

STW Technic, LP

STW CoDeSys v3 Library for IQ-S Combo Modules

V1.0.0.0

Jason Groeninger
8-23-2016

Welcome to the STW IQ-S Combo Module library!

The purpose of this library is to provide easy handling of CANbus communication between an STW ESX-3XL/-3XM and an STW IQ-S Combo Module.

The user will instance and execute a series of structures, functions, and function blocks for each IQ-S Combo Module in the CAN network.

Here is an overview.

----- User Data -----

All user data for each IQ-S Combo Module will be written to/from the CANbus via a series of structures, which can be found within the **Structures** directory. The user should declare an instance (most likely globally scoped) of each structure as needed to store IQ-S data.

--- Received data (from the IQ-S Combo Module) ---

T_IQS_C_STATUS - All received data is stored into this single structure, which is composed of the following sub-structures.

t_Digital_Inputs : T_IQS_C_PGN_DIGITAL_STATUS

- Digital state of all digital inputs

t_Analog_Inputs_1 : T_IQS_C_PGN_ANALOG_STATUS_1

- Analog values for inputs 5B and 6B

t_Analog_Inputs_2 : T_IQS_C_PGN_ANALOG_STATUS_2

- Analog values for inputs 7B and 8B

t_Frequency_Inputs : T_IQS_C_PGN_FREQUENCY_STATUS

- Hertz and Duty Cycle values for frequency inputs
- Count and Set Point values for high speed counter inputs

t_Software_Node_Status : T_IQS_C_PGN_STATUS_1

- Software and Hardware Version information, Fault Codes, User ID, and Status information

t_Config_Output_Status : T_IQS_C_PGN_STATUS_2

- Output configuration jumper states, Output A/B channel statuses, Input power port statuses, Power Buss statuses, Configuration Save counter

t_Controller_Info : T_IQS_C_PGN_STATUS_3

- Hardware configuration, Battery Voltage, Module Temperature

t_Output_Feedback_1 : T_IQS_C_PGN_STATUS_4

- Current feedback for outputs 1A, 1B, 2A, 2B

t_Output_Feedback_2 : T_IQS_C_PGN_STATUS_5

- Current feedback for outputs 3A, 3B, 4A, 4B

--- Transmitted data (to the IQ-S Combo Module) ---

T_IQS_C_CONFIG - All configuration data for the module is stored in this structure, which is composed of the following sub-structures.

t_Output_Cfg_1 : T_IQS_C_PGN_OUTPUT_CONFIG_1

- General module configuration, global output and input configuration

t_Output_Cfg_2 : T_IQS_C_PGN_OUTPUT_CONFIG_2

- Individual output and input configuration for I/O ports 1-7

t_Output_Cfg_3 : T_IQS_C_PGN_OUTPUT_CONFIG_3

- Individual input configuration for inputs 8A and 8B
- PI control configuration values for outputs 1A, 1B, and 2A

t_Output_Cfg_4 : T_IQS_C_PGN_OUTPUT_CONFIG_4

- PI control configuration values for outputs 2B, 3A, and 3B

t_Output_Cfg_5 : T_IQS_C_PGN_OUTPUT_CONFIG_5

- PI control configuration values for outputs 4, and 4B

t_Counter_Cfg : T_IQS_C_PGN_COUNTER_CONFIG

- High speed counter configuration for inputs 7A and 8A

T_IQS_C_OUTPUT_CONTROL – All control data for Digital and PWM output control is stored in this structure, which is composed of the following sub-structures.

t_Digital_Control : T_IQS_C_PGN_CONTROL_1

- All digital output controls
- All Input power port controls

t_PWM_Control_1 : T_IQS_C_PGN_CONTROL_2

- PWM control for outputs 1A, 1B, 2A, 2B

t_PWM_Control_2 : T_IQS_C_PGN_CONTROL_3

- PWM control for outputs 3A, 3B, 4A, 4B

----- Initialization -----

Functions can be found within the **Initialization_Functions** directory.

