

MQTTClient_Interface

MQTTClient_Interface		MQTT	
MQTTClient_Interface1			
SigCLib	InterfaceState		
0	MCIS_Connected		
Stdlib	ClientID		
0	"03934756"		
OpenSSL	ClientActive		
0	1		
cUseXML	CleanSession		
1	0		
cDataMapPubTimeout	SessionPresent		
10000	1		
cLogConfig	XMLFilename		
2#00000000000000000000...	"C:/MQTT_Client.xml"		
cDataMapSubTimeout	Reset		
10000	0		
cDataMapResetTimeout	ComDiagnosis		
10000	0		
cConnectTimeout			
20000			
cAutoResetTime			
5000			
cEventLog			
0			
cActivateTrace			
1			

Diese Klasse stellt Schnittstellen für den MQTT Client bereit. Der Client wird über eine „.xml“ Datei, welche auf der Steuerung liegt, oder über die Klasse MQTTClient_Interface_Config(Base) konfiguriert. Ein Muster solch einer „.xml“ Datei befindet sich am Ende des Dokuments im Anhang.

Schnittstellen

Clients

SigCLib	Objektkanal zur Klasse SigCLib. Muss nicht verbunden werden.	
	Datentyp	DINT
StdLib	Objektkanal zur Klasse _StdLib. Muss nicht verbunden werden.	
	Datentyp	DINT
OpenSSL	Objektkanal zur OpenSSL Klasse. Muss nicht verbunden werden.	
	Datentyp	DINT
cUseXML	Der Benutzer kann angeben, ob eine XML-Datei auf diesem Client gelesen werden soll: 1 – XML-Datei verwenden 0 – XML-Datei nicht verwenden	
	Datentyp	DINT
cLogConfig	Dieser Client zeigt die aktuelle Logging Konfiguration. Die Logging Information kann gesetzt werden durch: 1. Initialisierung an diesem Client. 2. Konfiguriert durch die .xml Datei	
	Datentyp	MQTTClient_Interface::t_bd_LogConfig
cDataMapPubTimeout	Timeout für die Publish Prozedur der Schnittstelle. (ms)	
	Datentyp	UDINT
cDataMapSubTimeout	Timeout für die Subscribe Prozedur der Schnittstelle. (ms)	
	Datentyp	UDINT
cConnectTimeout	Kommunikations- Timeout. (ms)	
	Datentyp	UDINT

Server

State	Dieser Server kann zum Aufrufen der globalen Methoden der Klasse benutzt werden. Zeigt den aktuellen Status der MQTT Client Schnittstelle.		
	Einheit	-	Datentyp
	Wertebereich	0-6	Write Protected
	Defaultwert	-	Retentive
ClientID	Hier muss die ID für den MQTT Client definiert werden. Diese ID muss im System einzigartig sein. Wird aus dem .xml File gesetzt.		
	Einheit	-	Datentyp
	Wertebereich	-	Write Protected
	Defaultwert	-	Retentive
ClientActive	Über diesen Server kann der MQTT Client aktiviert und deaktiviert werden. 0...deaktiviert 1...aktiviert.		
	Einheit	-	Datentyp
	Wertebereich	0-1	Write Protected
	Defaultwert	-	Retentive
CleanSession	Hier kann eingestellt werden ob die Session vom MQTT Client am Server gespeichert werden soll. (z.B. Subscribed Topics) 0...Der MQTT Server soll die aktuelle Session vom Client speichern, wenn die Verbindung verloren geht. 1...Der MQTT Server wird die vorherige Session verwerfen, wenn eine neue Verbindung aufgebaut wird oder die Verbindung verloren geht. Diese Einstellung wird nur während des Verbindungsaufbaues zum Server angewendet.		
	Einheit	-	Datentyp
	Wertebereich	0-1	Write Protected
	Defaultwert	-	Retentive
SessionPresent	Zeigt an ob eine aktive Session am MQTT Server gespeichert war.		

	0...Es gibt keine gespeicherte Session dieses Clients		
	1...Von diesem Client ist eine Session am Server vorhanden.		
	Einheit	-	Datentyp
	Wertebereich	0-1	Write Protected
XMLFilename	Defaultwert	-	Retentive
	FALSE		
	Hier muss der Pfad der „.xml“ Datei, zur Konfiguration des MQTT Clients, angegeben werden. (Inklusive Dateiname und Endung.		
Reset	Einheit	-	Datentyp
	Wertebereich	-	Write Protected
	Defaultwert	„C:/MQTT_Client.xml“	Retentive
	FALSE		
ComDiagnosis	Über diesen Server kann die Klasse aus dem Fehlerzustand zurückgesetzt werden. Der MQTT Client wird dabei nochmal neu initialisiert.		
	Muss auf 1 gesetzt werden.		
	Einheit	-	Datentyp
	Wertebereich	-	Write Protected
	Defaultwert	-	Retentive
	FALSE		
	Server, um den Kommandokanal des Objekts ComDiagnosis_MQTT zu verbinden. Dieser Server enthält die erforderlichen NewInst()-Methoden für das Funktionieren des Objekts.		
	Einheit	-	Datentyp
	Wertebereich	-	Write Protected
	Defaultwert	-	Retentive
	FALSE		

Globale Methoden

Init	Initialisierung der Klasse. Allokieren von Speicher.	
CyWork	Aufruf der internen Routine.	
MQTT_ReadXML	Kann aufgerufen werden um die Konfiguration der „.xml“ Datei neu zu lesen. Der MQTT Client wird dabei neu initialisiert.	
	◀ bReadStarted (BOOL)	True...Lesen der Konfigurationsdatei wurde angestoßen False...Lesen der Konfigurationsdatei konnte nicht angestoßen werden.
MQTT_ReadLogConfig	Mit dieser Methode kann die Logging Konfiguration vom Client cLogConfig neu gelesen werden.	
	◀ bUpdated (BOOL)	True...Logging Konfiguration wurde upgedated False...Logging Konfiguration konnte nicht upgedated werden.
MQTT_Subscribe	Mit dieser Methode kann man sich zu einem bestimmten Topic anmelden.	
	▶ pTopic (^UINT)	Zeiger auf den Topic Namen
	▶ iQoS (INT)	Hier kann die „Quality of Service“ für die Datenübertragung eingestellt werden. Diese bezieht sich immer zwischen Client und Server (Broker). 0...Das Senden der Daten erfolgt höchstens einmal. Der Empfänger bestätigt nicht den Empfang der Daten. Bietet dieselbe Garantie wie das darunter liegende TCP Protokoll. 1...Das Senden der Daten erfolgt mindestens einmal. Der Sender wartet auf eine Bestätigung des Empfängers. Erfolgt diese nicht in einer gewissen Zeit, so werden die Daten nochmal gesendet. Es besteht die Möglichkeit das die gleichen Daten öfter gesendet und empfangen werden. 2...Das Senden der Daten erfolgt exakt einmal. Hier ist sichergestellt, dass der Empfänger die Nachricht genau einmal bekommt. Hinweis: Umso höher der „Quality of Service“ eingestellt ist, umso länger dauert auch die Abarbeitung der Sende- und Empfangsroutine.

