



**BRODERSEN**  
simplifying systems

# Redundancy

## In the RTU32 / RTU32R Series



## RTU32 and RTU32R Redundancy Functions are designed for;

### **Withstand single point of failure**

- The RTUs are designed for providing redundancy in the actual application meaning that one-point-of-failure will NOT cause the general functionality to be interrupted. At the same time it will provide status about the failure and report it via communication interface, physical actions via outputs etc.

### **Simple and quick configuration of simple redundancy configurations**

- The software tools are designed for simple configuration of a redundancy setup. Default timing settings to be used for directly for simple applications – no need for advanced timing adjustments.

### **Advanced and flexible redundancy configuration/programming supported**

- The RTU32/RTU32R Series do also support functions available for handling more complex and advanced redundancy configuration with multiple drivers, advanced local control functions etc.
- You can design your own redundancy application as you have access to both a wide range of status monitoring and to control functions in the PLC programming tool.

### **Redundancy at NOT EXTRA COST**

- Redundancy function is standard in ALL RTU32 / RTU32R Series products.



### Level and types of typical redundancy configurations

#### **Redundancy Configuration 1: Single RTU with communication network redundancy**

- Redundancy in network communication only. Can be used with all drivers – even serial types. EN/IEC60870-5-104 with multiple connection in redundancy group.

#### **Redundancy Configuration 2: 2 x RTU in redundancy with single I/O configuration**

- Use RTU8 as the first I/O module (Intelligent I/O Controller) via serial port or other distributed I/O via LAN (e.g. ModbusTCP).

#### **Redundancy Configuration 3: 2 x RTU in redundancy with dual I/O configuration**

- Each RTU has full I/O configuration.
- All inputs and digital/relay outputs are double wired.
- Be careful with outputs as permanent active output will have short dropouts at switch over.
- Analogue outputs require special wiring via relay to avoid damage of output.

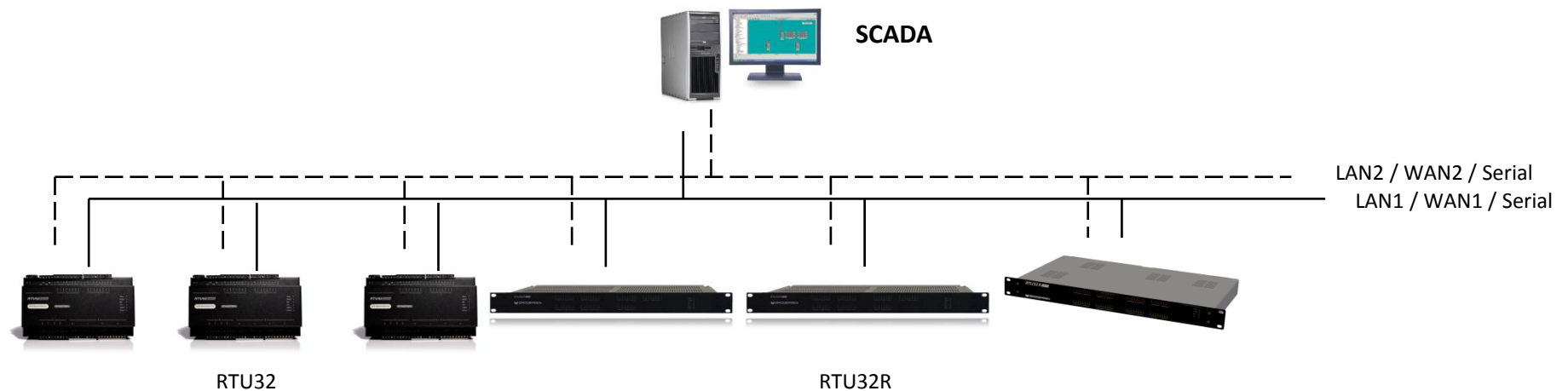
NOTE: All redundancy configurations requires quite complicated applications in both ends – and requires accurate specification, evaluation and tests of application programs.