F_IQS_C_Build_Defaults

- Inputs -

- **opt_IQS_C_Config** - Pointer to user's instance of **T_IQS_C_CONFIG** for this module

- Return -

- **C_NO_ERR** - Function executed correctly
- **C_NOACT** - Invalid pointer, defaults not loaded

- Usage -

This function is optional, but it is recommended to run this once during an initialization routine. The function populates the user's instance of **T_IQS_C_CONFIG** with a baseline configuration. This includes the following:

- Enables cyclic transmission of status messages **T_IQS_C_PGN_STATUS_2**, **T_IQS_C_PGN_STATUS_3**, **T_IQS_C_PGN_ANALOG_STATUS_1**, **T_IQS_C_PGN_ANALOG_STATUS_2**, **T_IQS_C_PGN_STATUS_4**, and **T_IQS_C_PGN_STATUS_5**
- Disables global output and input configuration overrides (MODE1 & 2)
- Configures all outputs as type Percent (PWM)
- Configures all inputs as type Digital
- Configures PI control parameters for all outputs to the default values of Kp=100 and Ki=100
- Sets all High Speed Counter input configurations to OFF

It is important to note that parameters are not sent individually to the Module via CAN, they are always sent in groups, so it is important to have all parameters defined before sending any of them.

F_IQS_C_Init_CAN_Objects

- Inputs -

- **ou16_Channel** - CANbus channel
- **ou8_Module_SA** - IQ-S module source address
- **opu32_RX_Handle** - Pointer to user's instance of an RX CAN object Handle (**UDINT**)
- **opu32_TX_Handle** - Pointer to user's instance of an TX CAN object Handle (**UDINT**)

- Return -

- **C_NO_ERR** - Function executed correctly
- **C_RANGE** - **ou16_Channel** or **ou8_Module_SA** range error, objects not initialized
- **C_NOACT** - Invalid **opu32_RX_Handle** or **opu32_TX_Handle** pointer, objects not initialized
- **C_DEFAULT** - CAN Object or Filter initialization error

- Usage -

This function is optional, but it is recommended to run this once during an initialization routine. Alternatively, the user can manually create the necessary CAN objects via the project's Devices window.

The function allocates and configures a Transmit CAN object and a Receive CAN object on CANbus **ou16_Channel** for communication with IQ-S Module **ou8_Module_SA**. The function then configures the necessary receive filter for managing IQ-S messages. The function returns Handles for the created RX and TX objects, which the user will later pass to the transmit and receive function blocks.

----- CAN Message Handling -----

Function Blocks can be found within the **CAN_Handling** directory

--- Receiving Data ---

FB_IQS_C_Receive

- Inputs -

- **ou32_RX_Handle** - Handle of the CAN RX object receiving the messages, generated by **F_IQS_C_Init_CAN_Objects** or by the user

- Outputs -

- **ot_IQS_C_Status** - All status message data, stored as **T_IQS_C_STATUS**
- **os16_Status** - Function and Function Block execution status:
 - **C_NO_ERR** - Normal execution
 - **C_DEFAULT** - Peer-to-peer message rejected with incorrect destination address
 - **C_RANGE** - CAN object handle **ou32_RX_Handle** not valid
 - **C_NOACT** - Incorrect decoding function selected
 - **C_CONFIG** - CAN object at **ou32_RX_Handle** is not type **X_CAN_OBJ_TYPE_RX**

- Usage -

The user should call this function block cyclically. The FB will process all data within the RX object buffer and populate the appropriate sub-structures within **ot_IQS_C_Status** as the data is decoded. The function block will also manage message timeouts and will report this data within each sub-structure.

