

Overview

WinPP101 is a test and simulation program for the telecontrol protocol IEC 60870-5-101. You can monitor the Messages of an existing link or simulate the master or the substation. The program creates a process image, see Menu: "View > Process image". You can view the process image online and use it to filter. Interfacing to the telecontrol equipment is done via the serial interface COM. The program supports two COM interfaces. It is possible to run the program several times simultaneously and thus support several COM interfaces. Do you want to monitor the command and monitoring direction simultaneously, you need two COM ports. For simulation a COM port is sufficient.

Additionally, it can listen and logically output ABB RP570 and RP571 telegrams. For test purposes it can also be used as a "Format 1.2" or as a "Byte" receiver, here you may set the parity, number of data bit and stop bit.

System Requirements: Windows 7, Vista or XP, Pentium, 100 MB RAM, 100 MB Disc, at least 1 COM port, USB port for dongle, VGA or better.

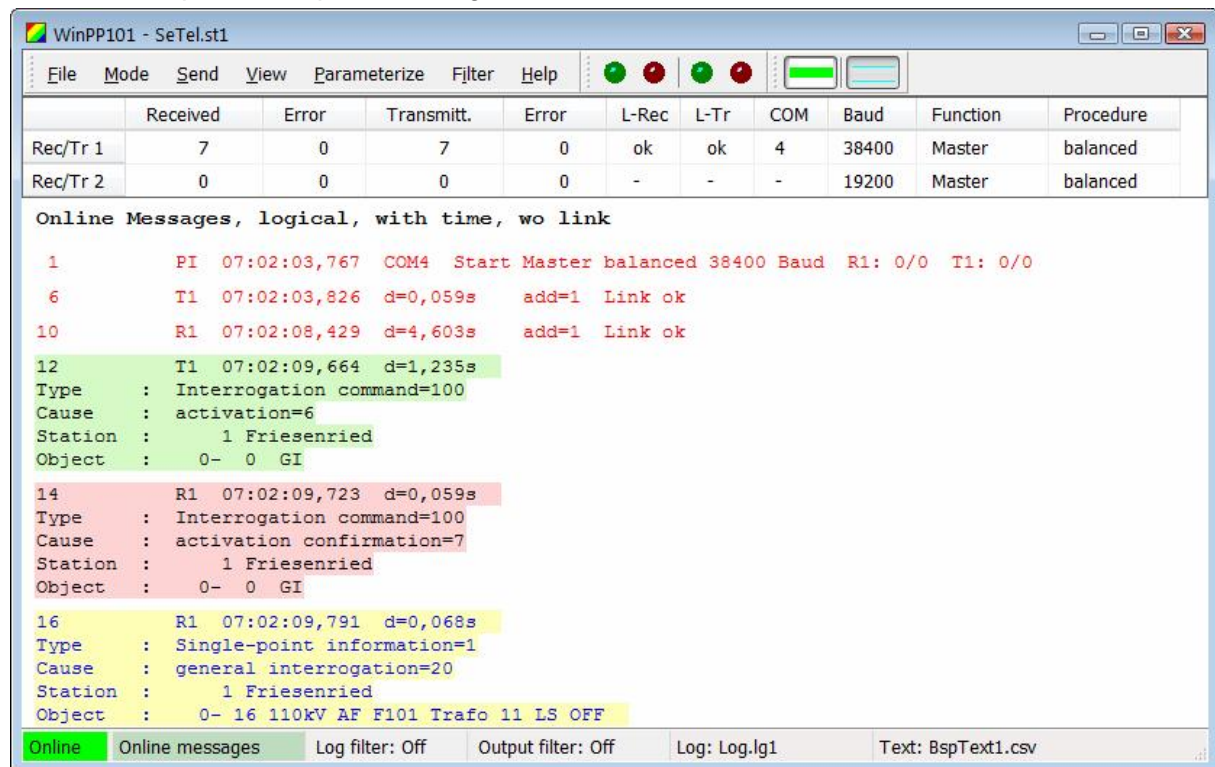


Fig. 1 WinPP101, Online message display

Functionality

You can configure the following parameters for example:

- Ø Program function (monitoring, master, station, byte receiver)
- Ø Plain text of the objects (CSV-file)
- Ø Simulation of command responses
- Ø Simulation answer of general interrogation
- Ø Messages into csv files send, see help text: message parameterize, type
- Ø Cyclic transmission of an extern telegram, loading lists, command responses, see help text: options parameterize
- Ø Structure of common and object address
- Ø Transmission mode (balanced/unbalanced)
- Ø Length of link address and of cause of transmission
- Ø COM Port, Baud rate

The program reads the plaintexts of the individual objects from a csv file, format: "Object address; Color code; Plaintext;"; see also file "ExText8.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 transmission errors, link faults and ASDU faults. 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 organized 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:

- Ø Message type, Cause of transmission, Quality descriptor
- Ø Common address, Object address
- Ø Time, message no, link address (and others)

With the time filter you can specify, for example those only messages from 02:00 till 08:00 should be stored.

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

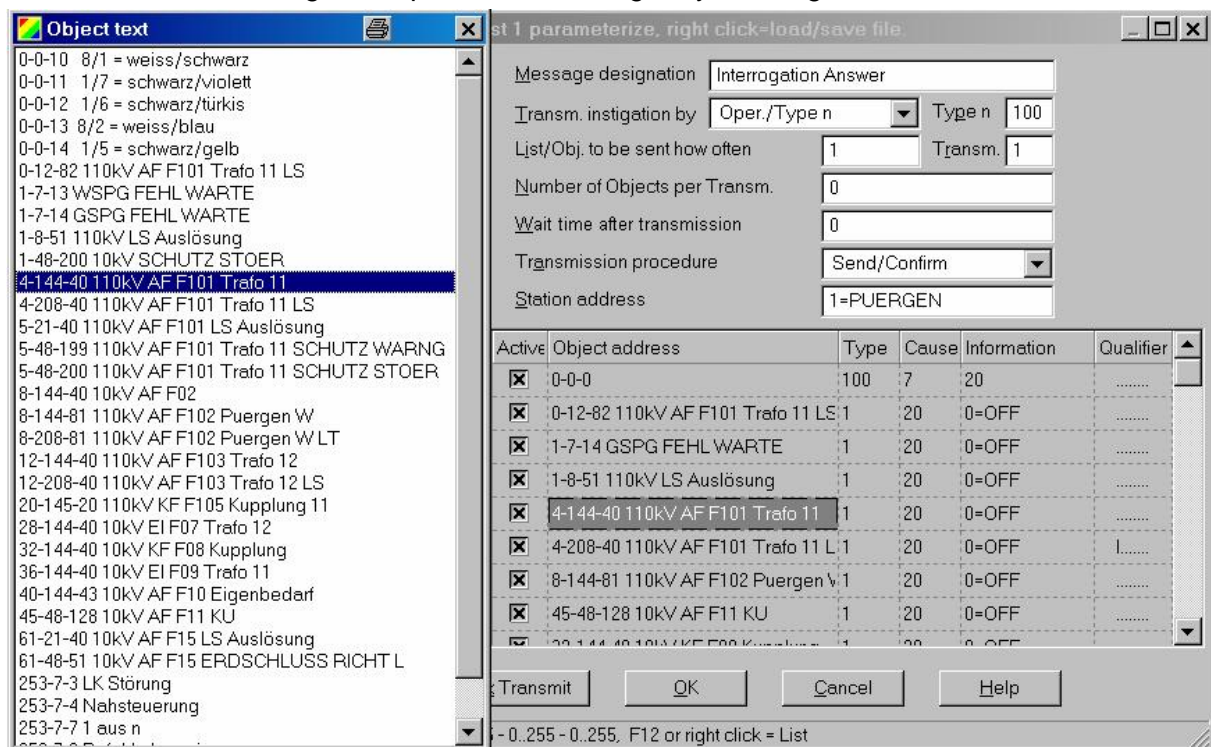


Fig. 2 Parameterizing a message list

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 link and data messages. For examples: send NACK instead of ACK, do not toggle the FCB bit, use private ASDU, send the check character incorrectly, 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. To 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	08:42:48	0- 1	0- 0- 2	Single-point information w.time=2	OFF 42:48,227	rim=11	0	0	0	0	0	0	1	0
2	08:42:49	0- 1	0- 0- 3	Double-point information=3	OFF IV	back=2	0	1	0	0	0	0	0	0
3	08:42:51	0- 1	0- 0- 4	Double-point information w.time=4	ON 42:51,409	req=5	0	0	0	0	0	0	0	1
4	08:42:52	0- 1	0- 0- 5	Step position information=5	33 IV BL	GI=20	0	0	0	1	0	0	0	0
