

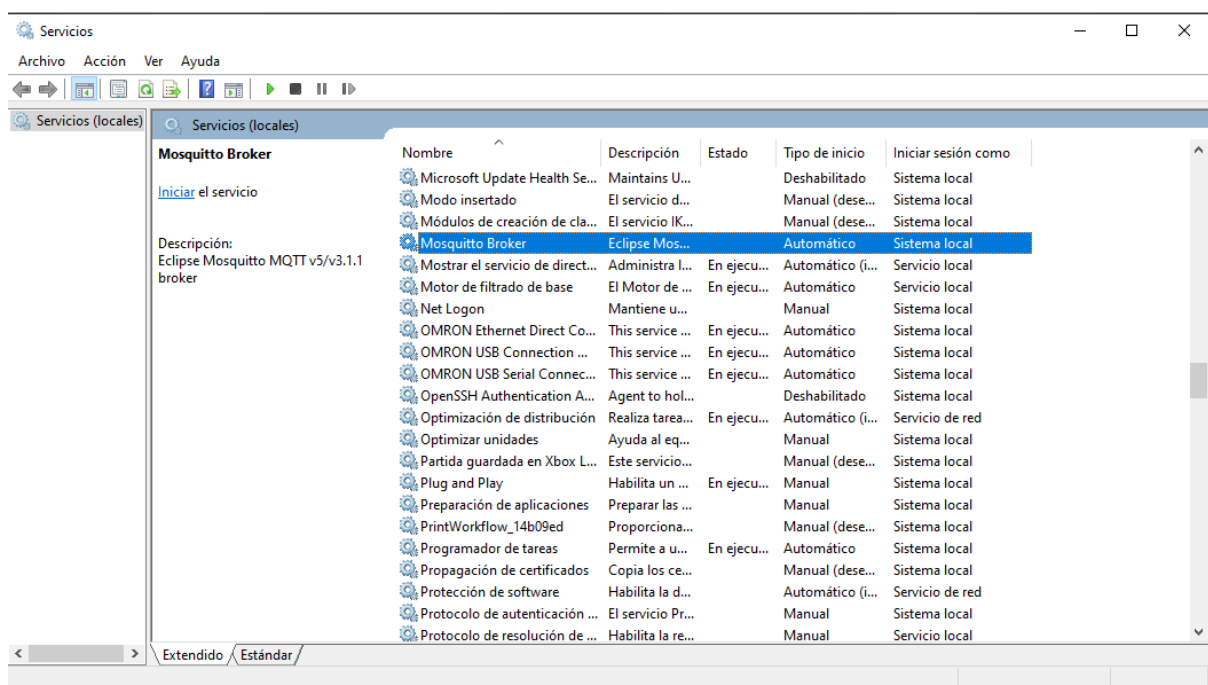
Cómo conectar con un servidor MQTT privado con mosquitto:

Primero descargamos mosquitto desde su página oficial(hay que tener en cuenta la versión de Windows que tiene usted, 32 o 64 bits):

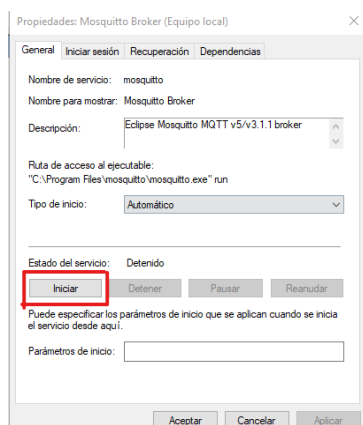
<https://mosquitto.org/download/>

Una vez descargado solo hace falta instalarlo (se recomienda que cuando le pida una ruta de descarga deje la predeterminada).

Para comprobar que mosquitto se ha instalado correctamente nos iremos al botón Windows de nuestro ordenador abajo a la izquierda de la pantalla y escribiremos “Servicios” para acceder a los servicios del sistema. Una vez abierto bajaremos hasta encontrar un servicio llamado “Mosquitto broker”



Tras encontrarlo le haremos doble clic y presionaremos iniciar si todavía no se está ejecutando.