	► pThis (^VOID)	This Zeiger vom aufrufenden Objekt.
	► pCallback (^VOID)	Zeiger auf die Callback Methode vom aufzurufen- den Objekt. Über den Callback werden Informationen zum angemeldeten Topic übergeben. Aufbau des Callbacks siehe im Anhang.
	► pMid (^INT)	Zeiger auf eine Variable, wo die Nachrichten ID gespeichert werden soll.
	► pApplMsgID (^UDINT)	Zeiger auf eine Variable, wo die interne Applikati- ons- Nachrichten ID gespeichert werden soll. Diese ID kann für die MQTT_Unsubscribe Funktion verwendet werden.
	◄ retCode (INT)	0...erfolgreich 1...kein Speicher mehr für eine weitere Anmeldung vorhanden. 3...ein Fehler ist aufgetreten
MQTT_Unsubscribe	Mit dieser Methode kann man sich von einem bestimmten Topic abmelden.	
	► pApplMsgID (^UDINT)	Zeiger auf eine Variable, wo die interne Applikati- ons- Nachrichten ID gespeichert wurde. (MQTT_Subscribe) Wenn der Parameter pTopic gesetzt ist, ist diese ID nicht notwendig.
	► pTopic (^UINT)	Zeiger auf den Topic Namen Wenn der Parameter pApplMsgID gesetzt ist, ist der Name nicht notwendig.
	► pMid (^INT)	Zeiger auf eine Variable, wo die Nachrichten ID gespeichert werden soll.
	► pThis (^VOID)	This Zeiger vom aufrufenden Objekt.
	► pCallback (^VOID)	Zeiger auf die Callback Methode vom aufzurufen- den Objekt. Über den Callback werden Informationen zur Ab- meldung übergeben. Aufbau des Callbacks siehe im Anhang.
	◄ retCode (INT)	0...erfolgreich 1...kein Speicher mehr vorhanden im Nachrichten Buffer. 3...Ungültige Übergabeparameter 4...Keine Verbindung zum MQTT Server

MQTT_Publish	Mit dieser Methode können Daten zu einem bestimmten Topic gesendet werden.	
	► pTopic (^UINT)	Zeiger auf den Topic Namen
	► iQoS (INT)	<p>Hier kann die „Quality of Service“ für die Datenübertragung eingestellt werden. Diese bezieht sich immer zwischen Client und Server (Broker).</p> <p>0...Das Senden der Daten erfolgt höchstens einmal. Der Empfänger bestätigt nicht den Empfang der Daten. Bietet dieselbe Garantie wie das darunter liegende TCP Protokoll.</p> <p>1...Das Senden der Daten erfolgt mindestens einmal. Der Sender wartet auf eine Bestätigung des Empfängers. Erfolgt diese nicht in einer gewissen Zeit, so werden die Daten nochmal gesendet. Es besteht die Möglichkeit das die gleichen Daten öfter gesendet und empfangen werden.</p> <p>2...Das Senden der Daten erfolgt exakt einmal. Hier ist sichergestellt, dass der Empfänger die Nachricht genau einmal bekommt.</p> <p>Hinweis: Umso höher der „Quality of Service“ eingestellt ist, umso länger dauert auch die Abarbeitung der Sende- und Empfangsroutine.</p>
	► pPayload (^VOID)	Zeiger auf die Daten, die gesendet werden sollen.
	► udPayloadLen (UDINT)	Länge der Daten in Byte.
	► bRetain (BOOL)	<p>Hier wird definiert ob die Daten, welche zum Server zu einem bestimmten Topic gesendet werden (Publish), gespeichert werden sollen. Meldet sich ein neuer Client an dieses Topic an, so werden ihm sofort die letzten gespeicherten Daten gesendet. Ansonsten bekäme der neue Client nur Daten, sobald wieder jemand etwas auf dieses Topic sendet.</p> <p>False...Daten werden nicht gespeichert True...Daten werden gespeichert</p>
	► pThis (^VOID)	This Zeiger vom aufrufenden Objekt.
	► pCallback (^VOID)	<p>Zeiger auf die Callback Methode vom aufzurufenen Objekt. Über den Callback werden Informationen zum Senden übergeben.</p> <p>Aufbau des Callbacks siehe im Anhang.</p>
	► pMid (^INT)	Zeiger auf eine Variable, wo die Nachrichten ID

		gespeichert werden soll.
	► pApplMsgID (^UDINT)	Zeiger auf eine Variable, wo die interne Applikations- Nachrichten ID gespeichert werden soll.
	◄ retCode (INT)	0...erfolgreich 1...kein Speicher mehr vorhanden 3...Übergabe Parameter ungültig 4...Keine Verbindung zum MQTT Server 18...Topic Name ist nicht richtig formatiert. (Unicode)
MQTT_SetCleanSession	Mit dieser Methode können alle offenen Sessions gelöscht werden. Alle eingetragenen Elemente für die „Publish“ & „Subscribe“ Vorgänge werden entfernt.	
	◄ bOk (BOOL)	True...Alle Sessions wurden gelöscht False...Löschen der Sessions fehlgeschlagen.
Logging_CreateLog	Mit dieser Methode können Logeinträge erstellt werden.	
	► usLevel (USINT)	Level des Logeintrags: Info (2#00001) Warning (2#00100) Error (2#01000) Debug (2#10000)
	► udMsgGroup (UDINT)	Gruppen Nummer der Nachricht
	► udMsgNbr (UDINT)	Nachrichten Nummer
	► pMsg (^CHAR)	Zeiger auf den Nachrichtentext
	► pPara1 (^DINT)	Zeiger auf einen optionalen Parameter
	► pPara2 (^DINT)	Zeiger auf einen optionalen Parameter
	► pPara3 (^DINT)	Zeiger auf einen optionalen Parameter
	► pPara4 (^DINT)	Zeiger auf einen optionalen Parameter
Logging_GetLastClientEvent	Mit dieser Methode kann einen eindeutigen Wert für den zuletzt aufgetretenen MQTTClient Log-Eintrag in der Applikation gelesen werden.	
	► pGroup (DINT)	Zeiger auf die Variable, auf die die Gruppennummer geschrieben werden soll.
	► pMessage (DINT)	Zeiger auf die Variable, auf die die Nachrichtennummer geschrieben werden soll.
	► Level (DINT)	Level des gewünschten zuletzt aufgetretenen Eintrags.

