

在 TIA Portal 中配置 1513R-1 PN 作为控制器，并配置 MQTT（例如，一个支持 OPC UA 的第三方设备或软件）与它通信。这里假设 MQTT 作为 OPC UA 客户端，1513R-1 PN 作为 OPC UA 服务器。

步骤：

1. 在 TIA Portal 中创建项目并添加 1513R-1 PN。
2. 配置 1513R-1 PN 的 OPC UA 服务器功能。
3. 设置 OPC UA 服务器的参数（如端口、安全策略等）。
4. 创建数据块和变量，并设置这些变量为 OPC UA 可见。
5. 编译并下载项目到 PLC。
6. 在 MQTT 中配置连接，添加 OPC UA 服务器端点，并订阅/读取变量。

注意：1513R-1 PN 的固件版本需要支持 OPC UA 服务器功能（通常需要固件版本 V2.0 或以上）。

下面详细说明：

第一步：创建项目和硬件组态

- 打开 TIA Portal，创建新项目。
- 添加新设备，选择控制器->SIMATIC S7-1500->CPU 1513R-1 PN，注意选择正确的固件版本（至少 V2.0）。

第二步：激活和配置 OPC UA 服务器

- 在设备视图中，选中 CPU，然后在属性窗口中，找到“OPC UA”->“服务器”配置。
- 勾选“激活 OPC UA 服务器”选项。
- 设置服务器端口，默认是 4840。
- 配置安全策略：可以选择无（None）、基本 128Rsa15、基本 256Sha256 等。根据安全需求选择，测试时可以选择无，但生产环境建议使用安全策略。

第三步：设置用户认证（可选）

- 在“用户管理”中，可以添加用户并设置权限。如果不需要，可以使用匿名访问。

第四步：创建数据并发布到 OPC UA

- 在程序中创建一个数据块（例如，全局数据块），并添加一些变量（如 Bool、Int、Real 等）。
- 在数据块的属性中，勾选“OPC UA”下的“公开”选项，使整个数据块在 OPC UA 地址空间中可见。或者，可以单独为每个变量设置 OPC UA 可见性。

第五步：编译和下载

- 编译项目，确保没有错误。
- 将项目下载到 PLC。

第六步：在 MQTT 中配置 OPC UA 客户端连接

- 打开 MQTT 软件（假设它支持 OPC UA 客户端功能）。
- 添加一个新的 OPC UA 连接，输入 PLC 的 IP 地址和端口（例如：opc.tcp://<PLC_IP>:4840）。
- 如果设置了安全策略和用户认证，需要在 MQTT 中相应配置。
- 连接后，浏览 OPC UA 服务器地址空间，找到之前发布的变量，并订阅或读写它们。

注意：如果网络有防火墙，需要确保 4840 端口（或自定义端口）是开放的。

另外，如果 MQTT 是作为 MQTT 客户端，而 1513R-1 PN 没有直接提供 MQTT 服务器功能，那么可能需要一个中间件（如 MQTT 网关）来将 OPC UA 数据转换为 MQTT。这里我们假设 MQTT 是 OPC UA 客户端。

如果 MQTT 是 MQTT 客户端，那么我们需要在 1513R-1 PN 和 MQTT 之间部署一个 OPC UA 到 MQTT 的网桥（例如，使用 Node-RED 或专业的工业网关软件）。这种情况下，1513R-1 PN 作为 OPC UA 服务器，网桥作为 OPC UA 客户端从 PLC 读取数据，然后转换为 MQTT 协议发布，MQTT 再作为 MQTT 客户端订阅这些数据。

