Internal Scan Time` = The internal scan time on DSAX 110 and DSAX 110A is 20 ms for input signals and 8 ms for output signals. For other I/O boards it is 0 ms.
- `Scan Interval CI856` = The scan interval on the CI856 is set for each I/O board or I/O channel and is determined by "scan interval" or "update interval" in the I/O hardware editor, under settings tab for selected I/O unit.

Calculation of CI856 CPU Load

For each I/O board the load on CI856 is calculated as:

```
BoardLoad = (BaseLoad + N*ChannelLoad)/CycleTime
```

- `BoardLoad` = the CPU load on the CI856 caused by the board (unit = percent).
- `BaseLoad` = the base load to handle the board, see Table 29 below.
- `ChannelLoad` = the additional load for each I/O channel used on the board, see Table 29 below.
- `N` = number of used I/O channels on the board.
- `CycleTime` = the cycle time or update interval set for the board or I/O channel (unit = 0.1 ms).

Table 29. BaseLoad and ChannelLoad of S100 I/O

Board	BaseLoad	ChannelLoad
DSAI 130/130A	20	125
DSAI 130D, DSAI 133/133A	20	40
DSAO	7	3.5
DSDI	35	0
DSDO	45	0
DSDP 010	12	22
DSDP 170 Function Mode = Pulse25	25	30
DSDP 170 Function Mode = Frequency	25	30
DSDP 170 Function Mode = Pulse + Frequency	25	61
DSDP 170 Function Mode = Pulse light2513	25	13

To allow scan task overhead and event treatment, the total load from all I/O boards should not exceed 80%.

Drivebus Communication with CI858 Unit

Data transfer on Drivebus is managed through datasets pairs. For standard drives 2 dataset pairs can be used and for Engineered drives up to 8 data set pairs can be defined.

Dataset Priority

Datasets can be given two priorities, High and Normal. High priority datasets are associated with the high priority execution table which is scanned every 2 ms. Normal priority datasets are associated with the normal priority execution table. This table is built-up of several channels (slots). The number of channels depends on the maximum number of normal priority Datasets defined in any drives unit on the bus. Every 2 ms one of the normal priority table channels is scanned.

Example Dataset Priority

If the maximum number of low priority datasets defined in a drives unit on the bus is 6, the normal priority execution table contains 6 channels, each channel is scanned every 12th millisecond (`2ms * 6=12ms`).

Dataset Pairs

The transfer times for dataset pairs, for example, DS10/DS11, includes transferring the message from the drive to the AC 800M (DS10) and the response message, including return value, back to the drives unit (DS11).

Drivebus (CI858) Response Time and Load

When calculating the response times between drives units and AC 800M on Drivebus the following has to be considered:

- Application task interval time in the host system, that is PM86x.
- Dataset execution queue and communication handler in the CI858,
- Bus transfer time, including data handling time in the communication ASICs on the CI858 and in the drives units.
- Drives unit application program.

Drivebus Response Time Formula

#DS_Channels: Max number of normal priority datasets in one drives unit on the bus.

AC 800M Application Program

Application program: Task interval time

High Priority Datasets

High priority dataset execution queue and communication handler: 2 ms

Drivebus transfer time: 1 ms

Inverter system application program:

DS10/11: 2 ms
DS12/13: 4 ms
(Other DS: 10 - 500 ms)

Normal Prio Datasets

Normal Prio dataset execution queue and communication handler:
`2 * #DS_Channels`

Drivebus transfer time: 1 ms

Inverter system application program:

DS10/11: 2 ms
DS12/13: 4 ms
Other DS: 10 - 500 ms

The response time on Drivebus consists of the sum of the following:

`TaskInterval + DataSet + DrivebusTransfTime + ApplTime`

- `TaskInterval` = Application task interval
- `DataSet` = DataSet Execution queue and communication handler
- `DrivebusTransfTime` = Drivebus transfer time
- `ApplTime` = Inverter system application time

Example

Consider a Drivebus containing five drive units. Each drives unit is using one high priority dataset pair (DS10/DS11). One of the drives units is using five normal priority dataset pairs DS12/DS13 to DS20/DS21. The other drives are using four normal priority dataset pairs DS12/DS13 to DS18/DS19. In the drives units the application program is using an update time of 100 ms for the normal priority datasets.

In the AC 800M the high priority datasets are attached to a high priority application task using a task interval time of 10 ms. The normal priority datasets are attached to a normal priority task using a task interval time of 250 ms.

