RTU32N Series

Compact RTU, PLC and Utility Controller

User Manual

BRODERSEN

simplifying systems

Table of Contents

1. Customer Information

1.1 Copyright Notice

Copyright 2018, Brodersen A/S, ALL RIGHTS RESERVED.
No part of this document may be reproduced, copied, translated, or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the prior written permission of the original manufacturer.

1.2 Trademark Acknowledgement

Brand and product names are trademarks or registered trademarks of their respective owners.

1.3 Disclaimer

Brodersen A/S reserves the right to make changes, without notice, to any product, including circuits and/or software described or contained in this manual in order to improve design and/or performance. Brodersen A/S assumes no responsibility or liabilities for the use of the described product(s), conveys no license or title under any patent, copyright, or mask work rights to these products, and makes no representations or warranties that these products are free from patent, copyright, or mask work right infringement, unless otherwise specified. Applications that are described in this manual are for illustration purposes only.
Brodersen A/S makes no representation or warranty that such application will be suitable for the specified use without further testing or modification.

1.4 Life Support Policy

BRODERSEN A/S's PRODUCTS ARE NOT FOR USE AS CRITICAL COMPONENTS, IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE PRIOR WRITTEN APPROVAL OF BRODERSEN A/S.
As used herein:
1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into body, or (b) support or sustain life and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

1.5 Brodersen Customer Services

Your satisfaction is our primary concern. Here is a guide to Brodersen customer services. To ensure you get the full benefit of our services, please follow the instructions below carefully.

1.6 Technical Support

We want you to get the maximum performance from your products. So if you run into technical difficulties, we are here to help. For the most frequently asked questions, you can easily find answers in the product documentation. These answers are normally a lot more detailed than the ones we can give over the phone. So please consult this manual first.
To receive the latest version of the user manual, please visit our Web site at:

http://www.brodersen.com,

Choose the product in question under product search and under each product you will find accompanying data sheets, manuals, and user guides etc.
If you still cannot find the answer, gather all the information or questions that apply to your problem, and with the product close at hand, call your dealer. Our distributors are well trained and ready to give you the support you need to get the most from your Brodersen products. In fact, most problems reported are minor and are able to be easily solved over the phone.
In addition, technical support is available from Brodersen engineers every business day. We are always ready to give advice on application requirements or specific information on the installation and operation of any of our products. Please do not hesitate to call or e-mail us on support@brodersen.com .

Denmark:
Brodersen A/S
Islevdalvej 187
DK-2610 Roedovre
Tel.: +45 45 35 26 27
sales@brodersen.com

1.7   Product Warranty

Brodersen warrants to you, the original purchaser, that each of its products will be free from defects in materials and workmanship for two years from the date of purchase.

This warranty does not apply to any products which have been repaired or altered by persons other than repair personnel authorized by Brodersen, or which have been subject to misuse, abuse, accident or improper installation. Brodersen assumes no liability under the terms of this warranty as a consequence of such events. Because of Brodersen's high quality control standards and rigorous testing, most of our customers never need to use our repair service. If a Brodersen product is defective, it will be repaired or replaced at no charge during the warranty period. For out-of-warranty repairs, you will be billed according to the cost of replacement materials, service time, and freight. Please consult your distributor for more details. If you think you have a defective product, follow these steps:

1.   Collect all the information about the problem encountered. (For example, Product type and s/n, hardware and software version etc.) Note anything abnormal and describe the error in a product failure report.
2.   Call your distributor and describe the problem. Please have your manual, product, and any helpful information readily available.
3.   If your product is diagnosed as defective, make arrangement with your distributor about this.
4.   Carefully pack the defective product, a complete failure report and a photocopy of proof of purchase date (such as your sales receipt) in a shippable container. A product returned without proof of the purchase date is not eligible for warranty service.
5.   Ship it to your distributor.

2   Introduction

2.1   RTU32N – Brodersen RTU, PLC and Industrial Controller

The Brodersen RTU32N series is based on a 32-bit IMX6 ARM platform, providing a small footprint RTU and PLC with leading edge functionality. As well as being a versatile RTU and PLC with flexible I/O designed to perform embedded data processing, control, data logging and monitoring, it is also a networking communicator for collecting, managing and communicating data via protocols on different physical interfaces upwards and downwards in an industrial environment.

Typical applications for RTU32N are:

- Substation automation applications
- Railway level crossing monitoring
- Control and monitoring of water pump stations
- Communication Gateway

The RTU32N hardware platform includes a 200-900MHz CPU with 128-256MB RAM and integrated 128MB NAND Flash. In addition, a removable Micro-SD Flash card can be used (default size of 4GB), provides an easy to use portable boot/access/upgrade option. The integrated Flash stores the RTU32 firmware including OS.

2.2   WebServer configuration and PLC programming in all IEC61131 languages

The RTU32 Series is configured via the integrated WebServer. Settings like LAN port addressing, time functions and 3g/4g modem credentials are handled in webpages. Online monitoring of connected physical I/O and display of graphical overviews is possible via webpages.

Programming of logic functions and configuration of communications drivers is managed using the Brodersen WorkSuite package software. WorkSuite is a powerful tool that is used for configuration and programming of all RTU32 Series products. Applications developed and running in the RTU32N will easily run directly on the other RTU32 Series products. The full suite WorkSuite software package is included in the product package, and it will run in demo mode without any license (with some runtime restrictions).

2.3   Drivers and runtime options

On The RTU32N you can easily activate additional driver runtime licenses. In case you need an additional driver you can just order it and activate it using a software license. For example if you have ordered just the IEC60870 Protocol suite and then you decide to also use IEC61850, you can send a purchase order mentioning the RTU ID.
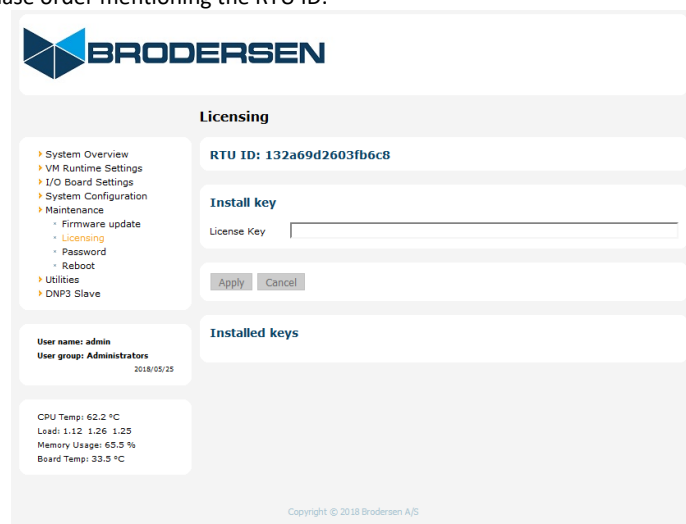
Figure 1

After receiving the activation code you can activate the desired driver on your RTU32N. It is the same for runtime options like number of tags and even for the processor speed.

See pricelist for the part numbers.

2.4 Ordering a Driver or utility

If a driver needs to be installed by user follow the below steps

1. Got to the RTU webpage like shown in chapter 2.3
2. Copy the RTU ID
3. Send a mail to sales@brodersen.com with the RTU ID and the need drivers and features.
4. After reception of the license code from Brodersen, paste the code on the same page as shown in chapter 2.3 and press apply.

The licensed drivers will appear on the main screen under system overview, see below picture.

Drivers installed

Modbus Master/Slave	Licensed
IEC 60870-5-101 Slave	Licensed
IEC 60870-5-104 Client	Licensed
IEC 60870-5-104 Server	Licensed
DF1 Master	Licensed
DNP3 Master	Licensed
DNP3 Slave	Licensed
IEC 61850 Client	Licensed
IEC 61850 Server	Licensed

Copyright © 2018 Brodersen A/S

3 Getting Started

3.1 Configuration and default settings

The LAN interface on the RTU32N is set to its factory default settings at delivery:

LAN1: 192.168.0.1, Subnet: 255.255.255.0

You can use the LAN1 interface to configure the basic settings in the RTU32N. The basic settings include setup of network parameters such as IP address, Subnet mask, default gateway, time settings, general username and password for accessing the web server pages and file system with FTP etc.

The configuration is done via web pages in the RTU32N Web Server. You must use your normal web browser to setup the basic settings for the RTU32N. The web pages are optimized for MS Internet Explorer, Firefox and Chrome. If you are experiencing problems with using the latest MS Internet Explorer version, please enable Compatibility settings in the browser.

4 How to connect and prepare your PC to configure the RTU32N

4.1 Wiring

Use a cross-wired or standard patch cable to connect your PC Ethernet interface to the LAN1 Ethernet interface of the RTU. Apply power to the RTU – make sure that you connect the correct power supply voltage.

4.2   Setting up your PC Ethernet driver (Windows)

Your PC Ethernet driver must be setup to work in the same network segment as the LAN interface on the RTU32 you use. You must setup your PC Ethernet TCP/IP driver to work in the segment IP 192.168.0.x.

To setup the Ethernet driver select *start – Settings – Network Connections – Local Area Connections* and the following window will appear:
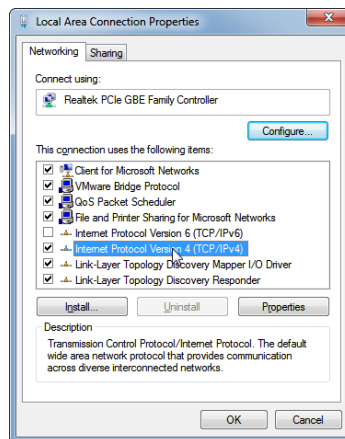
Click on Internet Protocol 4 (TCP/IPv4), and click *Properties*. A new window for setting the PC IP address will appear. Set an IP address in the same segment as the selected LAN IP address, for example 192.168.0.100 as shown here below.

NOTE: If you want to re-configure your Ethernet TCP/IP driver back to initial settings, it is recommended that you note the settings before you change them for setting up the RTU32N.
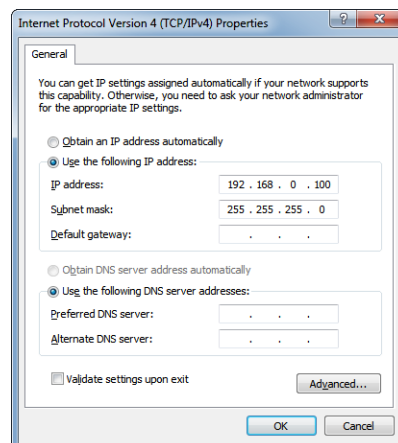
Click *OK* and *OK* again at the Local Area Connection Properties window. Now your PC Ethernet driver is setup for communicating with the RTU32N. Now apply power to the RTU32N. Wait for the RTU32N to boot up – takes approx. 20s. As a minimum the green Power LED on the front PANEL must be lit.

4.3   First time configuration

Start your Internet browser and enter the IP address of the RTU32N in the address field:

Now the first web page will appear in your browser and you will be asked to enter a password.

Please store the entered password in a safe place for future use.
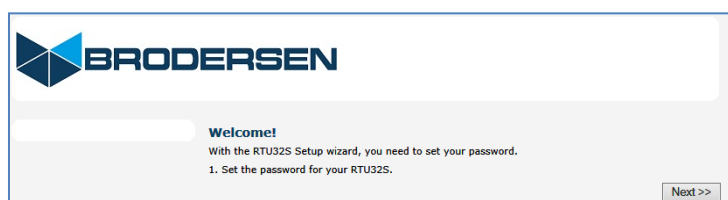
Click NEXT to continue.

Now you are asked to enter your administrator password. Note that you can always change it later. Username is always fixed = admin.

After typing in your password twice, select Finish to continue and save the settings.

The RTU will then restart and after ~30s you will get asked to login to the RTU32N web server as shown below.

The User name is fixed = admin. Enter your password.

This is the login you will meet every time you try to login in the future. After login in you will get to the first configuration web page.

5   Configuration parameters in the web pages

After the first time setting up the password, the main configuration page of the RTU32N will appear in your browser. Next time you connect to the web pages you will be asked to enter credentials to get access.

5.1   Settings overview

The first page that you access when you enter the RTU32N Web pages is the Settings Overview page:

BRODERSEN

Settings Overview

Settings Overview
Hardware Overview
Network Settings
Modem Settings
VM Runtime Settings ▷
I/O Board Settings
Time Settings
Change Password
Edit Config File
Boot
DNP3 Slave ▷

User name: admin
User group: Administrators
                    V2014/04/23

Local Area Network (LAN) settings

This section displays a summary of your LAN network settings. These settings indicate the current configuration of your LAN ports.

Network Settings LAN1	NUC900
Obtain an IP address via DHCP	DISABLED
Local IP Address:	192.168.1.221
Subnet mask	255.255.255.0
Default gateway:	192.168.1.1
Prefered DNS server	83.136.89.6
Alternate DNS server	192.168.1.100
MAC address	00-04-5f-93-49-b4
Host name	RTU32S

Project Information

Project Name	RTU32SIOTest
Project Runtime Start	2014/07/08 10:32:51.912
Project Build Time	2014/07/08 08:33:53
Project Build Version	7
Project Build CRC	16#67023e9a

RTU32S Version Information   1.55.1.0/1.18.10B1

RTU32S Firmware Date (RTU32S.exe)	2014/07/04
RTU32S Firmware ID (RTU32S.exe)	92746 W910
VM Runtime Version (RTU32.exe)	8.5.121210
RTU32S CE Image Date (NK.bin)	2014/07/03
RTU32S Windows CE Version:	6.0
RTU32S Windows CE Build Version:	1937

Drivers installed

IEC60870-5-101 Master/Slave	RTU32.exe	1.55.1.0
IEC60870-5-103 Master	RTU32.exe	1.55.1.0
IEC60870-5-104 Client/Server	RTU32.exe	1.55.1.0
Modbus Master/Slave	RTU32.exe	1.55.1.0
DNP3 Slave	T5BusExBDNP3SL3.dll	3.1.52.482

Copyright © 2014 Brodersen A/S

On this page you will get an overview of the network settings and versions of the primary system files and drivers. If you are contacting your support office or distributor, you will always be asked for the software version numbers.

The Drivers installed section lists the activated drivers in your RTU32N. The full Modbus Suite and IEC60870 Drivers are active in all RTU32N. Additional drivers like DNP3 Slave are installed if the Driver License is ordered with the RTU32N.

5.2 Hardware Overview

The hardware overview page shows you online the current I/O configuration status. It lists the physical connected I/O Expansion modules and you are able to read the status of each input or output. In the example below the RTU32N is equipped with internal I/O and input 0, 2, 3, 4and 6 are activated, value of analogue are also displayed.

5.3 Network settings

On the network setting page you can change the LAN1 settings to fit your local network. You must assign fixed IP addresses to gain access to the RTU32N with your browser, FTP client or Brodersen WorkSuite in your LAN network. If you do not know about the IP addresses and Subnet masks in your local network, please contact your local network manager or IT department for help.
You can also change the RTU Host Name. Default the host name of the RTU is RTU32N.

After entering new settings, click Apply to save. Note that the new settings will NOT be activated before you reboot the RTU32. Use the Boot function on the menu at the left side of the page.

5.4 HTTP, HTTPS, SSH

At the webpage under System Configuration/Services can HTTP, HTTPS, SSH be enabled or disabled.

5.5 Network Redundancy settings (HSR/PRP)

The Network redundancy settings must be done on the webpage. Go to "System configuration" and select Network Redundancy Below picture appears. Select in Dropdown box if network is HSR or PRP.

5.5.1 HSR selected

- Select HSR
- Select the correct HSR version
- HSR/PRP supervision "Specifies the last byte of the multicast address used for HSR/PRP supervision"
- Enter the IP Address that will be common for both LAN interfaces.
- Enter subnet mask or use default
- HSR clone MAC address Specifies whether the eth0 adapters MAC is used for both LAN adapters, this is only supported by HSR only

5.5.2 PRP Selected

- Select the correct PRP
- HSR/PRP supervision "Specifies the last byte of the multicast address used for HSR/PRP supervision"
- Enter the IP Address that will be common for both LAN interfaces.
- Enter subnet mask or use default
- HSR clone MAC is only supported by HSR.

5.5.3 Network Bonding

- Select the Network Bonding
- Select correct mode in drop down(Active Backup, Balance-RR, Balance-XOR, Broadcast) see below picture

- Enter IP address, default gateway, DNS information.

5.5.4 Adding USB Ethernet adapter

Important, use only Brodersen USB Ethernet adapter. After inserting USB Ethernet adapter into RTU, go to webpage. The new LAN adapter will show as ETHERNET LAN 3. Must be configured same way as ETHERNET LAN 1, and ETHERNET LAN 2

5.5.5   Modem Settings

If you use the Brodersen UCM-94 3G/GPRS modem you must configure the RTU32N to work with the modem.

