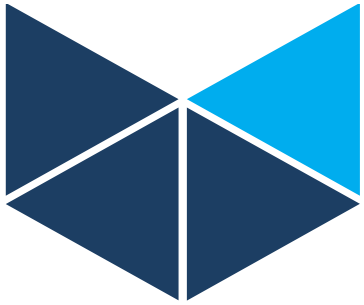
RTU32 IEC60870 Drivers Presentation



Short introduction

To the IEC60870 Protocol

IEC60870 Protocol short introduction

IEC60870-5-101 / 104 introduction

- The standard
- Attestation of Conformity
- 101 / 104 Layer Stack
- Application layer
  - event based communication
  - Buffering of data
  - Communication directions - monitor and control direction
  - ASDU definition and layout
  - ASDU types
  - COT – cause of transmission
  - Quality bits and time stamps
  - Addressing – Link Address, COA, IOA etc

IEC60870 Protocol short introduction

- IEC60870-5-101 and 104 are international standards for power system monitoring
- IEC60870-5-101 is serial based protocol and is based on following standard documents.
  - IEC60870-5-1, IEC60870-5-2, IEC60870-5-3, IEC60870-5-4, IEC60870-5-5 and IEC60870-5-101
- IEC60870-5-104 is TCP/IP (socket) based protocol and is based on the IEC60870-5-104 document
- These two protocols contains more or less the same application layer however the link layer is different.
- (NUC) Norwegian User Conventions provides the overview of the ICE60870-5-101 and 104 protocol in Norway when connecting to Statkraft
- The IEC60870-5-104 protocol in RTU32 is conformed by KEMA in Nederland both Controlled station and Controlling station in Standard and Reversed direction.

IEC60870 KEMA attestation of conformity

KEMA

ATTESTATION OF CONFORMITY

No. 30530046-Consulting 07-0508

issued to:
Brodersen Controls
Industrivej 3
DK-4000 Roskilde
Denmark

for the product:
RTU32
Controlled station implementation
Firmware Version: 1.11
Compilation date: Feb. 1, 2007

With the implemented communication protocol:

IEC 60870-5-104 (IS 2000)

Network Access for IEC 60870-5-101 using standard transport profiles and the Statkraft Norway PROSAM project requirements version 1.3 in Standard and Reversed direction[1]

The product has not been shown to be non-conforming to the specified protocol standard, including the interface requirements.

End-to-End data element tests for the information and control points as described in KEMA/manufacturer Protocol Implementation Conformance Statement (PICS) have been performed on the product's protocol implementation. Functional tests in controlled mode are performed for the following compatible Basic Application Functions:

• Station initialization[1]	• Test mode[1]
• Acquisition of events[1]	• Transmission of integrated totals[1]
• General Interrogation[1]	• Parameter loading[1]
• Command transmission[1]	• Redundancy and PROSAM requirements[1]

The test campaign did not reveal any errors in the product's protocol implementation.

This Attestation is granted on account of tests made at the KEMA Consulting Europe site in Arnhem The Netherlands, and performed with UniECim 60870-5-104 version 1.14.01 (December 2006) running CS104 Test Suite version CS104 V1.24. The results, including remarks and limitations, are laid down in KEMA report no. 30530046-Consulting 07-0508.

The tests have been carried out on one single specimen of the product, submitted by Brodersen Controls. The Attestation does not include an assessment of the manufacturer's production. Conformity of his production with the specimen tested by KEMA is not the responsibility of KEMA.

Arnhem, 9 March 2007

S.J.L.M. Janssen
Managing Director KEMA Consulting

P.H.S. Ermens
Test Consultant

[1] IMPORTANT: remarks apply to the implementation of this function. See the resulting report (chapter 4) for full details

KEMA Nederland B.V.
Utrechtseweg 310, 6812 AR Arnhem; P.O. Box 9035, 6800 ET Arnhem, The Netherlands.
Telephone + 31 26 356 91 11, Telefax + 31 26 443 38 43

KEMA

ATTESTATION OF CONFORMITY

No. 30530046-Consulting 07-1287

issued to:
Brodersen Controls
Industrivej 3
DK-4000 Roskilde
Denmark

for the product:
RTU32
Controlling station implementation
Firmware Version: 1.21
Compilation date: June 7 2007

With the implemented communication protocol:

IEC 60870-5-104 (IS 2000)

Network Access for IEC 60870-5-101 using standard transport profiles and the Statkraft Norway PROSAM project requirements version 1.3 in Standard and Reversed direction[1]

The product has not been shown to be non-conforming to the specified protocol standard, including the interface requirements.

End-to-End data element tests for the information and control points as described in KEMA/manufacturer Protocol Implementation Conformance Statement (PICS) have been performed on the product's protocol implementation. Functional tests in controlled mode are performed for the following compatible Basic Application Functions:

• *Station initialization*[1]	• *Test mode*[1]
• *Acquisition of events*[1]	• *Transmission of integrated totals*[1]
• *General Interrogation*[1]	• *Parameter loading*[1]
• *Command transmission*[1]	• *Redundancy and PROSAM requirements*[1]

The test campaign did not reveal any errors in the product's protocol implementation.

This Attestation is granted on account of tests made at the KEMA site in Arnhem, the Netherlands, and performed with UniECim 60870-5-104 version 1.15.01 (March 2007) running CS104 Test Suite version CS104 V1.25. The results, including remarks and limitations, are laid down in our report no. 30530046-Consulting 07-1286.

The tests have been carried out on one single specimen of the product, submitted by Brodersen Controls. The Attestation does not include an assessment of the manufacturer's production. Conformity of his production with the specimen tested by KEMA is not the responsibility of KEMA.

Arnhem, 15 June 2007

S.J.L.M. Janssen
Managing Director KEMA Consulting

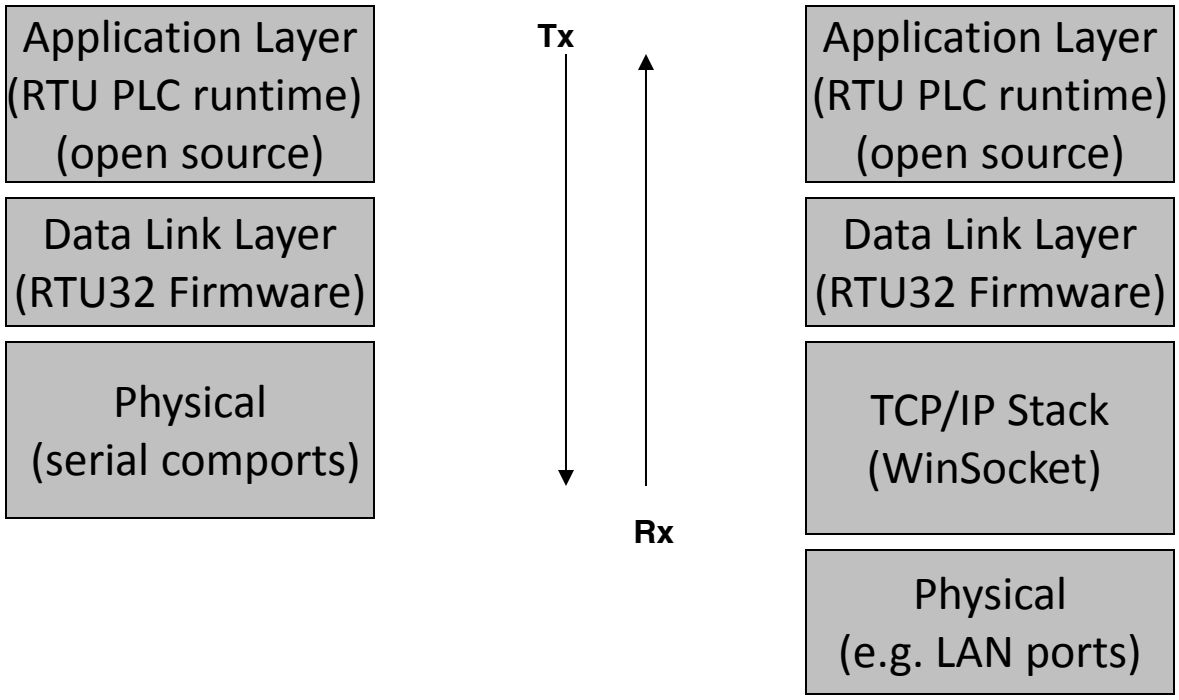
P.H.S. Ermens
Test Consultant

[1] IMPORTANT: remarks apply to the implementation of this function. See the resulting report (chapter 4) for full details.