但是，根据问题，我们假设是直接 OPC UA 通信。

下面是一个简化的步骤图：

TIA Portal 配置：

- 添加 CPU 1513R-1 PN
- 激活 OPC UA 服务器
- 设置端口和安全策略
- 创建数据块并设置为 OPC UA 可见
- 下载到 PLC

MQTT 配置（作为 OPC UA 客户端）：

- 新建连接，输入端点地址：opc.tcp://[PLC_IP]:4840
- 配置安全策略（与 PLC 一致）
- 浏览变量并添加订阅

如果通信成功，MQTT 将能够读取和写入 PLC 中的变量。

注意：如果遇到连接问题，请检查以下事项：

- PLC 和 MQTT 所在的电脑是否可以互相 ping 通。
- 防火墙设置是否阻止了 4840 端口。
- OPC UA 服务器设置的安全策略和用户认证是否与客户端匹配。

以上就是一个基本的配置指南。

下面提供一份关于 **MQTT (通常指 MQTT 协议) 与西门子 S7-1513R-1 PN 在 TIA Portal (博图) 中的通讯配置指南**。
与标准的 OPC UA 不同, MQTT 是一种基于发布/订阅模式的轻量级物联网协议。S7-1500 系列 (包括 1513R-1 PN) 从固件版本 V2.8 起, 开始原生集成 **MQTT 客户端** 功能, 这意味着 PLC 可以直接向 MQTT 代理 (服务器) 发布消息或订阅主题, 而无需额外的网关。

架构原理

在开始配置前, 理解其通信架构至关重要:

- **S7-1513R-1 PN 作为 MQTT 客户端。**
- 一个 **MQTT 代理**, 例如 **Mosquitto, EMQX, HiveMQ** 或云服务 (如 AWS IoT Core, 阿里云物联网平台)。它负责消息的路由。
- **TIA Portal** 是进行 PLC 编程和通信配置的工程软件。

通信流程: PLC 将特定数据 (如 DB 块中的数据) 作为消息**发布**到代理的某个主题, 或者从代理的特定主题**订阅**消息并写入 PLC 数据区。

配置步骤

以下是在 TIA Portal V17 或更新版本中配置 S7-1513R-1 PN 作为 MQTT 客户端的详细步骤。

第一步: 准备工作与硬件组态

1. **确认固件版本:**
 - 确保您的 CPU 1513R-1 PN 的固件版本**至少为 V2.8** (强烈建议使用最新版本)。这是支持原生 MQTT 客户端功能的最低要求。
 - 在 TIA Portal 中创建项目时, 务必选择正确版本的 CPU。
2. **添加硬件并配置网络:**
 - 在 TIA Portal 中创建新项目, 并添加设备“SIMATIC S7-1500” -> “CPU 1513R-1 PN”。
 - 进入“设备视图”, 为 CPU 分配正确的 IP 地址 (例如 192.168.0.1)。
3. **安装 MQTT 服务器证书 (如果服务器使用 TLS/SSL):**
 - 如果您的 MQTT 代理使用了自签名证书, 为了建立安全的 TLS 连接, 您需要先将代理的 CA 证书上传到 PLC。
 - 在项目树中展开您的 PLC, 找到“**证书管理**”。
 - 将您的 MQTT 代理的 CA 证书 (.crt 或 .der 格式) **拖拽**到“受信任的根证书”文件夹中。

第二步: 配置 MQTT 连接参数 (“连接管理”)

1. **打开连接管理:**
 - 在项目树中, 进入 PLC 的“**设备组态**”。
 - 在属性窗口中, 导航到“**MQTT**” -> “**连接管理**”。
2. **创建新连接:**
 - 点击“添加”按钮, 创建一个新的 MQTT 连接。
 - 您可以为连接命名, 例如 Connection_to_MQTT_Broker。
3. **配置连接详情:**
 - **常规:**
 - **客户端 ID:** 为您的 PLC 设置一个在该 MQTT 代理中唯一的 ID, 例如 S7-1513R-1。
 - **代理地址:** 输入 MQTT 代理的 IP 地址或域名, 例如 192.168.0.100。
 - **端口:** 默认为 1883 (非加密) 或 8883 (TLS 加密)。
 - **连接机制:**
 - **保持连接时间:** 默认为 60 秒。客户端会在此时间间隔内发送心跳包以保持连接。
 - **清除会话:** 通常勾选。断开重连后不保留之前的会话状态。
 - **最后意愿:** 可选项。配置当 PLC 异常断开时, 代理需要发布的最后一条消息。
4. **配置安全设置:**
 - 如果您的代理需要身份验证, 在“用户身份验证”部分输入**用户名和密码**。
 - 如果使用 TLS 加密 (强烈推荐用于生产环境):
 - 在“传输层安全”中, 选择“**使用 TLS**”。
 - 在“TLS 配置”中, 选择您之前导入的受信任根证书。

