Overview

WinPP104 is a test and simulation program for the telecontrol protocol IEC 60870-5-104. You can monitor the messages of an existing connection or simulate the client (master) or the server (substation). The program creates a process image and a TCP 104-connection table, see Menu: "View > Process image or Connections". You can view the process image and the connections online and use it to filter. You can start the program several times and simulate several Masters or Stations. Interfacing to the telecontrol equipment is done via the network interface (LAN, TCP/IP).

System Requirements: Windows 10, 8, 7, Vista or XP, Pentium, 100 MB RAM, 100 MB Disc, Network interface (LAN Ethernet, TCP/IP), USB port for dongle, VGA or better.

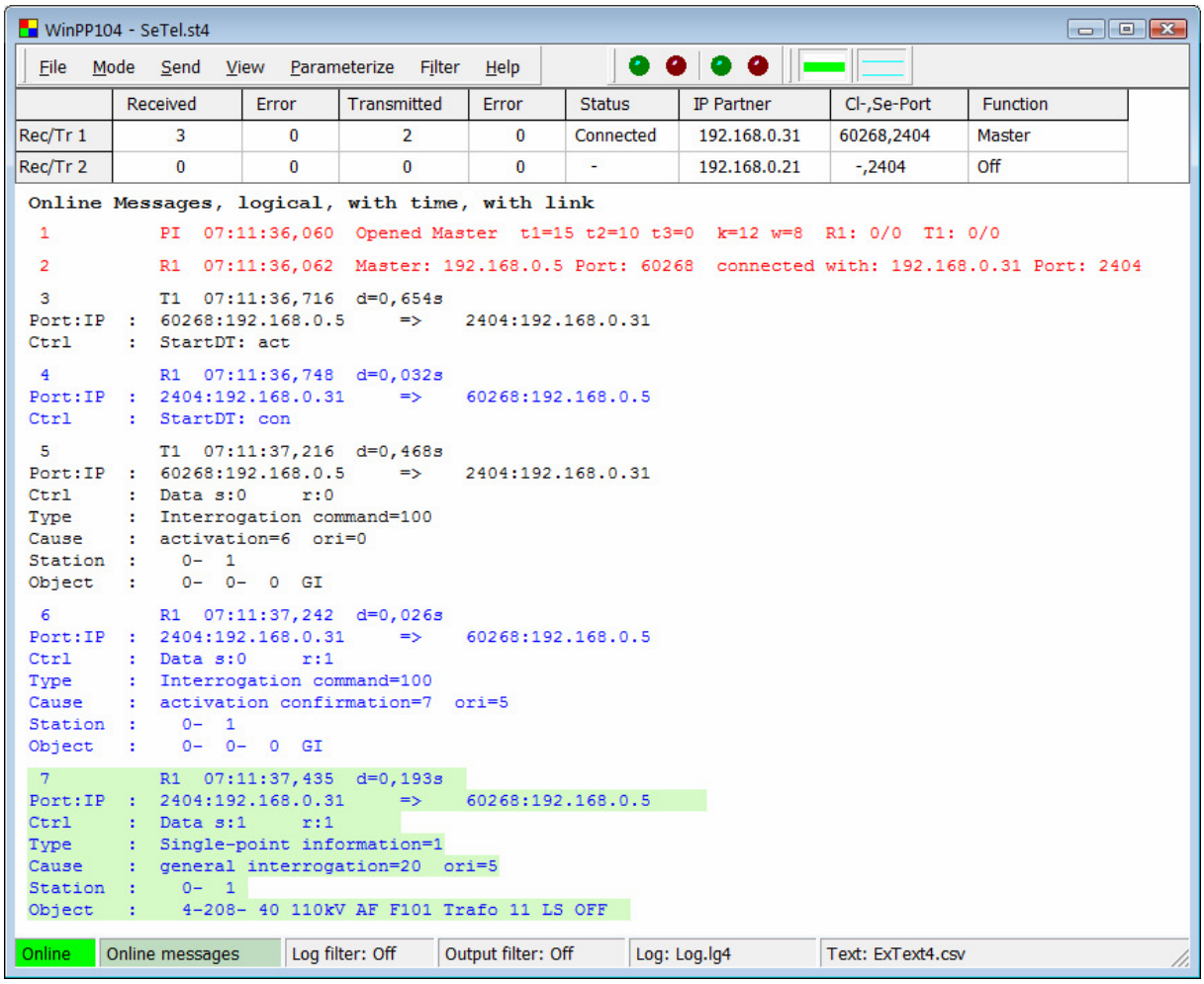


Fig. 1 WinPP104, Online message display

Functionality

Settable parameters (among others):