## Single RTU and Communication network redundancy

### Redundancy features and functions:

- Supported for a wide range of both Ethernet and Serial drivers like;
  - ModbusRTU Master and Slave
  - ModbusTCP Client and Server
  - IEC60870-5-101 Master/Slave
  - IEC60870-5-104 Client/Server via Redundancy Group with up to 6 connections
  - DNP3 Master/Slave
  - DNP3 Server/Client
  - Straton Fast Binding – simply configured via adding 2 IP addresses.
- Possible via the standard physical communication interfaces available on the RTUs
- Require that each driver(s) are configured for redundancy and data are handled accordingly in the PLC application

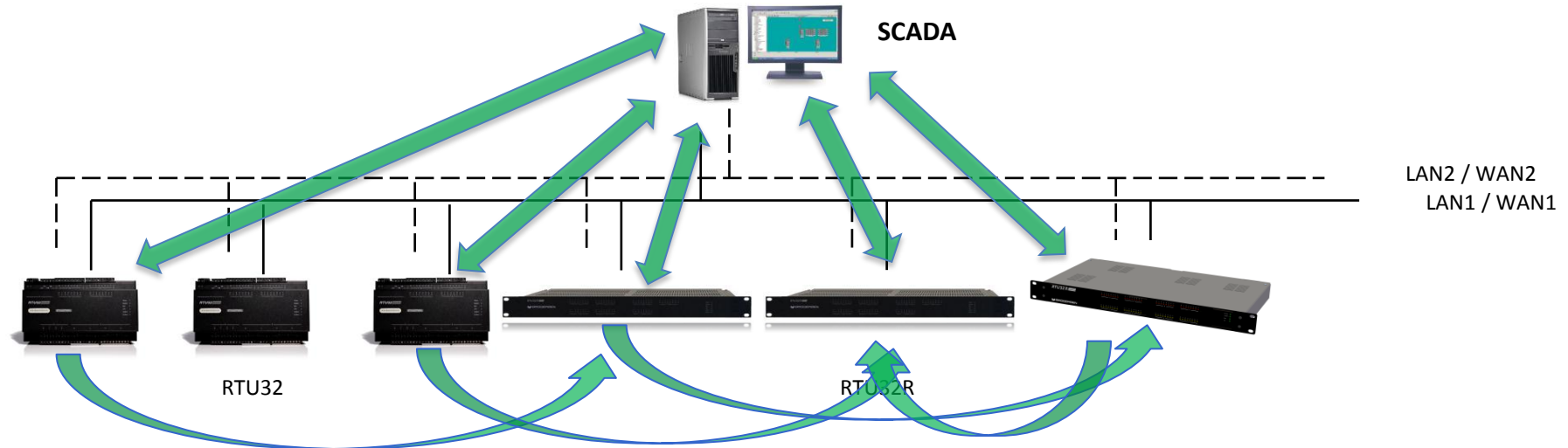




## Single RTU and Communication network redundancy using Fast Event Binding

### Straton Fast Event binding for total data distribution:

- **Straton Dual Binding** is a unique and powerful redundant communication setup
- Provides fast distributing of data from any PLC runtime to another PLC runtime (RTUs and PC/WorkStations)
- Simple and quick automated configuration – no need to setup advanced driver properties
- Status and time tag information means **secure** in communication
- Impressive track record – used for large applications like airports and power distribution automation.