Una vez esté en ejecución podemos cerrar “Servicios” y abrir una ventana de cmd, podemos buscarla como hemos hecho con los servicios o podemos pulsar Windows+R y escribir cmd.

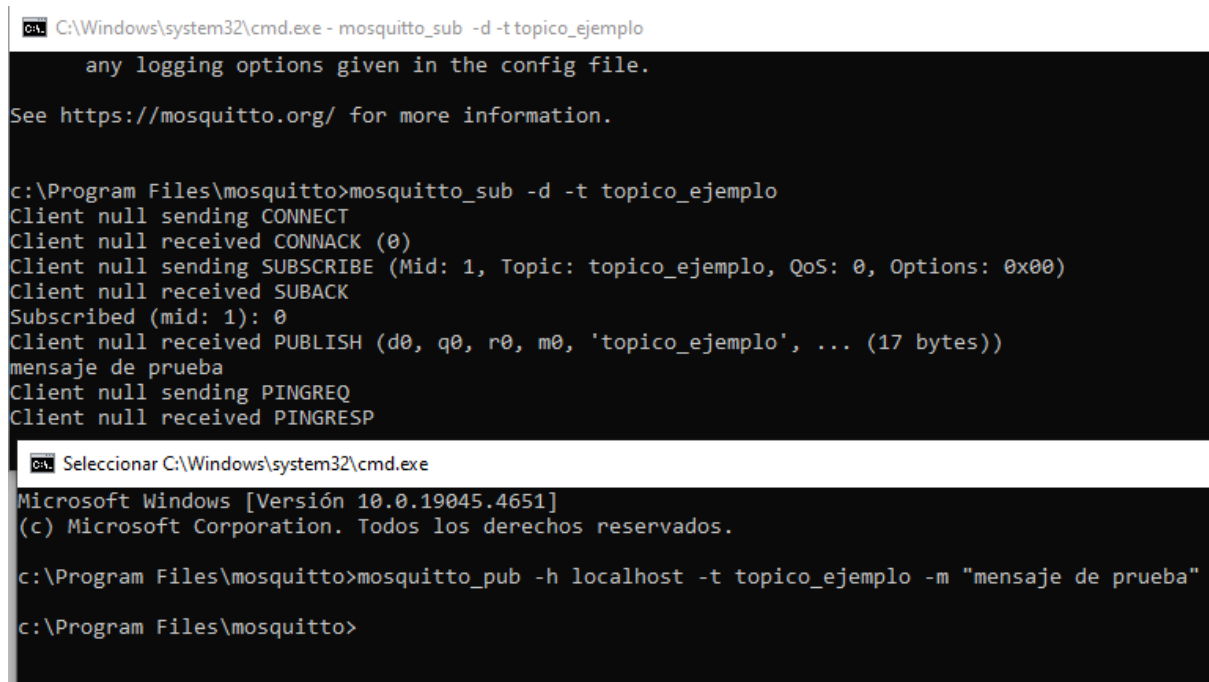
Tras abrir el cmd hay que ir a la carpeta de mosquitto escribiendo cd y su ruta por ejemplo si dejas la ruta por defecto que te crea el instalador sería “cd “c:/program files/mosquitto””. Se tienen que crear dos ventanas de mosquitto en cmd puedes crear otra ventana y volver a escribir la ruta o escribir “start” en la ya creada para crear otra automáticamente.

En la primera pestaña crearemos un subscribe para ello escribiremos “mosquitto_sub -d -t topico_ejemplo” y para la otra pestaña crearemos un publisher en host local escribiendo “mosquitto_pub -h localhost -t topico_ejemplo -m “mensaje de prueba””. Si usted quiere saber que hace cada comando puede escribir en la ruta de mosquitto “mosquitto -h”

Tras escribir el publisher tendría que salir en el subscribe el mensaje que usted ha escrito, si no es así compruebe que haya seguido correctamente los pasos o si le hace falta una guía más visual le dejamos un video aquí abajo para que lo siga.