- Program function (Master, Station, Monitoring)
- Send messages
- Send message lists
- Plain text of the objects, substations and IP addresses (Text file)
- Simulation of command responses, Check delay time of the commands
- Simulation answer of general interrogation
- Messages into csv files send, see help text: message parameterize > type.
- Cyclic transmission of an extern telegram, automatically loading lists, automatically

command responses, see help text: “Parameterize > Option > Cyclic Transm..”

- Structure of station address and of object address
- The parameters t_0, t_1, t_2, t_3, k and w
- IP address of partner station, IP address of the master

The program reads the texts from a text file (csv format) see also file “ExText4.csv” in the program directory. Select a CSV file in the menu "File | Object Texts Load". The currently used file is indicated in the status bar.

The program checks the received messages for correctness. Faulty messages are marked as such. Every transmitted and received message is allocated a time stamp and is stored in a Log file. The size of the Log file can be parameterized (Standard: 2 MB = ca. 45,000 messages). The messages in the Log file can be output to a printer or stored in a text file.

The Log file is organised as a circular buffer. When the file is full then the newest message overwrites the oldest message.

When storing or displaying messages you can filter them with respect to: Message type, Cause of transmission, Quality descriptor, IP address, Port number, Common address, Object address, Originator address, Time, Message number (and others)

The send messages are parameterized logically, see Fig. 2.

