Remote Terminal Unit RTU560

System Description Release 6.2

This document describes the features of the Remote Terminal Unit RTU560 Release 6.2 concerning the handling of process signal information and communication parameterization and diagnostics.

Revision

Document identity:		1KGT 150 537 V002 1
Revision:	0	Date: 05/2003
Revision:	1	Date: 09/2003 Minor Changes
Revision:	2	Date 11/2004 Support of 128 MB Compact Flash Card Support of Microsoft Windows XP Professional New Modems 23WT23 and 23WT24 New Function ‘Load Profile Archive’ New Function ‘Integrated HMI’ New Protocols for the HCI New Protocols for the SCI 2 pole Commands for RTU560E Archiving/Local Print of Commands Error Correction for ‘Setpoint Commands’

Contents

Abbreviations

AMI	**A**nalog **M**easured value **I**nput
ASO	**A**nalog **S**etpoint command **O**utput
BCU	**B**us **C**onnection **U**nit
BSI	**B**it **S**tring **I**nput (8, 16 bit)
CMU	**Com**munication and Data Processing **U**nit
CS	**C**ontrol **S**ystem
CSC	**C**ommand **S**upervision **C**hannel
CS-Command	**C**lock **S**ynch **C**ommand
DCO	**D**ouble **C**ommand **O**utput
DMI	**D**igital **M**easured value **I**nput (8, 16 bit)
DPI	**D**ouble **P**oint **I**nput
DSO	**D**igital **S**etpoint command **O**utput (8, 16 bit)
EPI	**E**vent of **P**rotection equipment **I**nput (1bit)
GCD	**G**eneral **C**onfiguration **D**ata
HCI	**H**ost **C**ommunication **I**nterface
IED	**I**ntelligent **E**lectronic **D**evice
IOC	**I/O C**ontroller (Controller on I/O Board)
IOD	**I**nput **O**utput **D**ata
IOM	**I/O** Bus **M**aster (Function of SLC)
ITI	**I**ntegrated **T**otals **I**nput
MFI	Analog **M**easured value **F**loating **I**nput
MPU	**M**ain **P**rocessing **U**nit
NCC	**N**etwork **C**ontrol **C**enter
PB	**P**eripheral **B**us
PBP	**P**eripheral **B**us **P**rocessor

PDP	**P**rocess **D**ata **P**rocessing
PLC	**P**rogrammable **L**ogic **C**ontrol
PPP	**P**oint to **P**oint **P**rotocol
PSU	**P**ower **S**upply **U**nit
RCO	**R**egulation step **C**ommand **O**utput
RTC	**R**eal **T**ime **C**lock
SBI	**S**ystem **B**us **I**nterface
SBO	**S**elect **b**efore **O**perate
SCADA	**S**upervision, **C**ontrol **a**nd **D**ata **A**cquisition
SCI	**S**ub-Device **C**ommunication **I**nterface
SCO	**S**ingle **C**ommand **O**utput
SEV	**S**ystem **Ev**ents
SLC	**S**erial **L**ine **C**ontroller
SOC	**S**trobe **O**utput **C**hannel
SPI	**S**ingle **P**oint **I**nput
STI	**St**ep position **I**nput (8 bit)
TSI	**T**ime **S**ynch **I**nput
TSO	**T**ime **S**ynch **O**utput

1. Application and Features

1.1. Application

The task to monitor and control the transportation network for energy to reach an economical operation control requests an ongoing penetration of the network down to the lowest levels. The increasing requirements to the availability of energy and their distribution and transportation also increases the demands on the control systems. Improved and new communication possibilities with a higher transmission bandwidth and new transmission media allows the telecontrol technology to realize these tasks.

The RTU560 is required to be configurable to nearly all demands made on remote stations in networks for electricity, gas, oil, water or district heating.

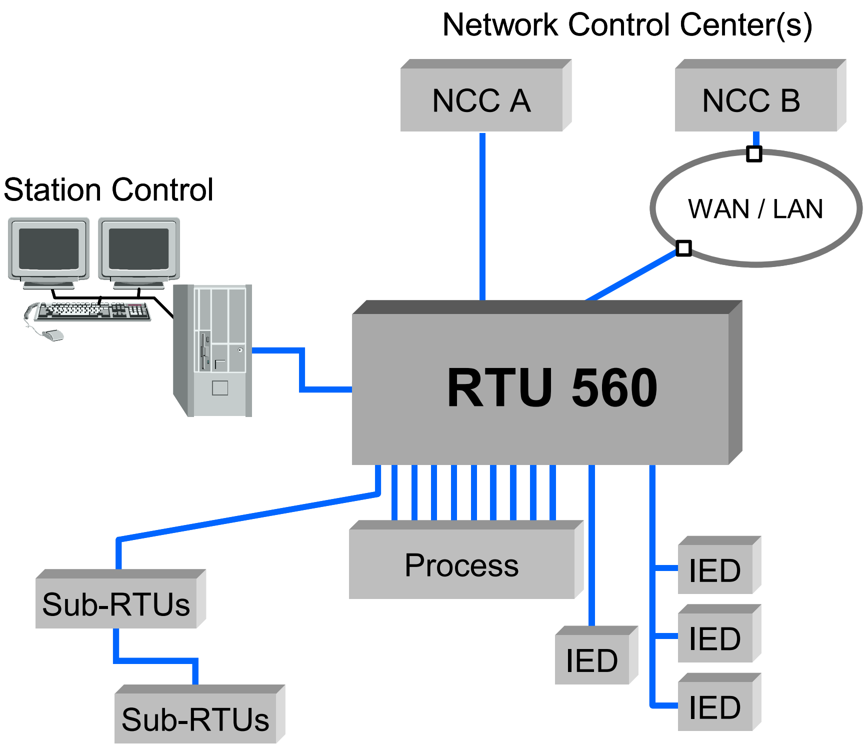


Figure 1-1: Typical configuration of a telecontrol system

Increasing capabilities of decentralized control and closed-loop control solutions allows to run more functions to be done in the station directly. The RTU560 supports this by own PLC programs which may use for control tasks on one side and by the capability to communicate with the external control, protection and monitoring units via serial lines on the other side. The RTU560 will distribute process information from these units on the demands for station- and network control to several network control centers (NCC).

The RTU560 is using a set of communication units (CMU) and I/O boards with a good modularity to build up the RTU configurations optimized for the application and data point profile in the station. Starting with a configuration for some I/O process data points and one communication unit for typical small pump stations or ring main unit stations over medium size stations for distribution up to large stations on transmission grid level.

The engineering work is a relevant cost factor that can be reduced by standardization of the process data model and the use of state-of-the-art engineering tools. The tool must support all type of configurations and communication network for telecontrol which are possible by the RTU560 family and the customers demand for the distributed stations.

Engineering of the process signals for the RTU560 is done by means of only one tool 'RTUtil 560' for all stations with RTU560 units and projects. Project is here in the definition of a telecontrol network with several remote stations combined by router stations etc. RTUtil 560 supports process signal routing from a small station on the lowest level up to the highest level for network control centers (NCC). Typically it includes the conversion from a telecontrol protocol 'A' to another telecontrol protocol 'B' used on the next level. For example from DNP 3.0 to IEC 870-5-104. RTUtil 560 generates all files requested to run the RTU560 units. To reduce traveling costs and to get a higher flexibility for configuration extensions or modifications, RTUtil 560 and the RTU560 concept allows to download the files into the RTU560 in the stations via INTRANET using WEB browser technology or via the communication line, when the protocol supports file transfer.

1.2. Features

The telecontrol system RTU560 should be in the position to transmit nearly all kind of process information, derived from various units in the station, to the control centers and to marshal commands received from the control centers to the addressed control unit within the station.

Beside the acquisition and processing of the directly parallel wired process signals to the RTU560 IO-process interface, the RTU560 is designed for the link of serial communication routes within the station as well to the higher control level. This can be another RTU560 router station or a network control center. Within the station it is the connection of other existing additional control, protection or monitoring devices (Intelligent Electronic Devices = IED) via serial interfaces.

The RTU560 concept allows the economical adaptation to the requested, different serial links by cascading the communication and processing units (CMU=Communication Unit) according to the number of needed serial interfaces.

Functional system features of the RTU560 to fulfill the requirements for remote control stations:

- High functional scope for telecontrol applications functions
- PLC capabilities to execute control and closed loop control applications for pump stations, hydro power plants, station interlocking for electrical substations, etc..
- Archiving of process and station events in a sequence of events list in the Flash memory. Accessible via Intranet or equivalent independent network.
- Archiving of Integrated Totals (ITI) and Analog Measured Values (AMI) in the Flash memory. Accessible via Intranet or equivalent independent network.
- Reading and archiving of disturbance files from protection relays on request of the protection relay. Reading of the disturbance files by file transfer over a separate communication network (e.g. Intranet) on user's demand. Independent and direct information of available new disturbance files in the disturbance file archive to the NCC.
- Possibility to build (engineer) group alarms for the typical alarm messages, beside a PLC program.
- Interfacing nearly all types and big numbers of IEDs in a station via serial telecontrol protocols, like IEC 609870-5-103, MODBUS, SPA-Bus, DNP 3.0.
- Marshalling and filtering process events to the connected NCCs . Decoupling transaction sequences and delay times to the different NCCs by using a separate process data base per NCC link.
- Remote access for diagnostic purposes via Web-Browser and Internet or Intranet. With detailed information down to each process signal.
- Integrated HMI (**H**uman **M**achine **I**nterface) for process super vision and control. Via Web-Browser and Internet or Intranet.

1.2.1. *RTU560 design features*

The RTU560 is an incremental development step based on ABB's proven RTU232 with a centralized and very flexible communication concept. The concept can be split into two parts

- a new communication unit. Highly configurable to the demands in a station or as router in a telecontrol communication network.
- the proven and still valid process I/O interfaces for direct connection of the process signals in monitoring and command direction. This I/O board set is taken over from the RTU 232.

Within the RTU560 family the communication unit and the I/O board family is a hardware system based on standard European format cards. To meet the requirements for typical medium stations with only some communication links on one side and large or modern stations with a higher number of IEDs on the other side, the RTU560, based on European format cards, is available in two versions

- RTU560A
  for configurations with higher demands on communications links. The parallel wired process interface is still part of the configuration.
- RTU560C
  for typical stations with a parallel wired process interface and some communication links only

The RTU560 system family is expanded by the

- RTU560E.

1.2.2. *RTU560E*

The target applications for the RTU560E are small stations with a typical number of IO points in the range of 50 signals. The RTU560E has a new hardware concept to meet the cost level for such small RTUs. But the system concept, developed for the RTU560 is fully taken over and available in the RTU560E.

- Same solutions as used for RTU560A and RTU560C in hardware design and software
- One tool for the complete RTU560 family
- Same software packages for the complete family = no RTU560E specials
- High flexibility for communication configurations adopted to the demand of a small station
- One housing only which fulfills the requirements for typical small stations.
- The RTU560E can be used as small RTU with signal processing and communication link, or

-
- can be used as passive IO process signal terminal to be located within a station next to the process signal source (e.g. within a feeder bay, next to a marshalling rack, a control unit etc.). In this version the RTU560E is connected to the RTU560 Peripheral Bus and controlled by a RTU560A or RTU560C.
- Binary Process voltage can be up to 220 V DC, for the directly connected input signals.
- Command relays fulfill the requirements to switch the circuit breaker coil directly. Same type, then used for protection relays. Process voltage max. 220 V DC.

1.3. System Concept

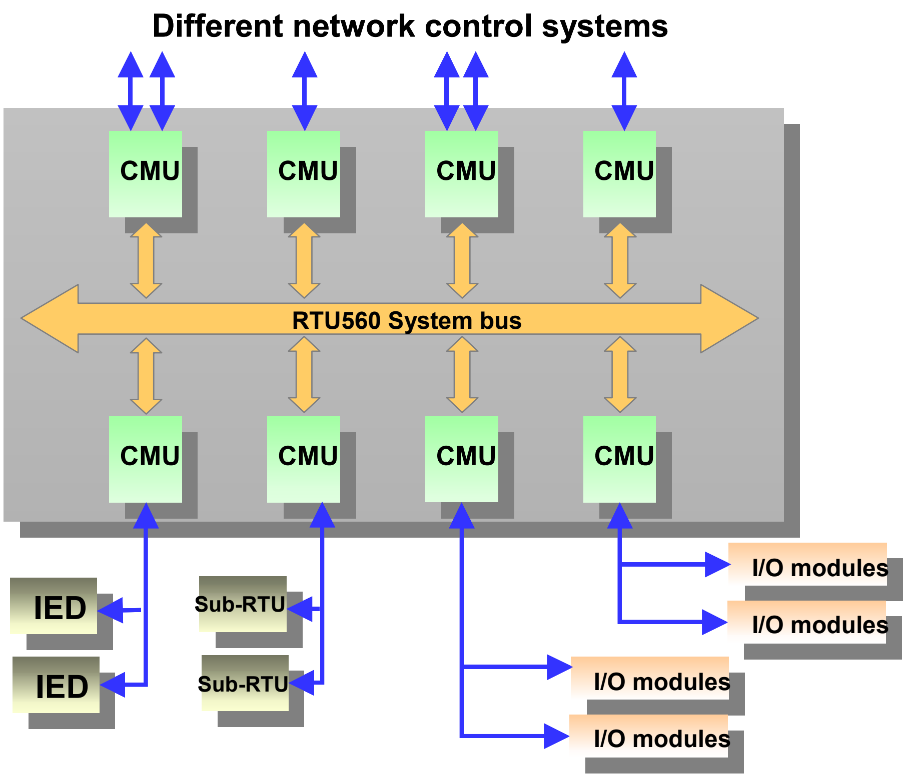


Figure 1-2: RTU560 System concept

The above figure shows the basic concept for the RTU560 family. The RTU560 is based on a communication node which is highly flexible. The number of CMU boards depends on the demands in a station or router RTU.

Each CMU type has a number of serial interfaces to connect serial communication links. Each CMU can run up to two different communication protocols either as Host Communication Interface (HCI=Slave protocol) or as Sub-device Communication Interface (SCI = master protocol).

This concept allows to cascade the number of CMUs to the demands on different protocols and interfaces.

The second main point is the internal communication concept. To avoid several special conversions etc. all process information, regardless from which interface received, are converted into in internal presentation and distributed to all CMUs via the RTU560 System bus. Therefore each protocol module needs always only the conversion into / from the internal presentation.

This requests also that each protocol module has its own process data base for signal processing etc.

This concept is also valid for the RTU560E. It allows, that all protocols for the RTU560A / C are also available for RTU560E. The RTU560E has only three serial interfaces on its CMU board available and can not be expanded.

1.4. RTU560 features

The basic design is taken from the RT232 and has demonstrated its stability in several installations world wide. New features are given by the RTU560 in:

- new processor technology for the PC104 CPU modules
- Communication sub-rack 560CSR01 for RTU560A stations to configure the several CMUs
- Redundancy concept for RTU560A / C:
- Redundant power supplies for the communication sub-racks of RTU560A
- Redundant communication links for serial lines and Ethernet possible on one CMU
- Redundant CMU concept with cold standby performance
- Process event archive in Flash memory
- Disturbance file archive in Flash memory with directory organization per protection relay and autonomous read out of the disturbance files
- Remote diagnostic via Web-Browser for all RTU560 families
- max. 3000 process data points within one RTU560A / C, including all direct connected process signals and all signals via serial links
- Increasing performance for event access via RTU560 Peripheral bus by splitting the IO-racks on bus segments. Maximum 4 bus segments.
- Possibility to run more then one PLC program in parallel
- Small number of different hardware units
- High availability
- One engineering tool for the complete RTU560 family RTUtil 560

2. System Overview

2.1. RTU560 System

The RTU560 system will be described in the following chapters

- Hardware family for:
  RTU560A
  RTU560C
  RTU560E
- Communication and Application modules
  CMU organization
  Communication node RTU560
  Application modules
- Application functions
  Telecontrol functions
  PLC (**P**rogrammable **L**ogic **C**ontrol)
  Local Print
  Process Archives
  Disturbance Data Archives
  Load Profile Archives
  Integrated HMI (**H**uman **M**achine **I**nterface)
- RTU560 configuration
  RTU560A
  RTU560C
  RTU560E

2.2. RTU560 Hardware Family

2.2.1. Racks and Housing

The RTU560 hardware can be split in two design groups.

- Standard European format boards (160 x 110 mm / 3HE) used for the RTU560A and RTU560C
- Double sized European format boards (320 x 256 mm / 6 HE) used for the RTU560E,

The RTU560A / C boards are placed in 19" standard racks. Several racks can be mounted in cubicles or wall housings of different types. A set of additional mounting parts, miniature circuit-breaker racks, connector terminals and adapter cables are available for installation and wiring in the cubicles.

The process signal connection is done indirect by connector terminals. Depending on cubicle type and space it is possible to mount a marshalling plate on the backside of the cubicle with terminators for the IO-board signals on one side and the process cables on the other side.

RTU560A

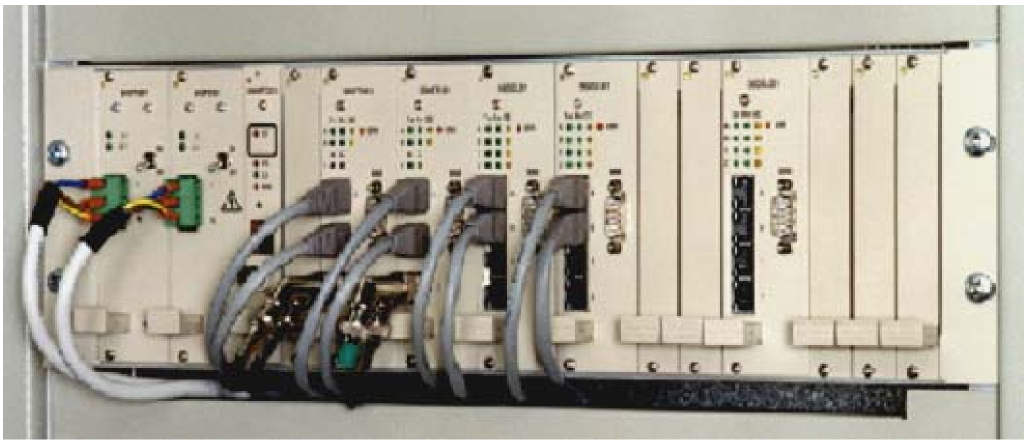


Figure 2-1: Communication sub-rack RTU560A

The communication sub-rack 560CSR01 is used to build up the communication node with several CMU boards. The maximum number of 560CSR01 sub-racks per RTU560A is limited to two racks. The link between the two racks is done by the bus connection unit 560BCU01 placed in both sub-racks and a special cable to connect the two RTU560 system bus segments together.

The sub-rack 560CSR01 is used for the RTU 560A.It allows to plug:

- max. 2 power supplies 560PSU01 which have the capacity to supply up to 4 CMU boards, the real time clock board and the bus connection unit. A second sub-rack 560CSR01 has to be used when more CMUs are requested.
- all types of communication and processing units (CMU).
- the real time clock boards for GPS (560RTC01) or DCF77 (560RTC02) time standard
- the bus connection unit 560BCU01 which is used to extend the RTU560 system bus, to handle the minute pulse as input signal from a station master clock or to distribute the RTU560 minute pulse to other RTU560 units etc. Two potential free relay contacts are available to indicate an alarm or warning of the RTU560A.

The sub-rack is totally shielded to fulfill the EMC conditions. All cables and connectors are on the front side.

RTU560C

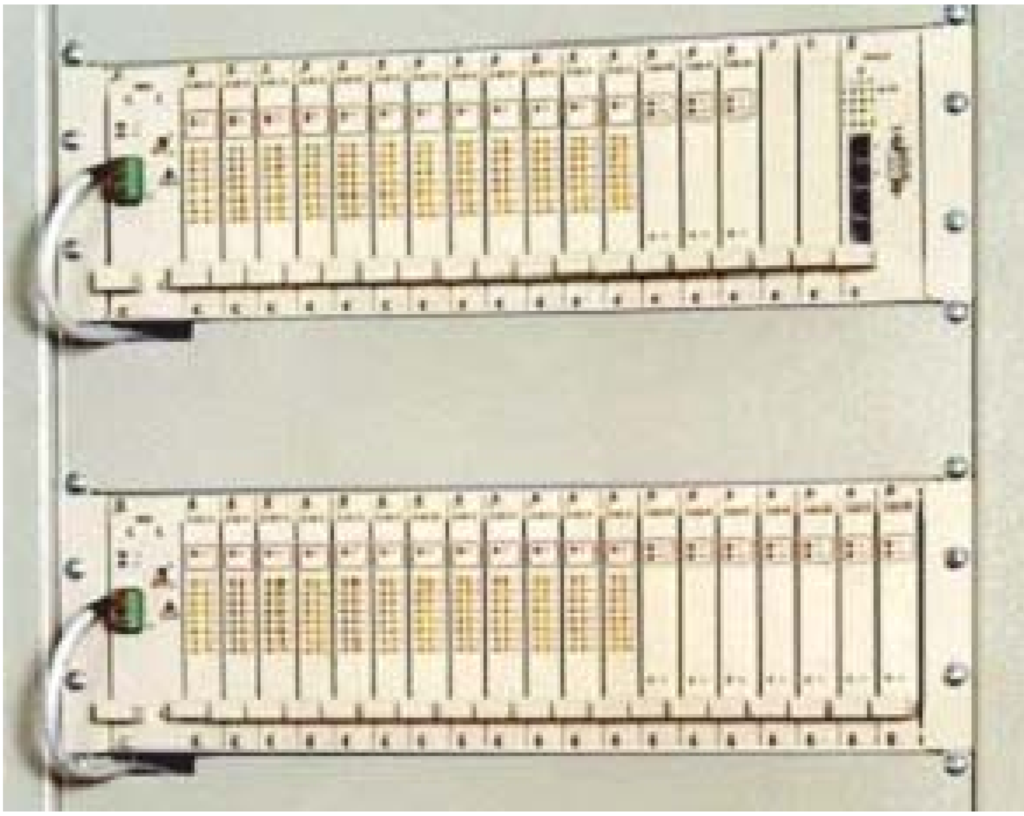


Figure 2-2: RTU560C with two IO Sub-racks23ET23 and one CMU board

The RTU560C is used for stations where mostly all process signals are wired to the IO boards and only some communication links needed. The CMU boards for the RTU560C can be installed in a normal IO sub-rack 23TP21 or 23ET23. The RTU560C allows to use maximum two CMU boards per RTU. They will be placed in the first IO sub-rack. The RTU560 system bus etc. for the two CMU boards is included in the bus connector modules 560BCU02 (23TP21) or 560BCU03 (23ET23).