--- Transmitting Data ---

-- Configuration --

FB_IQS_C_Output_Config_1_Transmit

- Inputs -

- **ou8_Module_SA** - IQS module source address, range 16#E0..16#EF. Global Constants **IQS_C_MODULE_SA_00** - **IQS_C_MODULE_SA_15** are provided for convenience.
- **ou8_Own_SA** - STW Controller source address, range 16#D1. Global Constant **IQS_C_MASTER_SA** is provided for convenience.
- **ot_Output_Config** - Configuration data, stored as **T_IQS_C_PGN_OUTPUT_CONFIG_1**
- **ou32_TX_Handle** - Handle of the CAN TX object transmitting the messages, generated by **F_IQS_C_Init_CAN_Objects** or by the user

- Outputs -

- **os16_Status** - Function Block execution status:
 - **C_NO_ERR** - Normal execution
 - **C_WARN** - **ou8_Own_SA** range error, message sent but module may not react
 - **C_RANGE** - **ou8_Module_SA** range error, message not sent
 - **C_DEFAULT** - Encoding or transmission error

- Usage -

The user should call this function block as needed to transmit Output Configuration 1 variables, as seen within the **T_IQS_C_CONFIG** structure as **t_Output_Cfg_1** :

T_IQS_C_PGN_OUTPUT_CONFIG_1. The CAN message will be transmitted once per execution of the function block.

- **u8_Ctrl_Mode_Reset** - Enables Controller Mode Output Reset
- **u8_Enable_Status2_Msg** - Enables the constant transmission of message **T_IQS_C_PGN_STATUS_2**, Configuration and Output Status
- **u8_Enable_Status3_Msg** - Enables the constant transmission of message **T_IQS_C_PGN_STATUS_3**, Controller Information
- **u8_Enable_Amp_Msg** - Enables the constant transmission of messages:
 - **T_IQS_C_PGN_STATUS_4**, Output Amperage Feedback OUT 1A-2B
 - **T_IQS_C_PGN_STATUS_5**, Output Amperage Feedback OUT 3A-4B
 - **T_IQS_C_PGN_ANALOG_STATUS_1**, Analog input values IN 5B, 6B
 - **T_IQS_C_PGN_ANALOG_STATUS_2**, Analog input values IN 7B, 8B
- **u8_Enable_24VDC** - Enables the low and over voltage fault limits for 24 vdc system
- **u8_Save_Config** - Saves the configuration to the module (otherwise changes only valid until a power cycle occurs), set to 1 to write configuration to module
- **u8_Analog_Raw_Value** – Internal use only
- **u16_FREQ1_Hz** - Sets the configuration of the frequency for all channels (30-1140 Hz)
- **u8_MODE1** - Sets the configuration of ALL the outputs, overrides Command 53 and 54. (0=NOT USED, 1=ON/OFF, 2=Data 0-4000, 3= Percent 0-100.0%(0-1000))
- **u8_MODE2** - Sets the configuration of the inputs. (0=disabled, 1=Digital Positive, 2=Digital Ground) No analog
- **u8_ID1** - User defined byte for configuration ID, this will be transmitted in the STAT message.

FB_IQS_C_Output_Config_2_Transmit

- Inputs -

- **ou8_Module_SA** - IQS module source address, range 16#E0..16#EF. Global Constants **IQS_C_MODULE_SA_00** - **IQS_C_MODULE_SA_15** are provided for convenience.
- **ou8_Own_SA** - STW Controller source address, range 16#D1. Global Constant **IQS_C_MASTER_SA** is provided for convenience.
- **ot_Output_Config** - Configuration data, stored as **T_IQS_C_PGN_OUTPUT_CONFIG_2**
- **ou32_TX_Handle** - Handle of the CAN TX object transmitting the messages, generated by **F_IQS_C_Init_CAN_Objects** or by the user

- Outputs -

- **os16_Status** - Function Block execution status:
 - **C_NO_ERR** - Normal execution
 - **C_WARN** - **ou8_Own_SA** range error, message sent but module may not react
 - **C_RANGE** - **ou8_Module_SA** range error, message not sent
 - **C_DEFAULT** - Encoding or transmission error

- Usage -

The user should call this function block as needed to transmit Output Configuration 2 variables, as seen within the **T_IQS_C_CONFIG** structure as **t_Output_Cfg_2** :

T_IQS_C_PGN_OUTPUT_CONFIG_2. The CAN message will be transmitted once per execution of the function block.

