Oversikt over alle arbeidsoppdrag

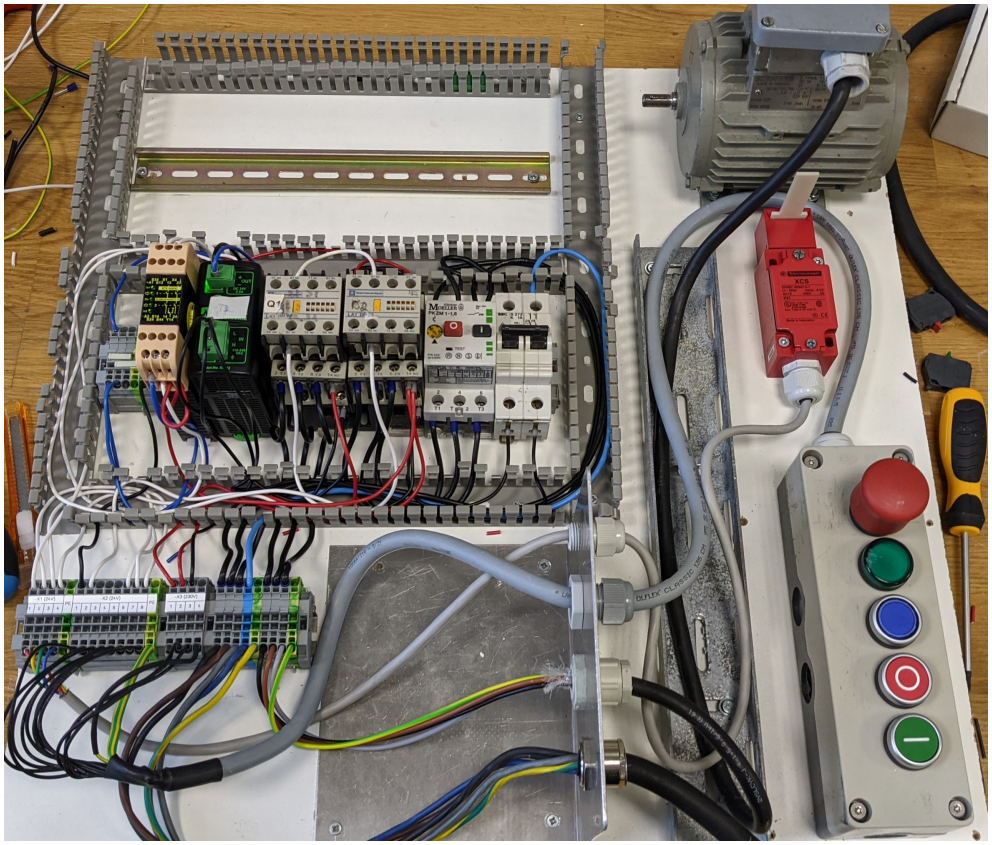
Oppgave 1

Styresystemer – Nivå 1

Arbidsoppdrag på Stasjon 16

Sikkerhetsrele med nødstopp og overvåking av bevegelig vern

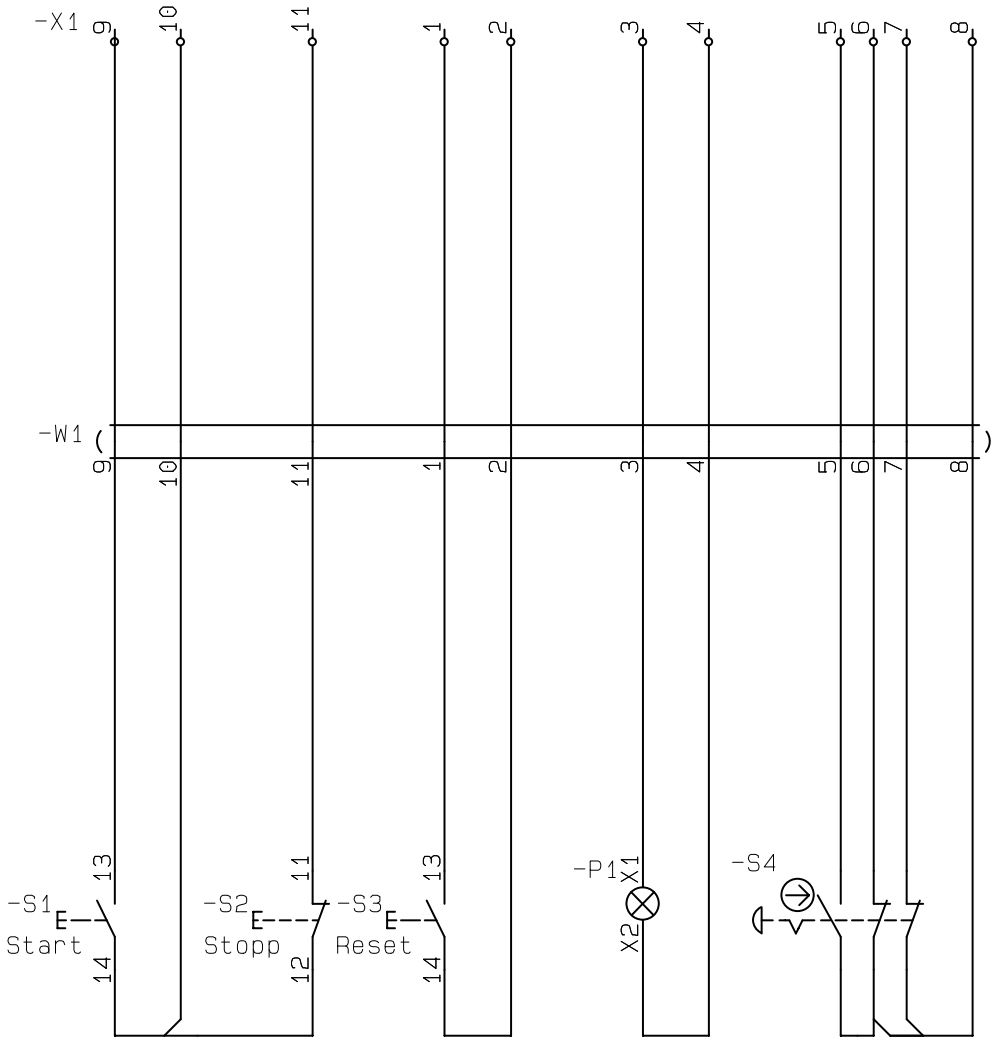
Introdusjon



Dette arbeidsoppdraget handler om sikkerhetsrelaterte deler av styre system. Du skal koble opp styrekrets til en tenkt maskin som er beskyttet med vern. Et bevegelig vern gir tilgang til maskinen. Det beveglige vernet overvåkes av en bryter som skal kobles til et Jokab RT9 sikkerhetsrele.

- Sikkerhetsrelatert stopp utløst av beveglig vern.
- Manuel Reset

Som et komplementært sikkerhetstiltak skal det monteres en nødsopp bryter. Det må også være vanlig drifts start og stopp.



Teorioppgaver
Oppgaver fra Kap 4

- Er en sikkerhetsfunksjon i styresystemet første valget ved konstruksjon av maskiner?
- Hva er en sikkerhetsfunksjon?
- Hva står PL for?

Oppgaver fra Kap 5.

- Gjelder EC machinery Directive i norge? Eventuelt hvilken lov er det implementert i?
- Hvilken sikkerhetsstandard er viktig grunnlag for maskin direktivet?
- Hva er en Type C standard under maskindirektivet?
- Beskriv prosedyren for risikoreduksjon av en maskin.
- Hvordan finner en frem til nødvendig sikkerhetsfunksjoner
- Hvilke sikkerhetsfunksjoner er definert i EN ISO 13849-1
- Hvordan bestemmer PLr for en sikkerhetsfunksjon.

Arbidsoppdrag – planlegging

Manualer
https://cdn.logic-control.com/docs/abb-jokab-safety/06.SafetyRelays.pdf
https://www.dguv.de/medien/ifa/en/pub/rep/pdf/reports-2019/report0217e/rep0217e.pdf
Tabell over fullførte oppgaver.

Liste over utstyr som du tranger		
Utstyr	Plass	sjekk
Jokab RT9		
24V strømforsyning		

Arbidsoppdrag – gjennomføring

Sjekkliste for lærer før oppkobling starter		
Utstyr	Plass	sjekk
Teorisprørsmål besvart		
Fremdriftsplan		
Verktøys- og utstyrsliste		
Risikovurdering		
Koblingsskjema for oppkobling		

Arbidsoppdrag – dokumentasjon
Beskriv hvordan du planlegger, gjennomfører og dokumenterer denne jobben.
Arbidsoppdrag – gjennomføring

Sjekkliste for lærer etter at sluttkontroll er utført	
Test	sjekk
Anlegget starter og stopper ved normal stopp	
Motor stopper når dørbryter eller nødstopp aktiveres	
Manuell resett virker i henhold til 13849-1	
Sjekk om kontaktorer har brent seg virker.	
Sluttkontroll er utført og dokumentert	
Elev har ført logg på Teams	
Elev har Tegnet koblingsskjema	

file i04821 mal

Styresystemer – Nivå 1

Arbidsoppdrag på Stasjon 1

Programmering av gandsfjoden Gondol med Wago PFC 100

Introdusjon

I Arbeidsoppdraget til gandsfjoden Gondol har du programmert stasjonen med Gand PLS 01. Nå skal du programmere stajonen med en Wago PLS 100. Stasjonen er ferdig oppkoblet og klar tl bruk.

Teorioppgaver

Nettsiden open PLC har gitt ut en stadard for hvordan en skal gå frem når en programmerer en PLS. I dette arbidsoppdraget skal du følge denne standarden den finner du her Presentasjon av prossedyren for programmering.

Planlegging

Sjekkliste for planlegging		
Oppgave	Elev	Lærer
Eleven har definert IO signaler (IO-liste)		
Eleven har definert signaler til sentralt kontrollsystem		
Eleven har definert alle HMI signaler, overstyringer og overvåkningsdata		
Elevn har del programmet opp i logiske deler (min 2.) Styreprogramm og utgangsprogramm		
Eleven har laget POU for hver av de logiske delene og programmene er godt kommentert		
Eleven har koble de fysike IO-ene mot variabler i programmet		

Gjennomføring

Dokumentasjon

Beskriv hvordan du planlegger, gjennomfører og dokumenterer denne jobben.
file i04822

Styresystemer – Nivå 2

Arbidsoppdrag på Stasjon 2

Dokumentering av IO-signaler

Introdusjon

I dette oppdraget skal du koble det opp mot Stasjon2 med Siemens TIA portal og bruke dette programmet til å sjekke at tilkoblede signaler virker. om skjma ikke stemmer, må du tegne hvordan de faktisk er koblet.

El-tegninger

Teorioppgaver

Sett eg inn i medfølgende el-tegniner og originalt skjema for skapet(finnesi papirform i skapet). Du skal være i stand til å følge signler og si hvilke som er korrekt tegnet og hvilke som ikke er det. Video tutorial Siemens TIA protal

Planlegging

Gjennomføring

Dokumentasjon

Beskriv hvordan du planlegger, gjennomfører og dokumenterer denne jobben.
file i04823

Styresystemer – Nivå 2

Arbidsoppdrag på Stasjon 3/4

Programmering av reguleringsstasjon

Introduksjon

Dette oppdraget skal du programmere en Gand reguleringsstasjon. Du kan velge mellom stasjon03 og stajson04. Du skal følge samme fremgangmåte som i programmeringen av nivå 1 oppdraget.
Programmet diss skal gjøre det mulig å regulere nivået i tanken og strømningen ut av tanken. Eksempel program med PID i codesys

Teorioppgaver Leseoppgave:

`https://autofaget.no/closedloop/node2.html`

Oppgavehefte:

https://autofaget.no/pdfs/Regulering.pdf

Planlegging

Gjennomføring

Dokumentasjon

file i04824

HMI – Nivå 1
Arbidsoppdrag på egen PC
HPHMI

Introdusjon

I dette arbeisoppdraget skal du lage en HMI etter prinsippene for HPHMI (High Performance HMI).

Du skal gjenskapes (ikke kopiere) prosjektet Intro til HPHMI codesys prsjekt

Teorioppgaver

Leseoppgave

Planlegging

Gjennomføring

Dokumentasjon

Beskriv hvordan du planlegger, gjennomfører og dokumenterer denne jobben.
file i04829

HMI – Nivå 2

Arbidsoppdrag på Stasjon 3/4

Lage HMI til reguleringsstasjon

Introdusjon Denne oppgaven bygger på programmet du har laget i oppgaven programmering av reguleringsstasjon. De anbefales å gjøre disse oppgavene sammen.

Du skal lage Følgende HMI bilder.

- Level2 For daglig betjening av stasjonen
- Level3 Som gir en P&ID oversikt over stasjonen
- Level4 for optimalisering av nivåregulering
- Level4 for optimalisering av strømnings regulering

Teorioppgaver

Planlegging

Gjennomføring

Dokumentasjon
Beskriv hvordan du planlegger, gjennomfører og dokumenterer denne jobben.
file i04830

Kommunikkasjon – Nivå 1

Arbidsoppdrag på Stasjon 14

Sjekk av komunkasjonsnett i klasserommet

Introdusjon I dette oppdraget skal du sjekke at nettverksdokumentasjonen for D426 stemmer. Denne finner i dokumentasjons mappen for D426. Du må sjakke at alle koblinger som vises på kart stemmer og at IP-adressene er slik som angitt i patchetabellen.

Teorioppgaver

1. Identify the practical purpose of each of these diagnostic utility programs, all accessible through the "command line" environment of a personal computer:

   - `ping`
   - `ipconfig` or `ifconfig`
   - `netstat`
   - `tracert` or `traceroute`
   - `nslookup`

   Be sure to bring your portable computer to class and experiment with each of these commands!

2. *Ethernet* actually encompasses several similar network standards, varying by cable type and bit rate (speed). The IEEE has designated special identifier names to denote each variety of Ethernet media. Identify which type of Ethernet each of these names refers to:

   - 10BASE2:
   - 10BASE5:
   - 10BASE-T:
   - 10BASE-F:
   - 100BASE-TX:
   - 100BASE-FX:
   - 1000BASE-T:
   - 1000BASE-SX:
   - 1000BASE-LX:

   file i04466

3. Each Ethernet device manufactured in the world today possesses a unique identifying number, known as a *MAC address* or *hardware address*. This address is 6 bytes, or octets, long (48 bits). An example of a valid Ethernet MAC address is shown here:

   `D2-48-1C-30-EA-B5`

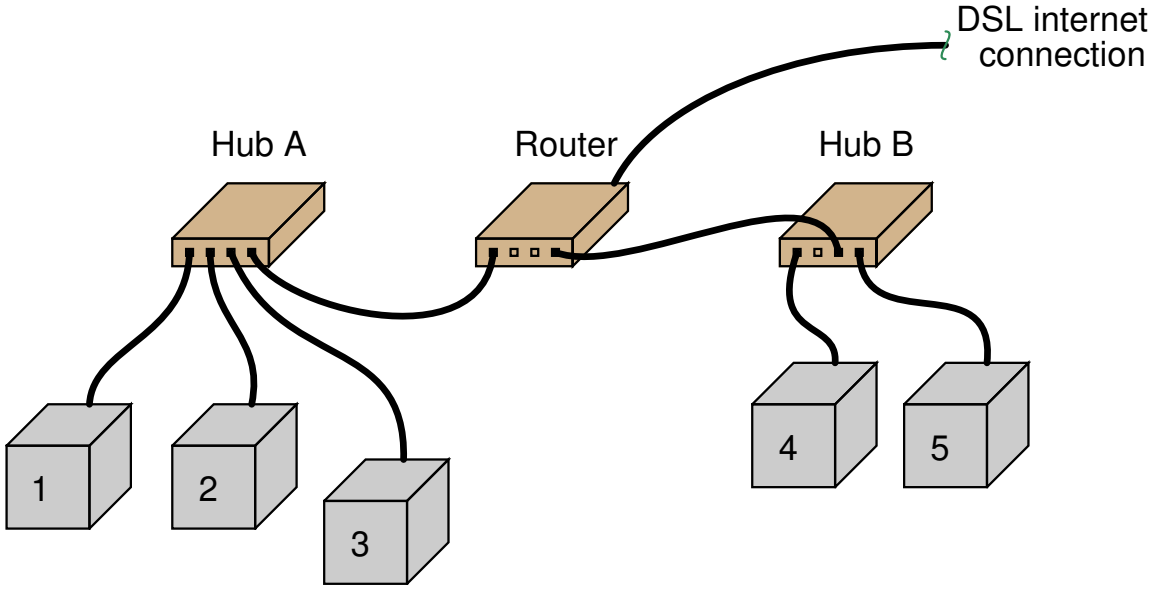
   Given the number of bits in a MAC address field, how many unique identifier addresses can exist in the world?

