

Instruktion för att använda modbusmodul M31-AXAX8080G-L och analog ingångsmodul GXGXX0800

Vad trevligt att du har köpt en modbusmodul av oss! Så här får du igång den:

Koppla 9-36v till modbusenheten och en patchkabel till datorn, eller till en switch om du redan har en AXBB eller UC300ETH kopplad till datorn, du kopplar då både modbusmodulen och tex AXBB till switchen, och sedan vidare till datorn från switchen.

Starta Configuration Tool V1.3.

Ändra till ethernet och klicka på search.

Huvudmodulen bör då dyka upp.

Markera den och klicka på Negotiation för att den ska hitta analogmodulen också.

När det står att det är klart, stäng av och sätt på huvudmodulen och klicka på search igen.



Du bör nu även se analogmodulen:

Device model	IP/Slave address	Type
▼ M31-AXAX8080G-L	192.168.1.7	
GAXAX8080	0	Master
GXGXX0800	1	Module

Klicka på Master modulen och Config.

DHCP	<input type="text" value="Disable"/>	Gateway	<input type="text" value="192.168.1.1"/>
Local IP address	<input type="text" value="192.168.1.7"/>	Local port	<input type="text" value="502"/>
Remote port	<input type="text" value="502"/>	DNS server	<input type="text" value="114.114.114.114"/>

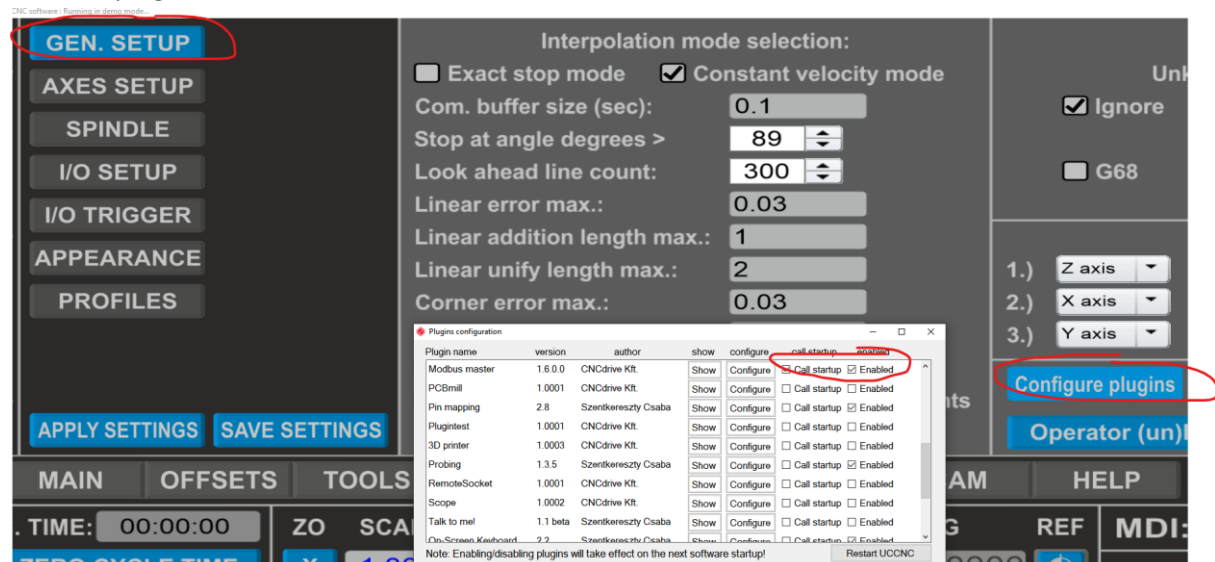
Ändra Local IP address till antingen ett ledigt ip nummer på ditt nätverk eller om AXBBn och modulen är kopplat direkt till datorn via en switch till tex 10.10.10.2 om din dator är 10.10.10.1 och AXBBn 10.10.10.10 etc.

Klicka på Save.

Nu över till UCCNC:

Du kan antingen ladda in vår testprofil och utgå från den, eller ändra i din befintliga:

Aktivera pluginen för modbus:



Starta om UCCNC.

Modbuspluginen startar nu automatiskt om du har klickat i "call startup".

Motioncontrollern behöver inte vara ansluten för att modbus ska fungera.