In order to ensure the modem is working you will need to enter all credentials correctly. The webpage include small help icons. Just put the cursor on them and you will get details for each point.

5.5.6 Modem control from Logic

If you want to control the connection from the PLC application, you must select Connection Control from application. And then in your PLC application use the Function Block 'Connect3G' (placed in the Dial-up function group).
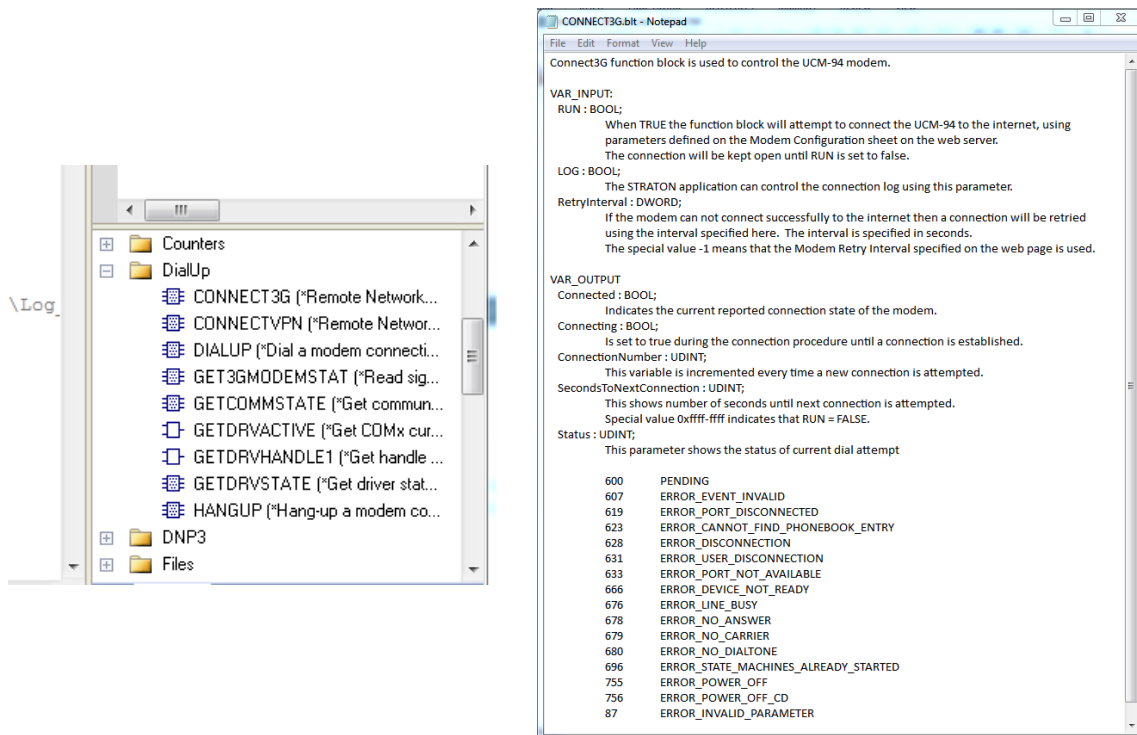
Note: The Settings Overview web page will include the modem connection status at the bottom of the web page if the modem is connected. It will also show you the IP address assigned on the 3G/GPRS network.

5.6 Runtime Settings / Main Settings

Under the menu item VM Runtime Settings can the PLC runtime settings be adjusted.
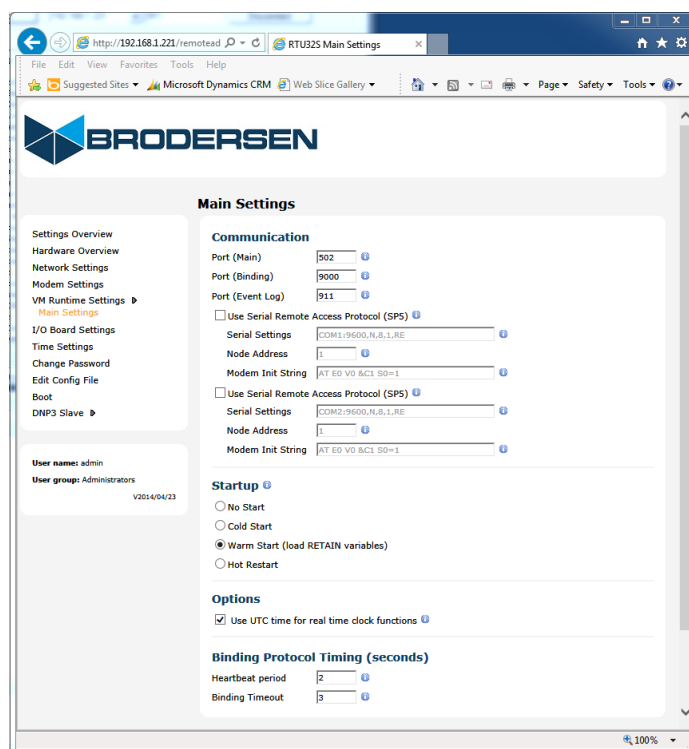
5.6.1   Communication

Configuration of basic start-up settings, communication ports and enabling of Remote Access Secure Protocols. Hold the cursor over the small icon   a you will get online help.

Port (Main)
The Main port is used for Brodersen WorkSuite communication for configuration and debugging. Default setting is 502.
This port is also used for the ModbusTCP Server.

Port (Binding)
The Binding Port is used by the event based binding protocol, where two or more RTU32 and PC with T5 runtimes can exchange data. Default setting is 9000.

Port (Event log)
The Event Log port is where the RTU32N System Log can be viewed - either by connecting using telnet or the Brodersen Event Viewer tool. Default setting is 911.

Enable Serial Remote Access Protocol (SP5)
This protocol is to be used for remote serial communication from the Brodersen WorkSuite and is particularly well adapted to communication systems with severe constraints such as radios and modems. e.g. if you have the RTU32N placed on a remote site connected via serial communication using multi-drop radio or line connection – or even dial connection, you are able to remotely connect to the RTU32N with the Brodersen WorkSuite and work online as if it was on your workshop table. The speed will however be slower depending on your actual serial communication speed.
It is possible to attach this protocol to two communication ports.

Serial settings
The syntax is: [M]COM{n}:{b},{p},{d},{s}[,h][,LL] and the parameters in brackets [ ] are optional:
[M] Use modem connection
{n} Number of the COM port to be used (1 … 8)
{b} Allowed speeds: 300,600,1200,2400,4800,9600,19200,38400,57600,115200
{p} Parity: N,E,O (None,Even,Odd)
{d} Data bits: 7 or 8
{s} Stop bits: 1 or 2
[h] RTS/CTS handshaking where h is either RD,RE,RT:l:t or RC:l:t. In the COM port section you will find details of how to use these parameters.
[LL] Log all bytes received and transmitted into the system log (port 911)

The following example shows setting for communicating on COM1 port, at 19200 baud, with Even parity, 8 data bits and 1 stop bit:
COM1:19200,E,8,1
Here a modem usage is enabled: MCOM2:9600,N,8,1
Here logging is enabled: COM1:9600,N,8,1,LL

Node address
Node Address (or Unit number) identifies the RTU32 on a multi drop network. The default number for RTU32N on a peer to peer network is 1. The address must be a value from 1 to 32767.

Modem Init String
Here you enter the initialization string (AT commands) when using a modem.
RTU32N default is: AT E0 V0 &C1 S0=1

5.6.2   Start-up

The start-up option specifies how the PLC application is started when the RTU32N is powered up. The possible options are:
        No Start – PLC application is NOT started when booting the RTU32.
        Cold Start – PLC application is started without loading retained variables.
        Warm Start – PLC application is started and retained variables are loaded (default).
        Hot Restart – Only supported with battery backed RAM option and in Redundancy configuration!

5.6.3 Options

Use UTC time for PLC real time functions
If enabled, it instructs all PLC real time functions (e.g. DTCurDate, DTCurTime and day_time) to return time in UTC. UTC time is not adjusted according to daylight savings. If this parameter is disabled then time is returned in local time which is adjusted to daylight saving if this option is enabled.

NOTE: Drivers that are implemented via PLC Functions – like the IEC60870 Driver – will use the PLC time stamp for events! Other drivers may take standard time or whatever time is defined in the applicable standards.

PLC Main Settings / Binding Protocol Timing (seconds)
When using binding on slower network connection like some optical fibre modems or GPRS/3G wireless connections, you will need to adjust the timing for the Binding driver.

The possible timing adjustments are:
    Heartbeat period – how often a heartbeat is sending to check the binding link connection.
    Binding Time-out – time-out on link check and general binding event.

5.7 I/O Board Settings

On this page you can:
- Read the actual I/O configuration on the RTU32.
- Lock an I/O configuration
- Adjust the settings for configurable inputs and outputs.

5.7.1 Current I/O Configuration

Here the actual I/O configurations are listed. In the example below you will see that the configuration of I/O boards include 28IO and a connected I/O Expansion module with 16DO/16DO, more options are available.
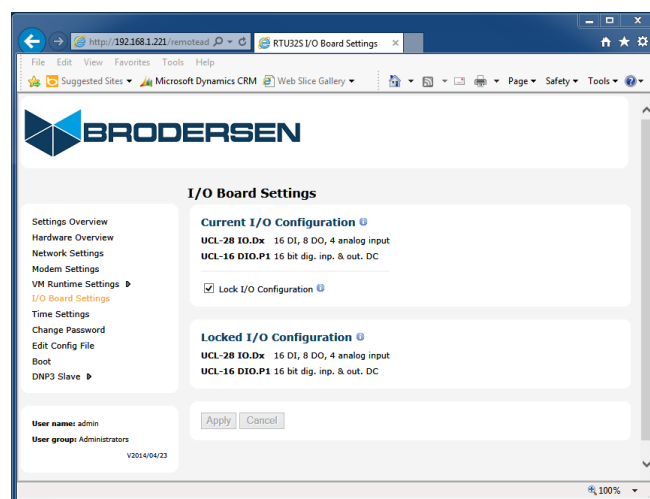
By default the I/O configuration is unlocked. When unlocked the actual I/O configuration is read every time the RTU32N is booted and when the PLC Runtime application is started.

By enabling lock I/O configuration, the actual I/O configuration is locked and stored, and NOT read at boot and/or application start-up. Use the Lock function if you want to be able to trace a disconnected or defective I/O module in the RTU32 System Log.

5.8   Time Settings for RTC – Real Time Clock

The time settings web page is used for configuring and adjusting the RTU32N real-time clock. PTP, NTP, SNTP

5.8.1   Time Settings

The RTU32N realtime clock can be configured from the Time Setting page.

I is possible to define if the RTU should be synchronized via a SNTP/NTP Server, or you can use other synchronization options like manual time setting or synchronization via utility drivers that include this functionality, such as IEC60870 or DNP3.

NOTE: The time shown on the webpage is running in software and is only updated when the page is refreshed.

When set for synchronizing to a SNTP Server you must define the SNTP/NTP Server IP address or domain name. If you use a domain name, you must ensure that you have defined a DNS Server address in the LAN1 configuration.

Update interval time can be set from 1 sec to 20160 minutes.

5.8.2 RTU time zone

If you use local time you have to set the time zone correctly in order to get the correct time stamps. By default it is setup for Central European Time (CET). You can also select to support automatic daylight saving adjustment. In this case it is following the standard adjustment procedures. Be careful to take this in to consideration if you are working with IEC60870 drivers and use Clock Synchronisation.

5.9 Change password

Change the password for your main login.
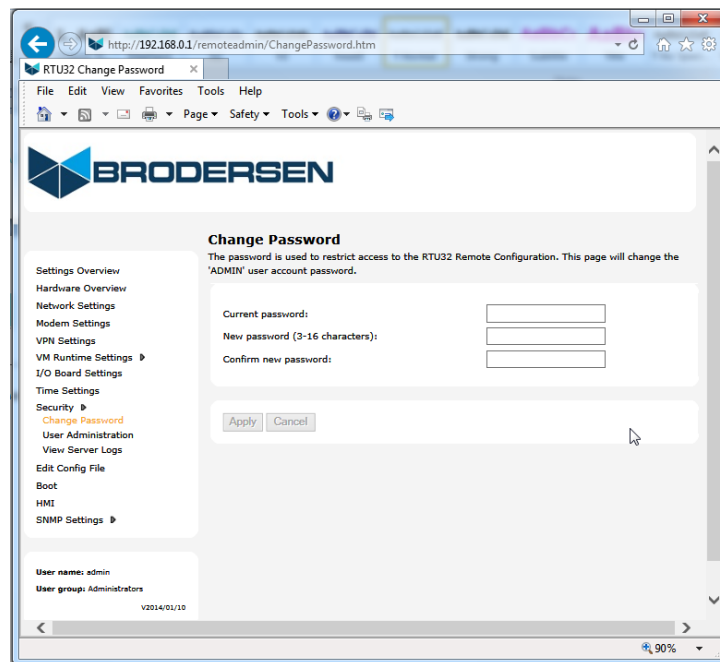
Note: the username and password covers access to all Servers in the RTU32N – including HTTPS, SSH

5.10 LDAP Authentication Service

If you want to use a central authentication service, it is possible to configure the RTU32 to utilize the Lightweight Directory Access Protocol to authenticate users against a remote Directory Server. It is configured in the menu "System Configuration" -> "Services".
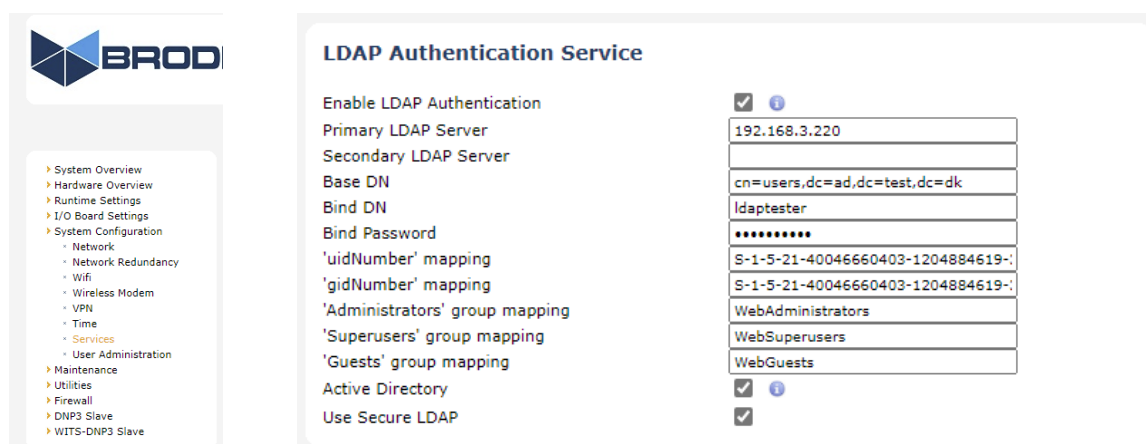
Primary LDAP Server:
IP address or FQDN of the Directory Server used as primary server.

Secondary LDAP Server:
IP address or FQDN of the Directory Server used as secondary server (Backup server if primary fails).

Base DN:
Specifies the root for searches in the Directory. Ideally, it should match the root of your Directory.

Bind DN:
Specifies the credential you are using to authenticate against a Directory Service when searching in the LDAP tree.

Bind Password:
Specifies the password used for the credential specified in the 'Bind DN'.

'uidNumber' mapping:
Specifies the object name used in the Directory Service for 'uid', i.e. User Id.

'gidNumber' mapping:
Specifies the object name used in the Directory Service for 'gid', i.e. Group Id.

'Administrators' group mapping:
Specifies the group name in the Directory Service which users must be member of in order to access the RTU32 with 'Administrator' privileges.

'Superusers' group mapping:
Specifies the group name in the Directory Service which users must be member of in order to access the RTU32 with 'Superuser' privileges.

'Guests' group mapping:
Specifies the group name in the Directory Service which users must be member of in order to access the RTU32 with 'Guest' privileges.

Active Directory:
Specifies whether the Directory Service is a Microsoft Active Directory Service.

Use Secure LDAP:
Specifies whether secure communication over TLS or SSL is used

5.11   Boot

After changing parameters on the configuration pages, you must unless otherwise noted, run the reboot function to activate the changes. After booting the RTU32N you can start working with Brodersen WorkSuite to create you application programs.

Brodersen A/S, Islevdalvej 187, DK-2610 Roedovre, Denmark, Tel: +45 45 35 26 27, Fax: +45 45 35 26 29, Email: sales@Brodersen.com   22

6 Getting Started with Brodersen WorkSuite tools

6.1 Introduction

The Brodersen WorkSuite Package includes all software required to configure and program the Brodersen RTU32 Series of products – including RTU32, RTU32S, RTU32R, RTU32N and RTU32E.