Table 30. Response Times for each Dataset

<table>
<tr><th>Dataset</th><th>Application Task Interval</th><th>DataSet Execution Queue and Comm. Handler</th><th>Drivebus Transfer Time</th><th>Inverter System Application Time</th><th>Response Time (ms)</th></tr>
<tr><td>DS10/DS11</td><td>10</td><td>2</td><td>1</td><td>2</td><td>15</td></tr>
<tr><td>DS12/DS13</td><td>250</td><td>2*5</td><td>1</td><td>4</td><td>265</td></tr>
<tr><td>DS14/DS15</td><td rowspan="5">250</td><td rowspan="5">2*5</td><td rowspan="5">1</td><td rowspan="5">100</td><td rowspan="5">361</td></tr>
<tr><td>DS16/DS17</td></tr>
<tr><td>DS16/DS17</td></tr>
<tr><td>DS18/DS19</td></tr>
<tr><td>DS20/DS21</td></tr>
</table>

PROFIBUS DP Limitations and Performance

For PROFIBUS DP there are some limitations and performance to take into consideration.

Limitations

- CI854A can only act as master.
- The network can have a maximum of 126 nodes. A maximum of 124 slaves can be connected to a CI854A since the node addresses 0 and 1 are reserved for CI854A.
- S800 I/O connected to CI840 and/or S900 I/O connected to CI920 supports cable redundancy together with slave redundancy.
- If the PROFIBUS master unit, CI854A, loses contact with a slave unit, for example due to a disconnected cable, input values are set according to ISP configuration. If the I/O unit does not support ISP, all input values will freeze.
- Reset of PROFIBUS DP master, CI854A, and the complete PROFIBUS is done if one of the following bus parameter settings are changed: Node address of CI854A, baud rate or highest station address (HSA). A change of the other bus parameters does not affect the running communication.
- If the CI854A is running with 12 Mbit/s, then in total 4000 bytes input and output data for the cyclic communication are allowed to be configured. For lower Baudrate than 12 Mbit/s there is no limitation.

S900 (CI920) and S800 (CI840 and CI801) support configuration change (changing the parameters) without disrupting the cyclic data communication.

Performance

The cycle time on PROFIBUS depends on the baud rate, the summary of I/O data and the slave timing parameter. The fastest cycle time is about 1 ms with a baud rate of 12 Mbit/s and only one slave device. The typical cycle time is about 10-20 ms with 1,5 Mbit/s and some slave devices.

CI854A slave devices can have node addresses in the range 2-125 (the node addresses 0 and 1 are reserved for the CI854A). The baud rate can be configured to be in the range of 9,6 kbit/s - 12 Mbit/s. There is a maximum length of I/O data at 4000 bytes of input and output data in total when using 12 Mbit/s. For slower baud rate, up to 1,5 Mbit/s, there is no limitation of the length of the I/O data.

PROFINET IO Limitations and Performance

The following limitations apply for PROFINET IO configurations with CI871 in AC 800M.

- Up to 12 CI871 per AC 800M controller.
- Up to 126 PROFINET IO devices per CI871.
- Up to 512 modules per device.
- One IOCR for each direction (Input and Output) per device,
each IOCR up to 1440 bytes of I/O data.
- Update time down to 1 ms (only if CI871 has only one device configured).
- For CPU-load calculation of CI871, the Ethernet frames for inputs and outputs need to be calculated. CI871 can handle as a maximum one frame per ms in each direction.
Example 1: Update times for all devices is configured to 32 ms (default), then up to 32 devices can be connected to CI871.
Example 2: Update times for all devices is configured to 8 ms, then up to 8 devices can be connected to CI871.

The limitation for the CPU load of CI871 is checked by the system during download. If the system detects that there is a CPU overload, then it is indicated in the Compilation Summary window and the download is blocked. The CI871 may not function properly when there is an overload. The user can check the CPU load before and after download by use of the Web Interface. The limit for the CPU load is 100%. Up to that value the CI871 works stable without any problems or restrictions.

IEC 61850

The IEC 61850-Ed1 specification for Substation Automation System (SAS) defines communication between Intelligent Electronic Devices (IED) in the substation and other related equipment.

The IEC 61850 standard itself defines the superset of what an IEC 61850 compliant implementation might contain.

CI868 is modeled as an IED with one Access point and supports communicating with other IEDs through IEC 61850 GOOSE as well as IEC 61850 MMS client protocol.

The CI868 IEC 61850 Hardware Library 3.x available in Control Builder, supports CI868 to communicate over IEC 61850 GOOSE and MMS client protocols.

This section describes the configuration and performance limits for CI868 module.

CI868 Configuration Limit for AC 800M

Table 31. CI868 Configuration Limit for AC 800M

Description	Limit	Remarks
Maximum number of CI868 modules connected per Non-Redundant AC 800M controller.	12	CI868 Non-redundant units
Maximum number of CI868 modules connected per Redundant AC 800M controller.	6	CI868 Application-redundant units

CI868 Performance for GOOSE Protocol Usage

The following are the configuration and performance parameters related to CI868 module configured for GOOSE communication.

