

Versuch 2 – Sensordaten mit Loxone Config

In diesem Versuch soll der Umgang mit Sensordaten innerhalb von Loxone Config erlernt werden. Dabei wird sowohl die Nutzung von Sensorsignalen für Steuerungs- und Regelungsaufgaben als auch die Wandlung und Umrechnung von Daten behandelt werden.

Jede gelöste Teilaufgabe muss immer zuerst getestet werden, bevor die nächste bearbeitet wird. Sie sollten sich an die vorgegebene Reihenfolge halten und bei jeder Teilaufgabe mit der Markierung **→Betreuer** einem Betreuer das Programm zeigen und das Projekt unter einem anderen Namen fortführen. Geben Sie dem Projekt einen prägnanten Namen.

Loxone Miniserver: Kommunikation über UDP

Im letzten Versuch haben Sie bereits virtuelle Ausgänge kennengelernt. Diese eignen sich jedoch nicht nur zur Darstellung von Werten in der Visualisierung des Webinterfaces oder der Handy-App, sondern ermöglichen auch die Kommunikation des Miniservers mit anderen netzwerkfähigen Geräten (z.B. PC, Server, Musikanlagen, etc.) über die Protokolle *TCP*, *UDP*, *HTTP* und *WO*.

Als einfaches Beispiel der Kommunikation über das Netzwerk soll sich nun näher mit dem *UDP*-Protokoll auseinandergesetzt werden. *UDP* (User Datagram Protocol) gehört zur Familie der Internetprotokolle und stellt eine einfache Möglichkeit dar, um in IP-basierten Netzwerken Daten zwischen zwei Geräten auszutauschen. Es gehört zu den sogenannten verbindungslosen Protokollen, d. h. zur Datenübertragung wird zwischen zwei Geräten kein Kanal mit definierten Übertragungsbedingungen (wie z.B. bei *TCP*) geöffnet, sondern die zu sendenden Daten nur mit einer Empfangsadresse (IP und Portnummer) versehen. Dies vereinfacht die zur Kommunikation nötigen Befehle, jedoch kann nicht garantiert oder überprüft werden, ob gesendete Daten auch beim Empfänger ankommen.

Um vom Miniserver aus *udp*-Daten zu versenden muss zuerst in Loxone Config über den Reiter „Peripherie“ ein neuer virtueller Ausgang hinzugefügt werden. Zusätzlich muss im Eigenschaftsfenster des virtuellen Ausgangs im Feld „Adresse“ folgendes eingetragen werden:

„/dev/udp/(IP des Empfängers)/(Portnummer)“

Der in Abbildung 1 gezeigte virtuelle Ausgang sendet Daten über den Port 7000 an die IP-Adresse 192.168.1.10. Haben in einer Zahl der IP-Adresse alle Bits den Wert „1“, dann sendet der virtuelle Ausgang an alle Geräte in diesem Datenbereich. Mit einer Empfänger-IP von 172.21.71.255 sendet der Miniserver also Daten an jeden PC des Poolraums im Praktikum. Mit einer Empfänger-IP von 255.255.255.255 würde der Miniserver entsprechend Daten an das gesamte Internet senden, dies wird in der Praxis durch die Beschränkung von Netzwerken auf eine entsprechende Subnetzmaske aber verhindert.

Wurde der virtuelle Ausgang korrekt konfiguriert, dann muss zum Senden von Daten für jeden Befehl noch über den Reiter „Peripherie“ ein „*Virtueller Ausgang Befehl*“ erstellt, entsprechend konfiguriert und wie bereits bekannt in die Programmierung eingefügt werden. Die Eingabe im Feld „*Befehl bei EIN*“ legt dabei fest welche Daten gesendet werden.