- **u8_OutMode_1A .. u8_OutMode_4B** - Mode 0x0=disabled, 0x1=On/Off, 0x2=Data(0-4000), 0x3=Percent(0-1000 = 0-100.0%), 0x4=Amps(0-4000 = 0-4.000 amps)
- **u8_InMode_5A** - Mode 0x0=disabled, 0x1=Digital Positive, 0x2=Digital Ground
- **u8_InMode_5B** - Mode 0x0=disabled, 0x1=Digital Positive, 0x2=Digital Ground, 0x3= 4-20ma (4000-20000), 0x4= 0-5V DC (0-5000, 0x5= 0-10V DC (0-10000), 0x6= 0-32V DC(0-32000), 0x9= Ratiometric (0-100.0% of source V DC)
- **u8_InMode_6A** - Mode 0x0=disabled, 0x1=Digital Positive, 0x2=Digital Ground
- **u8_InMode_6B** - Mode 0x0=disabled, 0x1=Digital Positive, 0x2=Digital Ground, 0x3= 4-20ma (4000-20000), 0x4= 0-5V DC (0-5000, 0x5= 0-10V DC (0-10000), 0x6= 0-32V DC(0-32000), 0x9= Ratiometric (0-100.0% of source V DC)
- **u8_InMode_7A** - Mode 0x0=disabled, 0x1=Digital Positive, 0x2=Digital Ground, 0x7=Digital Positive Frequency, 0x8= High Speed Counter, 0xA=Encoder
- **u8_InMode_7B** - Mode 0x0=disabled, 0x1=Digital Positive, 0x2=Digital Ground, 0x3= 4-20ma (4000-20000), 0x4= 0-5V DC (0-5000, 0x5= 0-10V DC (0-10000), 0x6= 0-32V DC(0-32000), 0x9= Ratiometric (0-100.0% of source V DC)

FB_IQS_C_Output_Config_3_Transmit

- Inputs -

- **ou8_Module_SA** - IQS module source address, range 16#E0..16#EF. Global Constants **IQS_C_MODULE_SA_00** - **IQS_C_MODULE_SA_15** are provided for convenience.
- **ou8_Own_SA** - STW Controller source address, range 16#D1. Global Constant **IQS_C_MASTER_SA** is provided for convenience.
- **ot_Output_Config** - Configuration data, stored as **T_IQS_C_PGN_OUTPUT_CONFIG_3**
- **ou32_TX_Handle** - Handle of the CAN TX object transmitting the messages, generated by **F_IQS_C_Init_CAN_Objects** or by the user

- Outputs -

- **os16_Status** - Function Block execution status:
 - **C_NO_ERR** - Normal execution
 - **C_WARN** - **ou8_Own_SA** range error, message sent but module may not react
 - **C_RANGE** - **ou8_Module_SA** range error, message not sent
 - **C_DEFAULT** - Encoding or transmission error

- Usage -

The user should call this function block as needed to transmit Output Configuration 3 variables, as seen within the **T_IQS_C_CONFIG** structure as **t_Output_Cfg_3** :

T_IQS_C_PGN_OUTPUT_CONFIG_3. The CAN message will be transmitted once per execution of the function block.

- **u8_InMode_8A** - Mode 0x0=disabled, 0x1=Digital Positive, 0x2=Digital Ground, 0x7=Digital Positive Frequency, 0x8= High Speed Counter (mode not used when 7A is set up as encoder)
- **u8_InMode_8B** - Mode 0x0=disabled, 0x1=Digital Positive, 0x2=Digital Ground, 0x3= 4-20ma (4000-20000), 0x4= 0-5V DC (0-5000, 0x5= 0-10V DC (0-10000), 0x6= 0-32V DC(0-32000), 0x9= Ratiometric (0-100.0% of source V DC)
- **u8_Port1A_Kp** - Output 1A proportional set point (0-250 = 0-2.50, >250=0) default 100
- **u8_Port1A_Ki** - Output 1A integral set point (0-250 = 0-2.50, >250=0) default 100
- **u8_Port1B_Kp** - Output 1B proportional set point (0-250 = 0-2.50, >250=0) default 100
- **u8_Port1B_Ki** - Output 1B integral set point (0-250 = 0-2.50, >250=0) default 100
- **u8_Port2A_Kp** - Output 2A proportional set point (0-250 = 0-2.50, >250=0) default 100
- **u8_Port2A_Ki** - Output 2A integral set point (0-250 = 0-2.50, >250=0) default 100