4. Each Ethernet device manufactured in the world today possesses a unique identifying number, known as a *MAC address* or *hardware address*. This address is 6 bytes, or octets, long (48 bits). An example of a valid Ethernet MAC address is shown here:

   `D2-48-1C-30-EA-B5`

   Given the number of bits in a MAC address field, how many unique identifier addresses can exist in the world?

5. This Ethernet network has a problem in it somewhere:



- 1 can "ping" 3
- 4 can "ping" `www.google.com`
- 4 can "ping" 5
- 2 cannot "ping" `www.google.com`

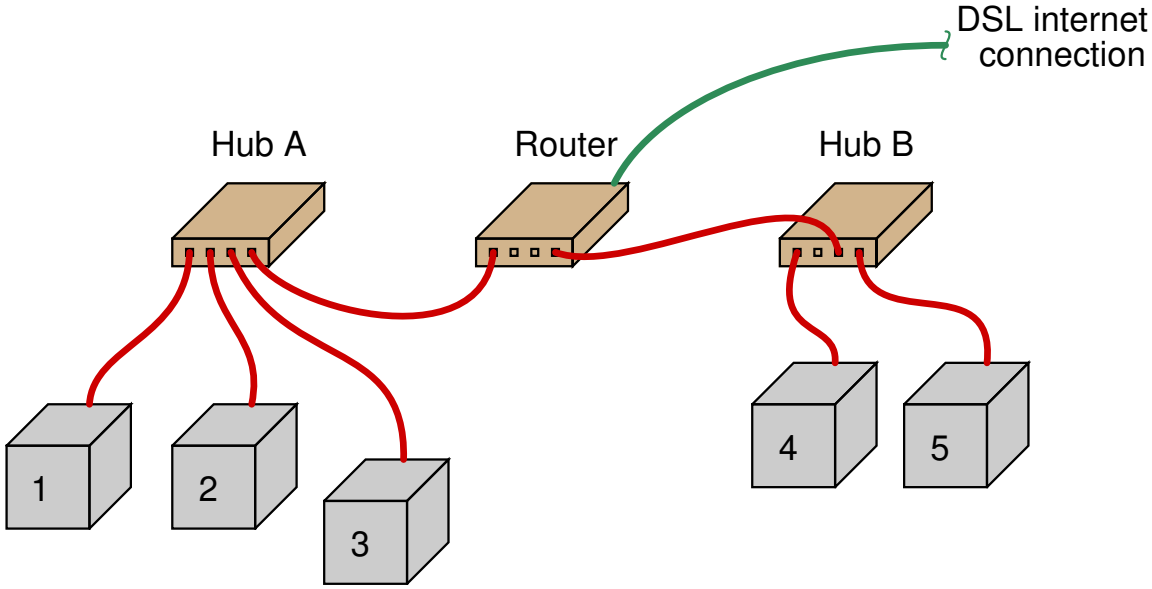
An excellent diagnostic strategy is to trace paths of data flow in a complex system, looking for places of intersection. Successful paths prove all points along the pathway are functioning. Places of overlap between unsuccessful paths prove that section is suspect.

Apply this strategy to the problem at hand, and use the results to narrow the field of possible faults.

Also, identify a good "next ping" test to do.

file i02123

6. The following Ethernet network has a problem. Someone trying to access the Internet from personal computer #4 cannot do so, and has called you to troubleshoot the problem:



Your first diagnostic test is to "ping" computer #4 from computer #5, and you find that this test is successful. Your next test is to check Internet connectivity at computer #5 by "pinging" `http://www.google.com`, and you find that test is successful as well.

Identify the likelihood of each specified fault for this network. Consider each fault one at a time (i.e. no coincidental faults), determining whether or not each fault could independently account for *all* measurements and symptoms in this network.

Fault	Possible	Impossible
Hub A failed		
Hub B failed		
Router failed		
Internet service provider failed		
Cable failed between computer #4 and Hub B		
Cable failed between Hub A and Router		
Cable failed between Hub B and Router		
Security settings (e.g. firewall) in computer #4		

Finally, identify the *next* diagnostic test or measurement you would make on this system. Explain how the result(s) of this next test or measurement help further identify the location and/or nature of the fault.

Planlegging

Gjennomføring

Dokumentasjon

file i04833

Kommunikkasjon – Nivå 1

Arbidsoppdrag på Stasjon 2

Oppkobling med Profinet til RIO Sjekk av komunkasjonsnett i klasserommet

Introduksjon
Målet med oppgaven er å lære hvordan vi kobler opp en PLS til en profinet RIO.

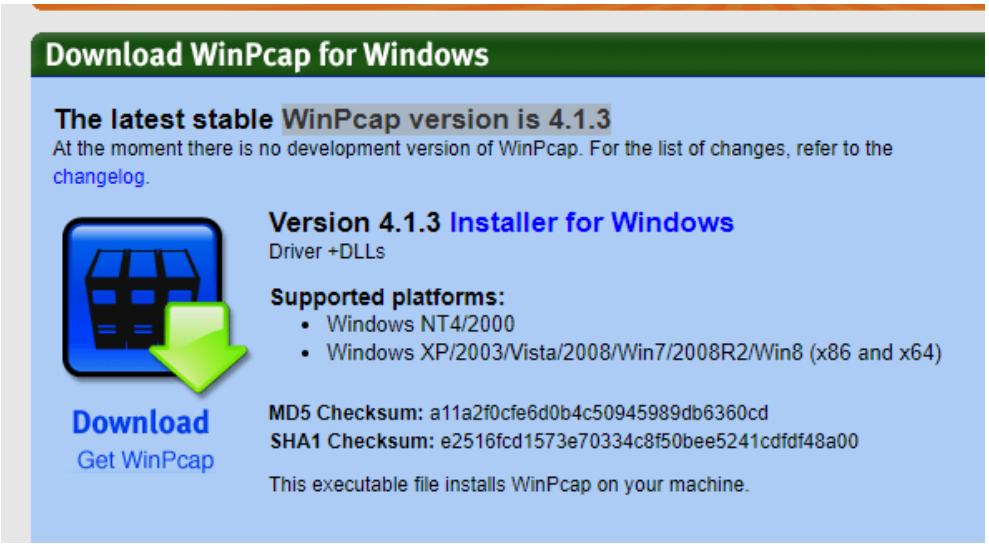
Arbidsoppdrag – teorioppgaver

1. Sett deg inn i manualen til aktuell DP-celle

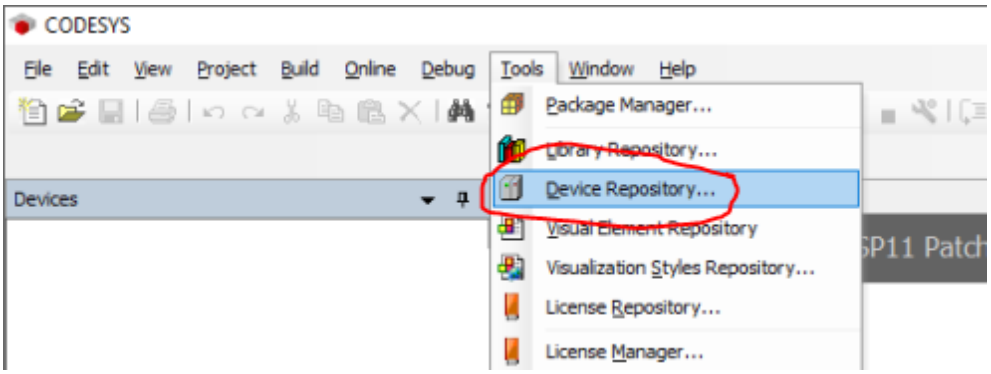
Arbidsoppdrag – planlegging
Utstyr:

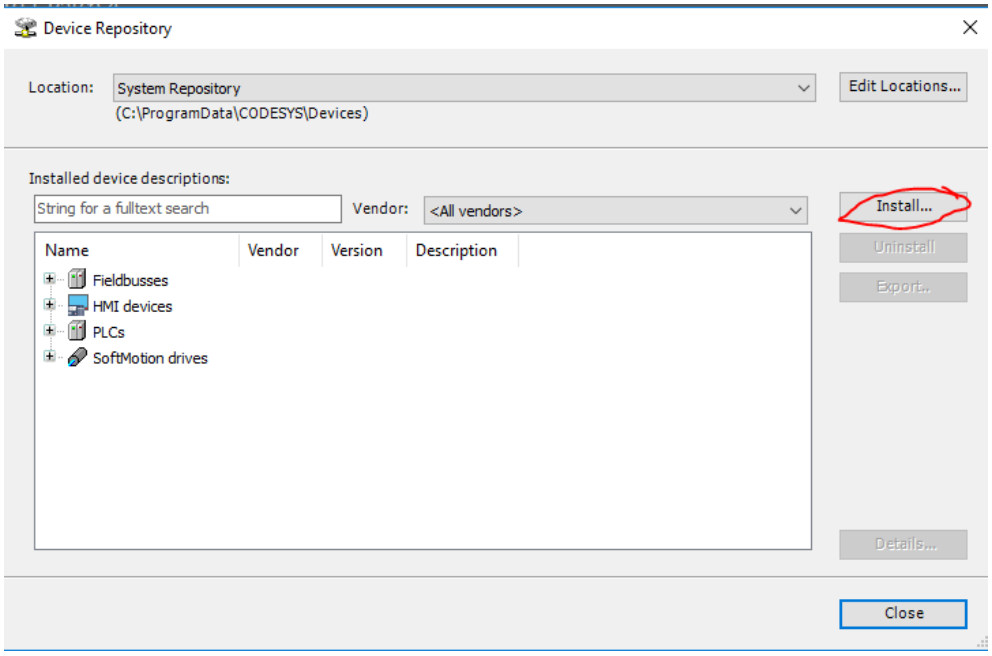
- Egen pc med codesys installert

Arbidsoppdrag – gjennomføring Profinet
Installer CODESYS 3.5 V17
Installer WinPcap version is 4.1.3 (https://www.winpcap.org/install/default.htm), denne finner du også i fagbiblioteket.

