

[Robótica] Comunicaciones: Conceptos generales



Contenidos

1. Documentos PDF

1. Conexión remota

2. Protocolos de comunicación para integración de robots y PLCs (2023)

3. TFG - Desarrollo de un entorno con HMI y PLC Siemens para tareas de paletizado de un robot industrial

2. Manuales ABB

1. [Parámetros del sistema: Manual de referencia técnica](#) Página 38 y siguientes.

- De forma predeterminada, el puerto X5 (LAN 3) se configura como una red aislada, lo cual permite la conexión del controlador de robot a una red externa. Típicamente, los PLC que controlan varios controladores de robot se conectan a LAN 3.

2. [Robotware: Opciones y funciones \(2023\)](#)

- Apartado **8.5 Socket Messaging [616-1]**

3. [Descripción general de RAPID: Manual de referencia técnica \(2021\)](#)

- Apartado **1.19 Funciones para cadenas de caracteres**

4. [Instrucciones, funciones y tipos de datos de RAPID: Manual de referencia técnica](#)

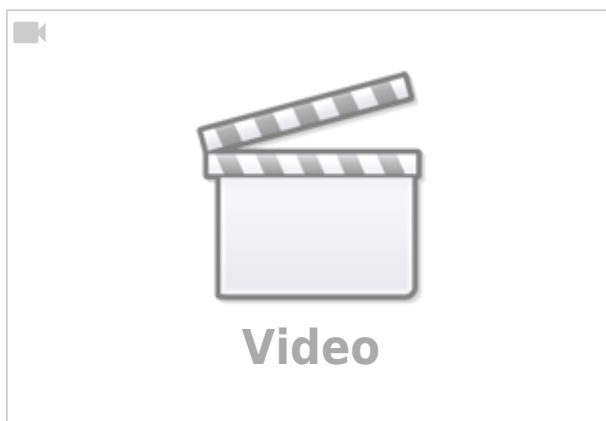
- En la sección de **Tipos de datos** buscar la palabra socket para comprobar las múltiples instrucciones con esa palabra.



Vídeos

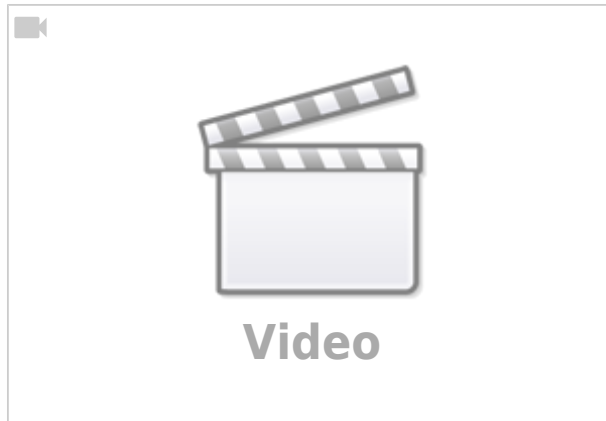
1

[Instrumentación - Protocolos de Comunicación Industrial](#)



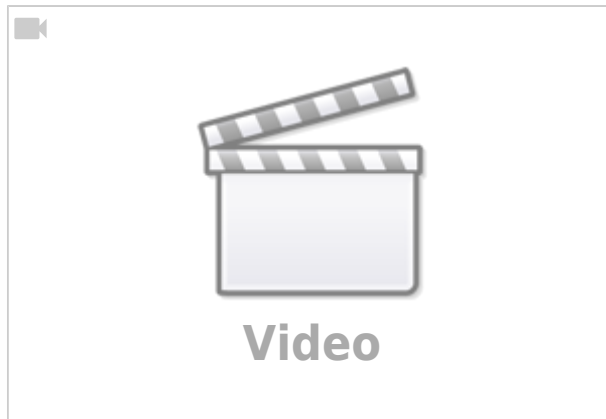
2

[ABB RobotStudio con Siemens por Socket](#) / Robot ABB como servidor ✓



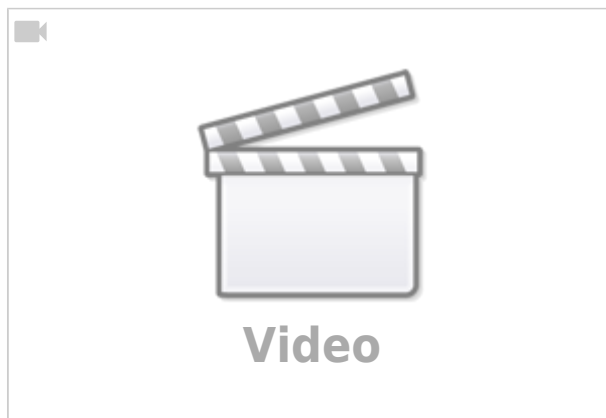
3

[Comunicación RobotStudio Protocolo TCP/IP \(sockets\) / Robot ABB como servidor ✓](#)



4

[Open User Communication \(TCON, TSEND, TRCV\) con PLC S7-1200, TIA PORTAL y Python con protocolo TCP / PLCs Siemens ✓](#)



Anotaciones sobre el vídeo 4 de PLCs

-  En este vídeo se usa para el servidor un PC con un programa Python pero, excluyendo esa parte, explica muy bien los pasos a realizar en el TIA Portal para configurar el PLC Siemens como cliente. Es por ello que con este vídeo pretendemos resolver la mitad del problema: **La configuración del lado cliente de un PLC Siemens S7-1200**

para luego poder comunicarse con un servidor, que en nuestro caso será el robot de ABB.  Las configuraciones están indicadas para el TIA Portal v15.

- **Material adjunto al vídeo**

1. Proyecto para TIA Portal v15
2. Programa Python TCPOpenComm.py

para usar el PC como cliente:

[TCPOpenComm.py](#)

```
import socket
import sys
socketc=socket.socket(socket.AF_INET, socket.SOCK_STREAM) #Crear Socket
server_address = ('192.168.0.241', 2020) #Asignar IP y puerto para servidor
socketc.bind(server_address) #TCP 2020
print('starting up on {} port {}'.format(*server_address))
socketc.listen(1)
comando='1'
while True:
    # Wait for a connection
    print("Esperando conexion")
    connection, client_address = socketc.accept()
    try:
        print("Conexion desde IP...", client_address)

        # Recibir datos en chunks (segmentos)
        while True:
            data = connection.recv(4)
            print("recibido {!r}".format(data))
            if data:
                print("Retornando datos al cliente")
                connection.sendall(comando)
            else:
                print("No hay datos desde cliente...", client_address)
                break

    finally:
        # Cerrar la conexion
        connection.close()
```

From:
<https://euloxio.myds.me/dokuwiki/> - Euloxio wiki

Permanent link:
https://euloxio.myds.me/dokuwiki/doku.php/doc:tec:robot:com_general:inicio?rev=1765200595

Last update: 2025/12/08 14:29