Klicka på Add connection och ställ in den med samma ip-nummer som du har satt på modulen:

Modbus master

Add Connection Remove Connection Add Function Remove Function Start loops Save Settings

00 AAAX444

- 00 Status
- 01 Set outputs
- 02 Read outputs
- 03 Read inputs
- 04 Read analog in

Connection settings Function settings Debug view Variables table

Modbus connection type

☒ TCP
☐ Serial ASCII
☐ Serial RTU

☒ Enable connection

Description AAAX444

IP address 192 . 168 . 1 . 7

Port number 502

Retry count 3

Timeout(ms) 100

Loop interval(ms) 100

Lägg till följande funktioner:

Modbus master

Add Connection Remove Connection Add Function Remove Function Start loops Save Settings

00 AAAX444

- 00 Status
- 01 Set outputs
- 02 Read outputs
- 03 Read inputs
- 04 Read analog in

Connection settings Function settings Debug view Variables table

☒ Enable function

Function name Status

Function type Read Holding Registers

Slave address 1

Modbus start register 30087

Register count 1

UCCNC start register 0

Modbus master

Add ConnectionRemove ConnectionAdd FunctionRemove FunctionStart loopsSave Settings

00 AAAX444

- 00 Status
- 01 Set outputs
- 02 Read outputs
- 03 Read inputs
- 04 Read analog in

Connection settingsFunction settingsDebug viewVariables table

☒ Enable function

Function nameSet outputs

Function typeWrite Multiple Coils

Slave address1

Modbus start register0

Register count10

UCCNC start register10

Modbus master

Add ConnectionRemove ConnectionAdd FunctionRemove FunctionStart loopsSave Settings

00 AAAX444

- 00 Status
- 01 Set outputs
- 02 Read outputs
- 03 Read inputs
- 04 Read analog in

Connection settingsFunction settingsDebug viewVariables table

☒ Enable function

Function nameRead outputs

Function typeRead Coils

Slave address1

Modbus start register0

Register count10

UCCNC start register20

Modbus master

Add Connection Remove Connection Add Function Remove Function Start loops Save Settings

00 AAAX444

- 00 Status
- 01 Set outputs
- 02 Read outputs
- 03 Read inputs
- 04 Read analog in

Connection settings Function settings Debug view Variables table

☒ Enable function

Function name Read inputs

Function type Read Inputs

Slave address 1

Modbus start register 0

Register count 10

UCCNC start register 30

Modbus master

Add Connection Remove Connection Add Function Remove Function Start loops Save Settings

00 AAAX444

- 00 Status
- 01 Set outputs
- 02 Read outputs
- 03 Read inputs
- 04 Read analog in