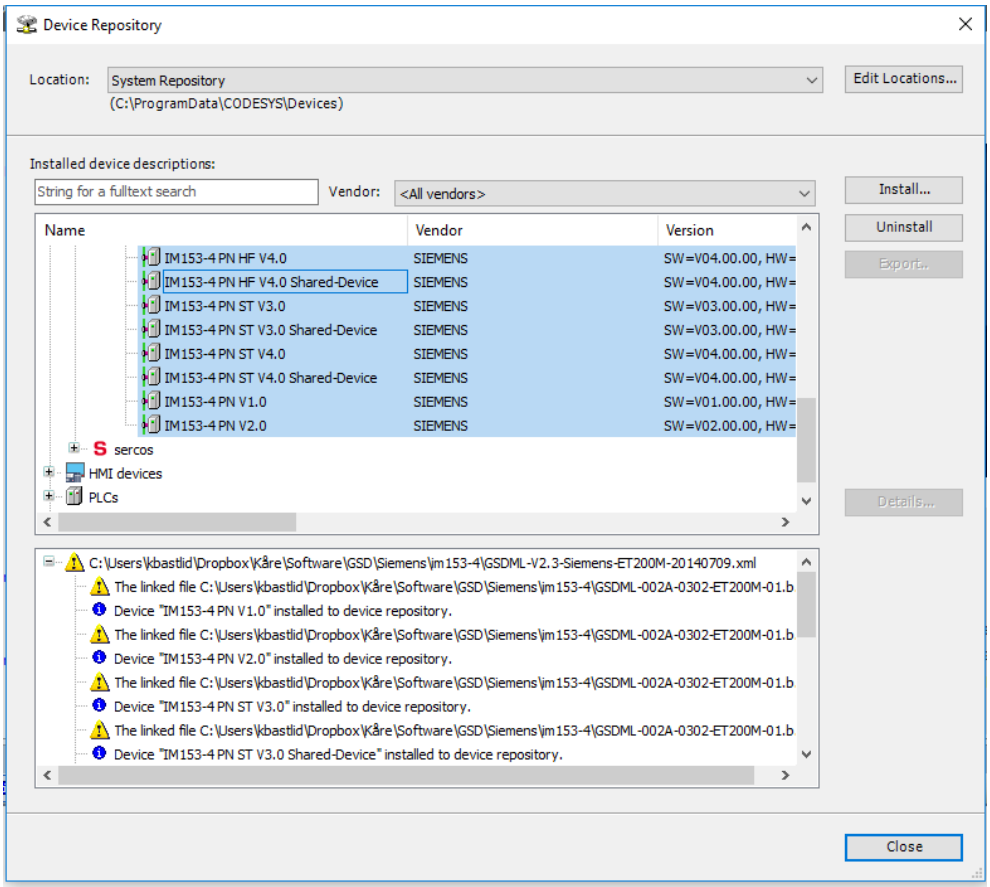


Restart PC
Start CODESYS
Installer GSD fil for Siemens IM153-4

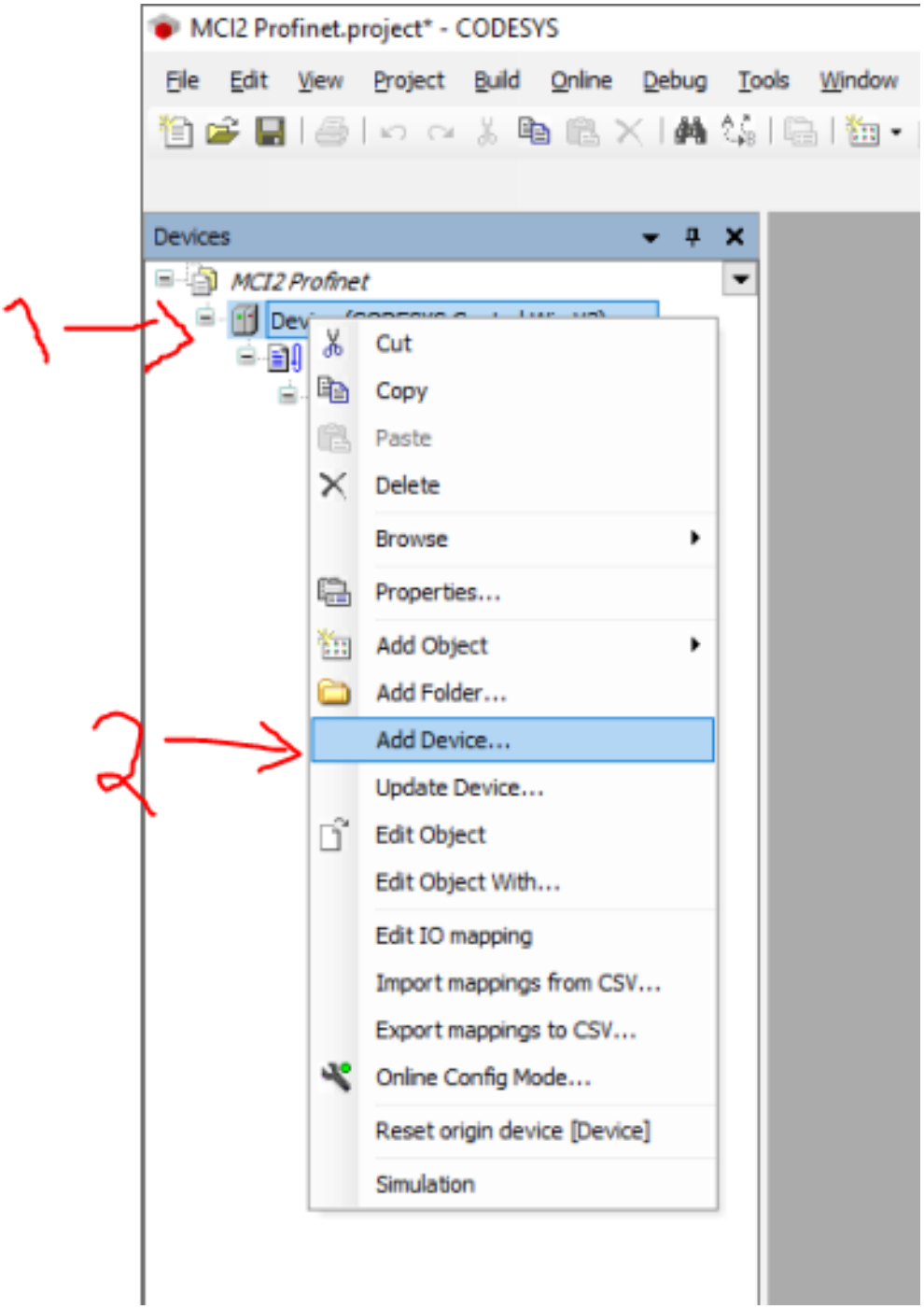


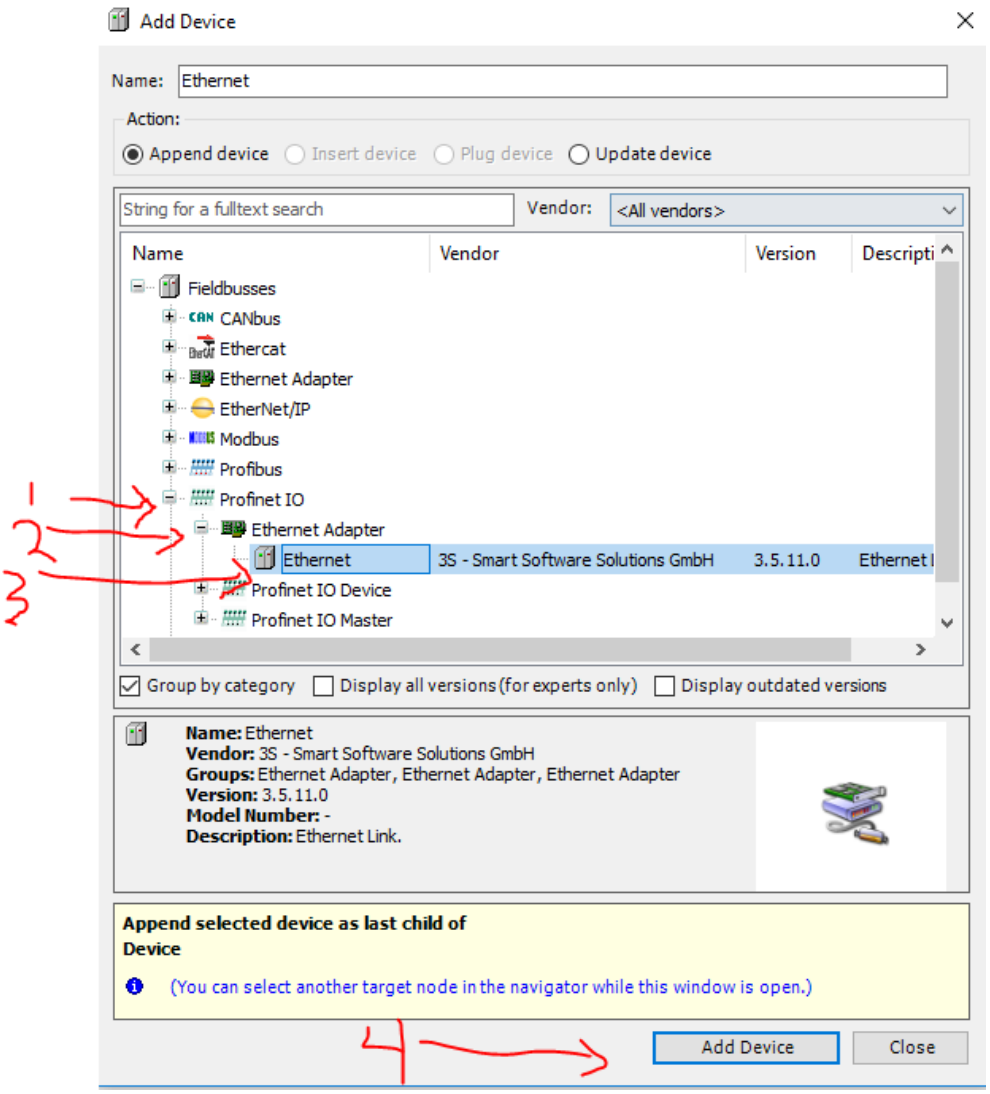


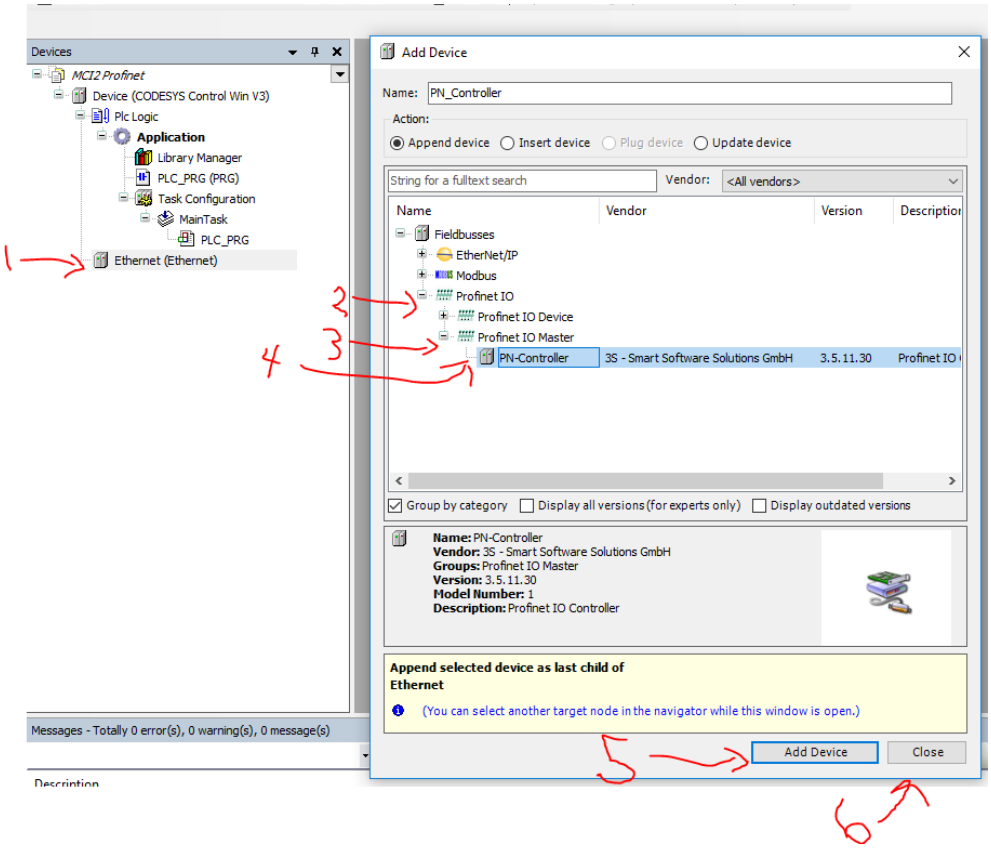
Velg denne lokasjonen i fagbiblioteket



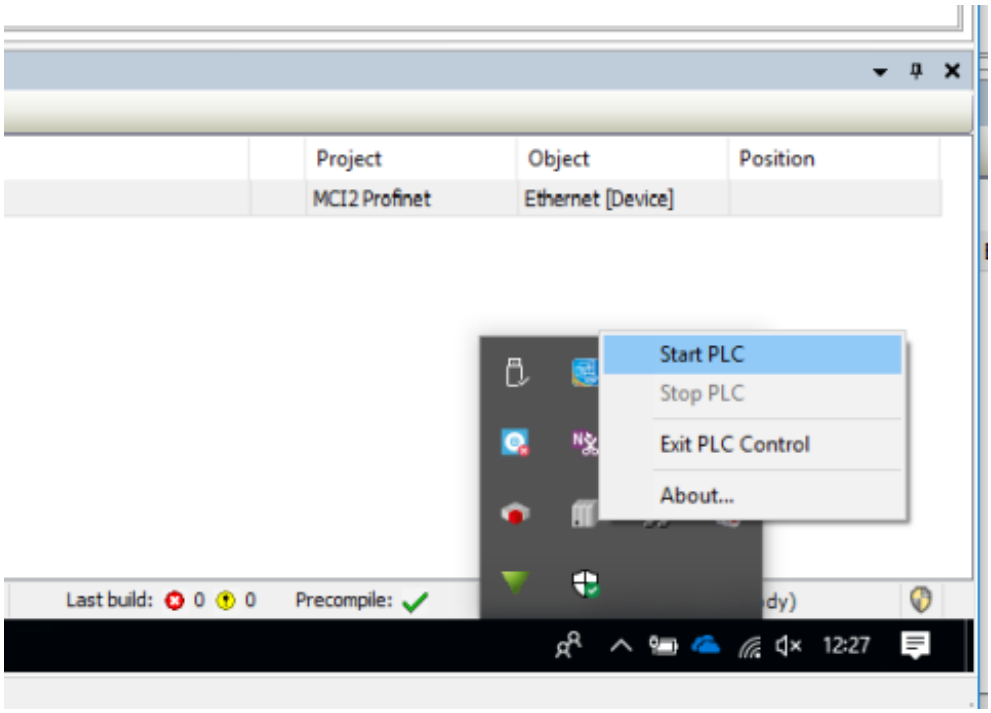
Lukk vinduet etter installasjonen er fullført
Lag et nytt prosjekt MCI2 Profinet
Legg til Profinet

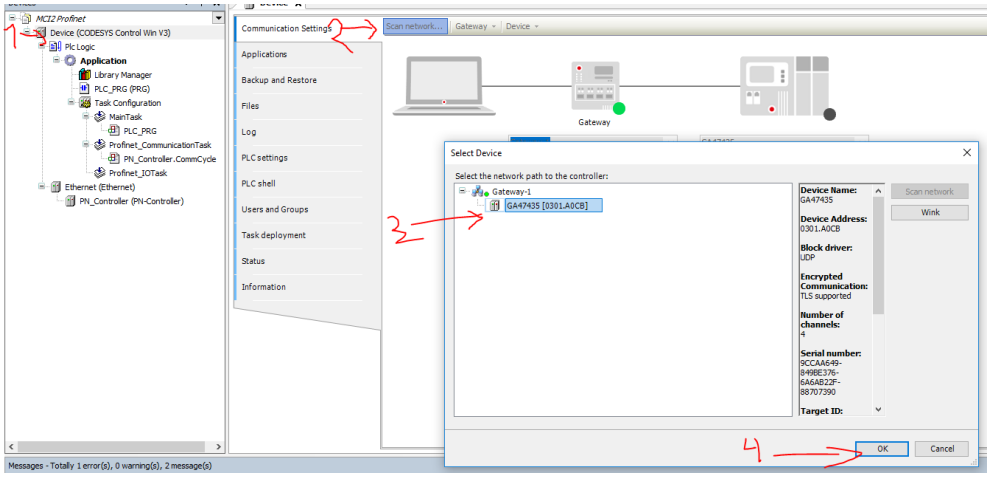

MCI2 Profinet.project* - CODESYS
File
Edit
View
Project
Build
Online
Debug
Tools
Window
Devices
MCI2 Profinet
1
2
Cut
Copy
Paste
Delete
Browse
Properties...
Add Object
Add Folder...
Add Device...
Update Device...
Edit Object
Edit Object With...
Edit IO mapping
Import mappings from CSV...
Export mappings to CSV...
Online Config Mode...
Reset origin device [Device]
Simulation

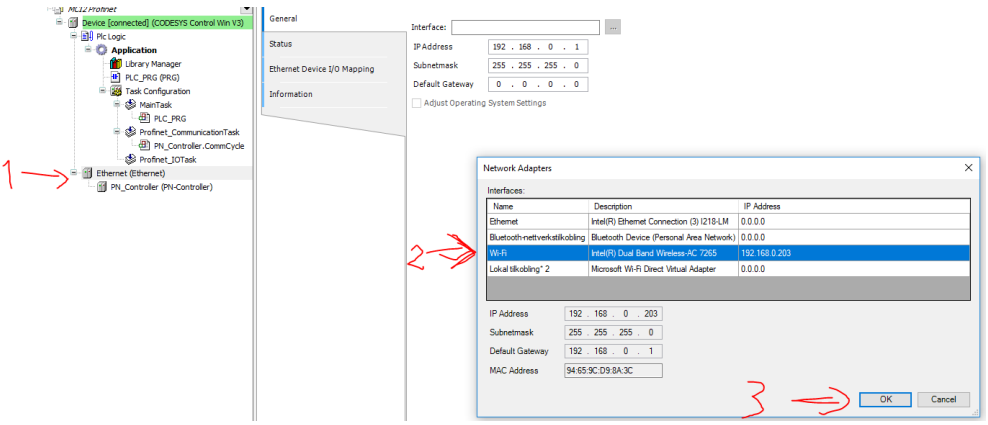

Add Device
Name: Ethernet
Action:
Append device
Insert device
Plug device
Update device
String for a fulltext search
Vendor: <All vendors>
Name
Vendor
Version
Descripti
Fieldbusses
CANbus
Ethercat
Ethernet Adapter
EtherNet/IP
Modbus
Profibus
Profinet IO
Ethernet Adapter
Ethernet
3S - Smart Software Solutions GmbH
3.5.11.0
Profinet IO Device
Profinet IO Master
Group by category
Display all versions (for experts only)
Display outdated versions
Name: Ethernet
Vendor: 3S - Smart Software Solutions GmbH
Groups: Ethernet Adapter, Ethernet Adapter, Ethernet Adapter
Version: 3.5.11.0
Model Number: -
Description: Ethernet Link.
Append selected device as last child of
Device
(You can select another target node in the navigator while this window is open.)
Add Device
Close
1
2
3
4


Devices
MCI2 Profinet
Device (CODESYS Control Win V3)
Plc Logic
Application
Library Manager
PLC_PRG (PRG)
Task Configuration
MainTask
PLC_PRG
Ethernet (Ethernet)
Messages - Totally 0 error(s), 0 warning(s), 0 message(s)
Add Device
Name: PN_Controller
Action:
Append device
Insert device
Plug device
Update device
String for a fulltext search
Vendor: <All vendors>
Name
Vendor
Version
Fieldbusses
EtherNet/IP
Modbus
Profinet IO
Profinet IO Device
Profinet IO Master
PN-Controller
3S - Smart Software Solutions GmbH
3.5.11.30
Group by category
Display all versions (for experts only)
Display outdated versions
Name: PN-Controller
Vendor: 3S - Smart Software Solutions GmbH
Groups: Profinet IO Master
Version: 3.5.11.30
Model Number: 1
Description: Profinet IO Controller
Append selected device as last child of
Ethernet
(You can select another target node in the navigator while this window is open.)
Add Device
Close
1
2
3
4
5
6