All three bus connector modules have the same functionality.

The residual slots in the sub-rack with CMUs can be used for the IO boards.

RTU560E

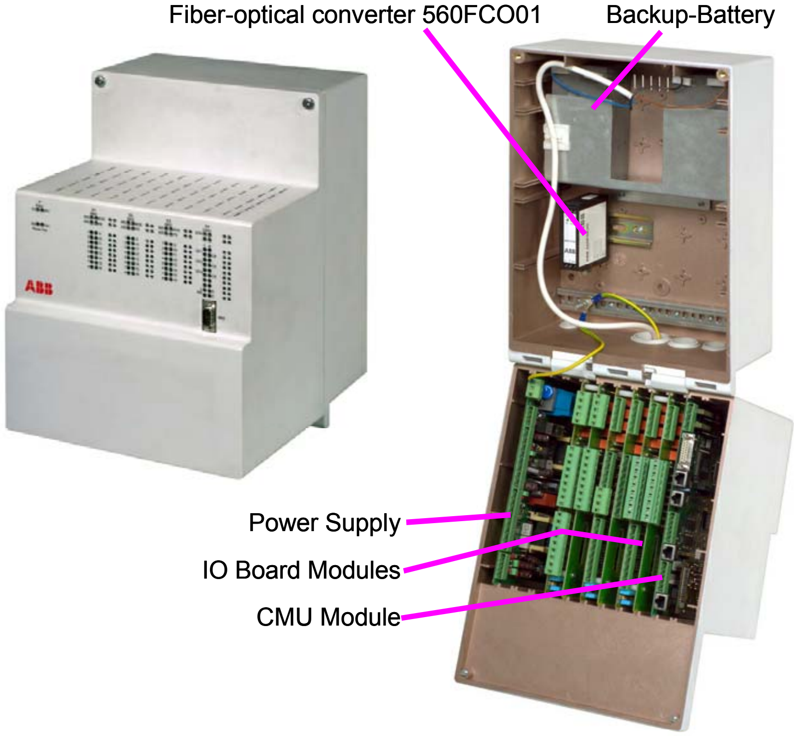


Figure 2-3: RTU560E housing and modules

The RTU560E housing allows to install the RTU56E near to the process. The shielding of the housing is sufficient to fulfill the EMC profile for electrical substations. Within the housing is space to install a battery for backup in case of AC voltage supply. In the lower part is space to install the optional optical interface unit 560FOC01 or other small modules (e.g. special modems). Thus no additional part may be needed to be installed next to the RTU560E housing for standard applications.

There are two types of housing available

- 560HOS01:
  for an RTU560E with CMU module, max. 3 IO board modules and power supply
- 560HOS02:
  for an RTU560E IO board extension with maximum 4 IO board modules and power supply

The RTU560E IO-box is an advantage for substations where the IO-signals are located in groups spread over the station area etc.

2.2.2. Main- and IO-boards RTU560A and RTU560C

A set of general boards and of IO boards can be used to configure the RTU560A / C. A more detailed description is given in chapter 5 of this manual.

An RTU560 is mainly specified by the CPU boards, general boards and the IO boards.

CMU boards:

- CMU 'Serial Line Adapter' 560SLI02
  PC104 module concept with processor, RAM and Flash
  4 serial line interfaces with RJ45 connectors
  2 driven by a UART CP1 and CP2 / RS232C or RS422
  2 driven by a serial line controller (SLC) for non UART protocols CPA and CPB / RS232C or RS485
  maximum Baud rate over all 4 channels 38,4 kbit/s
- CMU 'Ethernet Adapter' 560ETH02
  same PC104 module then 560SLI02
  1 Ethernet interface 10 Mbit/s in 10Base2 (Coax-connector)
  2 serial line interfaces with RJ45 connectors driven by a serial line controller (SLC) CPA and CPB / RS232C or RS485
- CMU 'Ethernet Adapter' 560ETH03 R001
  same PC104 module then 560SLI02
  1 Ethernet interface 10 Mbit/s in 10BaseT (RJ45 connector)
  2 serial line interfaces with RJ45 connectors driven by a serial line controller (SLC) CPA and CPB / RS232C or RS485
- CMU 'Ethernet Adapter' 560ETH03 R002
  same PC104 module then 560SLI02
  same as 560ETH03 R001 but with 2 Ethernet interfaces of type 10BaseT (RJ45 connector)

General boards:

- Master clock boards
  560RTC01 for GPS time standard
  560RTC02 for DCF77 standard
- Modem:
  23WT22 FSK modem according to CCITT frequencies and bit-rate from 50 to 2400 bit/s (R.35…R.38)
  23WT23 standard V.23 modem for 1200 bit/s
  23WT24 modem with 9600 bit/s bit-rate
- Fiber Optic Coupler 23OK22
  R5011 for optical glass fiber
  R5012 for optical plastic fiber
- Power Supply modules
  560PSU01 for the RTU560A communication sub-rack 560CSR01
  23NG24 for all other RTU560 sub-racks 23TP21, 23ET23
  Mains Adapter 23VG23 for AC/DC conversion and UPS functionality; sufficient to supply one I/O sub-rack (24 V DC / 2 A)
  Mains Adapter 23VG24 for AC/DC conversion; 24 V DC / 10 A

IO boards RTU560A and RTU560C

The IO board family is taken over from the RTU232 system. It has the advantage to upgrade existing RTU232 systems to RTU560A or RTU560C with higher performance and more communication flexibility when requested.

- Binary Input 23BE21/22
  16 input channels galvanic isolated and grouped in 2 groups of 8 signals
  24 … 60 V DC input
- Binary Output 23BA20
  16 output relay channels galvanic isolated and grouped in 2 groups of 8 relay contacts
  24 … 60 V DC switching voltage
  switching capacity 40VA (L/R = 30 ms)
- Analog Input 23AE21
  8 input channels
  Signals: 1 … 40 mA / 2 or 20 V DC bipolar
  Resolution: 12 bit + sign
  Accuracy: class 0,25
- Analog Output 23AA20
  2 output channels
  Signals: 2,5 … 20 mA bipolar
  Resolution: 11 bit + sign
  Accuracy: class 0,25

2.2.3. Main- and IO-boards RTU560E

The hardware design for the RTU560E is taken from the RTU560A / C. The main difference is the board organization to fulfill the requirements for a small RTU. The RTU560E boards are double size European format boards.

- CMU board 560CMU80
  same processor, RAM and Flash then for the other CMU boards
  R001 with 3 serial line interfaces (CP1, CP2, CPA), CP1, CP2 / RS232C CPA / RS485
  capability to plug in 2 modems (optional) for 2 serial lines
  R002 with 1 Ethernet interface 10 Mbit/s in 10BaseT (RJ45 connector) and 3 serial line interfaces CP1, CP2, CPA
  capability to plug in 2 modems (optional) for 2 serial lines

- IO-module 560MIO80
  includes 3 independent IO boards
  Binary input
    16 input channels galvanic isolated and grouped in 2 groups of 8 signals
    24 … 220 V DC input
  Binary output
    8 output relay channels galvanic isolated and grouped in 2 groups of 4 signals
    24 … 220 V DC switching voltage
    switching capacity 40 VA (L/R=30ms, R0001) or 50 VA (L/R=40ms, R0002)
  Analog input
    4 input channels
    Signals   1 … 40 mA / 2 or 20 V DC bipolar
    Resolution   12 bit + sign
    Accuracy: class 0,25
- Power Supply 560PSU80
  sufficient to supply one RTU560E
  R001:   input voltage 24 .. 60 V DC
  R002:   input voltage 110 … 220 V DC
- Power Supply 560PSU81
  sufficient to supply one RTU560E and to deliver the process signal voltage
  mains input: 230 V AC
  including battery backup and UPS functionality
  Process signal voltage: 24 V DC

2.3. Communication and Application Modules

2.3.1. RTU560 communication and module concept

The high processing performance of the RTU560 is accomplished by the efficient distribution of the tasks to the communication and processing units (CMU) and the micro-controllers on the I/O boards. The software concept is designed to communication and application modules which have clear interfaces between each other. This allows to configure and arrange the modules in a good flexibility to the customer project demands.

Already the RTU232 has the basis of this concept by splitting the work between the CPU board (23ZG21) and the IO boards. Each IO board has its own micro-controller and does the basic tasks for the connected process signals, like time stamping, threshold monitoring for analog input signals or command output supervision for switching commands.

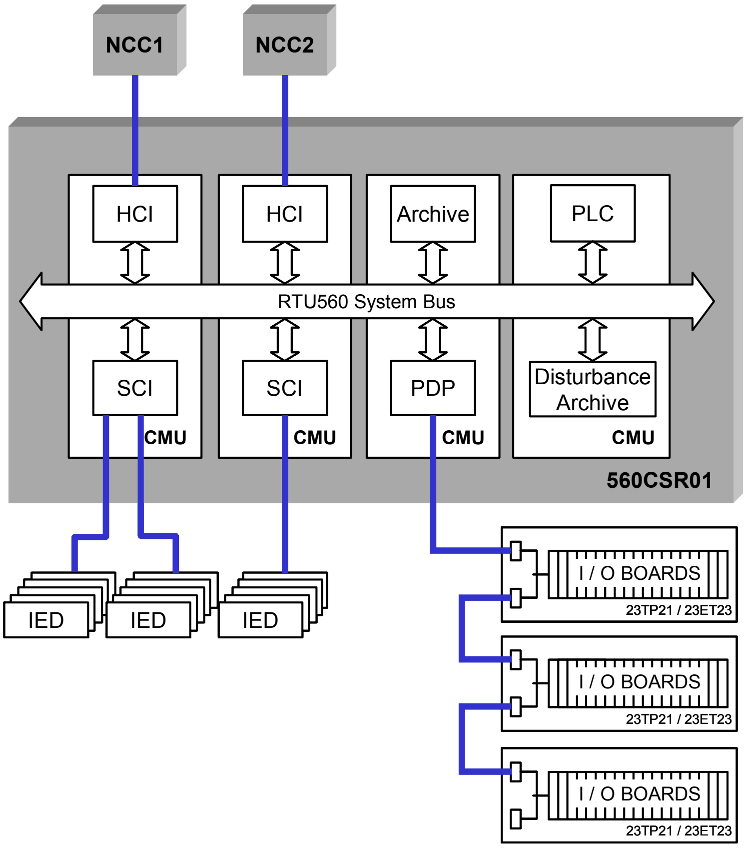


Figure 2-4: Example of a RTU560 organization in hardware modules

Backbone in the RTU560 is the internal communication concept. It allows this flexible configuration and work splitting. The modules may be organized to one CMU or spread over two or more CMUs. The organization is given by RTU560 system rules and performance capability of the CMUs. RTUtil 560 supports this concept for the engineering task in that way, that the modules will be directed to a CMU by selection.

The RTU560 internal communication is not limited to one CMU it is extended between the CMUs via the RTU560 system bus. The system bus operates in a producer-consumer principle. A message is put on the internal communication and therefore then also on the system bus by the producer and every module receives this message, analyze it and can then decide is it needed for its task or not. Control mechanism monitor the traffic and take care for a correct communication etc.

The above picture about the RTU560 organization shows the modularity by connecting the IEDs, IO modules and the NCCs to different CMU.

The below picture shows a configuration in modules which are needed for a typical medium RTU560 with one CMU (here 560SLI02).

The main modules are:

- HCI — Host Communication Interface
- SCI — Sub-device Communication Interface
- PDP — Process Data Processing
  (includes communication via the RTU560 serial peripheral bus with the IO boards)
- PLC — Programmable Logic Control
- Archive — Process event archive
  Integrated Total Information (ITI) archive
  Analog Measured Value (AMI) archive
- DIST. Archive — Disturbance Data Archive
- Load Profile — Archive for Alpha Counter Load Profiles
- Integrated HMI — Integrated **H**uman **M**achine **I**nterface
- Others — all internal and indirect used modules (e.g. Web Browser)

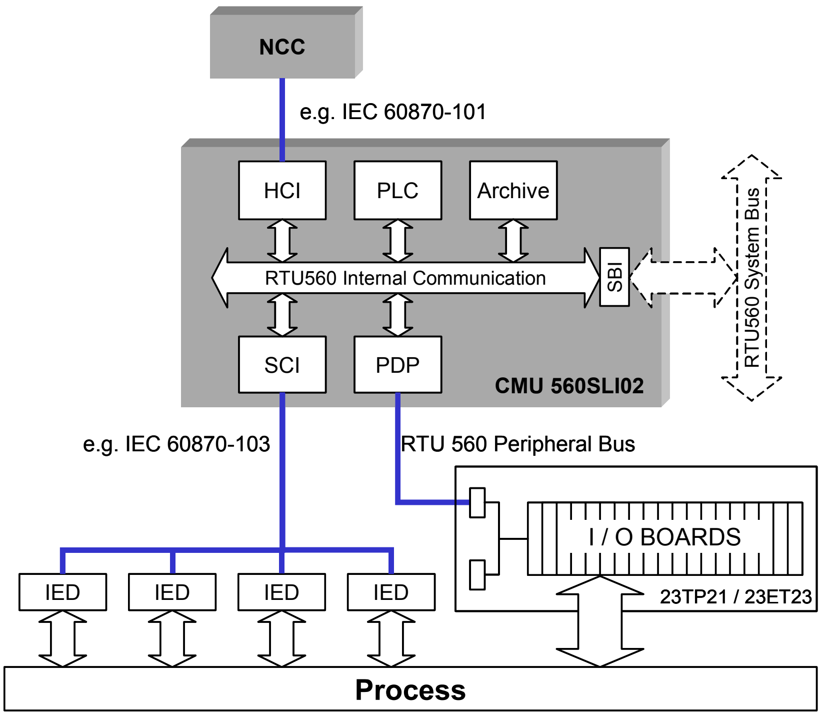


Figure 2-5: Example of communication and application modules within one CMU

Another advantage of the internal communication concept is, that a protocol handled by an HCI, SCI or by the PDP is always converted to the internal presentation of the process data points and vice versa. This allows an easy way to distribute process information to different NCCs etc. with different protocols.

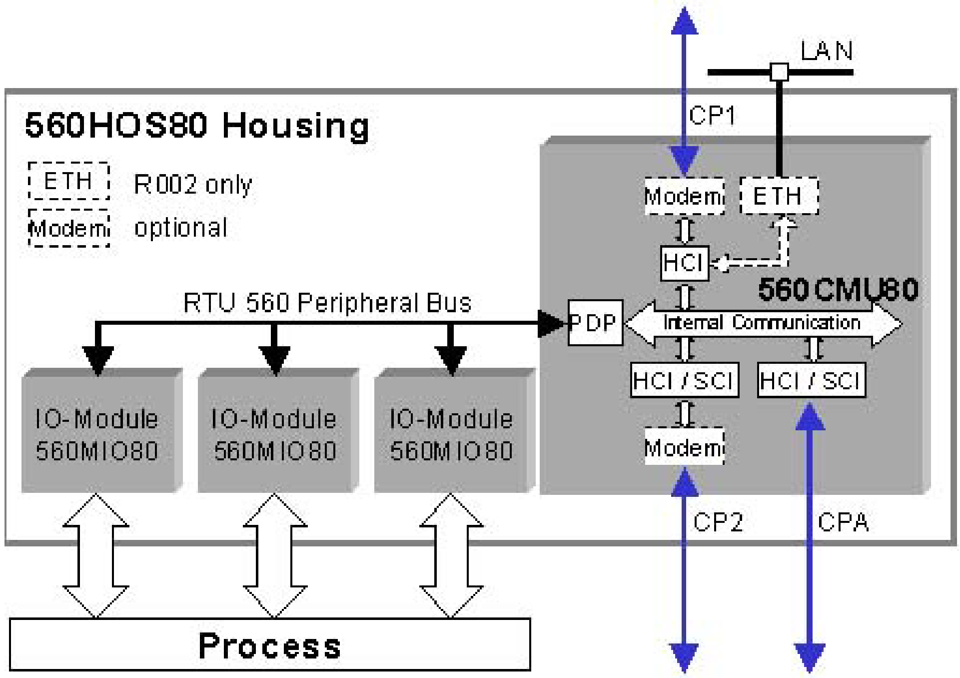
Beside the communication modules are some other application modules of interest for the RTU functionality (PLC, Archives). They have the same flexibility and so it is possible to locate some of them on a separate CMU. In this case is the performance increased because protocol communication is not given. This may be of interest for PLC modules and or archives. In addition this has to be regarded when it comes to redundant CMU solution.

The module concept requires that each module has its own task specific process data base. The process DB contains all information about the process data points relevant for the task of the module.

2.3.2. Communication and Modules for the RTU560E

The basic concept and the modules as such are the same for the RTU560E. The HCI, SCI and PDP modules are used in all RTU560 types.

Differences are given by the configuration concepts. The RTU560E has only one CMU



board with a defined number of communication possibilities.

Figure 2-6: RTU560E module organization on the 560CMU80

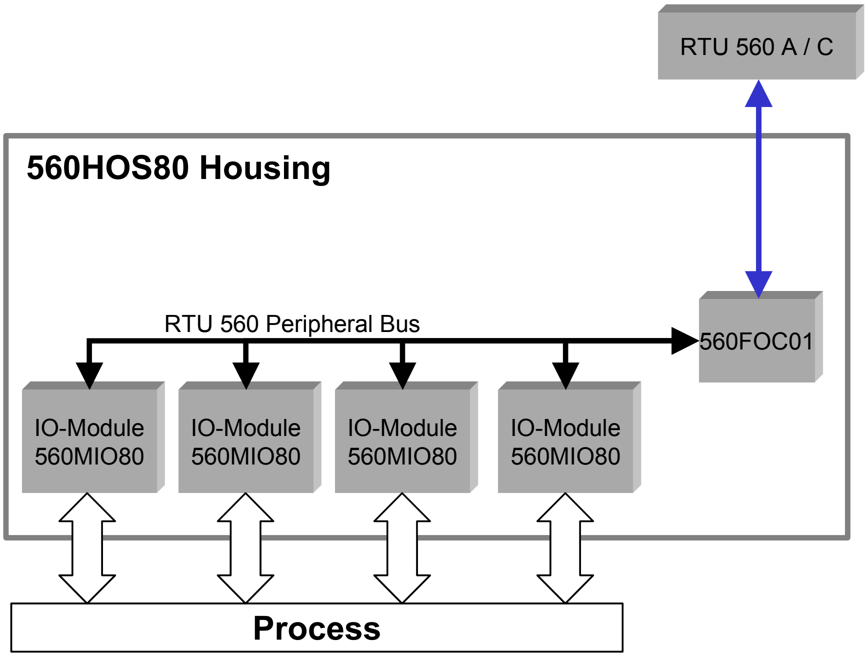


Figure 2-7: RTU560E IO Box with IO-Modules

The RTU560E R001 has either 3 serial communication ports where two are driven by a UART and one by a serial communication controller (SCI).

The RTU560E R002 has an additional Ethernet interface to run LAN protocols like IEC 60870-5-104.

The extended RTU560E IO box has up to four IO modules but no CMU module. It is possible to configure up to six additional RTU560E IO-boxes to one RTU560E with CMU. The RTU560E IO-boxes can also be connected to an RTU560A / C.

2.4. Application functions

The RTU560 supports a number of application functions which are requested for typical remote substations. The module concept gives the possibility to configure these functions on request and cost optimized.

2.4.1. Telecontrol functions

The main task of an RTU is the telecontrol task. Based on the experience with former RTU systems and updated by new requirements the RTU560 supports all the main functions that are needed for the process signals.

Binary Signals

- Binary signal acquisition with a time resolution of 1 ms
- Event detection and time stamping
- Digital filter for signal bouncing
- Chatter suppression of unstable signals
- Signal inversion
- Calculating group alarms (AND, OR) with a time stamp of the signal forcing the group alarm (no PLC program)
- Monitoring double indications and double commands
- Integrated totals with up to 125 Hz
- Integrated totals with up to 8 kHz by pre-divider and for continuous counting
- RTU560A / C supports (1-out-of-n)-check for interposing relays of output commands

Analog Signals

- Zero dead-band supervision
- Live-zero monitoring (4 … 20 mA)
- Smoothing
- Threshold monitoring with integration method
- Threshold monitoring with absolute threshold

2.4.2. General functions

- Support of different time synchronization methods
  by GPS receiver \ not
  by DCF77 receiver / RTU560E
  by telecontrol protocol (e.g. IEC 60870-5-101)
  by external minute pulse (e.g. from a master clock within the station)
- Distribution of the RTU560 produced minute pulse to other RTU560 units or IEDs etc.

2.4.3. Programmable Logic Control (PLC)

The PLC module has access to the controlling process via its process interface imaged in the RTU560 process DB actualized by the internal communication. That allows to use nearly all process information from direct connected process signals as well as from process data points received via serial communication line. Control information for actuators to the process will be handled in the same way from the PLC to the physical output signals etc. The overall transaction time for a PLC task is therefore given by the PLC cycle time plus the update time between the process actuators and sensors and the PLC's RTU560 process DB.