KEMA Nederland B.V.
Utrechtseweg 310, 6812 AR Arnhem; P.O. Box 9035, 6800 ET Arnhem, The Netherlands.
Telephone + 31 26 356 9111, Telefax + 31 26 443 38 43

IEC60870 101 /104 Protocol Layer Stack

101 protocol layers

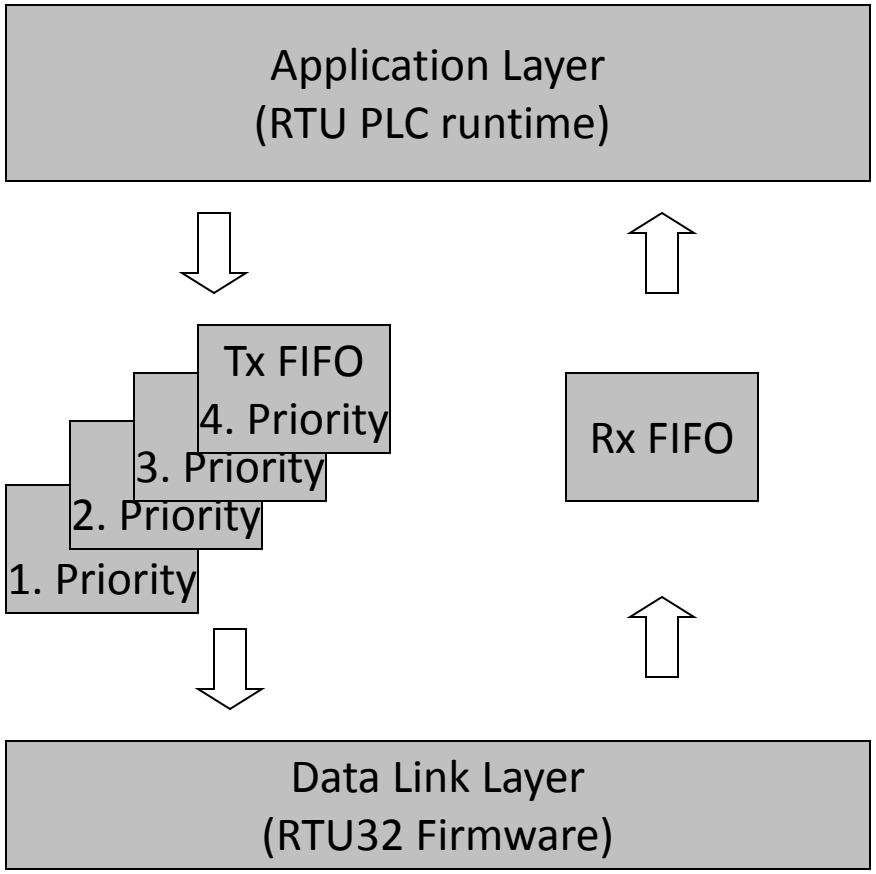


104 protocol layers

RTU32 has options to log either the link transfer or the application transfer into the System Log for debugging purposes

IEC60870 101 /104 Protocol Communication Buffers

Event based communication with communication FIFOs

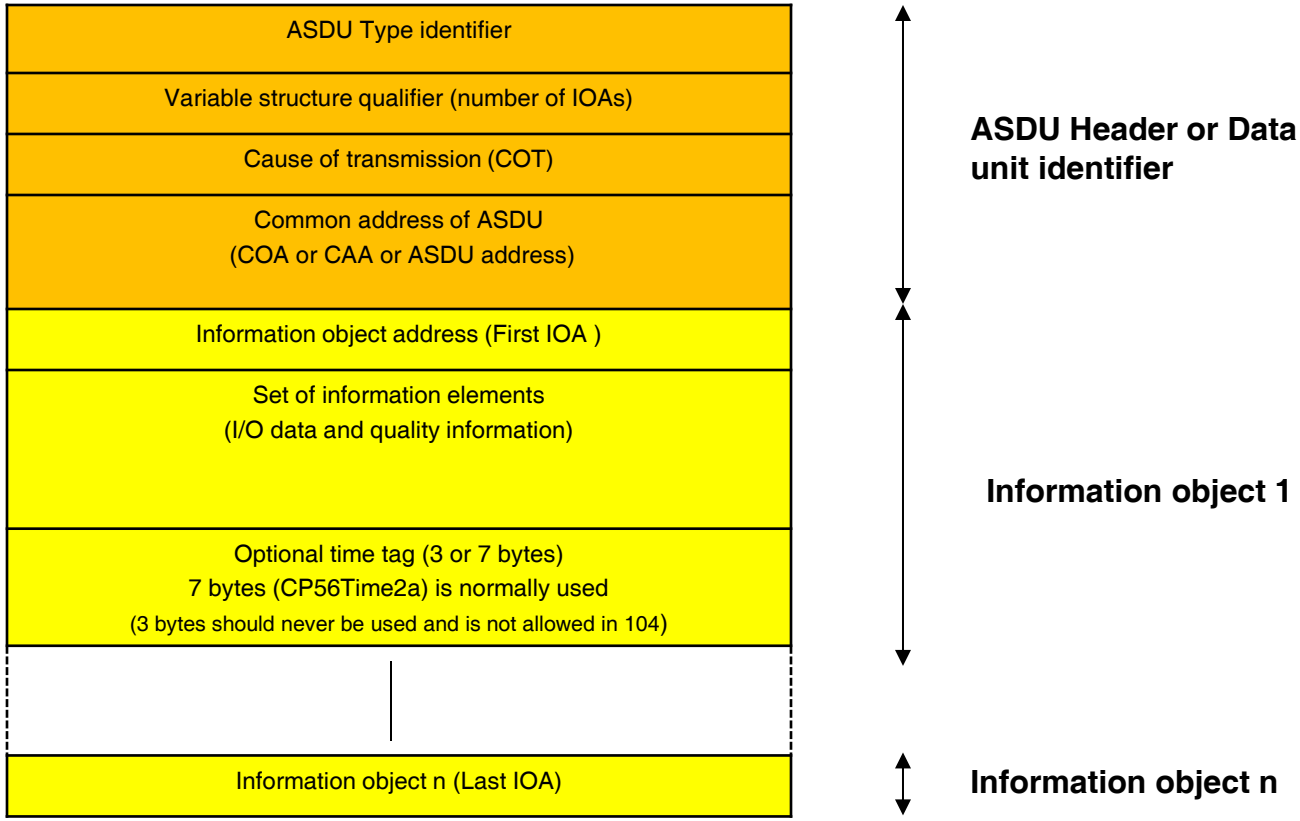


IEC60870 101 /104 Definitions

- Controlled Station
  - A station that is monitored or commanded by a master station (RTU)
  - Outstation, Remote Station, RTU, 101-Slave, or 104-Server
- Controlling Station
  - A station where a control of outstations is performed (SCADA)
  - Typically a PC with SCADA system, can also be a RTU32.
- Monitor Direction
  - A direction of transmission from controlled station (RTU) to the controlling station (PC)
- Control Direction
  - A direction of transmission from the controlling station, typical a SCADA system to controlled station typical an RTU
- Reversed Direction
  - A direction when monitored station is sending commands and controlling station is sending data in monitor direction.