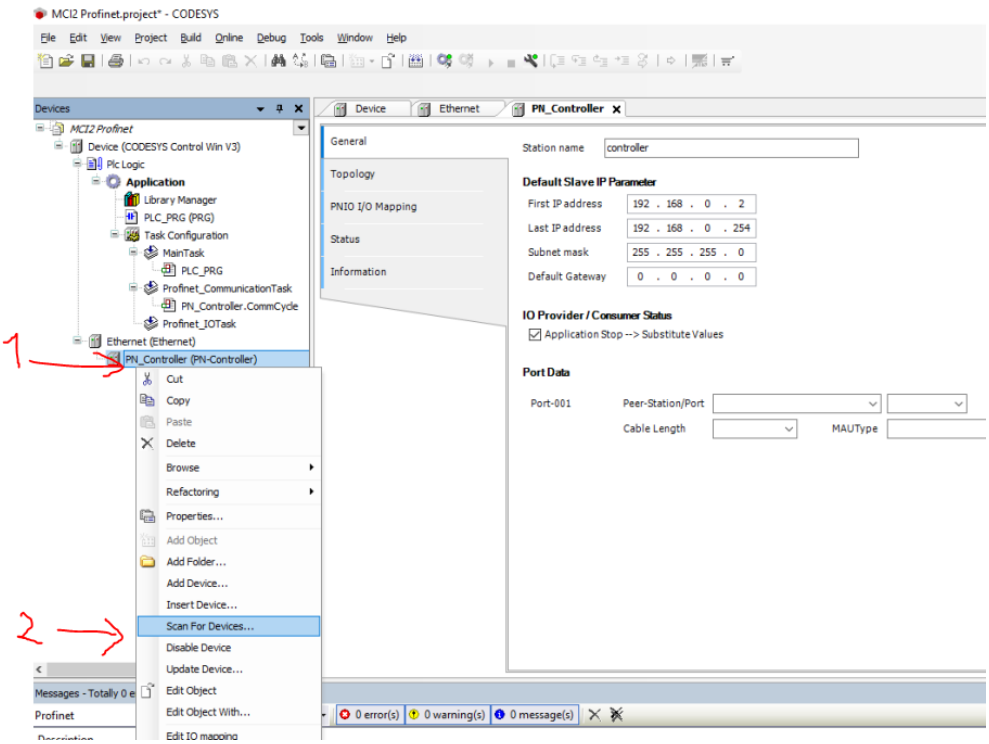
Start PLC

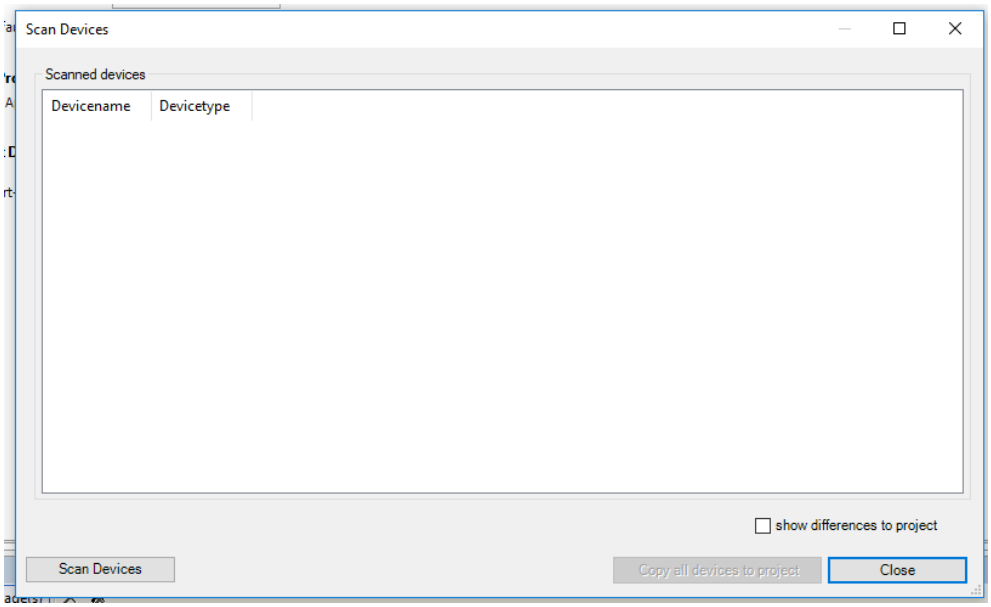

Project
Object
Position
MCI2 Profinet
Ethernet [Device]
Start PLC
Stop PLC
Exit PLC Control
About...
Last build: 0 0
Precompile:
12:27

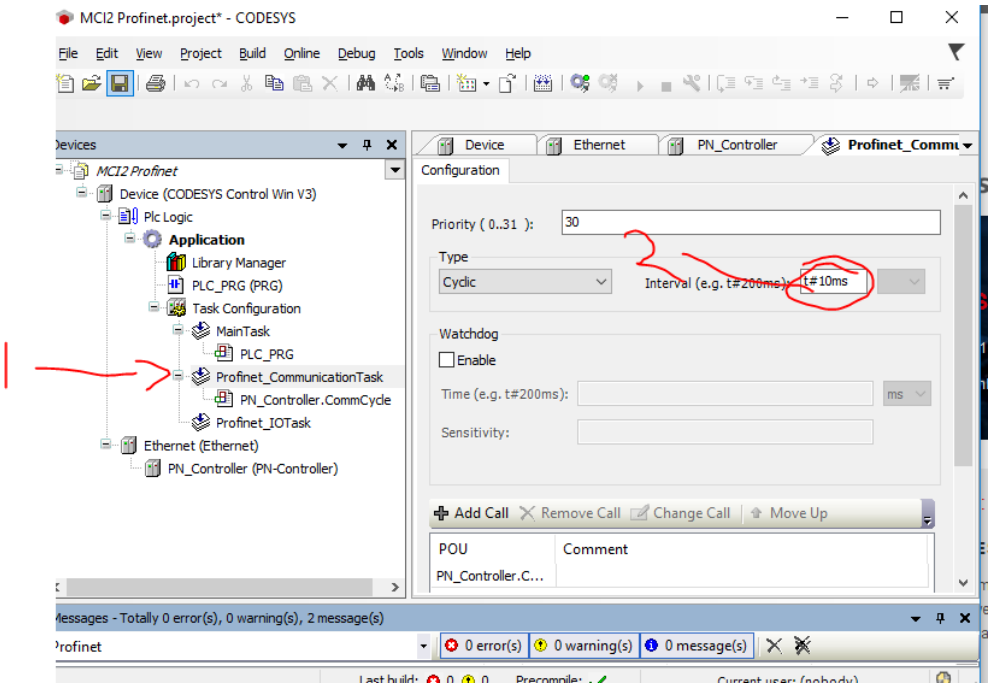




Om du bruker kabel må du velge Ethernet i istedenfor.







Set tiden til 4ms, det er den høyeste tiden IM153-4 kjører.
Oppgaven

Få kontakt med PLS D22 og demonstrer for lærer at du kan sette utgangen og ta imot inngangssignaler.

Arbidsoppdrag – dokumentasjon

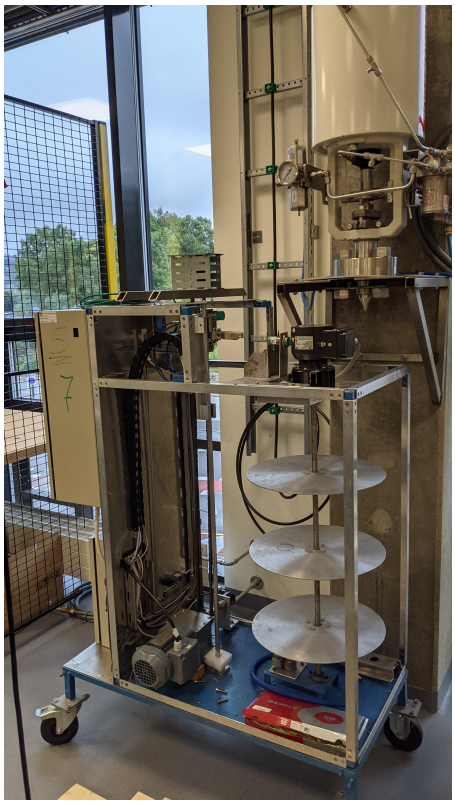
1. Beskriv hvordan du planla, gjennomførte og dokumentere jobben. Forklar eventuelle avvik dere måtte observere under oppdraget.

file i04834

Oppgave 9

Kommunikkasjon – Nivå 2
Arbidsoppdrag på Stasjon 7
IO-sjekk på autobar.

Introdusjon



Teorioppgaver

Planlegging

Gjennomføring

Dokumentasjon

Beskriv hvordan du planlegger, gjennomfører og dokumenterer denne jobben.
file i04835l

Styresystemer – Nivå 1

Arbidsoppdrag på Stasjon 11

Oppsett av målesystemer for temperatur

Introdusjon

I dette oppdraget skal du sette opp og kalibrere to målesystemer for temperatur. Det ene systemet skal benytte en transmitter fra GM (D1072S) og det andre en fra Pepperl+Fuchs (KFD2-UT2-Ex1). Begge disse transmitterne er laget for bruk i EX områder det skal vi ikke bry oss med i dette oppsraget.
Begge målesystemene må settes opp med software fra leverandøren:
Pepperl+Fuchs
For å programmere KFD2-UT2-Ex1, trenger du adapteren K-ADP-USB, samt programmet Pactware og Tilhørende DTM.

Programvaren laster du ned gratis hos oss:
PACTware 5.X
DTM Interface Technology

Veiledning Installation and configuration guide Device Type Manager (DTM)
Eventuellt trenger du kanskje denne, men prøv først uten:
Microsoft .NET

GMI
Software til GMI D1072s finner du i mappen software vi bruker og verktøy (SWC1090.exe)
Målesystemene skal ha følgende måleområder:

- GMI 10-40°C
- Pepperl+Fuchs -5-100°C

Det må fylles ut kalibreringssertifikat for begge målesystemene.

Arbidsoppdrag – teorioppgaver

Leseoppgave

Arbidsoppdrag – planlegging
Arbidsoppdrag – gjennomføring
Arbidsoppdrag – dokumentasjon
file i04841

Målesystemer – Nivå 1

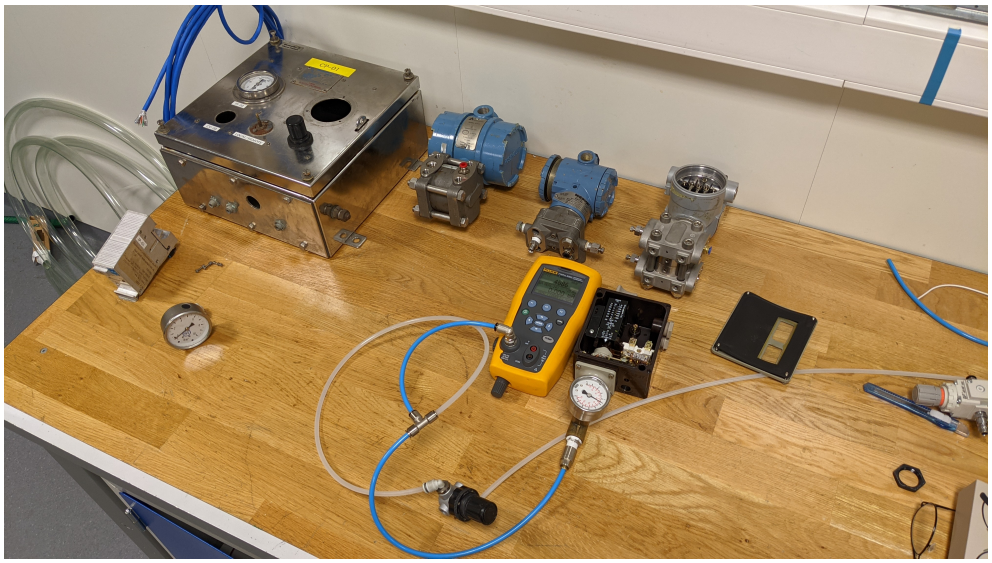
Arbidsoppdrag på Stasjon 10

Oppsett av målesystemer for trykk

Introdusjon
I denne oppgaven skal du justere og kalibrere utstyr for måling av trykk tre DP-celler og en trykkbryter.

- Rosemount 3051 manual
- Rosemount Model E1151 Analog DP-celle
- Fuji Eletric FCX-AII serie transmitter manual
- Danfoss RT200 trykkbryter manual

For Fjui Eletric transmitteren skal du tilpasse transmitteren til montering i ulike posisjoner(se manaul), såkalt zero calibration. For resten av transmitterne skal du justere og kalibrere for oppgitte LRV og URV(disse får du av lærer når du starter oppdraget)



Arbidsoppdrag – teorioppgaver

1. Les vedlagt bruksanvisning for å sette dere inn i virkemåten til utstyret.

Arbidsoppdrag – planlegging Eksempel og en verktøysliste for danfoss trykkbryteren kan være følgende:

- Trykkbryter danfoss
- trykkregulator og manometer
- multimeter

Følgende punkter kan være med med i en fremdriftsplan:
Oppgaver

1. Utføre en As found kalibrering
2. Justere inn til nye LRV og URV
3. Utfør en As left kalibrering
4. lagre dokumentasjon for jobben.

Arbidsoppdrag – gjennomføring
Arbidsoppdrag – dokumentasjon

1. Beskriv hvordan du planla, gjennomførte og dokumentere jobben. Forklar hvordan en slik pressostat kan nyttes til en pumpe eller kompressor. Forklar eventuelle avvik dere måtte observere under forsøket.
2. Dokumentasjonen skal inneholde skisser av oppkoblinger
3. Kalibreringsskjema
4. Bilder av oppkoblingene.

file i04842

Målesystemer – Nivå 1

Arbidsoppdrag på Stasjon 9

Oppsett av målesystemer for nivå

Introduksjon
I dette arbeidsoppdraget skal du jobbe med målesystemer for nivå. Du skal utføre sjekk og kalibrering for nivåmålesystemer basert på følgende prinsipper:

- Ultralyd
- Trykk
- Radar

Du skal sette i stand målesystemene for nivå på de tre tankene på denne stasjonen. Alt må koblet opp og virke mot stasjonens styresystem.

Kalibrering og justering range for en smart transmitter Målet med oppgaven er å lære hvordan vi justerer en transmitter for trykk.

Teorioppgaver

Leseoppgave

Planlegging

Gjennomføring

Dokumentasjon

1. Beskriv hvordan du planla, gjennomførte og dokumentere jobben. Forklar eventuelle avvik dere måtte observere under oppdraget.

2. Dokumentasjonen skal inneholde sløyfeskjema av alle sløyfer du har jobbet med.

3. Kalibreringsskjema for alle sløyfene

4. Bilder av transmittere

file i04843

Målesystemer – Nivå 2

Arbidsoppdrag på Stasjon 15

Digitale målesystemer

Introduksjon

I denne oppgaven skal du koble opp og teste ulike digitale målesystemer

1. Mekanisk endebryter med rulle
2. Mekanisk endebryter med fjær
3. Mekanisk endebryter med aksling....
4. kapasitiv nærhetssensor
5. induktiv nærhetssensor
6. Photo sensor

Teorioppgaver

Her er noen tekster der du kan lese deg opp på endebrytere

The Basics of Limit Switches - Eaton guide til endebrytere

Omron komplett katalog

Omron guide til limit switches

ukjent

Planlegging
Gjennomføring
Dokumentasjon

Arbidsoppdrag

Programmering av reguleringsstasjon

Introdusjon

Teorioppgaver

Planlegging

Gjennomføring

Dokumentasjon
Beskriv hvordan du planlegger, gjennomfører og dokumenterer denne jobben.
file i04844

Styresystemer – Nivå 1

Arbidsoppdrag på Egen PC

Optimalisering av simulerte prosesser

Introduksjon I Dette arbeidsoppdraget skal du jobbe med et Reguleringsundervisnings prosjekt for codesys.