5	08:42:46	0- 1	0- 0- 10	Single-point information=1	ON	spon=3	0	0	1	0	0	0	0	0
6	08:43:24	0- 1	0- 0- 10	Bitstring 32 bit w.date=33	\$7530 25.02.2013 08:43:24,291	back=2	0	1	0	0	0	0	0	0
7	08:43:45	0- 1	0- 0- 37	IEC-104_Set-point command, short fp.=50	2,2E010 E QL=0	act=6	0	0	0	0	0	1	0	0
8	08:42:59	0- 1	1- 1- 9	Measured value, normalized=9	1000= 3,053%	cyd=1	1	0	0	0	0	0	0	0
9	08:43:00	0- 1	1- 1- 10	Measured value, normalized w.time=10	-1000=-3,053% 43:00,768	back=2	0	1	0	0	0	0	0	0
10	08:43:02	0- 1	1- 1- 11	Measured value, scaled=11	10000	cyd=1	1	0	0	0	0	0	0	0
11	08:43:03	0- 1	1- 1- 12	Measured value, scaled w.time=12	-10000 43:03,896	back=2	0	1	0	0	0	0	0	0
12	08:43:05	0- 1	1- 1- 13	Measured value, short fp.=13	1,234568E007	cyd=1	1	0	0	0	0	0	0	0
13	08:43:07	0- 1	1- 1- 14	Measured value, short fp. w.time=14	-100000 43:07,008	back=2	0	1	0	0	0	0	0	0
14	08:43:08	0- 1	1- 1- 15	Integrated totals=15	SQ=0 123.456.789	spon=3	0	0	1	0	0	0	0	0
15	08:43:10	0- 1	1- 1- 16	Integrated totals w.time=16	SQ=0 -123.456.789 43:10,127	spon=3	0	0	1	0	0	0	0	0
16	08:43:11	0- 1	1- 1- 17	Protection event=17	OFF 00,000s 43:11,688	spon=3	0	0	1	0	0	0	0	0
17	08:43:13	0- 1	1- 1- 18	Start protection event=18	SL3 SL2 00,000s 43:13,247	spon=3	0	0	1	0	0	0	0	0
18	08:43:14	0- 1	1- 1- 19	Protection output circuit inf.=19	CL2 CL1 00,000s 43:14,809	spon=3	0	0	1	0	0	0	0	0
19	08:43:16	0- 1	1- 1- 20	Packed single-point inf. w.change=20	\$55AA1122	back=2	0	1	0	0	0	0	0	0
20	08:43:17	0- 1	1- 1- 21	Measured value, norm. w/o quality=21	2000= 6,105%	cyd=1	1	0	0	0	0	0	0	0
21	08:43:19	0- 1	1- 1- 22	Single-point information w.date=30	ON 25.02.2013 08:43:19,489	back=2	0	1	0	0	0	0	0	0
22	08:43:21	0- 1	1- 1- 23	Double-point information w.date=31	ON 25.02.2013 08:43:21,047	back=2	0	1	0	0	0	0	0	0
23	08:43:27	0- 1	1- 1- 27	Measured value, scaled w.date=35	10000 25.02.2013 08:43:27,537	back=2	0	1	0	0	0	0	0	0

The above example is an excerpt from a process image with 51 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.