Connection settings Function settings Debug view Variables table

☒ Enable function

Function name Read analog in

Function type Read Input Registers

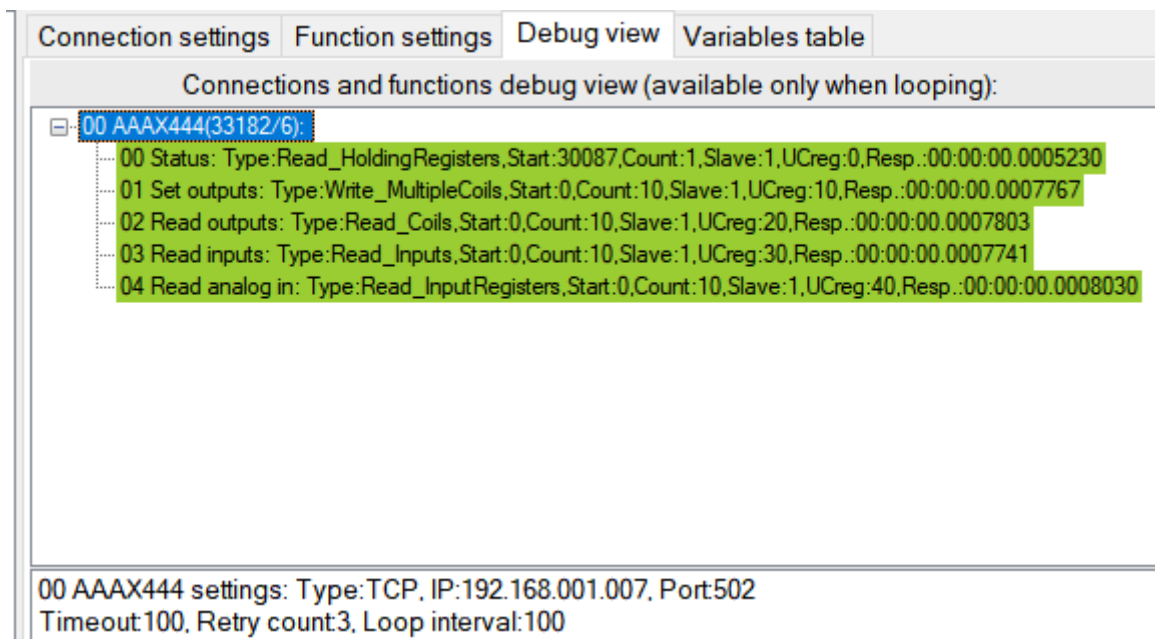
Slave address 1

Modbus start register 0

Register count 10

UCCNC start register 40

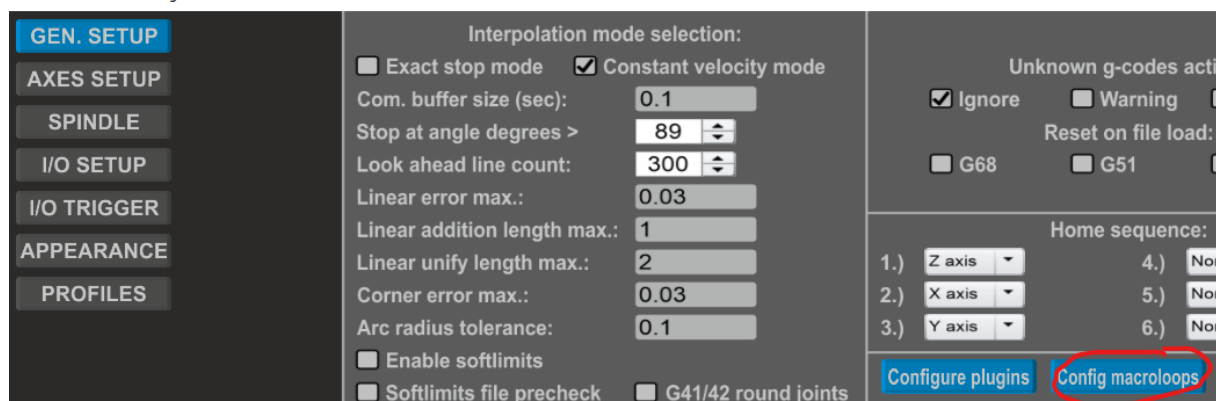
Klicka på save settings och klicka på Start loops och förhoppningsvis ser det då ut så här:



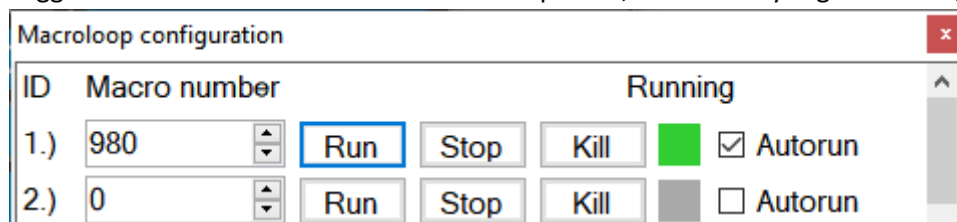
Om det inte fungerar, klicka på 00 AAAX444..... och det ska då stå det ip-numret som du har ställt in i nedre delen av rutan. Vi har varit med om att det har buggat lite i vissa versioner så att man måste stänga av och ändra till rätt nummer i profilen istället och sedan starta UCCNC igen.

När det fungerar kan du stänga av UCCNC och kopiera makrona M980, M20014 och M20015 till UCCNC\Profiles\dinMakroMapp, om du inte lägger in hela vår profil och makromapp.