Programming of the PLC program is done by a specific PLC programming tool Multiprog WT. The integration of the PLC task and the link between the IO interface of the PLC to the real process signals is supported by RTUtil 560 together with the PLC programming tool.

The PLC programming is done on the basis of IEC1131-3 libraries. More then one PLC task can be active. An advantage of the module concept is, that the RTU560 allows to have more then one PLC module in the RTU running.

2.4.4. Archive and Local Print Function

Beside the queuing of events in the specific protocol queues it is possible to store important information for a longer period and to access this information at a later time on demand.

The RTU560 supports this function with a local archive which is organized in the Flash memory. For special purposes it is also possible to print this events on a local printer, connected to the RTU560.

The following data types may be stored in the archive.

- Process events with time stamp
- Process commands
- RTU560 system events and system messages
- Login / logout to the Integrated HMI
- Analog measured values
- Integrated totals

2.4.5. *Disturbance Data Archive*

Protection relays and combined bay control and protection units (e.g. the ABB REF 54x units) have the capability to store information about a trip in a file. These files will be kept in the relay until a new file overwrites it or it is transmitted to an external unit.

The RTU560 supports the user with an automatic reading of the disturbance files out of the relays, when the relay indicates a new disturbance file. The file is stored in the RTU560 disturbance file archive. The reserved Flash memory can be configured and may be in the range of up to 128 MB. The RTU560 handles a disturbance file directory to store the files per protection relay. The total number of files per relay is fixed (typ. 8 files) and is given by the total number of relays in the station. A new file overwrites the oldest, when the number of files exceed the configured maximum number.

The RTU560 informs the NCC about new files. Transmission to the e.g. protection engineers workplace (PC with FTP facility) is done via the separate link by Intranet using the RTU560 Web Browser. This solution is taken to prevent an overload on the telecontrol line with a file transfer of several k byte data.

RTU560 support disturbance file reading via IEC60870-5-103 protocol from protection relays and from the REF542+ control and protection unit via SPA bus protocol.

When the file is stored on the protection engineers workplace it is automatically converted into Comtrade format. The Comtrade format is a de-facto standard and most of the disturbance file analyzes programs (e.g. ABB WinEve) can read these files.

2.4.6. *Load Profile Archive*

Meter Data (Load profiles and log-files) of alpha meters, which are generated in the devices, are transferred to the Compact Flash Card of one of the CMU's of a RTU560 system. Using the integrated Webserver of RTU560, it is possible to download these files to a local PC.

The file archive is running on any CMU in the system. In systems with redundant CMU's the file archive must be configured on a non-redundant CMU (group C).

For the storage of load profiles and log files you need at least:

- Communication line with protocol according to IEC 62056-21 (formerly IEC 61107)
- One or more alpha meters (IED) supporting the mode C.
- The file archive function running on a non-redundant CMU of the RTU560

The contents of the transmitted files is according to EDIS (Energy Data Identification System) ((E)DIN 43863 Part 3), including ASCII data and <CR>/<LF>.

2.4.7. *Integrated Human Machine Interface*

The RTU560 function 'Integrated HMI' is an easy possibility to realized small customer specific monitoring and control applications. For this function no additional SCADA product is required (see Figure 2-8).

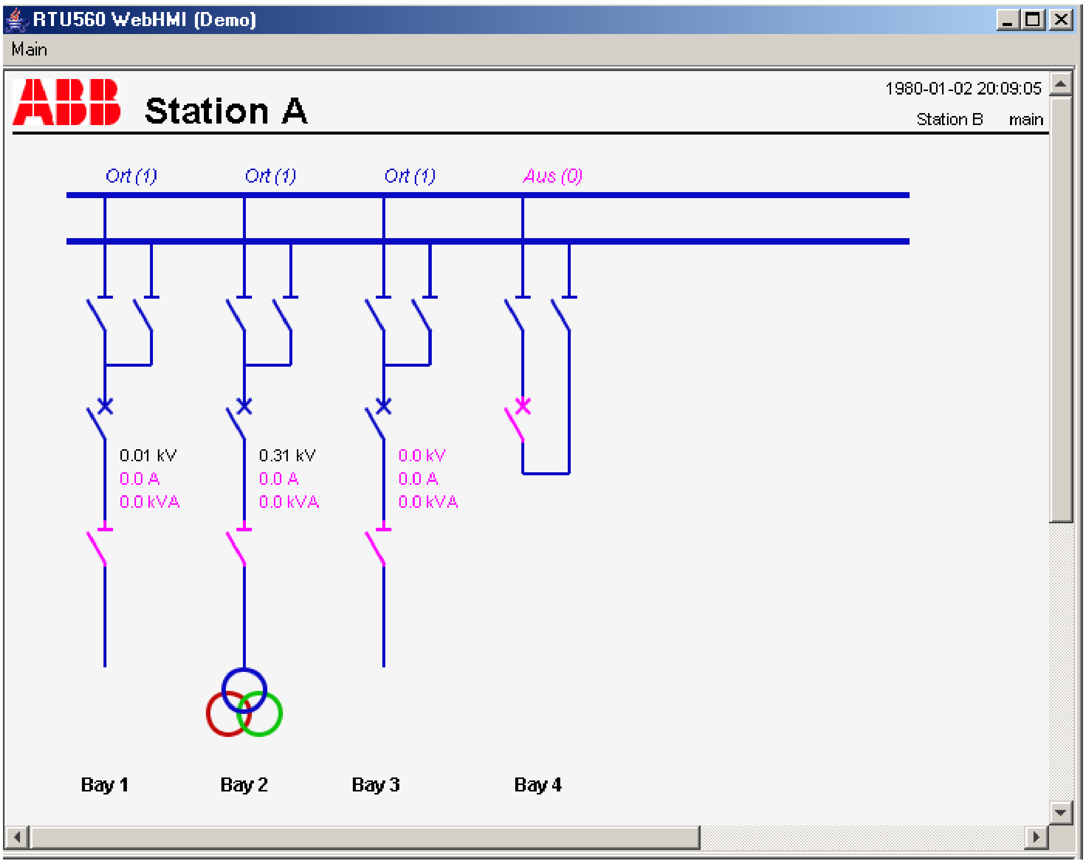


Figure 2-8: Example of the Integrated HMI

2.4.8. *Routing of SPAbus protocol telegrams*

The RTU560 supports the transmission of SPAbus telegrams which are encapsulated in transparent mode in the IEC 60870-5-101 or –104 protocol. A certain data type in the private range of the protocol is defined therefore. In that case the interpretation and handling of the SPA telegram is done by the subordinated SPAbus unit and the SPAbus master which is e.g. a MicroSCADA Pro system of ABB. The RTU560 operates as a passive router.

2.5. RTU560 configuration

The high flexibility in configuration and the modularization requires some configuration rules which have to be regarded for an RTU560.

Most of the rules are independent of the RTU type RTU560A, RTU560C or RTU560E, because the CMU concept and the memory etc. is the same for all three types. The limitation is given by the hardware possibilities.

2.5.1. Process IO board configuration RTU 560A / C

The RTU560 allows to split the total IO boards on maximum four RTU Peripheral Bus segments. Each PB-segment is handled by a PDP module which does the complete process data handling for the configured process data points. The advantage to split the IO signals to more PB-segments is given by the transaction time for events etc., because the PDP modules operates in parallel.

- Number of PDP modules per RTU560: 4
- Number of PB segments per PDP 2
- Number of PB-segments per RTU560: 4
- Number of IO sub-racks per PB-segment: 6
- Number of IO-boards per sub-rack 19
- Theoretical total number of sub-racks: 24
- Theoretical number of IO-boards: 24 x 19 = 456

2.5.2. Process IO board configuration RTU560E

The principles are the same.

- Number of PDP modules per RTU560E: 2
- Number of PB-segments per RTU560E: 2 (1 internal / 1 on CP A)
- Number of RTU560E IO-housings on PB-segment 2: 6
- Number of IO modules RTU560MIO80 per RTU560E housing: 3/4
- Theoretical total number of IO-housings: 6
- Theoretical number of 560MIO80 modules: 6 x 4 + 3 =27

2.5.3. Total number of process data points per RTU560A / C / E

The total number of process signal dada points per RTU560 process data base on HCI, SCI or PDP module is limited to

3000 Process Data Points (DP)

Process Data Points are:

- all direct connected wired process signals (IO boards)
- all virtual data points as a result of the PLC programs
- all process signals configured on all communication lines per RTU560 to be handled in the RTU560

2.5.4. Total number of communication lines and CMUs per RTU560A

Given by the hardware concept it is not possible to reach the maximum values on a RTU560C and RTU560E.

The RTU560 has to build up a management per communication link and user on that line. This results in a limitation definition which is less then the theoretical values.

- Number of communication sub-racks 560CSR01: 2
- Number of CMUs per 560CSR01: 4
  (limitation by power supply)
- Total number of CMUs per RTU560A: 8
- Total number of serial lines per RTU560A: 4 x 8 = 32
- Maximum number per RTU560A system: 30
- Maximum number of IED’s on a comm. line: 32
- Theoretical number of IED’s per RTU560A: 30 x 32 = 960
- Practical limitation per RTU560A system: approx. 100 … 120
  there is no fix limit because this is influenced by the total number of IO points per IED, needed RAM etc.
- Maximum number of host systems 8

2.5.5. Definitions per CMU

- Number of protocols per CMU (HCI, SCI) 2
  valid for 560SLI02, 560ETH02 / 03, 560CMU80
- Number of serial lines per 560SLI02: 4
- Number of serial lines per 560ETH02 / 03: 2 (SLC type)
- Organization of the Line interfaces (UART, SLC) to the HCI/SCI modules is given:
  - by the electrical interface request (RS232C, RS422, RS485)
  - by the protocol type and need of SLC, e.g. RTU Peripheral Bus
  - by the number of lines needed

Remark:

- When a PDP module is configured on a CMU, the port A and B are reserved for the RTU560 Peripherals bus only. The SCI is then responsible for the communication on the RTU560 PB.

2.5.6. Definitions for PLC

- 1000 Boolean instruction lines: 10 ms
- 1000 BOOL8 and INT instruction lines: 10 ms
- Shortest cycle period configurable standard: 10 ms
- Program memory capacity per POU configurable
  Approx. 1000 instruction need: 10 kByte RAM,
- In one POU 64 kByte program code can be executed.
- I/O image capacity configurable: Max. 1000 input signals and 1000 output signals
- Amount of user tasks: 15 task

2.5.7. Specification of the RTU560 Peripheral Bus

The RTU560 PB is defined by the RTU 560 peripheral bus protocol.

- Electrical interface: RS 485
- maximum length: approx. 1000 m
  according to the RS485 specification for EMC, equipotential bonding, etc.

It is recommended to install fiber optic cable in all cases where the conditions are unknown or critical. The 23OK22 board for the RTU560A / C and the 560FOC80 module for the RTU560E should be used therefore.

2.5.8. Definitions Integrated HMI

Property	Limits
System Requirements:	
Operating System	Microsoft Windows 2000 or XP
Java	Java Virtual Machine 1.4 or higher
Integrated HMI:	
Max. number of pages	20
Max. number of data points per page	200
Max. number of active clients	4
Refresh of current page	< 1 second
Update time for a new page	< 2 second This time may become longer at first update, if a large bitmap is used.
Execution time for a command	< 3 seconds dependant on the actual system load
Alarm List	
Max. number of data points, to be configured for the alarm list.	500
Number of alarm lists	1
Number of alarm classes	10
Refresh	By operator request. A blinking symbol will signal a new alarm
Update time	< 5 seconds
Event List	
Max. number of data points, to be configured for the event list	500
Max. number of entries	Configurable: 250 ... 10.000
Number of event lists	1
Refresh	By operator request A blinking symbol will signal a new event
Update time	< 5 seconds

3. Communication

3.1. General

The communication of the RTU560 family may be divided into two parts

- Internal communication with a neutral process data point presentation
- External communication to NCCs, IEDs, Sub-RTUs, etc.

3.2. Internal Communication

The internal communication has to handle:

- process data point information
- organizational information
- RTU560 system information

Information form external connected communication lines with their protocols will be converted into an internal presentation by the HCI and SCI modules. The internal protocol is based on the IEC60870-5-101 data presentation.

Once a process data point information from external protocol as well from the RTU560 IO-boards coming via the RTU560 Peripheral bus is converted into the internal data presentation, it is used by all other modules. Each HCI or SCI module converts only from internal presentation to external presentation and vice versa.

For the standard telecontrol data types the equivalent internal data types are listed. The link to the external protocol specific type is described in the corresponding protocol document.

Process data signal type	Mnemonic	RTU560		
		A	C	E
Binary input signals				
Single indication	SPI	X	X	X
Double indication	DPI	X	X	X
Digital measured value (8, 16 bit)	DMI08/16	X	X	X
Step position	STI	X	X	X
Integrated totals	ITI	X	X	X
Bitstring input (8, 16 bit)	BSI08/16	X	X	X
Binary output signals				
Single command	SCO	X	X	X
Double command	DCO	X	X	X
Regulation step command	RCO	X	X	X
Digital Setpoint command (8 bit)	DSO08	X	X	X
Digital Setpoint command (16 bit)	DSO16	X	X	
Bitstring output (1, 2, 8 bit)	BSO01/02/08	X	X	X
Bitstring output (16 bit)	BSO16	X	X	
Analog input signals				
Analog input values	AMI	X	X	X
Analog output signals				
Setpoint command	ASO	X	X	

3.3. Telcontrol Protocols

The telecontrol protocols can be defined into two groups.

- Host communication interfaces (HCI)
  protocols between an Network Control Center (NCC), Substation Control System (SCS) (e.g. MicroSCADA Pro of ABB) and the RTU. In this case the NCC is master and the RTU560 is Slave.
- Sub-device communication interfaces (SCI)
  protocols between the RTU560 and a subordinated unit like an IED in the stations itself or another subordinated RTU unit (eg. a station with an RTU560E). In this case the RTU560 is master and the sub-device is slave.

The RTU560 CMU boards are foreseen to run the HCI and/or the SCI module. As listed in chapter "RTU configuration" a CMU can run two of these modules HCI and/or SCI.

The below picture shows the principle structure of an HCI or SCI module. It consist mainly of three parts:

- the protocol handler: responsible to run the protocol and to manage the communication to the host or sub-device.
- the process data base: which contains the process signal image needed between the two partners RTU560 and the host / sub-device
- the line driver: responsible for the lowest protocol level and the physical conversion to the serial line or Ethernet.

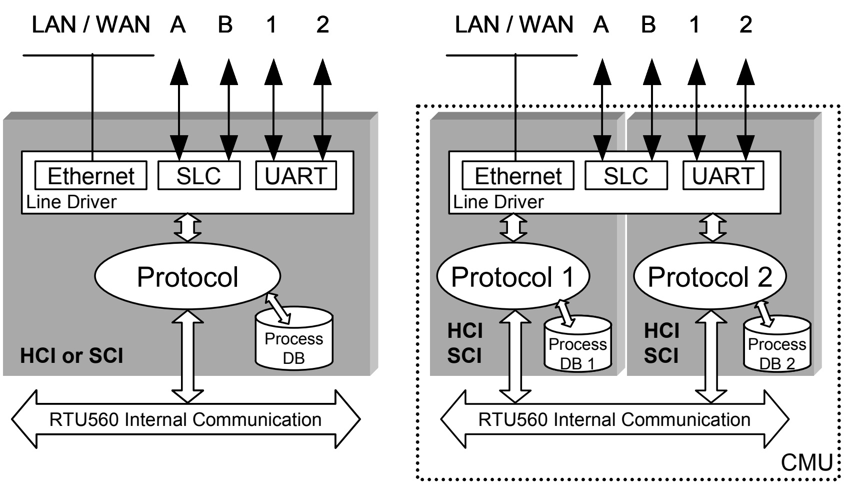


Figure 3-1.: HCI, SCI structure

The link between the protocol handler and the line driver is done by configuration (RTUtil 560). Given by the physical interface possibilities etc. the number of lines coordinated by one protocol can be between 1 and 4 serial lines (1 or 2 Ethernet lines).

Host communication (HCI) interfaces towards the NCC or router RTU

- IEC 60870-5-101 slave
- IEC 60870-5-104 slave
- DNP 3.0 serial slave
- DNP 3.0 Ethernet slave
- RP 570/571 slave
- Indactic 33 slave
- Indactic 35 slave
- Modbus (ASCII, RTU) slave
- TG 809 slave
- Conitel 300 slave
- Sinaut 8-FW slave

Subdevice communication (SCI) Interfaces to subordinated devices

• IEC 60870-5-101	master
• IEC 60870-5-102	master
• IEC 60870-5-103	master
• IEC 60870-5-104	master
• IEC 62056-21	master
• DNP 3.0 serial	master
• RP 570/571	master
• SPA Bus	master
• Modbus (ASCII, RTU)	master
• Indactic 21	master

Telecontrol protocol profile

The various telecontrol protocols have different data presentation and also different lists of supported process data points. The attributes and definitions for the protocol elements may also be different.

For each used protocol within the RTU560 is the conversion described and the interpretation for the RTU560 listed in the protocol document. These documents and their contents are the basis for the conversion. Please refer to the protocol documents for more details.

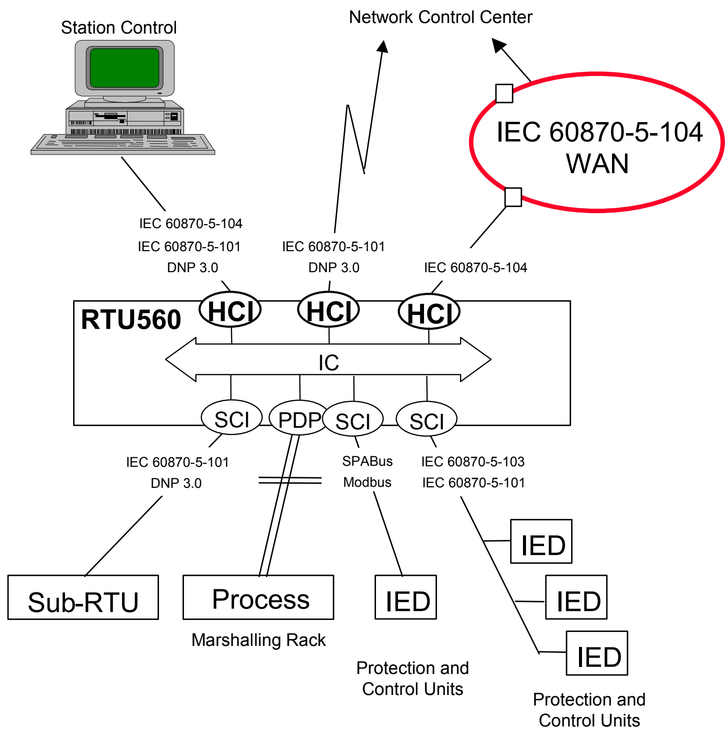


Figure 3-2: Example of an RTU560 Network with various protocols

Beside the above listed protocols supports the RTU560 additional telecontrol protocols which were developed for dedicated projects and or for a standard protocol in a country.

3.4. Host communication interfaces

The RTU560A / C allows the communication with up to eight different NCCs by using a number of CMU modules. The RTU560E allows the communication with up to two different NCC's by using the serial interfaces of the 560CMU80 R0001, or up to eight different NCC's when they have access to the Ethernet interface of the 560CMU80 R0002. The configuration of the interfaces with their protocols is completely done in RTUtil 560. There are no hardware switches to configure the interfaces.

For detailed information and protocol specific restrictions see the protocol description.

The assignment of UART host protocols to the serial interfaces is totally free. There are no dependencies between the different protocols on one CMU.

Protocols not using UART are restricted to the interfaces CPA (and CPB) on the CMU 560SLI02, 560ETH02/03 and 560CMU80 of RTU560E.

Ethernet and TCP/IP based protocols could only be used with the Ethernet interface on the CMUs 560ETH02/03 or 560CMU80 R0002.

3.4.1. Monitoring Direction

The process information needed for an NCC or SCS is engineered by RTUtil 560. It allows to select the information which is relevant for the host. The RTU560 supports a general selection by process signal filters, which are possible by the addressing model in the project. E.g. all information for the transmission grid will be routed to the responsible NCC by deselecting a filter. All information of the distribution network may be sent to another NCC by deselecting the filter.

In Monitoring Direction the following RTU560 information types are translated into protocol specific telegrams:

- SPI Single Point Information
- DPI Double Point Information
- STI Step Position Information
- BSI Bit String Information
- ITI Integrated Totals Information
- DMI Digital Measured Information
- AMI Analog Measured Information

Process data point attributes or information qualifier (e.g. Blocked, Substituted, Not Topical, Invalid) will be translated into the host protocol when they can be imaged to the host protocol and they may be handled either in the RTU560 itself or received from a sub-device which support the attribute / qualifier. For details see the protocol description.

3.4.2. Command Direction

NCCs or SCS as host can send commands to the RTU560. The HCI checks the command as much as possible and distributes it via the internal communication to all other modules. The responsible module (e.g. the PDP) checks the command for formal correctness and can acknowledge the command for the host. The acknowledge is send to the host by the HCI module.

A command addressed to a sub-device is routed by the SCI module to the addressed sub-device, which has to acknowledge the command. This method secures, that the unit responsible for the execution of the command confirms, that it is possible to do the command task.

The RTU560 allows to run several commands in parallel, when they are handled by different HCI and or SCI modules.