<https://www.youtube.com/watch?v=4CIUBtMNnKU>

Tras todo esto las dos ventanas de cmd se tendrían que ver algo parecido a esto:



```
C:\Windows\system32\cmd.exe - mosquitto_sub -d -t topico_ejemplo
any logging options given in the config file.
See https://mosquitto.org/ for more information.

c:\Program Files\mosquitto>mosquitto_sub -d -t topico_ejemplo
Client null sending CONNECT
Client null received CONNACK (0)
Client null sending SUBSCRIBE (Mid: 1, Topic: topico_ejemplo, QoS: 0, Options: 0x00)
Client null received SUBACK
Subscribed (mid: 1): 0
Client null received PUBLISH (d0, q0, r0, m0, 'topico_ejemplo', ... (17 bytes))
mensaje de prueba
Client null sending PINGREQ
Client null received PINGRESP

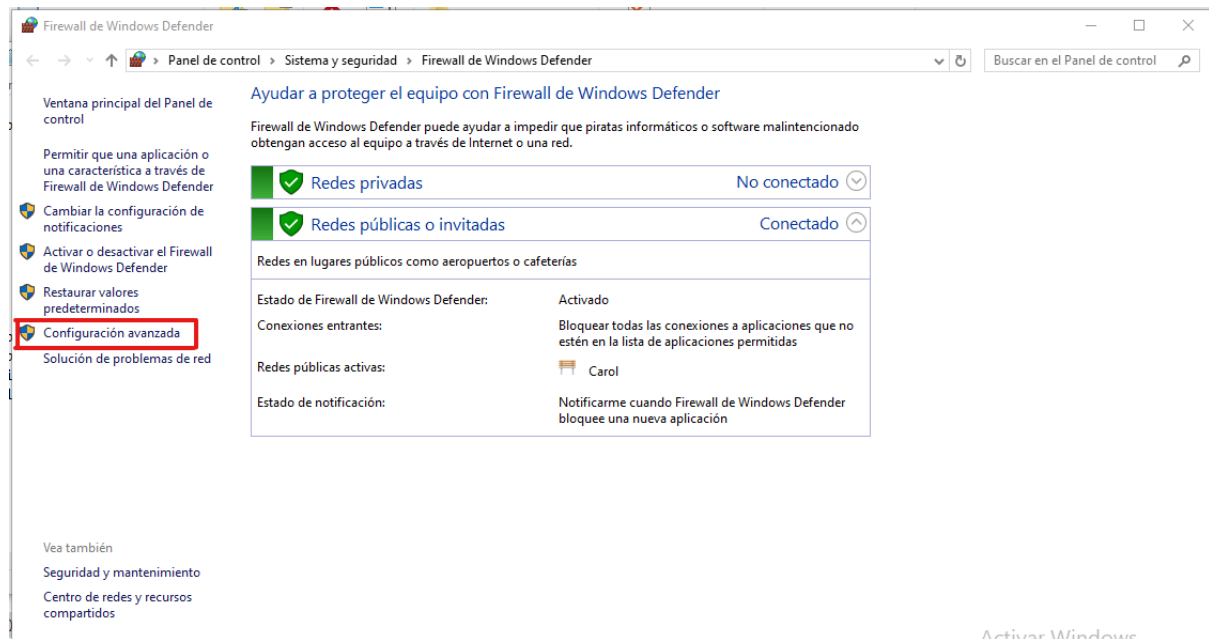
C:\Windows\system32\cmd.exe
Microsoft Windows [Versión 10.0.19045.4651]
(c) Microsoft Corporation. Todos los derechos reservados.

c:\Program Files\mosquitto>mosquitto_pub -h localhost -t topico_ejemplo -m "mensaje de prueba"
c:\Program Files\mosquitto>
```

Tras la prueba hay que abrir la carpeta de mosquitto desde el explorador de archivos de Windows (Es la misma ruta que hemos utilizado para buscar la carpeta de mosquitto en cmd).

Una vez dentro tenemos que buscar un archivo llamado “mosquitto.conf” lo abrimos y buscamos un apartado llamado “PSK based SSL/TLS support” (puedes buscarlo con más facilidad pulsado Ctrl+R. En este apartado hay que añadirle dos líneas en las que ponga “listener 1883” y “allow_anonymous true” tras introducir estas líneas guardamos el archivo y cerramos. Si no nos dejara guardar en la carpeta de mosquitto guardamos el archivo en el escritorio para luego borrar el que hay en la carpeta mosquitto y pegamos el que hemos dejado en el escritorio.