It is recommended to follow the mentioned limits to have CI868 CPU load levels less than 85% for satisfactory performance of CI868 module for GOOSE communication.

Following are the recommendations to be followed while engineering the Substation Configuration Description (SCD) file.

Table 32. Performance Parameters of CI868 Configured for GOOSE Protocol

Description	Limit	Remarks
Control Builder Configuration Data Per CI868 Module		
Maximum number of IEDs connected	80	
Maximum number of GOOSE Datasets	150	Subscribed Datasets count per CI868 monitored by the IEC 61850 Wizard.
Maximum number of Receive Blocks per IED	254	Each Receive Block constitutes 5 Data Objects. Each Data Object constitutes the Data Attributes. **For Example:** stVal / mag.f , q, t
Maximum number of LDs	10	
Maximum number of LNs per LD	253	
GOOSE Performance Data Per CI868 Module		
Maximum number of Static Data Objects (signals) configured	800	SCD file should not be configured for more than: Maximum 800 Data Objects (signals) from 80 IEDs 50 IEDs for 800xA SV5.0 80 IEDs for 800xA SV5.1 Analog, Integer or Boolean type of Data objects.

Table 32. Performance Parameters of CI868 Configured for GOOSE Protocol

Description	Limit	Remarks
Maximum number of Data Objects per Dataset	10	Analog, Integer or Boolean type of Data objects. This number should not be increased, otherwise the CI868 board could be overloaded. For more data signals to CI868 from same IED, use additional Datasets.
Maximum number of changing Data Objects Received	160 / sec	Analog, Integer or Boolean type of Data objects. Subject to SCD file configuration. Refer CI868 GOOSE Performance Graph.
Maximum number of changing Data Objects Sent	10 / sec	Analog, Integer or Boolean type of Data objects. Subject to SCD file configuration. Refer CI868 GOOSE Performance Graph.

CI868 GOOSE Performance Graph

The GOOSE performance of CI868 module is a function of the following parameters:

CI868 Receive

- Number of Static signals configured in scd-file for CI868 Receive.
  This is the sum of all Data Objects (DO) signals subscribed to CI868 in all Datasets in all IEDs.
- Number of changing signals out of static signals for CI868 Receive.
  This is the sum of all Data Objects signals subscribed to CI868 changing at a given moment across all IEDs.

CI868 Send

- Number of Static Signals configured in scd-file from CI868 to other IEDs.
  This is the sum of all Data Objects signals in all Datasets from CI868 to other IEDs.
- Number of changing send signals out of static signals from CI868 to other IEDs.
  This is the sum of all Data Objects signals changing at a given moment from CI868.

Figure 9 provides the CI868 GOOSE performance for different configurations of Static and Changing Receive signals for operating within optimal load of CI868 CPU (85%).

This chart is applicable for CI868 configured for GOOSE communication.

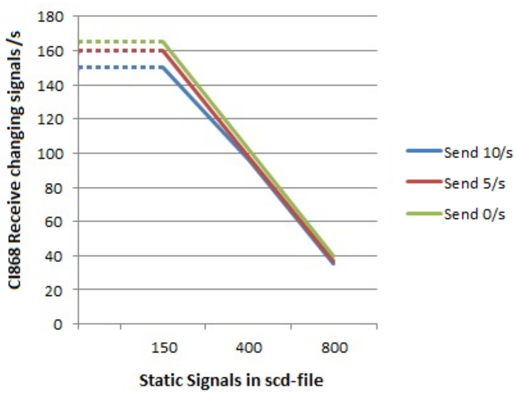


Figure 9. CI865 GOOSE Performance Chart

CI868 Performance for MMS Client Protocol Usage

The following are the configuration and performance parameters related to CI868 module configured for MMS communication.

It is recommended to follow the mentioned limits to have CI868 CPU load levels less than 85% for satisfactory performance of CI868 module for MMS communication.

Following are the recommendations to be followed while engineering the SCD file.

Table 33. Performance Parameters of CI868 Configured for MMS Client

Description	Limit	Remarks
Control Builder Configuration Data Per CI868 Module		
Maximum number of IEDs connected	20	
Maximum number of MMS Datasets	150	Subscribed Datasets count per CI868 monitored by the IEC 61850 Wizard.