In Command Direction the protocol specific telegrams are translated into the following RTU560 types:

- SCO Single Command Output
- DCO Double Command Output
- ASO Analog Setpoint Output (RTU560E: to Sub-device Interfaces only)
- DSO Digital Setpoint Output
- BSO Bitstring Output (RTU560E: to Sub-device Interfaces only)

3.4.3. General Interrogation

A general interrogation command from the host is answered by the HCI directly. This is possible, because the HCI has the complete and valid process signal image in its process DB.

All SCI and PDP modules are responsible to force a complete process signal image after start up or when a sub-device fails becomes online again. The HCI are informed about any changes by internal communication. This method secures, that the process data bases in all modules is kept actual.

3.5. Sub-Device Communication Interface

The SCI modules support various communication protocols. The protocol specific configuration parameters are described in the coresponding protocol documents.

The configuration of the SCI and the communication lines with their protocols is completely done in RTUtil 560. There are no hardware switches to configure the interfaces.

The SCI can manage up to 32 devices per communicatin line. In a RTU560 it is possible to have up to 30 sub-lines. See chapter "RTU560 configuration".

For detailed information and protocol specific restrictions see the protocol description.

The assignment of UART sub-protocols to the serial interfaces is totally free. There are no dependencies between the different protocols on one CMU. The only restriction is the number of communication protocols offered in one firmware package. Not all existing protocols can be combined on one CMU board. Only certain combinations of protocols are possible.

Protocols not using UART are restricted to the interfaces CPA (and CPB) on the CMU 560SLI02, 560ETH02/03 and 560CMU80.

Ethernet and TCP/IP based protocols can only be used with the Ethernet interface on the 560ETH02/03 CMU or 560CMU80 R0002 CMU.

3.5.1. Monitoring Direction

In Monitoring Direction the protocol specific telegrams are translated into the following RTU560 types.

- SPI Single Point Information
- DPI Double Point Information
- STI Step Position Information
- BSI Bit String Information
- ITI Integrated Totals Information
- DMI Digital Measured Information
- AMI Analog Measured Information

Process data point attributes or information qualifier (e.g. Blocked, Substituted, Not Topical, Invalid) will be translated into the sub-device protocol or internal protocol when they can be imaged to the sub-device protocol and they may be handled either in the RTU560 itself or received from a sub-device which support the attribute / qualifier. For details see the protocol description.

3.5.2. Command Direction

The SCI handles commands distributed via internal communication which are addressed to a sub-device, connected to one of the communication lines managed by the SCI. This commands can come from a host or from a PLC module.

In Command Direction the following RTU560 command types are translated into protocol specific telegrams:

- SCO Single Command Output
- DCO Double Command Output
- RCO Regulation Step Command
- ASO Analog Setpoint Output
- DSO Digital Setpoint Output
- BSO Bitstring Output

3.5.3. General Interrogation

The SCI will generate a General Interrogation Command for all connected devices after startup, or if an IED becomes normal again. Via internal communication all other modules can update their process data base. This ensures, that all modules have the actual complete process signal image.

3.6. Redundant Communication

The RTU560 has the possibility to handle some versions of redundant communication.

- Redundant line
- Multi host (in the sense of redundant communication)
- Both for serial lines or Ethernet connections

The redundant communication possibilities are independent from the redundant CMU concept, which is described in chapter 6.

In the actual RTU560 release only redundant communication to hosts (HCI modules) is supported.

3.6.1. Redundant Line

Redundant line on 560SLI02 (serial lines)

Redundant line on serial lines is only supported for telecontrol protocol IEC 60870-5-101, with the following conditions:

- unbalanced and balanced mode
- both redundant line must be connected to the same CMU
- Time synchronization via communication line is possible an both lines when configured.
- Dialed line modems are not supported
- Only fixed line modems 23WT2x for RTU560A / C or 560MOD80 for RTU560E

The following conditions are valid for the host:

- the host is responsible for a switchover
- supervision of the not used line has to be done by the host
- the redundant line function has to be handled according to the "Norwegian User Convention"

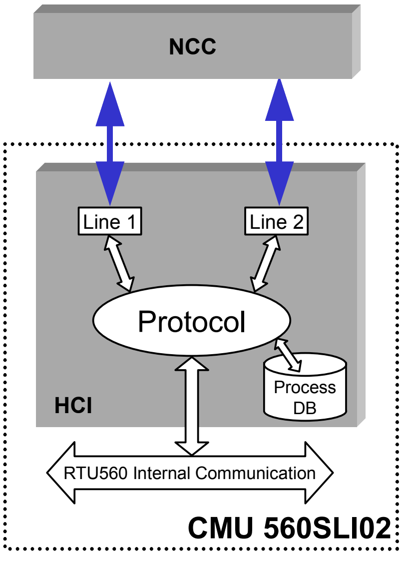


Figure 3-3: Redundant Line stratetgy for serial communication

The HCI module is managing both lines in a "parallel mode". That means it accepts telegrams from the host on both lines and responses with the answer on the same line. This implies, that the host has to control the two lines and to switch over, if the actual active line fails.

Redundant line on 560ETH03 R002 (Ethernet)

The 560ETH03 R002 has two Ethernet interfaces which enables the possibility for a redundant LAN interface.

The conditions for a redundant host communication on LAN are:

- only possible with IEC 60870-5-104 protocol
- configuration of up to 8 hosts possible in redundant mode
- all redundant host links have to be on the same 560ETH03
- for network configuration with different IP subnets

The following conditions are valid for the host:

- the host is responsible for a link switchover
- a general interrogation is needed after a link switchover
- line check for the inactive link is possible by open and close of the TCP/IP link

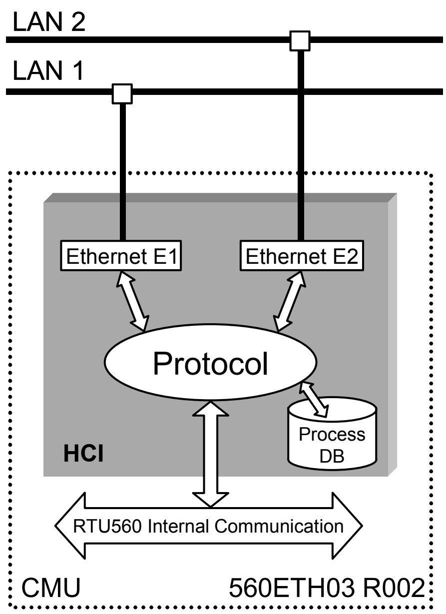


Figure 3-4. Redundant line (link) on Ethernet

3.6.2. Multi Host communication

Multi host for a redundant communication means, that two independent HCI modules are installed which operates independent of each other. But the communication is to the same host and with the same process data configuration. That means both HCI have a 100% identical process DB.

From the viewpoint of the host it has two communication links to one RTU560 station. The host has also here to manage the use of the two possible links. On the RTU560 both line are starting independently and waiting for a communication start of the host. When a host starts the communication it will receive all information in a sequence given by the protocol specifics. The host should be able to manage process information received on two communication links with the same contents.

The conditions for a multi host are:

- possible with all RTU560 HCI protocols
- configuration within the RTU560 as two independent hosts, but with the same process data point configuration
- the two HCI modules can be on the same or on different CMU boards
- fixed line modems 23WT2x for RTU560A / C or 560MOD80 for RTU560E possible
- Dialed line modems according the reference list for modems and protocols where the HCI supports dial up functionality
- Time synchronization possible from "both" hosts possible, when synchronization via communication line is allowed and configured.

The following conditions are valid for the host:

- the host is responsible for a switchover
- the host has to force a general interrogation after a switchover
- the must accept and can handle retransmission of process information and events, as part of the GI and new events during start up.
- The check of the inactive line depends on the protocol

Multi Host on serial line

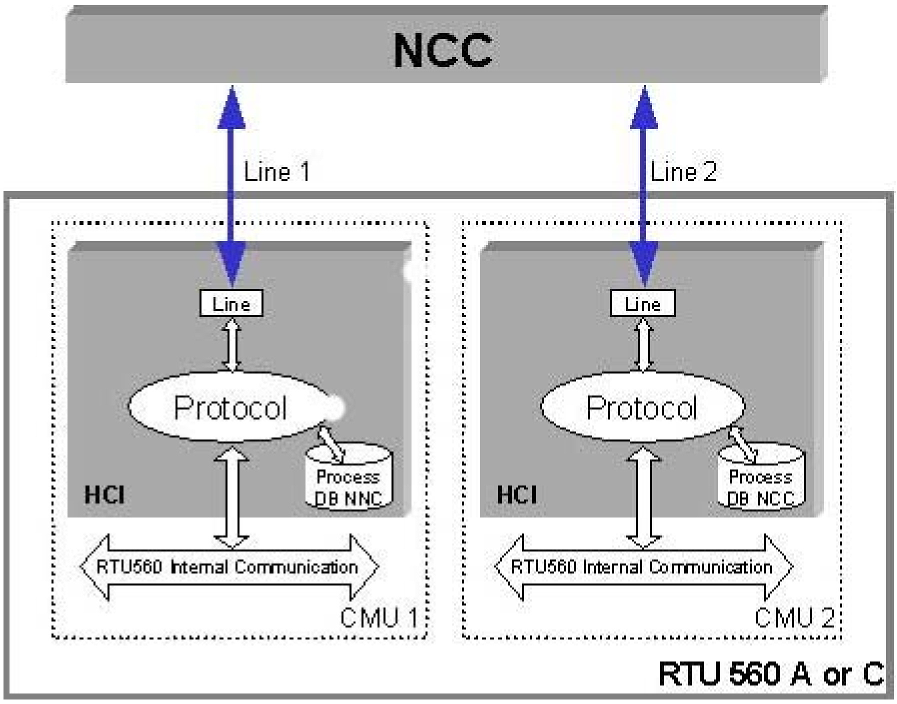


Figure 3-5: Multi host on serial lines and CMU 560SLI02

Multi Host on Ethernet

The conditions for multi host on Ethernet are:

- possible with IEC 60870-5-104 or DNP 3.0 over LAN / WAN
- configuration within the RTU560 as two independent hosts, but with the same process data point configuration
- the two HCI modules can be on the same (with 560ETH03 R002) or on different CMU boards

The conditions for the host on Ethernet are:

- the host is responsible for a link switchover
- the host has to force a general interrogation after a link switchover
- the must accept and can handle retransmission of process information and events, as part of the GI and new events during start up.
- line check for the inactive link is possible by open and close of the TCP/IP link

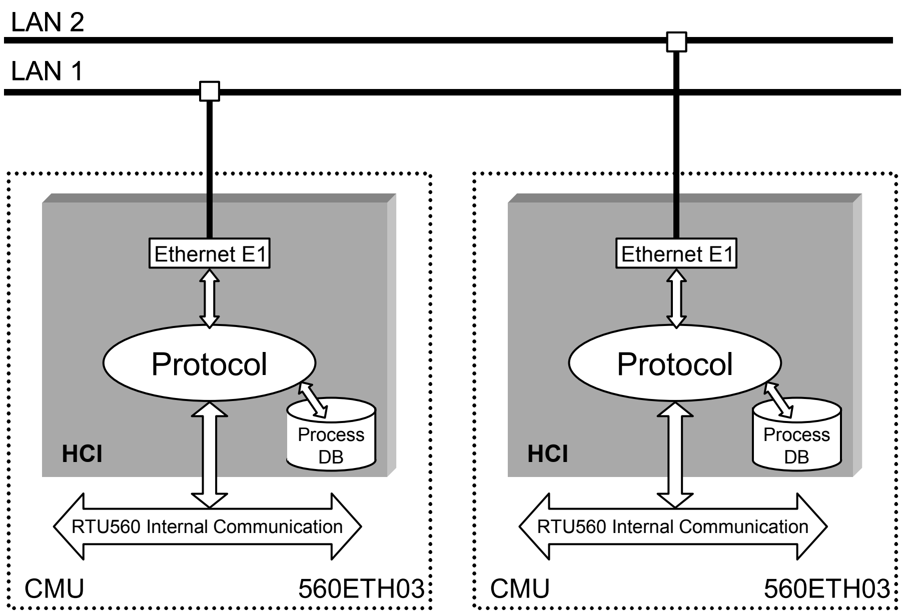


Figure 3-6: Multi host on Ethernet

4. RTUtil 560

4.1. Basics

RTUtil 560 is the configuration and engineering tool for the whole RTU560 family.

The basic topics of RTUtil 560 are:

- Configuration and data engineering tool for RTU560 projects
- The user interface is structured according to the principles of IEC 1346-1
- Documentation of all important project steps
- External data interface concept (Microsoft EXCEL import)
- Multilingual tool (user interface and help files)
- Delivery version with setup, installation and de-installation program on CD-ROM
- MS Windows NT 4.0 / 2000 / XP Professional platform
- The User Interface of RTUtil 560 is an application based on the Microsoft standard presentation format
- One tool for RTU560A, C and E

4.2. System requirements

The Hardware requirements for the data engineering tool RTUtil 560, particularly the free disc space, depends on the project size. Basic requirements are:

- PC based
- Operating system: MS Windows NT 4.0 / 2000 / XP Professional
- Memory: 128 MB RAM
- Processor: Pentium class > 500 MHz
- Hard disc: > 200MB free disc space
- Hard lock: Dongle

4.3. RTUtil 560 Structure

RTUtil 560 enables the user to control the whole engineering process of an RTU based system. All configuration data is managed by RTUtil 560.

To meet the requirements the internal software structure of RTUtil 560 is split up into different function parts as shown in the figure below.

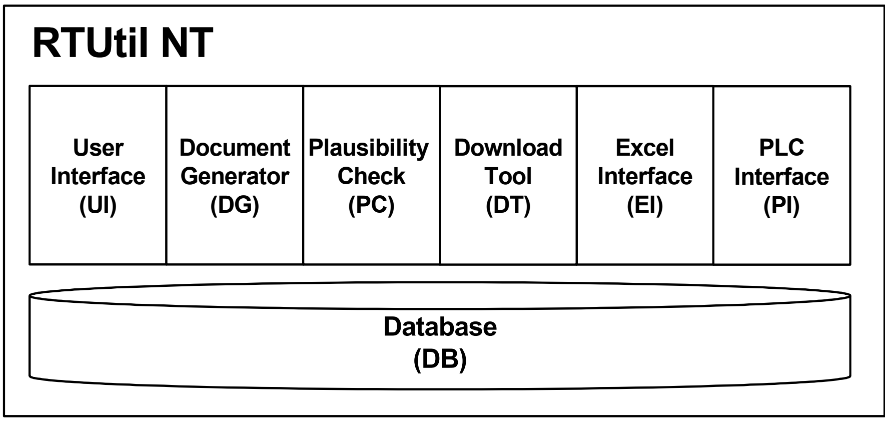


Figure 4-1: Overview RTUtil 560 components

4.4. General Data Structuring and View

The general view of the user on the engineering data is implemented on the basis of the international Standard IEC 1346-1. This standard describes the structuring principles and reference designations for industrial systems, installations and equipment.

In the user interface this standard is presented in trees showing the RTU from different points of view. To describe the whole process the view is split up into three trees. IEC 1346-1 defines how to split up a system (function-, product- and location-oriented structure).

The user interface structure offers three trees to describe the system structure used for an RTU560A, C and E.

- NetworkTree

  The NetworkTree shows the lines and protocols between the RTUs and/or NCCs for routing the data points through the network.

- SignalTree

  In the SignalTree the location and designation of signals are shown. The signal location describes the place of the data points in the primary process.

- HardwareTree

  The HardwareTree presents the structure of an RTU560A, C or E with the levels:

  Cabinet
  Rack,
  Board

  and the reference to the data points defined in SignalTree.

The structuring in trees provides a common presentation format and a general user interface of the RTU data.

4.5. General Tree Structure

The following example shows the implementation of the structuring principles according to IEC 1346-1. In the picture *Network - RTU - Process* the project views down to the RTU internal configuration are shown. This is only a small example of an RTU network. *RTU 01* is the concentrator station in this network. The right side of the picture shows the electrical process (one bay in a station, with the double point indication Q0). This scenario will be built up in the three trees: NetworkTree, SignalTree, HardwareTree.

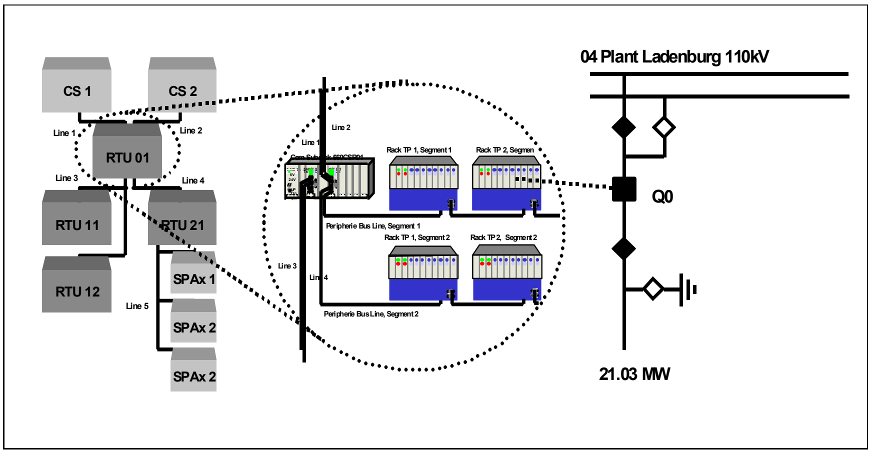


Figure 4-2: Network - RTU - Process

The root node of a tree is similar to the tree type (e. g. SignalTree, HardwareTree).

The NetworkTree is the representation of the network structure. The concentrator RTU 01 is chosen as root node in the NetworkTree because the concentrator RTU is normally a unique starting point in an RTU station network. The hosts (central systems) are represented below the root RTU.

In the SignalTree the electrical process is structured and the names of the single point indications are defined. The name for every data point is derived from the structure of the electrical process.

Some nodes have different locations of presentation (references) in several trees. These nodes are the line and station nodes from the NetworkTree and the data point nodes from the SignalTree. The line and station nodes have references in the HardwareTree and in the NetworkTree. The data point nodes have references in the HardwareTree and in the SignalTree.

The HardwareTree describes the internal structure of the station nodes (e. g. RTU 01). The RTU and Line node types in the HardwareTree are linked nodes from the NetworkTree. The double point indication Q0 in the electrical process in the SignalTree shall be linked with a binary input board in the HardwareTree.

5. Telecontrol Functions

5.1. General

Note: The following Telecontrol Functions are available within RTU560A or C, and also within RTU560E if not otherwise noticed.

As shown in Fig. 2-3 and Fig. 2-5 (refer to chapter 2) the communication units and the I/O boards share the processing of the telecontrol functions. The I/O boards take over the essential tasks of scanning and output of process signals, and the communication unit the communication with the NCCs as well as the organization and management of the process image in the data base. All time critical functions are concentrated on the I/O boards.

The I/O boards transmit process value changes or status changes as events. The I/O bus (IOC) of the communication unit detects and transmits the events to the communication unit (CMU) of the communication unit. To control the data flow, each I/O board has a FIFO buffer for the temporary storage of up to 50 events. All events are time stamped.

Commands to the I/O boards are checked for plausibility etc. and the outputs to the output channels will then be handled by the output boards autonomously.

The CMU handles those telecontrol functions, which overlap the I/O-boards (e.g. (1-out-of-n) check).

The telecontrol functions are divided in:

- Monitoring direction
  Indication processing
  Analog measured value processing
  Digital measured value processing
  Integrated total processing
  Bitstring input value processing
- Command direction
  Object command output
  Regulation command output
  Setpoint message output
  Bitstring output (not RTU560E)

In Figure 5-1 the distribution of the tasks among the I/O boards and the central control unit is shown by means of the example of indication processing.

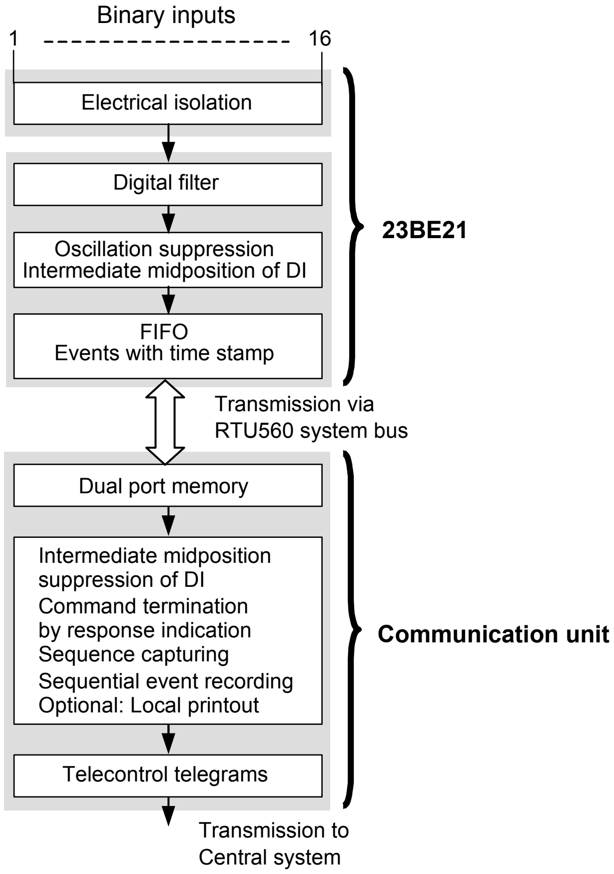


Figure 5-1: The principle of task sharing illustrated by indication processing

5.2. Monitoring Direction

5.2.1. Indication Processing

There are two types of indications:

- Single point input (SPI)
- Double point input (DPI)

Figure 5-2 shows the signal definition for SPI and DPI. Double indications are represented by two sequential bits within a 23BE21/22 or 560MIO80 board. The normal state of a DPI is an antivalent bit combination (10 or 01). The two intermediate positions 11 or 00 are handled different within the RTU560. An intermediate state is given during the runtime of a unit from one position to the other (e.g. an isolator switching from OFF to ON).