FB_IQS_C_Output_Config_4_Transmit

- Inputs -

- **ou8_Module_SA** - IQS module source address, range 16#E0..16#EF. Global Constants **IQS_C_MODULE_SA_00** - **IQS_C_MODULE_SA_15** are provided for convenience.
- **ou8_Own_SA** - STW Controller source address, range 16#D1. Global Constant **IQS_C_MASTER_SA** is provided for convenience.
- **ot_Output_Config** - Configuration data, stored as **T_IQS_C_PGN_OUTPUT_CONFIG_4**
- **ou32_TX_Handle** - Handle of the CAN TX object transmitting the messages, generated by **F_IQS_C_Init_CAN_Objects** or by the user

- Outputs -

- **os16_Status** - Function Block execution status:
 - **C_NO_ERR** - Normal execution
 - **C_WARN** - **ou8_Own_SA** range error, message sent but module may not react
 - **C_RANGE** - **ou8_Module_SA** range error, message not sent
 - **C_DEFAULT** - Encoding or transmission error

- Usage -

The user should call this function block as needed to transmit Output Configuration 4 variables, as seen within the **T_IQS_C_CONFIG** structure as **t_Output_Cfg_4** :

T_IQS_C_PGN_OUTPUT_CONFIG_4. The CAN message will be transmitted once per execution of the function block.

- **u8_Port2B_Kp** - Output 2B proportional set point (0-250 = 0-2.50, >250=0) default 100
- **u8_Port2B_Ki** - Output 2B integral set point (0-250 = 0-2.50, >250=0) default 100
- **u8_Port3A_Kp** - Output 3A proportional set point (0-250 = 0-2.50, >250=0) default 100
- **u8_Port3A_Ki** - Output 3A integral set point (0-250 = 0-2.50, >250=0) default 100
- **u8_Port3B_Kp** - Output 3B proportional set point (0-250 = 0-2.50, >250=0) default 100
- **u8_Port3B_Ki** - Output 3B integral set point (0-250 = 0-2.50, >250=0) default 100

FB_IQS_C_Output_Config_5_Transmit

- Inputs -

- **ou8_Module_SA** - IQS module source address, range 16#E0..16#EF. Global Constants **IQS_C_MODULE_SA_00** - **IQS_C_MODULE_SA_15** are provided for convenience.
- **ou8_Own_SA** - STW Controller source address, range 16#D1. Global Constant **IQS_C_MASTER_SA** is provided for convenience.
- **ot_Output_Config** - Configuration data, stored as **T_IQS_C_PGN_OUTPUT_CONFIG_5**
- **ou32_TX_Handle** - Handle of the CAN TX object transmitting the messages, generated by **F_IQS_C_Init_CAN_Objects** or by the user

- Outputs -

- **os16_Status** - Function Block execution status:
 - **C_NO_ERR** - Normal execution
 - **C_WARN** - **ou8_Own_SA** range error, message sent but module may not react
 - **C_RANGE** - **ou8_Module_SA** range error, message not sent
 - **C_DEFAULT** - Encoding or transmission error

- Usage -

The user should call this function block as needed to transmit Output Configuration 5 variables, as seen within the **T_IQS_C_CONFIG** structure as **t_Output_Cfg_5** :

T_IQS_C_PGN_OUTPUT_CONFIG_5. The CAN message will be transmitted once per execution of the function block.