Table 33. Performance Parameters of CI868 Configured for MMS Client

Description	Limit	Remarks
Maximum number of LDs under MyIED	10	
Maximum number of LDs under other IEDs	50	Starting from Position 200 under other IED in Control Builder hardware tree.
Maximum number of LNs per LD	253	
MMS Performance Data Per CI868 Module		
Maximum number of Static Data Objects (signals) configured	1000	SCD file should not be configured for more than: Maximum 1000 Data Objects (signals) from 20 IEDs 20 IEDs for 800xA SV5.1 FP480 Analog, Integer or Boolean type of Data objects. When creating scd file, high CI868 load for MMS signals can be avoided by grouping the frequently changing signals in the same dataset. **For Example**: Measurement signals must be grouped in one dataset, Status signals must be grouped in another dataset.
Maximum number of Data Objects per Dataset	10	Analog, Integer or Boolean type of Data objects. This number should not be increased, otherwise the CI868 board could be overloaded. For more data signals to CI868 from same IED, use additional Datasets.
Maximum number of changing Data Objects Received	80 / sec	Analog, Integer or Boolean type of Data objects.
Maximum number of MMS Control Command sent	1 / sec	MMS Control Commands sent from CI868 via CSWI and XCBR LNs.

MMS performance of CI868 module functions are as follows:

CI868 Receive

- **Memory**: Number of Static RCB signals configured in SCD file for CI868 Receive.
  This is the sum of all Data Objects signals subscribed to CI868 in all Datasets in all IEDs.
- **Load**: Number of changing DO signals for CI868 Receive.
  This is the sum of all DO signals subscribed to CI868 changing at a given moment across all IEDs. The number of MMS signals changing in lesser number of Datasets consumes less CI868 load as against the same number of signals changing in more number of datasets.

CI868 Performance for MMS Client and GOOSE Protocol Combined Usage

The following are the configuration and performance parameters related to CI868 module configured for GOOSE and MMS communication.

It is recommended to follow the mentioned limits to have CI868 CPU load levels less than 85% for satisfactory performance of CI868 module for GOOSE and MMS communication.

Following are the recommendations to be followed while engineering the SCD file.

Table 34. Performance Parameters of CI868 Configured for GOOSE and MMS Client

Description	Limit	Remarks
Control Builder Configuration Data Per CI868 Module Refer to Table 32 and Table 33 for GOOSE and MMS Configuration Limits.		
GOOSE + MMS Performance Data Per CI868 Module		
Maximum number of IEDs connected	20	

Table 34. Performance Parameters of CI868 Configured for GOOSE and MMS Client (Continued)

Description	Limit	Remarks
Maximum number of Data Objects per Dataset	10	Analog, Integer or Boolean type of Data objects. This number should not be increased, otherwise the CI868 board could be overloaded. For more data signals to CI868 from same IED, use additional Datasets.
Maximum number of changing Data Objects Received	60 / sec	Analog, Integer or Boolean type of Data objects.
Maximum number of MMS Control Command sent	1 / sec	MMS Control Commands via CSWI and XCBR LNs.

Calculation of I/O Copy Time Estimate for ControlNet with CI865 Unit

Each ControlNet node (200-ACN, 200-RACN and CI865) has its own I/O data memory that is asynchronously updated.

Different configurations and parameters, depending on the I/O system type that is used, determine the total I/O update time.

To estimate the maximum time, from I/O point status change until it is processed in the application program, all times from I/O point to Task Interval Time, t_{ti}, have to be added according to the formula below.

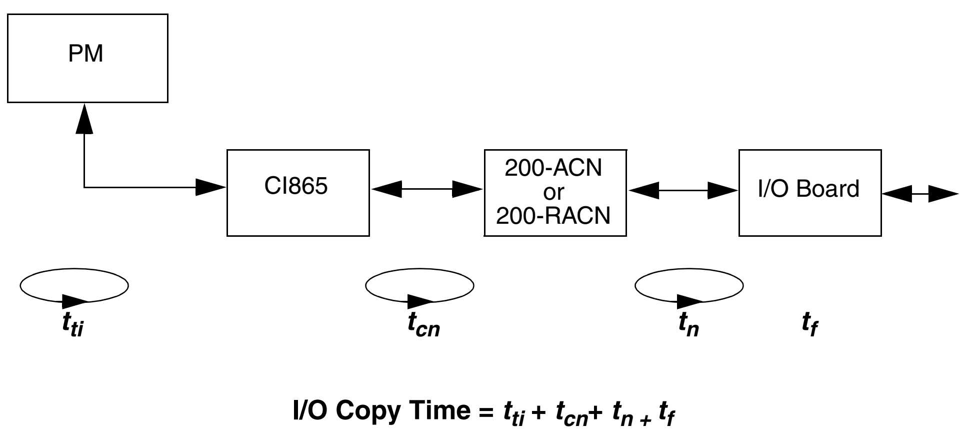


Figure 10. I/O Copy Schedule

Remote Series 200 I/O and Rack I/O

The transmission on the ControlNet network, t_{cn}, runs asynchronously with the execution of the application program and the I/O copy cycles on 200-ACN and 200-RACN, and is determined by the network parameters.

t_{cn} for input signals equals the EPR (Expected Package Rate) for the specific node. The EPR is a user definable setting, 5-60ms.

t_{cn} for output signals equals the NUT (Network Update Time) for the specific node. The NUT is a user definable setting, 5-60ms.