IEC60870 101 /104 Applications Messages (ASDUs)

ASDU telegram layout



ASDU types, Process information in monitor direction

TYPE IDENTIFICATION := UI8[1..8]<0..44>

•	<0>	:=	not defined	
•	<1>	:=	single-point information	M_SP_NA_1
•	<3>	:=	double-point information	M_DP_NA_1
•	<5>	:=	step position information	M_ST_NA_1
•	<7>	:=	bitstring of 32 bits	M_BO_NA_1
•	<9>	:=	measured value, normalized value	M_ME_NA_1
•	<11>	:=	measured value, scaled value	M_ME_NB_1
•	<13>	:=	measured value, short floating point number	M_ME_NC_1
•	<15>	:=	integrated totals	M_IT_NA_1
•	<20>	:=	packed single-point information with status change detection	M_PS_NA_1
•	<21>	:=	measured value, normalized value without quality descriptor	M_ME_ND_1
•	<22..29>	:=	reserved for further compatible definitions	
•	<30>	:=	single-point information with time tag CP56Time2a	M_SP_TB_1
•	<31>	:=	double-point information with time tag CP56Time2a	M_DP_TB_1
•	<32>	:=	step position information with time tag CP56Time2a	M_ST_TB_1
•	<33>	:=	bitstring of 32 bit with time tag CP56Time2a	M_BO_TB_1
•	<34>	:=	measured value, normalized value with time tag CP56Time2a	M_ME_TD_1
•	<35>	:=	measured value, scaled value with time tag CP56Time2a	M_ME_TE_1
•	<36>	:=	measured value, short floating point number with time tag CP56Time2a	M_ME_TF_1
•	<37>	:=	integrated totals with time tag CP56Time2a	M_IT_TB_1
•	<38>	:=	event of protection equipment with time tag CP56Time2a	M_EP_TD_1
•	<39>	:=	packed start events of protection equipment with time tag CP56Time2a	M_EP_TE_1
•	<40>	:=	packed output circuit information of protection equipment with time tag CP56Time2a	M_EP_TF_1
•	<41..44>	:=	reserved for further compatible definitions	

ASDU types, Process information in control direction

TYPE IDENTIFICATION := UI8[1..8]<45..69>

• **CON** <45>	:= single command	C_SC_NA_1
• **CON** <46>	:= double command	C_DC_NA_1
• **CON** <47>	:= regulating step command	C_RC_NA_1
• **CON** <48>	:= set point command, normalized value	C_SE_NA_1
• **CON** <49>	:= set point command, scaled value	C_SE_NB_1
• **CON** <50>	:= set point command, short floating point number	C_SE_NC_1
• **CON** <51>	:= bitstring of 32 bits	C_BO_NA_1
• <52..57>	:= reserved for further compatible definitions	

ASDUs for process information in control direction with time tag:

• **CON** <58>	:= single command with time tag CP56Time2a	C_SC_TA_1
• **CON** <59>	:= double command with time tag CP56Time2a	C_DC_TA_1
• **CON** <60>	:= regulating step command with time tag CP56Time2a	C_RC_TA_1
• **CON** <61>	:= set point command, normalized value with time tag CP56Time2a	C_SE_TA_1
• **CON** <62>	:= set point command, scaled value with time tag CP56Time2a	C_SE_TB_1
• **CON** <63>	:= set point command, short floating-point number with time tag CP56Time2a	C_SE_TC_1
• **CON** <64>	:= bitstring of 32 bits with time tag CP56Time2a	C_BO_TA_1
• <65..69>	:= reserved for further compatible definitions	

Process information in control direction may be sent with or without a time tag but must not be mixed when sending to a given station.

NOTE ASDUs marked "**CON**" in the control direction are confirmed application services and may be mirrored in the monitor direction with different causes of transmission. These mirrored ASDUs are used for positive/negative acknowledgements (verifications).

ASDU types, System information and parameters

System information in monitor direction

- TYPE IDENTIFICATION := UI8[1..8]<70..99>
- <70> := end of initialization — M_EI_NA_1
- <71..99> := reserved for further compatible definitions

System information in control direction

- TYPE IDENTIFICATION := UI8[1..8]<100..109>
- **CON** <100> := interrogation command — C_IC_NA_1
- **CON** <101> := counter interrogation command — C_CI_NA_1
- **CON** <103> := clock synchronization command (optional, see 7.6) — C_CS_NA_1
- **CON** <105> := reset process command — C_RP_NA_1
- **CON** <107> := test command with time tag CP56Time2a — C_TS_TA_1
- <108..109> := reserved for further compatible definitions

Parameter in control direction

- TYPE IDENTIFICATION := UI8[1..8]<110..119>
- **CON** <110> := parameter of measured value, normalized value — P_ME_NA_1
- **CON** <111> := parameter of measured value, scaled value — P_ME_NB_1
- **CON** <112> := parameter of measured value, short floating-point number — P_ME_NC_1
- **CON** <113> := parameter activation — P_AC_NA_1
- <114..119> := reserved for further compatible definitions

ASDU Cause of Transmission (COT)

Cause := UI6[1..6]<0..63>

• <0> :=	not used	
• <1> :=	periodic, cyclic	per/cyc
• <2> :=	background scan*	back
• <3> :=	spontaneous	spont
• <4> :=	initialized	init
• <6> :=	activation	act
• <7> :=	activation confirmation	actcon
• <8> :=	deactivation	deact
• <9> :=	deactivation confirmation	deactcon
• <10> :=	activation termination	actterm
• <20> :=	interrogated by station interrogation	inrogen
• <44> :=	unknown type identification	
• <45> :=	unknown cause of transmission	
• <46> :=	unknown common address of ASDU	
• <47> :=	unknown information object address	

IEC60870 101 /104 Quality bits and Time stamps

Quality bits

- Many information elements are sent with an quality descriptor (one byte)
  - BL not blocked or blocked
  - SB not substituted or substituted
  - NT topical or not topical
  - IV valid or invalid
- These bits are available as WorkSuite variables. Change of quality status will cause a spontaneous transmission.

Time stamps

- The protocol specifies to types of time stamp 3 bytes or 7 bytes
- 7 bytes contains full timestamp and should normally be specified
- Data sent as response to general interrogation does not contain timestamp
- Data sent spontaneously should contain timestamp
- Time can either be taken from the RTU32 clock or be specified by the application which can be relevant if the RTU32 is a gateway.

IEC60870 101 /104 Addressing: Link, COA and IOA

There are 3 addressing levels defined in the IEC60870 protocol:

- Link Address
  - This address is used by the link layer and is not known by the application layer.
  - On 101 the slave or node address (one or two bytes) is the link address
  - On 104 the IP address is the link address of the 104 server (slave)
- COA, Common Address of ASDU also called CAA or ASDU address
  - COA address is used by the application layer and is not known by the link layer.
  - COA is typically the application address of the slave.
  - Slave can also be sectionised into different sections each having their own COA address
  - COA is always 2 bytes in the 104 protocol.
  - The length of COA can be selected either as 1 or 2 bytes in the 101 protocol. The length is fixed per system.
- IOA, Information Object Address,
  - IOA is the address of single information object.
  - A information object is either a physical I/O point or a computed value from PLC runtime
  - IOA is always 3 bytes in the 104 protocol
  - The length of IOA can be selected either as 1, 2 or 3 bytes in the 101 protocol. The length is fixed per system.

IEC60870 101 /104 Addressing: Link, COA and IOA

104 Client
Controlling Station

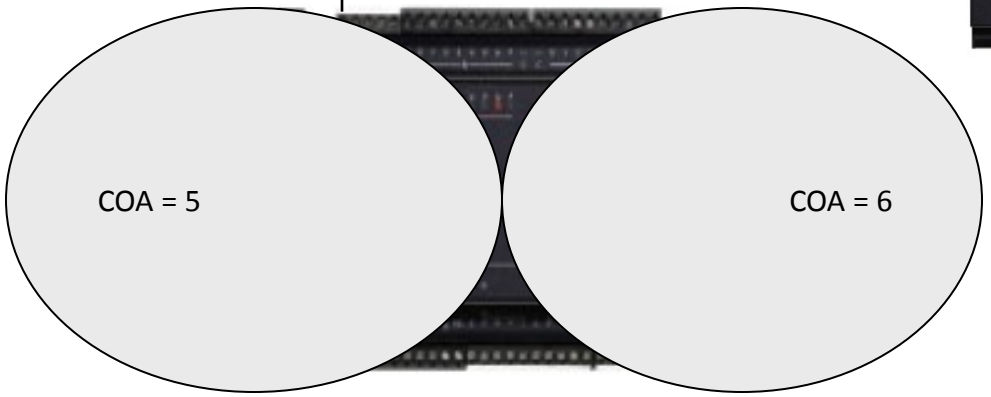
LAN

104 Servers
Controlled Stations

Link Address : eg 192.168.1.2
COA=4

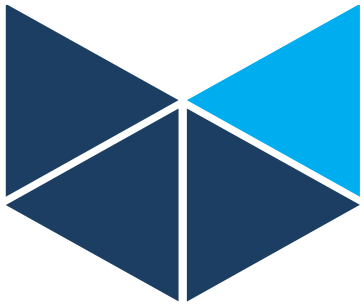
Link Address : eg 192.168.1.3
COA = 7

Link Address : eg 192.168.1.4
COA = 8



Sectionised in 3 different COA sections

IOA=1 **IOA=2** **IOA=3** **IOA=4** **IOA=n**



RTU32 IEC60870 Configurator

IEC60870 Configuration basics

General

- All configuration data is entered in Excel sheets by the user.
- XML database contains the whole IEC60870 definitions as small program templates.
- A Code Generator takes the information entered in an Excel sheet and generates the IEC60870 WorkSuite application program code using the XML database.

