

Sensoren und Aktoren mit MICA verbinden



Pushing Performance

People | Power | Partnership

Eines der hervorstechendsten Merkmale des MICA-Ökosystems ist die hohe Flexibilität beim Anschluss externer Geräte und die einfache Datenerfassung vor Ort. In diesem Whitepaper werden einige der häufigsten Anwendungsfälle veranschaulicht. Der erste Fall ist das "Hello World" Äquivalent und zeigt die Steuerung von einfachen digitalen E/A. Nachdem einige digitale Kanäle gelesen und eingestellt wurden, wird als nächstes ein Analogeingang gelesen. Der letzte Fall widmet sich der Kommunikation mit IO-Link-Geräten.

Hardwareeinrichtung

Diese Veranschaulichung baut auf MICA Basic auf und verwendet Standard TCP/IP-basierte industrielle E/A-Feldmodule, in diesem Fall

- ein Digital-E/A-Modul
- ein Analog-E/A-Modul
- ein IO-Link Master Modul.

Die E/A-Module wurden aus der TBEN-Serie von TURCK ausgewählt. Als beispielhafte Sensoren und Aktoren wurden ein Bestückungstaster von Banner, ein PT1000-Temperatursensor und ein IO-Link-Durchflussmesser von ifm ausgewählt. Natürlich können auch E/A-Module und Sensoren von anderen Herstellern verwendet werden.

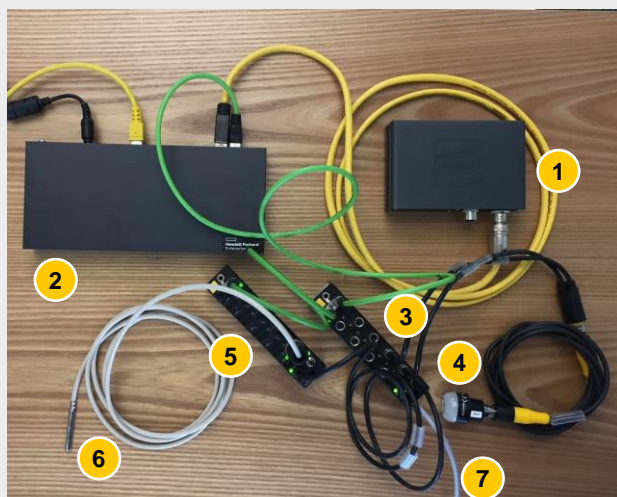


Abb. 1: Hardware-Layout mit MICA, E/A-Modulen und digitalen / analogen Sensoren / Aktoren

Abb. 1 zeigt die Hardwareeinstellung. Der MICA (①) wird über einen Power-over-Ethernet-Switch (②) mit dem ersten TURCK-Modul, einem 8-Kanal-Digital-E/A TBEN-S1-8DXP (③) verbunden. Das Pick-to-Light-Gerät (④)

wird mit dem Taster am Digitaleingang 0 und der Leuchte an Digitalausgang 1 angeschlossen. Das zweite TURCK-Modul, ein TBEN-S2-4AI mit 4 analogen Eingängen (⑤) wird mit dem Ersten verbunden. Ein PT1000-Temperatursensor (⑥) wird am ersten Anschluss angeschlossen (über die Pins 1 und 3, die Polarität spielt keine Rolle). Die TURCK-Module benötigen eine externe 24VDC-Spannungsversorgung (⑦, hier nicht dargestellt).

Einrichtung und Konfiguration der Software

Da wir zur Steuerung unserer Hardware Node-RED-Abläufe verwenden, muss der Node-RED-Container zuerst heruntergeladen und auf dem MICA installiert werden (siehe <http://www.mica-container.com/>).

Öffnen Sie zunächst die Homepage Ihres MICA in einem Browser und klicken Sie auf die Kachel *Install*. Es öffnet sich eine Dialogseite. Klicken Sie auf *Select File*, um die Node-RED-Archivdatei auszuwählen, die Sie zuvor heruntergeladen haben. Klicken Sie auf *Execute*, um den Container zu installieren.

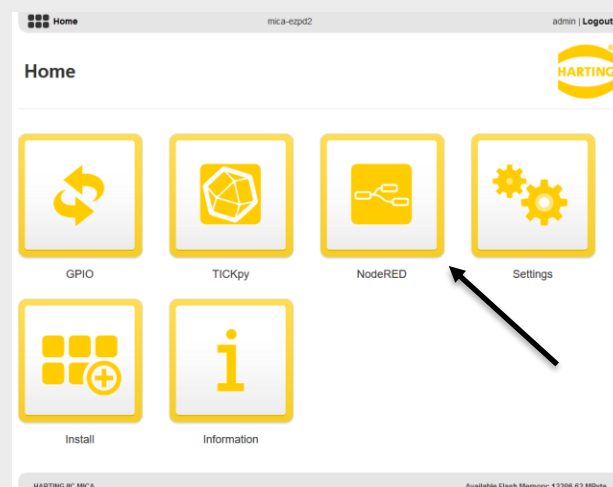


Abb. 2: MICA-Homepage mit installiertem Node-RED-Container

Sensoren und Aktoren mit MICA verbinden



Pushing Performance

People | Power | Partnership

Nach der Installation wird eine neue Kachel für den Node-RED-Container angezeigt (siehe Abb. 2). Klicken Sie mit der rechten Maustaste auf diese Kachel, um das Kontextmenü zu öffnen. Klicken Sie auf *Options*, um die Netzwerkeinstellungen des Containers zu konfigurieren. Beachten Sie, dass sich das MICA-Basissystem, der Node-RED-Container und die E/A-Module im gleichen IP-Adressbereich befinden müssen, um ordnungsgemäß zu funktionieren. Klicken Sie anschließend im Kontextmenü auf *Start App*, um den Container zu starten.