**NOTE:** This works also with **zenon** SCADA from Copa-Data

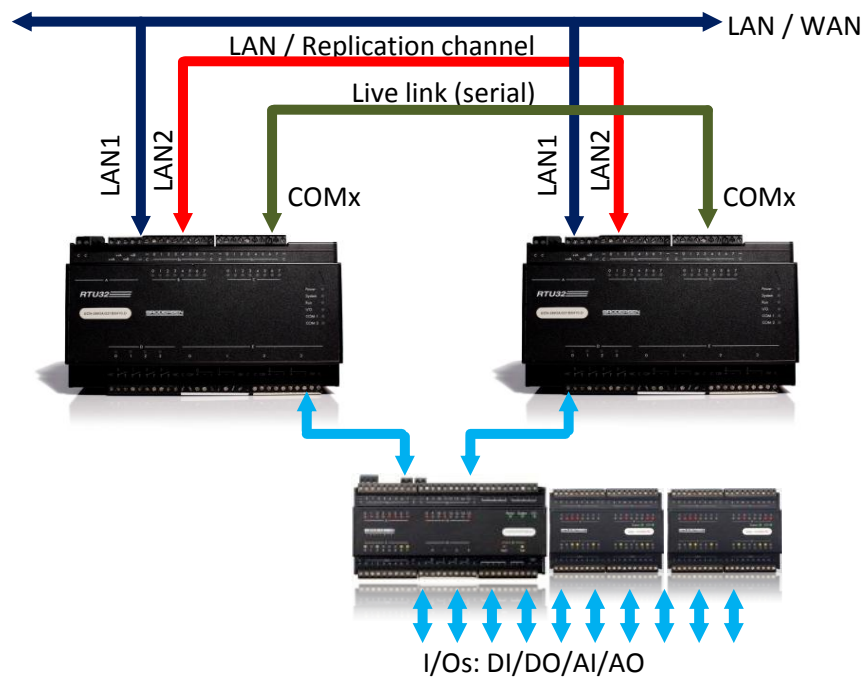


## With Power Supply, CPU and COM redundancy with shared I/Os

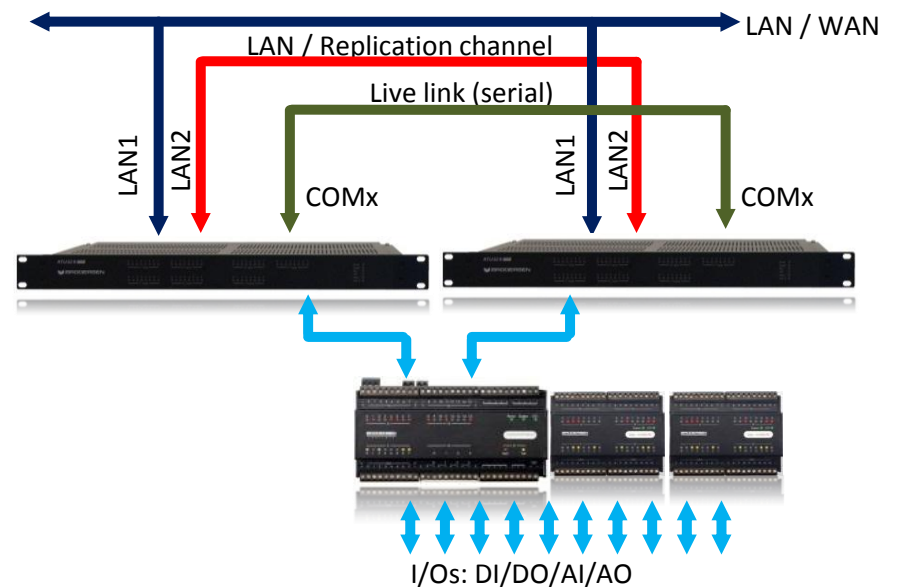
### 2 x RTU in redundancy with single I/O configuration

- Power Supply, CPU and COMs are provided integrated into a compact hardware product – so hardware redundancy means two complete equal RTUs in a hardware redundancy setup.