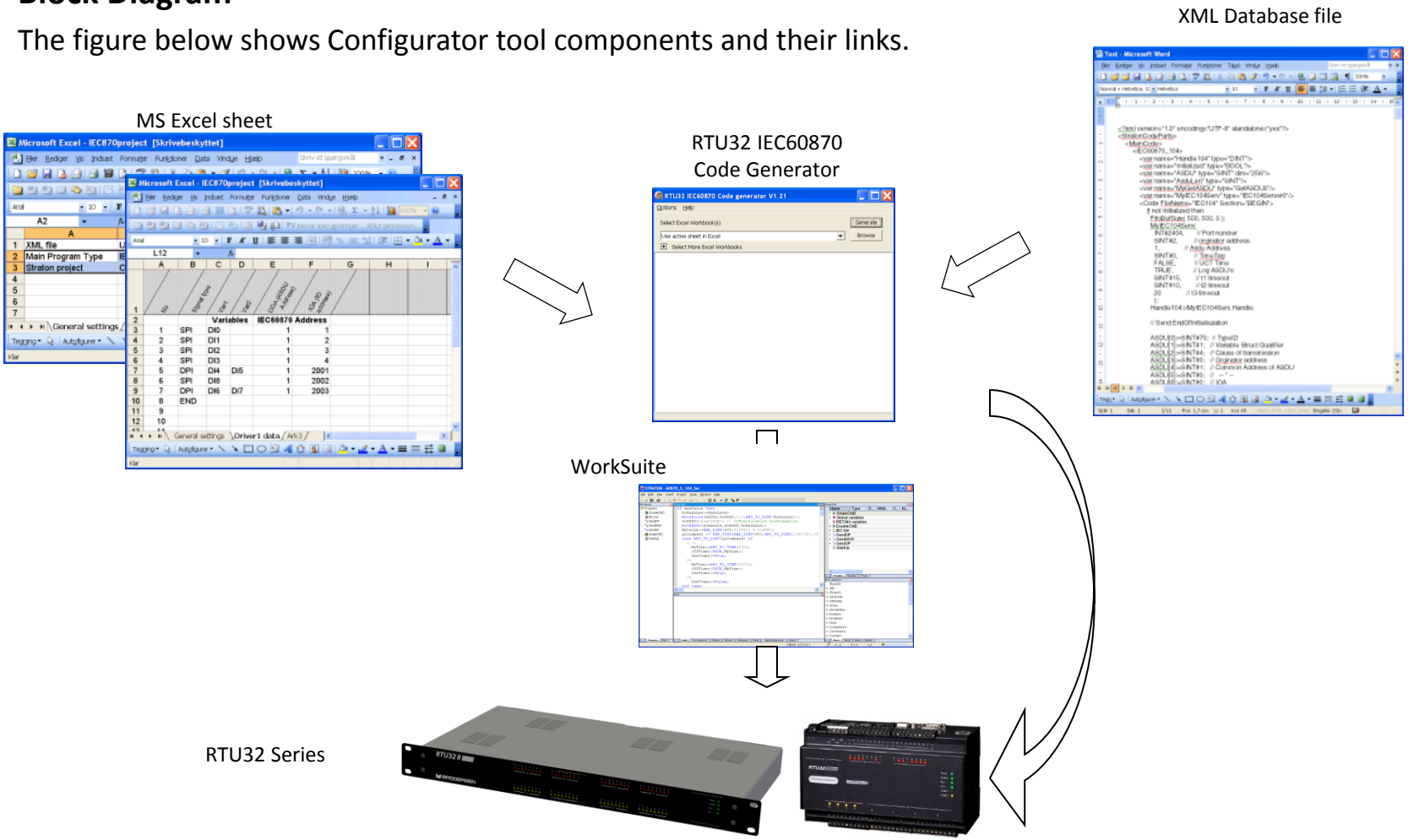
User Interface

- The Code generator user interface is designed to be simple.
- A normal user should only use the Generate bottom to activate the code generation.
- The code generator provides 3 options:
  - Opt1 Generate new or update an existing WorkSuite IEC60870 application
  - Op2 Same as Opt1 and also download and start the application in RTU32
  - Opt3 Same as Opt2 and also open the RTU project in the WorkSuite for editing and debugging.

IEC60870 Configurator for RTU32

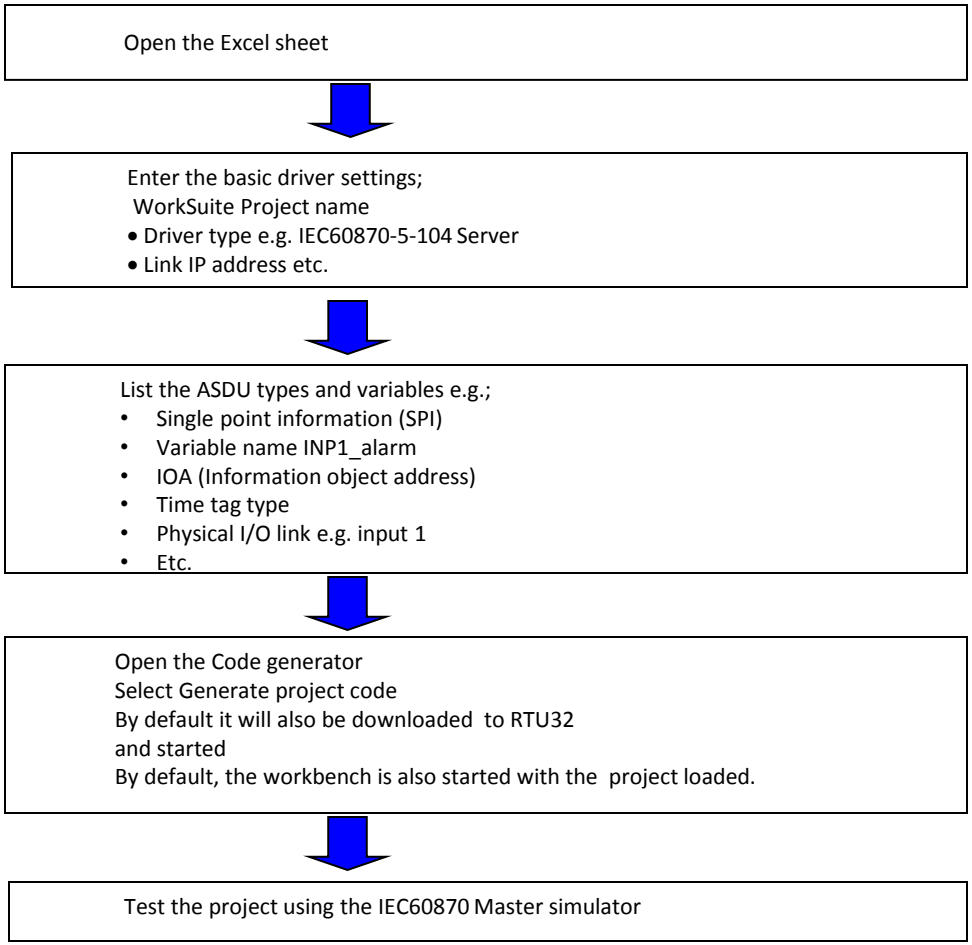
Block Diagram

The figure below shows Configurator tool components and their links.