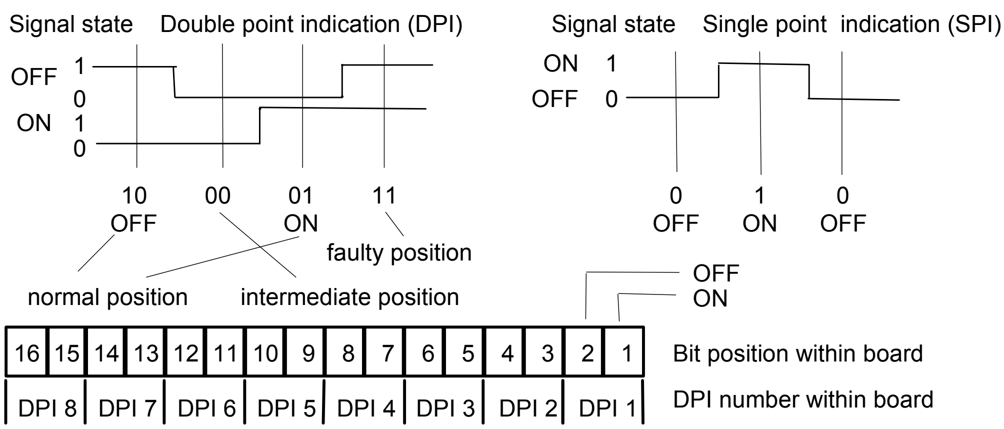


Figure 5-2: Indication type definition

An indication board consisting of max. 16 bit. SPI and DPI can be mixed. But a DPI can start on an odd bit-position only. On 23BE21/22 or 560MIO80 board it is possible to mix any type of binary inputs. E.g. inputs not assigned to DPI or SPI may be configured as pulse counter indications as, digital measured values on bit-string inputs. Digital measured values and bit-string inputs must be configured such to start with bit position 1/9.

Functions

The process data acquisition functions for indications processed by the RTU560 can be split into functions handled by the:

- I/O controller (IOC) of the binary input board 23BE21/22 or 560MIO80
- Process data processing (PDP) part of the CMU
- Protocol specific communication interface part of a CMU

The data processing functions of the communication interface are described in the documentation of the respective communication protocol.

23BE21/22 or 560MIO80 functions:

- Reading input register (every millisecond)
- Digital filter (contact bouncing)
- Oscillation suppression (signal chattering)
- Signal inversion
- Time out monitoring for DPI intermediate position
- Store events in FIFO with time stamp

CMU - PDP:

- Intermediate midpoint position handling for DPI
- Command output response
- Group signals
- Transmission to internal communication

5.2.2. *Analog Measured Value Processing*

Analog Measured Value Types

Each analog value is converted by the analog digital converter (ADC) of the 23AE21 or 560MIO80 board into a signed integer presentation. The presentation is shown in Figure 5-3. The 100 % input signal value is represented by 12 bits plus sign.

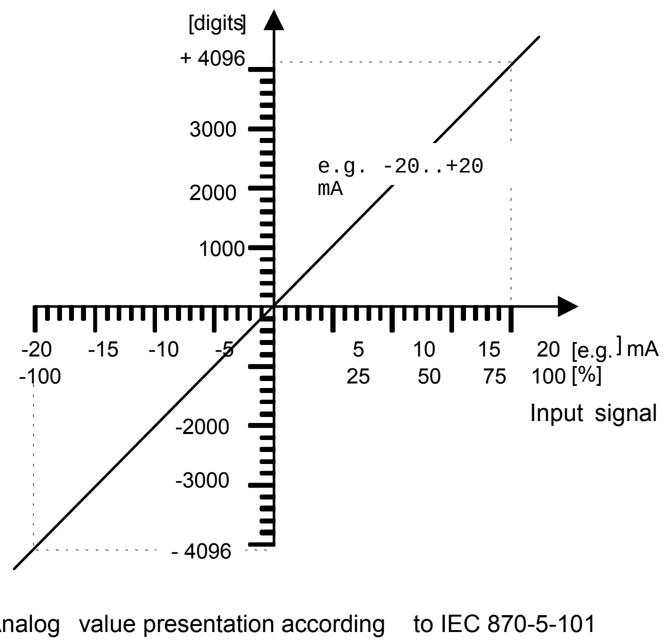


Figure 5-3: Analog value presentation by ADC

The PDP converts the value to a normalized presentation.

AMI Functions

The process data acquisition functions for analog measured values (AMI analog measured value input) processed by the RTU560 can be split into functions handled by:

- IOC of the analog input board 23AE21 or 560MIO80
- Process data processing (PDP) part of the CMU
- Protocol specific communication interface at a CMU

The data processing functions of the communication interface are described in the documentation of the respective communication protocol.

23AE21 or 560MIO80:

Scan analog input cyclically
Zero value supervision and switching detection
Smoothing
Threshold supervision on integrator algorithm
Periodic update of RTU data base
Store events into FIFO with time stamp

CMU - PDP functions:

Unipolar and live zero conversion
Scaling
Threshold supervision on absolute threshold value
Transmission to internal communication

5.2.3. Digital Measured Value Processing

There are two types of digital measured values (DMV):

- Digital measured value (DMI)
- Step position value (STI)

The RTU560 can handle different bit patterns to read them and convert them into a digital measured value :

- 8 bit digital measured value (DMI8)
- 16 bit digital measured value (DMI16)
- 8 bit step position value (STI)

The RTU can handle conversions for:

- binary data (BIN)
- binary coded decimals (BCD)
- Gray code (GRAY)

The maximum length of a digital measured value is the 16-bit word of (= one 23BE21/22 or 560MIO80 board). Double word values (32 bit) are not supported.

Digital measured value presentation

Each type is converted and scaled by the PDP.

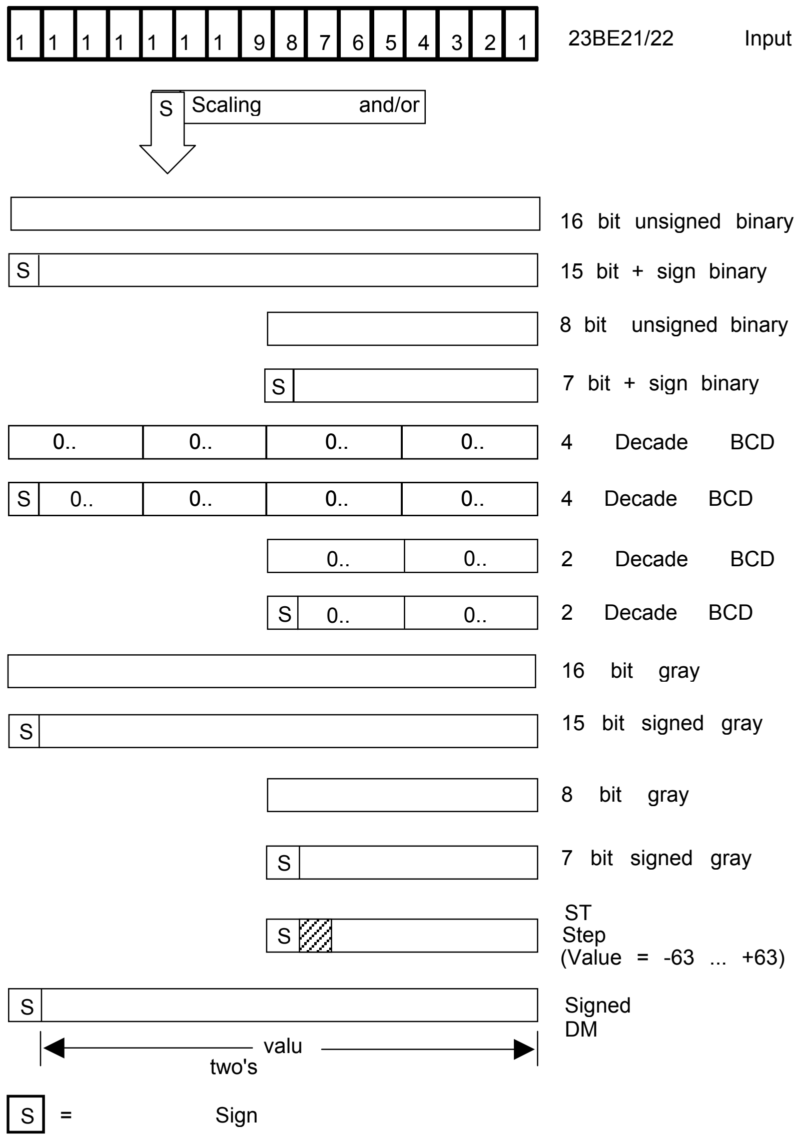


Figure 5-4: Digital Measured Value presentation

An digital measured input value DMI is scaled to a normalized 16 bit value representation (+/- 100 %) For the step position information (STI) of a transformer or Petersen coil the value can only be in the range of -63 ... +63.

If an eight bit pattern is selected the residual 8 bit of the 23BE21/22 or 560MIO80 board can be used for another digital value, for pulse counter values or indications.

DMI Functions

The data acquisition functions for digital measured values (DMI) processed by the RTU560 can be split into functions handled by:

- IOC of the binary input board 23BE21/22 or 560MIO80
- Process data processing (PDP) part of the CMU
- Protocol specific communication interface part at a CMU

The data processing functions of the communication interface is described in the documentation of the specific communication protocol.

23BE21/22 or 560MIO80 functions:

Reading input register (every millisecond)
Digital filter (contact bouncing)
Consistency check
Store events in FIFO with time stamp

CMU - PDP:

Signal inversion
Scaling and format conversion
Transmission to internal communication

5.2.4. *Integrated Total Processing*

Integrated Total Value Types

There are two types of integrated total values (ITI) defined in the RTU560:

- End of period reading counters (EPR)
- Intermediate reading counters (IR)

ITI are transmitted periodically in fixed periods. Both types have one source and the IR is only an intermediate value of the corresponding EPR.

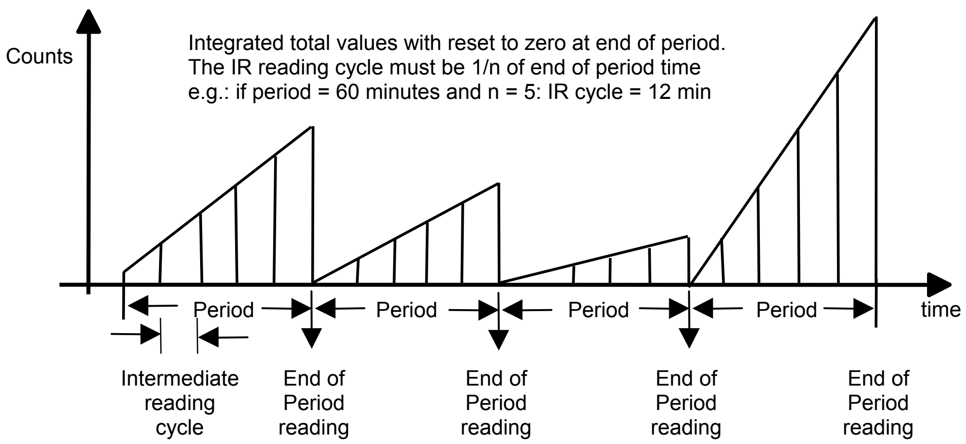


Figure 5-5: Integrated Total Values Definition for EPR and IR

Integrated total value presentation

Although the internal value representation is by a 32-bit signed integer the RTU560 supports positive ITI values only on its local inputs. This allows ITI values between:

- 0..and...+ 2 147 483 647

ITI Functions

The process data acquisition functions for ITIs processed by the RTU560 can be split into functions handled by:

- IOC of the binary input board 23BE21/22 or 560MIO80
- Process data processing (PDP) part of the CMU
- Protocol-specific communication interface part on a CMU

The data processing functions of the communication interface are described in the documentation of the respective communication protocol.

23BE21/22 or 560MIO80 functions:

Reading input register (every millisecond)
Digital filter (contact bouncing)
Increment integration register
Freeze integration register into relocation register

CMU - PDP:

Freeze and read ITIs periodically
Transmission to internal communication

5.2.5. Bitstring Input Value Processing

The RTU560 can handle bit patterns to read them and convert them into a bitstring input value (BSI):

- 8 bit bitstring (BSI8)
- 16 bit bitstring (BSI16)

The maximum length of a bitstring is the 16-bit word of (= one 23BE21/22 or 560MIO80 board). Double word values are not supported.

Bitstring Value Presentation

Bitstring input values are transparently mapped into 32-bit BSI messages of the internal communication. If an eight-bit pattern is selected the residual 8 bits of the 23BE21/22 or 560MIO80 board can be used for another bitstring value, digital measured value, integrated total values or indications.

BSI Functions

The data acquisition functions for bitstring values processed by the RTU560 can be split into functions handled by:

- IOC of the binary input board 23BE21/22 or 560MIO80
- Process data processing (PDP) part of the CMU
- Protocol-specific communication interface part on a CMU

The data processing functions of the communication interface are described in the documentation of the specific communication protocol.

23BE21/22 or 560MIO80 functions:

Reading input register (every millisecond)
Digital filter (contact bouncing)
Consistency check
Store events in FIFO with time stamp

CMU - PDP:

Transmission to internal communication

5.3. Command Processing

5.3.1. General

Signal outputs to the process and the sequential control of the operational equipment are processed and executed by the command processing function. The following command types are covered by RTU560:

- Object commands (single / double commands)
- Regulation commands (double commands)
- Setpoint messages (analog (not RTU560E)/ digital)
- Bitstring output command (not RTU560E)

Only one command of each type can be processed and output at the same time.

The communication unit accepts and checks the received command telegrams from the central system and releases them for execution if the check has been positive.

Depending on the command type the central control unit processes the commands like data base update or checks and if the tests are positive, it prepares the command-specific output procedures. Then the command is transmitted to the output board via the I/O bus.

Depending on the command type the command is released to the process by the binary output board 23BA20/560MIO80 or the analog output board 23AA20. Moreover the output board carries out some checks before output forcing.

The RTU560 acknowledges commands to the NCC within a time window of approx. 1.5 seconds. During that time the central control unit checks the command, transmits it to the output board and expects a return information that the command is accepted and executed or initiated. If an error has occurred a negative acknowledgement will be sent to the central system. The normal case of course is a positive acknowledgement.

In most cases commands are subject to a security philosophy depending on most critical process applications. For this reason the RTU560 terminal unit executes extensive test and security procedures for each command to be processed:

- Each board carries out self-tests for hardware faults
- Each command telegram received is checked for plausibility by comparison with the configured data in the database
- Each command is tested in the binary output board module 23BA20/560MIO80 by reading back the relay register
- The switching voltage for the output relays is monitored before and during the output of the command
- The command output pulse duration is monitored in the central control unit and on the output board

The output relays on the output board module 23BA20/560MIO80 are not switched before successful completion of the tests and enabling.

If an error is detected during the tests the command output will be aborted. This is also indicated directly on the output board module by the means of a LED.

For additional monitoring of the process output circuit for object commands, the command output monitoring board 23BA22 can be used, which allows a (1-out-of-n) check of the interposing relay.

5.3.2. Object Commands

This command type is used for the switching commands, e.g. for circuit breakers or isolators. Object commands are always configured as impulse double commands with two independent output relays (ON / OFF). The following modes of operation are possible:

- Command output without (1-out-of-n) check
  1-pole relay interfacing
  2-pole relay interfacing
- Command output with (1-out-of-n) check (not RTU560E)
  1.5-pole relay interfacing
  2-pole relay interfacing

The commands are output to the process via the relay contacts of the binary output board 23BA20/560MIO80. The following processing functions are possible:

- (1-out-of-n) check (command supervision)
- two step commands (Select before operate SBO sequence)
- command termination by a response indication
- persistent output

Single Object Command Output

A single command has only one output relay. It can be configured as pulse ON or OFF command or as persistent output.

Single object commands will be wired with one relay contact per direction or with two relay contacts per direction.

Single object commands are pulse outputs. The pulse duration is specified by the parameter *Command pulse length* per command (0.1 ... 25.5 sec.). Only the configured direction is used for pulse output.

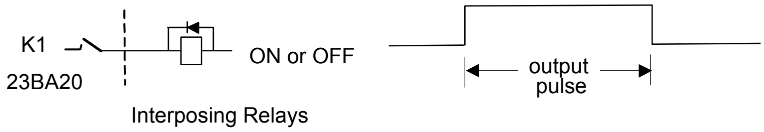


Figure 5-6: Single command definition: pulse output

Single object commands can be configured as persistent outputs. In the persistent mode an ON command switches the relay persistent on and the OFF command switches the relay to off.

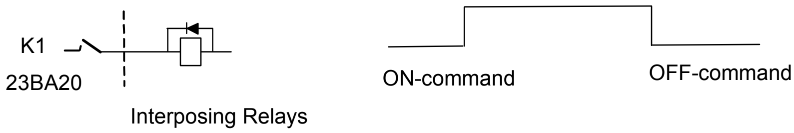


Figure 5-7: Single command definition: persistent output

Double Object Command Output

A double command has two independent output relays:

- one relay for ON direction
- one relay for OFF direction

Double object commands can be wired with one relay contact per direction or with two relay contacts per direction.

Double object commands are pulse outputs. The pulse duration is specified by the parameter *Command pulse length* per command. Only one channel ON or OFF can be active at the same time. The two relays occupy two consecutive bits within a 23BA20/560MIO80 output board. The ON-relay is always assigned to the odd channel and the OFF-relay to the even channel.

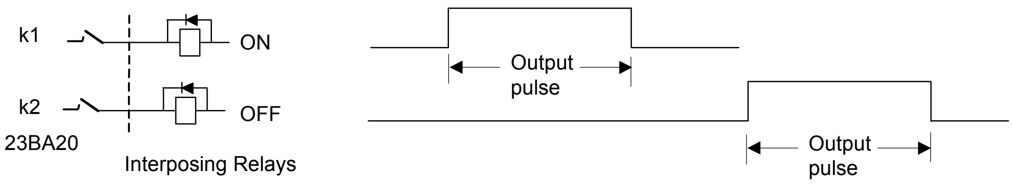


Figure 5-8: Double command definition: pulse output

Command Control by Response Indication

The pulse output of a running object command can be terminated earlier by the corresponding response indication. For a more secure switching a response delay time can be configured /typ. 200 ms).

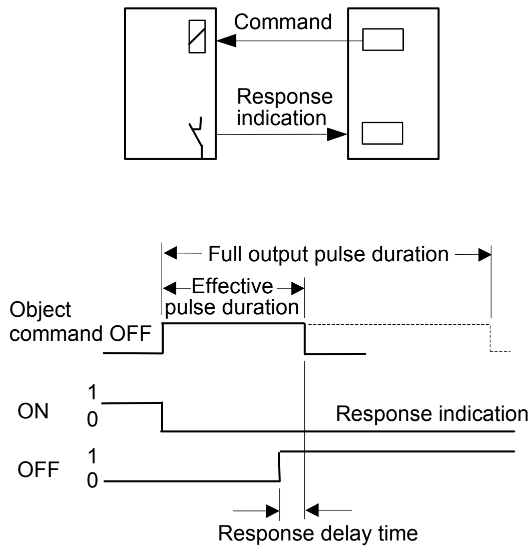


Figure 5-9: Command termination by response indication

(1-out-of-n) Check

The (1-out-of-n) check is not supported by RTU560E.

The (1-out-of-n) check detects errors in the external circuit like short circuits, open circuits or doubled relays which prevent the output of the command.

After receipt and successful check of an object command by the communication unit and the output board module the following steps will be executed before final command release:

- The (1-out-of-n) check on the command output monitoring board 23BA22 is activated.
- The resistance of the interposing relay in the switched output circuit is measured and compared with the parameterized upper and lower limits. If the resistance is within the limits the object command output to the selected equipment ~~is~~ will be activated via the binary output board 23BA20.
- The command output pulse timer is started, pulse duration is monitored, and the command output is deactivated by response indication or when the pulse time has elapsed on the 23BA22
- If the test conditions during the checks are not fulfilled the command will be cancelled.

In normal applications only one command output monitoring board 23BA22 is needed for the (1-out-of-n) check in a RTU station. In case of inserting different interposing relay types with different resistance values, two independent check circuits can be driven by the 23BA22 board, if the auxiliary test voltage is generated by a separate isolated voltage source. Concerning these conditions it is possible to include a maximum of 16 command output monitoring boards 23BA22 in a RTU560 remote terminal unit.

Select Before Operate Sequence

Before transmitting the final object in this operating mode, the NCC transmits a select command for this object. The communication unit in the RTU560 makes a prior check within the actual database, whether the selected channel is configured, is not activated and no error is present.

If the tests are successfully completed, then the relevant channel is designated as being selected, an acknowledged is generated for the selected command and transmitted to the central system.

To process the command and activate the command output an additional execute object command has to be sent from the central system within a time span of approx. 20 seconds. If the RTU560 does not receive an execute command then the selection of the object will be deactivated.

The following steps are executed on receipt of a valid execute object command:

- The object number is compared with the object number of the previously received selected command.
- Command output processing on the binary output board 23BA20/560MIO80 is activated and executed.

If any of the tests has a negative result the command processing will be aborted, the object selection will be reset and a negative acknowledge will be transmitted to the central system.

5.3.3. Regulation Step Command Output

These commands ensure the continuous fine tolerance adjustment of plant equipment, e.g. earth-fault neutralizers. Regulation commands are pulse double commands with two separated output relays.

Regulation step command outputs (RCO)

- can be wired for one-pole and two-pole switching
- cannot be wired with command supervision
- allows single step and two step command (Select before operate sequence)
- cannot be terminated by a response indication

Retrigger of Regulation Commands

The output pulse duration of a regulation command can be expanded if the same command is received within the output pulse time and can be sent to the output board before the time has elapsed. 23BA20/560MIO80 starts the timer again.

