

How to Maintain Stable Ethernet Connection Without Disconnection

1. Enable KeepAlive Function

In the main.lua code file, find the following line of code:

```
LIB_IEC61850ServerStart("SPI_1",ETHRST_PIN,mac,ip,subm,gw,dns,port,60)
```

The example code above has already enabled KeepAlive with a time = 60 seconds. When greater than 0 seconds, it will periodically send KeepAlive probe packets to the peer client. If the probe fails, it will disconnect the TCP connection, which prevents the local machine from maintaining an invalid TCP connection, thus allowing the client to reconnect to the local machine.

Note: In all the reference examples we provide in main.lua code files, KeepAlive is enabled by default for 60 seconds. Users can modify this value as needed.

2. Add NetAutoFix Network Self-Recovery Function

Open the main.lua code file, add the LIB_IEC61850NetAutoFixCfg() configuration function before the code enters the main loop, and add the LIB_IEC61850NetFixDeamon() function in the main loop. The example code after adding is as follows:

```
-- Add all required 61850 telemetry data points
LIB_IEC61850AddNode("YC_RM", YC_List)
-- Enable system 10ms timer to start working
LIB_10msTimerConfig("ENABLE")
-- Configure W5500 Ethernet module SPI: MISO,MOSI,CLK,CS use D1D2D3D4 pins, RST uses D5
pin
-- and start IEC61850 MMS Server service (TCP Server), keepalive=60s, disable SNTP request
LIB_IEC61850ServerStart("SPI_1",ETHRST_PIN,mac,ip,subm,gw,dns,port,60)--,
{192,168,0,105},123,30)
-- Auto restart system if no message received for 7200 seconds
LIB_IEC61850NetAutoFixCfg(7200, "SYS_REBOOT")
-- Enter main loop
while(GC(1) == true)
do
    -- Update telemetry and data point values every MB_POLL_TIME_MS milliseconds
    if UpdatePeriodCntMs > MB_POLL_TIME_MS then
        UpdatePeriodCntMs = 0

        -- Read Modbus slave data and sync to IEC61850
        ModbusToIec61850()
    end
    LIB_IEC61850NetFixDeamon()
end
```

If your code already has the `LIB_IEC61850NetAutoFixCfg()` and `LIB_IEC61850NetFixDaemon()` functions, please check and remove the "--" comment symbols in front.

Explanation of configuration parameters in `LIB_IEC61850NetAutoFixCfg(7200, "SYS_REBOOT")`:

The two parameters **7200** and **"SYS_REBOOT"** define: if no message is received from the peer for 7200 consecutive seconds, the local machine will automatically restart (including the Ethernet interface restart). This strategy is different from the KeepAlive above - it is an upper-layer monitoring strategy at the application layer that monitors whether the IEC61850 master station has conducted 61850-level Ethernet TCP data communication with this protocol converter within 7200 seconds. In the case of no monitoring for 7200 consecutive seconds, this protocol converter will automatically execute device reset restart and Ethernet TCP communication re-initialization configuration.

Some important considerations:

- The NetAutoFix function is not mandatory, and may even be prohibited. For example, it's also normal for the 61850 master station to manually set a situation where it doesn't communicate with this protocol converter for 7200 consecutive seconds.
- The NetAutoFix function cannot solve all network disconnection situations, such as when the 61850 master station's own Ethernet TCP network communication has problems, or when the switch network has problems.
- The NetAutoFix function is somewhat like a watchdog for the device's own 61850 communication, used to ensure that when the device itself has problems, it can recover network communication in time (the probability of this situation is also extremely low, at least we haven't encountered it so far, so NetAutoFix is not added by default in our main.lua code).
- Finally, if users feel that adding NetAutoFix protection will give them more peace of mind, that's also fine. However, before adding it, they need to work with the 61850 master station designers to ensure that the communication interval between the master station and this protocol converter will not be greater than the set time (e.g., 7200 seconds).