Ahora vamos otra vez al botón Windows y escribimos “firewall” y abrimos el firewall de Windows defender. Una vez abierto le pulsamos en “Configuración avanzada”.



Luego pulsamos en “Reglas de entrada” y creamos una nueva regla. Esta nueva regla será de tipo “Puerto” número 1883 con permiso de la conexión y la llamaremos “mosquito_regla_entrada”. Luego hay que hacer lo mismo pero para las “Reglas de salida” y tras crearlas y comprobar que estén activas reiniciamos el ordenador.

Cuando ya tengamos reiniciado el ordenador volveremos a probar que funcione correctamente mosquito poniendo otra vez el publisher y el subscribe, pero esta vez con la IP de nuestro ordenador (si no sabes la IP de tu ordenador escribe “ipconfig” para poder ver tus IPs) en nuestro caso es 192.168.5.225. Aquí un ejemplo de como quedaría (si no funcionase compruebe que tenga el Mosquito broker en ejecución en los servicios del sistema):

```
C:\Windows\system32\cmd.exe - mosquito_sub -d -t topico_ejemplo

c:\Program Files\mosquitto>mosquitto_sub -d -t topico_ejemplo
Client null sending CONNECT
Client null received CONNACK (0)
Client null sending SUBSCRIBE (Mid: 1, Topic: topico_ejemplo, QoS: 0, Options: 0x00)
Client null received SUBACK
Subscribed (mid: 1): 0
Client null received PUBLISH (d0, q0, r0, m0, 'topico_ejemplo', ... (17 bytes))
mensaje de prueba
Client null sending PINGREQ
Client null received PINGRESP

Seleccionar C:\Windows\system32\cmd.exe

Microsoft Windows [Versión 10.0.19045.4651]
(c) Microsoft Corporation. Todos los derechos reservados.

c:\Program Files\mosquitto>mosquitto_pub -h 192.168.5.225 -t topico_ejemplo -m "mensaje de prueba"
c:\Program Files\mosquitto>
```