	<table> <tr> <td>◀ bOK (BOOL)</td><td>True...OK False...pGroup = NIL / pMessage = NIL.</td></tr> </table>	◀ bOK (BOOL)	True...OK False...pGroup = NIL / pMessage = NIL.						
◀ bOK (BOOL)	True...OK False...pGroup = NIL / pMessage = NIL.								
Log- ging_GetLastInterface Event	<p>Mit dieser Methode kann einen eindeutigen Wert für den zuletzt aufgetretenen MQTTClient_Interface Log-Eintrag in der Applikation gelesen werden.</p> <table> <tr> <td>▶ pGroup (DINT)</td><td>Zeiger auf die Variable, auf die die Gruppennummer geschrieben werden soll.</td></tr> <tr> <td>▶ pMessage (DINT)</td><td>Zeiger auf die Variable, auf die die Nachrichtennummer geschrieben werden soll.</td></tr> <tr> <td>▶ Level (DINT)</td><td>Level des gewünschten zuletzt aufgetretenen Eintrags.</td></tr> <tr> <td>◀ bOK (BOOL)</td><td>True...OK False...pGroup = NIL / pMessage = NIL.</td></tr> </table>	▶ pGroup (DINT)	Zeiger auf die Variable, auf die die Gruppennummer geschrieben werden soll.	▶ pMessage (DINT)	Zeiger auf die Variable, auf die die Nachrichtennummer geschrieben werden soll.	▶ Level (DINT)	Level des gewünschten zuletzt aufgetretenen Eintrags.	◀ bOK (BOOL)	True...OK False...pGroup = NIL / pMessage = NIL.
▶ pGroup (DINT)	Zeiger auf die Variable, auf die die Gruppennummer geschrieben werden soll.								
▶ pMessage (DINT)	Zeiger auf die Variable, auf die die Nachrichtennummer geschrieben werden soll.								
▶ Level (DINT)	Level des gewünschten zuletzt aufgetretenen Eintrags.								
◀ bOK (BOOL)	True...OK False...pGroup = NIL / pMessage = NIL.								
Log- ging_GetLastTopicEv ent	<p>Mit dieser Methode kann einen eindeutigen Wert für den zuletzt aufgetretenen MQTTTopic Log-Eintrag in der Applikation gelesen werden.</p> <table> <tr> <td>▶ pGroup (DINT)</td><td>Zeiger auf die Variable, auf die die Gruppennummer geschrieben werden soll.</td></tr> <tr> <td>▶ pMessage (DINT)</td><td>Zeiger auf die Variable, auf die die Nachrichtennummer geschrieben werden soll.</td></tr> <tr> <td>▶ Level (DINT)</td><td>Level des gewünschten zuletzt aufgetretenen Eintrags.</td></tr> <tr> <td>◀ bOK (BOOL)</td><td>True...OK False...pGroup = NIL / pMessage = NIL.</td></tr> </table>	▶ pGroup (DINT)	Zeiger auf die Variable, auf die die Gruppennummer geschrieben werden soll.	▶ pMessage (DINT)	Zeiger auf die Variable, auf die die Nachrichtennummer geschrieben werden soll.	▶ Level (DINT)	Level des gewünschten zuletzt aufgetretenen Eintrags.	◀ bOK (BOOL)	True...OK False...pGroup = NIL / pMessage = NIL.
▶ pGroup (DINT)	Zeiger auf die Variable, auf die die Gruppennummer geschrieben werden soll.								
▶ pMessage (DINT)	Zeiger auf die Variable, auf die die Nachrichtennummer geschrieben werden soll.								
▶ Level (DINT)	Level des gewünschten zuletzt aufgetretenen Eintrags.								
◀ bOK (BOOL)	True...OK False...pGroup = NIL / pMessage = NIL.								
Log- ging_GetLastGlobalE vent	<p>Mit dieser Methode kann einen eindeutigen Wert für den zuletzt aufgetretenen Log-Eintrag in der Applikation gelesen werden.</p> <table> <tr> <td>▶ pGroup (DINT)</td><td>Zeiger auf die Variable, auf die die Gruppennummer geschrieben werden soll.</td></tr> <tr> <td>▶ pMessage (DINT)</td><td>Zeiger auf die Variable, auf die die Nachrichtennummer geschrieben werden soll.</td></tr> <tr> <td>▶ Level (DINT)</td><td>Level des gewünschten zuletzt aufgetretenen Eintrags.</td></tr> <tr> <td>◀ bOK (BOOL)</td><td>True...OK False...pGroup = NIL / pMessage = NIL.</td></tr> </table>	▶ pGroup (DINT)	Zeiger auf die Variable, auf die die Gruppennummer geschrieben werden soll.	▶ pMessage (DINT)	Zeiger auf die Variable, auf die die Nachrichtennummer geschrieben werden soll.	▶ Level (DINT)	Level des gewünschten zuletzt aufgetretenen Eintrags.	◀ bOK (BOOL)	True...OK False...pGroup = NIL / pMessage = NIL.
▶ pGroup (DINT)	Zeiger auf die Variable, auf die die Gruppennummer geschrieben werden soll.								
▶ pMessage (DINT)	Zeiger auf die Variable, auf die die Nachrichtennummer geschrieben werden soll.								
▶ Level (DINT)	Level des gewünschten zuletzt aufgetretenen Eintrags.								
◀ bOK (BOOL)	True...OK False...pGroup = NIL / pMessage = NIL.								

Anhang

Definition des Callbacks für die Methoden MQTT_Subscribe(), MQTT_Unsubscribe() und MQTT_Publish():

```

FUNCTION GLOBAL MQTTTopic::PubSubData_Callback
VAR_INPUT
    pThis      : ^void;
    MsgType    : MQTTClient_Interface::t_e_PubSubRetCodes;
    iMid       : INT;
    iGrantedQoS : INT;
    pMessage   : ^MQTTClient::t_s_MQTTMessage;
END_VAR

```

pThis (^VOID)	Zeiger auf das Objekt
MsgType (t_e_PubSubRetCodes)	Art der Nachricht: PSRC_Subscribed...Rückmeldung zum Subscribe Vorgang. PSRC_Unsubscribed...Rückmeldung zum Unsubscribe Vorgang. PSRC_PublishReveived...Daten vom „subscribed“ Topic erhalten. PSRC_Published...Rückmeldung zum Publish Vorgang PSRC_Disconnected...MQTT Client was disconnected
iMid (INT)	Nachrichten ID
iGrantedQoS (INT)	Vom MQTT Server gewährte Quality of Service. Entspricht dieser Wert 16#80 und der MsgType ist PSRC_Subscribed, dann ist der „Subscribe“ Vorgang fehlgeschlagen.
pMessage (^t_s_MQTTMessage)	Zeiger auf die Nachrichten Strukur.

Muster für die Konfigurations.xml Datei:

```
<?xml version="1.0" encoding="UTF-8"?>
<MQTTClient>

  <Config>
    <ClientID>UniquelD</ClientID>
    <Connection>
      <hostname>10.10.55.50</hostname>
      <port>1883</port>
      <username>username</username>
      <password>sigmatek</password>
      <keepalive>60</keepalive>
      <max_inflight_messages>20</max_inflight_messages>
    </Connection>
    <Reconnection>
      <min_interval>3</min_interval>
      <max_interval>30</max_interval>
      <exponential_scale>1</exponential_scale>
    </Reconnection>
    <TLS>
      <cafile>ca.crt</cafile>
      <certfile>certfile.cert</certfile>
      <keyfile>keyfile.key</keyfile>
      <pw>password</pw>
    </TLS>
    <Logging>
      <info>1</info>
      <notice>1</notice>
      <warning>1</warning>
      <error>1</error>
      <debugging>1</debugging>
    </Logging>
    <Will>
      <topic>MyLastTopicWill</topic>
      <payloadtext>My belongins will be handed to...</payloadtext>
      <qos>2</qos>
      <retain>0</retain>
    </Will>
  </Config>
```

```
<DataMap>
<topicsection name="FirstTopicPart">
<topicsection name="SecondTopicPart">
<topicsection name="ThirdTopicPart">
<topicsection name="FourthTopicPart">
  <topicendpoint name="TopicNameDINT" element="Objectname.Servername1" operation="subscribe"
    datatype="dint" qos="2" retain="false"/>
  <topicendpoint name="TopicNameUDINT" element="Objectname.Servername2" operation="poll"
    interval="500" datatype="uint" qos="2" retain="true"/>
  <topicendpoint name="TopicNameREAL" element="Objectname.Servername3" operation="change"
    interval="500" datatype="real" qos="1" retain="true"/>
  <topicendpoint name="TopicNameString" element="Objectname.Servername4" operation="change"
    interval="500" datatype="string" qos="0" retain="false"/>
</topicsection>
</topicsection>
</topicsection>
</DataMap>

</MQTTClient>
```