Du må starte dette kokumentet i Codesys og gjennomføre:

- Repitisjon Z&N
- Oppgaven Her skal du optimalisere alle prosessene med følgende metoder
  - Tims rule of tumb
  - Z&N
  - Skogestads metode

Teorioppgaver

Planlegging

Gjennomføring

Dokumentasjon
Lag et dokument som viser hvordan du gikk frem for å optimalisere med de ulike metodene. Beskriv hvordan du planlegger, gjennomfører og dokumenterer denne jobben.
file i04853

Oppgave 15

Reguleringstkenikk – Nivå 2

Arbidsoppdrag på Stasjon 3/4

Nivå- og Flow regulering

I dette arbeidsoppdraget skal du sette i drift nivåregulering på en stasjon. Du skal prøve optimalisere PID parametrene på følgende måter

- Z&N svingemetode
- Skogestads metode

I dette arbeidsoppdraget skal du sette i drift strømningsregulering på en stasjon. Du skal prøve optimalisere PID parametrene på følgende måter

- Z&N svingemetode
- Skogestads metode
- Lærerens jukse metode

Teorioppgaver
Planlegging
Gjennomføring
Dokumentasjon
Lag et dokument som viser hvordan du gikk frem for å optimalisere med de ulike metodene.

Arbidsoppdrag

Programmering av reguleringsstasjon

Introdusjon

Teorioppgaver

Planlegging

Gjennomføring

Dokumentasjon
Beskriv hvordan du planlegger, gjennomfører og dokumenterer denne jobben.
file i04857

Pådragsoraner – Nivå 1

Arbidsoppdrag på Stasjon 13

Kalibrering og vedlikehold av ventil

Introduksjon I Dette oppdraget skal du utføre vedlikehold og kalibrering av en reguleringsventil. På en annen reguleringsventil skal du snu funksjonen fra normålt åpen til normalt lukket.

Introdusjon

Teorioppgaver

Planlegging

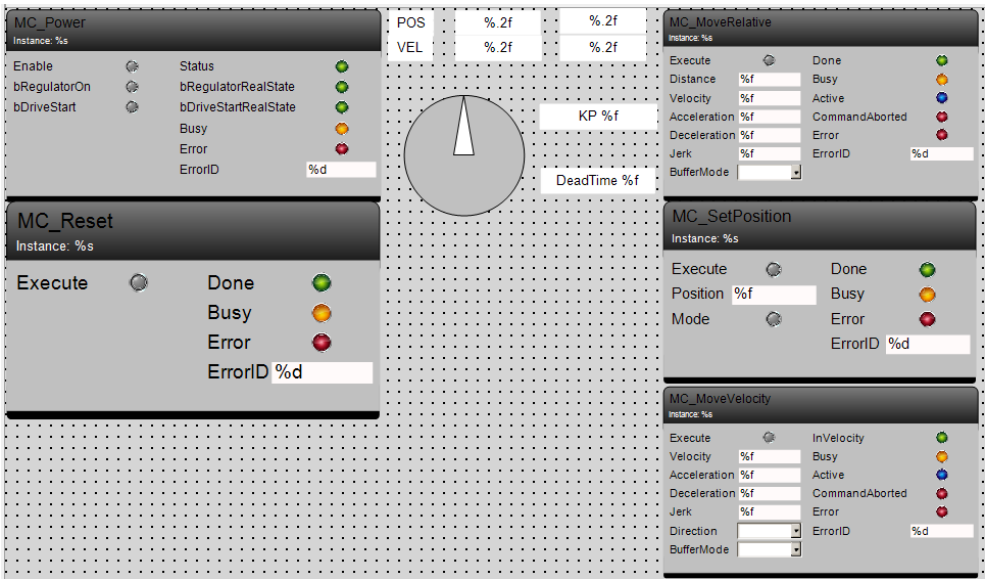
Gjennomføring

Dokumentasjon

Beskriv hvordan du planlegger, gjennomfører og dokumenterer denne jobben.
file i04857

Pådrag – Nivå 2
Arbidsoppdrag på Stasjon 20
Servomotor

Introduksjon I dette arbeidsoppdraget skal du starte opp og styre en servomotor ved hjelp av motion controll blokker. .



Teorioppgaver

Function Blocks for Motion Control

Planlegging
Gjennomføring Codeys prosjekt **Dokumentasjon**
Beskriv hvordan du planlegger, gjennomfører og dokumenterer denne jobben.

Arbidsoppdrag
Programmering av reguleringsstasjon

Introdusjon

Teorioppgaver

Planlegging

Gjennomføring

Dokumentasjon
Beskriv hvordan du planlegger, gjennomfører og dokumenterer denne jobben.
file i04858

Industrirobot – Nivå 1
Arbidsoppdrag på egen PC
Grunnleggende om robot

I dette oppdraget trenger du to ulike robotstasjoner til robotstudio. Disse finner du her:

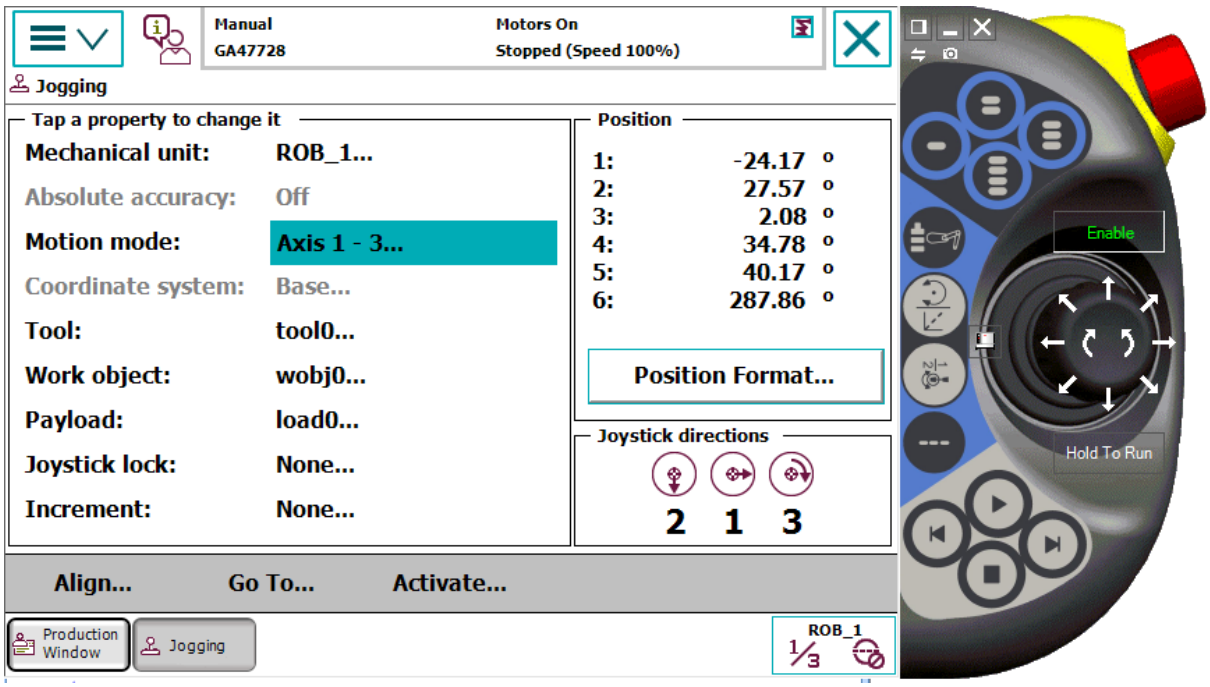
Robotstasjon1

Robotstasjon2

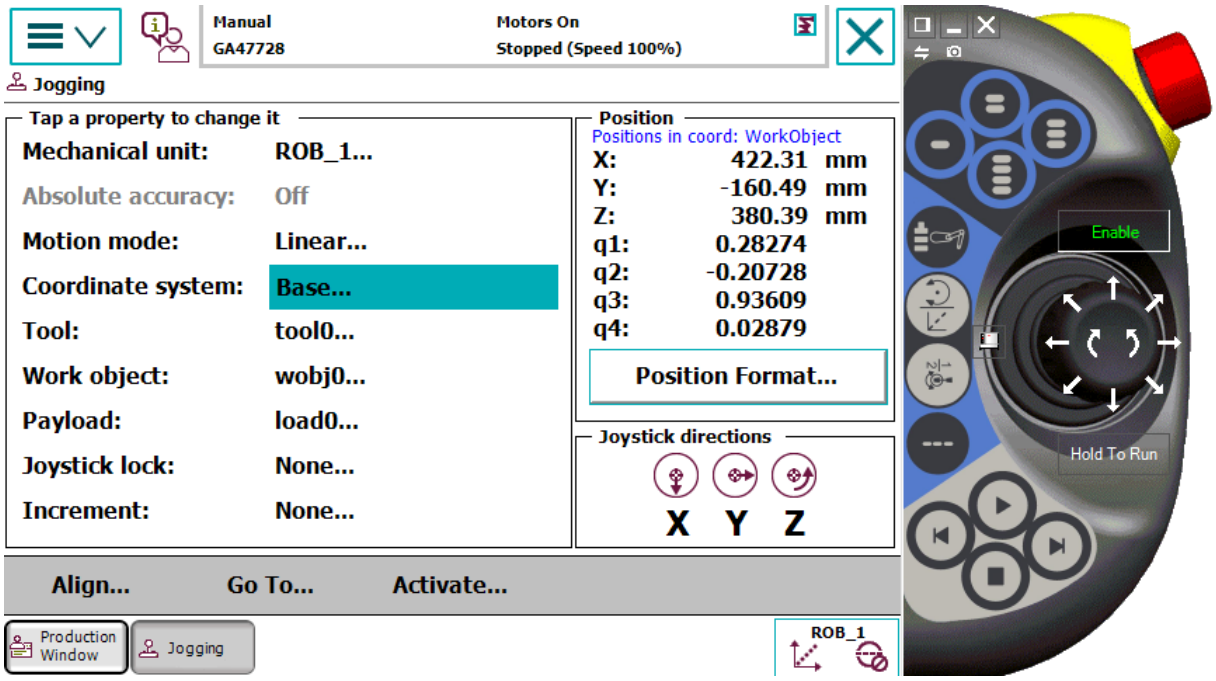
Arbeidsoppdrag - Jogging og TCP
Jogging

For å komme til jogge vindu trykker du følgende:

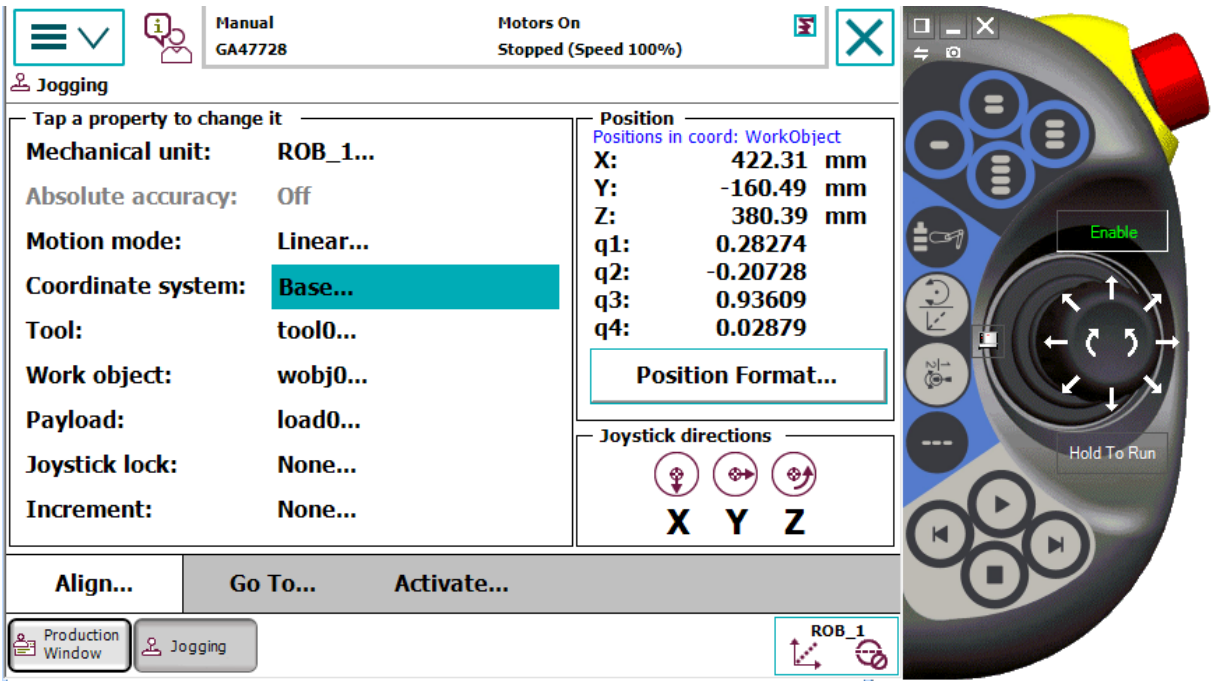
Prøv ut de forskjelleige Motion Mode: Axis 1 - 3, Axis 4-6, Linear og Reorient. Finn ut hva som er forskjellen og forbred deg på å forklare for andre denne forskjellen.



Jogg robot i forskjellige koordinatsystem, Base/ tool/Workobject. Finn ut hva som er forskjellen og forbred deg på å forklare for andre denne forskjellen.



Align robot vil rette verktøyet ut bruker mot den nærmeste aksen x, y eller Z. Prøv ut denne funksjonen.



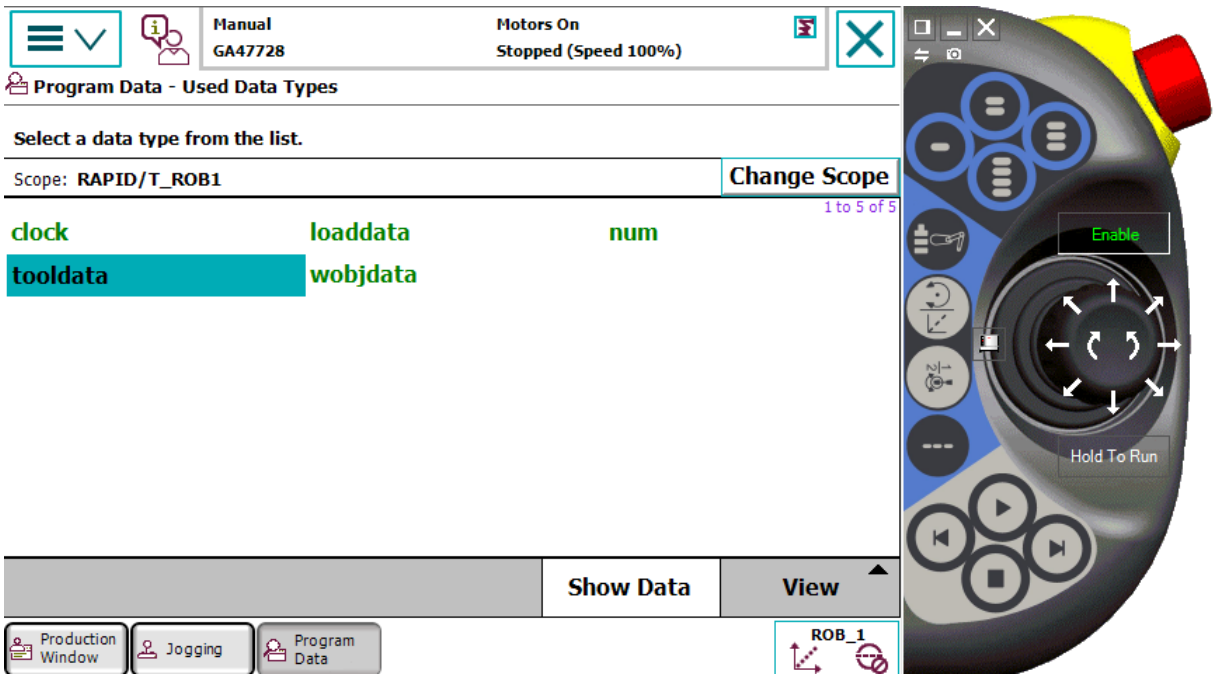
Hvilken annen informasjon får du fra jogge vinduet?