Series 200 I/O

The 200-ACN I/O memory is updated cyclically, asynchronously with the execution of the application program. The node update time, t_n, is determined by the number and type of I/O units. The approximate copying times are 0.05ms for digital I/O units and 0.2ms for analogue I/O units. There is an overhead of about 2ms for each cycle.

Example 1:
A 200-ACN configured with 8 analogue I/O units gives the following node update time:

$t_n \approx 2+8*0.2 \approx 3.6$ms

Example 2:
A 200-ACN configured with 8 digital I/O units gives the following node update time:

$t_n \approx 2+8*0.05 \approx 2.4$ms

Rack I/O

The 200-RACN I/O memory is updated cyclically, asynchronously with the execution of the application program. The node update time, t_n, is determined by the number and types of connected to 200-RACN.

The copying of the analogue input boards is spread out in time due to the relative long copying time. One analogue input board is copied each cycle (for example, if there are three analog input boards, each one of them will be copied every third cycle).

The approximate copying times are 0.14 ms for digital boards and analogue output boards and 1.2 ms for analogue input boards. There is an overhead of about 1ms for each cycle.

Example 1:
A 200-RACN is configured with 12 digital boards, 2 analogue output boards and 2 analogue input boards. The node update time, t_n, for this rack is calculated according to the following:

One cycle corresponds to: $1+14*0.14+1*1.2$ ms ≈ 4.2ms

Two cycles are needed to copy all analogue input boards, which gives the total node update time for this node: $t_n \approx 2*4.2 \approx 8.4$ms

Example 2:
A 200-RACN is configured with 11 digital boards, 2 analogue output boards and 3 analogue input boards. The node update time, t_n, for this rack is calculated according to the following:

One cycle corresponds to: 1+13*0.14+1*1.2 ms ≈ 4.0ms

Three cycles are needed to copy all analogue input boards which gives the total node update time for this node: $t_n \approx 3*4.0 \approx 12$ms

Filter Time

The I/O filter time, t_f has to be added for input boards/units.

EtherNet/IP and DeviceNet

For EtherNetIP / DeviceNet configurations with CI873 the following dimensioning guidelines needs to be taken into account.

General

The limitations, with respect to the various devices in general are:

- CI873 can act only as a scanner.
  It does not accept class 1 and class 3 connections from any other scanner.
- The number of I/O modules that can be connected under Ethernet/IP or DeviceNet device adapter type device is 63.
- The number of configuration parameters supported per EtherNet/IP or DeviceNet device is 1000.
- The CI873 supports Listen only connection with EtherNet/IP device, provided there is already Exclusive owner connection in the device. The CI873 does not support Redundant owner connections for EtherNet/IP devices.
- The Read only parameter and monitoring parameters in EDS file are not supported in this release.
- The tag based Class 1 information should be there in EDS file for communication with Allen Bradley PLC where Class 3 tag can be added along with Class 1 connection.

- The total number of Input and Output bytes along with channel status bytes should not exceed more than 80Kb per CI873.
- The Configuration assembly size of 200 is supported per EtherNet/IP or DeviceNet device.
- The CI873 supports 20 CIP connections (including Class 1 and Class 3) per EtherNet/IP device. CI873 supports total of 128 connections.
- The CI873 only supports devices which uses EtherNet/IP encapsulation of CIP.
- CI873 does not support PCCC, Modbus encapsulation.
- CI873 supports CH1 Ethernet interface with a speed of 100 Mbps.
  CH2 is not supported.
- A maximum of 6 non redundant CI873 can be connected to each AC 800M controller.

EtherNet/IP

The limitations, with respect to the EtherNet/IP device involved are:

- EtherNet/IP supports three Class 1 connection and three Class 3 tag per Allen Bradley Control Logix PLC. The CI873 supports three Class 3 tags with 100ms cycle time.
- The data transfer, using the Class 3 connection, will be slower than the Class 1 connection.
- The Class 3 connection is not supported for any EtherNet/IP devices except Allen Bradley Control Logix PLC. The CI873 uses tag based Class 3 to write data to it.
- The maximum number of bytes support for Class 1 read tag is 496 and for Class 3 write tag is 432.
- 1000 bytes per Class 1 connection is supported,
  for example O->T: 500 and T->O : 500.

DeviceNet

The limitations, with respect to the LD 800DN linking device (for DeviceNet) are:

- The maximum number of input bytes supported by LD 800DN is 496 bytes. If the total number of input bytes of all DeviceNet slaves configured under the linking device exceeds 496 bytes, download is stopped.
- The maximum number of output bytes supported by LD 800DN is 492 bytes. If the total number of output bytes of all DeviceNet slaves configured under the linking device exceeds 500 bytes, download is stopped.
- A maximum of four LD 800DN linking devices can be connected under one CI873.
- Multiple CI873 cannot listen to same LD 800DN data.
- The maximum number of DeviceNet connections per device is restricted to 5.