Node-RED Ablauf für den ersten Anwendungsfall

Wir werden den Informationsfluss aller Beispiele innerhalb der Node-RED-Umgebung modellieren und beginnen mit dem Anschluss eines einfachen digitalen Gerätes. Der verwendete Pick-to-Light-Taster hat nur einen einzigen Eingang bzw. Ausgang.

Klicken Sie auf die Node-RED-Kachel auf der MICA-Homepage. Die Homepage des Containers öffnet sich (siehe Abb. 3). Erweitern Sie den Abschnitt *Node-RED* und klicken Sie auf die Schaltfläche *Open in new window*, um die Benutzeroberfläche in einer neuen Registerkarte zu öffnen. Beim ersten Öffnen wird möglicherweise vorher ein Anmeldebildschirm eingeblendet. Verwenden Sie als Benutzername *admin* und als Passwort *admin*, um sich anzumelden.

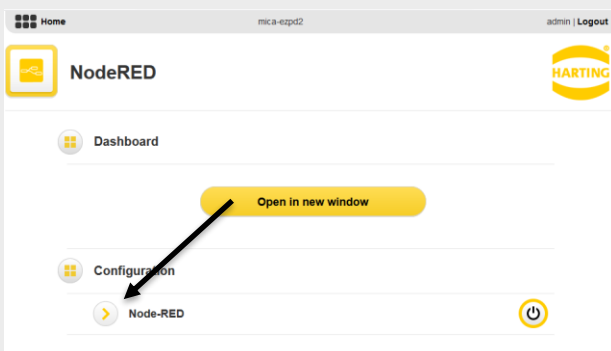


Abb. 3: Homepage des Node-RED-Containers

Es öffnet sich eine leere Arbeitsfläche (Abb. 4).

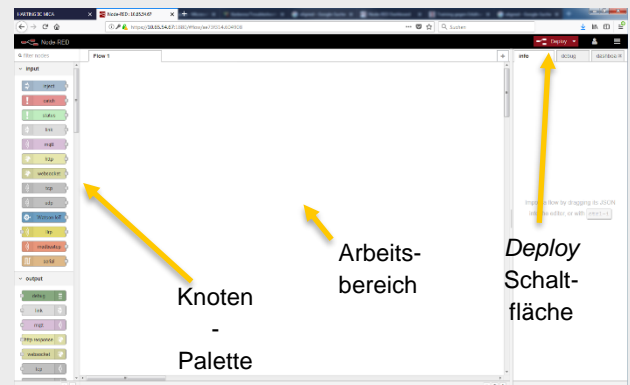


Abb. 4: Node-RED Benutzeroberfläche

Links wird eine Palette mit allen verfügbaren Knoten angezeigt, die im Arbeitsbereich angeordnet werden können. Änderungen müssen durch Betätigen der Schaltfläche *Deploy* übermittelt werden.

Um die Daten aus E/A-Modulen nutzen zu können, müssen Modbus-Verbindungen hergestellt werden. Beginnen wir mit dem digitalen E/A-Modul. Ziehen Sie einen *Modbus-Read*-Knoten und einen *Debug*-Knoten aus der Palette (Bereich *modbus* bzw. *output*) auf den Arbeitsbereich und verbinden Sie diese wie abgebildet.



Abb. 5: Lese-Verbindung zum digitalen E/A-Modul

Konfigurieren Sie als nächsten den Modbus-Knoten durch doppelklicken. Geben Sie dem Knoten einen aussagekräftigen Namen und geben Sie die Daten an, die Sie auslesen möchten: Geben Sie den auszuführenden Funktionscode sowie die Adresse und Datenmenge an. In unserem Fall wollen wir ein Eingangsregister (16 Bit, Funktionscode 4) ab Adresse null auslesen. Wenn Sie andere Hardware verwenden, müssen Sie ggf. in der Dokumentation Ihres E/A-Moduls nachschauen, wo die Eingangsdaten abgelegt werden. Stellen Sie das Abfrageintervall ein, z. B. einmal pro Sekunde. Wählen Sie abschließend den Server für die Operation aus. Da noch kein Server konfiguriert ist, klicken Sie auf die Schaltfläche mit dem Stift-Symbol, um den ersten Server einzurichten.