Elemente	Beschreibung
<MQTTClient>	Erstes Element
<Config>	Hier beginnt der Konfigurationsbereich
<ClientID>	Hier muss eine einzigartige ID für den MQTT Client gesetzt werden
<Connection>	Hier beginnt der Bereich der Verbindungsparameter
<hostname>	Hier muss die Adresse des MQTT Servers eingetragen werden.
<port>	Hier muss der Port des MQTT Servers eingetragen werden
<username>	Hier kann falls gefordert ein Benutzername definiert werden
<password>	Hier kann falls gefordert ein Passwort definiert werden
<keepalive>	Angabe in Sekunden wie oft ein Keep Alive Packet geschickt wird
<max_inflight_messages>	Hier wird definiert wie viele Nachrichten zur selben Zeit verarbeitet werden können
<Reconnection>	Hier beginnt der Bereich für die Wiederverbindungsparameter
<min_interval>	Angabe des minimalen Verbindungsversuchsintervalls in Sekunden
<max_interval>	Angabe des maximalen Verbindungsversuchsintervalls in Sekunden
<exponential_scale>	Angabe ob das Verbindungsversuchsintervall exponentiell steigen soll.
<TLS>	Hier beginnt der Bereich für TLS Verschlüsselungsparameter. (Optional)
<cafile>	Pfad der PEM kodierten CA Zertifikat Datei, welche die vertrauten „Root Server“ enthält. Wird kein Pfad angegeben, so werden die default Zertifikate unter dem Pfad C:\LSLSYS\SSL\ verwendet.
<certfile>	Pfad einer optionalen PEM kodierten Datei, welche die Client Zertifikate (Zertifikat Kette) enthält. Diese Datei muss im Verzeichnis C:\LSLSYS\SSL\ liegen. Der Pfad darf eine Länge von 255 Zeichen nicht überschreiten. Wird ein Pfad übergeben so muss auch eine gültige „Key-„ Datei übergeben werden. (<keyfile>)
<keyfile>	Pfad einer PEM Kodierten Datei, die die „private Keys“ des Clients enthält. Diese Datei muss im Verzeichnis C:\LSLSYS\SSL\ liegen. Der Pfad darf eine Länge von 255 Zeichen nicht überschreiten.
<pw>	Optionales Passwort das zur Verschlüsselung der „Key-„ Datei verwendet wurde. Das Passwort darf eine Länge von 1023 Zeichen nicht überschreiten.
<Logging>	Hier beginnt der Bereich für die Logging Parameter
<info>	Gibt an ob Meldungen vom Level Info geloggt werden sollen
<notice>	Gibt an ob Meldungen vom Level Notice geloggt werden sollen
<Warning>	Gibt an ob Meldungen vom Level Warning geloggt werden sollen
<Error>	Gibt an ob Meldungen vom Level Error geloggt werden sollen
<debugging>	Gibt an ob Meldungen vom Level Debug geloggt werden sollen
<Will>	Hier beginnt der Bereich für „Will“ Parameter. (Optional) Verliert der MQTT Server die Verbindung zum MQTT Client, so schreibt

	der Server auf das definiert Topic die „Will“ Nachricht des Clients.
<topic>	Topic Name für den letzten Willen
<payloadtext>	Letzter Wille Nachricht
<qos>	Quality of Service für den letzten Willen
<retain>	Gibt an ob der letzte Wille im MQTT Server gespeichert werden soll.
<DataMap>	Hier beginnt der Bereich für das Daten mapping. (Optional)
<Topicsection>	Topic Name. Sind mehrere <TopicSection> verschachtelt so wird der Topic Name immer erweitert bis zum <topicendpoint>
<Topicendpoint>	<p>Element das gemappt werden soll.</p> <p>name ... letzter Teil des Topic Namens. Wird an die Verschachtelungen von <TopicSection> angehängt.</p> <p>element ... Referenz auf das Element (Objektnamen.Servername)</p> <p>operation ... definiert die Operation für das Element. (subscribe, poll (Element wird im Intervall gesendet), change (Element wird bei Änderung gesendet))</p> <p>interval ... Intervall Zeit für die Operationen poll und change. [ms]</p> <p>datatype ... dint, uint, real, string</p> <p>qos ... Quality of Service</p> <p>retain ... gibt an ob die Nachricht im MQTT Server gespeichert werden soll.</p>

Definition der Log-Einträge, die von den Klassen im MQTT Package erzeugt werden können:

MQTTTopic

Group	Number	Type	Text
30	0	ERROR	Subscribe: Executing the Subscribe method in the MQTTClient_Interface failed. Para1=Mid, Para2 = MQTTClient_interface::MQTT_Subscribe() Retcode.
30	1	ERROR	Subscribe: The MQTTTopic instance subscription has failed. Para1=Last ApplMsgID
30	2	ERROR	Subscribe: Waiting for the Subscription confirmation timed out. The MQTTTopic instance is now in error state. Para1 = Last ApplMsgID
30	3	ERROR	Unsubscribe: Executing the Unsubscribe method in the MQTTClient_Interface failed. The MQTTTopic instance is now in error state. Para1 = RetCode. Para2 = Last ApplMsgID
30	4	ERROR	Unsubscribe: Waiting for the Unsubscribe Confirmation timed out. the MQTTTopic instance is now in error state. Para1 = Last ApplMsgID
30	5	ERROR	Publish: The User passed NIL data to the User_GetPublishData() method, the publish cannot continue, the MQTTTopic instance is now in Error State.
30	6	ERROR	Publish: Executing the Publish method in the MQTTClient_Interface failed. Para1=Retcode. Para2=ApplMsgID
30	7	ERROR	Publish: Waiting for the Publish confirmation timed out. The MQTTTopic instance is now in Error State. Para1=Last ApplMsgID. Para2=mid
32	2000	INFO	Subscribe: The MQTTTopic instance is now Subscribed. Para1=Mid, Para2=Granted QoS
32	2001	INFO	Unsubscribe: The MQTTTopic instance is now Unsubscribed. Para1=Mid
32	2002	INFO	Subscribe: Data has been received on the subscribed topic. Para1=Mid
32	2003	INFO	Publish: The Publish by the MQTTTopic instance has completed successfully. Para1=Mid
32	2004	INFO	Connection: The MQTT Client has been disconnected, the MQTTTopic instance is now waiting for a new connection. Para1= Last ApplMsgID
32	2005	INFO	Publish: The Publish interval has passed, the MQTTTopic instance will now attempt to publish.
32	2006	INFO	Publish: The Publish Data has changed, the MQTTTopic instance will now attempt to publish.
32	2007	INFO	Publish: The Manual Publish method has been called, the MQTTTopic instance will now attempt to publish.
32	2008	INFO	FSM: The MQTTTopic instance Config has changed, the instance will now re-initialize. Para1 = Last ApplMsgID
33	3000	DEBUG	FSM: The MQTTTopic instance error has been resetted. Para1= Last ApplMsgID