Performance Data

Typical performance of the CI873 is:

- CI873 can handle a maximum of 10 CIP connections with 10ms RPI. However it can handle a maximum of 128 CIP connections.
- The reaction time of CI873, that is, the time from changed input channel to the time setting an output channel is less than 100ms at a maximum CI873 CPU Load of 80%, provided the data is sent over a connection operating at an RPI of 50ms or less.
- Data throughput of 1000 CIP I/O packets receive/second and 500 CIP I/O Packets sent/second can be achieved at an optimum load of 85%.
  Each I/O packet can have data size ranging from 4 to 500 bytes.
- Redundancy Switchover time is 120ms for 10 CIP connections operating at 10ms RPI, that is, the time I/O communication stops in primary to the time I/O communication starts in switched primary.

Communication

IAC and MMS Communication

Communication performance is affected by *bandwidth, message length* and *cyclic load*.

Higher load on the CPU will cause lower throughput in the communication, and lower load will give higher throughput.

The 10 Mbit/s is an ethernet speed which is in balance with the performance of the AC 800M controller. The maximum data flow to and from the software in an AC 800M is less than 10 Mbit/s. This means that the data flow for one AC 800M is not limited due to its ethernet speed of 10 Mbit/s.

The Ethernet standard allows bandwidth transmission at 10 Mb/s, 100 Mb/s (fast Ethernet), and 1000 Mb/s (Gbit Ethernet) and AC 800M supports 10 Mb/s and 100 Mb/s (PM891 only).

In a system with several controllers and PCs a switched network should be used between the nodes. If hubs are used instead of switches the number of connected nodes plays an important role for the throughput of the network and a single node may get an ethernet throughput which is less than the nominal network speed. With switches this is however not the case. Each node gets an ethernet throughput which is more or less independent of the number of connected nodes. This means that the data flow in the complete system is also not limited by AC 800M's ethernet speed of 10 Mbit/s.

For networks with several switches ABB recommend to use 100 Mbit/s or 1 Gbit between switches since those ports need to manage data from several nodes to several nodes. 10 Mbit/s should only be used on the ports where AC 800M controllers are connected. Those ports only need to manage data for one node.

The actual communication throughput for a controller thus mainly depends on other factors than the ethernet speed, for example the cycle times of the applications and the CPU load in the controller.

Connections Cannot Block Each Other

The controller can handle a number of concurrent communication connections. All connections are handled in a round robin fashion. This means that no connection can block communication for any other connection.

For example this means that it is guaranteed that variable access from one controller to another can always be executed even if a Control Builder is downloading a very large application domain to one of the controllers.

Number of Connections

The controller's communication stack handles several simultaneous connections. Messages are treated in a round robin fashion that guarantees that no connection is starved, but the transmission rate through the stack decreases slightly with the number of active connections. With 20[1] or less connections the performance decrease per additional connection is however small. With more than 20[1] connections the amount of buffers per connection is reduced. This may decrease the performance for the connections substantially more, at least for connections transmitting much data.

Communication Load

Table 35 shows the execution time for transferring one variable of type dword between two AC 800M controllers using IAC or MMS, server or client.
It also tells the caused cyclic and total load for communicating one dword per second.

The measurements were done by transferring as many dwords as possible in each transaction, that is, non-SIL IAC = 350 dwords, non-SIL MMS = 166 dwords.
The task interval time was set to 200ms, and the IAC interval time to 100 ms.

Table 35. Load Caused by IAC and MMS Communication

	PM864
IAC Server, Out variables	
Execution time/dword [μs]	0,85714
Cyclic load /(dword/sec) [%]	0,00004

1. 30 in PM866 and 40 in PM891.

Table 35. Load Caused by IAC and MMS Communication

	PM864
Total load /(dword/sec) [%]	0,00032
IAC Client, In variables	
Execution time/dword [µs]	0,53333
Cyclic load /(dword/sec) [%]	0,00001
Total load /(dword/sec) [%]	0,00035
MMS Server, Read	
Execution time/dword [µs]	0,00000
Cyclic load /(dword/sec) [%]	0,00000
Total load /(dword/sec) [%]	0,00188
MMS Client, Read	
Execution time/dword [µs]	0,30120
Cyclic load /(dword/sec) [%]	0,00000
Total load /(dword/sec) [%]	0,00434

Communication Throughput at 40% Cyclic Load

Table 36 shows how many dwords an AC 800M can transfer via IAC or MMS, to or from, another controller in case its cyclic load (without communication) is at 40%.