#### RTU32 Basic Redundancy Configuration



#### RTU32R Basic Redundancy Configuration



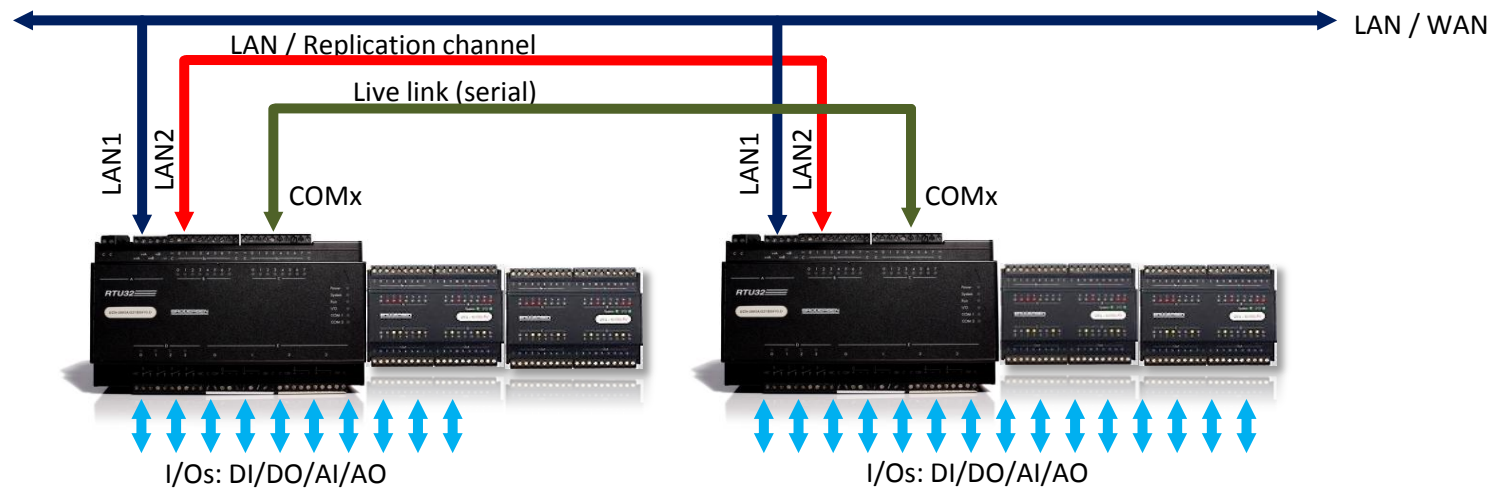


## With Power Supply, CPU, COM and I/O Redundancy

### 2 x RTU32 in redundancy with complete redundancy configuration

- Redundancy examples;

#### RTU32R Basic Redundancy Configuration



- Two communication links are used for data replication and status monitoring in order to withstand **One point of failure**.

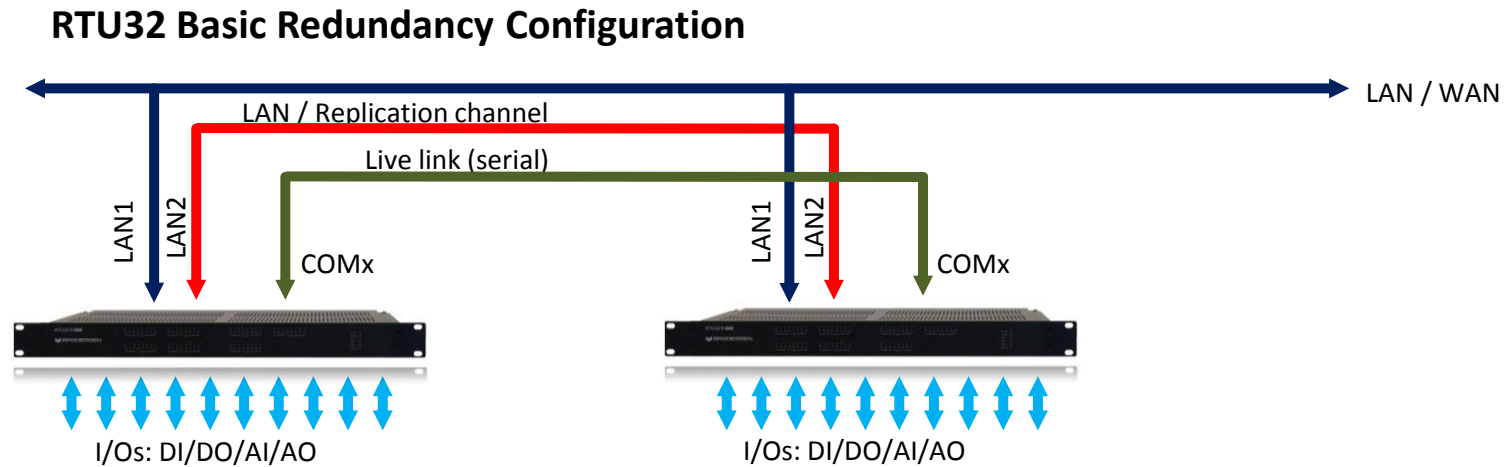
# Redundancy Configuration 3-2



## With Power Supply, CPU, COM and I/O Redundancy

### 2 x RTU32R in redundancy with complete redundancy configuration

- Redundancy examples;



- Two communication links are used for data replication and status monitoring in order to withstand **One point of failure**.