The software components included in the package include:

- Brodersen WorkSuite PLC configuration and programming tool.
- WIBUKey driver for the USB license key dongle for WorkSuite.
- RTU32 Series tool package including:
    - o Event Viewer – the telnet tool used for monitoring the System Log(s) in RTU32 Series products.
    - o RTU32 Search tool for scanning and finding RTU32 products running on networks.
    - o RTU32 Series Remote Desktop tool
    - o RTU32 Update tool for managing software updates for many RTUs over network connections.

All software is installed in one installation process. The installation process will ask you what software to install and after making your selection, the installation process will run automatically and install and configure the all software, making it easy for you to begin working with configuration and programming of the Brodersen RTU32 Series products.

6.2 Installing Brodersen WorkSuite Package

Go to our website and find the WorkSuite Installation Guide. It will guide you through all the installation details.

6.3 WorkSuite License

The WorkSuite tool requires a license to configure and program RTUs. Without a license the software package works in DEMO mode.

The license is locked to a USB hardware dongle with is included in the package. In the package you find also the WorkSuite License Certificate with License id number required to enter in the License utility after installing the software.

The standard license offers the possibility to create an unlimited number of I/O points in the RTU application program. All variables used for drivers are counted as I/Os.

6.4 User documentation and training/guideline videos

When installing the Brodersen WorkSuite package, User Guides and Manual are installed as well. You get access to these documents from the Help menu within the WorkSuite tool.

WorkSuite has also implemented a help function that means that you can get help by pressing F1. If no valid help opens, please check the PDF manuals available under the general WorkSuite Help menu or check our website http://brodersen.com/videotraining for documentation.

We recommend that you use the Brodersen Training/Guideline Videos. They are found by visiting:

www.youtube.com/brodersensystems
or
http://brodersen.com/videotraining

Here you will find step by step guidelines with details of how to get started and how to setup drivers etc.

We keep on adding more videos. So be updated with our newsletter to keep up sign up here.

IEC60870-5-101:
IEC60870-5-101 Slave
IEC60870-5-101 Master

IEC60870-5-104:
IEC60870-5-104 Server
IEC60870-5-104 Client

IEC61850:
IEC61850 Server
IEC61850 Server with Goose Messages
IEC61850 Client

DNP3:
DNP3 Client
DNP3 Server
DNP3 Master
DNP3 Slave

Modbus:
Modbus Master Ethernet
Modbus Slave Ethernet
Modbus Master Serial
Modbus Slave Serial

Miscellaneous:
Binding Protocol
Firmware update RTU32
Getting started with RTU32
MQTT Client
SNMP Extension Agent
RTU32 Redundancy Mode

7 IO & Display

7.1 Configure on-board

There are different types of built in I/O. General for them all is that if it is intelligent I/Os they can also be setup at the webpage. Or through Brodersen Worksuite. If there is no built in I/O, there is no display on the RTU. Go to the RTU webpage and select I/O board settings, and select the settings required for your project.

There are different types of I/O combinations:

16 IO   16 Digital Input, 16 PNP Output
28 IO   16 Digital Input, 8 PNP Output, 4AI
42 IO   16 Digital Input, 8 PNP Output, 6AI, 8DIO(configurable input or output), 4x Fast DI can each be used as DI or 2kHz counters.

- Digital inputs, 10-30VDC (all DI channels count)
- Digital outputs, 10-30VDC (High Side Switched)
- Fast DI/counter inputs, 5-30VDC, up to 2 KHz
- Analog Inputs (16bit), multi-range:
  0-5V, 1-5V, 0-10V, 0-20mA, 4-20mA

7.1.1 Local Display

The local display is for showing the status of all types of I/O, and also to give the field engineer the possibility to see the analogue values on the RTU. The display turns automatically of if more than 1 min idle. If any button on the front of RTU is pressed, LCD automatic turns on.

- **Vin:**   **Voltage input to RTU**
- **TMP:**   **Is the temperature inside RTU**
- **Type:**   **42-IO A indicates the type is a RTU with 42 built in I/O**
- **INT Bus ACTIVE**   **Indicates expansion modules on the bus is ACTIVE**
- **INT Bus PASIVE**   **Indicates expansion modules on the bus is deactivated, or cable defective**
- **AI**   **Gives the value of the selected analogue channel**
- **AI0 !OVF!**   **Selected Analogue channel 0 has Overflow**
- **AI0 !UDF!**   **Selected Analogue channel 0 has Underflow**
- **DI**   **Gives status of selected inputs**
- **DO**   **Gives status of selected output**
- **CT**   **Gives value of selected counter.**

The LCD auto scrolls through IO and RTU status displays. After 60s user inactivity, the display backlight turns off. backlight. Press Enter or Escape to halt/restart the auto scroll. When the scroll is halted, the up and down arrows can be used to navigate through the various displays. Note: if extra IO modules are added to the RTU, the additional values are then included in the display list. Press any key to invoke the

7.1.2 Configure onboard I/O

Onboard I/O can be configured on the webpage or in the Brodersen worksuite. The below example has also 8 DIO, so each pin can be either input or output, and is selected in drop down menu. Intelligent I/Os can be set up frim Webpage. LB2 I/O modules can also configured from worksuite see document 40430 LB2 User manual at Brodersen webpage.

8   Adding backplane I/O modules to RTU32N

8.1   INTRODUCTION to LB2 I/O series
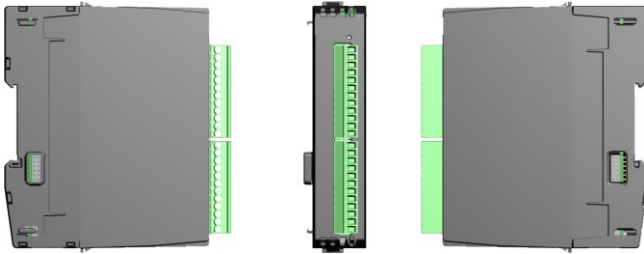
The Brodersen LB2 modules can be used with RTU32N & RTU32M series. The I/O modules are in two parts, bottom part containing the backplane bus, and top part containing the I/O board and logic. All LB2 modules are hot plug. LB2 modules are all equipped with 200 MHz processor to process I/O, handle filtering, SOE, debounce, module clock and general module logic. Diagnostic variables are available for all LB2 modules. See the module datasheets for full info. Firmware update is handled from RTU level in Brodersen worksuite.

Use only genuine Brodersen bus cables for connection to Brodersen RTUs and extension of I/O module blocks. The connection cables for LB2 is special made to handle the power requirements and shielding to run communication. The maximum overall length of complete system is 5m. Each I/O module & Power supply module is calculated as 2 cm. The cables are as the length indicates, e.g. UCC-610/1 count as 100 cm.

Maximum possible system configuration is 250 I/O modules on one LB2 Bus.

8.1.1   LB2 Cables

Description	Part Number
100cm LB2 Cable	UCC-610/1
200cm LB2 Cable	UCC-610/2
25cm LB2 Cable	UCC-610/25
50cm LB2 Cable	UCC-610/50

8.1.2   IO MODULE BACKPLANE PART

Description	Part Number
Bus I/O module Start	**BB-1LS.10**
BUS I/O module, Middle	**BB-1LI.10**
BUS I/O module, Expansion (End)	**BB-1LE.10**
BUS Power supply module, Start	**BB-1PS.10**
BUS Power supply module, Middle	**BB-1PI.10**
BUS CPU module, Start	**BB-1CS.10**
BUS CPU module, Middle	**BB-1CI.10**

9   Module configurator in Worksuite LB2 modules

All modules are configured in the Brodersen worksuite. It is recommended to see our video about how to configure and insert LB2 modules in I/O system . It can be found on our webpage or by pressing this link: VIDEO

9.1   Module configurator Toolbar

Figure 2 shows the configurator toolbar with icon explanations.
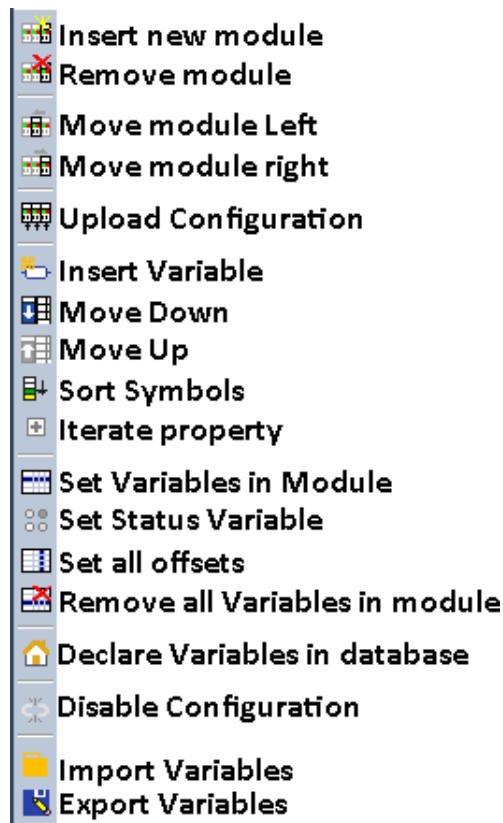
Figure 2

9.2 Inserting a module in Brodersen Worksuite

To open the I/O configurator press the ICON in the top menubar "open I/Os"

The I/O configurator is now open.
Modules can be inserted in several ways.
- Left click in the I/O module window and select "Upload Configuration" This will upload modules connected to the online RTU.
- Left click in the I/O module window and select ADD Module
  Drag and drop from module list in the right side of I/O module window
- A existing module can be copied by "Ctrl+c" & "Ctrl + V" → Windows copy paste
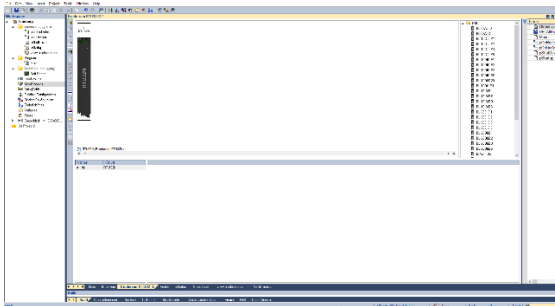
Figure 3

9.3 Browsing I/O into Worksuite

To open the I/O configurator press the ICON in the top menubar "open I/Os"   and press upload modules.
Press ok and the I/O are implemented and graphical interface generated.
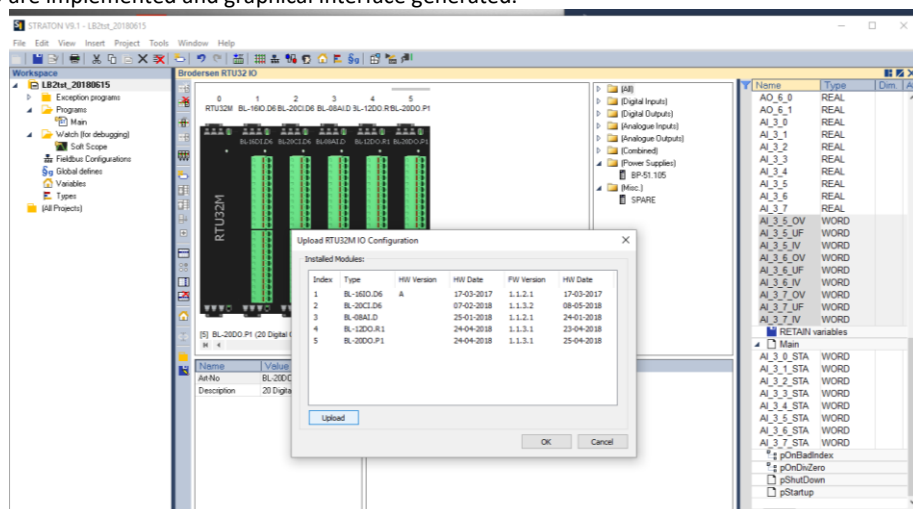
9.4 Move a module in Brodersen Worksuite

- Modules can be moved left or right by pointing the mouse cursor on the module, press and hold left mouse button and drag module to desired position.
- Modules can be moved left or right by pointing the mouse cursor on the module, press right mouse button and select "Move Right" or "Move Left"

9.5 Remove a module in Brodersen Worksuite

- Modules can Removed/Deleted by pointing the mouse cursor on the module, press right mouse button and select "Remove Module"
- Modules can Removed/Deleted by pointing the mouse cursor on the module and press Delete on keyboard

9.6 Add variables to a module in Brodersen Worksuite

In left pane of the I/O modules press the Icon ⌗ "Set Variables In Module" a Set variables window appears. Remember to check Declare Variables in Database if Variables are to be used later as Global Variables.
Use the Drop Down list to select a syntax, or make you own.
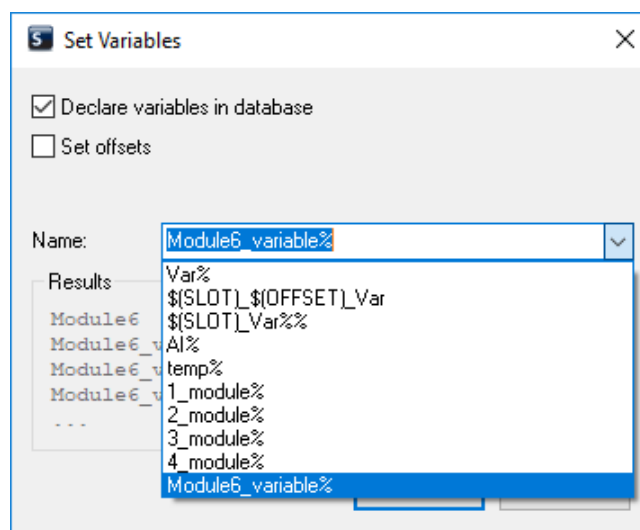
Figure 4

If as an example the syntax $(SLOT)_$(OFFSET)_VAR is selected the variables will be like in Figure 5 for a 8 Analogue input module.

Figure 5

The way the syntaxes are built, give the flexibility to engineer your own. If the syntax is modified it could be:

Module_Nr_$(SLOT)_Analouge_in_$(OFFSET)

And the result would be see Figure 6

Figure 6

9.6.1   Rename auto generated variables

In some cases there are some variables names that can't be generated through a syntax. In this case point the mouse on the variable see Figure 6 where the line is blue. Double click on it. And Figure 7 appears
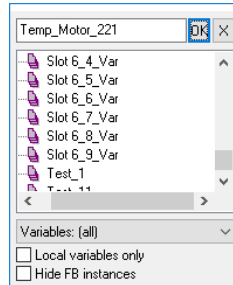
Figure 7

Now the variable name can be changed to temperature of motor 221 e.g. Temp_Motor_221, press OK and Figure 8 appears.
Remember in Figure 8 to select
"Rename the Variable"
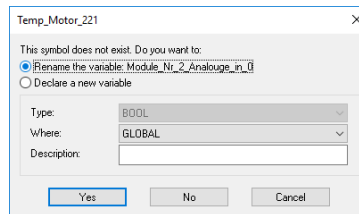
Figure 8

9.7 Configure Modules

Use the datasheet of the modules to see what specific features there is available on the module. In this example a combination module and an analogue input module will be shown.

In the module configurator see Figure 9 DoubleClick on a module
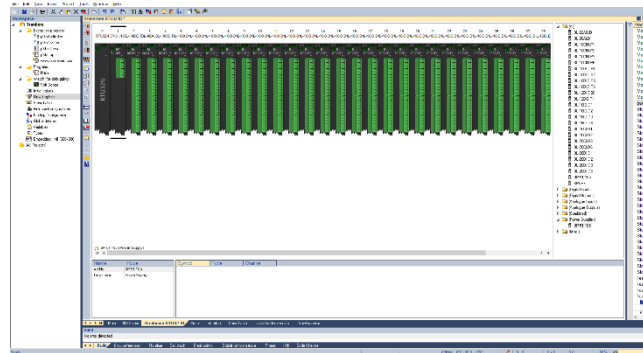
Figure 9

After DoubleClick on the module the configuration page appears. See Figure 10.

9.7.1 Configure Combination DI/DO module example
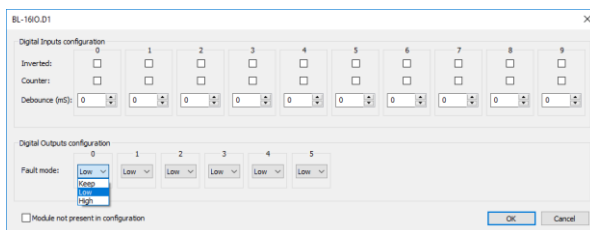
Figure 10

In this illustration a Combination module with 10 DI and 6 DO has been selected.