The number of dwords were increased until the controller was considered to be throttled. The criteria for throttling were set to:

- Max. cyclic load
  - Controller: 70%
- Numbers of transactions/sec is maximized and does not increase anymore

The underlined value shows what criteria caused the throttling. The given values show the maximum communication throughput.

The measurements were done by transferring as many dwords as possible in each transaction, that is, non-SIL IAC = 350 dwords, non-SIL MMS = 166 dwords. The task interval time was set to 200ms, and the IAC interval time to 100 ms.

Table 36. IAC and MMS Communication Throughput at 40% Cyclic Load

	PM891	**PM864**
IAC Server, Out variables		
Max no of dwords communicated	11550	6300
Transactions/sec	180	90
Cyclic load [%]	41	42
Total load [%]	54	69
IAC Client, In variables		
Max no of dwords communicated	5950	5250
Transactions/sec	90	87
Cyclic load [%]	40	41
Total load [%]	50	67
MMS Server, Read		
Max no of dwords communicated	7470	3652
Transactions/sec	225	110
Cyclic load [%]	40	40
Total load [%]	71	89
MMS Client, Read		

Table 36. IAC and MMS Communication Throughput at 40% Cyclic Load

	PM891	PM864
Max no of dwords communicated	4150	1992
Transactions/sec	125	60
Cyclic load [%]	40	40
Total load [%]	91	93

Modbus RTU Master Communication

AC 800M 50% Load in the Controller 300 Booleans in Each Telegram	Max Transmission Rate (Total Transactions/second)			
	PM864A / PM866 / PM891			
	MBWrite		MBRead	
	1 channel	4 channels	1 channel	4 channels
1200 baud (8 data bits, 1 stop bit, odd parity)	1.3	2	1.7	5.6
19200 baud (8 data bits, 1 stop bit, odd parity)	10.1	19.8	10.5	36.3

MODBUS TCP

Table 37. MODBUS TCP Performance Data. Reading Dint using one CI867 as Master

Number of Slaves Connected and Communicating	Message Length (in Dint)	Total Transactions/s (Sum of all Slaves)	Average Transaction /Slave[1]
1 task time=100 ms	60	149	149
1 task time=250 ms	60	89	89
5 task time=100 ms	60	158	31
5 task time=250 ms	60	160	33
10 task time=100 ms	60	150	15
10 task time=250 ms	60	160	17
20 task time=100 ms	60	94	6
20 task time=250 ms	60	110	9
30 task time=100 ms	60	97	3
30 task time=250 ms	60	123	4

1 Cyclic read at maximum possible rate is used.

Table 38. MODBUS TCP Performance Data. Reading Boolean using one CI867 as Master

Number of Slaves Connected and Communicating	Message Length (in Boolean)	Total Transactions/s (Sum of all Slaves)	Average Transaction /Slave[1]
1 task time=100 ms	60	114	114
1 task time=250 ms	60	104	104
5 task time=100 ms	60	130	26
5 task time=250 ms	60	129	26
10 task time=100 ms	60	113	11
10 task time=250 ms	60	121	12
20 task time=100 ms	60	100	5
20 task time=250 ms	60	141	7

Table 38. MODBUS TCP Performance Data. Reading Boolean using one CI867 as Master (Continued)

Number of Slaves Connected and Communicating	Message Length (in Boolean)	Total Transactions/s (Sum of all Slaves)	Average Transaction /Slave[1]
30 task time=100 ms	60	113	3
30 task time=250 ms	60	124	4

1 Cyclic read at maximum possible rate is used.

Table 39. MODBUS TCP Performance.Data. Reading Real using one CI867 as Master

Number of Slaves Connected and Communicating	Message Length (in Real)	Total Transactions/s (Sum of all Slaves)	Average Transaction /Slave[1]
1 task time=100 ms	60	104	104
1 task time=250 ms	60	95	95
5 task time=100 ms	60	120	24
5 task time=250 ms	60	120	24

1 Cyclic read at maximum possible rate is used.

Table 40. MODBUS TCP Performance Data. Reading Dint using one CI867 as Slave

Number of Masters Connected and Communicating	Number of Data in Dint	Total Transactions/s (Sum of all Slaves)	Average Transaction /Master
1 task time=50 ms	50		20
1 task time=50 ms	100		20
2 task time=50 ms	50		20
2 task time=50 ms	100		20
8 task time=50 ms	50		20
8 task time=50 ms	100		20

Table 41. MODBUS TCP Performance. Data. Reading Boolean using one CI867 as Slave

Number of Masters Connected and Communicating	Number of Data in Boolean	Total Transactions/s (Sum of all Slaves)	Average Transaction /Master
1 task time=100 ms	1		10
1 task time=100 ms	525		10

Table 41. MODBUS TCP Performance. Data. Reading Boolean using one CI867 as Slave (Continued)