# Redundancy Configuration 3-3

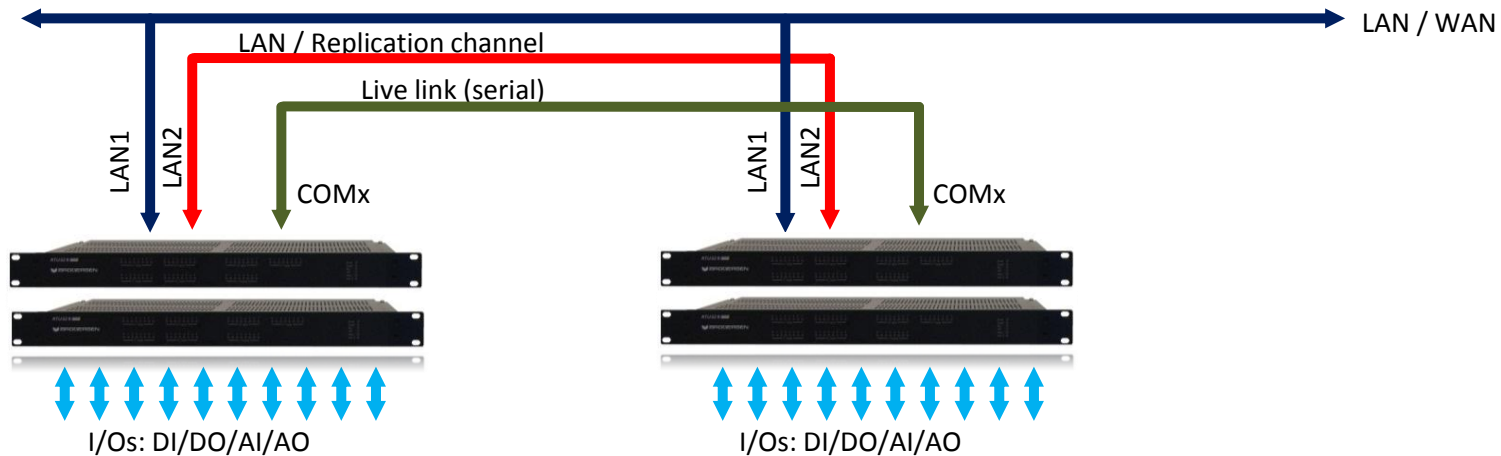


## With Power Supply, CPU, COM and I/O Redundancy

### 2 x RTU32R in redundancy with complete redundancy configuration

- Redundancy examples;

#### RTU32R Basic Redundancy Configuration



- Two communication links are used for data replication and status monitoring in order to withstand **One point of failure**.