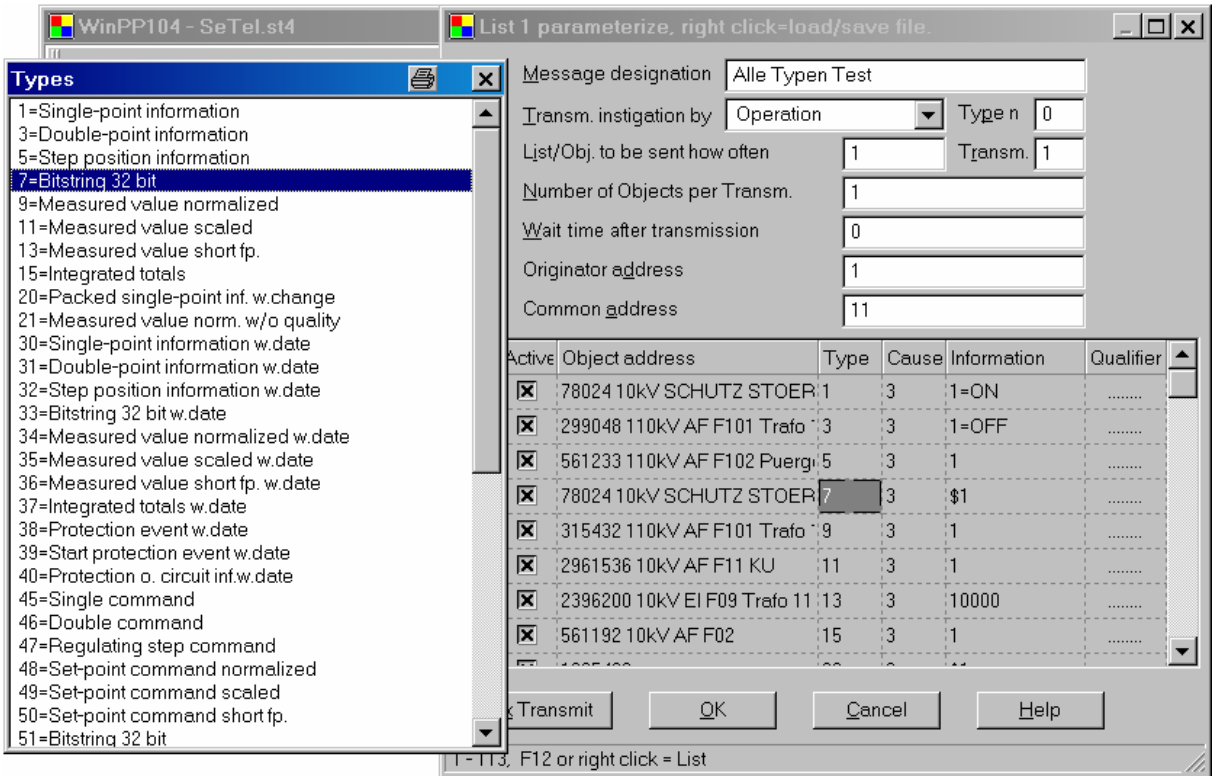


Fig. 2 Parameterizing a message list

There are 12 single messages and 12 message lists available. In a list you can parameterize up to 3000 objects. For the simulation of command responses 1000 objects are available. The transmission instigation for the messages and lists takes place via operation or via an event. An event can be: reception of a particular type of message or successful establishment of a link. You can then send an interrogation command, answer an interrogation command automatically, send commands, simulate responses, transmit cyclic measured values or simulate an avalanche of messages. The parameterized messages and lists can be saved and loaded.

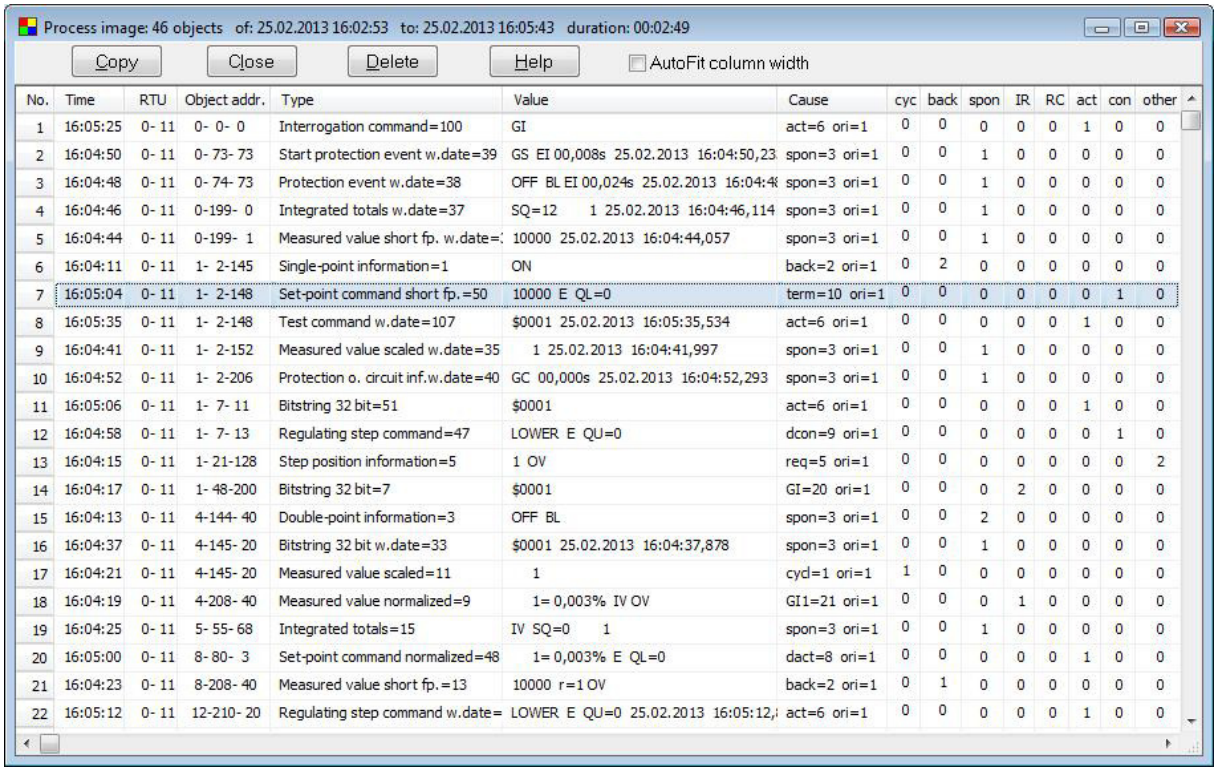
For test purposes you can send illogical messages. For example: send private ASDU, increment the send sequence number by 2, or do not send ACKs, etc.