- **u8_Port4A_Kp** - Output 4A proportional set point (0-250 = 0-2.50, >250=0) default 100
- **u8_Port4A_Ki** - Output 4A integral set point (0-250 = 0-2.50, >250=0) default 100
- **u8_Port4B_Kp** - Output 4B proportional set point (0-250 = 0-2.50, >250=0) default 100
- **u8_Port4B_Ki** - Output 4B integral set point (0-250 = 0-2.50, >250=0) default 100

FB_IQS_C_Counter_Config_Transmit

- Inputs -

- **ou8_Module_SA** - IQS module source address, range 16#E0..16#EF. Global Constants **IQS_C_MODULE_SA_00** - **IQS_C_MODULE_SA_15** are provided for convenience.
- **ou8_Own_SA** - STW Controller source address, range 16#D1. Global Constant **IQS_C_MASTER_SA** is provided for convenience.
- **ot_Counter_Config** - Configuration data, stored as **T_IQS_C_PGN_COUNTER_CONFIG**
- **ou32_TX_Handle** - Handle of the CAN TX object transmitting the messages, generated by **F_IQS_C_Init_CAN_Objects** or by the user

- Outputs -

- **os16_Status** - Function Block execution status:
 - **C_NO_ERR** - Normal execution
 - **C_WARN** - **ou8_Own_SA** range error, message sent but module may not react
 - **C_RANGE** - **ou8_Module_SA** range error, message not sent
 - **C_DEFAULT** - Encoding or transmission error

- Usage -

The user should call this function block as needed to transmit Counter Configuration variables, as seen within the **T_IQS_C_CONFIG** structure as **t_Counter_Cfg** :

T_IQS_C_PGN_COUNTER_CONFIG. The CAN message will be transmitted once per execution of the function block.

- **u8_Counter7A_OnOff** - Enable Counter 7A, 00 = Off, 01 = On (used in encoder mode)
- **u8_Counter7A_Reset** - Reset Counter 7A, 00 = Off, 01 = On (used in encoder mode)
- **u8_Counter7A_Rollover_Enabled** - Enable Counter 7A Rollover, count continues after set point is achieved, 00 = Off, 01 = On (not used in encoder mode)
- **u8_Counter7A_Enable_Out3A** - Enable Output 3A when Counter 7A is enabled and set point hasn't been reached, 00 = Off, 01 = On (not used in encoder mode)
- **u8_Counter8A_OnOff** - Enable Counter 8A, 00 = Off, 01 = On (not used in encoder mode)
- **u8_Counter8A_Reset** - Reset Counter 8A, 00 = Off, 01 = On (not used in encoder mode)
- **u8_Counter8A_Rollover_Enabled** - Enable Counter 8A Rollover, count continues after set point is achieved, 00 = Off, 01 = On (not used in encoder mode)
- **u8_Counter8A_Enable_Out4A** - Enable Output 4A when Counter 8A is enabled and set point hasn't been reached, 00 = Off, 01 = On (not used in encoder mode)
- **u16_Counter7A_Setpoint** - Set point for totalized count on Counter 7A
- **u16_Counter8A_Setpoint** - Set point for totalized count on Counter 8A

-- Control --

FB_IQS_C_Control_1_Transmit

- Inputs -

- **oq_Enable** - Enable transmission of control message
- **ou8_Module_SA** - IQS module source address, range 16#E0..16#EF. Global Constants **IQS_C_MODULE_SA_00** - **IQS_C_MODULE_SA_15** are provided for convenience.
- **ou8_Own_SA** - STW Controller source address, range 16#D1. Global Constant **IQS_C_MASTER_SA** is provided for convenience.
- **ot_Control** - Output control data, stored as **T_IQS_C_PGN_CONTROL_1**
- **ou32_TX_Handle** - Handle of the CAN TX object transmitting the messages, generated by **F_IQS_C_Init_CAN_Objects** or by the user

- Outputs -

- **os16_Status** - Function Block execution status:
 - **C_NO_ERR** - Normal execution
 - **C_WARN** - **ou8_Own_SA** range error, message sent but module may not react
 - **C_RANGE** - **ou8_Module_SA** range error, message not sent
 - **C_DEFAULT** - Encoding or transmission error

- Usage -

The user should call this function block cyclically. The function block maintains an internal timer while the **oq_Enable** input is **TRUE**. When the timer expires, the message is sent and the timer resets and begins again.