An output pulse can also be shortened by a new command with the DEACTIVATION flag. If a DEACTIVATION command is received the running regulation command will be stopped immediately.

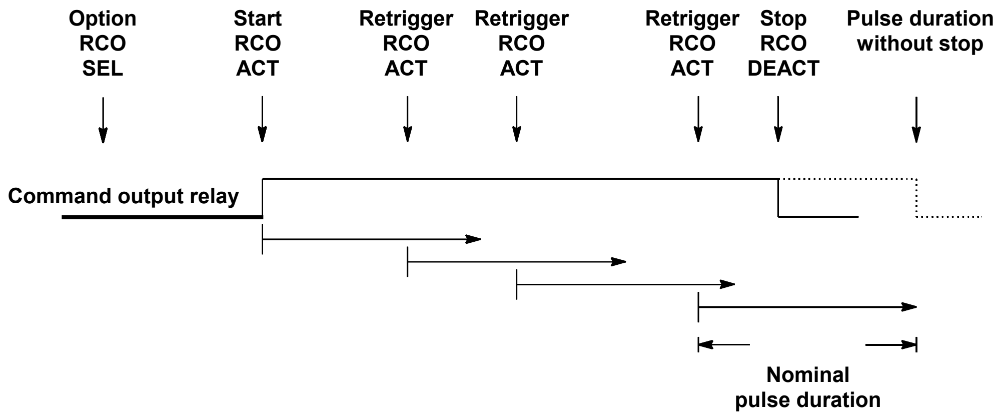


Figure 5-10: Retrigger / stop of a regulation command

5.3.4. Setpoint Messages

A setpoint message is used for the output of setpoints as control variables in control loops. The Select before operate sequence is supported.

Setpoint messages can be specified as digital or analog output values. Analog setpoint values are output via the 23AA20 analog output board and digital setpoint values via the 23BA20/560MIO80 binary output board. The RTU560 outputs a strobe pulse in addition to the setpoint value for the clear identification of a new value. The output value is validated with the strobe pulse (not RTU560E).

The resolution of the analog setpoint message is 12 bits plus sign (with 23AA20). Digital setpoint message outputs are 8 bits (23BA20/560MIO80) or 16 bits (23BA20) wide.

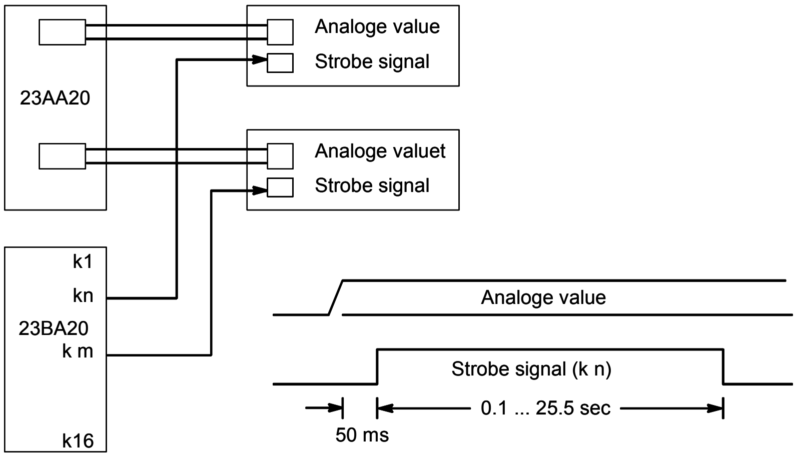


Figure 5-11: Timing diagram analogue setpoint message

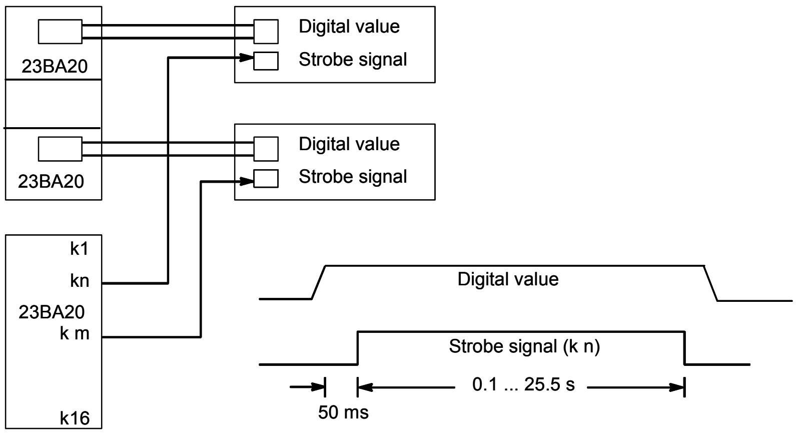


Figure 5-12: Timing diagram digital setpoint message

Two step commands (select before operate) are possible for setpoint messages.

5.3.5. *Bitstring Output*

A bitstring output (BSO) is a persistent output on a 23BA20. The following types are possible:

- Bitstring output 1 bit (BSO1)
- Bitstring output 2 bit (BSO2)
- Bitstring output 8 bit (BSO8)
- Bitstring output 16 bit (BSO16) (not RTU560E)

Bitstring output values are transparently mapped to 23BA20 output channels. The output value is switched on the output board and stays stable until a new value overwrites the existing one.

The maximum length of a digital measured value is 16-bit word (= one 23BA20 board) or 8-bit value for the 560MIO80. Double word values are not supported.

5.4. RTU560 Time Synchronization

5.4.1. Time Synchronization Principle

RTU560 provides a general time base that may be synchronized in three different modes:

Clock Synchronization Command (CS Command)
Synchronized by a (cyclic) time message from a Central System

CS Command & external minute pulse (not RTU560E)
Synchronized by a cyclic message via a host communication interface plus an external minute pulse wired to the TSI (Time Synch. Input on 560BCUxx)

Radio Clock only (not RTU560E)
Synchronized to the GPS or DCF 77 standard (middle Europe only)

Radio Clock and CS Command as backup (not RTU560E)
Synchronized to the GPS or DCF 77 standard. If the Radio Clock signal is missing or the clock becomes faulty, cyclic messages from a Central System are accepted.

The RTU560 MASTER CMU decides during start-up - by reading the GCD configuration - what kind of time synchronization is configured. It synchronizes the RTU time to the provided synchronization mode and acts as the **Time Master**.

The **Time Master** CMU keeps the time information for the entire RTU. It generates a controlled 10 KHz clock and the internal TSO minute pulse which are needed by all time slaves and the I/O master. It distributes the absolute time information in time message telegrams to the time slaves and the I/O master. (Note: If a Real Time Clock is used in a configuration, this board must be placed in the same rack as the Time Master CMU.)

Differences between the internal time and the received time on the time master are regulated by scalable predivider registers. This method allows a soft regulation of time differences and a long-time correction of crystal clock errors.

The **time slave** CMUs are hard coupled with the 10 kHz clock and the TSO generated by the Time Master. They cyclically receive a time message by the Time Master via Internal Communication and synchronize their time accordingly.

The **I/O master** (IOM) - on every CMU - is hard coupled with the 10 kHz clock and the TSO generated by the time master. It cyclically receives a time message from the MPU via the DPRAM interface and synchronizes its time accordingly.

The IOM again transmits a time synchronization instruction (broadcast) cyclically to all **I/O controllers** (IOC) on the I/O boards via I/O bus (typically every 2 seconds). The IOCs independently regulate deviations between their internal current time and the cyclic synchronization instructions. All I/O boards are time synchronized by the I/O with a resolution of ± 100µs and accuracy of ± 0.3ms.

5.4.2. Time Synchronization via the Telecontrol Line

For synchronization via the telecontrol line the NCC transmits a time synchronization instruction cyclically to the RTU560 remote terminal units (Time message).Discrepancies between the time received in the time message and the local RTU560 system time are continuously compensated by the CMU. If the discrepancy is greater than a fixed predefined tolerance the RTU560 system time will be set immediately. This control routine enables the RTU560 to correct the quartz drift of the internal system clock. Using this control mechanism the highly accurate time is attained in the terminal unit after about 1-2 hours following start-up. The long-term accuracy will reach up to ± 5 ms the TSI telegram is received regularly. According to IEC 60870-4 the time resolution corresponds to the time resolution class TR3 (<10 ms).

5.4.3. Time Synchronization by means of DCF 77 / GPS Receiver

If one of the real time clocks 23RTC01 or 23RTC02 is configured in the RTU560 A or C remote terminal unit, it is the master of time synchronization by using its minute pulse. The real time clock 23RTC02 receives the signal from the DCF77 transmitter, the 23RTC01 receives them from the GPS. The exact time synchronization is carried out with the minute pulse of the 23RTC01 board or the 23RTC02 board. The control mechanisms are then applied, in the same way, as for synchronization via the telecontrol line. The use of the 23RTC02 real time clock is limited to Central Europe because only in this region the signals of the long wave transmitter DCF77 can be received. For world-wide use the 23RTC01 real time clock is available.

5.4.4. Time Synchronization by Means of an External Minute Pulse

It is also possible to synchronize the RTU560 time with an external minute pulse derived from a local clock. In this case the system time has to be set by the time message and is synchronized by the external minute pulse (not RTU560 E).

5.4.5. Redundant Time Synchronization

Redundant time synchronization is not supported by RTU560E.

It is also possible to configure a redundant time synchronization mode. During normal operation the radio clock signal is used. In case of missing radio clock signal or the clock becomes faulty, RTU560 will also accept the cyclic time synchronization instruction from the Central System (not RTU560 E).

Note: If more than one Host Communication Line is connected to the RTU560, the line with the lowest *Host Number* is used for time synchronization.

5.5. PLC Development System

The RTU 560 PLC Development System is a standard programming and runtime system for IEC 1131-3 designed PLCs. It is based on the standard IEC 1131-3. RTU560 PLC consists of the ProConOS runtime system and the MULTIPROG wt programming system.

ProConOS (**Pro**grammable **Con**troller **O**perating **S**ystem) is an efficient PLC runtime system for complex control applications. It has been especially designed for IEC 1131 and includes nearly the full range of IEC 1131 features.

ProConOS is delivered in conjunction with the IEC 1131-3 programming system MULTIPROG wt. MULTIPROG wt allows an easy programming in function block diagram FBD and instruction list IL under Windows NT.

5.5.1. Components of the Development System

The PLC program is first programmed in MULTIPROG wt and then it is downloaded to RTU 560 where it is stored in the ProConOS memory. After this the PLC can be started. The process signals are transmitted via the local I/O or distributed router and sub RTUs to ProConOS. ProConOS stores the input signals coming from the process in its memory. During the working cycle the PLC program calculates the output signals on the base of the input signals. The output signals are transmitted from the ProConOS memory to the local I/O or distributed router and sub RTUs.

The debug kernel of ProConOS offers multiple debug functions such as overwriting and forcing variables, setting breakpoints, variable and address status with power flow. Debugging is normally done while the process is running. Nevertheless programming errors can also be corrected when the process is stopped by setting breakpoints. All debug operations are executed via the MULTIPROG wt user interface.

In MULTIPROG wt the compiler leaves the program in a PLC intermediate code. The ProConOS compiler translates the PLC intermediate code into an executable code.

5.5.2. MULTIPROG wt

The programming system offers features for the different development phases of a PLC application:

- Edit
- Compile
- Debug
- Print

The programming system is based on a windows technology providing comfortable handling using zooming scrolling, customizable toolbars, drag & drop operations, a shortcut manager and movable windows.

The system especially allows the handling of several configurations and resources within one project, including libraries and disposes of a powerful debug system. Projects are displayed and can be edited using a comfortable project tree editor in order to make the complexity of the IEC 1131-3 structure as simple and transparent as possible. The project tree editor allows the easy insertion and editing of program organization units (POUs), data types, libraries and configuration elements.

The programming system consists of a PLC independent kernel for programming in the IEC programming languages, IL (textual language) and FBD (graphical language). Each editor provides an Edit Wizard for the fast and easy insertion of pre-edited keywords, statements, operators, functions and function blocks.

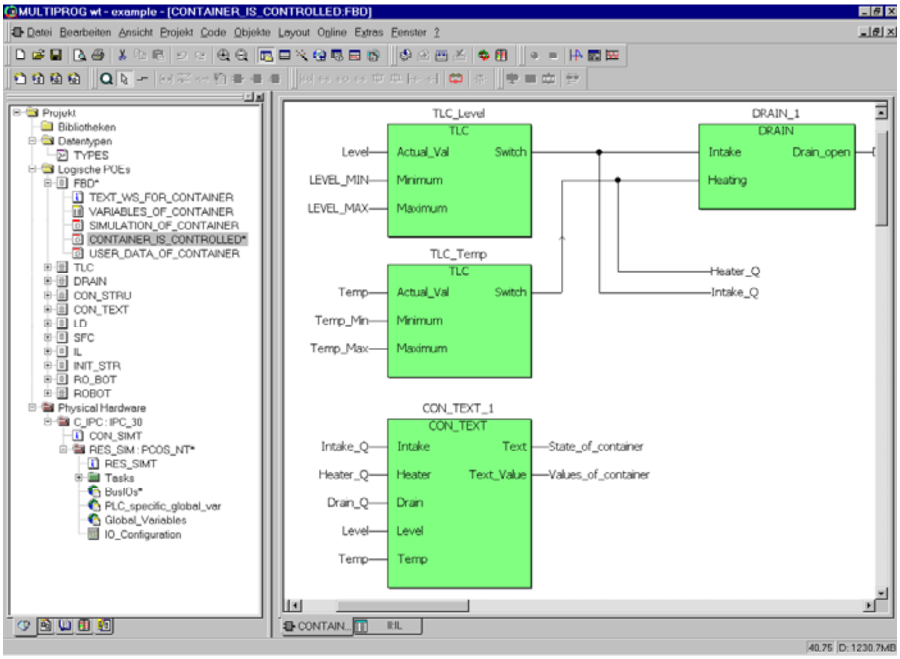


Figure 5-13: PLC language function block diagram

The Edit Wizard can also be used for declaring variables and data types. The independent kernel is completed with a specific part adapted to the RTU 560 PLC.

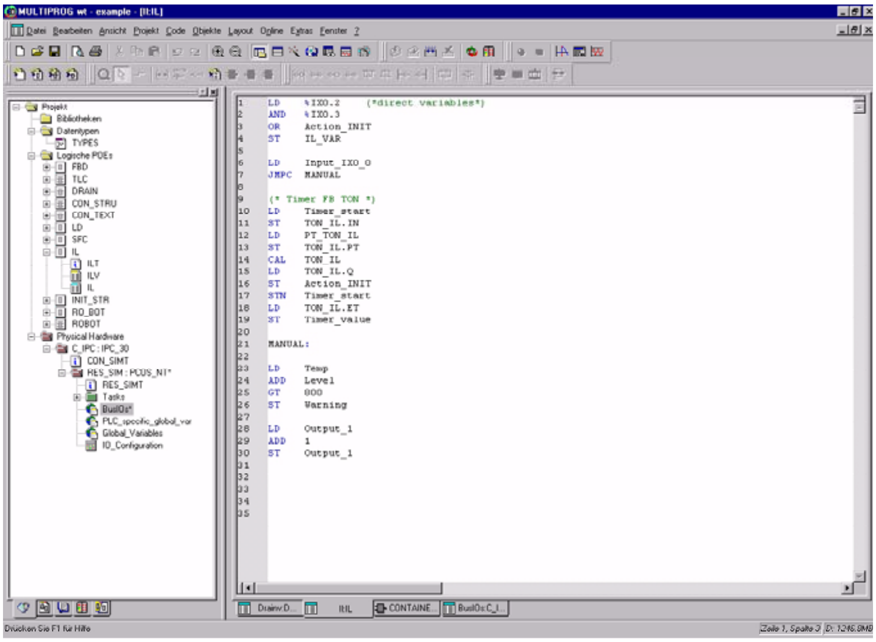


Figure 5-14: PLC Edit Wizard

The new easy online handling and the 32 bit simulation offers a fast power flow debug functionality.

A comfortable tool for project documentation is implemented for printing the project documentation either in a time-saving optimized way (using less paper) or with a stylish customized page layout.

5.5.3. PLC Performance Data

- 1000 Boolean instruction lines: 10 ms
- 1000 BOOL8 and INT instruction lines: 10 ms
- Shortest cycle period configurable standard: 10 ms
- Program memory capacity per POU configurable
  Approx. 1000 instruction need: 10 kByte RAM,
- In one POU 64 kByte program code can be executed.
- I/O image capacity configurable: Max. 1000 input signals and 1000 output signals
- Amount of user tasks: 15 task

5.6. Archive and Local Print Function

To get direct information about the data processing in a RTU some data may be printed directly to a local printer or archived in special files on the RTU560 file system. The two functions that gives these functionality are:

- The Local Print function

  Online output to a printer device connected on one of the serial interfaces.

- The Local Archive function

  Output into files stored on the flash file system on RTU560. From there it may be loaded to a host PCs file system or displayed by RTU560 Web-Server.

There are four different data types for local printout:

- Process events
- System events and System messages
- Analog measured values
- Integrated totals

The archived data is stored in files of the type *.log.

5.6.1. Configuration

The format of the recorded data as well as specific texts etc. is configurable by RTUtil 560. The configuration file for the functions local archive and local print is an ASCII file with the extension *.ptx. The hardware configuration file for a RTU (*.gcd) contains the information about the board where the local print or archive task will be started.

5.6.2. Local Print Example

Generally the output on printer and archive file is done in the same sequence, however in case of printer output there is a header with some information as date or table description on top of every page.

Printer output example:

```
Date: 02.06.22  RTU: Mannheim-East
S  Time         TIV Type   Object Text      Value   IV Class
S  17:26:38.687 TIV SPI    Switchgear276_1  ON      IV
S  17:26:38.687     SPI    Switchgear276_1  OFF     IV   A
```

5.6.3. Example Local Archive

The local archive files could be presented in a table on a web page. Only the last 50 entries in the event file are shown on this web page. To get the full file, it is feasible to load the ASCII files *.log to a PC and analyze the file in a standard editor.

Archive file output example:

```
Date: 02.06.22
S  Date     Time         TIV Type Object text      Value IV Class
  80.01.01 00:00:34.687 TIV  SPI  Switchgear276_1  ON    IV
S 02.06.22 17:26:38.687      SPI  Switchgear276_1  OFF        A
```

5.7. Disturbance Data Archive

5.7.1. Introduction

Protection equipment is limited in memory and therefore only a small amount of disturbance-record files can be stored in protection equipment memory before it is overwritten. To simplify the analysis of disturbance faults and to reduce the time to get information of these faults, it is necessary to collect them in an centralized archive. This archive is provided by RTU560.

The uploaded files are stored on the RTU560 file system. To avoid that flash memory on RTU560 file system exceeds, a file administration function is included.

To get access to the uploaded disturbance-data-files, the archive is displayed by RTU560 Web-Server, where it is also possible to upload files.

In addition it is also possible to upload disturbance data from protection equipment connected to Sub-RTUs by using file transfer function with the communication protocols IEC60870-5-101 or IEC60870-5-104.

5.7.2. Disturbance Data Archive Function

The following drawing**Fehler! Verweisquelle konnte nicht gefunden werden.** shows the structure of the RTU560 disturbance data archive and upload function. The following functions are required:

- Protection equipment supporting upload of disturbance records
- RTU560 CMU (560SLI02, 560ETH02/03, 560CMU80) with compact flash file system for archiving the disturbance files
- Work place PC with Web-Browser for uploading the disturbance data

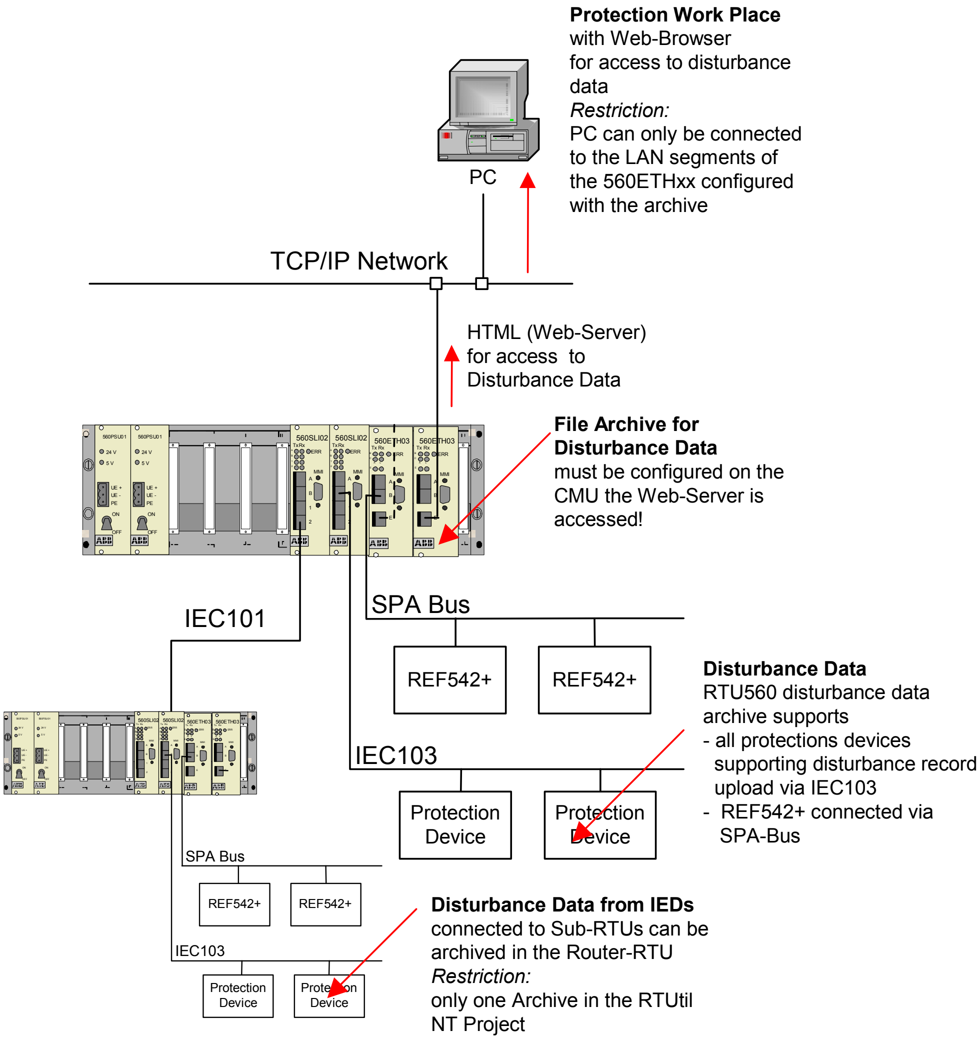


Figure 5-15: Disturbance Data Archive

5.7.3. Protection Equipment supported by RTU560

The RTU560 disturbance data archive supports following protection equipment:

- all protection devices supporting disturbance record upload via IEC103
- REF542+ connected via SPA-Bus

The protection device can be connected via an IEC103 line to an RTU560. In addition REF542+ is supported connected to a SPABUS line.

