29.3 IEC 60870-5-104

IEC 60870-5-104 Master

IEC 60870-5-104 protocol (in short IEC 104) is a part of IEC Telecontrol Equipment and Systems Standard IEC 60870-5 that provides a communication profile for sending basic telecontrol messages between two systems in electrical engineering and power system automation. Telecontrol means transmitting supervisory data and data acquisition requests for controlling power transmission grids.

IEC 104 provides the network access to IEC 60870-5-101 (in short IEC 101) using standard transport profiles. In simple terms, it delivers IEC 101 messages as application data (L7) over TCP, usually port 2404. IEC 104 enables communication between control station and a substation via a standard TCP/IP network. The communication is based on the client-server model.

To set up TLS connection for both IEC104 Master and Slave, refer to sections Excel configuration and Certificates. All keys and certificates should be provided in the PEM format.

If no configuration is set up, IEC104 Master and Slave services are not started.

Configuring IEC 104 Master datapoints

To use IEC 60870-5-104 Master in WCC Lite, it has to be configured via an Excel configuration. This configuration contains two Excel sheets where parameters have to be filled in Devices and Signals.

IEC 60870-5-104 Master parameters for *Devices* tab

Parameter	**Type**	**Description**	**Required**	**Default Value** (when not specified)	**Range**	
					Min	Max
name	string	User-friendly name for a device	Yes			
description	string	Description of a device	No			
device_alias	string	Alphanumeric string to identify a device	Yes			
enable	boolean	Enabling/disabling of a device	No	1	0	1
protocol	string	Protocol to be used	Yes		IEC 60870-5-104 master	
asdu_address	integer	Application Service Data Unit address	Yes		0	65535
asdu_size	integer	Common address size in bytes	No	2	1	3
time_sync_interval_sec	integer	Time frame between Time Synchronization requests in seconds	No	60		
gi_interval_sec	integer	Time frame between General Interrogation requests in seconds, if 0 requests are disabled	No	300		
port	integer	TCP port	Yes		0	65535

ioa_size	integer	Information object address (IOA) size in bytes	No	3	1	3
swt	integer	Send window (SWT)	Yes			
rwt	integer	Receive window (RWT)	Yes			
cot_size	integer	Cause of transmission (COT) size in bytes	No	2	1	2
host	string	Host IP address (ipv4)	Yes			
t1	integer	Acknowledge timeout t1 (sec)	Yes			
t2	integer	Connection ACKRSN clock t2 (sec)	Yes			
t3	integer	Connection TESTFR clock t3 (sec)	Yes			
originator	integer	Provides a means for a controlling station toexplicitly identify itself	No	0	0	255

IEC 60870-5-104 Master parameters for *Signals* tab

Parameter	**Type**	**Description**	**Required**	**Default Value** (when not specified)	**Range**	
					Min	Max
signal_name	string	User-friendly signal name	Yes			
device_alias	string	Alphanumeric string to identify a device	Yes			
signal_alias	string	Unique alphanumeric name of the signal to be Yes used	Yes			
source_device_alias	string	device_alias of a source device	For commands			
source_signal_alias	string	signal_alias of a source signal	For commands			
enable	boolean	Enabling/disabling of an individual signal	No	1	0	1
log	integer	Allow signal to be logged. If **log is 0 signal** will not be logged. If **log is more than 0** signal will be logged	No	0		
gi	boolean	Including/excluding (1 or 0) signal from General Interrogation	No	0	0	1
common_address	integer	Address of a destination device	Yes			
function	integer	Function number	No	0		

info_address	integer	Information object address	Yes			
data_type	integer	ASDU type identificator	No			
select_ms	integer	Time limit in milliseconds for command execution. Command select has to be performed before execution if this parameter is specified. Direct command execution can be performed only if this field is left empty or set to zero.	No	0		

Debugging a IEC 60870-5-104 Master aplication

If configuration for IEC 60870-5-104 devices is set up, the handler for the protocol will start automatically. If a configuration is missing or contains errors, the protocol will not start. It is done intentionally to decrease unnecessary memory usage.

If IEC 60870-5-104 does not work properly (e.g. no communication between devices, data is corrupted, etc.), a user can launch a debug session from command-line interface and find out why link is not functioning properly or use WCC Utility to do that.

To launch a debugging session, a user should stop the *iec104-master* process and run the *iec104-master* command with respective flags.

- Step 1: Service must be stopped by entering the following command into the wcclite: **/etc/init.d/iec104-master stop**
- Step 2: After service is stopped it must be started with the preferred configuration file (JSON files found in /etc/ folder) and a debug level 7: **iec104-master -c /etc/iec104-master/iec104-master.json -d7**
- Step 3: Once the problem is diagnosed normal operations can be resumed with the following command: **/etc/init.d/iec104-master start**

IEC 60870-5-104 command line debugging options

```
-h [ –help ] Display help information
-V [ –version ] Show version
-d<debug level> Set debugging level
-c [ –config ] Config path
-r [ –raw ] Show raw telegram data
-f [ –frame ] Show frame data
-e [ –redis ] Show redis message
-R [ –readyfile ] Ready notification file
```

IEC 60870-5-104 Slave

IEC 60870-5-104 Slave is designed not to lose data acquired from Master protocols. The data that arrives from Master protocols is stored in cache. This data is checked every second to manage further data sending. The data that leaves IEC 60870-5-104 Slave has output caches. They’re built to provide switching between multiple sessions (redundant SCADA). If a new connection arrives, the old one is dropped, but data, that is stored in cache, not sent and not confirmed by SCADA is transfered to new connection.