Sensoren und Aktoren mit MICA verbinden



Pushing Performance

People | Power | Partnership

node properties

Settings

Name: Read DI

Topic: Topic

Unit-Id:

FC: FC 4: Read Input Registers

Address: 0

Quantity: 1

Poll Rate: 1 second(s)

Delay on start:

Server: TBEN-S1-8DXP

Abb. 6: Lese-Konfiguration des digitalen E/A-Moduls

Es öffnet sich ein neues Menü (Abb. 7). Wählen Sie den Host (Ihr digitales E/A-Modul) anhand seiner IP-Adresse aus und geben Sie eine Bezeichnung ein. Verlassen Sie dieses Menü indem Sie die Schaltfläche *Add* betätigen (oder *Update*) und das darauffolgende Menü mit der Schaltfläche *Done*.

Edit modbus-client node

Name: TBEN-S1-8DXP

Type: TCP

Host: 10.10.10.57

Port: 502

TCP Type: DEFAULT

Unit-Id: 1

Timeout (ms): 1000

Reconnect timeout (ms): 2000

Log states changes:

Queue commands:

Queue delay (ms): 1

Abb. 7: Host-Konfiguration

Durch Betätigen der Schaltfläche *Deploy* werden die Änderungen übermittelt und der Ablauf gestartet. Einmal pro Sekunde wird dann eine neue Meldung mit dem aktuellen Eingangsstatus im Debug-Fenster rechts (Abb 8) eingeblendet. Sie sehen, dass die Taste, die an Pin Null angeschlossen ist, zwei Sekunden lang gedrückt wurde (Hinweis: Um zu verhindern, dass Debug-Meldungen gedruckt werden, deaktivieren Sie den Debug-Knoten mit seiner kleinen Taste auf der rechten Seite).

Sensoren und Aktoren mit MICA verbinden



Pushing Performance

People | Power | Partnership

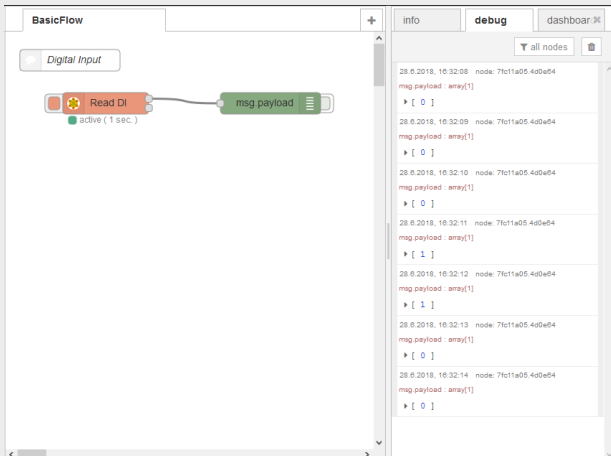


Abb 8: Debug-Ausgabe

Digitalausgang

Um Daten an das E/A-Modul zu senden, benötigen wir einen Modbus-*Write*-Knoten. Ziehen Sie einen *Write*-Knoten auf den Arbeitsbereich und verbinden Sie zwei *inject*-Knoten mit seinem Eingangsanschlüssen.

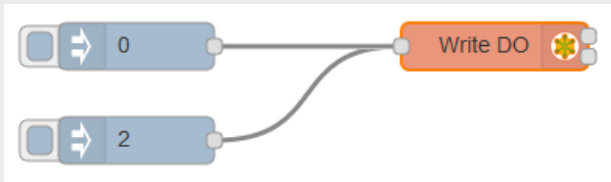


Abb. 9: Schreib-Verbindung zum digitalen E/A-Modul

Modbus-Knoten konfigurieren (siehe Abb. 10). Wählen Sie das bereits erstellte E/A-Modul als Server, Funktionscode 6, um ein einzelnes Register als Operation zu senden, und geben Sie die Adresse ein, an welche die Daten geschrieben werden sollen. In unserem Fall sollen die Ausgangsdaten an die Adresse 2048 gesendet werden. Die Adresse, die Ihr E/A-Modul hierfür verwendet, können Sie in dessen Handbuch finden. Ändern Sie den Typ *payload* der beiden *inject*-Knoten auf *number* und setzen Sie sie auf *Null* bzw. *zwei*. Zwei, d. h. das zweite Bit wird gesetzt und schaltet unseren Ausgang ein, der an Anschluss 2 angeschlossen ist, während Null alle Bits zurücksetzt und damit abschaltet. Vergessen Sie nicht, die Schaltfläche *Deploy* zu betätigen. Betätigen Sie die kleinen Schaltflächen an den *inject*-Knoten, um die Lampe der Pick-to-Light-Taste ein- und ausschalten.

