LANSEN

WIRELESS BUILDING TECHNOLOGY

CONFIGURE LAN-WMBUS-C-T(H) V1.0

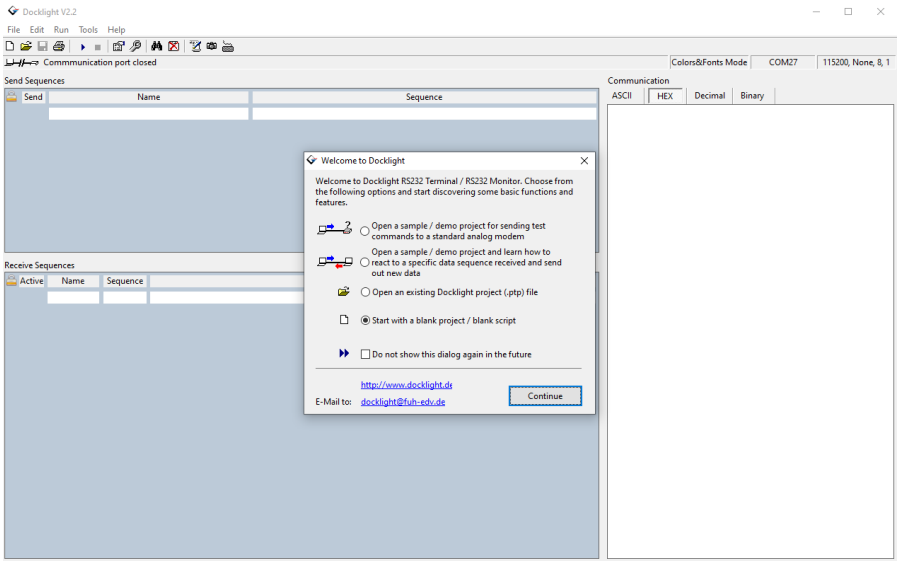
Contents

Setup Docklight + script

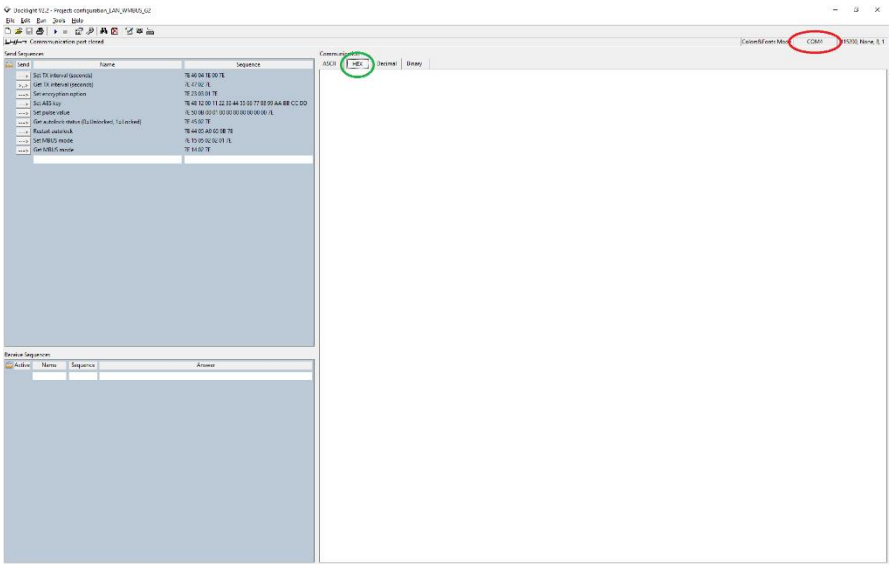
To configure a device, our script *configuration_LAN_WMBUS_C_T(H)* must be used together with a program – In this case Docklight. If you have not received the script, contact us for more information and we will provide it to you.

The program can be download using the following link: https://docklight.de/download/Docklight.zip

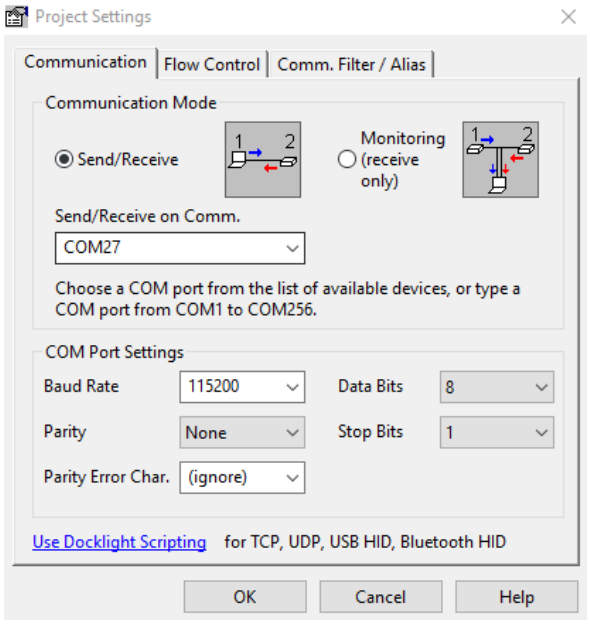
Installing Docklight and start the program - This will bring up the figure below. Select the option "**Open an existing Docklight project (.ptp) file**" and select our script.