TCP (Tool Center Point)

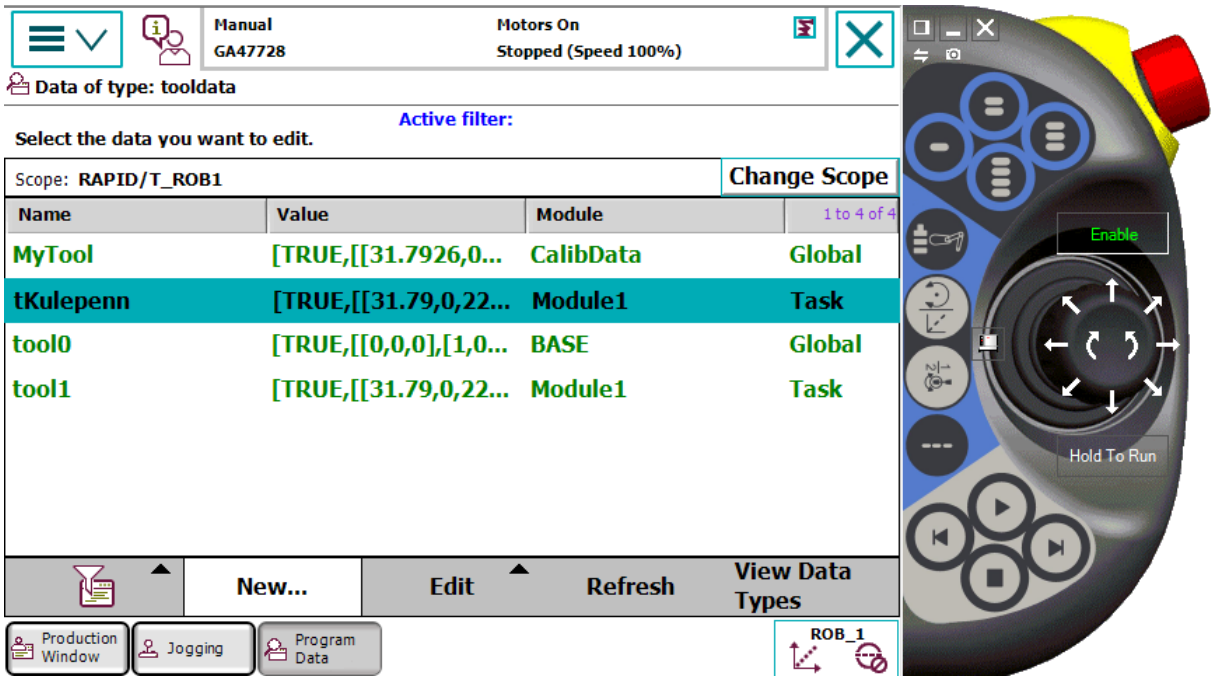
Mål: Lære å opprette TCP manuelt og ved hjelp av robot.

Å sette tool center point er en måte å gjøre roboten klar ovor senter på verkøyet du bruker er. For å plassere verktøyer er vi helt avhengig av at dette er definert.

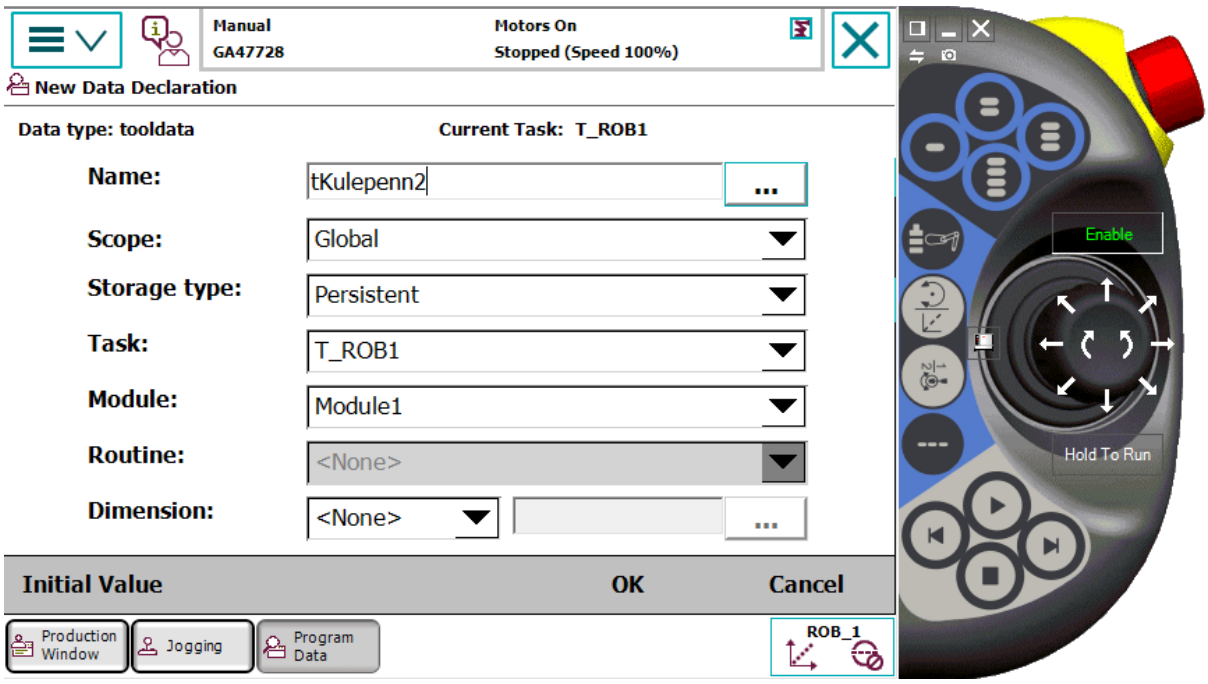
Velg program data og velg tooldata



Trykk på new.



Vi kaller TCP ‘en «tKulepenn2». Sørg for at du velger samme alternativ som på bildet.

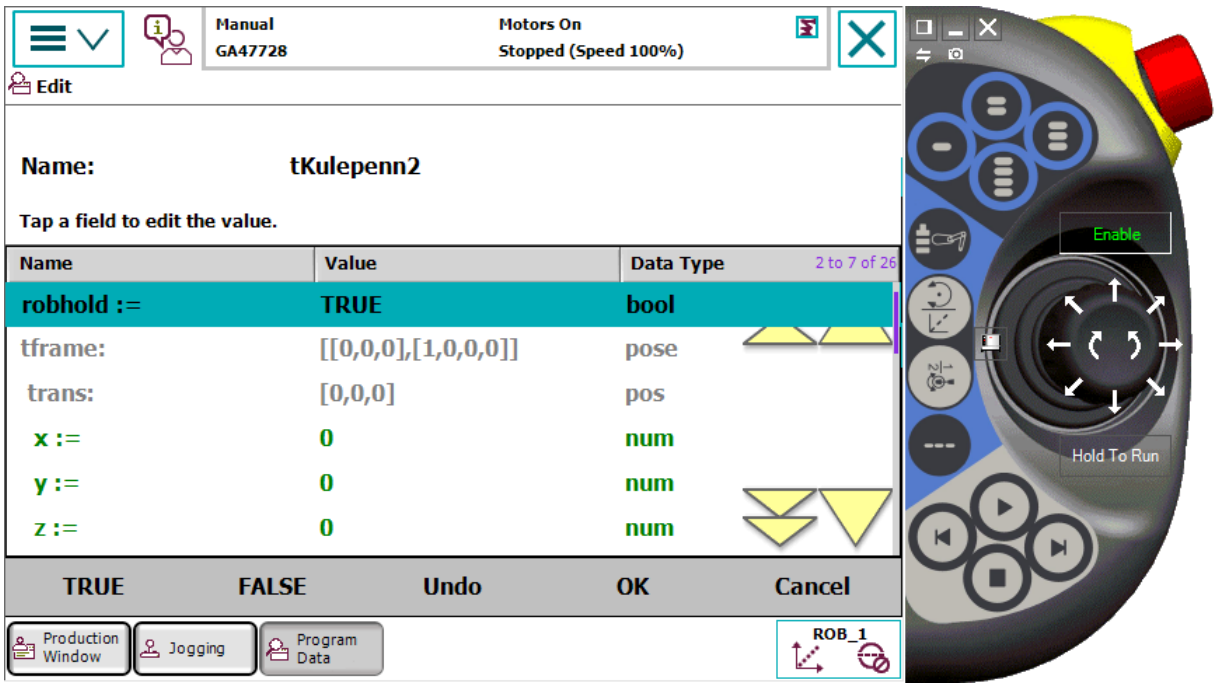


Vi skal nå legge inn avstanden fra flens på robot til tuppen av kulepennen.
Trykk OK.

Velg Inital Value og legg inn følgende verdier:

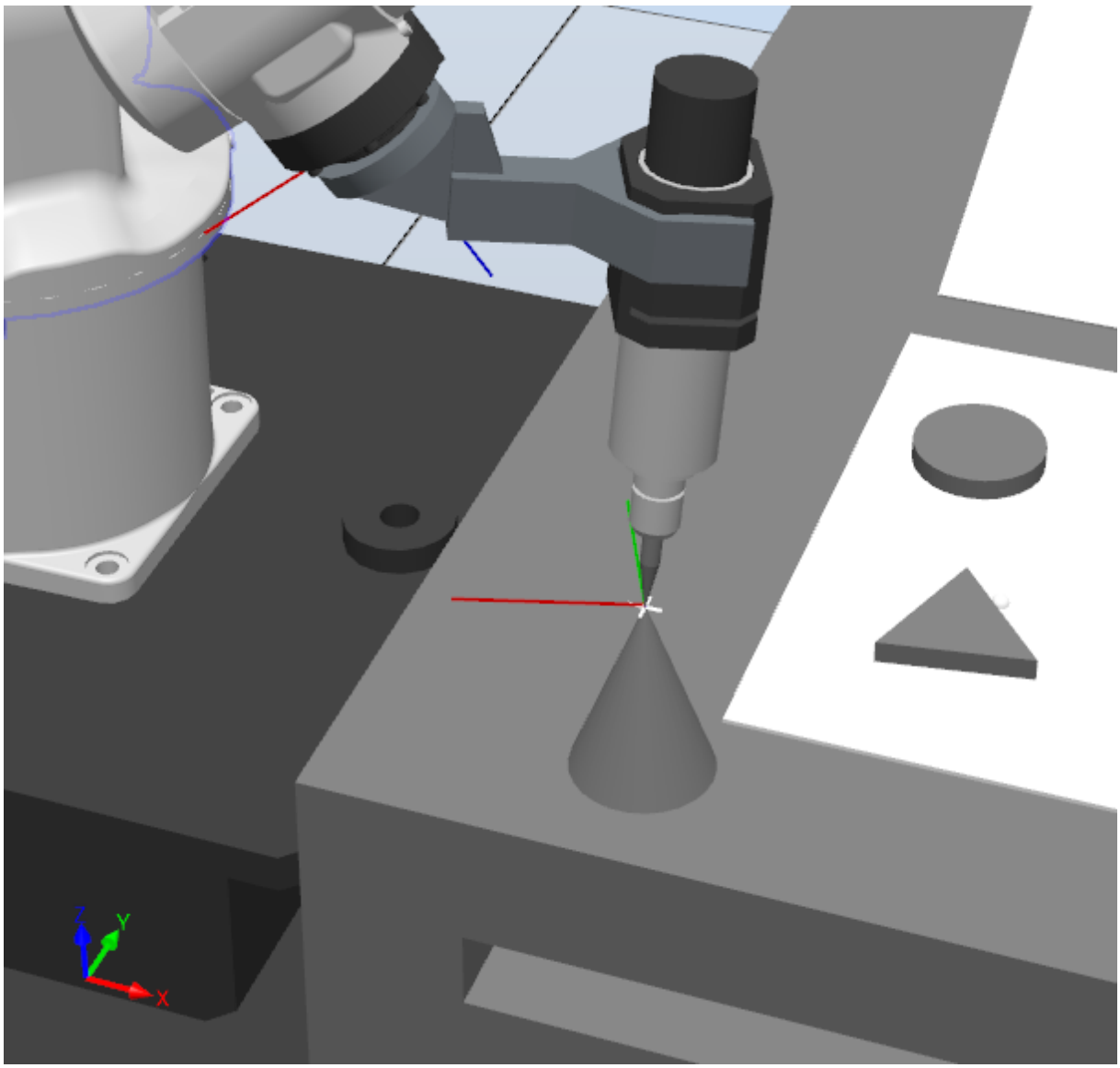
- Toolframe Z = 70 (mm) og X = 130 (mm).
- Mass = 1 kg.
- COG Z= 100 mm.

NB! Dersom enn glemmer å definere vekt og COG, vil du ikke kunne bruke det aktuelle «tool»