# Redundancy Configuration 3-4

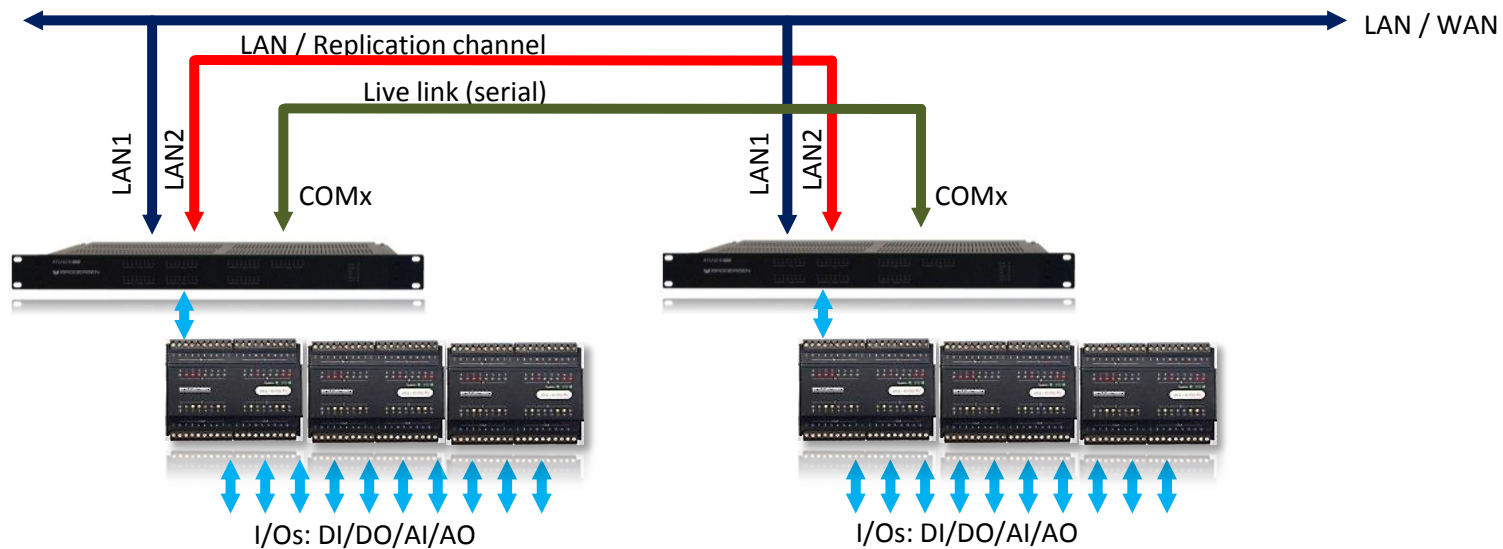


## With Power Supply, CPU, COM and I/O Redundancy

### 2 x RTU32R in redundancy with complete redundancy configuration

- Redundancy examples;

#### RTU32R Basic Redundancy Configuration



- Two communication links are used for data replication and status monitoring in order to withstand **One point of failure**.



## Communication Drivers in Redundancy applications

Redundancy applications with PLC functions alone are simple! But redundancy application that includes one or several communication drivers can be complex to handle. The reason is that communication protocols are designed differently and therefore also must be handled differently in redundant applications.

In addition it is important to realize that communication redundancy requires two partners – the Master (serial) or Client (Ethernet) in one end and the Slave (serial) or Server (Ethernet). And they must use the same redundancy strategy to succeed. In all redundancy applications the host/Master/Client likewise must be able to handle switching between the primary(active) and secondary(passive) RTU.

**In the following pages are listed comments to the use of different available protocols in redundancy applications**



## IEC60870 Communication Drivers in Redundancy applications

In general the IEC60870 Driver are using link layer buffers in both Monitor and Control Direction. And for that reason the IEC60870-5-104 protocol standard define specific Redundancy Group functions – which the RTU32 Series supports for redundancy communication (one RTU!).

For serial driver redundancy the RTU32 Series IEC60870-5-101 Slave support up to three parallel physical interfaces to the same data queue system – and one even support dial-up with active dial as a fall back communication option.

When redundant hardware are required (2 x RTU32), you must install equal applications in both RTUs and let the Host control which link is active and ignore information's in monitor direction from passive RTU. Host/Master/Client should in other words handle the validity of data received in monitor direction. The RTUs includes a list of features to monitor partner RTUs and monitor and control link drivers and data buffers.

RTU32 EN/IEC60870-5-104 Drivers are is certified in compliance with the latest redundancy requirements in the protocol specifications.

NOTE: it requires advanced knowledge of IEC60870 to develop proper redundancy applications.



## DNP3 Communication Drivers in Redundancy applications

DNP3 drivers are quite similar to the IEC60870 Driver suite as it is using link layer buffers in both Monitor and Control Direction. But the way the buffers are used by the application layer is different. STRATON runtime redundancy support DNP3 drivers - but the driver configuration must be considered carefully when using redundancy.

The DNP3 Slave (serial) driver support +3 concurrent Master connections (via 3 COM ports).

NOTE: it requires advanced knowledge of DNP3 to develop proper redundancy applications.



## IEC61850 Communication Drivers in Redundancy applications

The IEC61850 Client/Server Protocol Specification does not yet defined any redundancy functionality.

If a redundancy application with IEC61850 Server or Client are required, it needs to be defined in both ends (Client/server) how the configuration and communication going to be handled. Handling the models and monitoring the communication links will be a challenge.



## ModbusRTU and ModbusTCP Drivers in Redundancy applications

The Modbus Suite are simple polled communication protocols which are able to be handled in the STRATON Redundancy functions. That goes for both Ethernet and serial based Modbus as well as Master/Slave and Client/Server.

The Redundancy function in STRATON, which automatically takes care of handle active and passive runtimes in configurations of two RTU32 deviecs. Only one application program is needed/necessary.

The Host/Master/Client is responsible to handle the two communication links as they will have different addresses.



## STRATON Fast Event Binding Driver in Redundancy applications

The STRATON Binding driver for fast event communication over Ethernet is 100% designed supporting;

- Communication redundancy (with one RTU and two redundant networks)
- Device redundancy (with two RTUs setup in a redundancy configuration)

Only one application program is necessary – the driver configuration just needs two IP addresses.





## SNMP Agent Drivers in Redundancy applications

The SNMP Extension Agent driver (Ethernet) does NOT support STRATON Redundancy.

If redundancy are required, it must be established by having two RTUs and let the SNMP Management System handle what Agent that should be the active communicator.