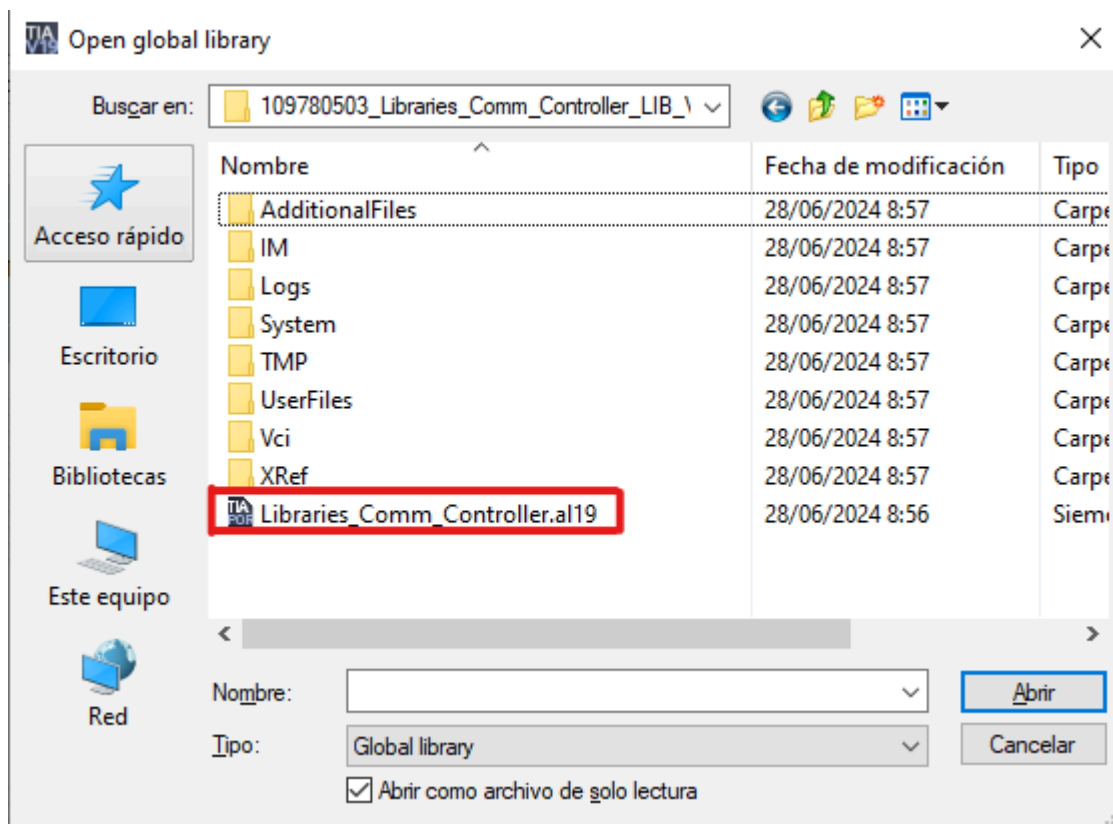
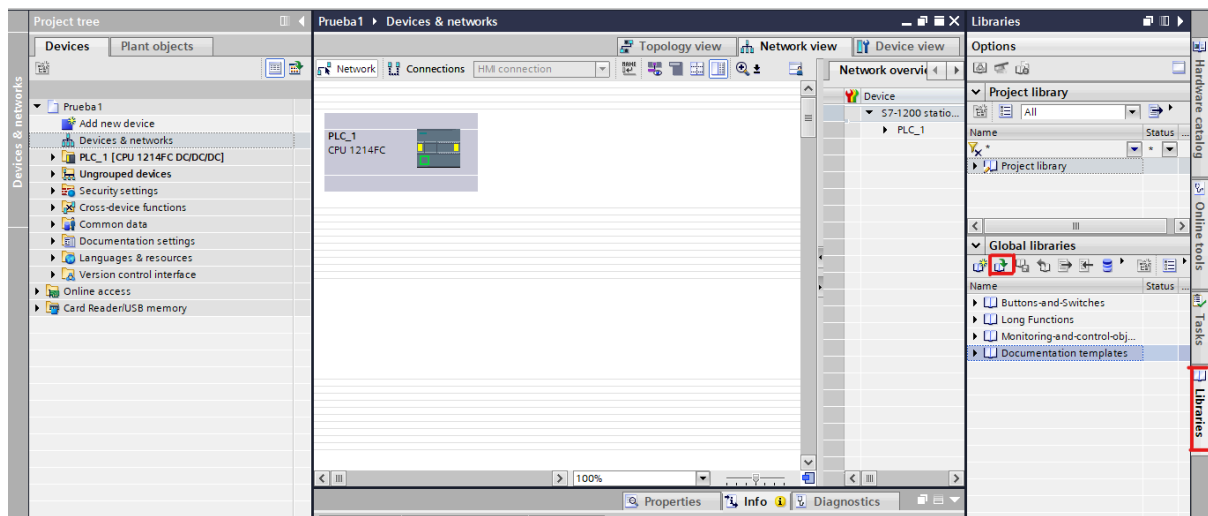
Ahora podemos abrir TIA Portal para crear nuestro publisher(para este ejemplo aprovechamos el mismo subscriber que hemos creado para ejemplificar, pero usted puede crear uno nuevo).

(Para este ejemplo estamos usando un TIA Portal 19v y un S71200)

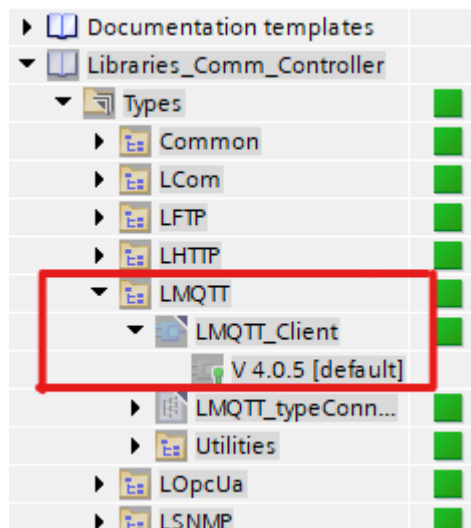
Primero descargamos la librería MQTT de Simatic desde la página web oficial(en nuestro caso descargamos la versión compatible con TIA Portal 19).