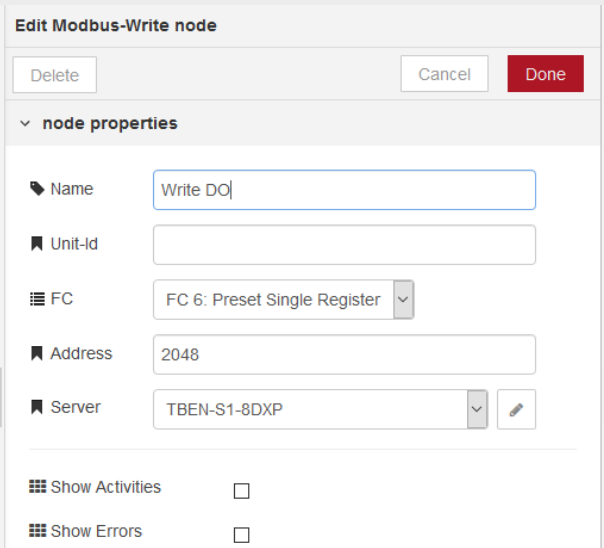


Abb. 10: Schreib-Konfiguration des digitalen E/A-Moduls

Analogeingang

Als nächstes wollen wir analoge Daten von unserem Temperatursensor lesen.

Die Vorgehensweise ist identisch mit der am Digitaleingang. Daher benötigen wir einen weiteren *Modbus-Read*-Knoten. Ziehen Sie einen *Read*-Knoten auf den Arbeitsbereich und verbinden Sie einen neuen *debug*-Knoten mit seinem Ausgang.



Abb. 11: Lese-Verbindung zum analogen E/A-Modul

Konfigurieren Sie als nächsten den Modbus-Knoten durch doppelklicken. Geben Sie dem Knoten eine Bezeichnung, wählen Sie Funktionscode vier, Startadresse null, Datenmenge eins und Abfrageintervall einmal pro Sekunde. Abschließend muss ein neuer Server für das analoge E/A-Modul initialisiert und ausgewählt werden.

Nachdem die Änderungen übernommen wurden (Schaltfläche *Deploy*), wird der Ablauf gestartet. Meldungen mit ganzzahligen Zahlen werden einmal pro Sekunde im *Debug*-Fenster auf der rechten Seite angezeigt, die der Temperatur entsprechen.

Sensoren und Aktoren mit MICA verbinden



Pushing Performance

People | Power | Partnership

Grafische Benutzeroberfläche

Dem Datenstrom, den wir für unsere Sensoren und Aktoren eingerichtet haben, eine grafische Benutzeroberfläche hinzuzufügen ist sehr einfach. Dashboards, wie in Abb. 12 dargestellt, können mit minimalem Aufwand erstellt werden.

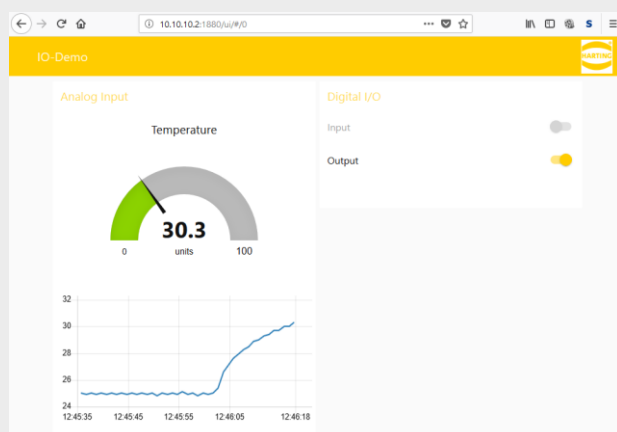


Abb. 12: Visualisierte Daten

Aus Gründen der Kürze gehen wir hier nicht weitere Details ein. Im Anhang ist jedoch ein kompletter Ablauf einschließlich einer Benutzeroberfläche aufgeführt. Kopieren Sie einfach den Text und importieren Sie ihn über das Optionsmenü oben rechts (*Import*, dann *Clipboard*) in Node-RED.

IO-Link Sensor

Als letzten Anwendungsfall integrieren wir ein IO-Link-Gerät. Dazu müssen wir ein I/O-Link-Mastermodul an die bereits angeschlossenen Module anschließen, in unserem Fall ein TURCK TBEN-4IOL. Es ist ein sehr gutes Beispiel für die Flexibilität, die der MICA-Ansatz bietet. Sie können problemlos zusätzliche E/A-Module verketteten, um weitere beliebige Geräte anzuschließen. Hier haben wir beispielhaft einen IO-Link-Durchflussmesser (ifm SD6000) ausgewählt.



Abb. 13: IO-Link-Durchflussmesser an ein IO-Link-Mastermodul angeschlossen

Der Datenfluss unterscheidet sich dieses Mal von den Vorhergehenden. Da die vom IO-Link-Master gelesenen Rohinformationen verarbeitet werden müssen, um die enthaltenen Informationen zu extrahieren, benötigen wir einen *function*-Knoten zwischen dem *Modbus-Read* und dem *Debug*-Knoten. Ziehen Sie alle auf den Arbeitsbereich und verbinden Sie sie wie unten dargestellt.



Abb. 14: Lese-Verbindung zum IO-Link-Mastermodul

Die Konfiguration des Modbus-Knotens erfolgt auf die bisher gezeigte Weise. Geben Sie dem Modul eine Bezeichnung, wählen Sie Funktionscode vier, die Startadresse ist in diesem Fall zwei (d. h. der erste Anschluss unseres IO-Link-Mastermoduls), die Datenmenge beträgt nun vier und das Abfrageintervall ist einmal pro Sekunde. Zuletzt muss ein neuer Server für das I/O-Link-Mastermodul initialisiert und ausgewählt werden.