After a new disturbance record is available the device must send spontaneous a “list of recorded disturbances” (ASDU23) to the RTU560. The upload of disturbance records must be possible according to IEC103 section 7.4.7

After a spontaneous update of the “list of recorded disturbances” (ASDU23) from the IEC 103 device RTU560 checks for new files within the directory. The new disturbance files are uploaded with low priority and stored in the RTU560 disturbance file archive on the compact flash card.

For REF542+ with SPABUS RTU560 requests cyclical the device directory, detects new files and upload them to the disturbance file archive.

For other SPABUS devices the transparent SPABUS data channel in IEC101 or IEC104 can be used (see 1KGT 150 466 V003 1: RTU560 Host Communication Interface with IEC 60870-5-101). This function does not store disturbance records in the RTU560. The complete reading and storing of the disturbance records must be handled by the Control System (e. g. MicroSCADA Pro).

5.7.4. Access to RTU560 Disturbance Data Archive

The RTU560 disturbance data archive is accessed from a work place PC by a Web-Browser (e. g. MS-Internet-Explorer). The Web-Server of the RTU560 provides access to the archived disturbance data files.

In addition the Web-Browser on the working PC different conversion programs must be available. These conversion programs must be installed with the RTUtil 560 installation CD.

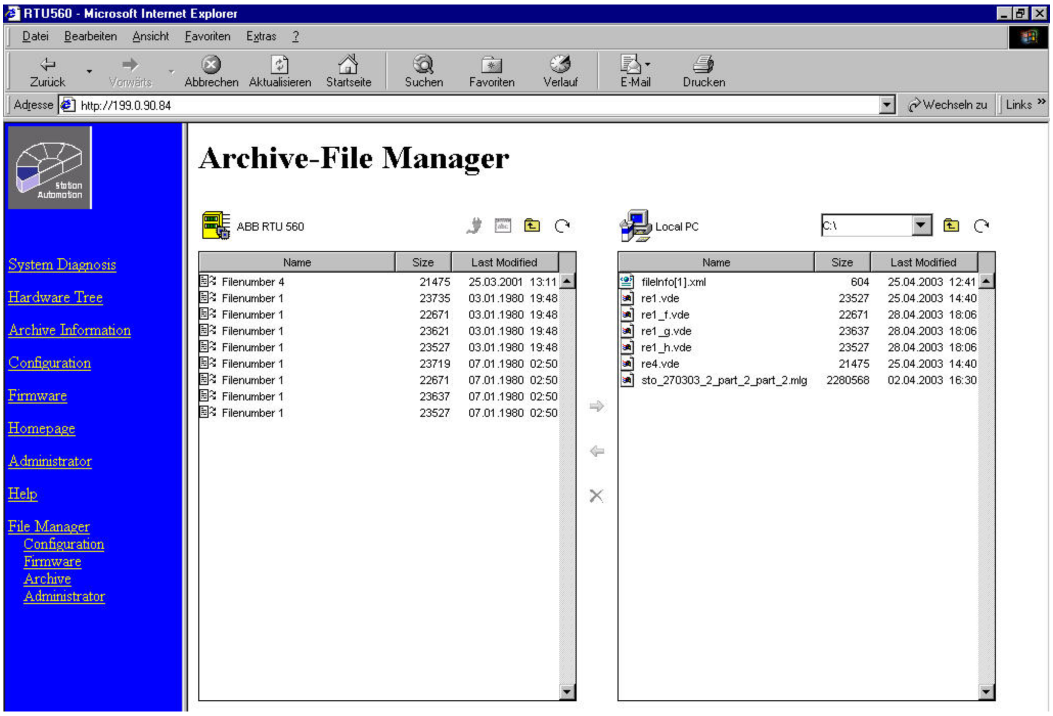


Figure 5-16: Disturbance Data Archive File Manager

The Web-Server on the CMU with disturbance archive provides a separate page for the disturbance file handling. This page shows the different available disturbance files in a user configurable directory structure. This page although provides the upload of the disturbance files to a PC. After the upload the disturbance files with different formats are converted into COMTRADE data format. It is possible to configure the different conversion programs for each disturbance file.

5.8. Load Profile Archive

5.8.1. Features

Meter Data (Load profiles and log-files) of alpha meters, which are generated in the devices, are transferred to the Compact Flash Card of one of the CMU's of a RTU560 system. Using the integrated Webserver of RTU560, it is possible to download these files to a local PC.

The contents of the transmitted files is according to EDIS (Energy Data Identification System) ((E)DIN 43863 Part 3), including ASCII data and <CR>/<LF>.

The Meter Data is requested in Protocol Mode C according to. Only some features of this protocol mode are supported by the subdevice communication interface (see Figure 5-17). **R6-command** is used for reading in order to transmit the files in separate blocks with separate checksum.

Mode C Feature	Support
Bi-directional data exchange	Supported
Baud rate switching	Not supported
Data readout	Supported
Programming with enhanced security	Not supported
Manufacturer-specific modes	Not supported

Figure 5-17: Supported Features of Mode C

5.8.2. Access to the File Archive

Within RTU560 Webserver there is an own page for the file archive. This page shows the files in a variable structure, configurable by the user. This page is also used for the file transfer of the files to the workplace PC.

The menu bar on top of the table contains (from left to right) a pushbutton to open a dialog for the conversion properties, a pushbutton to navigate in the folder, and a pushbutton to refresh the page.

The left hand part of the table shows all available directories within the RTU560, the right hand part shows the directory structure of the local PC.

For each directory displayed in the Archive File Manager it is possible to configure a target directory and a target filename. Entered values will be saved on the local PC for the current logged in user, if the operation system supports user accounts. Therefore the entered values have only be entered once, even if the web-browser or the RTU560 will be restarted.

5.8.3. Conversion Parameter

The field 'Target directory name' contains the name of the target directory on a local PC. This entry is only necessary for automatic download of meter data, initiated by the local PC.

In the field 'Target Filename' the resulting name of the selected file after copying to the local PC can be entered. If no value entered, a default name will be used.

Both values are strings, that supports wildcard usage. With the wildcards it is possible to define a target filename or a conversion call for all files in a directory.

Wildcards starts and ends always with the percentage-sign `%` (e.g. `%nameoffile%`). While processing, wildcards will be replaced by the corresponding values of the selected file.

The following wildcards are supported by the RTU560 archive manager:

Wildcard	Meaning
`nameoffile`	Name of File
`revindex`	Directory entries, that can not be differed by name of file or creation time, have different revision indices. Revision index starts always with 'a' and will be incremented
`cyear2`	Year of creation time in format YY. Example: Year '2003' will be displayed as '03' Range: 0 – 99
`cmonth`	Month of creation time in format MM. Example: Month 'December' will be displayed as '12' Range: 1 – 12
`cdayofmonth`	Day of month of creation time in format DD. Range: 1 – 31
`chourofday`	Hour of creation time in format HH. Range: 0 – 24
`cminute`	Minute of creation time in format MM. Range: 0 – 59
`csecond`	Second of creation time in format XX. Range: 0 – 59
`syear`	Year of storage time in format YY. Example: Year '2003' will be displayed as '03' Range: 0 – 99

Wildcard	Meaning
`smonth`	Month of storage time in format MM. Example: Month 'December' will be displayed as '12' Range: 1 – 12
`sdayofmonth`	Day of month of storage time in format DD. Range: 1 – 31
`shourofday`	Hour of storage time in format HH. Range: 0 - 24
`sminute`	Minute of storage time in format MM. Range: 0 – 59
`ssecond`	Second of storage time in format XX. Range: 0 – 59
`workingdir`	The path of the directory selected on the local PC selected in the right-panel.
`parentdir`	Name of the parent directory the selected file is located in as displayed in the header of the dialog.
`targetfilename`	The result of the processed target filename string. That means all wildcards are already replaced by the corresponding values of the selected file.

Wildcards which are numbers can be extended to a fixed number of characters by adding a '0' (Zero) followed by the number of characters of the item.

6. Integrated HMI

6.1. Overview

The RTU560 function ’Integrated HMI’ is an easy possibility to realized small customer specific monitoring and control applications. For this function no additional SCADA product is required.

The function Integrated HMI consists of two parts:

- An editor for the offline configuration
- A runtime system installed on the RTU560

With the offline editor customer specific pages can be created. Therefore a library package is provided with usable components. It is possible to connect dynamic components to RTU560 data points. Therefore the editor reads the data point list from the RTUtil 560 configuration files. The editor generates one HMI project file per RTU560 Integrated HMI function. This additional configuration file has to be uploaded to RTU560.

The following connection between RTU560 and the PC are possible:

- Local area network (LAN)
- Wide area network (WAN)
- Dialed line or modem connection (PPP)
- Direct serial connection (PPP)

Requirements for the local PC:

- Microsoft Windows compatible operating system
- Microsoft Internet Explorer (recommended) or Web browser
- Java Runtime Environment (available on the RTUtil 560 installation CD)

6.2. Function Distribution

The Integrated HMI function can be divided into the following parts:

- HMI Editor
- HMI Library
- User Project
- HMI Server
- HMI Application
- HMI Client

The HMI Editor is used to create user projects consisting of components provided by the HMI Library.

A user project is structured in pages where the components are inserted in. Components can be configured with the HMI Editor by the user.

The Components are placed on the *Pages* of the *Project.* These components are elements of a Components Library. The selected component can be changed in position, size, color and style. Static components will not change their style on the runtime system, while dynamic components will change dependant on the actual state of the process. The Component Library is described in this document.

6.3. Operation of the Editor

The HMI Editor has the same ‚look-and-feel' as other Microsoft compatible products. The HMI Editor is operated by pull-down menus placed in the header of the editor screen (see Figure 6-1).

Figure 6-1: Pull-down menu bar

6.3.1. Pull-down Menu 'File'

Using the menu button ***‚File'*** the user can:

- Create a *New project*
- *Open* an existing *project*
- *Save* an open *project*
- *Save* an open project with another name
- *Close* an open *project*
- *Exit* the Editor

6.3.2. Pull-down Menu 'Page'

The menu button ***'Page'*** is activated only, if a project is open. Using this menu button the user can:

- Create a *New* page
- *Open* an existing page
- *Rename* an active page
- *Close* the active page
- *Import* a page from another project
- *Export* the active page
- *Delete* the active page

After opening a new page, an empty worksheet is displayed in order to place the components on it. The layout of this worksheet can be configured with the *‚Options'* dialog.

While opening pages in a project, a selection of pages is shown, and the user can select one, several or all pages .After opening the selected pages, the pages are available in the worksheet (see Figure 6-2).

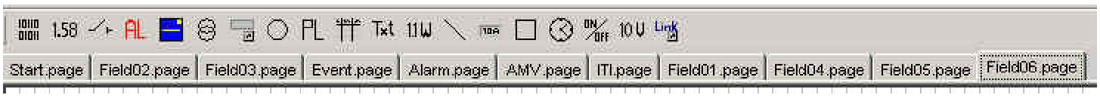


Figure 6-2: Available pages in a project

6.3.3. *Pull-down Menu 'Insert'*

Components will be selected by the '**Insert'** dialog or by choosing the component in the component tool bar (see Figure 6-3).

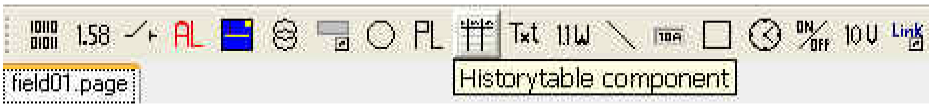


Figure 6-3: Component tool bar

6.3.4. *Pull-down Menu ‚Project'*

Project wide settings are set in the **Project Settings** dialog. The following **Application Settings** are available:

- Date and Time format — Format of the *System Time Component*
- Time format — Time format in *Table Components*
- Frame title — Title of all pages in this project
- Start page — Name of the page, that will be shown after start up
- Language — The dialog language of the runtime system

The project wide **Color Settings** are used to define the color change of dynamic components in the runtime system dependant on the state of the connected process object.

Used images in a project have to be imported with the **Project Image** dialog, before they can used by the *Image Component*. Supported image-types are **'gif'** and **'jpg'**. The available images can be selected in a dialog.

The configuration file of the tool RTUtil 560 has to be assigned to the active project in the **Project Configuration** dialog. If the project was assigned to a configuration before, the last used configuration is shown in the dialog.

In case of important differences between the *last used configuration* and the new *selected configuration,* the differences are listed and the user has to accept the selected configuration as the actual configuration.

The system is able to update the links between dynamic components und connected process objects. If the process object name is not available any more in the selected configuration, the link to this component we be deleted. This will be checked in the Consistency Dialog.

After finalizing a project, the user has to check his project with the help of the **Project Consistency** check. This feature will check

- The availability of a link to one *start page* in a project
- For all *dynamic elements* the necessary links to process objects
- All links between the *pages*
- The links to available images in the *image components*

Found errors are displayed in a separate dialog.

6.3.5. Pull-down Menu 'Options'

The **Grid** is used to make the drawing more simple. It is useful to connect the components to each other. If necessary, the grid can be turned off. *The 'Grid'* is the visible part on the screen, the gap between the lines is calculated in ‚Pixel'. *‚Snap'* is an additional invisible grid.

Normally *‚Snap'* is less or equal to *‚Grid'*, respectively *‚Grip'* is a multiple of *‚Snap'*.

The language used by the HMI Editor is selectable in the dialog **Options Language**. Available languages are English and German. The HMI Editor has to be restarted, before the new selected language will become the active language of the editor.

6.3.6. Pull-down Menu 'Help'

The dialog **Help About** will show the actual version of the HMI Editor and the used libraries (see Figure 6-4).

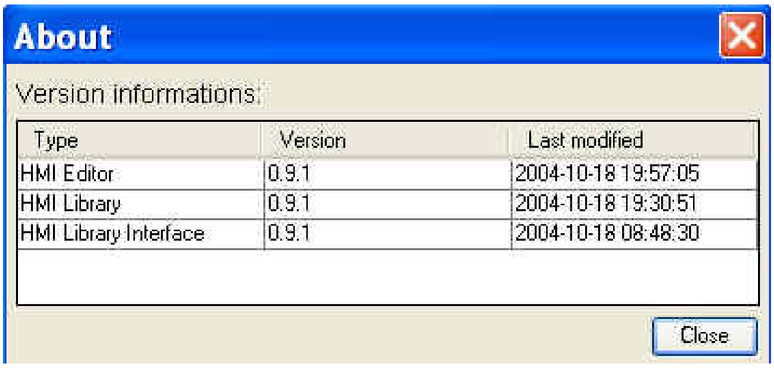


Figure 6-4: Version information

6.4. The HMI Library

The HMI Library consists of two files, *hmiLib.jar* and *hmiLibInterface.jar*. These files contains all components that can be inserted into a page of a user project.

The components of the HMI Library may be divided into the following sections:

- Static components (Line, Rectangle, Ellipse, Label, Image)
- Dynamic components (Byte Value, Floating Point, Integrated Total, Text Field, System time)
- Dynamic components with control (Circuit breaker, disconnector, Tap position, Normalized value, Bit string)
- Label (Links, Push buttons)
- Tables (History table, Process archive, Alarm list)

The components will be selected in the components tools bar or in the pull-down menu **Insert**.

6.5. Log In / Log Out

A user has to log into the running application of the Integrated HMI by putting in his user name and the corresponding password (see Figure 6-5). Each combination of user name and password is associated with several privileges.

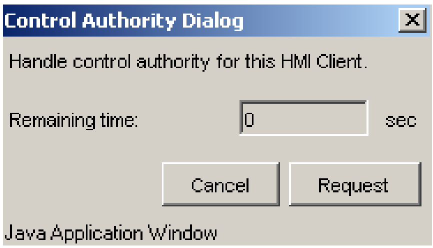
The login is done in the dialog **File → Login**

Figure 6-5: Login dialog

If the login was not successful, the login is confirmed, and an entry is generated in the event list, if available. If the user name or password is faulty, the access to the integrated HMI is denied.

6.6. Control Authority

The system takes care, that only one user is authorized to perform commands with the system. The Control Authority has to be requested by the user. This request will be notified to all connected host systems by a *System Event* (#100). Additionally it is possible -by a parameter- to reject commands from a control system, as long as this request is active. The duration of this request is limited by a configurable timer. The user has to request the Control Authority (see Figure 6-6) in the dialog **File → Control**



Authority

Figure 6-6: Requesting control authority

If the user is allowed to request the control authority, the supervision timer is started and the user may perform controls. If not, the demand is rejected.

6.7. Performing Controls

If the operator has the necessary privileges, it is possible to perform commands out of the single line diagrams. The symbol components will change to a three-dimensional style, if the operator touches the symbol with his mouse cursor (see Figure 6-7).

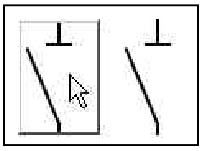


Figure 6-7: Component for process commands

After a click on the component symbol, a dialog is opened to perform the command (see Figure 6-8). A confirmation is necessary, before the selected command is executed.

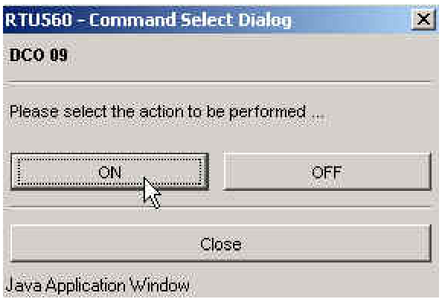


Figure 6-8: Dialog for process commands

6.8. Tabulated Representations

6.8.1. History Table

This component shows in tabular form the progress in time of an Indication, Measurement or Integrated Total (see Figure 6-9).

Number	Value	Qualifier	Time Tag
1	true		1980-01-01 23:59:22.023 IV
2	false		2004-10-25 10:29:40.538 SY
3	true		2004-10-25 10:29:41.574 SY
4	false		2004-10-25 10:29:44.737 SY
5	true		2004-10-25 10:29:45.598 SY
6	false		2004-10-25 10:29:49.074 SY
7	true		2004-10-25 10:29:49.669 SY

Figure 6-9: The History Table representation

6.8.2. Process Archives

The RTU560 supports three types of process archives:

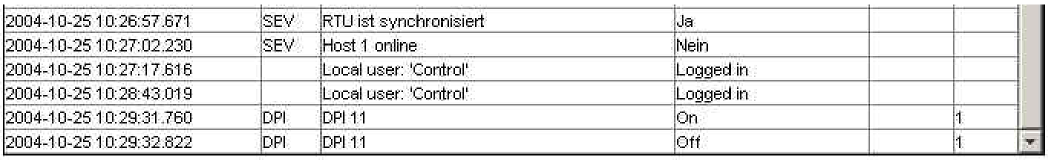
- Measurement value archive
- Counter value archive
- Event archive

All archives have the same representation in the Integrated HMI. Each page contains 50 entries. Dependant on the resolution of the used monitor, it may be necessary to scroll within this page.