MQTTClient

<u>Group</u>	<u>Number</u>	<u>Type</u>	<u>Text</u>
20	0	ERROR	TCP: Receiving MQTT Packet timed out, packet dropped. Para1=Packet ID
20	1	ERROR	Publish: QoS 2 Outgoing Publish, PUBCOMP Max Retries reached, Packet Dropped. Para1=PacketID
20	2	ERROR	Publish: QoS 2 Outgoing Publish, PUBREL Max Retries reached, Packet Dropped. Para1=PacketID
20	3	ERROR	Publish: QoS 2 Outgoing Publish, PUBREL Insert into Packet Queue Failed, Dropped. Para1=PacketID
20	4	ERROR	Publish: QoS 2 Outgoing Publish, PUBREC Max Retries reached, Packet Dropped. Para1=PacketID
20	5	ERROR	Publish: QoS 2 Outgoing Publish, PUBLISH Max Retries reached, Packet Dropped. Para1=PacketID
20	6	ERROR	Publish: QoS 2 Outgoing Publish, PUBLISH Insert into Packet Queue Failed, Dropped. Para1=PacketID
20	7	ERROR	Publish: QoS 1 Outgoing Publish, PUBACK Max Retries reached, Packet Dropped. Para1=PacketID
20	8	ERROR	Publish: QoS 1 Outgoing Publish, PUBLISH Max Retries Reached, Packet Dropped. Para1=PacketID
20	9	ERROR	Publish: QoS 1 Outgoing Publish, PUBLISH Insert Packet into Packet Queue failed, Dropped. Para1=PacketID
20	10	ERROR	Publish: QoS 0 Outgoing Publish, PUBLISH Max Retries Reached, Packet Dropped. Para1=PacketID
20	11	ERROR	Publish: QoS 0 Outgoing Publish, Message Insert Packet into Packet Queue Failed, Dropped. Para1=PacketID
20	12	ERROR	Publish: QoS 2 Incoming Publish, sending PUBCOMP max retries reached, dropped. Para1=PacketID.
20	13	ERROR	Publish: QoS 2 Incoming Publish, inserting PUBCOMP into Packet Queue failed, dropped. Para1=PacketID
20	14	ERROR	Publish: QoS 2 Incoming Publish, waiting for PUBREL maximum retries reached, dropped. Para1=PacketID.
20	15	ERROR	Publish: QoS 2 Incoming Publish, sending PUBREC maximum retries reached, dropped. Para1=PacketID.
20	16	ERROR	Publish: QoS 2 Incoming Publish, failed to insert PUBREC response message into Message Queue. Para1=PacketID
20	17	ERROR	Publish: QoS 1 Incoming Publish, sending PUBACK max retries reached, dropped. Para1=PacketID.
20	18	ERROR	Publish: QoS 1 Incoming Publish, could not insert the PUBACK into the Packet Queue, dropped. Para1=PacketID.
20	19	ERROR	Unsubscribe: Outgoing Unsubscribe, sending UNSUBSCRIBE Packet max retries reached, dropped. Para1=PacketID.
20	20	ERROR	Subscribe: Outgoing Subscribe, sending SUBSCRIBE Packet max retries reached, dropped. Para1=PacketID.
20	21	ERROR	Publish: Incoming Publish Message, input values not valid, packet dropped. Para1=QoS.
20	22	ERROR	Publish: Incoming Publish Message, adding inflight message failed, Packet Dropped. Para1=PacketID.

20	23	ERROR	Publish: Incoming PUBCOMP Message, incorrect size message received, message not interpreted, dropped.
20	24	ERROR	Publish: Incoming PUBCOMP Message, corresponding PUBLISH message not found, dropped. Para1=PacketID.
20	25	ERROR	Publish: Incoming PUBCOMP Message, corresponding message state not OK. Para1=PacketID.
20	26	ERROR	Publish: Incoming PUBREL Message, incorrect size message received, message not interpreted, dropped.
20	27	ERROR	Publish: Incoming PUBREL Message, corresponding PUBLISH not found, PUBCOMP dropped. Para1=PacketID.
20	28	ERROR	Publish: Incoming PUBREL Message, corresponding message state not OK. Para1=PacketID.
20	29	ERROR	Publish: Incoming PUBREC Message, incorrect size message received, message not interpreted, dropped.
20	30	ERROR	Publish: Incoming PUBREC Message, corresponding PUBLISH not found, PUBCOMP dropped. Para1=PacketID.
20	31	ERROR	Publish: Incoming PUBREC Message, corresponding message state not OK. Para1=PacketID.
20	32	ERROR	Publish: Incoming PUBACK Message, incorrect size message received, message not interpreted, dropped.
20	33	ERROR	Publish: Incoming PUBACK Message, corresponding PUBLISH not found, dropped. Para1=PacketID.
20	34	ERROR	Publish: Incoming PUBACK Message, corresponding message state not OK. Para1=PacketID.
20	35	ERROR	Unsubscribe: Incoming UNSUBACK Message, incorrect size message received, message not interpreted, dropped.
20	36	ERROR	Unsubscribe: Incoming PUBACK Message, corresponding PUBLISH not found, dropped. Para1=PacketID.
20	37	ERROR	Subscribe: Incoming SUBACK Message, incorrect size message received, message not interpreted, dropped.
20	38	ERROR	Subscribe: Incoming SUBACK Message, corresponding SUBSCRIBE not found, dropped. Para1=PacketID.
20	39	ERROR	Subscribe: Incoming SUBACK, 0 granted QoS Pairs received, Dropped. Para1=PacketID.
20	40	ERROR	Keepalive: Incoming PINGREQ Message, incorrect data length, message dropped.
20	41	ERROR	Connection: Incoming CONNACK Message, Incorrect state - CONNACK Unexpected, message has been dropped. Para1=Retcode
20	42	ERROR	Connection: Incoming CONNACK Message, Unexpected RetCode, Connection Failed. Para1=Retcode.
20	43	ERROR	KeepAlive: Outgoing PINGRESP Message, Failed to add to Inflight Messages, Packet Dropped.
20	44	ERROR	KeepAlive: Outgoing PINGREQ Message, Failed to add to Inflight Messages, Packet Dropped.
20	45	ERROR	TCP: TCP Connection Dropped. Para1=TCP ErrorGroup, Para2=TCP ErrorCode. Para3=_FSM_TCP_USER.
20	46	ERROR	TCP: TCP Packet append failed, packet dropped. Para1=Incoming PacketQueueID. Para2=AvailableBytes.

20	47	ERROR	TCP: TCP Communication, Socket Error. Para1=PacketQueue Return. Para2=Packet Len.
20	48	ERROR	MQTT: MQTT Incoming Packet Queue, client received unknown packet, Protocol Error. Dropped.
20	49	ERROR	MQTT: MQTT Incoming Packet Queue, client received DISCONNECT, Protocol Error. Dropped.
20	50	ERROR	MQTT: MQTT Incoming Packet Queue, client received UNSUBSCRIBE, Protocol Error. Dropped.
20	51	ERROR	MQTT: MQTT Incoming Packet Queue, client received SUBSCRIBE, Protocol Error. Dropped.
20	52	ERROR	MQTT: MQTT Incoming Packet Queue, client received CONNECT, Protocol Error. Dropped.
20	53	ERROR	MQTT: MQTT Incoming Packet Queue, Decoding packet length failed, packet dropped.
20	54	ERROR	TCP: TCP Communication, failed to send packet, data dropped. Para1=MessageID. Para2=TCP SendData Retcode.
20	55	ERROR	Connection: Disconnect procedure, sending DISCONNECT timed out, Disconnected.
20	56	ERROR	Connection: Connect procedure, adding CONNECT Packet to Packet Queue Failed, reconnecting.
20	57	ERROR	TCP: Connecting procedure, creating TCP connection timed out, reconnecting.
20	58	ERROR	TCP: Connecting procedure, adding TCP connection failed, reconnecting. Para1=TCP AddConnection Retcode.
20	59	ERROR	Connection: Connecting procedure, failed to get IP address for hostname, reconnecting.
20	60	ERROR	Configuration: Setting the amount of Inflight Messages failed not initialized. Para1=max_inflight_messages.
20	61	ERROR	Configuration: Setting Inflight Messages failed, No mem. Para1=num_msgs, Para2=Mem Avail, Para3=Mem Req
20	62	ERROR	Configuration: Setting Inflight Messages, ReAlloc Failed. Para1=num_msgs, Para2=Mem Avail, Para3=Mem Req.
20	63	ERROR	Subscribe: Subscribe request failed, input not OK.
20	64	ERROR	Publish: Could not accept publish request, invalid input values.
20	65	ERROR	Configuration: Starting the Multitask Thread Failed. Para1=_MultiTask::RESUME() RetCode
20	66	ERROR	Connection: Connect Async Call Failed, the Input to the method is invalid.
20	67	ERROR	MultiTask: The MQTTClient failed to suspend the Multitasking Thread. Para1=_MultiTask::SUSPEND() RetCode
20	68	ERROR	MultiTask: Creating the Multitask Thread Failed.
20	69	ERROR	TCP: The OS TCP User Interface is not available, the connection cannot be made
21	1000	WARNING	Publish: QoS 2 Outgoing Publish, PUBREL wait timed out, Retrying. Para1=PacketID
21	1001	WARNING	Publish: QoS 2 Outgoing Publish, PUBCOMP wait timed out, Retrying. Para1=PacketID
21	1002	WARNING	Publish: QoS 2 Outgoing Publish, PUBREC wait timed out,