Work Flow

When developing an IEC60870 Driver in the RTU32, the work flow would be as shown below;



IEC60870 Configurator for RTU32

Excel Workbook contains 7 sheets

- **Main** sheet
  - XML file
  - WorkSuite Project name
  - IP address of the RTU32 (used for download)
  - Main Program Type (driver type)
  - Name of the main program.
- **Variables** sheet
  - Configures all signals that are sent in monitor and control direction.
  - ASDU types
  - Addresses COA, IOA, Variables names for WorkSuite and optionally hardware addresses of the I/Os
- **101 Slave** driver sheet
  - Configures IEC60870-5-101 Slave – including dial connections.
  - Configures all necessary parameters for the 101 Slave (controlled station) driver
- **101 Master** driver sheet
  - Configures IEC60870-5-101 Master – single connection only.
  - Configures all necessary parameters for the 101 Master (controlling station) driver
- **104 Server** driver sheet
  - IEC60870-5-104 Server – communication in normal and reverse direction.
  - Configures all necessary parameters for the 104 Server (controlled station) driver
- **104 Client** driver sheet
  - IEC60870-5-104 Client – communication in normal and reverse direction.
  - Configures all necessary parameters for the 104 Client (controlling station) driver
- **103 Master** driver sheet
  - IEC60870-5-103 Master – communication in normal direction.
  - Configures all necessary parameters for the 103 Master (controlling station) driver

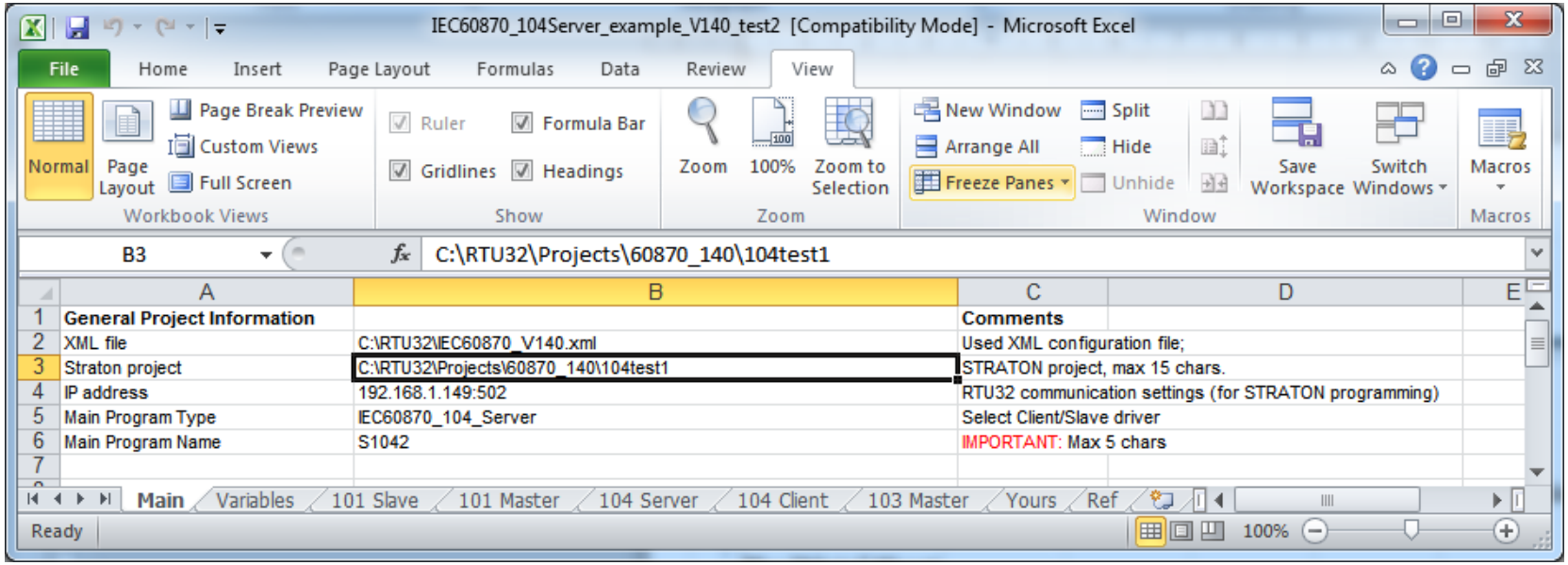
MS Excel Workbook – the User Interface

Work Sheet Structure

The Excel Workbook includes 6 sheets; Main, variables and specific driver data configurations

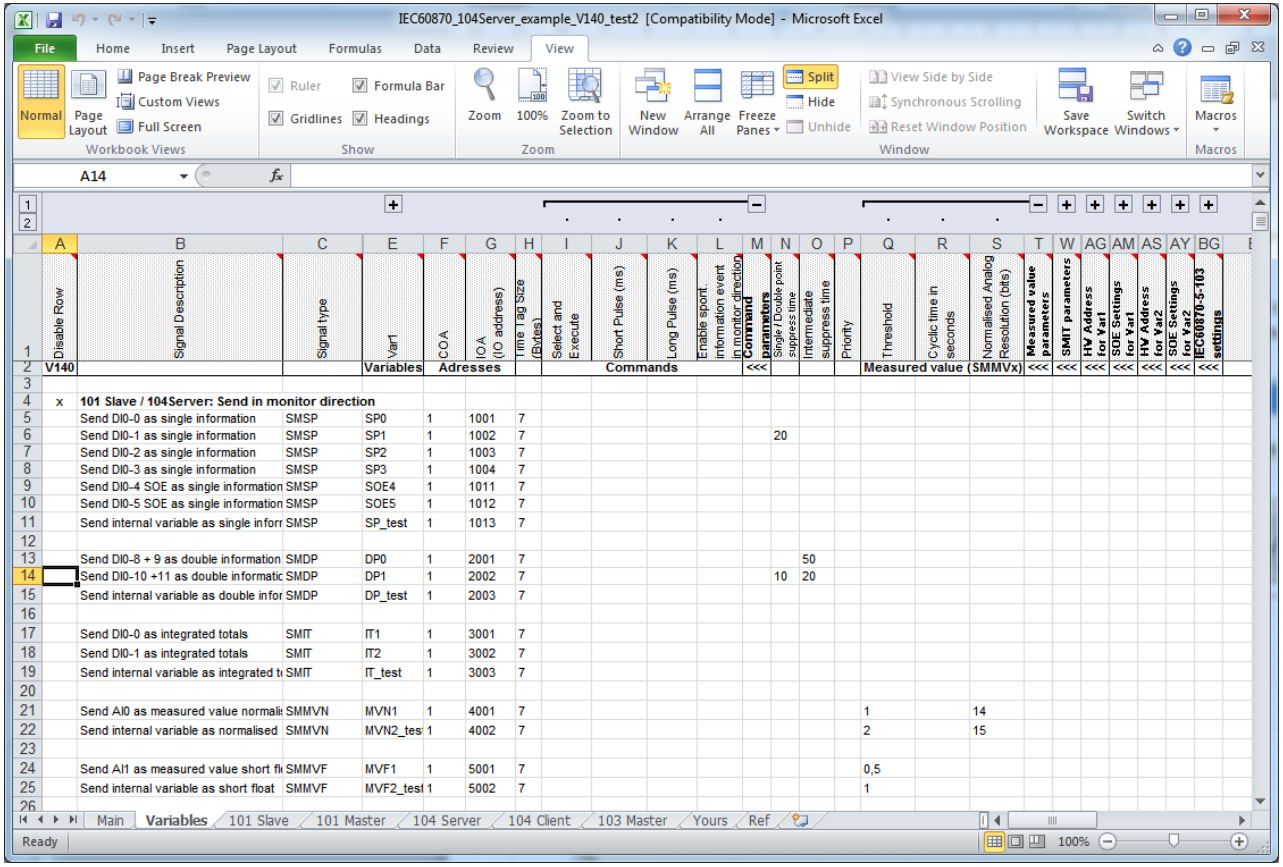
The "Main" Sheet

In the Main sheet all basic driver setting are entered – see figure below:



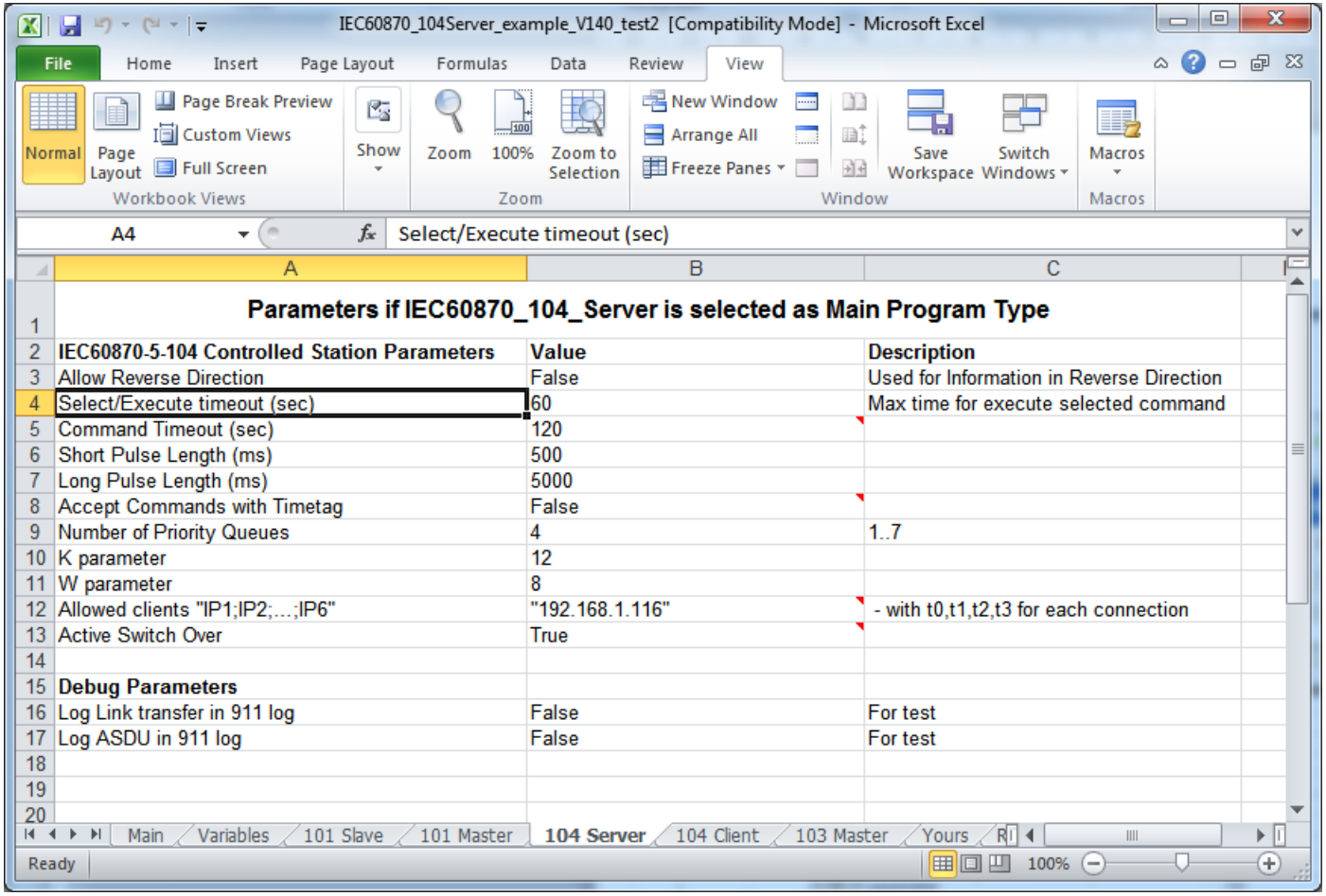
The Variables sheet

The Variables sheets after the Main defines all the signals (I/Os) that should be sent and received. Each row contains one signal. The collection of all the variables on this sheet is called a Process Image or a Redundancy Group (104 protocol)



Work Sheet example– 104 Slave

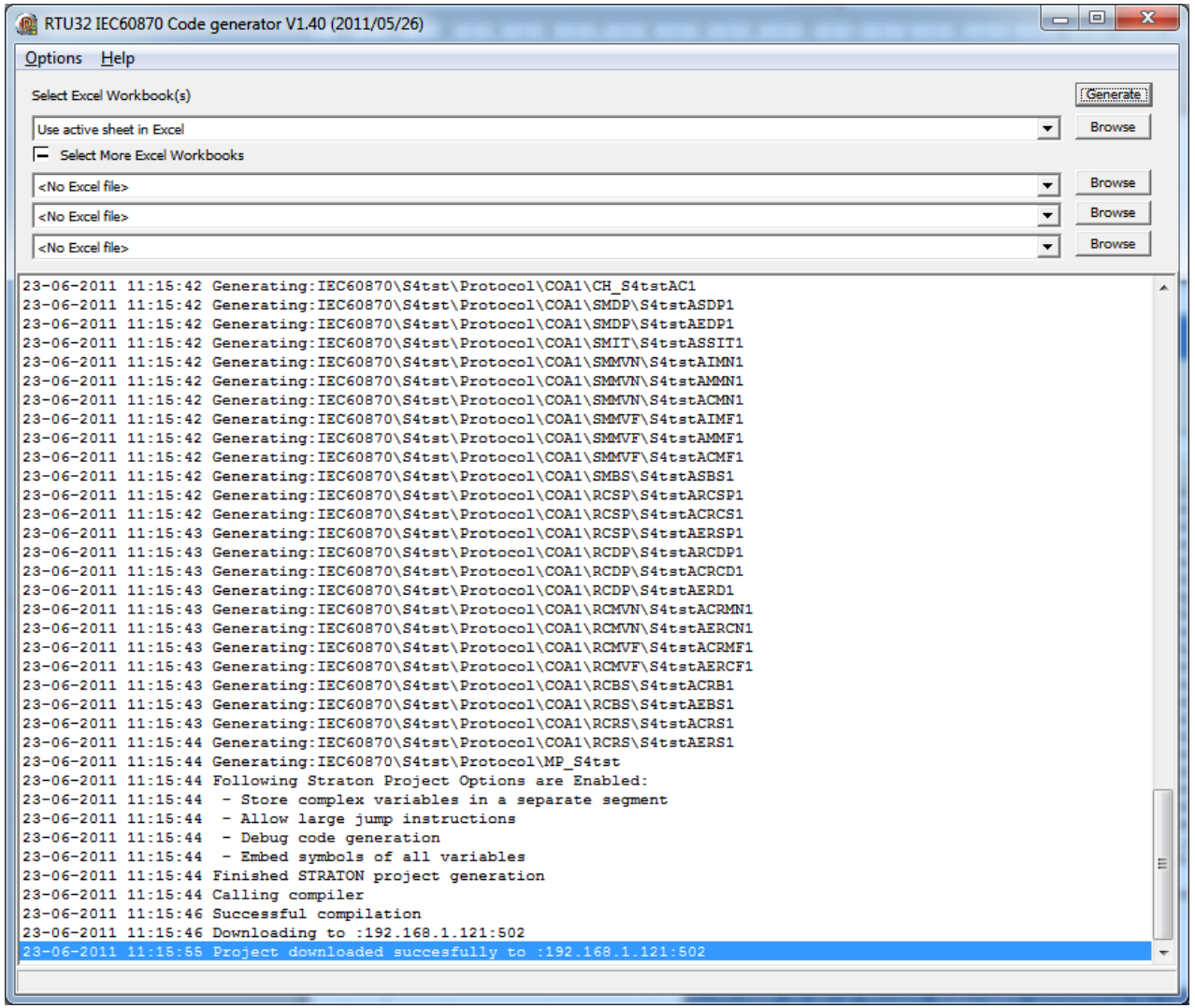
The work sheets after the Variables define the specific driver data settings:



	A	B	C
1	**Parameters if IEC60870_104_Server is selected as Main Program Type**		
2	**IEC60870-5-104 Controlled Station Parameters**	**Value**	**Description**
3	Allow Reverse Direction	False	Used for Information in Reverse Direction
4	Select/Execute timeout (sec)	60	Max time for execute selected command
5	Command Timeout (sec)	120	
6	Short Pulse Length (ms)	500	
7	Long Pulse Length (ms)	5000	
8	Accept Commands with Timetag	False	
9	Number of Priority Queues	4	1..7
10	K parameter	12	
11	W parameter	8	
12	Allowed clients "IP1;IP2;...;IP6"	"192.168.1.116"	- with t0,t1,t2,t3 for each connection
13	Active Switch Over	True	
14			
15	**Debug Parameters**		
16	Log Link transfer in 911 log	False	For test
17	Log ASDU in 911 log	False	For test
18			
19			
20			