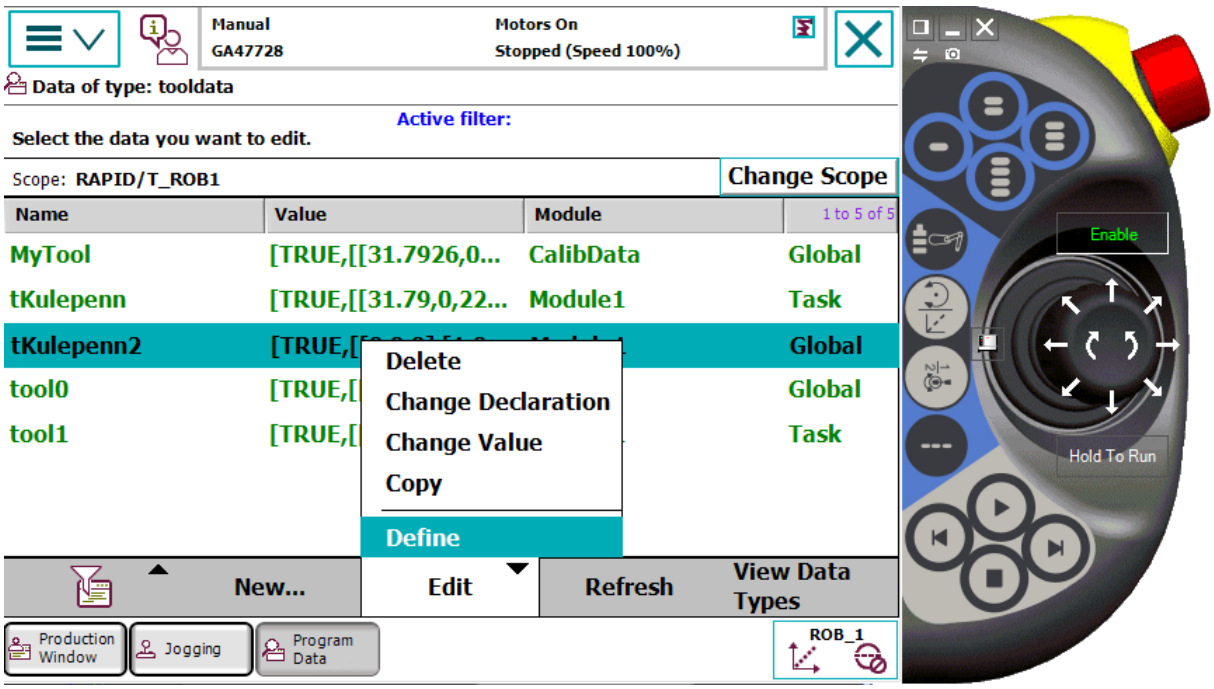


Trykk OK
Jog roboten I reorient og se hvordan det virker

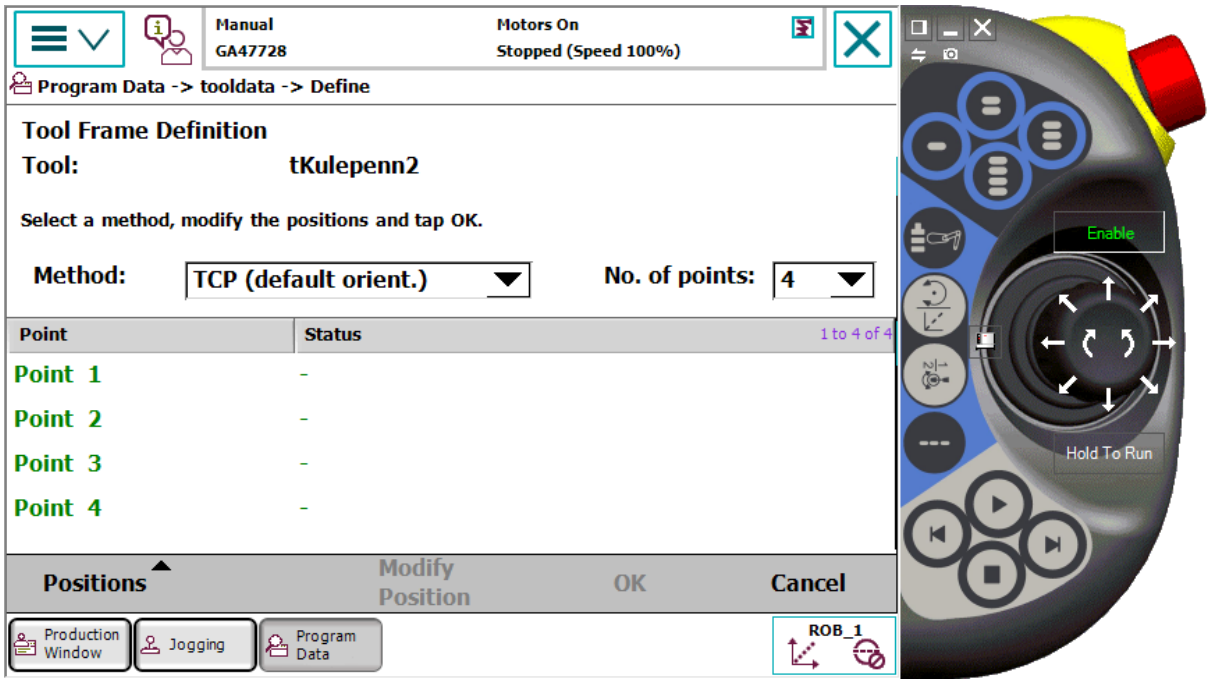
Definer tcp ved hjelp av robot (4punkt).



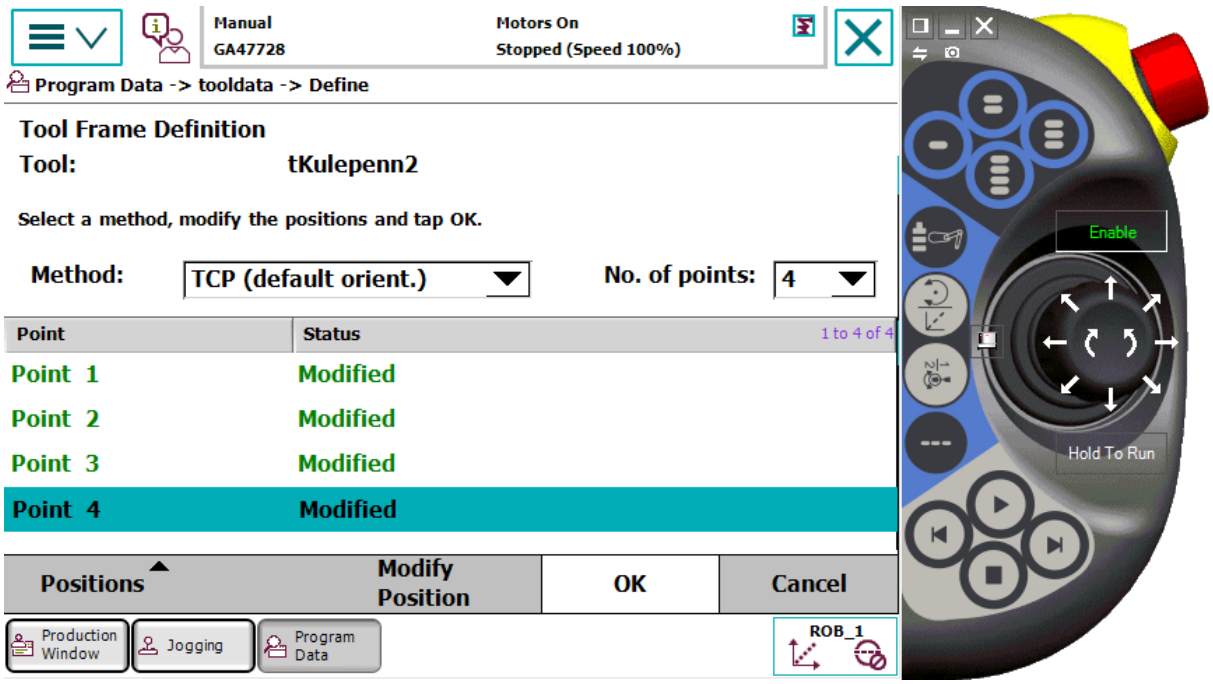
Marker tKulepenn2, trykk edit og velg define.



Jog robot spiss mot spiss på 4 forskjellige måter (se første bilde i oppgave) Størst vinkel mellom punktene vil gi den mest nøyaktige TCP.
Husk å trykk «modify position» for hver gang du står spiss mot spiss.



Når du har satt alle 4 posisjonene, trykk OK.



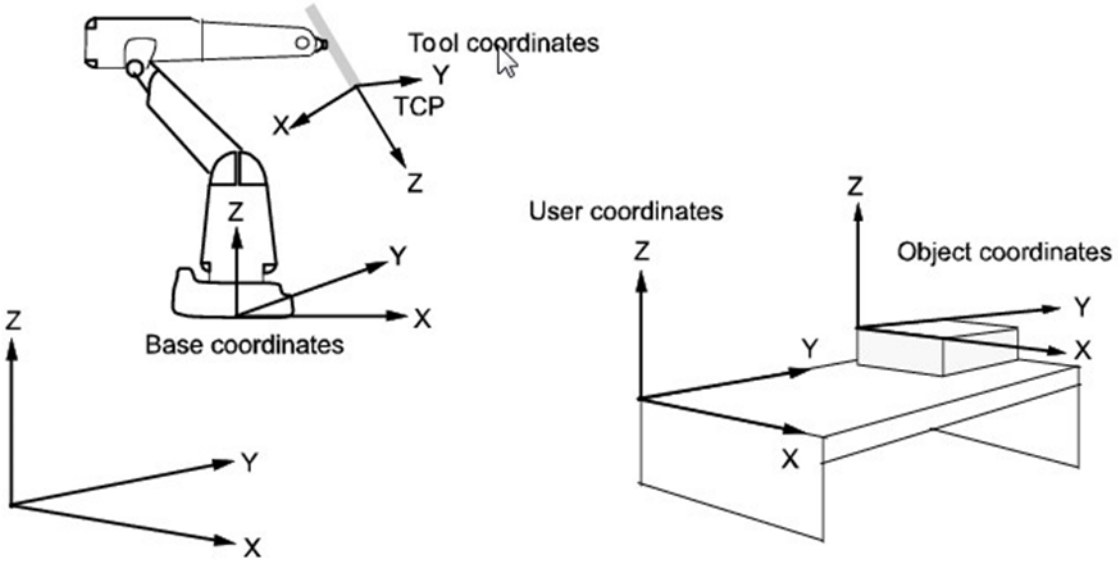
Robot vil nå kalkulere krysningspunkt for alle posisjonene, og lage en TCP ut av resultatet.

TIPS! Etter du har trykket OK, vil du få opp en side som gir deg alle de utregna verdiene. Mean error bør være under 0,5, da har du en nøyaktig TCP.

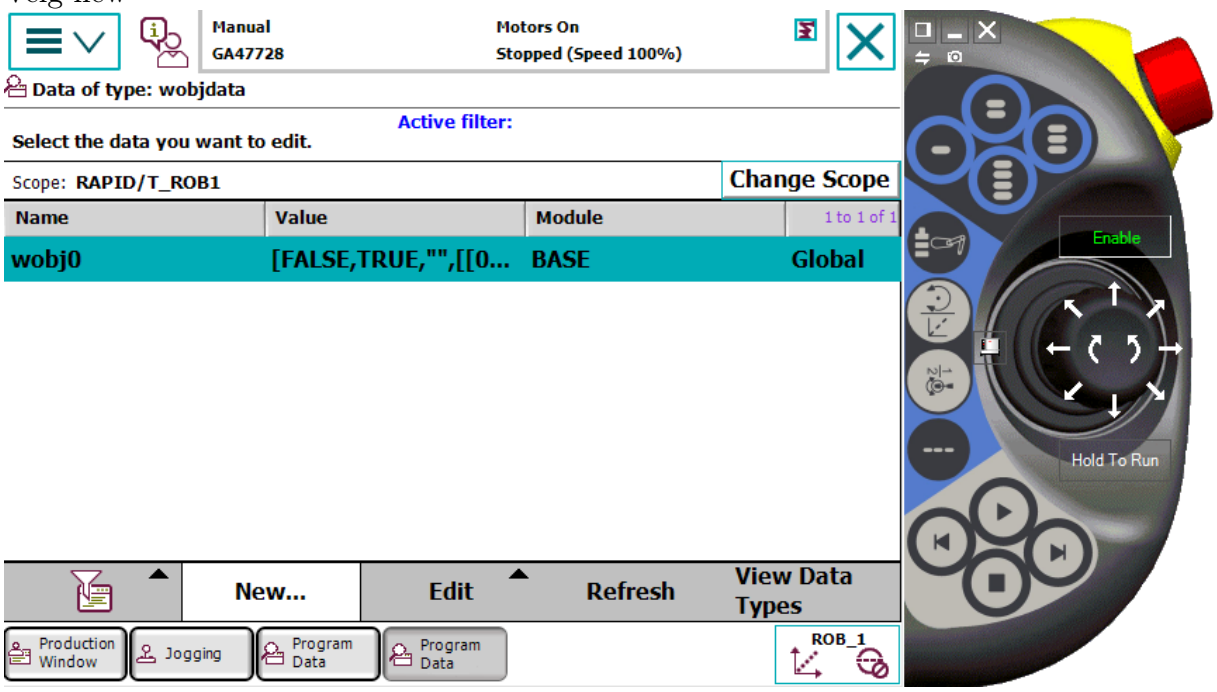
Sjekk: Lærer sjekker om eleven kan utføre:

- Jogging
- Bytte av workobjekt
- Reorientering av tool
- Align av tool
- Vise at reorient virker på tool om elev har laget selv.

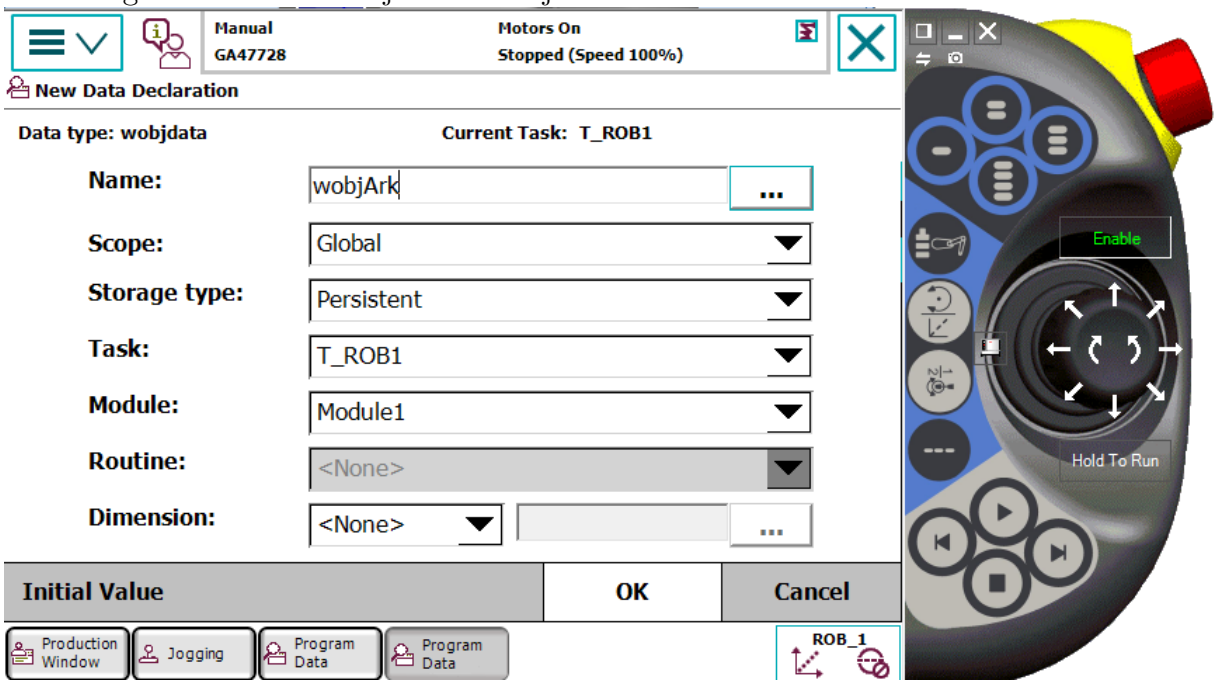
Arbeidsoppdrag 2 - Workobject og bevegelser
Workobject
Mål: Lære å opprette ett work object, og definere origo (nullpunkt)



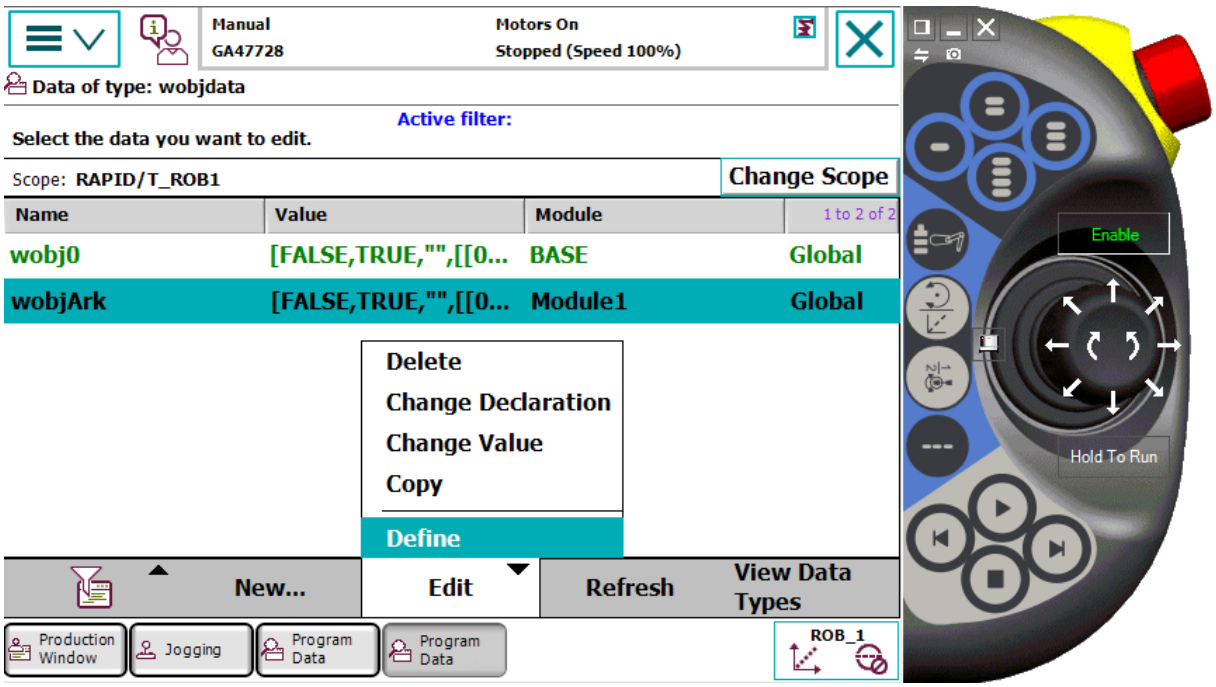
Velg program data i hovedmeny, trykk deretter på wobjdata.
Velg new