			Retrying. Para1=PacketID
21	1003	WARNING	Publish: QoS 2 Outgoing Publish, PUBLISH wait timed out, Retrying. Para1=PacketID
21	1004	WARNING	Publish: QoS 1 Outgoing Publish, PUBACK wait timed out, Retrying. Para1=PacketID.
21	1005	WARNING	Publish: QoS 1 Outgoing Publish, PUBLISH wait timed out, Retrying. Para1=PacketID
21	1006	WARNING	Publish: QoS 0 Outgoing Publish, PUBLISH wait timed out, Retrying. Para1=PacketID
21	1007	WARNING	Publish: QoS 2 Incoming Publish, could not deliver to application as the OnMessage Callback is not set.
21	1008	WARNING	Publish: QoS 2 Incoming Publish, waiting for PUBCOMP send timed out, retrying. Para1=PacketID.
21	1009	WARNING	Publish: QoS 2 Incoming Publish, waiting for PUBREL message timed out, retry sending PUBREC. Para1=PacketID.
21	1010	WARNING	Publish: QoS 2 Incoming Publish, sending PUBREC response timed out, retrying. Para1=PacketID
21	1011	WARNING	Publish: QoS 1 Incoming Publish, could not deliver to application as the OnMessage Callback is not set.
21	1012	WARNING	Publish: QoS 1 Incoming Publish, waiting for PUBACK send timed out, retrying. Para1=PacketID.
21	1013	WARNING	Publish: QoS 0 Incoming Publish, could not deliver to application as the OnMessage Callback is not set.
21	1014	WARNING	Unsubscribe: Outgoing Unsubscribe, waiting for UNSUBSCRIBE Packet send timed out, retrying. Para1=PacketID.
21	1015	WARNING	Subscribe: Outgoing Subscribe, waiting for SUBSCRIBE Packet send timed out, retrying. Para1=PacketID.
21	1016	WARNING	Subscribe: Incoming SUBACK, too many granted QoS Pairs, Truncated. Para1=PacketID.
21	1017	WARNING	Keepalive: MQTT Keepalive, PINGREQ <-> PINGRESP timed out, attempting reconnect.
21	1018	WARNING	Connection: Connect procedure, awaiting CONNACK timed out, attempting reconnect.
21	1019	WARNING	Connection: Reconnect procedure, reconnect wait timed out, retrying.
21	1020	WARNING	Configuration: Setting the Inflight Messages failed, supplied value too low, setting ignored. Para1=Min Value, Para2=Set Value
21	1021	WARNING	Configuration: Set Inf Msgs failed, incorrect MQTTClient state. Para1=MQTTClientState
21	1022	WARNING	Subscribe: Subscription ignored, the MQTTClient is in the incorrect state. Para1=MQTTClientState
21	1023	WARNING	Publish: Publish ignored, the MQTTClient is in the incorrect state. Para1=MQTTClientState
21	1024	WARNING	Connection: A call to mqtt_loop_stop() was made whilst the connection is still active, the call was ignored.
22	2000	INFO	Publish: QoS 1 Outgoing Publish, Message successfully sent. Para1=PacketID
22	2001	INFO	Publish: QoS 2 Outgoing Publish, Message successfully sent.

			Para1=PacketID
22	2002	INFO	Publish: QoS 0 Outgoing Publish, Message successfully send. Para1=PacketID
22	2003	INFO	Publish: QoS 2 Incoming Publish Delivered to Application. Para1=mid
22	2004	INFO	Publish: QoS 1 Incoming Publish Delivered to Application. Para1=mid
22	2005	INFO	Publish: QoS 0 Incoming Publish Delivered to Application. Para1=mid
22	2006	INFO	Publish: Incoming PUBCOMP successfully Processed. Para1=mid
22	2007	INFO	Publish: Incoming PUBREL successfully Processed. Para1=mid
22	2008	INFO	Publish: Incoming PUBREC successfully Processed. Para1=mid
22	2009	INFO	Publish: Incoming PUBACK successfully Processed. Para1=mid
22	2010	INFO	Unsubscribe: Incoming UNSUBACK successfully Processed. Para1=mid
22	2011	INFO	Subscribe: Incoming SUBACK successfully Processed. Para1=mid
22	2012	INFO	Connection: Incoming CONNACK Message, Connection Rejected, Not Authorized.
22	2013	INFO	Connection: Incoming CONNACK Message, Connection Rejected, Username/Password Incorrect.
22	2014	INFO	Connection: Incoming CONNACK Message, Connection Rejected, Server Unavailable.
22	2015	INFO	Connection: Incoming CONNACK Message, Connection Rejected, ID Rejected.
22	2016	INFO	Connection: Incoming CONNACK Message, Connection Rejected, Protocol Version Mismatch.
22	2017	INFO	Connection: Incoming CONNACK Message, Connection Accepted.
22	2018	INFO	Configuration: Set Inflight Messages success. Para1=num_msgs, Para2=Mem Avail, Para3=Mem Req.
22	2019	INFO	Subscribe: Successfully added an Inflight Message for a Subscription. Para1=mid, Para2=QoS
22	2020	INFO	Publish: Successfully added inflight message for publish. Para1=mid
22	2021	INFO	MultiTask: The MultiTask Thread was stopped.

MQTTClient Interface

Group	Number	Type	Text
10	0	ERROR	Init: Failed to assign memory for DataMap Pub & Sub Entries
10	1	ERROR	Init: Failed to assign memory for DataMap Entries
10	2	ERROR	Init: Failed to assign Memory for PubSubList Main Structure
10	3	ERROR	Init: Failed to assign Memory for PubSubList Entries
10	4	ERROR	FSM: Creating Connection the the MQTT Broker Failed, Para1 = MQTTClient Retcode
10	5	ERROR	FSM Initialize: Failed to read the specified XML File.
10	6	ERROR	FSM Initialize: Preparing DataMap Failed - Pointer to DataMap Entries is NIL.
10	7	ERROR	FSM Prepare: Setting client ID failed, not enough memory.
10	8	ERROR	FSM Prepare: Error during setting MQTTClient Parameters from XML File, Setting ClientID Failed, Input Value / Client State Not OK.
10	9	ERROR	FSM Prepare: Error during setting MQTTClient Parameters from XML File, Setting Username & Password Failed, Not Enough memory
10	10	ERROR	FSM Prepare: Error during setting MQTTClient Parameters from XML File, Setting Username & Password Failed, Input Value / Client State Not OK
10	11	ERROR	FSM Prepare: Error during setting MQTTClient Parameters from XML File, Setting Max Inflight Messages Failed, Input Value / Client State Not OK
10	12	ERROR	FSM Prepare: Setting will failed. Not enough memory
10	13	ERROR	FSM Prepare: Setting will failed. Invalid input parameters or wrong client state
10	14	ERROR	FSM Prepare: Setting will failed. Invalid Payload size. Para1 = Payload size
10	15	ERROR	FSM Prepare: Setting will failed. Malformed UTF8
10	16	ERROR	FSM Prepare: Setting reconnect failed. Invalid input values or state. Para1 = T Min. Para2 = T Max. Para3 = Exp
10	17	ERROR	FSM Prepare: Setting TLS Data failed. Not enough Memory
10	18	ERROR	FSM Prepare: Setting TLS Data failed. Invalid input parameters or invalid client state
10	19	ERROR	Subscribe: Subscribe failed, Incorrect Input Para. Para1= pTopic. Para2=Qos. Para3=pThis. Para4 = pCallback
10	20	ERROR	Subscribe: Subscribe Failed, incorrect state. Para1 = InterfaceState
10	21	ERROR	Subscribe: Subscribe failed, No Mem in PubSub list. Para1=NumEntries. Para2=Max Num Entries