Unser Sensor überträgt folgende Informationen: Zählwerk, Durchflussrate, Temperatur und zwei binäre Signale; kodiert in vier Registern (64 Bit). Hierzu müssen einen Blick in die IO-Link Device-Description-Datei des Sensors werfen (zugänglich über die Homepage des Herstellers), um zu sehen, wo wir die entsprechenden Informationen finden können. Platzieren Sie den folgenden Code-Abschnitt in den Inhalt des Funktionsknotens, um die Daten auszulesen:

```
const buffer = Buffer.allocUnsafe(8);  
  
buffer.writeUInt16BE(msg.payload[0], 0);
```

Sensoren und Aktoren mit MICA verbinden



Pushing Performance

People | Power | Partnership

```
buffer.writeUInt16BE(msg.payload[1], 2);
buffer.writeUInt16BE(msg.payload[2], 4);
buffer.writeUInt16BE(msg.payload[3], 6);

var newMsg = {};
newMsg.payload = {};
newMsg.payload.totalizer = buffer.readFloatBE(0);
newMsg.payload.flowrate =
  buffer.readInt16BE(4) * 0.1;
newMsg.payload.temperature =
  (buffer.readInt16BE(6) >> 2) * 0.1;
newMsg.payload.out2 = Boolean(buffer[7] & 2);
newMsg.payload.out1 = Boolean(buffer[7] & 1);

return newMsg;
```

Wie Sie sehen können, kopieren wir die 8 Bytes aus dem eingehenden Datenpaket und speichern es in einem temporären Puffer. Dann extrahieren wir die verschiedenen Messwerte und skalieren sie nach Bedarf. Anschließend, senden wir ein neues Datenpaket mit den fünf Feldern. Nach dem Start des Ablaufs wird die Ausgabe wie in Abb 15 dargestellt erzeugt.

```
info debug dashboard x
all nodes
29.8.2018, 11:00:41 node: 5f6a9a30.7ded0c
msg.payload : Object
▶ { totalizer: 15014.4462890625, flowrate: 0.4,
  temperature: 27.6, out2: false, out1: false }
29.8.2018, 11:00:41 node: 5f6a9a30.7ded0c
msg.payload : Object
▶ { totalizer: 15014.4462890625, flowrate: 0.4,
  temperature: 27.6, out2: false, out1: false }
29.8.2018, 11:00:41 node: 5f6a9a30.7ded0c
msg.payload : Object
▶ { totalizer: 15014.4462890625, flowrate: 0.4,
  temperature: 27.6, out2: false, out1: false }
29.8.2018, 11:00:42 node: 5f6a9a30.7ded0c
msg.payload : Object
▶ { totalizer: 15014.4462890625, flowrate: 0.4,
  temperature: 27.6, out2: false, out1: false }
29.8.2018, 11:00:42 node: 5f6a9a30.7ded0c
msg.payload : Object
▶ { totalizer: 15014.4462890625, flowrate: 0.4,
  temperature: 27.6, out2: false, out1: false }
29.8.2018, 11:00:42 node: 5f6a9a30.7ded0c
msg.payload : Object
▶ { totalizer: 15014.4462890625, flowrate: 0.9,
  temperature: 27.6, out2: false, out1: false }
29.8.2018, 11:00:42 node: 5f6a9a30.7ded0c
msg.payload : Object
▶ { totalizer: 15014.4462890625, flowrate: 0.8,
  temperature: 27.6, out2: false, out1: false }
29.8.2018, 11:00:43 node: 5f6a9a30.7ded0c
msg.payload : Object
▶ { totalizer: 15014.447265625, flowrate: 4.3,
  temperature: 27.6, out2: false, out1: false }
```

Abb 15: Debug-Ausgabe des IO-Link-Sensors



Pushing Performance

Sensoren und Aktoren mit MICA verbinden

People | Power | Partnership

Anlage

Mit den folgenden Skripten stehen die kompletten Abläufe für eigene Experimente zur Verfügung. Kopieren Sie einfach den Text und importieren Sie ihn über das Optionsmenü oben rechts (*Import*, dann *Clipboard*) in Node-RED.