- Configure Input:
  - o **Inverted:** Select if input is Low level inverted. Can also be inverted in the PLC logic.
  - o **Counter:** Select if input is a counter. If an input is a 5 Khz counter, but not defined as a counter. Then the bus will be spammed with 5000 pulses/sec. equal to a pulse every 200 μs.
  - o **Debounce:** Debounce in milliseconds, determines how long time the I/O must be high or low before it is detected as state change.
  - o **Fault Mode** Fault mode is where module has no contact with a CPU. It is necessary to select for all channels.
    - o Keep (I/O module will keep last state)
    - o Low (Output will be forced Low)
    - o High (output will be forced High)

9.7.2 Configure AI module example

In the module configurator see Figure 9 DoubleClick on a module
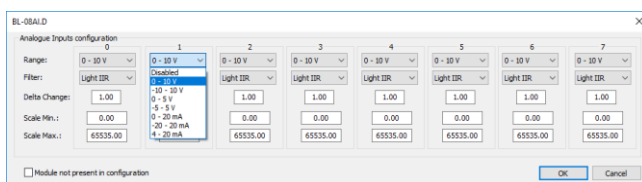
Figure 11

After DoubleClick on the module the configuration page appears. See Figure 11

- Configure Analogue Input:
  - o **Range:** Select input type for each channel in dropdown menu.
  - o **Filter:** select filter type for each channel in dropdown menu. See Figure 12
  - o **Delta change:** See AI module datasheet
  - o **Scale Min:** See AI module datasheet
  - o **Scale Max:** See AI module datasheet
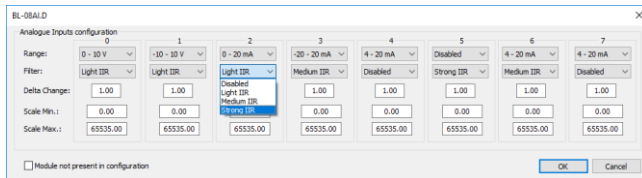
Figure 12

9.8 Export/Import Module variables.

9.8.1 Export:

Select the I/O module and press Export Variables, see Figure 2. Now the CSV export file can edited in EXCEL.
Scale min, Scale Max, Variable name etc.

	A	B	C	D	E	F	G	H
1	NAME	TYPE	OFFSET	SCALEMIN	SCALEMAX	HWMIN	HWMAX	#F
2	4_module1	IX	0	0.0	65535.0	0	65535	-1
3	4_module2	IX	1	0.0	65535.0	0	65535	-1
4	4_module3	IX	2	0.0	65535.0	0	65535	-1
5	4_module4	IX	3	0.0	65535.0	0	65535	-1
6	4_module5	IX	4	0.0	65535.0	0	65535	-1
7	4_module6	IX	5	0.0	65535.0	0	65535	-1
8	4_module7	IX	6	0.0	65535.0	0	65535	-1
9	4_module8	IX	7	0.0	65535.0	0	65535	-1
10	4_module9	IX	8	0.0	65535.0	0	65535	-1
11	4_module10	IX	9	0.0	65535.0	0	65535	-1
12	4_module11	QX	0	0.0	65535.0	0	65535	-1
13	4_module12	QX	1	0.0	65535.0	0	65535	-1
14	4_module13	QX	2	0.0	65535.0	0	65535	-1
15	4_module14	QX	3	0.0	65535.0	0	65535	-1
16	4_module15	QX	4	0.0	65535.0	0	65535	-1
17	4_module16	QX	5	0.0	65535.0	0	65535	-1

9.8.2 Import:

Press the Import Variables see Figure 2. Select the CSV file and import.

9.9 Firmware update I/O modules

Firmware update of LB2 modules is done through Worksuite, see LB2 user manual. Worksuite automatically detects if new firmware are available for the configured I/O modules. There must be internet connection to the PC where Worksuite is installed. Worksuite downloads the firmware from Brodersen Domaine. Firmware update is only possible through Worksuite to avoid errors in the process. All connected I/O modules can be firmware updated in the same process.

10 WorkSuite I/O Board Driver for UCL series I/O modules

The I/O Board driver makes it possible to declare I/O in sections as per the physical board layouts. I.e. it is possible to add all 32 digital inputs in one process using the I/O Boards driver. At declaration it automatically declares the variables and puts them in the variable list.

Procedure for declaring board I/O:

- Select Open I/O Board Configurator (1)
- Double click on the section 0 to add your first board (2)
- From the list of I/O board select the one you want (3)
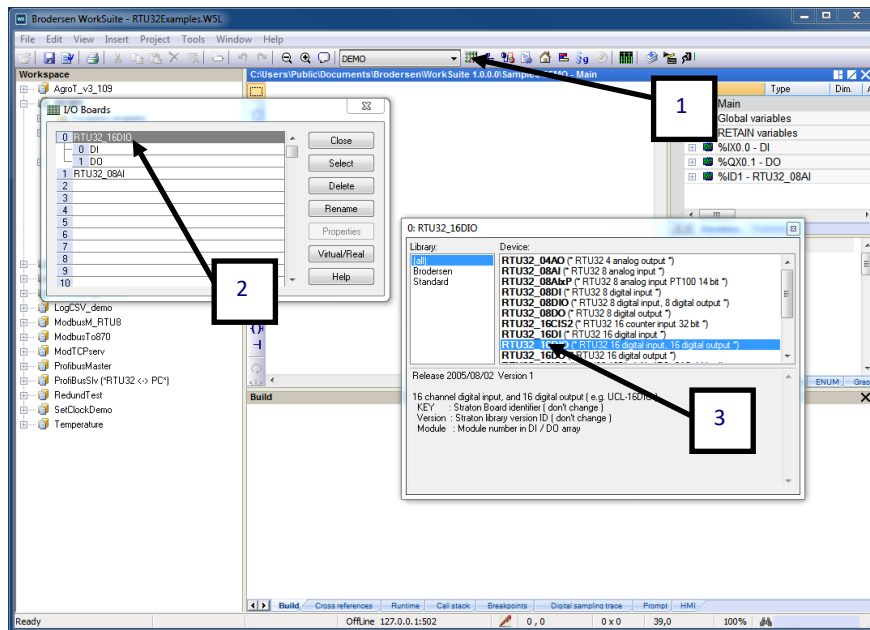
After the board is selected you will see it in your list. Now select properties to configure the board. If the board you have selected has different types of I/O they will be shown individually in their own line. Every data type requires configuration.
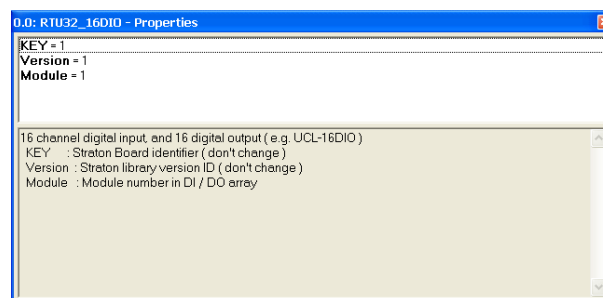
The properties for digital I/O are simple. Only the Module address needs to be setup. Use the addressing example to determine the Module address in your application.
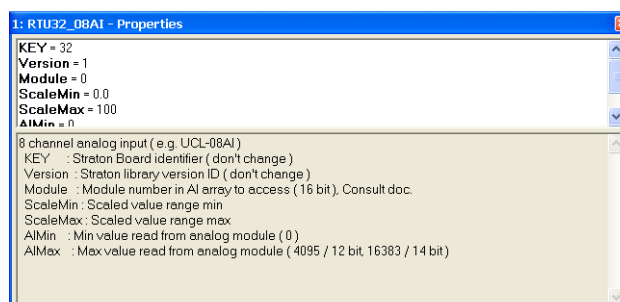
The properties for analogue I/O require configuration. Besides the Module address of the first analogue channel, you can configure the scaling of the analogue I/O. Please note that the scaling will cover all I/O in the actual board section. On some board I/O it is possible to set the range of every input and output individually (for Internal I/O and newer analogue expansion modules).

After setting up the boards, you will find all the board I/O in the variables list.
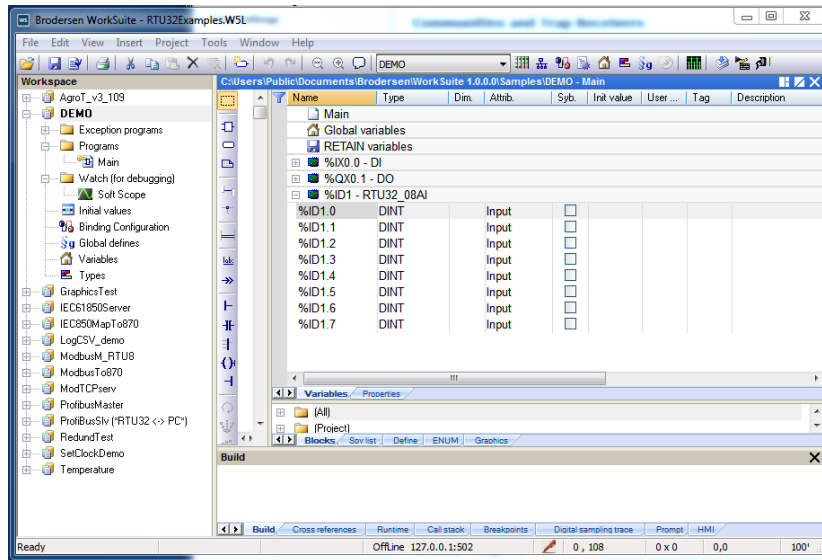
The I/O is now ready to be used in your runtime application program.

For some of the analogue I/O you will find some qualifier inputs which are used for status information. They report underflow and overflow of the analogue value.

10.1   WorkSuite I/O Profiles Driver

The I/O Profile driver is used for declaring I/O one by one directly in the variable list – or by using the Profile Editor. Any digital or analogue I/O can be declared this way. If you want status from RTU32 basic database, you must always use the I/O Profile driver.

10.1.1   Option 1 - Declaring I/O Profiles in variable list

In the Variable list you can add new variables under the Global variable structure (right click and select 'Add Variable'). It is recommended to use the Global Variable structure to allow access to the I/O from any program in your WorkSuite project.

After creating a new variable, double click on the variable name and enter an appropriate name for your I/O – e.g. DI1_0 for the first digital input bit in the second DI module. Set the variable type to WORD or USINT.
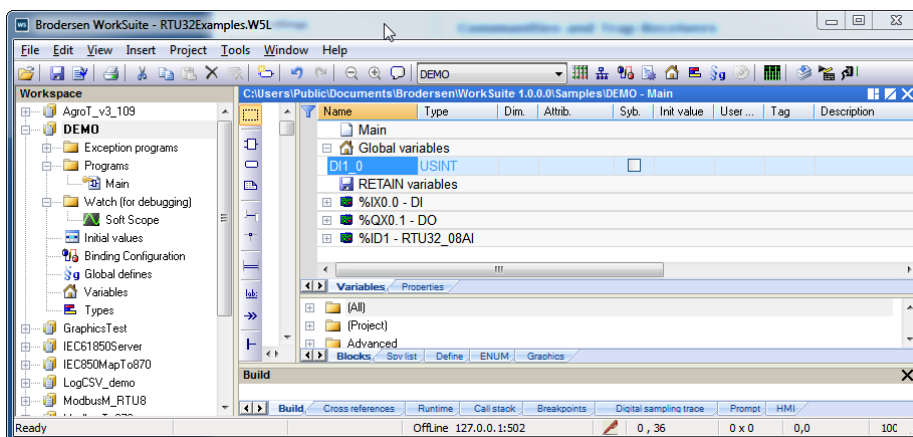
Right click on the DI1_0 variable and select Properties:
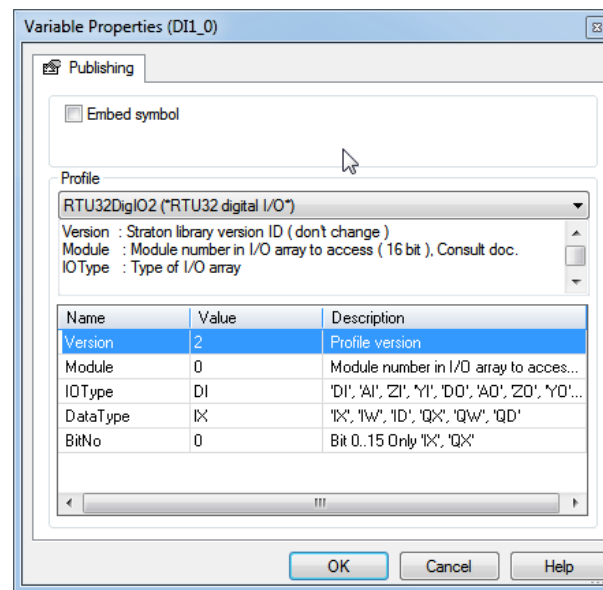
Select "RTU32DigIO" for digital I/O, set Module = 1 (second digital input module (read: Word)).
Select "DI" for Digital Input register.
Select "IW" for Input Word.
Say OK to store the settings.

Now the physical I/O is defined as a variable and can be used in your application program. If you want a single bit value you can either declare the variable as a bit (IX) or you can just write "DI1_0.2" to read the 3rd bit value.
The same procedure is used for reading the status words.

10.1.2   Option 2 - Declaring I/O Profiles using the Profile Editor

Another and often much faster way to declare an I/O Profile for RTU32 Series I/O is to use the Profile Editor in Brodersen WorkSuite.

The Profile Editor is a more user friendly way to setup several drivers including the Brodersen I/O LocalBus driver.
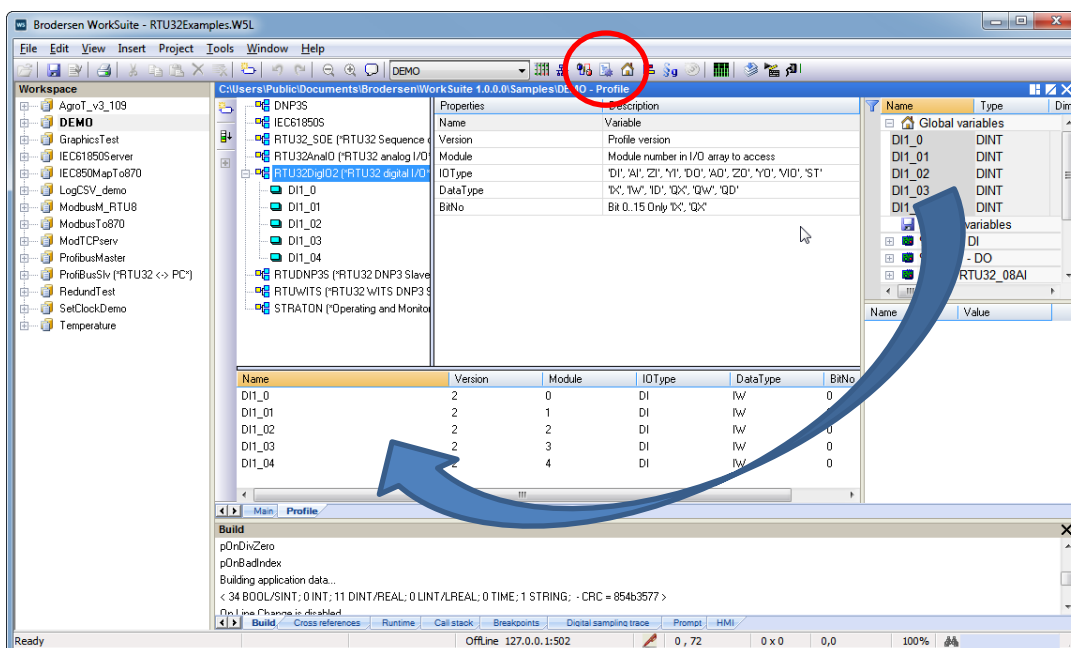
As shown in example above the Profile editor gives you a better overview, and it is easy to edit the declared variables. It also supports drag and drop from the variable list. This means that you can create the variables first and then drag and drop complete lists of variables to the Profile I/O as shown above. See WorkSuite Help for more details.

10.1.3   Counter inputs

Some I/O boards have support for counter inputs. Auxiliary inputs (ZI) are used for reading 32bit counter inputs. The counter values are stored in registers (modules) and can be reset by using a virtual output (ZO).

Reset of counters is done on an output raising edge. Make sure that the reset output is held ON to ensure that that one I/O scan has been performed. This could be done by using a timer in the WorkSuite application program.