10	22	ERROR	Subscribe: Subscribe failed, Para1=ReturnCode from client
10	23	ERROR	Publish: Publish failed, invalid input para. Para1=pTopic. Para2=QoS. Para3=Payload Len. Para4=pPayload
10	24	ERROR	Publish: Publish failed, No Broker connection or not initialized. Para1=InterfaceState
10	25	ERROR	Publish: Publish failed, No Mem in PubSub list. Para1=NumEntries. Para2=Max Num Entries
10	26	ERROR	Publish: Publish failed, Para1=ReturnCode from client
10	27	ERROR	Unsubscribe: Unsubscribe failed, Could not find topic in PubSub list by name
10	28	ERROR	Unsubscribe: Unsubscribe failed, Para1=ReturnCode from client
10	29	ERROR	Unsubscribe: Unsubscribe client request failed, incorrect input parameter. Para1 = Appl Msg Id. Para2 = mid
10	30	ERROR	Unsubscribe: Unsubscribe client request failed, MQTT Client is not connected to the broker. Para1 = Appl Msg Id. Para2 = mid
10	31	ERROR	Unsubscribe: Unsubscribe client request failed, No free memory in message buffer. Para1 = Appl Msg Id. Para2 = Mid
10	32	ERROR	Unsubscribe: Unsubscribe failed, Subscription could not be found in the PubSub List. Para1=ApplMsgID, Para2=mid
10	33	ERROR	Unsubscribe: Unsubscribe failed, incorrect input para
10	34	ERROR	Set Clean Session: Resetting the session failed, PubSub list is not Initialised
10	35	ERROR	Set Clean Session: Resetting the Session failed, MQTT Interface is not in the correct state. Para1 = InterfaceState
10	36	ERROR	Process XML: Root element in XML File is invalid. Cannot process
10	37	ERROR	Process XML Config: Processing Client ID Section of configuration XML file failed - Missing Entries.
10	38	ERROR	Process XML Config: Processing Connection Section of configuration XML file failed - Missing Entries.
10	39	ERROR	Process XML Config: Processing Will section of configuration XML file failed - Missing Entries.
10	40	ERROR	Process XML Config: Processing reconnection section of configuration XML file failed - Missing Entries.
10	41	ERROR	Process XML Config: Processing TLS section of configuration XML file failed - Missing Entries
10	42	ERROR	Process XML Config: Processing Logging section of configuration XML file failed - Missing Entries
10	43	ERROR	Process XML DataMap: Unexpected input in DataMap Section
10	44	ERROR	Process XML DataMap: Failed processing a Topic Section in the DataMap section.
10	45	ERROR	Process XML DataMap: Failed processing a Topic Endpoint in the DataMap section
10	46	ERROR	Process XML DataMap Topic Section: Processing Entry Failed - Section title too long.
10	47	ERROR	Process XML DataMap Topic Section: Failed Processing Entry - Section title is empty

10	48	ERROR	Process XML DataMap, Max topic depth (Nested Topic Sections) exceeded. Para1=Topic depth. Para2 = Max topic depth
10	49	ERROR	Process XML DataMap Topic Endpoint: Element name is too long or Zero, Para1 = Len
10	50	ERROR	Process XML DataMap Topic Endpoint: Element text is too long
10	51	ERROR	Process XML DataMap Topic Endpoint: Unknown operation type
10	52	ERROR	Process XML DataMap Topic Endpoint: Interval string too long, Para1 = Len
10	53	ERROR	Process XML DataMap Topic Endpoint: Unknown datatype
10	54	ERROR	Process XML DataMap Topic Endpoint: QoS string too long, Para1 = Len
10	55	ERROR	Process XML DataMap Topic Endpoint: QoS invalid. Para1 = QoS
10	56	ERROR	Process XML DataMap Topic Endpoint: Retain string too long. Para1 = Len
10	57	ERROR	Process XML DataMap Topic Endpoint: Retain value not known. Para1 = Retain value
10	58	ERROR	Process XML DataMap Topic Endpoint: Topic too long, Para1 = Len
10	59	ERROR	Process XML DataMap Topic Endpoint: Topic name too long, Para1 = Len
10	60	ERROR	Process XML DataMap Topic Endpoint: Adding Element to DataMap failed. Para 1 = DataMap RetCode.
10	61	ERROR	Process XML DataMap, Failed processing topic endpoint. All required attributes have not been set
10	62	ERROR	DataMap: Adding DataMap entry failed, Invalid input parameters received.
10	63	ERROR	DataMap: Adding DataMap entry failed, Topic name too long.
10	64	ERROR	DataMap: Adding Element failed - Element name too long. Para 1 = Len
10	65	ERROR	DataMap: Adding Element failed - Element not found.
10	66	ERROR	DataMap: Adding Element Failed - Type Mismatch.
10	67	ERROR	DataMap: Adding Publish Element Failed - Too Many Entries, see client DataMapEntries.
10	68	ERROR	DataMap: Adding Subscribe Element Failed - Too Many Entries, see client DataMapEntries.
10	69	ERROR	DataMap: Failed to allocate memory to send the string for the entry. Para1 = Mid. Para2 = String Object pThis
10	70	ERROR	DataMap: Publish Request Failed. Para1 = Mid, Para2 = MQTTClient RetCode, Para3 = pData
10	71	ERROR	DataMap: Publish Entry timed out whilst waiting for confirmation. Para1 = Mid, Para2 = Timeout, Para3 = pData
10	72	ERROR	DataMap: Request to subscribe failed. Para1 = Mid. Para2 = Retcode, Para3 = pData
10	73	ERROR	DataMap: Subscribe Entry timed out whilst waiting for confirmation. Para1 = Mid, Para2 = Timeout, Para3 = pData

10	74	ERROR	DataMap: Request to unsubscribe failed. Para1 = Mid, Para2 = Retcode, Para3 = pData
10	75	ERROR	DataMap: Unsubscribe Entry timed out whilst waiting for confirmation. Para1 = Mid, Para2 = Timeout, Para3 = pData
10	76	ERROR	DataMap: Failed to remove unsubscribe entry. Para1=Mid, Para2=Index
10	77	ERROR	DataMap: Removing Element Failed, Invalid index supplied. Para1 = Index
10	78	ERROR	DataMap: Received data, there is not enough memory available to process the message. Para1=iMid. Para2 = Message Mid
10	79	ERROR	DataMap: Subscription confirms with invalid QoS. Para1=iMid. Para2=Message Mid.Para3=Granted QoS.Para4=Requested Qos
10	80	ERROR	Connection: The connection was refused, the Protocol Versions mismatch. Para1 = CONNACK RetCode
10	81	ERROR	Connection: Connection failed. Client ID Rejected. Para1 = retcode
10	82	ERROR	Connection: Connection Rejected. Server Unavailable. Para1 = retcode
10	83	ERROR	Connection: Connection Rejected. Username and or Password incorrect. Para1 = retcode
10	84	ERROR	Connection: Connection Rejected. Not Authorized. Para1 = retcode
10	85	ERROR	Connection: Unexpectedly disconnected, KeepAlive Ping timeout or TCP error occurred. Para1 = RetCode
10	86	ERROR	Connection: Unexpectedly disconnected, Retcode unknown. Para1 = RetCode
10	87	ERROR	Publish: Received unexpected Publish Confirm from the MQTTClient, PubSub List not Initd. Para1 = mid
10	88	ERROR	Publish: Received unexpected Message (PUBLISH) from the MQTTClient, PubSub List not Initd. Para1 = mid
10	89	ERROR	Logging: Received a log request with no text. Para1 = Log Level
10	90	ERROR	Logging: Received a log request with incorrectly specified level. Para1 = Log Level
10	91	ERROR	Config Object: Setting the specified Client ID Failed, the string is not set or is too long
10	92	ERROR	Config Object: Setting the specified Hostname Failed, the string is not set or is too long
10	93	ERROR	Config Object: Setting the Port number failed, the specified port is outside of the the valid range. Para1 = Specified Port
10	94	ERROR	Config Object: Setting the KeepAlive Time failed, the specified value is outside of the the valid range. Para1 = Specified Time
10	95	ERROR	Config Object: Setting the Max Inflight Messages failed, the specified value is too low. Para1 = Min Value
10	96	ERROR	Config Object: Setting the specified Username Failed, the string is not set or is too long. Para1 = Max Len
10	97	ERROR	Config Object: Setting the specified Password Failed, the string is not set or is too long. Para1 = Max Len
10	98	ERROR	Config Object: Setting the specified Will Topic Failed, the string is not set or is too long. Para1 = Max Len