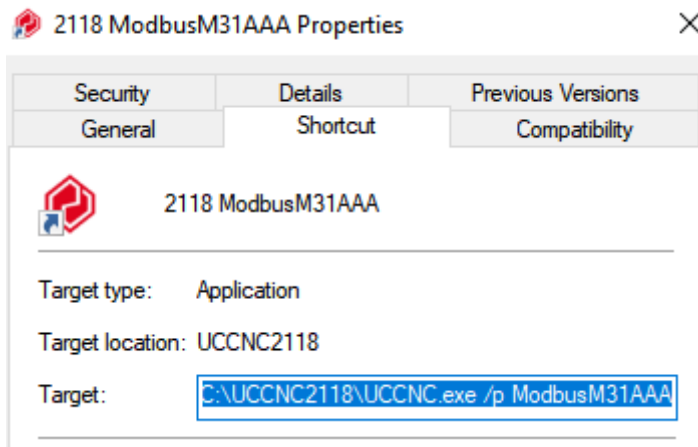
Gå in på Config makro loops:



Lägg till 980 och klicka i Autorun samt klicka på Run, det ska då lysa grönt till höger som nedan:



Om du använder vår profil kan du starta den genom att antingen modifiera en befintlig genväg eller skapa en ny från c:\UCCNC\UCCNC.exe
Högerklicka på genvägen och lägg till /p ModbusM31AAA
UCCNC2118 är bara för att jag har flera versioner på min dator så ändra inte till det 😊



I M980 makrot kan du avaktivera FRO, SRO eller JOG om du inte vill använda något av dem genom att byta ut true till false.

Du kan även ändra vilket modbusregister som du vill använda genom att ändra REG_FRO, JOG eller SRO, i modulen finns 8st analoga ingångar och de ligger på modbusregister 40-47.

Du kan även ändra skalan genom att ändra min och max för respektive funktion.

Skalning råvärde är min och max som analogmodulen skickar till uccnc och dessa ska du inte ändra.

```
// =====  
// UCCNC Modbus -> Overrides  
// =====  
  
// ----- AKTIVERING + REGISTerval -----  
const bool ENABLE_FRO = true;  
const bool ENABLE_SRO = true;  
const bool ENABLE_JOG = true;  
  
const int REG_FRO = 40; // Modbus register för FRO  
const int REG_JOG = 41; // Modbus register för Jog rate  
const int REG_SRO = 42; // Modbus register för Spindle override  
  
// ----- FIELDS -----  
const int FIELD_FRO = 232; // FRODRO  
const int FIELD_SRO = 233; // SRODRO  
const int FIELD_JOG = 913; // Jogfeedrate (0-100% i JogSettings)  
  
// ----- SKALNING RÅVÄRDE -----  
const double RAW_MIN = 0.0;  
const double RAW_MAX = 10000.0;  
  
// ----- UTVÄRDES-SPANN -----  
const int FRO_MIN = 0;  
const int FRO_MAX = 300;  
  
const int SRO_MIN = 0;  
const int SRO_MAX = 300;  
  
const int JOG_MIN = 0; // 0%  
const int JOG_MAX = 100; // 100%
```

Ändra en utgång:

För att ändra en utgång skriver du i cnc-koden eller i MDI: M20015 E10 Q1 för att sätta utgång 1 på modulen.

E10 = modbusregister 10 och utgångarna ligger på 10-17.

Ändra till Q0 för att stänga av utgången.

Vänta på ingång:

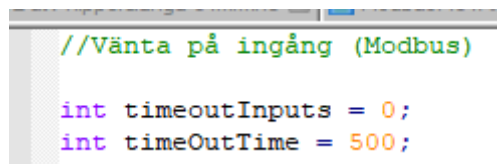
För att vänta på en ingång skriver du i koden eller i MDI tex M20014 E30 Q1 för att vänta på att modbusregister 30 ska bli 1.

Ingångarna ligger på register 30-37.

För att vänta på att ingången ska bli 0 ändrar du till Q0 istället.

Om ingången inte aktiveras inom 5sek så stoppas körningen och du får ett meddelande i statusfältet.

På fjärde raden i M20014 makrot kan du ändra max tiden, timeOutTime om så önskas, den står på 500 som default och det är lika med 500x10ms dvs 5sek.



```
//Vänta på ingång (Modbus)

int timeoutInputs = 0;
int timeOutTime = 500;
```

Hör av dig till johan@jbcnc.se om du kör fast.

Mycket nöje!

/ Johan Brundin

JB CNC & Linear Components