Basis-Ablauf:

```
[[{"id":"fda89d6.8cd9b8","type":"tab","label":"BasicFlow","disabled":false,"info":"","id":"b544d36.889c9","type":"ui_template","z":"fda89d6.8cd9b8","group":"4a1bafd9.e29fc","name":"","order":3,"width":0,"height":0,"format":"<div style='position:fixed;top:5px;z-index:1000;right:20px;background-color:white'><n <img style='height:50px' src='/u/harting_logo.png'></div>","storeOutMessages":true,"fwInMessages":true,"templateScope":"local","x":80,"y":1100,"wires":[]}],{"id":"ec40e680.2ddc68","type":"modbus-read","z":"fda89d6.8cd9b8","name":"Read DI","topic":"","showStatusActivities":false,"showErrors":false,"unitid":"","dataType":"InputRegister","adr":"0","quantity":"1","rateUnit":"s","delayOnStart":false,"startDelayTime":"","server":"4a6b1c76.ec6e74","useOfFile":false,"ioFile":"","useOfPayload":false,"x":140,"y":100,"wires":[["7fc11a05.4d0e64"]]],{"id":"7fc11a05.4d0e64","type":"debug","z":"fda89d6.8cd9b8","name":"","active":false,"tosidebar":true,"console":false,"tostatus":false,"complete":"payload","x":140,"y":100,"wires":[]},{"id":"602d125b.60505c","type":"modbus-write","z":"fda89d6.8cd9b8","name":"Write DO","showStatusActivities":false,"showErrors":false,"unitid":"","dataType":"HoldingRegister","adr":"2048","quantity":"1","server":"4a6b1c76.ec6e74","x":360,"y":240,"wires":[["da2afa77.0757e"],"type":"inject","z":"fda89d6.8cd9b8","name":"","topic":"","payload":"2","payloadType":"num","repeat":"","crontab":"","once":false,"onceDelay":0.1,"x":130,"y":300,"wires":[["602d125b.60505c"]]],{"id":"a305acd7.65ed28","type":"inject","z":"fda89d6.8cd9b8","name":"","topic":"","payloadType":"num","repeat":"","crontab":"","once":false,"onceDelay":0.1,"x":130,"y":240,"wires":[["602d125b.60505c"]]],{"id":"2d321057.e3b6d8","type":"comment","z":"fda89d6.8cd9b8","name":"Digital Input","info":"","x":90,"y":40,"wires":[]},{"id":"c4180b7b.73672","type":"comment","z":"fda89d6.8cd9b8","name":"Digital Output","info":"","x":90,"y":180,"wires":[]},{"id":"281a0a55.dcc9d6","type":"comment","z":"fda89d6.8cd9b8","name":"Analog Input","info":"","x":90,"y":380,"wires":[]},{"id":"c4180b7b.73672","type":"modbus-read","z":"fda89d6.8cd9b8","name":"Read AI","topic":"","showStatusActivities":false,"showErrors":false,"unitid":"","dataType":"InputRegister","adr":"0","quantity":"1","rateUnit":"s","delayOnStart":false,"startDelayTime":"","server":"3673605e.c40b3","useOfFile":false,"ioFile":"","useOfPayload":false,"x":130,"y":440,"wires":[["80a87d67.ecab1"]]],{"id":"80a87d67.ecab1","type":"debug","z":"fda89d6.8cd9b8","name":"","active":false,"tosidebar":true,"console":false,"tostatus":false,"complete":"payload","x":370,"y":440,"wires":[]},{"id":"4a1bafd9.e29fc","type":"ui_group","z":"","name":"Digital I/O","tab":"31749c17.f31444","order":2,"disp":true,"width":"8","collapse":false},{"id":"4a6b1c76.ec6e74","type":"modbus-client","z":"","name":"TBMEN-S1-8DXP","clientType":"tcp","bufferCommands":true,"stateLogEnabled":false,"tcpHost":"10.10.10.57","tcpPort":"502","tcpType":"DEFAULT","serialPort":"/dev/ttyUSB","serialType":"RTU-BUFFERD","serialBaudrate":"9600","serialDatabits":"8","serialParity":"none","serialConnectionDelay":"100","unit_id":"1","commandDelay":"1","clientTimeout":"1000","reconnectTimeout":"2000"},{"id":"3673605e.c40b3","type":"modbus-client","z":"","name":"TBMEN-S2-4AI","clientType":"tcp","bufferCommands":true,"stateLogEnabled":false,"tcpHost":"10.10.10.56","tcpPort":"502","tcpType":"DEFAULT","serialPort":"/dev/ttyUSB","serialType":"RTU-BUFFERD","serialBaudrate":"9600","serialDatabits":"8","serialParity":"none","serialConnectionDelay":"100","unit_id":"1","commandDelay":"1","clientTimeout":"1000","reconnectTimeout":"2000"},{"id":"31749c17.f31444","type":"ui_tab","z":"","name":"IO-Demo","icon":"dashboard"}]]
```