10	99	ERROR	Config Object: Setting the specified Will Payload Failed, the string is not set or is too long. Para1 = Max Len
10	100	ERROR	Config Object: Setting the Will QoS failed, the specified value is outside of the the valid range. Para1 = Specified QoS
10	101	ERROR	Config Object: Setting the Min Reconnect Time failed, the value is negative or >= Max Reconnect time.
10	102	ERROR	Config Object: Setting the Max Reconnect Time failed, the value is negative or <= Min Reconnect time.
10	103	ERROR	Config Object: Setting the specified TLS CAFile failed, the string is not set or is too long. Para1 = Max Len
10	104	ERROR	Config Object: Setting the specified TLS Cert File failed, the string is not set or is too long. Para1 = Max Len
10	105	ERROR	Config Object: Setting the specified TLS Key File failed, the string is not set or is too long. Para1 = Max Len
10	106	ERROR	Config Object: Setting the specified TLS Password failed, the string is not set or is too long. Para1 = Max Len
11	1000	WARNING	ClientActive: Disconnect request made although no connection is active
11	1001	WARNING	ClientActive: Unexpected retcode received whilst requesting MQTT Client Disconnect.
11	1002	WARNING	DataMap: Max String Len Exceeded, String Truncated. Para1=Max Payload Len, Para2=Received String Len.
11	1003	WARNING	DataMap: Publish Entry Temp Memory pointer was not NIL, this should however be the case, attempting free again. Para1 = Mid
11	1004	WARNING	DataMap: After publishing a String entry, the Temp Memory Pointer is NIL, this should not be the case, not freed. Para1 = mid, Para2 = String Object pThis
11	1005	WARNING	DataMap: Removing Element failed, DataMap not initialized. Para1 = Index
11	1006	WARNING	DataMap: Received callback whilst DataMap is not initialized. Para1=mid. Para2 = Message Mid
11	1007	WARNING	DataMap Received Data but did not find corresponding entry. Para1 = iMid. Para2 = Message Mid
11	1008	WARNING	DataMap: Subscription confirmed, however the Entry was in the wrong state. Para1 = mid, Para2 = Message mid, Para3 = Entry internal State
11	1009	WARNING	DataMap: Received data too long, truncated. (UDINT). Para1 = iMid. Para2 = Message Mid. Para3 = Received Len
11	1010	WARNING	DataMap: Received data too long, truncated. (DINT). Para1 = iMid. Para2 = Message Mid. Para3 = Received Len
11	1011	WARNING	DataMap: Received data too long, truncated. (REAL). Para1 = iMid. Para2 = Message Mid. Para3 = Received Len
11	1012	WARNING	DataMap: Received empty message. Para1 = iMid. Para2 = Message Mid
11	1013	WARNING	Connection: Received an unexpected Connect Callback. Para1 = Retcode
11	1014	WARNING	Connection: Received an unexpected Disconnected Callback. Para1 = Retcode

11	1015	WARNING	Publish: Received unexpected Publish Confirm from the MQTTClient, item not in the PubSub List. Para1 = mid
11	1016	WARNING	Publish: Received a Publish Confirm from the MQTTClient that could not be delivered to the application, Callback details incorrect. Para1 = mid
11	1017	WARNING	Message: Received a Message from the MQTTClient, the PubSubList is empty. Para1 = mid
11	1018	WARNING	Message: Received a Message from the MQTTClient that could not be delivered to the application, Callback details incorrect. Para1 = mid
11	1019	WARNING	Subscribe: Received an unexpected Subscribe Confirm from the MQTTClient. Para1 = mid
11	1020	WARNING	Subscribe: Received an unexpected Subscribe Confirm from the MQTTClient, the entry is a Publish Entry. Para1 = mid
11	1021	WARNING	Subscribe: Received a Subscribe confirmation from the MQTTClient that could not be delivered to the application, Callback details incorrect. Para1 = mid
11	1022	WARNING	Subscribe: Received unexpected Subscribe Confirm from the MQTTClient, item not in the PubSub List. Para1 = mid
11	1023	WARNING	Unsubscribe: Received an Unsubscribe confirmation from the MQTTClient, the PubSubList is empty. Para1 = mid
11	1024	WARNING	Unsubscribe: Received an unexpected Unsubscribe Confirm from the MQTTClient, the matching entry in the PubSub List is not an Unsubscribe request. Para1 = mid
11	1025	WARNING	Unsubscribe: Received an Unsubscribe confirmation from the MQTTClient that could not be delivered to the application, Callback details incorrect. Para1 = mid
12	2000	INFO	FSM: MQTTClient_Interface has been activated
12	2001	INFO	Subscribe: Element successfully added to subscribe list. Para1= Mid
12	2002	INFO	Unsubscribe: Unsubscribe Request Successfully passed to MQTT Client. Para1 = Appl Msg Id. Para2 = mid
12	2003	INFO	DataMap: Requested Publish for an Entry. Para1 = Mid, Para2 = pData
12	2004	INFO	DataMap: Publish complete. Para1 = Mid, Para2 = pData
12	2005	INFO	DataMap: Requested subscription for an Entry. Para1 = Mid, Para2 = pData
12	2006	INFO	DataMap: Requested Unsubscribe for DataMap Entry. Para1 = Mid, Para2 = pData
12	2007	INFO	DataMap: Subscription Confirmed, Granted QoS differs from requested. Para1 = mid, Para2 = Message mid, Para3 = Granted QoS, Para4 = Requested QoS
12	2008	INFO	DataMap: Unsubscribe for Entry Received. Para1 = iMid. Para2 = Message Mid.
12	2009	INFO	Connection: Connected. Para1 = Retcode, Para2 = Flags
12	2010	INFO	Connection: Disconnected. Para1 = RetCode
12	2011	INFO	Publish: Received Publish confirmation from the MQTTClient.

			Para1 = mid
12	2012	INFO	Unsubscribe: Received Unsubscribe confirmation from the MQTTClient. Para1 = mid
13	3000	DEBUG	ClientActive: Disconnect request Initiated
13	3001	DEBUG	Unsubscribe: Unsubscribe requested, however no callback to the Application is set. Para1=udAppID, Para2=mid
13	3002	DEBUG	Unsubscribe: Unsubscribe requested for a topic where atleast one other party in the application is still subscribed, the Unsubscribe request will not be sent. Para1 = Requested Unsub ApplID, Para2 = ApplID of first subscriber that is still active.