Process image

When you are monitoring or simulating the program builds a process image. The output is via the menu "View > Process Image" or via the context menu. The process image is useful for a quick overview of the state of the objects and to filter for an object in the log file. You filter for an object by clicking in the "No" column of the row.

Each table row corresponds to an object. The objects are grouped by RTU address, Object address and type.

The number in the "No." column is a sequential number, "time" is the last reception time, "RTU" is the RTU address, "type" is the object type, "value" is the last value of the object, "cause" is the cause of transmission, "cyc back spon IR RC act con other" are counters of the causes of transmission: cyclic, background, spontan, interrogated, requested counter, activation, confirmation, and all other causes.



No.	Time	RTU	Object addr.	Type	Value	Cause	cyc	back	spon	IR	RC	act	con	other
1	16:05:25	0- 11	0- 0- 0	Interrogation command=100	GI	act=6 ori=1	0	0	0	0	0	1	0	0
2	16:04:50	0- 11	0- 73- 73	Start protection event w.date=39	GS EI 00,008s 25.02.2013 16:04:50,23	spon=3 ori=1	0	0	1	0	0	0	0	0
3	16:04:48	0- 11	0- 74- 73	Protection event w.date=38	OFF BL EI 00,024s 25.02.2013 16:04:48	spon=3 ori=1	0	0	1	0	0	0	0	0
4	16:04:46	0- 11	0-199- 0	Integrated totals w.date=37	SQ=12 1 25.02.2013 16:04:46,114	spon=3 ori=1	0	0	1	0	0	0	0	0
5	16:04:44	0- 11	0-199- 1	Measured value short fp. w.date=	10000 25.02.2013 16:04:44,057	spon=3 ori=1	0	0	1	0	0	0	0	0
6	16:04:11	0- 11	1- 2-145	Single-point information=1	ON	back=2 ori=1	0	2	0	0	0	0	0	0
7	16:05:04	0- 11	1- 2-148	Set-point command short fp.=50	10000 E QL=0	term=10 ori=1	0	0	0	0	0	0	1	0
8	16:05:35	0- 11	1- 2-148	Test command w.date=107	$0001 25.02.2013 16:05:35,534	act=6 ori=1	0	0	0	0	0	1	0	0
9	16:04:41	0- 11	1- 2-152	Measured value scaled w.date=35	1 25.02.2013 16:04:41,997	spon=3 ori=1	0	0	1	0	0	0	0	0
10	16:04:52	0- 11	1- 2-206	Protection o. circuit inf.w.date=40	GC 00,000s 25.02.2013 16:04:52,293	spon=3 ori=1	0	0	1	0	0	0	0	0
11	16:05:06	0- 11	1- 7- 11	Bitstring 32 bit=51	$0001	act=6 ori=1	0	0	0	0	0	1	0	0
12	16:04:58	0- 11	1- 7- 13	Regulating step command=47	LOWER E QU=0	dcon=9 ori=1	0	0	0	0	0	0	1	0
13	16:04:15	0- 11	1- 21-128	Step position information=5	1 OV	req=5 ori=1	0	0	0	0	0	0	0	2
14	16:04:17	0- 11	1- 48-200	Bitstring 32 bit=7	$0001	GI=20 ori=1	0	0	0	2	0	0	0	0
15	16:04:13	0- 11	4-144- 40	Double-point information=3	OFF BL	spon=3 ori=1	0	0	2	0	0	0	0	0
16	16:04:37	0- 11	4-145- 20	Bitstring 32 bit w.date=33	$0001 25.02.2013 16:04:37,878	spon=3 ori=1	0	0	1	0	0	0	0	0
17	16:04:21	0- 11	4-145- 20	Measured value scaled=11	1	cycl=1 ori=1	1	0	0	0	0	0	0	0
18	16:04:19	0- 11	4-208- 40	Measured value normalized=9	1= 0,003% IV OV	GI1=21 ori=1	0	0	0	1	0	0	0	0
19	16:04:25	0- 11	5- 55- 68	Integrated totals=15	IV SQ=0 1	spon=3 ori=1	0	0	1	0	0	0	0	0
20	16:05:00	0- 11	8- 80- 3	Set-point command normalized=48	1= 0,003% E QL=0	dact=8 ori=1	0	0	0	0	0	1	0	0
21	16:04:23	0- 11	8-208- 40	Measured value short fp.=13	10000 r=1 OV	back=2 ori=1	0	1	0	0	0	0	0	0
22	16:05:12	0- 11	12-210- 20	Regulating step command w.date=	LOWER E QU=0 25.02.2013 16:05:12,	act=6 ori=1	0	0	0	0	0	1	0	0

The above example is an excerpt from a process image with 46 objects.