Erweiterter Ablauf mit grafischer Benutzeroberfläche:

```
[[{"id":"4b84aaf.713dd54","type":"tab","label":"Extended Flow","disabled":false,"info":"","id":"fe64f16f.abbec","type":"modbus-read","z":"4b84aaf.713dd54","name":"Read DI","topic":"","showStatusActivities":false,"showErrors":false,"unitid":"","dataType":"InputRegister","adr":"0","quantity":"1","rateUnit":"ms","delayOnStart":false,"startDelayTime":"","server":"4a6b1c76.ec6e74","useOfFile":false,"ioFile":"","useOfPayload":false,"x":140,"y":100,"wires":[["5b565d3d.b81b5c"]]],{"id":"8923aab.1bc1258","type":"modbus-write","z":"4b84aaf.713dd54","name":"Write DO","showStatusActivities":false,"showErrors":false,"unitid":"","dataType":"HoldingRegister","adr":"2048","quantity":"1","server":"4a6b1c76.ec6e74","x":340,"y":260,"wires":[["db1b1782.985ad8"],"type":"comment","z":"4b84aaf.713dd54","name":"Digital Input","info":"","x":90,"y":40,"wires":[]},{"id":"b3068f7e.f0c6","type":"comment","z":"4b84aaf.713dd54","name":"Digital Output","info":"","x":90,"y":180,"wires":[]},{"id":"9cadcb85.684b9","type":"comment","z":"4b84aaf.713dd54","name":"Analog Input","info":"","x":90,"y":340,"wires":[]},{"id":"baf6a366.cfe36","type":"modbus-read","z":"4b84aaf.713dd54","name":"Read AI","topic":"","showStatusActivities":false,"showErrors":false,"unitid":"","dataType":"InputRegister","adr":"0","quantity":"1","rateUnit":"s","delayOnStart":false,"startDelayTime":"","server":"3673605e.c40b3","useOfFile":false,"ioFile":"","useOfPayload":false,"x":130,"y":400,"wires":[["7874f0c4.b288b8"]]],{"id":"7874f0c4.b288b8","type":"function","z":"4b84aaf.713dd54","name":"scale","func":"","to be adjusted depending on your actual sensor unit/msg.payload = msg.payload[0] / 10;\nreturn msg,\"outputs\":1,\"noerr\":0,\"x\":330,\"y\":400,\"wires":[[\"d7b5e68b.35666\", \"83ba1387.636d78\"]]],{"id":"d7b5e68b.35666","type":"ui_gauge","z":"4b84aaf.713dd54","name":"","group":"8f083b51.582b2","order":0,"width":0,"height":0,\"type\":\"gage\",\"title\":\"Temperature\",\"label\":\"units\",\"format\":\"{value}\",\"min\":0,\"max\":\"100\",\"colors\":{\"#00b050,\"#e66600,\"#ca3838\"},\"seg1\":\"\",\"seg2\":\"\",\"x\":550,\"y\":400,\"wires\":[]},{"id":"83ba1387.636d78","type":"ui_chart","z":"4b84aaf.713dd54","name":"","group":"8f083b51.582b2","order":0,\"width\":0,\"height\":0,\"label":"","chartType\":\"line\",\"legend\":\"false\",\"format\":\"HH:mm:ss\",\"interpolate\":\"linear\",\"nodata\":\"\",\"dot\":\"false\",\"ymin\":\"\",\"ymax\":\"\",\"removeOlder\":\"5\",\"removeOlderUnit\":\"60\",\"cutout\":0,\"useOncolor\":\"false\",\"colors\":[\"#f1774d\", \"#aec6e8\", \"#ff7f0e\", \"#2ca02c\", \"#98df8a\", \"#d62728\", \"#ff7f0e\", \"#9467bd\", \"#5b0bd5\"],\"useOldStyle\":\"false\",\"x\":530,\"y\":460,\"wires\":[]},{"id":"99c0e76e.78a556","type":"ui_switch","z":"4b84aaf.713dd54","name":"","label\":\"Output\",\"group\":\"4a1bafd9.e29fc\",\"order\":2,\"width\":0,\"height\":0,\"passthru\":\"false\",\"decouple\":\"false\",\"topic":"","style":"","onvalueType\":\"num\",\"onicon":"","oncolor":"","offvalueType\":\"num\",\"officon":"","offcolor":"","x\":110,\"y\":260,\"wires\":[]},{"id":"8923aab.1bc1258","type":"comment","z":"4b84aaf.713dd54","name":"","label\":\"Input\",\"group\":\"4a1bafd9.e29fc\",\"order\":1,\"width\":0,\"height\":0,\"passthru\":\"false\",\"decouple\":\"true\",\"topic":"","style":"","onvalue\":\"true\",\"onvalueType\":\"bool\",\"onicon":"","oncolor":"","offvalueType\":\"bool\",\"officon":"","offcolor":"","x\":530,\"y\":100,\"wires":[]},{"id":"5b565d3d.b81b5c","type":"function","z":"4b84aaf.713dd54","name":"transform","func":"msg.payload = msg.payload[0] == 1 ? msg.enabled == false ?\nreturn msg,\"outputs\":1,\"noerr\":0,\"x\":340,\"y\":100,\"wires":[[\"f2a63ab2.3503a8\"]]],{"id":"f2a63ab2.3503a8","type":"ui_template","z":"4b84aaf.713dd54","group":"4a1bafd9.e29fc","name":"","order":3,"width":0,"height":0,\"format\":\"<div style='position:fixed;top:5px;z-index:1000;right:20px;background-color:white'><n <img style='height:50px' src='/u/harting_logo.png'></div>","storeOutMessages":true,"fwInMessages":true,"templateScope":"local","x":100,\"y\":1100,\"wires":[]}],{"id":"4a6b1c76.ec6e74","type":"modbus-client","z":"","name":"TBMEN-S1-8DXP","clientType":"tcp","bufferCommands":true,"stateLogEnabled":false,"tcpHost":"10.10.10.57","tcpPort":"502","tcpType":"DEFAULT","serialPort":"/dev/ttyUSB","serialType":"RTU-BUFFERD","serialBaudrate":"9600","serialDatabits":"8","serialParity":"none","serialConnectionDelay":"100","unit_id":"1","commandDelay":"1","clientTimeout":"1000","reconnectTimeout":"2000"},{"id":"3673605e.c40b3","type":"modbus-client","z":"","name":"TBMEN-S2-4AI","clientType":"tcp","bufferCommands":true,"stateLogEnabled":false,"tcpHost":"10.10.10.56","tcpPort":"502","tcpType":"DEFAULT","serialPort":"/dev/ttyUSB","serialType":"RTU-BUFFERD","serialBaudrate":"9600","serialDatabits":"8","serialParity":"none","serialConnectionDelay":"100","unit_id":"1","commandDelay":"1","clientTimeout":"1000","reconnectTimeout":"2000"},{"id":"8f083b51.582b2","type":"ui_group","z":"","name":"Analog Input","tab":"31749c17.f31444","order":1,\"disp\":true,\"width":"8","collapse":false},{"id":"4a1bafd9.e29fc","type":"ui_group","z":"","name":"Digital I/O","tab":"31749c17.f31444","order":2,\"disp\":true,\"width":"8","collapse":false},{"id":"31749c17.f31444","type":"ui_tab","z":"","name":"IO-Demo","icon":"dashboard"}]]
```