第三步: 定义 MQTT 通信任务 (“发布/订阅”)

这是核心步骤, 定义了 PLC 要发送和接收什么数据。

1. **创建数据块:**
 - 创建一个全局数据块 (例如 MQTT_Data), 并在其中定义您要通过 MQTT 传输的变量结构。
 - 例如:

```
pascal
"MQTT_Data" (DB100)
{
    Temperature : Real;
    Pressure : Int;
    Setpoint : Real;
    Motor_Status : Bool;
}
```

2. **配置“发布”任务 (PLC -> 代理):**

- 在“连接管理”中，选中您创建的连接，然后在下方的“发布”表中点击“添加”。
- **主题：** 输入发布主题，例如 factory1/line1/sensor_data。
- **有效负载：** 点击“...”按钮，选择数据源。这里您可以选择整个数据块 MQTT_Data，或其中的某个特定变量。
- **内容类型：** 可以选择 JSON，这样 PLC 会将数据块内容自动转换为 JSON 格式发布，便于其他系统解析。
- **发布间隔：** 设置定时发布的周期，例如每 5 秒发布一次。

3. 配置“订阅”任务 (代理 -> PLC)：

- 在“订阅”表中点击“添加”。
- **主题：** 输入要订阅的主题，例如 factory1/line1/commands。
- **有效负载：** 点击“...”按钮，选择一个 PLC 中的数据区 (如 MQTT_Data.Setpoint) 来接收来自该主题的消息。
- 当任何客户端向 factory1/line1/commands 主题发布消息时，消息内容会被自动写入到 PLC 的 Setpoint 变量中。

第四步：编程与下载

1. 编译项目：

- 确保所有配置无误，编译整个项目。

2. 下载到 PLC：

- 将硬件配置和程序下载到 S7-1513R-1 PN 中。
- **重要：** 下载完成后，需要将 PLC 从“STOP”模式切换到“RUN”模式，MQTT 客户端连接才会开始尝试建立。

测试与诊断

1. 使用 MQTT 客户端工具测试：

- 在您的电脑上使用一个 MQTT 客户端工具 (如 **MQTT Explorer**, **MQTT.fx**)。
- 连接到同一个 MQTT 代理。
- **订阅** PLC 发布的主题 (如 factory1/line1/sensor_data)，查看是否能收到 PLC 定时发送的 JSON 数据。
- **向** PLC 订阅的主题 (如 factory1/line1/commands) 发布一个消息 (例如 {"Setpoint": 75.5})，查看 PLC 中的 Setpoint 变量是否更新。

2. PLC 诊断：

- 在线连接到 PLC 后，在“连接管理”中可以看到 MQTT 连接的状态 (例如“已建立”)。
- 在“诊断缓冲区”中可以查看连接建立失败等错误信息。

优势总结

通过这种原生集成的方式，S7-1513R-1 PN 可以：

- **直接上云/上平台：** 无需额外硬件网关，直接与云端的 IoT Hub 或第三方 SCADA/MES 系统通信。
- **降低系统复杂度：** 减少了中间转换环节，提高了系统可靠性和实时性。
- **标准化数据交换：** 使用 JSON 等通用格式，便于与 IT 系统集成。

这份指南涵盖了从硬件配置到通信测试的全过程。实际应用中，请根据您的网络环境和 MQTT 代理的具体要求调整安全设置和主题结构。