Number of Masters Connected and Communicating	Number of Data in Boolean	Total Transactions/s (Sum of all Slaves)	Average Transaction /Master
2 task time=100 ms	1		10
2 task time=100 ms	525		10
8 task time=100 ms	1		10
8 task time=100 ms	525		10

Control Network Clock Synchronization

Table 42. Control Network Clock Synchronization

Type of Clock Synchronization	Accuracy per node
High Precision SNTP	1 ms
SNTP	200 ms
CNCP (between AC 800M)	1 ms
CNCP (AC 800M to AC 800C/Advant Controller 250)	200 ms
CNCP (AC 800M to PPA)	200 ms
MB300 Network	3 ms

MasterBus 300 Network

The MasterBus 300 network can have maximum 100 nodes on a CI855 in a control area. The maximum performance is 200 data set per second. Switch over time to a redundant bus is 3 seconds.

INSUM Network

Table 43. INSUM Design Limitations

Limitation		
Limitation type	**Value**	**Reason**
Number of MCUs per controller	128	Execution time for IEC 61131-3 application and system heap memory
Number of MCUs per CI857	128	CPU performance on CI857
Number of Gateways per CI857	2	CPU performance on CI857 and memory on CI857
Number of CI857 per AC 800M	6	CPU performance

Table 44. INSUM Communication Interface CI857 Performance

Response time			
Action	**Result**	**Condition**	**Comments**
Start/stop, - 64 MCUs - 128 MCUs	 5-8 s 15-16.5 s	Five NVs subscribed per MCU	Time measured inside the IEC 61131-3 application, from the time it sends the first command with INSUMWrite to NVDesState until it receives the last state change with INSUM Receive from NVMotorStateExt.
Stop one MCU due to chain interlock from other MCU	500 ms	Task cycle 250 ms, 66 MCUs, five NVs subscribed per MCU	Time measured on electrical state signals on the MCUs from the time the first MCU stop until the second MCU stop.

OPC Server for AC 800M

The OPC Server for AC 800M collects data from controllers via MMS, and makes it available to OPC clients. Performance depends on the amount of MMS traffic between the OPC server and controllers. This, in turn, depends on the number of items and the rate at which the items are updated in the OPC Server. The following information is based on an OPC Server for AC 800M running on a PC with an Intel Xeon®, 2.40 GHz processor and 4Gbyte RAM.

Table 45. OPC Read Performance

Limitation type	Value
Number of subscribed variables in total at 1000 ms update rate	300 000
Maximum OPC Servers for one controller	3
Maximum OPC Clients for one OPC Server	5

Performance also depends on the controllers ability to provide the OPC server with data. This ability is controller-dependent and is shown in the table below.
The table shows how many variables (bool) that can be subscribed to, with an update rate of 1000 ms, from a controller with 50 % cyclic load, and how many MMS telegrams this corresponds to.

Table 46. Controller Response Performance

CPU	Number of subscribed items (Boolean)	Max. number of MMS transactions/second	Total System Load	Cyclic Load
PM864	40 000	75	89	50
PM866	42 000	82	78	50
PM891	60 000	120	67	50

The table below tells how long time it takes to write 1000 boolean variables to a controller when the OPC server already subscribes to 1 000, 50 000 and 100 000 variables from that controller.

Table 47. OPC Write Performance

Number of Subscribed Items at 1000 ms requested update rate	Simultaneous Write of 1000 Items
1 000	3 ms
50 000	106 ms
100 000	213 ms

Index

Contact us

ABB AB
Control Technologies
Västerås, Sweden
Phone: +46 (0) 21 32 50 00
e-mail: processautomation@se.abb.com
www.abb.com/controlsystems

ABB Automation GmbH
Control Technologies
Mannheim, Germany
Phone: +49 1805 26 67 76
e-mail: marketing.control-products@de.abb.com
www.abb.de/controlsystems

ABB S.P.A.
Control Technologies
Sesto San Giovanni (MI), Italy
Phone: +39 02 24147 555
e-mail: controlsystems@it.abb.com
www.abb.it/controlsystems

ABB Inc.
Control Technologies
Wickliffe, Ohio, USA
Phone: +1 440 585 8500
e-mail: industrialitsolutions@us.abb.com
www.abb.com/controlsystems

ABB Pte Ltd
Control Technologies
Singapore
Phone: +65 6776 5711
e-mail: processautomation@sg.abb.com
www.abb.com/controlsystems

ABB Automation LLC
Control Technologies
Abu Dhabi, United Arab Emirates
Phone: +971 (0) 2 4938 000
e-mail: processautomation@ae.abb.com
www.abb.com/controlsystems

ABB China Ltd
Control Technologies
Beijing, China
Phone: +86 (0) 10 84566688-2193
www.abb.com/controlsystems

3BSE041586-511

Power and productivity
for a better world™