IO-Link Flow:

```
[[{"id":"74b42118.fc23c","type":"tab","label":"IO-Link Flow","disabled":false,"info":"","id":"9ac05b49.5456f8","type":"ui_template","z":"74b42118.fc23c","group":"4a1bafd9.e29fc","name":"","order":3,"width":0,"height":0,\"format\":\"<div style='position:fixed;top:5px;z-index:1000;right:20px;background-color:white'><n <img style='height:50px' src='/u/harting_logo.png'></div>","storeOutMessages":true,\"fwInMessages\":true,\"templateScope":"local","x":80,\"y":1100,\"wires":[]}],{"id":"6d0decbf.4f1d5c","type":"debug","z":"74b42118.fc23c","name":"","active":false,\"tosidebar\":true,\"console\":false,\"tostatus\":false,\"complete":"payload","x":610,\"y":100,\"wires":[]},{"id":"97e89a31.cd158","type":"comment","z":"74b42118.fc23c","name":"IO-Link Input","info":"","x":90,\"y":40,\"wires":[]},{"id":"5b20444d.37eb24","type":"function","z":"74b42118.fc23c","name":"SD6000","func":"const buffer = Buffer.allocUnsafe(8);\nbuffer.writeUInt16BE(msg.payload[0], 0);\nbuffer.writeUInt16BE(msg.payload[2], 4);\nbuffer.writeUInt16BE(msg.payload[3], 6);\nreturn newMsg = {};\nnewMsg.payload = buffer.readFloatBE(0); // bytes 0 - 3, [ / min ]\nnewMsg.payload.flowrate = buffer.readFloatBE(4); // bytes 4 + 5, [ / min ]\nnewMsg.payload.temperature = buffer.readInt16BE(6) >> 2 * 0.1; // bytes 6 + 7 [ C ]\nnewMsg.payload.out2 = Boolean(buffer[7] & 2);\nnewMsg.payload.out1 = Boolean(buffer[7] & 1);\nreturn newMsg,\"outputs\":1,\"noerr\":0,\"x\":400,\"y\":100,\"wires":[[\"6d0decbf.4f1d5c\"]]],{"id":"68779e63.087a0c","type":"modbus-read","z":"74b42118.fc23c","name":"Read IO-Link device","topic":"","showStatusActivities":false,"showErrors":false,"unitid":"","dataType":"InputRegister","adr":"2","quantity":"4","rateUnit":"s","delayOnStart":false,"startDelayTime":"1000","server":"4a6b1c76.ec6e74","useOfFile":false,"ioFile":"","useOfPayload":false,"x":170,\"y":100,\"wires":[[\"5bd2044d.37eb24\"]]],{"id":"4a1bafd9.e29fc","type":"ui_group","z":"","name":"Digital I/O","tab":"31749c17.f31444","order":2,\"disp\":true,\"width":"8","collapse":false},{"id":"4a6b1c76.ec6e74","type":"modbus-client","z":"","name":"TBMEN-4IOL","clientType":"tcp","bufferCommands":true,\"stateLogEnabled\":false,\"tcpHost":"10.10.10.58","tcpPort":"502","tcpType":"DEFAULT","serialPort":"/dev/ttyUSB","serialType":"RTU-BUFFERD","serialBaudrate":"9600","serialDatabits":"8","serialParity":"none","serialConnectionDelay":"100","unit_id":"1","commandDelay":"1","clientTimeout":"1000","reconnectTimeout":"2000"},{"id":"31749c17.f31444","type":"ui_tab","z":"","name":"IO-Demo","icon":"dashboard"}]]
```