11 COM Port Settings

When using serial drivers the COM port parameters are defined as "COM1:9600,N,8,1". In addition some specific control and debug parameters can be added for the RTU32N.
The COM port setting strings follow the normal WorkSuite conventions. COMa:b,c,d,e (e.g. COM1:9600,N,8,1).

a) COM port number:	1...3	
b) Baud rate:	300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200	
c) Byte size:	7, 8 (bit)	
d) Parity	N, E, O (none, even, odd)	
e) Stop bits	1, 2	

A range of additional parameters on the COM port settings are supported to control:

- Setup the COM port for modem dial support – enable the Modem driver.
- Hardware handshake signal on null modem driver.
- Provide communication log in the general RTU32N System Log.

The extensions for these functions and features are:

- When using the modem driver a 'M' is added (e.g. MCOM1:9600,N,8,1)
- RTS / CTS hardware handshake control on the null modem serial driver is supported and enabled by the options as follows:
  'RD'   RTS is kept inactive (low) at all time.
  'RE'   RTS is kept active (high) at all time.
  'RT:Leading:Trailing' RTS is inactive when receiving data, and becomes active when transmitting data. The RTS Leading setting defines the delay from activating the RTS to when the first character is transmitted. The RTS Trailing setting defines the delay from when the last character is transmitted to RTS is deactivated.
  Leading and trailing time is in units of msec. - e.g. COM1:9600,N,8,1,RT:50:10

  'RC:Leading:Trailing' RTS is inactive when receiving data, and is activated when the RTU wants to transmit data. After activating RTS, the RTU will wait for CTS to become active, before start transmitting. The RTS Leading delay is still valid in this mode, and an adjustable delay from CTS activated to first character is then possible. However, by setting the Leading time to zero there is no unnecessary delay from CTS to first character (like a normal RTS / CTS function). After activating RTS the RTU waits up to 10 sec for the CTS signal. If timeout occurs, the transmission is discarded. The RTS Trailing setting defines the delay from when the last character is transmitted to RTS deactivated.

- Logging of serial data in the System Log.

  'LL'   Log Link data. All bytes transmitted and received on the COM port, are logged in the internal log message system, and are available using telnet at port 911. This is primarily for debug purpose.
  Examples: MCOM3:9600,N,8,1,LL
                     COM2:57600,E,8,1,RT:30:10,LL

Note! If you just connect a serial device, e.g. ModbusRTU communication via a Null Modem cable to the RS232 COM port, it is NOT required to add any special extensions.
Note! Brodersen WorkSuite has a limitation of max 31 chars in the settings string.

If you enable the SP5 (Secure Protocol for remote communication) on the RTU32 Webpage, you must set the COM parameters settings for the specific port on the webpage. If the SP5 setting is enabled, the COM settings overrule any later settings in WorkSuite.

The serial port settings use the same syntax no matter where you configure it.

11.1 Modbus Drivers

The RTU32N PLC runtime supports function for several types of Modbus protocols:

- ModbusRTU Master and Slave
- ModbusASCII Slave
- ModbusTCP Client and Server

The drivers are setup using the Fieldbus Configurator in WorkSuite. Refer to WorkSuite training and programming examples. You will find support for ModbusUDP, and Function Blocked handled Modbus drivers in the WorkSuite help.

11.1.1 ModbusRTU Master

The RTU32N supports several Master/Client drivers. You can in fact setup as many Master drivers as you have COM ports available. The Modbus Master/Client is setup using the Fieldbus Configurator. See the WorkSuite help for details.

11.1.2 ModbusRTU Slave

The RTU32N supports only one Slave/Server driver. You must add a MBSLAVERTU function block in the program logic for the Serial Modbus Slave driver (MBSLAVERTUEX function is NOT supported – supported in other RTU32 Series products). The MBSLAVERTU function defines the Modbus Slave address and port settings. You can find the function block in the "Advanced" function block folder. All variable assignments, addressing etc. are done in the Fieldbus Configurator. See WorkSuite for details.

11.2 IEC60870-5-101/103/104 Drivers

The Utility protocols IEC60870-5-101, 103 and -104 are implemented as a basic link driver in the RTU32N. The application layer which includes all the specific handling of request and events communication is managed in the RTU application program. A range of WorkSuite functions and example are provided for the different protocol functions and ASDU handling.

WorkSuite include drivers for:

- IEC60870-5-101 Master
- IEC60870-5-101 Slave
- IEC60870-5-103 Master
- IEC60870-5-104 Server
- IEC60870-5-104 Client

The basic link drivers includes handles for the link driver where you can define buffers, link address etc. The fact that the serial link is similar to the IEC60870-5-103, provides the option to implement application layer support for this protocol as well.

To use the IEC60870 driver, you must use the RTU32 Series Code Generator tool. This is used for simple configuration of any IEC60870 Driver.

11.3 Watchdog

The RTU32N has two built-in watchdog functions – all managed by a separate CPU on the interface board.

The system watchdog is monitoring all internal processes including the OS and communication processes between consoles and databases. In case of error the watchdog will wait 30sec for the error to be either self-healed or the process to be restored. If this is not happening the watchdog will boot the RTU.

The application watchdog is monitoring the RTU/PLC runtime applications including all related tasks such as drivers. If any of the related tasks fail, stop or freeze for more than 10sec, the application watchdog will boot the RTU.

11.4   Real time / Real time Clock

The RTU32N has a hardware real time clock controlled by the operating system. It provides the RTU32N millisecond resolution in order to make accurate time stamps. From the RTU application program you have access to the real time clock via some special functions as shown below. You can both read and set the clock.
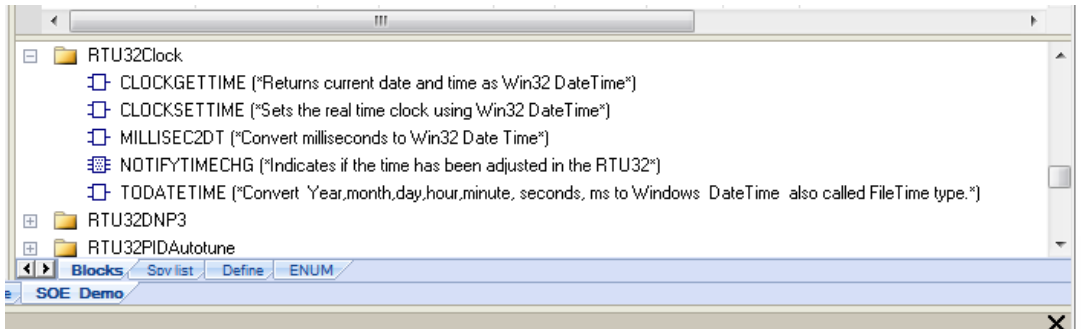
On the configuration webpages you can also define how to synchronise the clock. Options for manual setup and SNTP Server synchronisation are possible. If you use utility drivers to synchronise the RTC you must disable SNTP on webpage by setting it to manual sync.

12   12 Modems

12.1   Modem Functions for UCM-94 3G/GPRS Modem Series

WorkSuite provides a range of modem control Functions. The DialUp library includes:

- Functions to control and monitor a UCM-94 3G/GPRS modem.

- Functions to control and monitor A PSTN/GSM modem dial-in/out

12.1.1   Dial functions for GSM and PSTN Modems

Dial functions for communication with serial drivers over PSTN and GSM modems are provided.

The low level modem handling (send AT commands, connect detection etc.) is done by firmware. However it is up to the application programmer to decide which protocol driver should be activated, when and where to dial, and respond to the modem connection state. A number of WorkSuite functions are available for this:
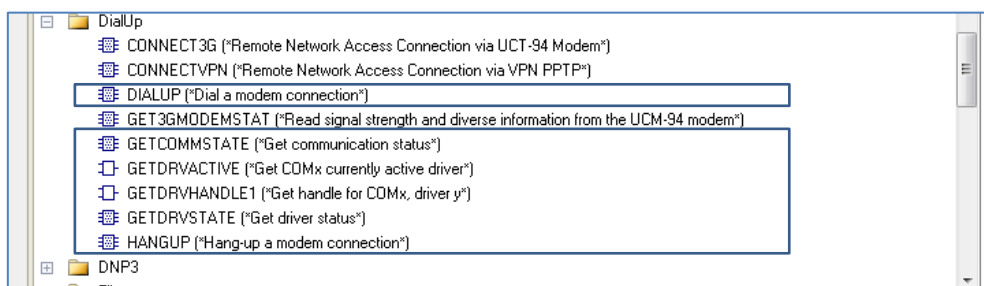
The modem function is enabled by adding M to the COM settings. See details in COM settings section.

After enabling the modem dial driver, you must add a Modem driver handle in your application program (GetDrvHandle). Your dialup function must now refer to this handle. The DialUp function includes all the necessary parameters to establish and control a dial-up or dial-in session.

GetDrvHandle (Get driver handle).

This function is used to get the handle of built-in protocol drivers, like Modbus RTU master or slave. This should be used when operating a built in protocol driver (e.g. dialing). To obtain the handle, COM port number and driver number must be provided.
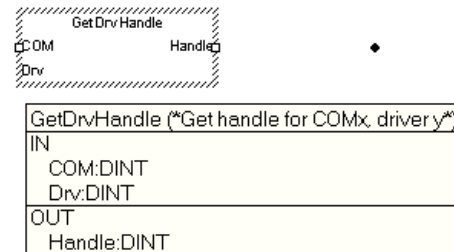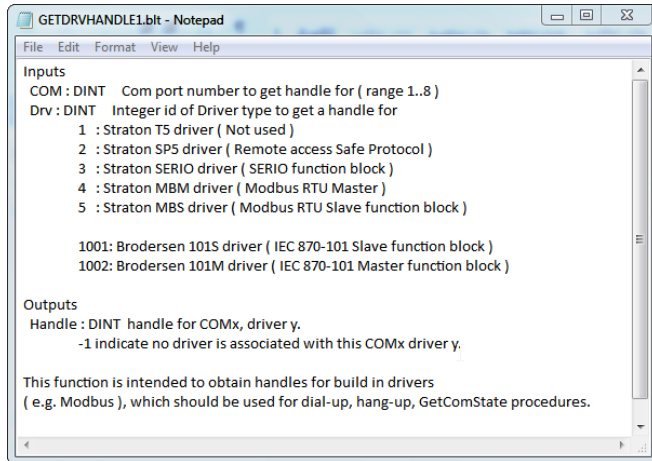
Dialup - Start a dial-up connection for one of the protocol drivers.
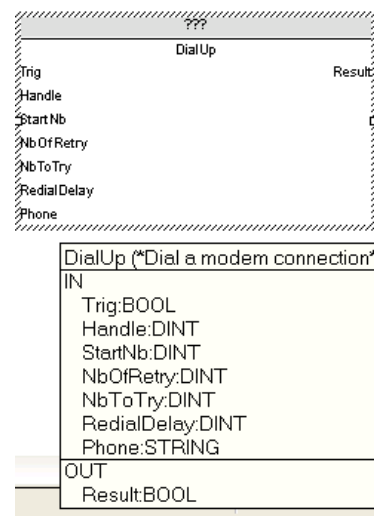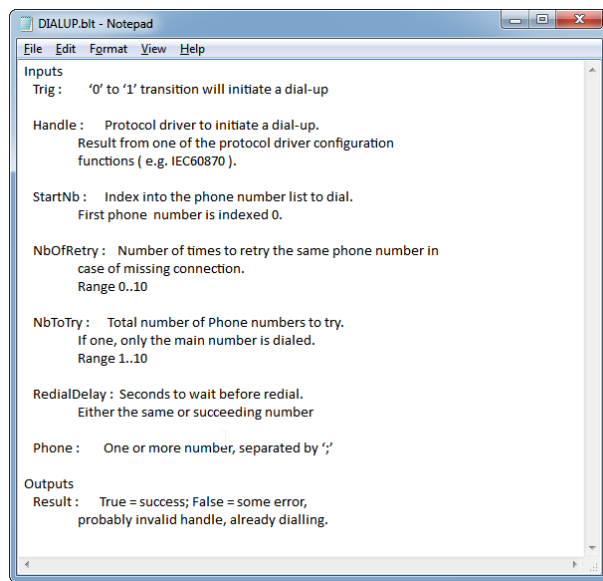
NOTE: One instance of this function block must be declared for each modem driver (physical modem) used.

HangUp - Hang-up current connection
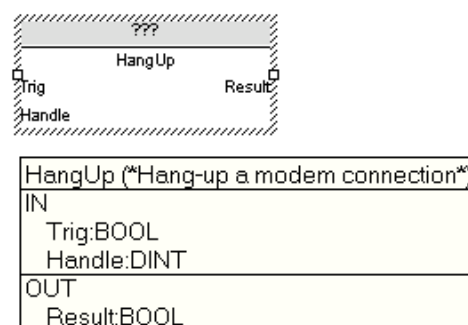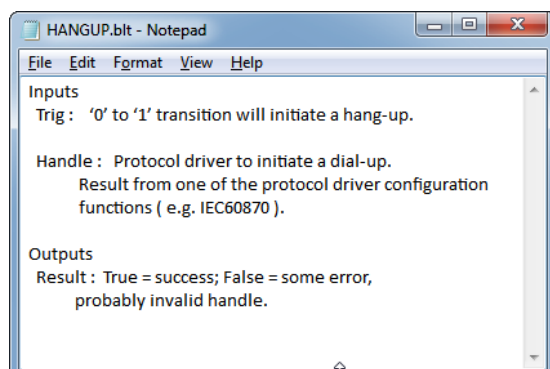
NOTE: One instance of this function block must be declared for each modem driver (physical modem) used.

GetCommState - Get communication status.

```
GETCOMMSTATE.blt - Notepad
File  Edit  Format  View  Help

Inputs
 Handle :  Serial driver to get communication state from.
         Result from one of the protocol driver configuration
         functions. ( e.g. IEC60870 )

Outputs
 Result :  True = success, return parameters are valid.
         False = some error, probably invalid handle.

 State :  Bit 0..3
         1. Idle       Driver is waiting for action.
         2. Init       Initializing modem.
         3. InitError  Not able to initialize modem ( keep trying ).
         4. Dialing    Driver is dialing a number, and waiting for connect.
         5. Con. Out   Outgoing call, Modem is successfully connected,
                       driver is running.
         6. Con. In    Incoming call, Modem is successfully connected,
                       driver is running.
         7. NoCon.     No connection. The dialed number was busy,
                       did not answer etc. Waiting to redial.
         8. Hang-up    Modem is hanging up the connection.


         Bit 4..6      Not used.

         Bit 7 Con. Err Dial-up procedure failed.
                 No connection was made to main, or any
                 sub phone numbers. Dial is suspended.
                 The bit is reset when a hang-up and new
                 dial-up is executed, or an incoming call is received.

         Bit 8..31     Not used

 TlfNr :  Telephone number in the dial-up string,
         which are currently active.

 HandShake: bit 0      CTS   ( in )     Clear To Send
         bit 1      DSR   ( in )      Data Set Ready
         bit 2      RI    ( in )      Ringing Indicator
         bit 3      DCD   ( in )      Data Carrier Detect
         bit 4      DTR   ( out )     Data Terminal Ready
         bit 5      RTS   ( out )     Request To Send
         Bit 6..31     Not used

 RxByte :  Number of serial bytes received.

 TxByte :  Number of serial bytes transmitted.

 RxFrame :  Number of protocol frames successfully received.
         Used to detect communication is running.
         In modem mode the counter is reset to zero when line is hang-up.

 TxFrame :  Number of protocol frames successfully transmitted
         Used to detect communication is running.
```


Note: This function is primarily intended for use with a modem driver, but could also be used to monitor data transmission of null modem drivers. Data returned when using multiple protocol drivers on the same COM port, is more port related than protocol driver related. Obtaining RxFrame values using handles from either passive or active drivers on the same COM port will return the same value.

Check your RTU32 CD or our homepage for useful examples of how to use the modem drivers.

GetDrvActive - Get active driver
This function will return the ID of the currently active driver (if any) on a COM port. It can be used by the RTU application to identify which protocol is presently active.

GETDRVACTIVE.blt - Notepad

File   Edit   Format   View   Help

```
Inputs
 COM : DINT    Com port number to get activr driver from ( range 1..8 )

Outputs
 Drv : DINT    Integer id of the currently activated Driver
        -1 : No driver is currently active
         1 : Straton T5 driver ( Not used )
         2 : Straton SP5 driver ( Remote access Safe Protocol )
         3 : Straton SERIO driver ( SERIO function block )
         4 : Straton MBM driver ( Modbus RTU Master )
         5 : Straton MBS driver ( Modbus RTU Slave function block )

        1001: Brodersen 101S driver ( IEC 870-101 Slave function block )
        1002: Brodersen 101M driver ( IEC 870-101 Master function block )

This function will identify the currently active driver ( if any )
at COM port x.
```

GetDrvState - Get driver status
This function can be called regularly by the RTU application to get the status of a driver.