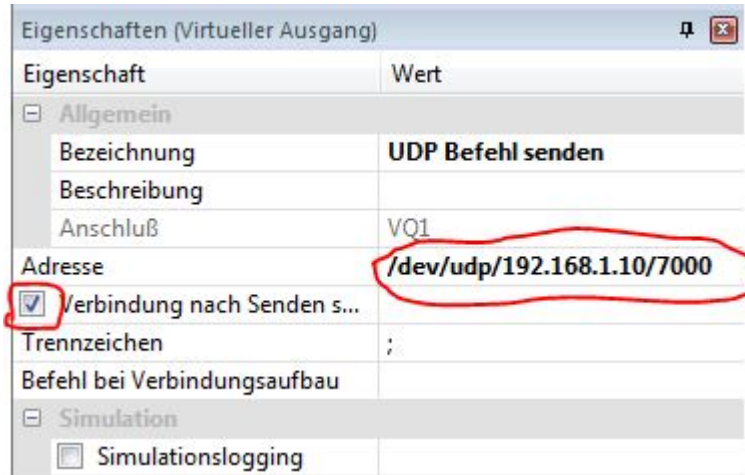


Abbildung 1: Einstellungen virt. Ausgang für udp-Kommunikation

Der in Abbildung 2 gezeigte Befehl ist als Analogausgang konfiguriert. Er sendet bei jeder Änderung des an seinem Eingang anliegenden Wertes den String „Sensorwert ANLIEGENDER WERT MIT DREI NACHKOMMASTELLEN“ an die vorher definiert Empfängeradresse. Ebenso ist auch die Konfiguration als Digitalausgang möglich. Ein Digitalausgang sendet bei Impuls am Eingang den gewählten Datenstring im Feld „Befehl bei EIN“.

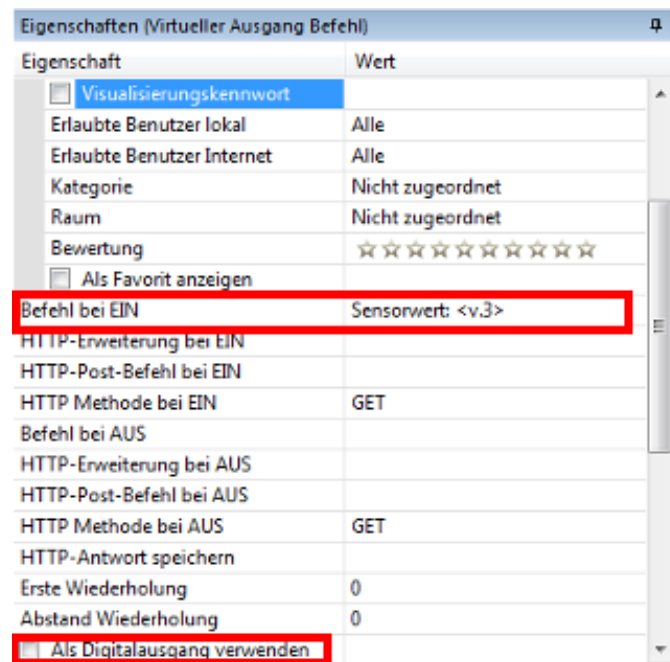


Abbildung 2: UDP-Kommunikation: Befehl

Analog zum Senden von Daten können über einen virtuellen Eingang auch Daten empfangen werden. Hierzu muss bei selektiertem Eintrag „*Virtuelle Eingänge*“ im Peripheriebaum ein „*Virtueller UDP Eingang*“ mit entsprechendem „*Virtuellem UDP Eingang Befehl*“ erstellt werden. Die IP-Adresse und Port des Senders werden im Eigenschaftsfenster des virtuellen Eingangs eingestellt. Mithilfe des *UDP-Monitors* können Befehle auch eingelesen werden.

Aufgabe 1: Temperaturregelung mit Loxone Config

In dieser Aufgabe soll mithilfe der Heiz-/Kühleinheit oben links im Deckel des Experimentierplatzes eine Temperaturregelung realisiert werden. Die Heiz-/Kühleinheit besteht aus einem Lüfter (Miniserver: Digitaler Ausgang 7), einem Heizwiderstand (Digitaler Ausgang 6) und einem 1-Wire Temperatursensor, die alle auf dem gleichen Kühlkörper montiert und damit in thermischem Kontakt sind. Die vom 1-Wire Temperatursensor gemessene Temperatur dieses Systems soll nun durch den Einsatz des Heizers und des Lüfters auf eine konstante Solltemperatur geregelt werden.