The message contents are defined within the **T_IQS_C_OUTPUT_CONTROL** structure as

t_Digital_Control : **T_IQS_C_PGN_CONTROL_1**.

- **u8_OnOff_Output_1A** .. **u8_OnOff_Output_4B** - Turns the output on when in "On/Off" Mode, (not used when using any other mode)
- **u8_OnOff_Input_Power_Port5** .. **u8_OnOff_Input_Power_Port8** - Enable for Port # Input Power, (00 = off), (01 = on)

FB_IQS_C_Control_2_Transmit

- Inputs -

- **oq_Enable** - Enable transmission of control message
- **ou8_Module_SA** - IQS module source address, range 16#E0..16#EF. Global Constants **IQS_C_MODULE_SA_00** - **IQS_C_MODULE_SA_15** are provided for convenience.
- **ou8_Own_SA** - STW Controller source address, range 16#D1. Global Constant **IQS_C_MASTER_SA** is provided for convenience.
- **ot_Control** - Output control data, stored as **T_IQS_C_PGN_CONTROL_2**
- **ou32_TX_Handle** - Handle of the CAN TX object transmitting the messages, generated by **F_IQS_C_Init_CAN_Objects** or by the user

- Outputs -

- **os16_Status** - Function Block execution status:
 - **C_NO_ERR** - Normal execution
 - **C_WARN** - **ou8_Own_SA** range error, message sent but module may not react
 - **C_RANGE** - **ou8_Module_SA** range error, message not sent
 - **C_DEFAULT** - Encoding or transmission error

- Usage -

The user should call this function block cyclically. The function block maintains an internal timer while the **oq_Enable** input is **TRUE**. When the timer expires, the message is sent and the timer resets and begins again.

The message contents are defined within the **T_IQS_C_OUTPUT_CONTROL** structure as **t_PWM_Control_1** : **T_IQS_C_PGN_CONTROL_2**.

- **u16_PWM_Output_1A** - PWM Output, range dependent on configuration
- **u16_PWM_Output_1B** - PWM Output, range dependent on configuration
- **u16_PWM_Output_2A** - PWM Output, range dependent on configuration
- **u16_PWM_Output_2B** - PWM Output, range dependent on configuration

FB_IQS_C_Control_3_Transmit

- Inputs -

- **oq_Enable** - Enable transmission of control message
- **ou8_Module_SA** - IQS module source address, range 16#E0..16#EF. Global Constants **IQS_C_MODULE_SA_00** - **IQS_C_MODULE_SA_15** are provided for convenience.
- **ou8_Own_SA** - STW Controller source address, range 16#D1. Global Constant **IQS_C_MASTER_SA** is provided for convenience.
- **ot_Control** - Output control data, stored as **T_IQS_C_PGN_CONTROL_3**
- **ou32_TX_Handle** - Handle of the CAN TX object transmitting the messages, generated by **F_IQS_C_Init_CAN_Objects** or by the user

- Outputs -

- **os16_Status** - Function Block execution status:
 - **C_NO_ERR** - Normal execution
 - **C_WARN** - **ou8_Own_SA** range error, message sent but module may not react
 - **C_RANGE** - **ou8_Module_SA** range error, message not sent
 - **C_DEFAULT** - Encoding or transmission error

- Usage -

The user should call this function block cyclically. The function block maintains an internal timer while the **oq_Enable** input is **TRUE**. When the timer expires, the message is sent and the timer resets and begins again.

The message contents are defined within the **T_IQS_C_OUTPUT_CONTROL** structure as **t_PWM_Control_2** : **T_IQS_C_PGN_CONTROL_3**.

- **u16_PWM_Output_3A** - PWM Output, range dependent on configuration
- **u16_PWM_Output_3B** - PWM Output, range dependent on configuration
- **u16_PWM_Output_4A** - PWM Output, range dependent on configuration
- **u16_PWM_Output_4B** - PWM Output, range dependent on configuration

--- Other Functions ---

The directories **RX_Functions** and **TX_Functions** contain several functions which are used by the **CAN_Handling** function blocks. It is not necessary for the user to directly access these functions.