Once the script has been loaded, all available options will be presented in the program. Before proceeding, make sure the program is set to HEX (green circle in picture below) and that the configuration cable is inserted in the USB-connection on your laptop/computer.



Double-click in the field marked by the red circle in the figure above. This will bring up a popup called “**Project Settings**”, such as figure below. Click in the field “**Send/receive on Comm.**”, choose the correct USB-port and then click on the button “**OK**”.



Connect configuration cable to a device

Separate the front piece from the back. Locate the text “**CONFIG**” on the PCB – This is where the configuration cable is to be inserted. Make sure to match the three holes on the PCB with the pins on the configuration cable.

Check configuration status of device

In Docklight, click on the play symbol (▶). The first thing to do now is to check if configuration is enabled on the connected device. Follow the steps below.

1. Send the command **Get autolock status** by clicking on the arrow (--->) to the left of the text. See chapter

2. **Get autolock** status for information on how to interpret the packets.

3. Check the return value from the device and compare with the options below.
   - If return value is 0x00, then the device can be configured. Proceed with configuration.
   - If return value is 0x01, then configuration must be enabled. Go to step 3.
   - If the return value is 0x02, then the first three bytes of the AES-key were incorrect. Go to step 4.

4. Adjust the command **Restart autolock** according to chapter

6. Restart autolock and then send the command. Check the return value.
    - If return value is 0x00, then the device can be configured. Proceed with configuration.
    - If return value is 0x02, then the first three bytes of the AES-key was incorrect. Go to step 4.

7. The first three bytes was incorrect. Adjust the command **Restart autolock** and make sure the correct first three bytes of the AES-key is used.

    Note: For each failed attempt, the device will add 30 seconds before a new retry is possible. So first fail needs 30 s before a retry, second fail needs 60 s, up to maximum 4.5 hours.

    Send the command **Restart autolock** again.
    - If the return value is 0x02, then the first three bytes of the AES-key was incorrect. Redo this step.
    - If the return value is 0x00, then the device can be configured. Proceed with configuration.

Important! Serial protocol to and from the sensor

Each frame begins and ends with a ‘flag sequence’ and always consists of the binary sequence 01111110 (hexadecimal 0x7E). This ‘flag sequence’ is a frame separator and only one flag sequence is required between two frames, i.e., the end of one frame is the beginning of another. Two consecutives ‘flag sequence’ constitute an empty frame, which is ignored, and not counted as an FCS-error.

To avoid 0x7E in the actual data special care must be taken when sending 0x7E and 0x7D to the G2-card.

Receive:
That means that when the PC receive:
0x7D followed by 0x5E the real data is 0x7E.
Meaning that 0x7D, 0x5E should be replaced with 0x7E.

0x7D followed by 0x5D the real data is 0x7D
Meaning that 0x7D, 0x5D should be replaced with 0x7D.

Transmit:
When you transmit data to the module you must also think about this coding for all data contained between the start 0x7E and the stop 0x7E.

That means that when you want to transmit:
0x7D you must replace 0x7D with 0x7D,0x5D
0x7E you must replace 0x7E with 0x7D,0x5E

Possible commands to send

The following chapters will explain the different commands available in the script. Note that some commands are only applicable for some devices – This is noted for affected commands.

Note: The values are sent and received as hexadecimal.

Set TX interval

Sending this command will set the transmission time interval of the device, i.e., how often the device sends out wM-Bus data packets. Note that a shorter time interval will decrease the battery lifetime.

Transmit to device	
Start of packet	0x7E
Command byte	0x46
Length of configuration data	0x04
New transmission time interval **Note:** The bytes are sent "inverted" **Example**: 400s = 0x0190 = send as 9001	0x9001
End of packet	0x7E

Response from device	
Start of packet	0x7E
Command byte	0x47
Length of configuration data	0x04
Current set transmission time interval **Note:** The bytes are received "inverted" **Example:** 400s = 0x0190 = received as 9001	0x9001
CRC	0xXXXX
End of packet	0x7E

Get TX interval

Send this command to the device to check what the current transmission time interval of the device is.

Transmit to device	
Start of packet	0x7E
Command byte	0x47
Length of configuration data	0x02
End of packet	0x7E

Response from device	
Start of packet	0x7E
Command byte	0x47
Length of configuration data	0x04
Current transmission time interval **Note:** The bytes are received “inverted” **Example:** 300s = 0x012C = received as 2C01	0x2C01
CRC	0xXXXX
End of packet	0x7E

Set encryption option

Send this command to the device to enable or disable the encryption.

