

# DrvModbus

---

Drivrutinen DrvModbus används för att kommunicera via Modbus TCP, RTU och ASCII.

**” Information:**

För mer information om protokollet Modbus läs till exempel <https://en.wikipedia.org/wiki/Modbus>

DrvModbus använder sig av en modbus-master från proconX.

**” Information:**

Detta dokument används i kombination med kapitel 3 i manualen för Web Port.

För att använda DrvModbus skapas först en IO-enhet som sedan används för att kommunicera med den anslutna enheten. Taggar används sedan för att definiera vilka modbusregister som skall läsas och skrivas (se kapitel Taggar).

Drivrutinen har inte några egna inställningar utan dessa sätts unikt för varje IO-enhet som skapas.

## Installation

Installationen av DrvModbus görs genom att välja DrvModbus vid installationen av Web Port. Se *kapitel 1* för mer information kring installationsprocessen.

## IO-enhet

För att använda DrvModbus skapas först en IO-enhet av typen DrvModbus. För mer information se kapitel 3.2.1 i manualen.

Utöver generella inställningar (se kapitel 3.2.4 i manualen) finns följande inställningar för en IO-enhet av typen DrvModbus.

**Läsfördröjning**

Fördröjning i millisekunder från svar till nästa läsning från enheten. Används för att begränsa hur ofta frågor ställs till en modbus-enhet. Detta kan vara användbart vid långsamma enheter.

**Återläsningsförsök**

Hur många omedelbara återförsök som görs vid misslyckad läsning.

**Max lästid**

Den maximala tid en läsning får ta i sekunder.

**Big Endian heltal**

Anger om Big Endian används vid läsning av stora heltal, alltså vilket register som är mest signifikant vid en läsning över två register. Ex: Om aktiverat på tagg med adress 40001 med datatyp LONG kommer värdet från register 40001 betraktas som det mest signifikanta registret (innehållande värden > 65535) och värdet från register 40002 betraktas som det minst signifikanta registret (innehållande värden < 65536). Får man värden i helt fel storleksordning på t.ex LONG- eller ULONG-taggar så kan det vara den här inställningen som behöver justeras.

**Växlade flyttal**

Anger om växlad bitordning används vid flyttal.

**Slav-adress**

Slavadress för enheten. Kan lämnas tom och istället anges på taggadress, se inforuta under *Taggar*

**Modbustyp**

Anger vilken typ av modbus som skall användas: TCP, Encapsulated RTU, RTU (Seriell), ASCII.

**Initieringsadress**

Ange valfri fungerande modbus-adress för att göra utökad kommunikations-kontroll vid anslutning.

**Använd KeepAlive**

Genomför läsning från initieringsadress varje sekund. Använd endast vid problem.

**Skriv HR med FC16**

Skriver till holdingregister med FC16 (write multiple registers) som standard.

**(TCP) Enhetsadress**

IP-adress till modbus-enhet eller modbus-gateway. Finns namnuppslagning på nätverket kan även enhetens namn användas. Används endast vid **TCP** och **Encapsulated RTU**.

**(TCP) Port**

Port för kommunikation. För Modbus TCP används vanligen port 502. Används endast vid **TCP** och **Encapsulated RTU**.

**(RTU) COM-port**

COM-port på datorn för seriell kommunikation.

**(RTU) Baudrate**

Hastighet för seriell kommunikation.

**(RTU) Paritet**

Paritet för seriell kommunikation.

**(RTU) Databitar**

Antal databitar för seriell kommunikation.

**(RTU) Stoppbitar**

Antal stoppbitar för seriell kommunikation.

**Länka till**

Används för att länka ihop flera IO-device av DrvModbus-typ. Detta används då flera seriella modbus-enheter finns på samma slinga. Kommunikationsinställningar sätts då för den första enheten på slingan medan de andra länkas till denna. Observera att Slav-adressen måste anges för samtliga enheter.

För mer information om hur IO-enheter skapas, redigeras och tas bort se kapitel 3 i manualen.

## Taggar

För att läsa och skriva modbusregister via en IO-enhet måste taggar skapas och knyts till densamma. För mer information om hur tagglistor och taggar skapas se kapitel 4 i manualen.

**” Tips:**

Skapa en tagglista för varje IO-enhet. Detta för att få en bra struktur och enkelt hitta taggar i listorna.

Taggar som knyts till en IO-enhet av typen DrvModbus kan ha följande datatyper:

<b>DIGITAL</b>	Används för <b>Discrete Input</b> and <b>Output (Coils)</b> .
<b>(U)INT</b>	Används för 16-bitars heltal, <b>Input-</b> och <b>Output (Holding)</b> -register.
<b>(U)LONG</b>	Används för 32-bitars heltal, <b>Input-</b> och <b>Output (Holding)</b> -register. Läser från två sekventiella register.
<b>(U)INT64</b>	Används för 64-bitars heltal, <b>Input-</b> och <b>Output (Holding)</b> -register. Läser från fyra sekventiella register.
<b>REAL</b>	Används för 32-bitars flyttal, <b>Input-</b> och <b>Output (Holding)</b> -register.

**" Information:**

Adressfältet anger den modbus-adress som skall läsas / skrivas.

Adresser kan anges både med fullständig registerplats samt med funktionskod. Då funktionskod används anges adressen på formen **funktionskod:adress**.

**" Exempel:**

<b>Registertyp</b>	<b>Med funktionskod</b>	<b>Fullständigt register</b>
<i>Discrete outputs (Coils)</i>	<i>1:1 – 1:65536</i>	<i>00001 - 065536</i>
<i>Discrete Inputs</i>	<i>2:1 - 2:65536</i>	<i>10001 - 165536</i>
<i>Input registers</i>	<i>4:1 – 4:65536</i>	<i>30001 - 365536</i>
<i>Output (Holding) registers</i>	<i>3:1 – 3:65536</i>	<i>40001 – 465536</i>

**" Tips:**

För att invertera en tagg eller göra en jämförelse kan adressen avslutas med ett booleskt uttryck. Datatypen förblir densamma men taggens värde blir 1 eller 0 beroende på om uttrycket uppfylls eller inte.

**" Exempel:**

40001=0                      Returnerar 1 då register 40001 = 0.

För mer information se kapitel 4.3 i manualen.

**" Tips:**

För att maska ut enskilda bitar från ett register kan följande syntax användas på adressen: Adress.bit-index. Index väljs mellan 0-15 där 0 är den minst signifikanta. Som datatyp väljs INT.

40001.0                      Returnerar den minst signifikanta biten (lägst till höger)

**" Information:**

För att ange slavadresser direkt på taggadresserna, använd formatet *slavadress@register*, t.ex *1@40001*.