GETDRVSTATE.blt - Notepad

File   Edit   Format   View   Help

```
Inputs
 Handle :  Serial driver to get communication status from.
       Result from one of the protocol driver configuration
       functions. ( e.g. IEC60870 )

Outputs
 Result :  True = success, return parameters are valid.
       False = some error, probably invalid handle.

 Active :  True = Driver is currently activated.
       False = Driver is currently deactivated.

 RxFrame :  Number of protocol frames successfully received by
       this protocol driver.
       In modem mode the counter is reset to zero when line is hang-up.

 TxFrame :  Number of protocol frames successfully transmitted
       by this prorocol driver.
       In modem mode the counter is reset to zero when line is hang-up.

Note! RxFrame and TxFrame values are protocol driver specific,
and will differ from GetCommState FB RxFrame / TxFrame values,
in multi protocol setup.
```

12.1.2 Functions for control and monitoring the UCM-94 3G/GPRS Modem

Control and monitoring functions are available for controlling and monitoring the UCM-94 family of modems used for 3G/GPRS communication.
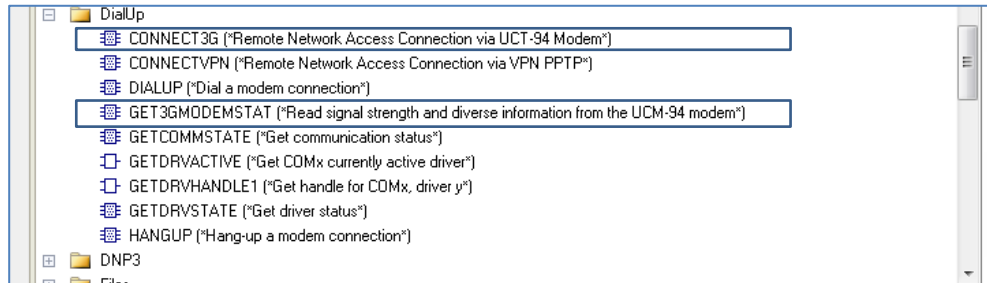
The UCM-94 modem connect procedure can be done automatically at start-up or controlled in the PLC program using Functions listed shown below. Automatic or PLC controlled connection is defined on the modem configuration webpage. If automatic connection is used, the monitoring Function is still applicable to use for monitoring the connection status.

The low level modem handling (send AT commands, connect detection etc.) is done by firmware.

The available Functions are listed below.

Connect3G Function for PLC control

The Connect3G function is used when the PLC application controls the Modem connection. If automatic connection is selected on the RTUs web page this Function is not used.

Get3GModemState Function

The Get3GModemState function can be used for monitoring the state of the UCM-94 modem and the 3G/GPRS connection status.

GET3GMODEMSTAT.blt - Notepad

File Edit Format View Help

Function Get3GModemStat.

This function is used to read different status information from the UCM-94 3G/GPRS modem. The modem needs to be connected to the RTU and enabled on the configuration page in the web-browser.

The initial values are -1, which means that the variable has not received any value. Some of the status information is scanned with 20 seconds interval. Others are reported spountanously by the modem.

Input parameters:
  Run : BOOL;
    When RUN is true the function continuously reads the status information from the modem. The Run parameter should be false for at least 40 seconds when the modem is repowered.
  Log : BOOL;
    If Log is true, then different information regarding the scanning of the modem is sent to the System Event Log.

???
GET3GMODEMSTAT
Run   ModemEnabled
Log   StatusPortOpen
                             ModemRegistered
                             SignalStrength_0_31
                             SignalStrength_0_5
                             NetworkRegistration
                             NetworkService_0_2
                             NetworkService_0_4
                             Operator

GET3GMODEMSTAT.blt - Notepad

File Edit Format View Help

Output parameters:
  Following status information is available:

  ModemEnabled : BOOL;
    This indicates if the UCM-94 modem is enabled or not on the configuration page on the web server and the Run parameter is true

  StatusPortOpen : BOOL;
    Returns true when the function is connected to the status port of the modem.

  ModemRegistered : BOOL;
    Returns true if the modem is registered in registry. If modem is not registered then there will be no attempt to read the other parameters. Also Connect3G function should only be set in Connect state when this is true.

  SignalStrength_0_31 : DINT;
    This status is scanned every 20 second using the AT+CSQ command. The signalstrength is reported by the modem in the range 0..31. Value 99 means error. Initial value is -1.

  SignalStrength_0_5 : DINT;
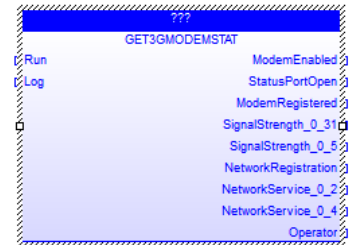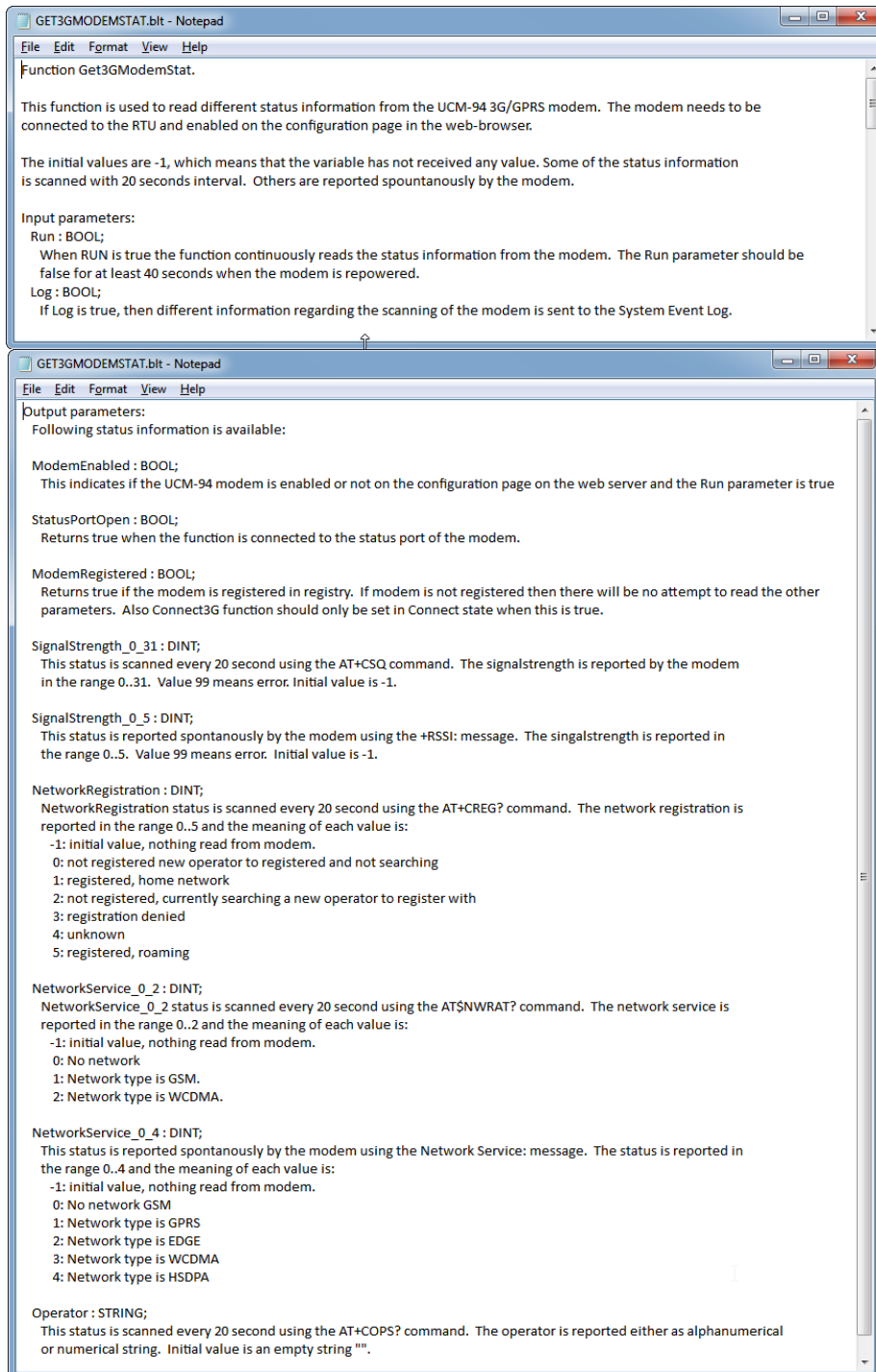   This status is reported spontanously by the modem using the +RSSI: message. The singalstrength is reported in the range 0..5. Value 99 means error. Initial value is -1.

  NetworkRegistration : DINT;
    NetworkRegistration status is scanned every 20 second using the AT+CREG? command. The network registration is reported in the range 0..5 and the meaning of each value is:
      -1: initial value, nothing read from modem.
       0: not registered new operator to registered and not searching
       1: registered, home network
       2: not registered, currently searching a new operator to register with
       3: registration denied
       4: unknown
       5: registered, roaming

  NetworkService_0_2 : DINT;
    NetworkService_0_2 status is scanned every 20 second using the AT$NWRAT? command. The network service is reported in the range 0..2 and the meaning of each value is:
      -1: initial value, nothing read from modem.
       0: No network
       1: Network type is GSM.
       2: Network type is WCDMA.

  NetworkService_0_4 : DINT;
    This status is reported spontanously by the modem using the Network Service: message. The status is reported in the range 0..4 and the meaning of each value is:
      -1: initial value, nothing read from modem.
       0: No network GSM
       1: Network type is GPRS
       2: Network type is EDGE
       3: Network type is WCDMA
       4: Network type is HSDPA

  Operator : STRING;
    This status is scanned every 20 second using the AT+COPS? command. The operator is reported either as alphanumerical or numerical string. Initial value is an empty string "".

The 3Gmodem status function provides status information as listed above. The field strength that indicates radio link level is submitted in two ranges. It is recommended to use the SignalStrength_0_31 output for monitoring the field strength. Compared to rssi dBm levels, the reported 0-31 range can be explained like this:

SignalStrength_0_31 value 0 = -113 dBm or less
SignalStrength_0_31 value 1 = -111 dBm
SignalStrength_0_31 value 2...30 = -109... -53 dBm
SignalStrength_0_31 value 31 = -51 dBm or greater
SignalStrength_0_31 value 99 = not known or not detectable

To have a proper radio link connection, the field strength on this scale should not be less than 20. Note that the field strength does not equal QoS on the actual 3G/GPRS network connection. This is dependent on the network provider's equipment status and load on the provider network.

NOTE: It is recommended to use scanned values only as the spontaneously reported values are not always reported (due to the network provider account settings and type of SIM card).

13 Data Logging

WorkSuite provides functions for logging data into one or more file(s) in the RTU32N. In general 2 functions are available:

1. LOGCSV Function - a dedicated RTU32 Series logging function. You can find an example of this being used in the WorkSuite Demo examples installed together with your WorkSuite Package. This data log function handles the read/write functions in a background scan and does not interfere with the general PLC cycle. You will find details of the LOGCSV file function below.

2. LogFileCSV – simple function for logging data into a CSV-file. It uses a standard file write function and it is not recommended to use for data logging as a standalone Function.
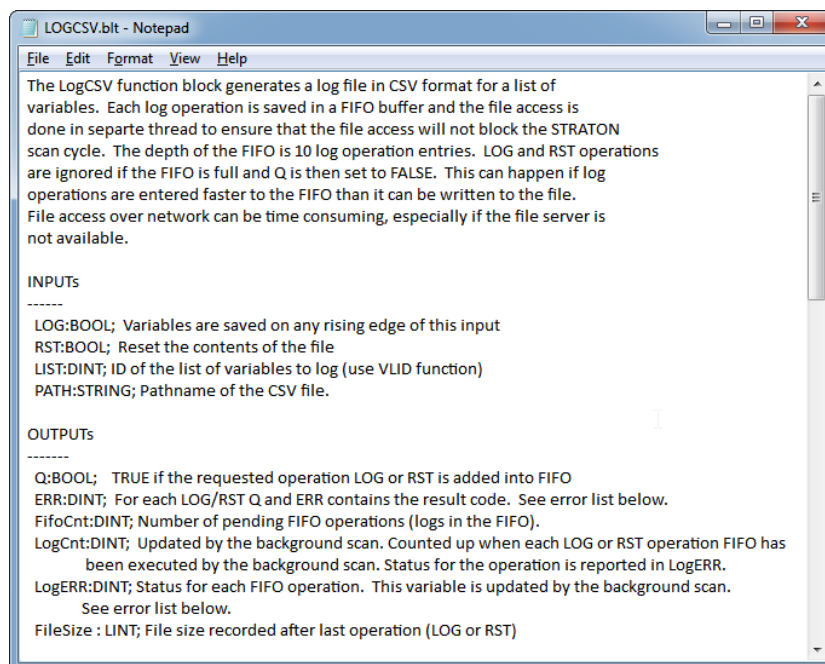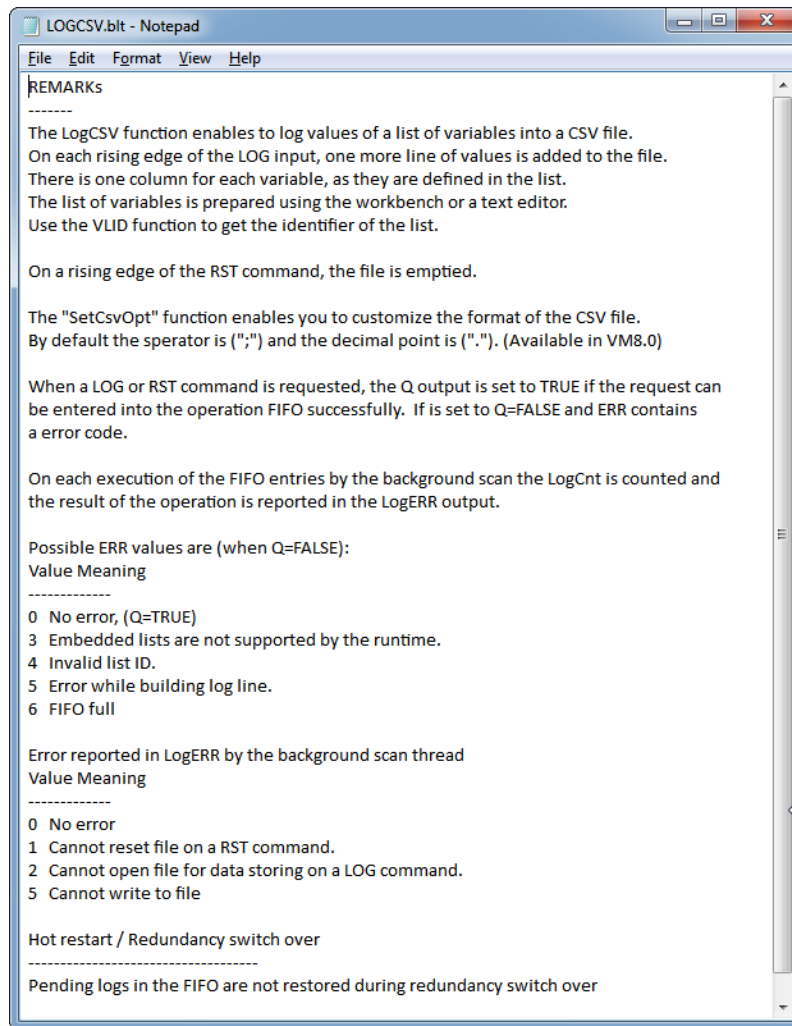
LOGCSV Function

The LOGCSV Function is found in the Brodersen Function library folder.

The LogCSV function block generates a log file in CSV format for a list of variables. Each log operation is saved in a FIFO buffer and the file access is done in a separate thread to ensure that the file access will not block the PLC runtime cycle. The depth of the FIFO is 10 log operation entries. LOG and RST operations are ignored if the FIFO is full and Q is then set to FALSE. This can happen if log operations are entered faster to the FIFO than it can be written to the file.

File access over network can be time consuming, especially if the file server is not available.

In the RTU32Example projects in WorkSuite you will find a WorkSuite Demo project LOGCSV_Demo with a log UDFB (User Defined Function Block) block named TMultiFileLog. With this UDFB you will be able to control the number of files to log and limit file size. This allows management of log files in a ring buffer, ensuring that your log files will not fill up the available storage memory.