STRATON includes several functions for monitoring and controlling redundancy system handling (like see if partner RTU is alive and re-start a RTU)

The SNMP Agent driver is an Extension Agent that use the WinCE SNMP low level functions. This make a redundancy setup only possible with two separate RTU32 SNMP applications running.

The WinCE basic MIBII SNMP function offers getting WinCE OS information on requests from SNMP Management System to monitor SNMP Agents in the network.

If you want special actions running in case of primary RTU break-down, it has to be controlled from the SNMP Management System.

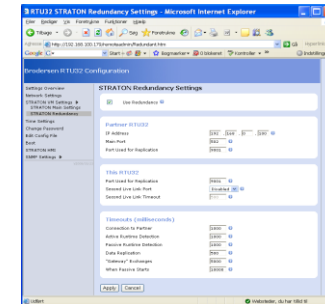
# Redundancy Configuration in the RTU32/RTU32R Series



## PLC runtime redundancy configuration

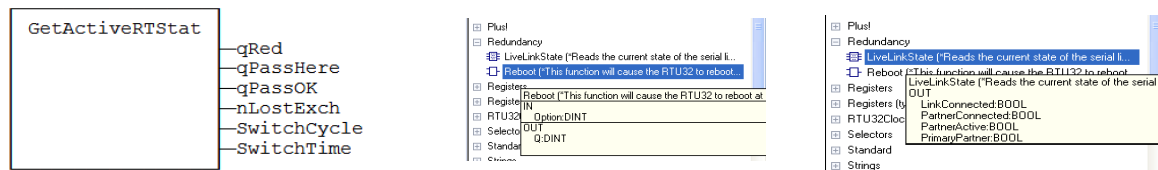
### Fast and simple configuration;

- Configuration of PLC Redundancy on Webpages.
- User friendly with information to each configuration parameter.



### PLC Functions diagnostic and control;

- GetActiveRTStat function: General info about Runtime status
- LiveLinkState: Status on serial live link to redundancy partner
- Reboot: Control function to boot the RTU – when switch of active partner



**NOTE:** Beside the dedicated Redundancy PLC Functions - any redundancy application will require monitoring of status registers for communication etc. – in order to detect any kind of fault that shall create switchover.



You are in the process of designing or evaluating redundancy requirements – here are some important issues to have in mind:

**Any communication will have delays that influence how fast you will see a break-down (detect a failure that would force a switch-over to redundant RTU):**

- Serial drivers on direct lines has timeout before errors are reported. Modbus e.g. is typical seconds.
- Ethernet network includes often several routers that as any WinSock connection provides general slow link error reports.
- Ethernet network components as router and switches may close down ports periodically as maintenance (in order to remove old TCP sockets not closed down properly)
- Even the fastest switch-over from primary to a secondary application in a redundancy suffer from uncontrolled time slots – no matter what suppliers of equipment claims!



## Timing specifications for 2 x RTU32 in Redundancy Configuration

**The switch time between RTUs (defined as a failure is detected till the backup RTU is running) depends directly on a number of parameters like;**

- PLC cycle time (size of application).
- Use replication channel point-2-point or over local network.
- Timeouts in links for communication protocols
- Etc.

On a medium application size with optimal software and hardware configuration the fastest switch time will be around 300-400msec.

On a typical application with no special measures done to optimize for redundancy and just configured by adding the mark in the configuration website in RTU32 - the typical switch over time is 1-2 sec.



## Facts about RTU32/RTU32R Redundancy

- When using RTU32 in hardware redundancy it means that you have to take two standard RTU32 and connect between them a Ethernet cable (data base replication channel) and a serial cable (additional live-link).
- Redundancy function in RTU32 PLC runtime covers redundancy in PLC runtime – one RTU32 is running as active and second is running as passive PLC runtime.
- Drivers directly configured in STRATON is in general covered by the redundancy function. That includes e.g. all Modbus drivers, IEC61850, DNP3 and Binding.
- Drivers that is NOT configured in STRATON is not covered by the PLC redundancy function. That includes e.g. SNMP and IEC60870 Suite (IEC60870 support the Redundancy Group function defined in the IEC60870 standards).
- As the RTUs in a redundancy configuration is not based on a backplane, the Ethernet/LiveLink cable connections are considered to have the same function as the backplane for communication between the CPUs.