Transmit to device	
Start of packet	0x7E
Command byte	0x23
Length of configuration data	0x03
New encryption option	0x00 (Disable encryption) 0x01 (Enable encryption)
End of packet	0x7E

Response from device	
Start of packet	0x7E
Command byte	0x24
Length of configuration data	0x03
Currently set encryption option	0x00 (Encryption off) 0x01 (Encryption on)
CRC	0xXXXX
End of packet	0x7E

Set AES-key

Send this command if the AES-key should be changed.

Transmit to device	
Start of packet	0x7E
Command byte	0x48
Length of configuration data	0x12
New AES-key **Example:** 0x00112233445566778899AABBCCDDEEFF	0x00112233445566778899AABBCCDDEEFF
End of packet	0x7E

Response from device	
Start of packet	0x7E
Command byte	0x49
Length of configuration data	0x12
Currently set AES-key **Example:** 0x00112233445566778899AABBCCDDEEFF	0x00112233445566778899AABBCCDDEEFF
CRC	0xXXXX
End of packet	0x7E

Get autolock status

Send this command to get the autolock status.

Transmit to device	
Start of packet	0x7E
Command byte	0x45
Length of configuration data	0x02
End of packet	0x7E

Response from device	
Start of packet	0x7E
Command byte	0x45
Length of configuration data	0x03
Current autolock status	0x00 = Unlocked 0x01 = Locked 0x02 = Locked, wrong AES-key
CRC	0xXXXX
End of packet	0x7E

Restart autolock

Send this command to enable configuration for 8 hours.

Transmit to device	
Start of packet	0x7E
Command byte	0x44
Length of configuration data	0x05
First three bytes of the AES-key **Example:** AES-key: 0x00112233445566778899AABBCCDDEEFF First three bytes: 0x001122	0x001122
End of packet	0x7E

Response from device	
Start of packet	0x7E
Command byte	0x45
Length of configuration data	0x03
Current autolock status	0x00 = Unlocked (first start) 0x01 = Locked 0x02 = Locked, wrong AES-key
CRC	0xXXXX
End of packet	0x7E

Set pulse value

Send this command to change the pulse value on a channel.

Note: This command is only valid for our devices *LAN-WMBUS-G2-P* and *LAN-WMBUS-G2-OOP*.

Transmit to device	
Start of packet	0x7E
Command byte	0x50
Length of configuration data	0x0B
Channel index **Note**: This value sets which pulse channel to set	0x00 = Pulse channel 0 0x01 = Pulse channel 1
Pulse value **Note:** The bytes are sent "inverted" **Note**: Valid values are 0-999 999 999 999 **Example**: 1 250 000=0x1312D0=send as D01213	0xD012130000000000
End of packet	0x7E

Response from device	
Start of packet	0x7E
Command byte	0x51
Length of configuration data	0x0B
Channel index	0x00 = Pulse channel 0 0x01 = Pulse channel 1
Current pulse value for selected channel index **Note:** The bytes are received "inverted" **Example:** 1 250 000=0x1312D0=received as D01213	0xD012130000000000
CRC	0xXXXX
End of packet	0x7E

Set M-Bus mode

Send this command to change the output mode of the device. Supported modes are:

- S1 Packet Format A
- T1 Packet Format A
- C1 Packet Format A or B

Transmit to device	
Start of packet	0x7E
Command byte	0x15
Length of configuration data	0x05
M-Bus Input mode	0x01 = S1 0x02 = T1 and C1
M-Bus Output mode	0x01 = S1 0x02 = T1 0x03 = C1
M-Bus output frame format	0x01 = A 0x02 = B
End of packet	0x7E

Response from device	
Start of packet	0x7E
Command byte	0x14
Length of configuration data	0x05
M-Bus Input mode	0x01 = S1 0x02 = T1 and C1 0xFF = Device does not support input
M-Bus Output mode	0x01 = S1 0x02 = T1 0x03 = C1
M-Bus output frame format	0x01 = A 0x02 = B
CRC	0xXXXX
End of packet	0x7E

Get M-Bus mode

Transmit to device	
Start of packet	0x7E
Command byte	0x14
Length of configuration data	0x02
End of packet	0x7E

Response from device	
Start of packet	0x7E
Command byte	0x14
Length of configuration data	0x05
M-Bus Input mode	0x01 = S1 0x02 = T1 and C1 0xFF = Device does not support input
M-Bus Output mode	0x01 = S1 0x02 = T1 0x03 = C1
M-Bus output frame format	0x01 = A 0x02 = B
CRC	0xXXXX
End of packet	0x7E

Document history

Date	Changed by	Version	Details
2019-10-22	MaSt	1.0	Document created for C-TH.