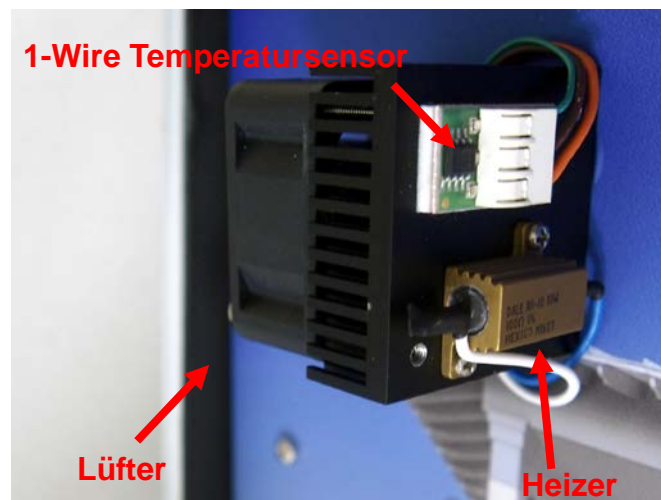


Abbildung 3: Heiz-/Kühleinheit

Die Umsetzung einer Temperaturregelung ist in Loxone Config sowohl durch eine geeignete Verschaltung grundlegender Logikelemente als auch durch den Einsatz ausgearbeiteter Programmbausteine möglich. Diese Programmbausteine finden Sie unter dem Menüpunkt „Regler und Steuerungen“ im Reiter „Programm“ von Loxone Config. Im Rahmen dieser Aufgabe soll sich besonders auf die Programmbausteine „2-Punkt-Regler“ und „3-Punkt-Regler“ konzentriert werden.

Zwei-Punkt Regler

Einfacher Sollwertregler mit Hysterese.

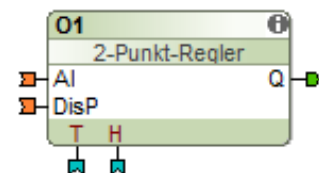
AI: Analogeingang der Regelgröße.

T: Eingang für den Sollwert.

H: Eingang für den Hysteresewert.

Q: Regelausgang; Q wird eingeschaltet, wenn $AI < (T-H)$ und ausgeschaltet wenn $AI > (T+H)$.

DisP: Digitaleingang, deaktiviert den Regelausgang Q



Drei-Punkt-Regler

Nicht-stetiger Regler mit drei möglichen Ausgangszuständen.

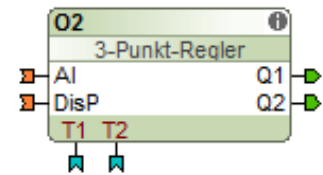
AI: Analogeingang der Regelgröße.

T1: Schaltschwelle 1

T2: Schaltschwelle 2

Q1/Q2: Regelausgänge; Q1 wird eingeschaltet wenn $AI < T1$, Q2 wird eingeschaltet wenn $AI > T2$. Für $T1 < AI < T2$ sind beide Ausgänge deaktiviert.

DisP: Digitaleingang, deaktiviert den Regler



Aufgabenstellung

Im ILIAS-Portal finden Sie im Ordner des Versuchs 2 eine Programmierungsvorlage für Loxone Config, die bereits alle für den Versuch nötigen Einstellungen für Extensions, Sensoren und UDP-Kommunikation enthält. Wählen Sie die zu Ihrem Experimentierplatz passende Vorlage. Die Benennung der Dateien erfolgt nach dem Muster „Miniserver_(Letzte Zahl der IP-Adresse)“.

- Erstellen Sie eine Temperaturregelung mithilfe der Heiz-Kühleinheit und dem Zwei-Punkt Regler. Binden Sie den 1-Wire Temperatursensor dazu in ihre Programmierung ein. Die Solltemperatur sowie die Hysterese sollen dabei im Webinterface einstellbar sein. Der mögliche Wertebereich der Solltemperatur muss auf 26 °C bis 30 °C und der Hysterese auf 1 °C bis 0 °C beschränkt werden.