Time	Type	Object	Value	Qualifier	Class
2004-10-22 09:10:00.000	ITI	ITI 14	0000001756		
2004-10-22 09:20:00.000	ITI	ITI 14	0000003511		
2004-10-22 09:30:00.000	ITI	ITI 14	0000005266		
2004-10-22 09:40:00.000	ITI	ITI 14	0000007021		
2004-10-22 09:50:00.000	ITI	ITI 14	0000008774		
2004-10-22 10:00:00.000	ITI	ITI 14	0000010527		
2004-10-22 10:10:00.000	ITI	ITI 14	0000001752		

Time	Type	Object	Value	Qualifier	Class
2004-10-19 16:22:31.757	AMI	AMI 01	+92.09%		
2004-10-19 16:34:52.885	AMI	AMI 01	+92.13%		
2004-10-19 16:41:31.283	AMI	AMI 01	+92.11%		
2004-10-19 17:04:11.755	AMI	AMI 01	+92.09%		
2004-10-20 08:51:26.364	AMI	AMI 01	+92.11%		
2004-10-20 09:58:59.362	AMI	AMI 01	+92.09%		

Figure 6-10: Integrated Total and Measured Value Archive



2004-10-25 10:26:57.671	SEV	RTU ist synchronisiert	Ja		
2004-10-25 10:27:02.230	SEV	Host 1 online	Nein		
2004-10-25 10:27:17.616		Local user: 'Control'	Logged in		
2004-10-25 10:28:43.019		Local user: 'Control'	Logged in		
2004-10-25 10:29:31.760	DPI	DPI 11	On		1
2004-10-25 10:29:32.822	DPI	DPI 11	Off		1

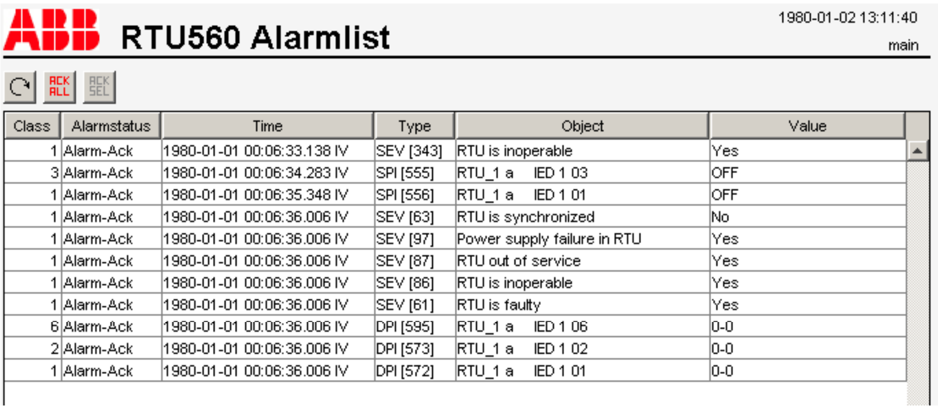
Figure 6-11: Event List archive

6.8.3. *Alarm List*

The RTU560 Integrated HMI is supporting one Alarm List for

- Persistent alarms
- Fleeting unacknowledged alarms

It is a part of the data entry, to specify the value(s) which will generate an alarm.



Class	Alarmstatus	Time	Type	Object	Value
1	Alarm-Ack	1980-01-01 00:06:33.138 IV	SEV [343]	RTU is inoperable	Yes
3	Alarm-Ack	1980-01-01 00:06:34.283 IV	SPI [555]	RTU_1 a IED 1 03	OFF
1	Alarm-Ack	1980-01-01 00:06:35.348 IV	SPI [556]	RTU_1 a IED 1 01	OFF
1	Alarm-Ack	1980-01-01 00:06:36.006 IV	SEV [63]	RTU is synchronized	No
1	Alarm-Ack	1980-01-01 00:06:36.006 IV	SEV [97]	Power supply failure in RTU	Yes
1	Alarm-Ack	1980-01-01 00:06:36.006 IV	SEV [87]	RTU out of service	Yes
1	Alarm-Ack	1980-01-01 00:06:36.006 IV	SEV [86]	RTU is inoperable	Yes
1	Alarm-Ack	1980-01-01 00:06:36.006 IV	SEV [61]	RTU is faulty	Yes
6	Alarm-Ack	1980-01-01 00:06:36.006 IV	DPI [595]	RTU_1 a IED 1 06	0-0
2	Alarm-Ack	1980-01-01 00:06:36.006 IV	DPI [573]	RTU_1 a IED 1 02	0-0
1	Alarm-Ack	1980-01-01 00:06:36.006 IV	DPI [572]	RTU_1 a IED 1 01	0-0

Figure 6-12: Example of an Alarm List

A persistent alarm has to be acknowledged by the operator, if this feature was enabled during data entry.

If an acknowledged alarm (*Alarm-Ack*) disappears, the entry will be deleted from the alarm list.

If an acknowledgement is not required, a fleeting alarm will be deleted from the alarm list.

If the acknowledgement is required, a fleeting alarm will stay in the alarm list, and has to be acknowledged later.

7. CMU-Redundancy

7.1. Overview

To keep access to a remote station can be very important for stations with critical responsibilities in an energy transportation grid. Mainly it is capability to keep access to station and to the important process units.

The RTU560 takes care of this requirement with a graceful degradation concept. Especially for the central part, the CMU boards in the RTU560. The RTU560A / C concept is designed to support redundant functionality.

The redundancy is available in steps:

- Redundant communication capabilities for RTU560A / C with
  Redundant lines
  Multi host
- Redundant power supplies in the communication sub-rack 560CSR01 for RTU560A
- Redundant CMU boards in the RTU560A / C

The hardware design of the RTU560 is state-of-the-art for industrial electronic products with respect to the environment conditions for outdoor stations and EMC conditions in electrical substations. The RTU560 modules have a high availability, proven by a low return rate of defective boards over the past.

7.2. RTU560A redundant CMU concept

The main concept is to have two sets of CMU boards for those communications lines and functions which are critical for the operation of the station. In case of an error condition the RTU560 will switchover to the standby CMU set and this CMU set starts to take over the tasks after a cold start.

For some function modules like archives and PLC it is not useful to have a redundancy because of loss of information for the archive or loss of process control for the PLC. These modules are only installed once in the RTU560A system and so called non redundant modules.

The RTU560A CMU modules can be defined in three groups:

- the active group (group A) — which is the active running RTU
- the standby group (group B) — which monitors the active group and is prepared to take over as active group
- the non redundant group (group C) — which continuous operation all the time independent of which group is the active one

To supervise the status of the RTU560A / C under this conditions it is necessary, that the standby group and the active group monitors each other to be in the situation to take over the other state or not. When for example a CMU board in the standby CMU set fails, it is not allowed to switch over on one side and it is necessary to inform the host about the failure in the standby set by the active CMUs. On the other side the standby CMU set must detect when the active CMU set fails without any alarm or warning message to take over the active state.

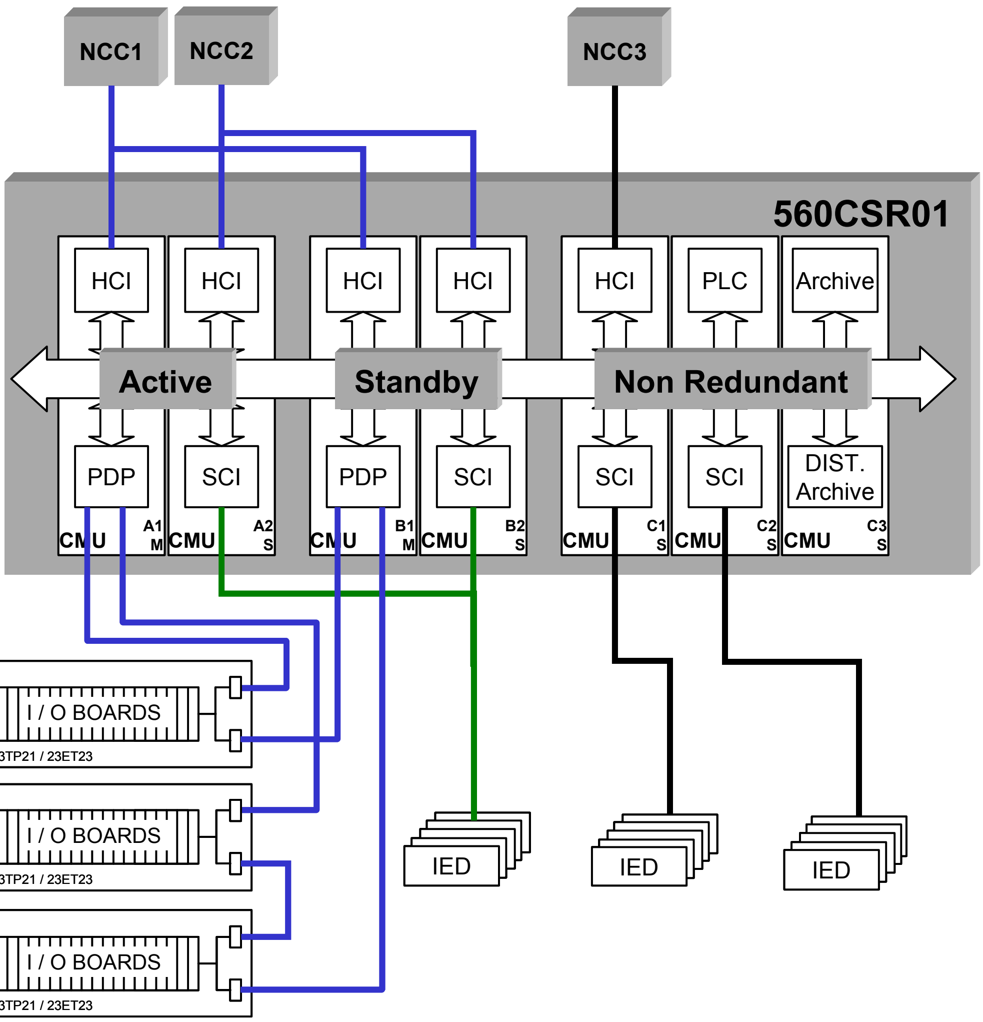


Figure 7-1: RTU560A configuration with redundant CMU modules

The above picture shows an example of a redundant RTU560A. The redundant group A and B may have the following configuration:

- NCC1 and NCC2 are communicating via a serial line protocol (e.g. DNP 3.0)
- The IO boards are organized in two PDP groups and connected to the CMU 1 of each group.
- some IEDs (e.g. the protection relays for the main transformers) are of high importance and therefore connected to the redundant group A and B

The non redundant group may have the tasks:

- a third NCC
- PLC
- Process event / Disturbance file / Load profile archive
- IEDs (e.g. additional protection relays)

7.2.1. Master / Slave concept for the CMU modules

In case of an RTU560A / C configuration of more then one CMU board it is necessary to define CMU Master within the RTU560. All other configured CMU modules are in that definition Slave. This definition is independent of CMU redundancy. RTUtil 560 is used to define Master and Slave CMUs.

During start up the Master CMU is responsible to inform about system problems etc. It is also responsible to run all system functions which have to be done by one CMU only; e.g. the time master in the RTU560 system.

Master / Slave concept with redundant CMU modules

Within each CMU group A and B must be one CMU defined as Master and all the other CMUs in the group as Slave. This implies, that all CMUs in group C can only be Slave CMUs. See the definitions shown in the above picture.

It is defined within the RTU560, that under normal conditions and no CMU board failures the group A is the active group and therefore group B the standby group. When it comes to a failure at start up of group A, group B will take over the active state and restarts the complete RTU560.

7.3. Redundancy switchover

A redundancy switchover will be caused by detection of system errors for one of the CMUs in the active group.

- CMU Master failures:
  CMU detects an internal error and stops (self supervision)
- Slave CMU failures:
  CMU detects an internal error and stops (self supervision)
- Watchdog activation on a CMU board. This causes a board reset.

A redundancy switchover will not be caused by

- Failure of a communication link to a master system or to a subsystem.
- Firmware or configuration errors
- PLC alarm condition initiated by a PLC program

Redundancy switchover

When it comes to a controlled switchover between the two groups, the active CMUs stop their activities and will do an internal restart. The line driver on the communication interfaces will go to high impedance of the tristate.

The standby CMUs will continue their start procedure from the point they have stopped by definition of standby. From the viewpoint of the RTU560 system it is a cold start. The now active CMUs starts communication on the serial lines and will initialize the communication to their host and sub-devices. The IO boards will do a reset. The PDP module takes over the communication on the RTU560 peripheral bus and reads all values from the IO boards.

7.4. Influence on RTU Functions

7.4.1. PDP and Counter Values

The IO boards will do a reset. The PDP module takes over the communication on the RTU560 peripheral bus and reads all values from the IO boards.

The counter values for the integrated totals start with their reset values; normally zero.

7.4.2. PLC Function

PLC Functions configured on a redundant CMU board

In case of redundancy switchover the PLC program on the new active CMU waits for the complete refresh of I/O data for process data base of the PLC module. Then a cold start of the PLC application is performed.

Note:

- The *.pro PLC configuration file has to be loaded to both redundant boards. It will not be distributed automatically.
- After a restart of a PLC program timers and storage functions start with their initial values.

PLC Function configured on a non redundant CMU board

This PLC program is not stopped because of a redundancy switchover. During the switchover task the PLC will continue running, but using the last actual values. The PLC program can read the status of the system, which allows to program the activities of the PLC program during the switchover period. After start up of the new active CMU group all data points coming from redundant CMUs are updated. The PLC can continue with normal operation.

It is possible to mix redundant PLC tasks and not redundant PLC tasks in one RTU560 system.

7.4.3. Group Functions

The Group function task to build summary alarms by AND and OR logic runs always on the active master CMU. So group functions will be always on a redundant CMU board. Calculating the group signals will be started again after the complete switchover by the new active CMU.

7.4.4. Archive and Local Print Function

Archive for process data, the disturbance data archive, the load profile archive and the local print function have to run on non redundant CMUs (group C).

7.4.5. Integrated Human Machine Interface

If process archives are used in the Integrated HMI, the Integrated HMI has to run on non redundant CMUs (group C), where the process archives are configured. If not, the Integrated HMI can run also on redundant CMU's.

7.5. RTUtil 560 Configuration

The configuration of redundant CMUs is part of the engineering tool RTUtil 560. The definition for each board (master / slave, which group) will be done here. RTUtil 560 identifies two configuration aspects.

Administration Mode

Initial Redundancy Mode

Administration Mode is a function required during system startup and for system supervision. This is identified by the Master CMU board. The master board coordinates system startup, system supervision and time administration. Each redundant group need one **Master** and all other CMU boards of that group must be **Slave. So f**or a redundant system a Master must be defined for each group A and B.

Initial Redundancy Mode defines the CMU function concerning redundancy. This mode although defines the default redundancy function after startup of a correct working system. The Initial Redundancy Modes are **Active (group A)**, **Standby (group B)** and **Non redundant (group C)**.

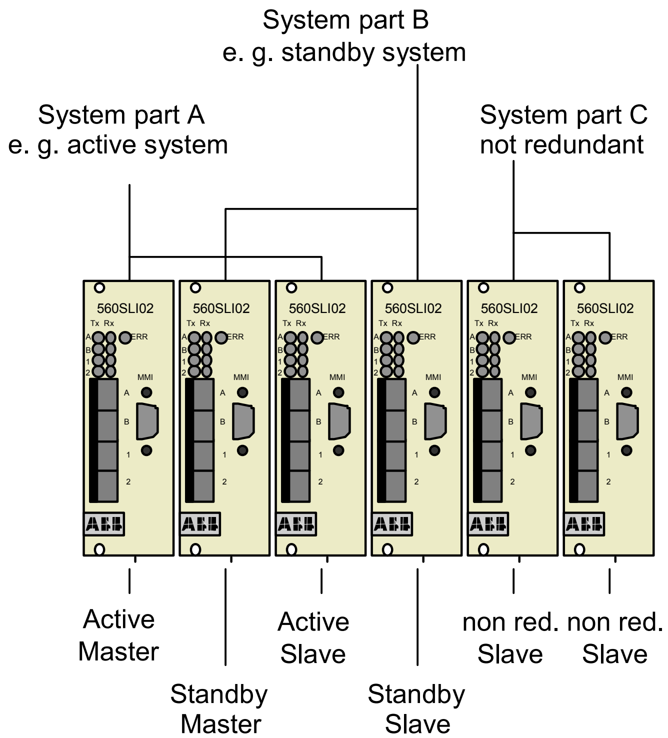


Figure 7-2: Administration and redundancy modes

8. Status and Diagnostic Information

8.1. Status and Error Report to NCC

RTU560 may report about its status and error conditions to the NCC via system events

The communication protocol used, determines the way, in which the system event number is processed into address information and which data type is used for state transmission to the NCC.

8.2. Web-Server Diagnosis

The RTU560 Web-Server is the maintenance and diagnosis tool of RTU560. In this chapter the diagnostic functions are described. For the full functionality of the Web-Server refer to the "RTU560 Web-Server User's Guide" (1KGT 150 523).

8.2.1. System Diagnosis

For indication of the RTU560 system status **system messages** are provided.

Example:
in a RTU with two CMUs -> a MASTER CMU in slot 3 and a SLAVE CMU in Slot 4, the following system message will be output for the CMU in slot 4 after removing and integrating the CMU in slot 3:

80.01.01, 00:00:00->CMU in rack 0, slot 4: STARTUP
80.01.01, 00:00:00->CMU in rack 0, slot 3: STARTUP
80.01.01, 00:00:42->CMU in rack 0, slot 4: STARTUP READY
80.01.01, 00:00:45->CMU in rack 0, slot 3: STARTUP READY
02.11.12, 14:36:02->RTU is synchronized
02.11.12, 23:27:08->Slave CMU in rack 0, slot 3: Error
80.01.01, 00:00:00->CMU in rack 0, slot 3: STARTUP
02.11.12, 23:33:10->CMU in rack 0, slot 3: STARTUP READY
02.11.12, 23:33:12->Slave CMU in rack 0, slot 3: OK

8.2.2. Status Information

The Web-Server's status information page gives information about the configuration of RTU560 and the actual system event list.

The status information page contains additionally:

- A wrong measured value, the missing I/O board or an invalid value are shown as a red “iv” sign next to the I/O card.
- More information about the actual values can be displayed by selecting the input board. The actual values and the status of the process objects are displayed (invalid (iv) and overflow (ov)).
- If a 560ETH02/03 board is configured you will get more information about this board (IP- Address, Net- Address and Net mask).
- The System Event list is shown after clicking the “RTU“ symbol in the tree structure.

8.3. LED's, Alarm and Warning

RTU560 supports signalization to report its error state. There are three different states:

OK / Warning / Alarm

8.3.1. LED Signaling on each CMU

The CMU alarm and warning conditions are caused by receiving and evaluating system messages.

The signaling on a CMU is as follows:

OK:	"ERR" – LED is off
Warning:	"ERR" – LED is flashing (approx. 1 Hz)
Alarm:	"ERR" – LED is on

8.3.2. RTU560 Alarm and Warning Relays

The error state of any CMU in RTU560 is reported to the CMU with administration mode MASTER, which evaluates a RTU560 sum state.

The RTU560 error state is signalized via a Board Connection Unit (560BCU01 /02 /03) device as follows:

OK:	All relays contacts OPEN
Warning:	Warning relays CLOSED
Alarm:	Warning relays CLOSED Alarm relays CLOSED

The MASTER CPU cyclically refreshes the watchdog register of the BCU device to enable RTU560 to activate the alarm relays in case of a MASTER CMU error after 30 sec.

8.3.3. LED Indications on the IO boards

All RTU560 boards have LED's to indicate errors or operating modes. These LED's allow a general visual check of the overall situation of the RTU560.

9. Additional Documents

9.1. Data Sheet for RTU560 A and C

Table 9-1 contains all available documents, describing the modules of the RTU560 A and C in detail.

ABB Ident Number	Title of Document
1KGT 150 401	Analog Output 23AA20 Data Sheet
1KGT 150 311	Analog Input 23AE21 Data Sheet
1KGT 150 312	Binary Output 23BA20 Data Sheet
1KGT 150 404	Command Supervision 23BA22 Data Sheet
1KGT 150 405	Relay Output 23BA30 Data Sheet
1KGT 150 406	Binary Input 23BE21 Data Sheet
1KGT 150 525	Binary Input 23BE22 Data Sheet
1KGT 150 407	Binary Input 23BE30 Data Sheet
1KGT 150 412	Sub-Rack 23ET23 Data Sheet
1KGT 150 414	Power Supply 23NG24 Data Sheet
1KGT 150 415	Fiber Optic Coupler 23OK22 Data Sheet
1KGT 150 419	Cabinet 23SC20 Data Sheet
1KGT 150 421	Cabinet 23SR20 Data Sheet
1KGT 150 422	Sub-Rack 23TP21 Data Sheet
1KGT 150 423	Main Adapter 23VG23 Data Sheet
1KGT 150 424	Main Adapter 23VG24 Data Sheet
1KGT 150 425	Cabinet 23WG20 Data Sheet
1KGT 150 426	Cabinet 23WG22 Data Sheet
1KGT 150 428	CCITT Modem 23WT22 Data Sheet

ABB Ident Number	Title of Document
1KGT 150 538	V.23 Modem 23WT23 Data Sheet
1KGT 150 539	9600 bit/s Modem 23WT24 Data Sheet
1KGT 150 408	Bus Connection Unit 560BCU01 Data Sheet, for 560CSR01
1KGT 150 409	Bus Connection Unit 560BCU02 Data Sheet, for 23TP21
1KGT 150 410	Bus Connection Unit 560BCU03 Data Sheet, for 23ET23
1KGT 150 411	Communication Sub Rack 560CSR01 Data Sheet
1KGT 150 466	Ethernet Communication Unit 560ETH02 Data Sheet
1KGT 150 518	Ethernet-Communication Unit 560ETH03 R0001 Data Sheet
1KGT 150 529	Ethernet-Communication Unit 560ETH03 R0002 Data Sheet
1KGT 150 416	Power Supply 560PSU01 Data Sheet
1KGT 150 418	Real Time Clock (GPS) 560RTC01 Data Sheet
1KGT 150 447	Real Time Clock (DCF) 560RTC02 Data Sheet
1KGT 150 478	Serial Communication Unit 560SLI02 Data Sheet
1KGT 150 400	RTU560 Environmental Data

Table 9-1: Data Sheets RTU560 A and C

9.2. Data Sheet for RTU560 E

Table 8-2 contains all available documents, describing the modules of the RTU560 E in detail.

ABB Ident Number	Title of Document
1KGT 150 493	Communication Unit 560CMU80 Data Sheet
1KGT 150 492	Fiber Optic Coupler 560FOC80 Data Sheet
1KGT 150 498	Housing 560HOS80 Data Sheet
1KGT 150 491	Multi I/O-Unit 560MIO80 Data Sheet
1KGT 150 495	V.23-Modem 560MOD80 Data Sheet
1KGT 150 496	Dial Up 560MOD81 Data Sheet
1KGT 150 497	GPS Modem 560MOD82 Data Sheet
1KGT 150 506	Power Supply 560PSU80 Data Sheet
1KGT 150 494	Power Supply 560PSU81 Data Sheet

Table 9-2 Data Sheets RTU560 E