"Copy" copies the objects in CSV format to the clipboard.

"Delete" will delete the objects data.

"AutoFit Column Width" If selected, the columns will change when outputting for the longest text.

TCP 104 connections

While listening, the program is building a TCP 104 connection table. The output is via the menu "View > Connections" or via the context menu. The connection table is useful for a quick overview of the existing connections, to filter for a connection, to monitor unsafe connections (wireless connection) or when the devices do not behave as expected. To filter for a connection, click in the column "CNo." (Connection number) of the respective row.

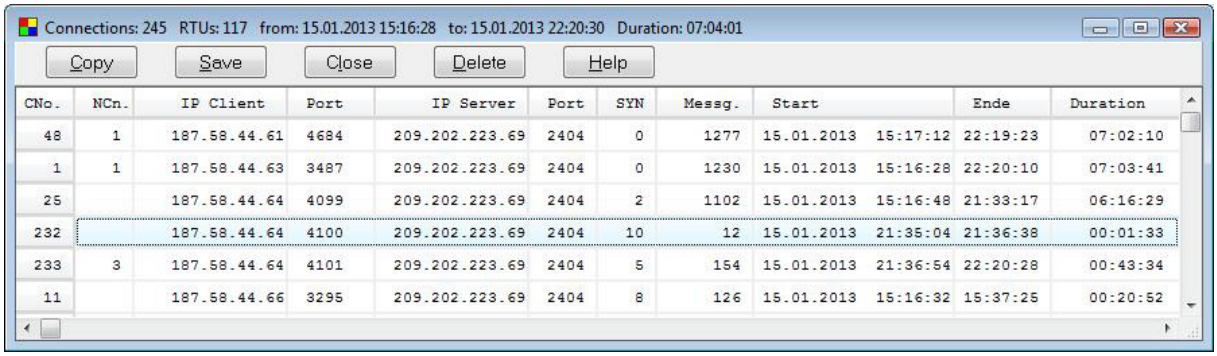
The number in the "CNo." column indicates the chronological order in which the connections were monitored. "1" corresponds to the first connection.

The column "NCn" indicates the number of connections per station. If not specified, there are multiple connections to this station. In the last connection is then the number of connections.

Each table row corresponds to a connection. The connections are sorted by IP client, IP server, Port server, CNo.

In the column "SYN" You can see the number of messages with SYN, FIN or RESET bit. A number greater than zero indicates connection, disconnections while monitoring. "Messg." specifies the number of received messages. "Start" and "end" is the time of the first or last listened telegram. "Duration" is the difference between start and end. Details are shown in the log file.

Is the time interval between two telegrams greater than two hours, the new message a new connection is assigned. The port number of the client is extended with "-n", wherein n is 2, 3, 4 and so on.



CNo.	NCn.	IP Client	Port	IP Server	Port	SYN	Messg.	Start	Ende	Duration
48	1	187.58.44.61	4684	209.202.223.69	2404	0	1277	15.01.2013 15:17:12	22:19:23	07:02:10
1	1	187.58.44.63	3487	209.202.223.69	2404	0	1230	15.01.2013 15:16:28	22:20:10	07:03:41
25		187.58.44.64	4099	209.202.223.69	2404	2	1102	15.01.2013 15:16:48	21:33:17	06:16:29
232		187.58.44.64	4100	209.202.223.69	2404	10	12	15.01.2013 21:35:04	21:36:38	00:01:33
233	3	187.58.44.64	4101	209.202.223.69	2404	5	154	15.01.2013 21:36:54	22:20:28	00:43:34
11		187.58.44.66	3295	209.202.223.69	2404	8	126	15.01.2013 15:16:32	15:37:25	00:20:52

The above example is an excerpt from a table with 245 connections and 117 stations. The station 187.58.44.64 has established three connections.

"Copy" copies the connection data in CSV format to the clipboard.

"Save" saves the connection data in the log file.

"Close" closes the window.

"Delete" will delete the connection data.