Die vom 1-Wire-Sensor gemessene Ist-Temperatur soll zusätzlich über UDP an einen PC gesendet werden. Laden Sie aus dem Ilias-Portal das Hilfsprogramm „V2_1Wire_TemperaturMonitor_UDP.exe“ herunter und zeichnen Sie damit mehrere Regelperioden der Regelung für zwei verschiedene Hysteresewerte auf.

Beantworten Sie folgende Fragestellungen:

- Welchen Einfluss hat der Hysteresewert auf Ihre Regelung?
- Welche Probleme erkennen Sie beim Einsatz des Zwei-Punkt-Reglers?

Hinweis:

Mit einem Rechtsklick auf den Verlaufsgraph des Hilfsprogramms und Auswahl der entsprechenden Option im sich öffnenden Menü können die aufgezeichneten Daten exportiert werden (z.B. nach Microsoft Excel). Dies kann den Vergleich zwischen verschiedenen Einstellungen erleichtern.

→Betreuer

- Ersetzen Sie nun den Zwei-Punkt-Regler in Ihrer Programmierung durch den 3-Punkt-Regler und binden Sie zusätzlich zum Heizer auch den Lüfter in ihre Regelung ein. T1 und T2 soll wieder über das Webinterface einstellbar sein. Wählen Sie den Wertebereich entsprechend einer Solltemperatur von 28 °C.

Zeichnen Sie wieder mehrere Regelperioden mit dem Programm „V2_IWire_TemperaturMonitor_UDP.exe“ auf. Welche Unterschiede im Verhalten sind beim 3-Punkt-Regler im Vergleich zum 2-Punkt-Regler erkennbar?

→Betreuer

Aufgabe 2: Sensordaten mit Loxone Config

In diesem Versuch werden Sie die Programmierung ihres Miniservers um verschiedene Routinen erweitern, die in späteren Versuchen zur Verarbeitung und Darstellung von Messwerten verschiedener Sensorsignale benötigt werden. Die Messwerte werden später über die RS232-Extension des Miniservers empfangen und verarbeitet. Da über die RS232-Extension noch keine Daten übertragen werden, wird in diesem Versuch die Datenübertragung durch das Ausführen des Hilfsprogramms „V2_SensorSimulator_UDP.exe“ auf ihrem PC simuliert. Dieses sendet über das bereits bekannte UDP-Protokoll Sensordaten an ihren Miniserver, die dann verarbeitet und wieder über das Netzwerk zur Darstellung an den PC zurückgeschickt werden sollen.

Die vom PC empfangenen werden über den virtuellen Eingang „Simulierter Sensorwert“ im Unterordner „UDP Empfangen“ in die Programmierung eingebunden. Sie liegen in der Form eines 16-Bit Datenwortes vor. Das Datenwort besteht dabei aus einem gewandelten Sensorwert mit 12 Bit in 2er-Komplementdarstellung und einem Steuercode in den höchstwertigen vier Bit, der den Sensor identifiziert.

Bit[15...12]	Bit [11...0]
Steuercode	12 Bit Messwert vom A/D-Wandler in
0001: Sensor 1	2er Komplementdarstellung
0010: Sensor 2	(Bit[11] → Vorzeichenbit)
0011: Sensor 3	

Tabelle 1: Datenwort

Aufgabenstellung

- 1) Fügen Sie auf der Seite „Sensorerkennung“ die nötigen Programmbausteine hinzu, um aus dem simulierten 16-Bit Sensorwert die Sensorkennzahl und den 12-Bit Sensorwert zu ermitteln.
- 2) Erstellen Sie auf der Seite „Umrechnung Sensorwert“ eine Routine zur Wandlung des 12-Bit Sensorwerts vom Zweierkomplement-Format in eine reelle Zahl. Testen Sie ihre Umrechnung mit dem Hilfsprogramm „V2_SensorSimulator_UDP.exe“.

→Betreuer

→Betreuer