Configuring IEC 104 Slave datapoints

To use IEC 60870-5-104 Slave in WCC Lite, it has to be configured via an Excel configuration. This configuration contains two Excel sheets where parameters have to be filled in Devices and Signals.

IEC 60870-5-104 Slave parameters for *Devices* tab

Parameter	**Type**	**Description**	**Required**	**Default Value** (when not specified)	**Range**	
					Min	Max

name	string	User-friendly name for a device	Yes			
description	string	Description of a device	No			
device_alias	string	Alphanumeric string to identify a device	Yes			
enable	boolean	Enabling/disabling of a device	No	1	0	1
protocol	string	Protocol to be used	Yes		IEC 60870-5-104 slave	
asdu_size	integer	Common address size in bytes	No	2	1	3
time_sync	boolean	Enable/disable (1 or 0) time synchronization	Yes			
port	integer	TCP port	No	2404	0	65535
ioa_size	integer	Information object address (IOA) size in bytes	No	3	1	3
swt	integer	Send window (SWT)	No	12		
rwt	integer	Receive window (RWT)	No	8		
cot_size	integer	Cause of transmission (COT) size in bytes	No	2	1	2
host	string	Space separated remote host IP addresses (ipv4)	Yes			
bind_address	string	Bind to local IP address (ipv4)	No	0.0.0.0		
t1	integer	Acknowledge timeout t1 (sec)	Yes			
t2	integer	Connection ACKRSN clock t2 (sec)	Yes			
t3	integer	Connection TESTFR clock t3 (sec)	Yes			
sp_time	boolean	Add (1 or 0) CP56Time2a information to single point signals	No	0	0	1
dp_time	boolean	Add (1 or 0) CP56Time2a information to double point signals	No	0	0	1
me_time	boolean	Add (1 or 0) CP56Time2a information to measurements	No	0	0	1
message_size	boolean	Maximum length of a message	Yes		0	255
cache_size	integer		Yes		0	1000
tls	boolean	Enable/disable use of TLS	No	0	0	1

tls_local_certificate	string	Local certificate for TLS connection	Yes (for TLS)			
tls_peer_certificate	string	Certificate authority file for TLS connection	No			
tls_private_key	string	File consisting of private key for TLS connection	No			

IEC 60870-5-104 Slave parameters for *Signals* tab

Parameter	**Type**	**Description**	**Required**	**Default Value** (when not specified)	**Range**	
					Min	Max
signal_name	string	User-friendly signal name	Yes			
device_alias	string	Alphanumeric string to identify a device	Yes			
signal_alias	string	Unique alphanumeric name of the signal to be Yes used	Yes			
source_device_alias	string	device_alias of a source device	For commands			
source_signal_alias	string	signal_alias of a source signal	For commands			
enable	boolean	Enabling/disabling of an individual signal	No	1	0	1
log	integer	Allow signal to be logged. If **log is 0 signal** will not be logged. If **log is more than 0** signal will be logged	No	0	0	1
gi	boolean	Including/excluding (1 or 0) signal from General Interrogation	No	0	0	1
common_address	integer	Address of a destination device	Yes			
info_address	integer	Information object address	Yes			
data_type	integer	ASDU type id. Types are identified automatically if this field is not set.	No	0		
select_ms	integer	Time limit in milliseconds for command execution. Command select has to be performed before execution if this parameter is specified. Direct command execution can be performed only if this field is left empty or set to zero.	No	0		

Debugging a IEC 60870-5-104 Slave aplication

If configuration for IEC 60870-5-104 devices is set up, the handler for the protocol will start automatically. If a configuration is missing or contains errors, the protocol will not start. It is done intentionally to decrease unnecessary memory usage.

If IEC 60870-5-104 does not work properly (e.g. no communication between devices, data is corrupted, etc.), a user can launch a debug session from command-line interface and find out why the link is not functioning properly or use WCC Utility to do that.

To launch a debugging session, a user should stop the *iec104-slave* process and run the *iec104-slave* command with respective flags.

- Step 1: Service must be stopped by entering the following command into the wcclite: **/etc/init.d/iec104-slave stop**
- Step 2: After service is stopped it must be started with the preferred configuration file (JSON files found in /etc/ folder) and a debug level 7: **iec104-slave-c /etc/iec104-slave/iec104-slave.json -d7**
- Step 3: Once the problem is diagnosed normal operations can be resumed with the following command: **/etc/init.d/iec107-slave start**

IEC 60870-5-10 command line debugging options

```
-h [ –help ] Display help information
-V [ –version ] Show version
-d<debug level> Set debugging level
-c [ –config ] Config path
-r [ –raw ] Show raw telegram data
-f [ –frame ] Show frame data
-e [ –redis ] Show redis message
-R [ –readyfile ] Ready notification file
```

Revision #10
★Created 11 October 2021 08:25:57 by Tautvilis
Updated 5 May 2022 08:23:09 by Lukas Taroza