13.1   Status information functions

RTU32N Series products have hardware status information available.
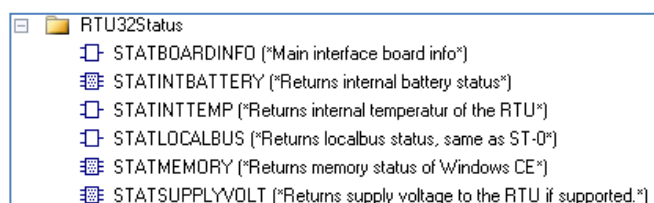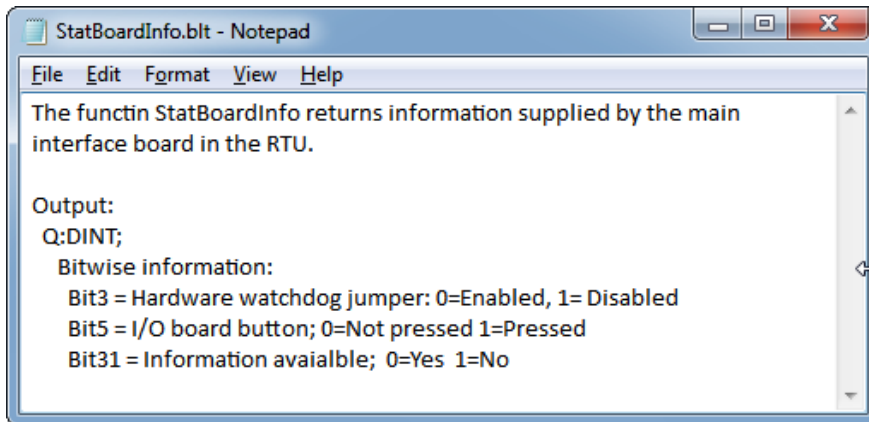

Only some of the status functions are available in the RTU32N.   The following sections list the functions supported by the RTU32N.
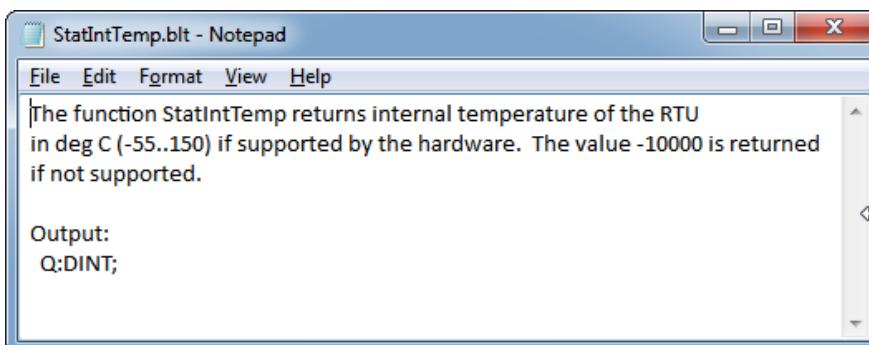
13.1.1 Interface Board Status Info (STATBOARDINFO)

The function StatBoardInfo returns information supplied by the main interface board in the RTU32N.
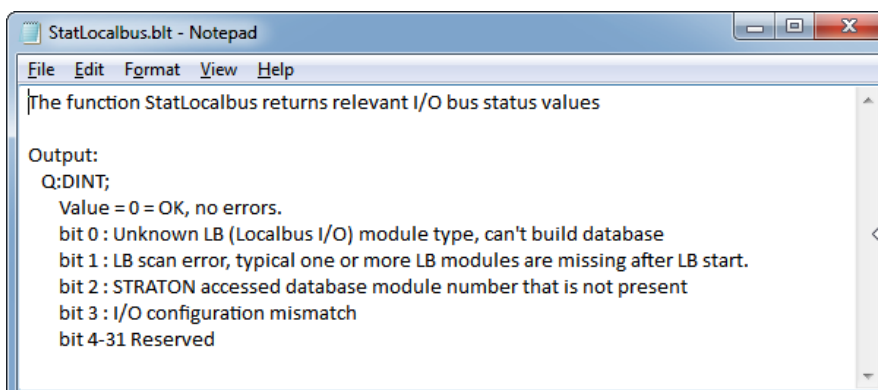
StatBoardInfo.blt - Notepad
File   Edit   Format   View   Help

The functin StatBoardInfo returns information supplied by the main interface board in the RTU.

Output:
  Q:DINT;
    Bitwise information:
      Bit3 = Hardware watchdog jumper: 0=Enabled, 1= Disabled
      Bit5 = I/O board button; 0=Not pressed 1=Pressed
      Bit31 = Information avaialble; 0=Yes 1=No

STATBOARDINFO
                Q   IntBoardInfo = 0

13.1.2 RTU Internal Temperature (STATINTTEMP)

The function StatIntTemp returns internal temperature of the RTU32N.

StatIntTemp.blt - Notepad
File   Edit   Format   View   Help

The function StatIntTemp returns internal temperature of the RTU
in deg C (-55..150) if supported by the hardware. The value -10000 is returned
if not supported.

Output:
  Q:DINT;

STATINTTEMP
                Q   IntTemp = 42

13.1.3 LocalBus Status (STATLOCALBUS)

The function StatLocalbus returns relevant I/O bus status values

StatLocalbus.blt - Notepad
File   Edit   Format   View   Help

The function StatLocalbus returns relevant I/O bus status values

Output:
  Q:DINT;
      Value = 0 = OK, no errors.
      bit 0 : Unknown LB (Localbus I/O) module type, can't build database
      bit 1 : LB scan error, typical one or more LB modules are missing after LB start.
      bit 2 : STRATON accessed database module number that is not present
      bit 3 : I/O configuration mismatch
      bit 4-31 Reserved

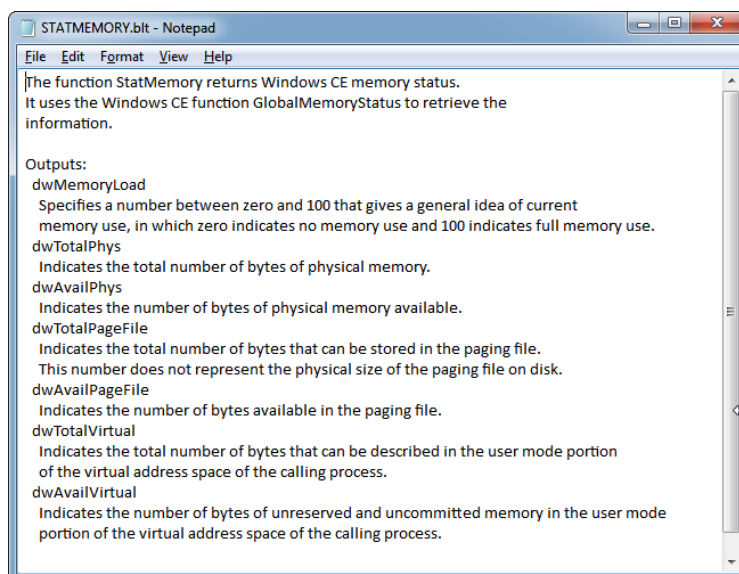If value of Word register = 0, then all is fine.

Here are some typical errors for the LocalBus;

Bit 0	Bit 0: if equal true then it is typically when an unsupported I/O modules is connected to the RTU.
Bit 1	Bit 1: Indicate LB scanning error – this is the typical error indication when an I/O Module fails or are disconnected in a running application.
Bit 2	Bit 2: Will be true if you in your runtime project address an I/O point that is not available in the database. Be careful if you use same project for several I/O configurations.
Bit 3	Bit 3: The I/O configuration mismatch will be true if the I/O configuration expected by the runtime is not present at start-up. If you have locked the I/O configuration this bit will go true at runtime start-up – and so will bit 1.

STATLOCALBUS
Q ── LocalBusStat = 0

13.1.4   RTU memory Status (STATMEMORY)

The function StatMemory returns memory status.

STATMEMORY.blt - Notepad

The function StatMemory returns Windows CE memory status.
It uses the Windows CE function GlobalMemoryStatus to retrieve the
information.

Outputs:
dwMemoryLoad
  Specifies a number between zero and 100 that gives a general idea of current
  memory use, in which zero indicates no memory use and 100 indicates full memory use.
dwTotalPhys
  Indicates the total number of bytes of physical memory.
dwAvailPhys
  Indicates the number of bytes of physical memory available.
dwTotalPageFile
  Indicates the total number of bytes that can be stored in the paging file.
  This number does not represent the physical size of the paging file on disk.
dwAvailPageFile
  Indicates the number of bytes available in the paging file.
dwTotalVirtual
  Indicates the total number of bytes that can be described in the user mode portion
  of the virtual address space of the calling process.
dwAvailVirtual
  Indicates the number of bytes of unreserved and uncommitted memory in the user mode
  portion of the virtual address space of the calling process.

The commonly used monitoring functions are dwMemory Load, dwTotalPhys and dwAvailPhys.

The dwTotalPhys tells you how much RAM program memory is available in the RTU. The rest of the RAM memory is used for storage (file system etc.). Memory can vary in size depending on the product code.

The dwAvailPhys tells you how much available program memory you have in your RTU. The rest is used by the Operating System and your PLC application.

The dwMemoryLoad tells you in percent the actual RAM program memory load. With a medium PLC runtime application you will see a load of approx. 60-63%. Only applications with multiple driver connections and a lot of logic will require larger amount of memory. The maximum load recommended on a RTU is 90%.

Brodersen A/S, Islevdalvej 187, DK-2610 Roedovre, Denmark, Tel: +45 45 35 26 27, Fax: +45 45 35 26 29, Email: sales@Brodersen.com   48
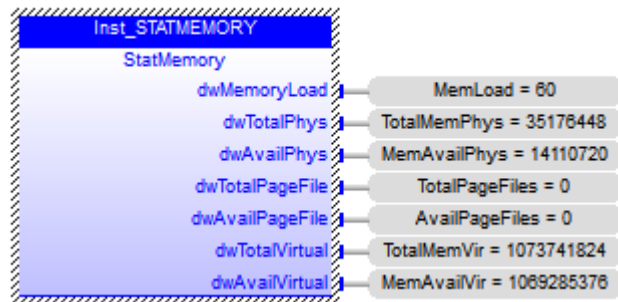
Typical memory status in a running RTU32N

13.2   Reading Text Files from WorkSuite

To provide the functionality to read variables and strings from an external file in the RTU32N, a few simple read file functions are supported. The functions are found in the IniFiles folder of the WorkSuite logic blocks
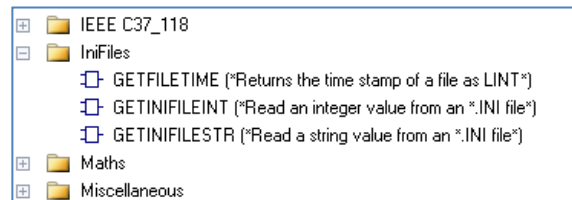
Note: You have to place the text file in the RTU32N file system manually – use FTP to do this. MS Explorer can be used to explorer the RTU32 files – enter in the address field, like: ftp://192.168.0.1/. An empty folder will come up – right click in it and select login using your username and password (same as used for entering the web pages).

The file structure must be the same as other RTU .ini files (such as SNMP.txt and RTU32.ini). The structure is simple and looks like this:

[ABC]
 def=123
 ghi=456

ABC is called the section and def the item. NOTE: all are case sensitive.

The functions available are:

- GetFileTime

  With this function you can check the time of the file so you only read the variables in case of changes.

- GetIniFileInt

  Read an Integer value from a section and item.

GetIniFileStr

Read a String variable from a section and item.

```
GetIniFileStr (*Read a string value from an *.INI file*)
IN
  Section:STRING
  Item:STRING
  Default:STRING
  FileName:STRING
OUT
  Q:STRING
```

The idea with these functions is that you can place your 'dynamic' variables outside the RTU runtime application program. You can then use the same application program for several sites and manage the differences for each site in the file.

You can edit the file from the web editor with your browser or directly send a new file to the RTU32N with FTP.

More help is found in WorkSuite by pressing F1.

14   Distributed Event Binding Protocol / Redundancy Binding Communication

14.1   General

The Distributed Binding protocol is a proprietary protocol for fast time stamped event based data transfer directly from one Brodersen RTU runtime to another. The Binding protocol is communicating directly from one RTU32 to another RTU32 in a LAN/WAN network. It can run regardless of any other application. In a network with several Brodersen RTUs you can use the Global Binding Editor to bind any global declared variables together. Binding can be used between any type of RTU32 Series product and even with STRATON PC runtimes (soft PLCs).

Redundancy Binding is used in a redundancy network connection using 2-6 separate network segments. Both the unit publishing variables (producer) and the unit subscribing variables (consumer) has to be equipped with 2-6 Ethernet network interfaces. Each network connection must be setup on the units as separate network segments using different TCP/IP subnet masks. The compiler also takes care of handling the sorting out of timing issues of any changes in values. This means that you on the subscriber side always will get the correct value. The first received event will be reported at the subscriber side and the same event with the same time stamp received on other network interfaces will be ignored.

14.2   Network settings in RTU32 (redundancy binding)

If you want to run redundancy Binding using the RTU32 you must configure all Ethernet interfaces. No additional settings are necessary as the networks is seen as direct accessible parallel networks. Routing to the networks is done as default. If you ping an address on any of the one of selected network segments, the ping request will find its way to the correct segment automatically.

14.3   Procedure for setting up binding/redundancy binding in WorkSuite

You need to configure all the RTU32 (and PCs with STRATON runtime) before starting to bind (link) I/Os together. I/Os and variables to bind must be declared as Global variables. Use variable naming that makes sense for you. You could for example call all input variables published for binding *BindInput_DI0* and *BindInput_DI1*, and you could call outputs *BindOutput_DO0.* After defining the variables for use in both comms direction for all relevant projects (RTUs), you must start the Global Binding Editor (find it in WorkSuite under the Tools menu). Note that in WorkSuite you will need all Projects used for binding in the same project list.
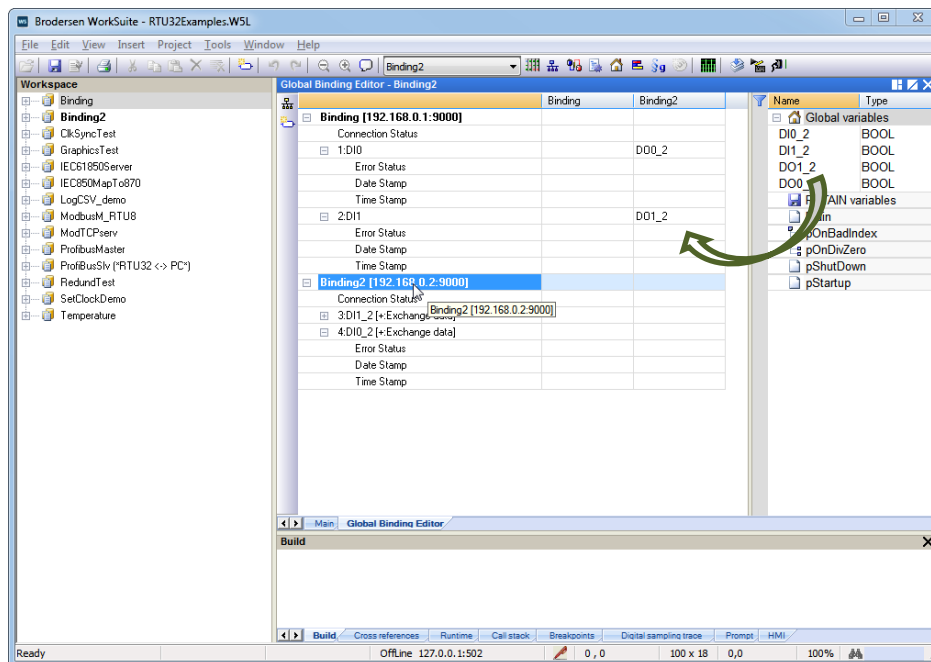
You assign the IP addresses to each project. If using redundancy Binding, you add up to maximum 6 x "IP:TCP port" to the address of the RTU. If two network interfaces are used for binding (dual redundant binding), the two IP addresses are defined with a separator "|" – like 192.168.1.222:9000|192.168.2.21:9000"

As show above, the OUTPUTS from Publishing RTU project are linked to the subscribing RTUs INPUTS.

After configuration you download ALL projects to the relevant RTUs. Use the "Download All Projects…" in the Project menu,

Use our How-to documents for details about setting up a binding application.

15 RTU32N cycle / scan time

16 General

16.1   RTU32N Scan cycle mechanisms

All tasks run independent of each other, parallel tasks or multithreaded.

The two I/O cycles are running as fast as possible to ensure that the basic RTU32N I/O database at any time reflects the physical I/Os – both internal I/O and I/O on the expansion modules connected. The PLC runtime program cycle time is depending on the size of the application program, and can vary from less than 1 msec to >100 msec.