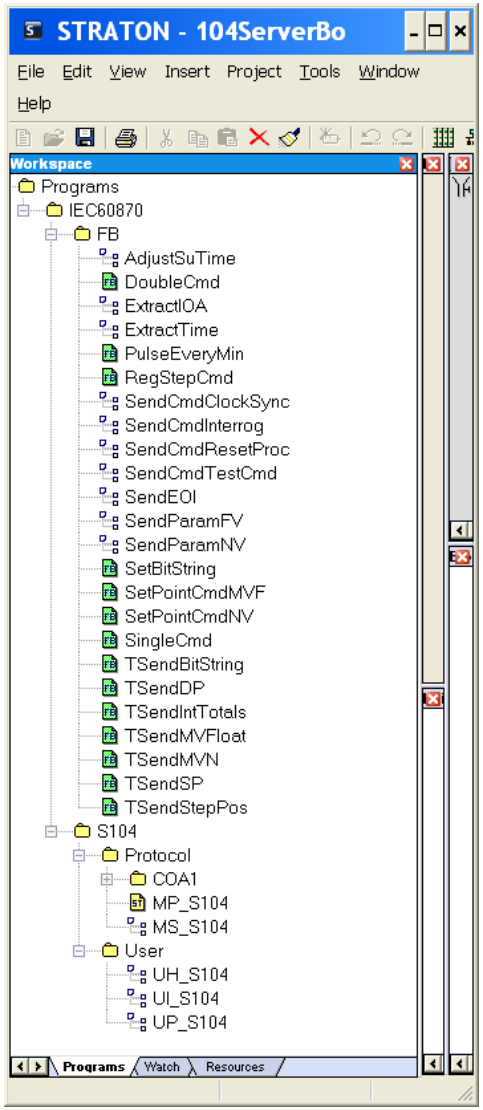
IEC60870 Configurator for RTU32

Code generator



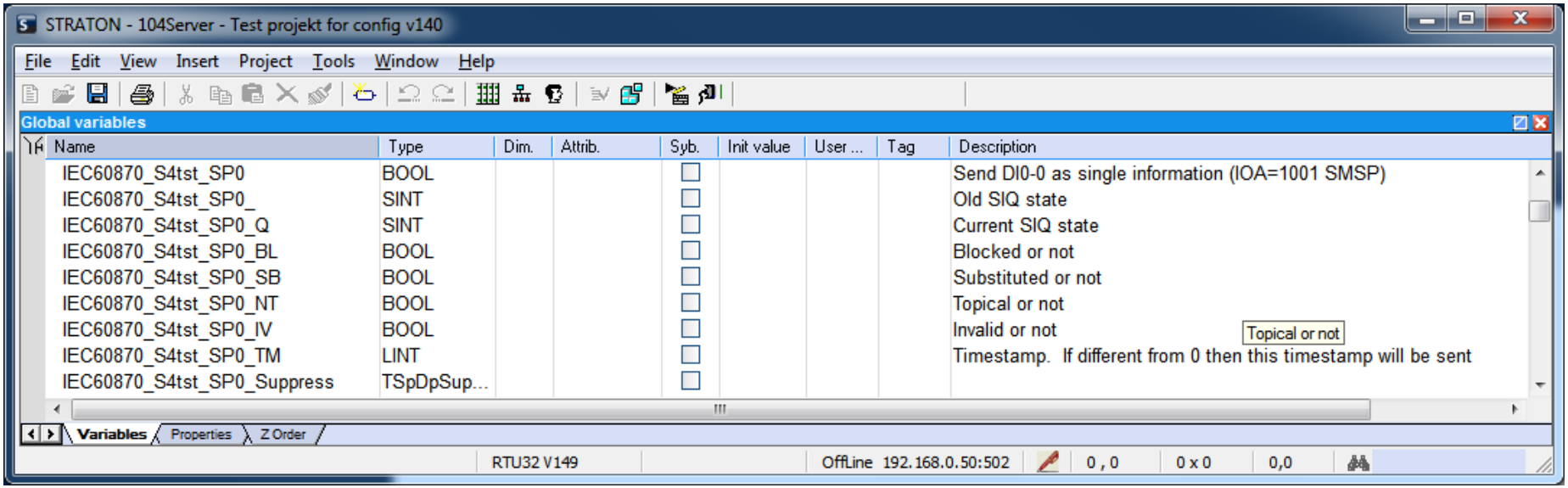
Code in WorkSuite

- The code generator creates a program structure using the driver function blocks and Structured Text code programs for the handling of the protocol.
- The folder \IEC60870\FB contains the Function Blocks that are common to one or more protocol drivers.
- The folder \IEC60870\S104\Protocol contains the protocol for the “Main Program” called S104.
- The folder \IEC60870\S104\User is use to make your own additions to the protocol:
  - The file UI_xxxx is called once when the driver is initializing. Here you can enter your own initialization code.
  - The file UP_xxxx is called in the main loop of the driver. Here you can make your own extensions to the protocol.
  - The file UH_xxxx is called every time the driver receives a application message. Here you can overwrite the default driver reaction.
  - This folder and these files are never overwritten
  - An alternative to editing these files and making your additions more permanent and available for other projects is to make your changes inside the XML file.



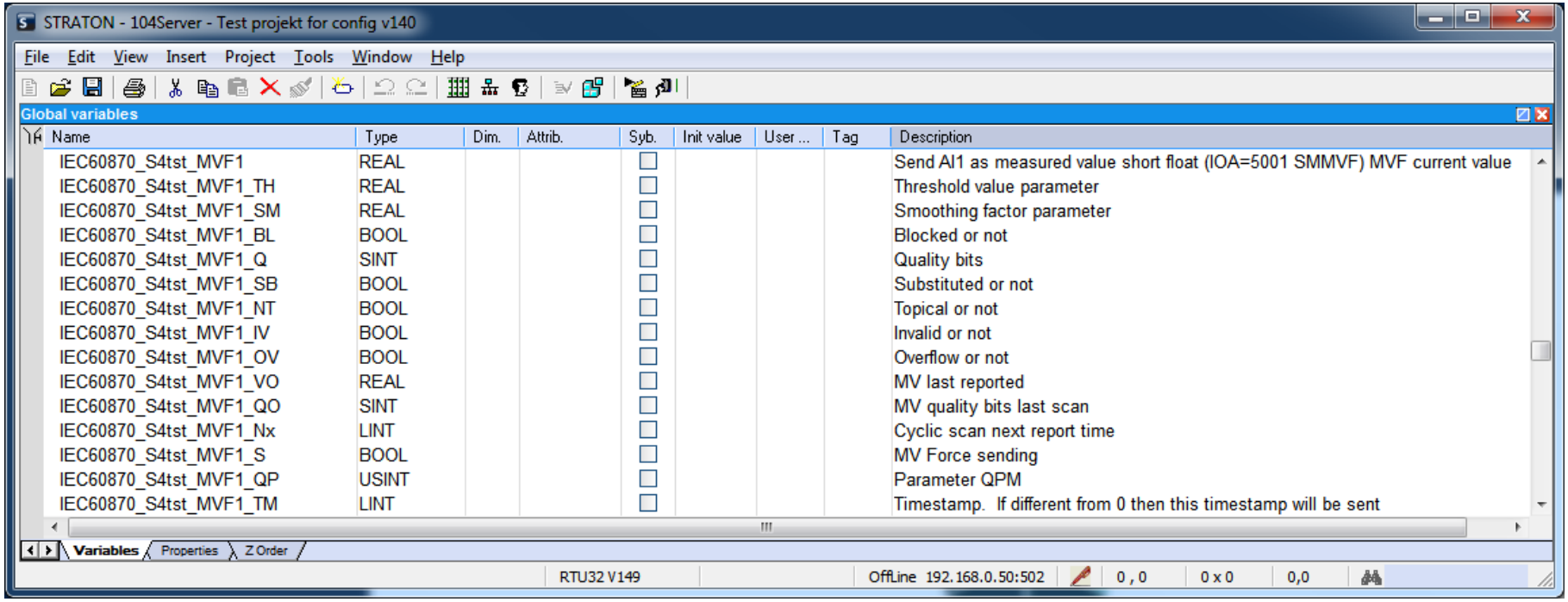
Variables in WorkSuite

- Each variable entered on the variable sheet will created many WorkSuite variables.
- The variable name you enter is used to name all the variables. But the generator adds some extensions.
- The description field gives an indication the purpose of each variable.
- By changing values of some of these variables you can trigger events to be sent.
- The picture shows a Single Point variable called xx_SP defined using the signal type SMSP (send monitor direction single point)