<https://support.industry.siemens.com/cs/document/109780503/libraries-for-communication-for-simatic-controllers?dti=0&lc=en-ES>

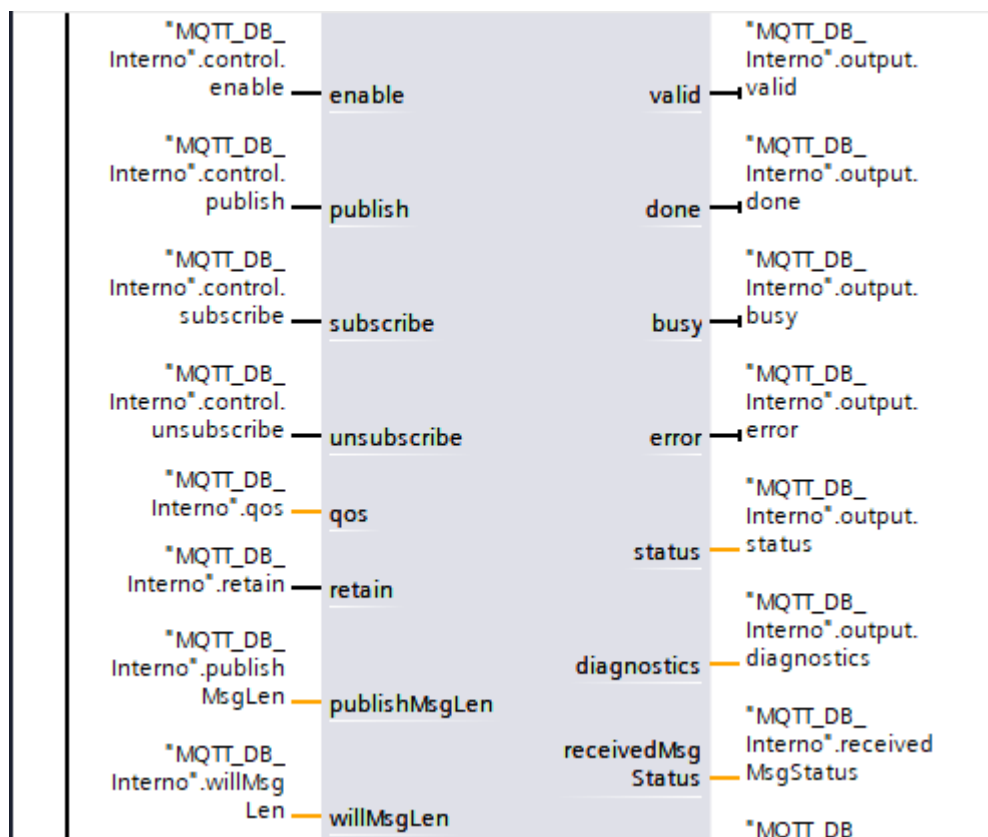
Una vez descargado y extraído vamos al TIA Portal y le clicamos dentro de “Librería” en donde hay un símbolo de libro y una flecha verde lo cual abrirá un navegador de archivos en el cual buscaremos nuestro archivo de la librería:



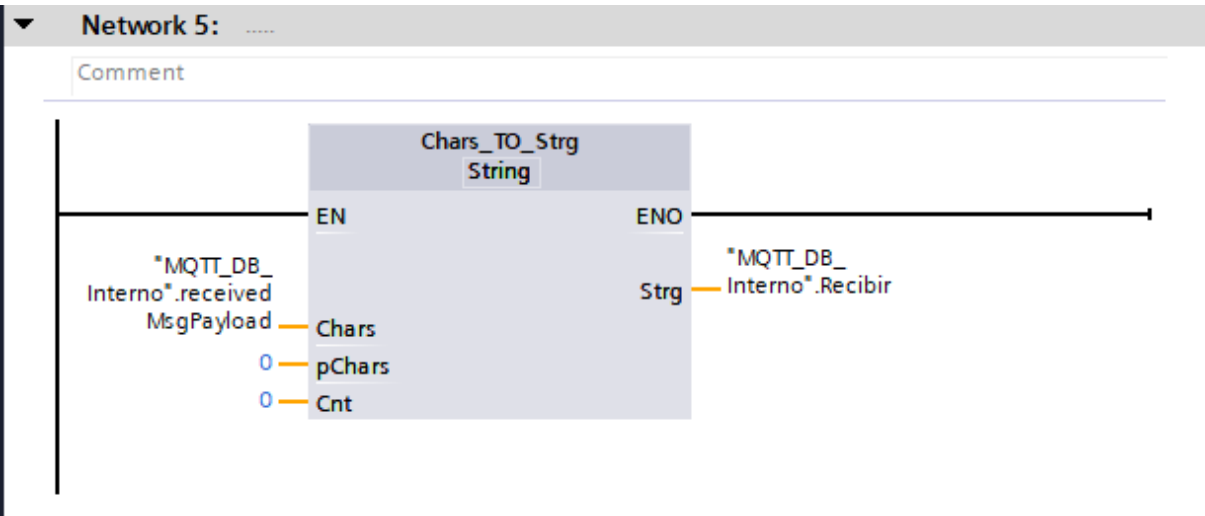
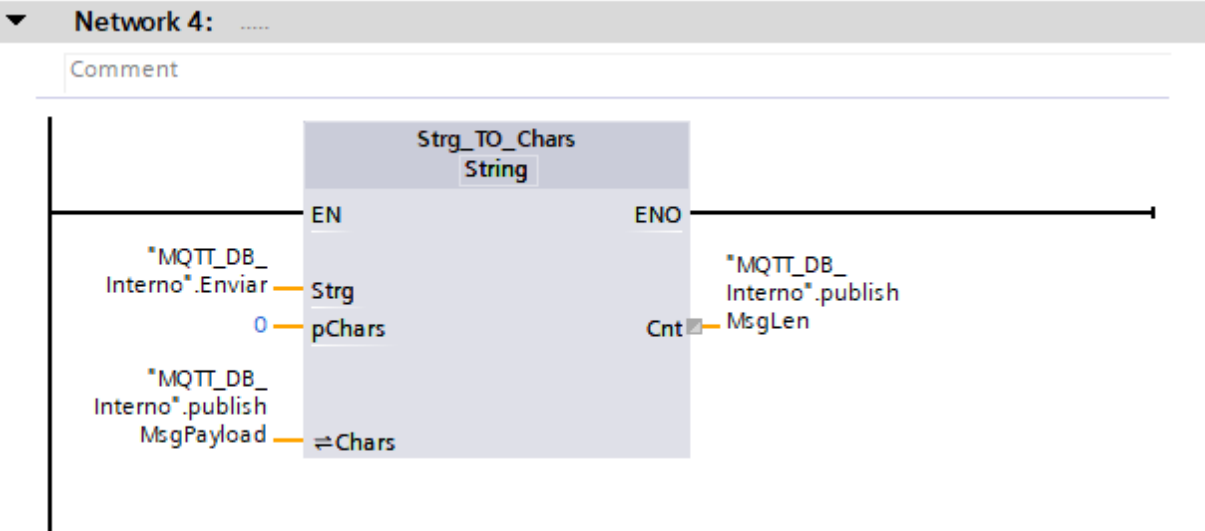
Tras instalarla buscamos dentro de la carpeta de la libreria la carpeta de “LMQTT” y dentro de la carpeta “LMQTT_Client” encontraremos el bloque FB de cliente MQTT:



Una vez puesto el bloque solo hay que relacionar las entradas y salidas del bloque con sus contrapartes del DB como se ve a continuación:



También se tendrán que añadir 2 segmentos las como se muestran a continuación:



También habrá que crear por arriba un segmento como este:



A continuación se deja una tabla que explica la función de cada apartado del bloque:

Nombre	Tipo	Tipo de Data	Comentario
enable	Input	Bool	TRUE: La conexión al MQTT Broker está establecida y mantenida FALSE: La conexión está rota.
publish	Input	Bool	Publica "publishMsgPayload" to "mqttTopic" con "retain" y "qos".
subscribe	Input	Bool	Subscribe a "mqttTopic" con "qos".
unsubscribe	Input	Bool	Desubscribe de "mqttTopic".
retain	Input	Bool	TRUE:La data es enviada con la flag "retain" FALSE:La data es enviada sin la flag "retain".
qos	Input	USInt	Calidad del servicio (Quality of Service) 0: El mensaje es enviado o suscrito con QoS 0 1: El mensaje es enviado o suscrito con QoS 1 2: El mensaje es enviado o suscrito con QoS 2 (El FB no soporta QoS 2 para suscripciones).
publishMsgLen	Input	UDInt	Tamaño actual de la data válida en el parámetro array de "publishMsgPayload".
willMsgLen	Input	UInt	Tamaño actual de la data válida en el parámetro array de "willMsgPayload".
timeOut	Input	Time	Parámetro opcional para configurar el tiempo de monitorización. Después de que el tiempo pase, el "trabajo" actual será considerado como fallido.
valid	Output	Bool	TRUE: El FB ejecuta sus funciones sin errores.
done	Output	Bool	TRUE: El "trabajo" actual (publish/subscribe/desubscribe) fue ejecutado correctamente.
busy	Output	Bool	TRUE: El FB está ocupado.
error	Output	Bool	TRUE: Un error ha ocurrido. Si "busy" es TRUE al mismo tiempo, el bloque intenta corregir el error por sí mismo.
status	Output	Word	Status y códigos de error en el pdf del final del documento (capítulo 5.2.2)
diagnostics	Output	"typeDiagnostics"	Información diagnósticos avanzados en el pdf del final del documento (capítulo 12.2)
receivedMsgStatus	Output	USInt	Indica por un ciclo al mismo tiempo cuando un mensaje nuevo a sido recibido (suscripción): 0: No hay mensajes nuevos recibidos. 1:Nuevo mensaje válido recibido. 2:Nuevo mensaje recibido, pero el mensaje es invalido o la data recibida es más grande que la área de memoria del tópico o el receptor de mensajes.

RecivedMsgDataLen	Output	UDInt	Numero de data valida en el "recivedMsgPayload"en parámetros de bytes en array.
connParam	InOut	"L MQTT_type ConnParam"	Parámetro al que establecer una conexión al MQTT Broker.
clientIdentifier	InOut	WString	Identificador de cliente usado cuando estableces una conexión
username	InOut	WString	Opcional:Nombre de usuario para establecer conexión
password	InOut	WString	Opcional:Contraseña para establecer conexión
willtopic	InOut	WString	Opcional: Tópico al cual el mensaje "LastWill" es enviado.
willMsgPayload	InOut	Array [*] of bytes	Opcional: Mensaje enviado como "LastWill".
mqttTopic	InOut	WString	Tópico usado para publicar/suscribir/desuscribir.
publishMsgPayload	InOut	Array [*] of bytes	Mensaje que es transmitido como data de usuario al publicar.
recivedTopic	InOut	WString	Tópico suscrito al cual un mensaje se ha recibido.
recivedMsgPayload	InOut	Array [*] of bytes	Data de usuario recibida en un mensaje del tópico suscrito.

Una vez hecho esto solo nos queda definir el largo de los caracteres del mensaje y el mensaje que queremos enviar.

Si se ha hecho todo correctamente el resultado el el mosquitto _sub sería el siguiente:

```
C:\Users\TecnicoCAM>cd C:\Program Files\mosquitto
C:\Program Files\mosquitto>mosquitto_sub -d -h 192.168.250.101 -Topico
Error: Unknown option '-Topico'.

Use 'mosquitto_sub --help' to see usage.

C:\Program Files\mosquitto>mosquitto_sub -d -h 192.168.250.101 -t Topico
Client null sending CONNECT
Client null received CONNACK (0)
Client null sending SUBSCRIBE (Mid: 1, Topic: Topico, QoS: 0, Options: 0x00)
Client null received SUBACK
Subscribed (mid: 1): 0
Client null received PUBLISH (d0, q0, r0, m0, 'Topico', ... (0 bytes))
Client null sending PINGREQ
```

También dejamos aquí abajo un pdf en el que se explica todo con más detalle

<https://mega.nz/folder/hygSxYIZ#nCYbpJVOV0Q0RaLuluCMSw>