At runtime application start, all inputs are read from the I/O database and at the end of each cycle the outputs are written to the I/O database.

The runtime application scan time (time between each cycle start) can be adjusted – see section *RTU scan time* section for details.

The link timing between these processes can best be explained in a graphical way:
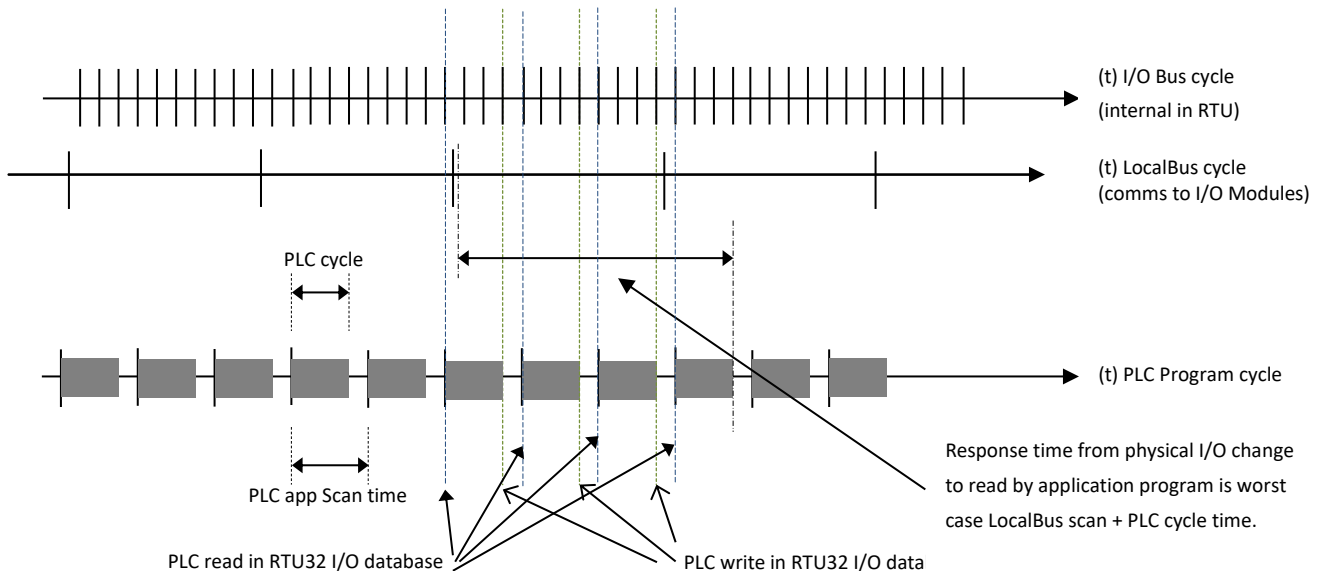
Note: The timing in the figure is shown to explain the link between the cycles. The timing of the cycles in a real application will be according to the actual RTU application program size and number of connected I/O expansion modules.

16.2   RTU scan time

The RTU scan time can be set to 0 (zero) which automatically assigns some time for operating system household activities between each runtime program cycle. In this case the program runs as "fast" as possible. The automatic assignment of household execution time ensures all necessary operating system processes can be maintained with a good margin.
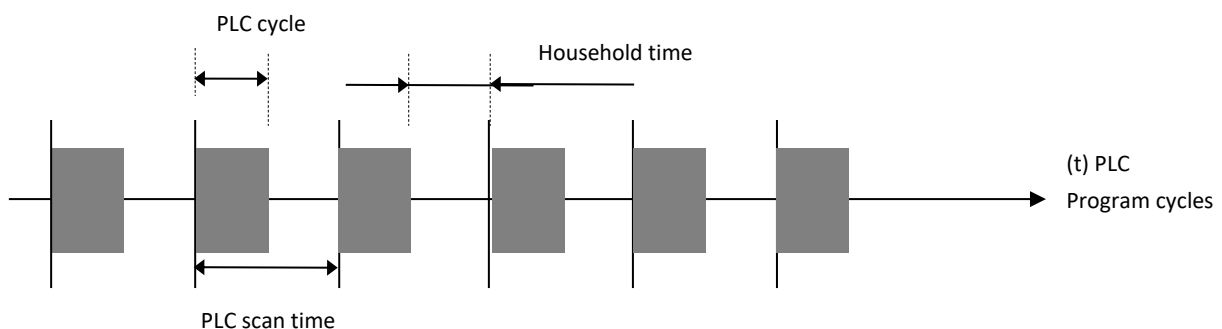
If you set the PLC scan time to a fixed value, you can in some cases obtain a better general scan time for your application program. But it is IMPORTANT to realise that you are then responsible to leave time for household operations.

IMPORTANT NOTICE: If you leave too short time for household operations, the firmware will automatically ensure that your PLC program does NOT compromise necessary household time. It also means that you will get a slower scan cycle time than adjusted. If your RTU/PLC application includes many TCTP/IP based connections over LAN connections, it is recommended to fix the scan time to a value leaving more time to the OS household than automatically assigned.

16.3   LB2 and Internal I/O Bus scan time (LB2 I/O Modules)

The LocalBus 2 driver is for the LB2 I/O Expansion.

LocalBus 2 and the RTU32N internal I/O scan time on the RTU32N is in micro seconds. Therefore no delay has to be added from I/O to CPU.

16.4   LocalBus scan time (for UCL I/O modules)

The LocalBus driver is used for internal I/O boards and all connected I/O Expansion modules.

The scan time of the local bus to Expansion I/O is running at a fixed initial scan time and is also dependant on the number of I/O Expansion modules connected. Data communication is a simple and reliable shift register bus.

The procedure for the communication is that the I/O modules are initialised at boot up. After that the internal I/O database in configured and the I/Os are scanned. Cyclically the I/O configuration is checked for errors.

LocalBus scan time on the RTU32N is from 80-100msec depending on I/O configuration.

17   RTU32 Utilities and System Event Log

17.1   General

Brodersen provide some utility programs for the RTU32 Series on the RTU32 CD included in each package. The tools include:

- Software Bulk Update Tool
- RTU32 Search Tool
- Event Viewer for RTU32 maintenance and configuration and Event Viewer for RTU32 system log monitoring
- Remote Display Tool

17.2   RTU32 Software Bulk Update Tool

See the Bulk Update Tool User Guide on CD for details.

17.3   RTU32 Search Tool
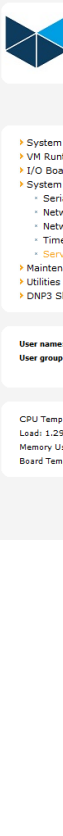
The RTU Search Tool is used to find RTUs on an Ethernet network. It will search using the network card/interface you define under the LAN Interface. Be careful that you select the right one before you start the search procedure.

The Local Subnet will automatically detect the network details and if you place your cursor on the "i" you will get network details.

When you start the search process, the network will be scanned for Brodersen RTUs. Only RTUs with their RTU PLC runtime running will answer. If you select an RTU on the list of RTUs found on the network and right-click, you will be able to open either the Webpage or the Event Viewer (status Log info).

17.4   System Log/Event Viewer

The system log provides you with information about started processes in the RTU32N, errors and status information's. You can also view debug or information text from your WorkSuite application program which will be printed in the system log. COM port communications can be logged if you add the right parameters to the COM port setup – see COM port settings for details.

The System Log can be viewed by using the Brodersen Event Viewer utility or a standard Telnet window. The Event Viewer is a simple Telnet utility. You can enable/disable Eventlog server and other services. The System Log is available on port 911 and look like this in the Event Viewer:

The RTU32 allows max 20 continuous connections and the log FIFO include max 100 messages.

17.5   Network Packet Capture

You can enable logging of the RTU's network traffic with this feature.

You can specify the connection and port to be captured and after pressing start RTU will fill e rotary queue with the size you have specified. After pressing stop, you can download the file with the latest RTU network's traffic. And open the file for analysing in e.g. "wireshark"

18 RTU32N Technical Description

18.1 General

The RTU32N series is based on 32-bit Imx6 architecture and provides the RTU/PLC with power and leading edge functionality. RTU32N provides open and adjustable platform with both the power and functionality required to control advanced industrial applications. An industrial power supply is integrated to support industrial supply levels. The basic RTU32N includes a range of communication and other interfaces. The RTU32N provides interfaces like LocalBus and LB2 for expansion I/O and integrated I/O.

The RTU32N software is stored on its internal Flash or micro SD card. During start-up, the operating system and applications are moved to RAM where it is executed. Retained variables are stored in FE RAM a Non volatile RAM (NV RAM)

Integrated I/O and LocalBus for external I/O connectivity are controlled in an implemented I/O database. This enables the PLC runtime application program to be executed in the RTU32N.

Using the Ethernet network for primary communication provides all the advantages of existing TCP/IP networking communication facilities. Fast, reliable and secure communication is the main features and all networking components (software, routers, switches, etc.) are available. In addition, serial ports for interfacing to application specific protocols (e.g. Modbus, Fieldbus, utility and traffic proprietary protocols etc.) are available.

Detailed technical information about the RTU32N hardware is provided in the RTU32N Data Sheet and in the RTU32N Wiring and Mounting Instruction manual.

18.2 PLC Runtime

The WorkSuite programming tool fully supports EN/IEC61131 and is used for developing PLC programs in the RTU32N. WorkSuite offers complete PLC functionality as found in both normal PLC runtimes and in SoftPLCs, and supports all features needed in today's industrial environment. WorkSuite supports programming languages including Structured Text, Function Block, Ladder, Instruction List and Sequential Function Chart. The VM PLC runtime supports cold restart, hot restart and on-line changes.

The Brodersen WorkSuite is used for configuration protocols, programming and debugging. It supports several tools for multi-program handling and documentation. It is also a powerful tool for complete system design and programming, providing unique functions for event based communication. The Global Binding Editor makes it possible to publish and subscribe variables in a large network with minimum communication load. The events are time stamped and can also be used directly in ZenOn SCADA HMI applications.

Programming, debugging and upload and download of application programs can be done remotely via Ethernet or RS232.

The RTU32N application kernel is used in many products and applications worldwide. More details about WorkSuite including training and help can be gained by contacting your local Brodersen distributor or Brodersen support.

18.3 RTU32 LB2 series for I/O Expansion

See more on http://brodersen.com/products/ all datasheets, userguides, videos etc. are avalible on our webpage.

18.4 Firmware Versions

The Settings overview page in the RTU32 includes the software version for the actual build.

The software in the RTU32N includes 3 main software parts:

- Basic RTU32N Version – like above *1.55.1.0/1.18.10*
- Runtime version

The Runtime project details include the name and build version of the PLC runtime in the RTU32N.

RTU32N Firmware includes the specific RTU32N functions and includes all files in the internal Flash. The RTU32N firmware is assigned an id (e.g. 92312), a date and a version.
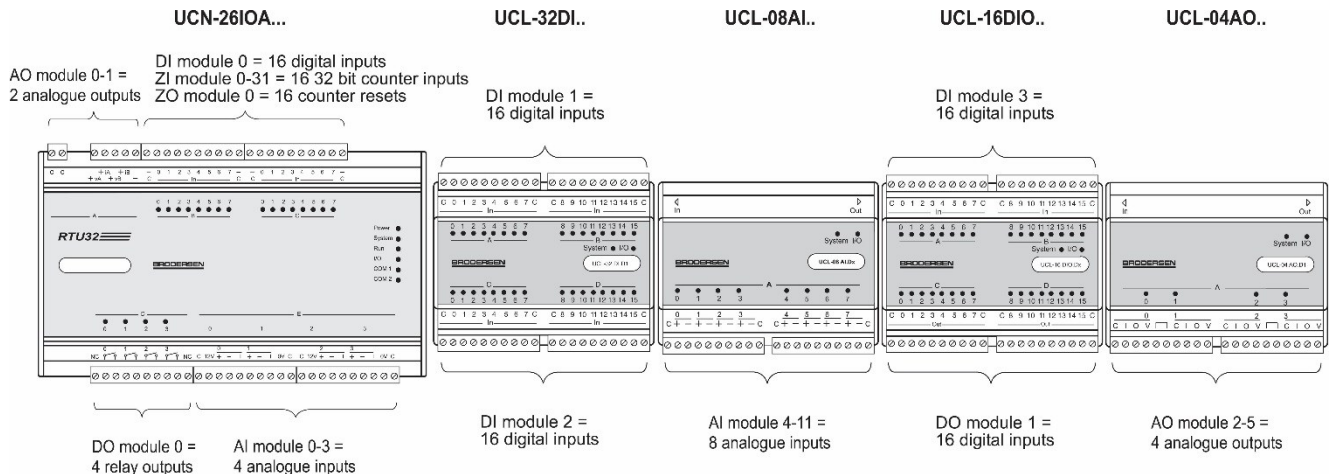
If you contact Brodersen or your local distributor / system integrator for support, it is important that you have the **RTU32N Version Information** ready together with the serial number of your device(s).

20 Appendix 1

RTU32 I/O addressing example 1

The addressing example shows the RTU32 - the same structure is used by ALL RTUs in the Brodersen RTU32 Series.

NOTE: Analogue input module with 16bit resolution and single isolated channels type UCL-08AIC does covers two modules (two words) for each input. First Module (Word) is the 16bit analogue values and the second Module (Word) does report channel status information like overflow, underflow, disabled, error etc.

21 Appendix 2

RTU32 I/O addressing example 2

The addressing example shows the RTU32 - the same structure is used by ALL RTUs in the Brodersen RTU32 Series.
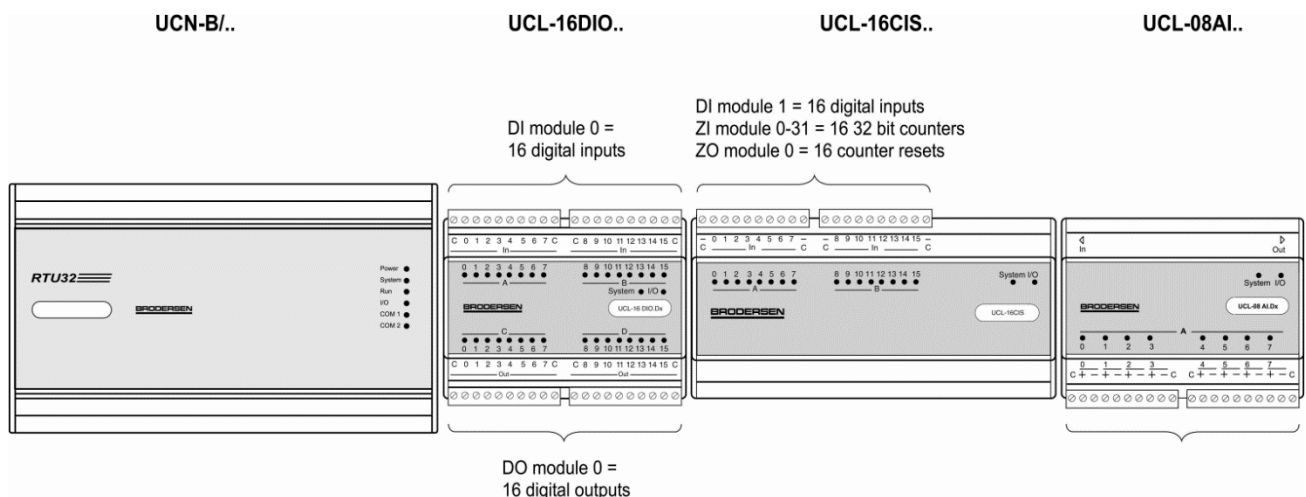
NOTE: Analogue input module with 16bit resolution and single isolated channels type UCL-08AIC does covers two modules (two words) for each input. First Module (Word) is the 16bit analogue values and the second Module (Word) does report channel status information like overflow, underflow, disabled, error etc.

22 Appendix 3

RTU32N complete Reset to factory default.

You can make a complete reset of the RTU32N that will reset it back to factory default settings. Note that during reset process all settings in the RTU32N are lost.

To make a complete reset you will need a small screwdriver, pointed pen or paperclip for activating the reset button.
On the underside of the RTU32N (to the right of the SD card location) you will find a small 3mm hole. Behind this is a small push button that is used in the reset process.

Procedure:

Power down the RTU32N.

Use a small pointed device to push/activate the reset button while you apply power to the RTU32N. Keep the reset button pressed for a minimum of 9 seconds before it is released.

The internal Flash memory of the RTU32N is now cleaned and a new installation of the firmware is installed during boot. The RTU32N will then start-up with factory default settings. LAN IP address = 192.168.0.1 and password erased.

23 Appendix 4

RTU32N firmware update

Go to RTU32 webpage under maintenance. See below picture.

Important that the RTU is not powered off during firmware update. The RTU will reboot after firmware update.