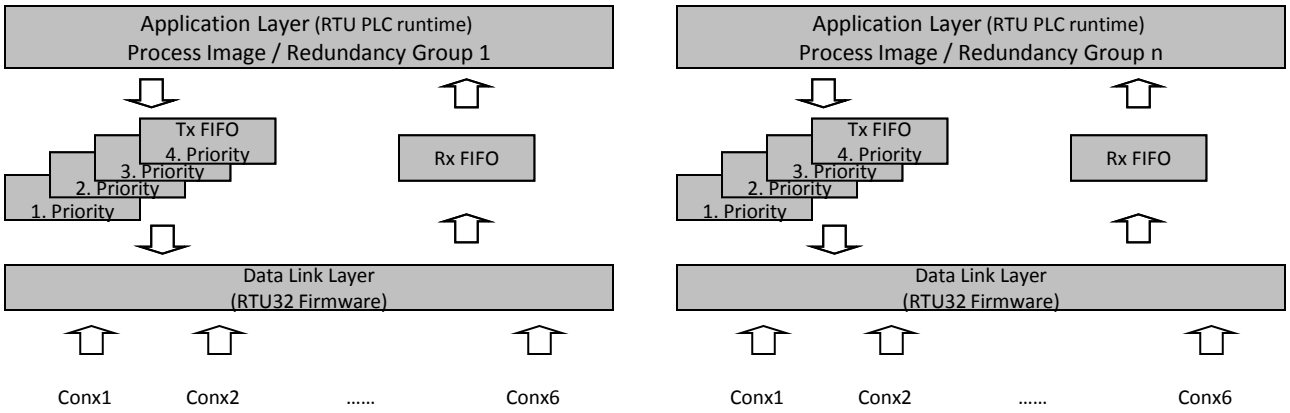
Variables in WorkSuite

The picture shows a Floating Point variable called xx_MVFV defined using the signal type SMMFV (send monitor direction measured value floating point)

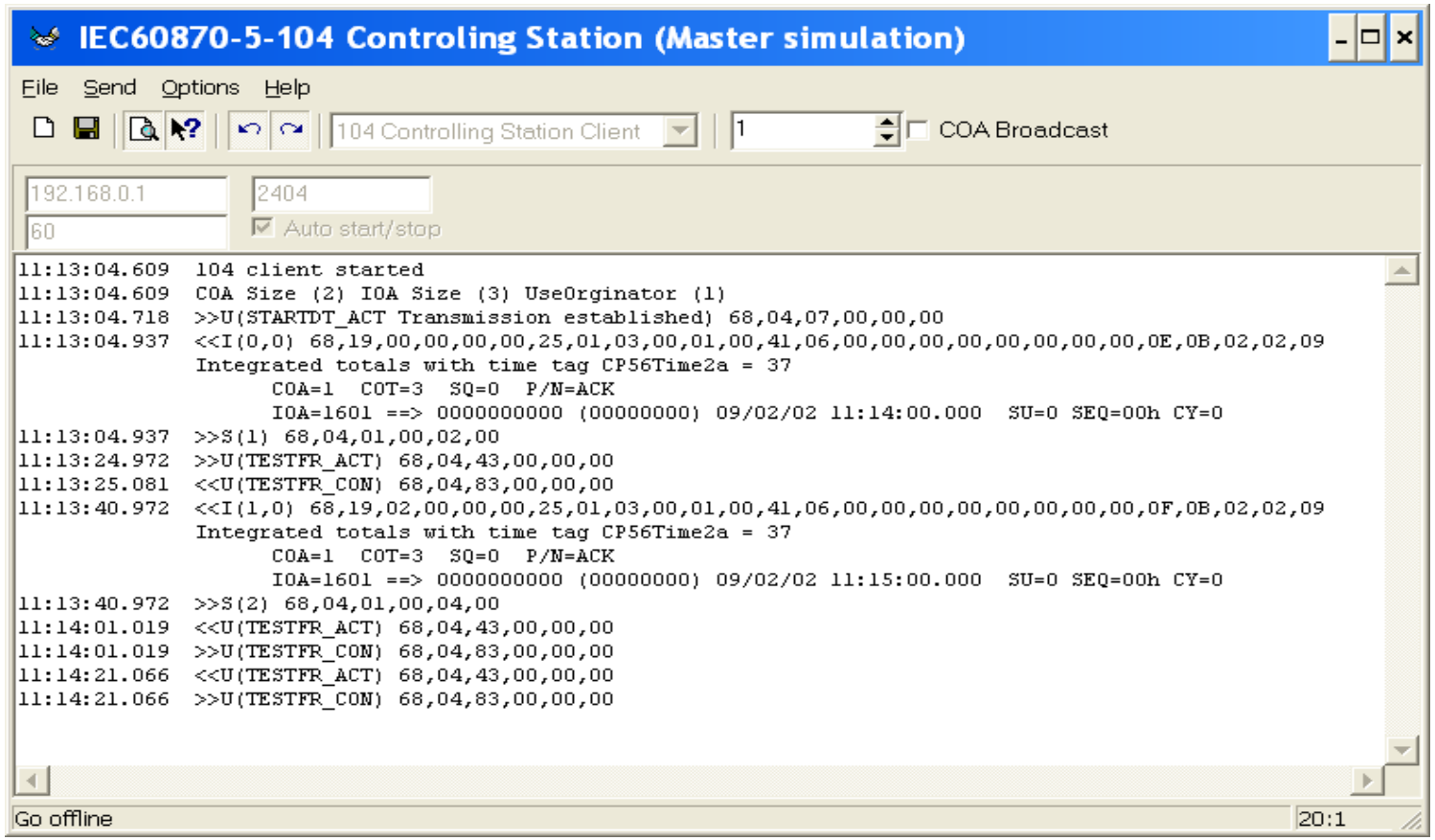


What is Redundancy Group in 104 protocol

- The collection of the variables on the variable sheet is called a Process Image or a Redundancy Group.
- A redundancy group shall rely upon only one process image.
- There can be until 6 redundant connection to a redundancy group. The allowed connections are specified on the 104 Server sheet.
- Time settings can be adjusted individually for each client connection.
- Only one connection is in the started state and sending and receiving data at a time for one redundancy group. The others are in not started state if they are connected.
- The controlling station decides which one of the 6 connection is to be in started state.
- If more than one controlling station need to access the same controlled station simultaneously, each controlling station must be assigned to a different redundancy group (process image)
- Each redundancy group is generated in a separate Excel Sheet. The “Main Program Name” and variable names must be different for each redundancy group.
- Number of redundancy groups is only limited by WorkSuite program space.



IEC60870 101 / 104 Master simulation (test tool)



The XML Database File

General

- The XML file is a database file for the Code Generator. It includes more or less all settings for the Code Generator which are not entered in the Excel sheet.
- All standard WorkSuite code including basic driver settings and functions, and definitions of Excel sheet data readings are stored in a XML file.
- The database XML file does include almost all settings used by the Configurator Tool – formatted and in clear text. An experienced user can enter own special configuration parameters to this XML file.

XML File Structure

- The XML file is divided in 4 basic sections;
- XML file section <Main code>
- XML file section <FunctionBlocks>
- XML file section <SignalTypes>
- XML file section <Macros>

XML File Section <Main code>

- This section includes;
- All main code for the different IEC60870 driver types supported by the Configurator.
- Details of Excel user input file. Specifications like sheet number, row and columns where data is stored.

XML File Section <FunctionBlocks>

- This section includes;
- WorkSuite function blocks for the supported ASDU data types

XML File Section <SignalTypes>

- This section includes;
- Definitions of the different variable types in WorkSuite
- Definitions of all the ASDU types supported by the Configurator.

XML File Section <Macros>

- This section includes macros code that is to be used many places for the various signal types like;
- Macros for TimeTag handling
- Macros for Select and Execute functions.

Note: Editing requires special knowledge of the configurator. Brodersen offers RTU32 Series IEC60870 driver training for advanced users. Contact your local Brodersen office or Brodersen certified partner for details.