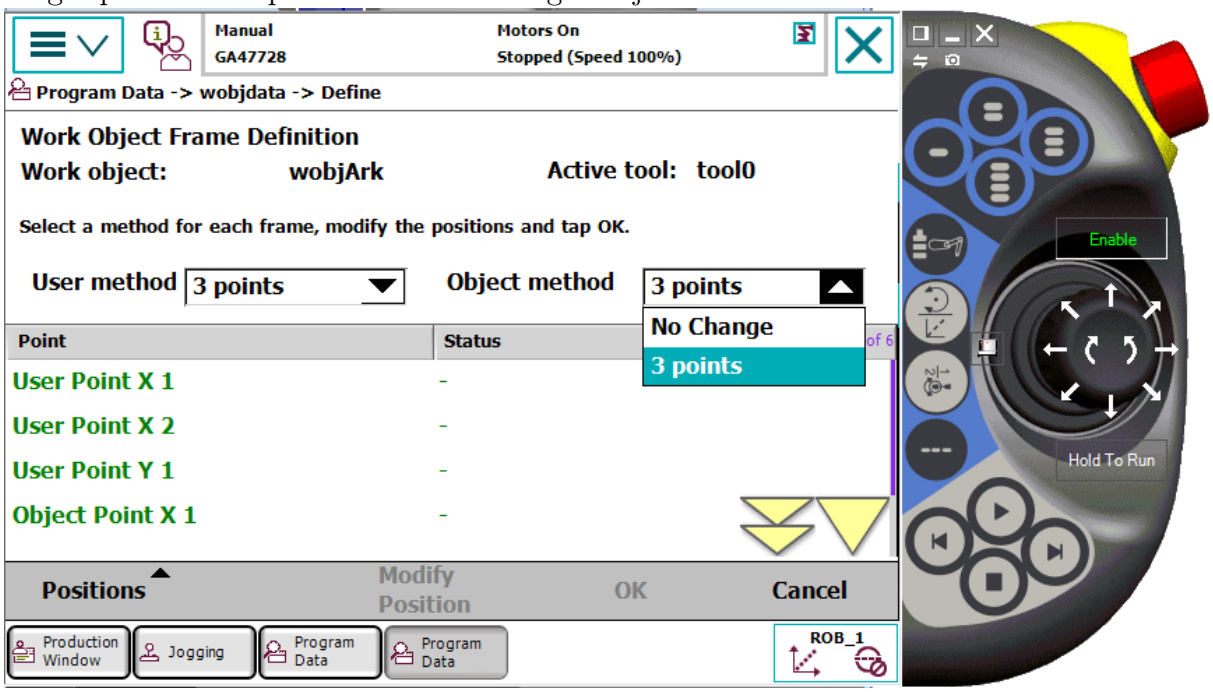
3. Vi velger å kalle work objectet «wobjArk».



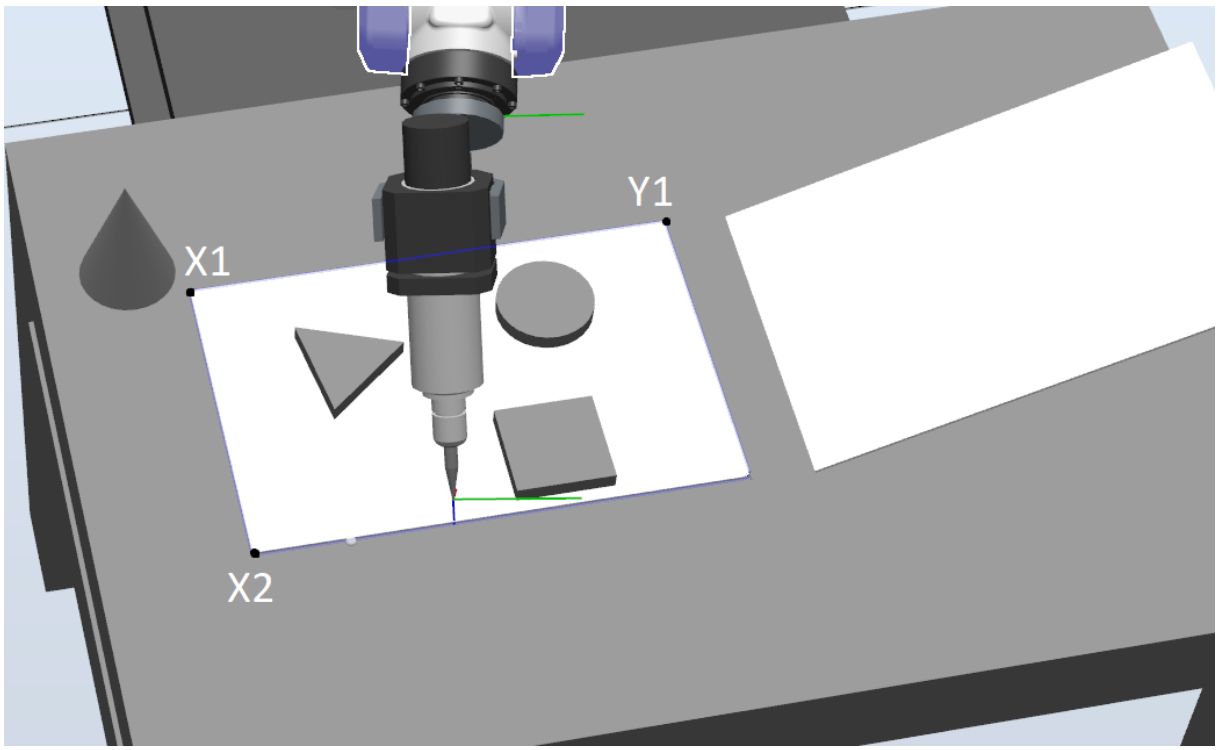
Trykk OK
Vi skal nå definere null punktet. Marker wobjArk, trykk på Edit, og velg Define.



Velg 3 points både på «User method» og «Object method»

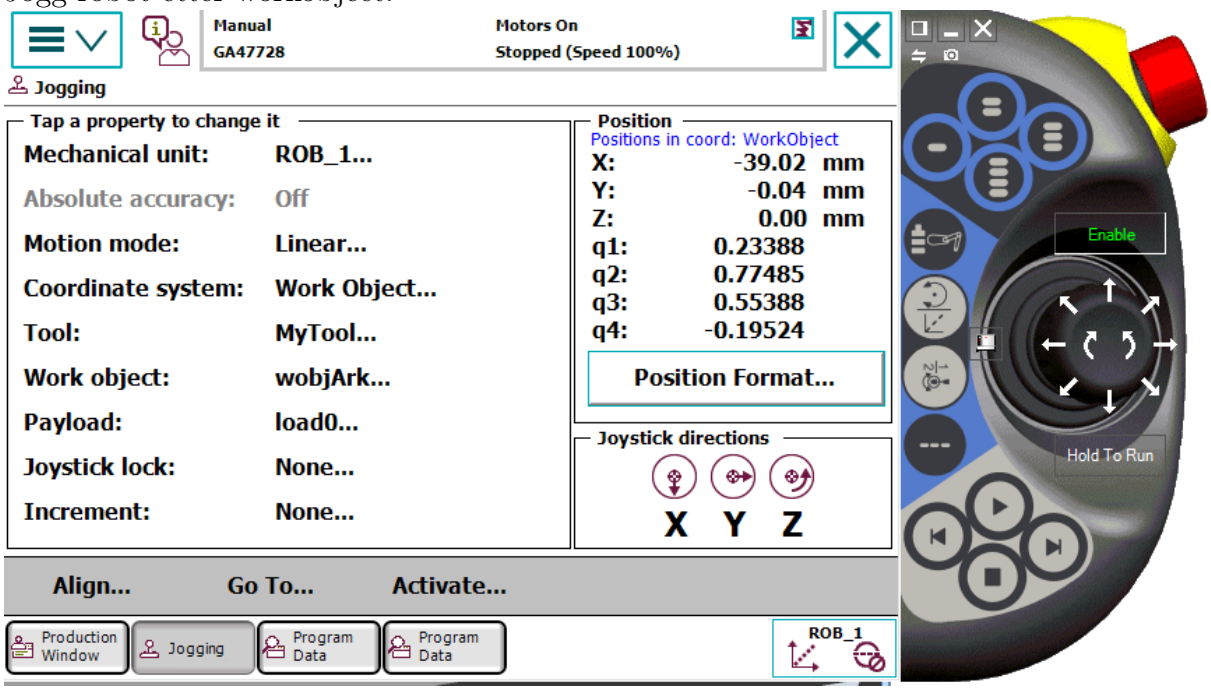


Vi skal nå definere origo for work objectet, jog robot til X1, marker User point X1 og trykk Modify position. I samme position trykk modify position for object point X1. Repeter for X2 og Y1.



Trykk OK

Jogg robot etter workobject.



«Align» robot etter workobject

Lag ett nytt workobject på en skrå flate (For eksempel en perm).

Jogg robot etter workobject OBSERVER.

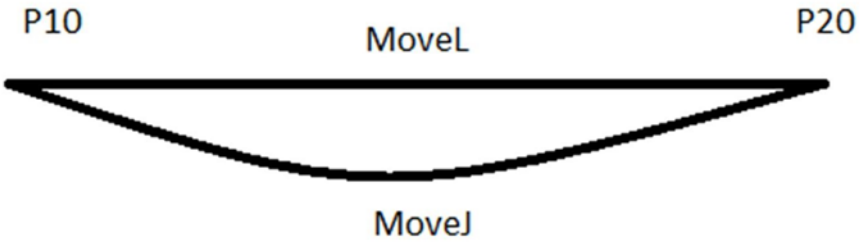
«Align» robot etter workobject.

Bevegelser

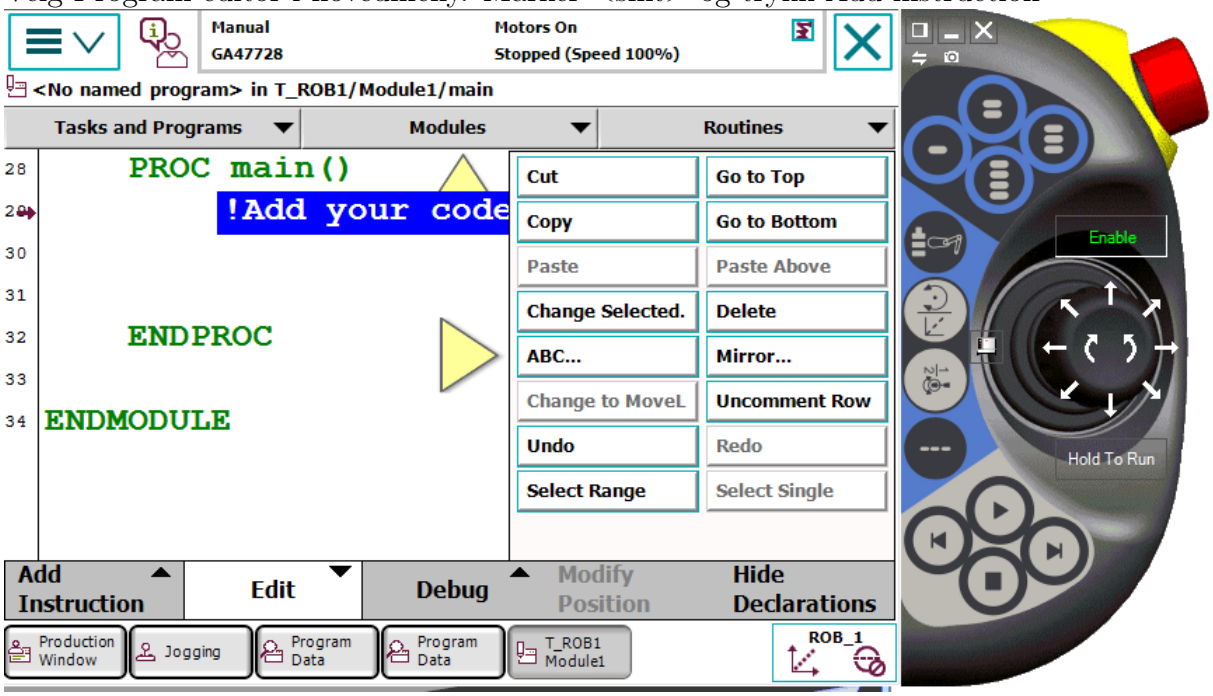
Mål: Lære bruk og forskjell på MoveL og MoveJ.

Roboten skal nå kjøre lineært fra en posisjon (p10) til en annen (p20), og deretter tilbake med en MoveJ bevegelse.

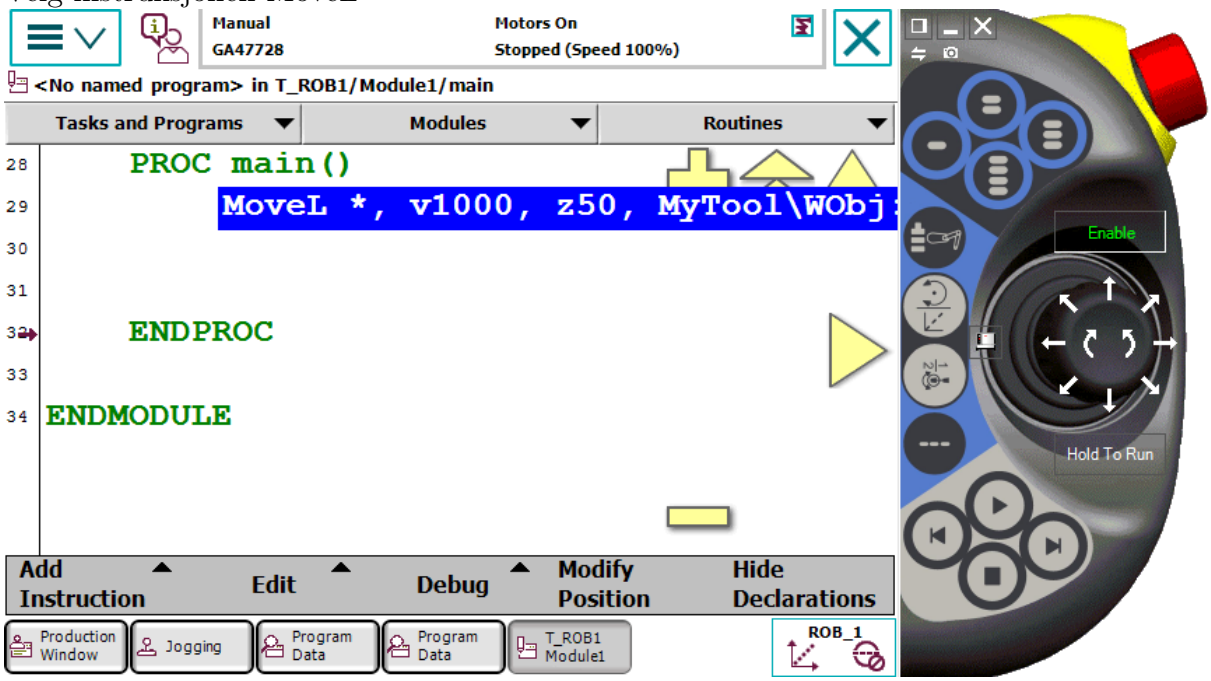
Bevegelsen vil se slik ut:



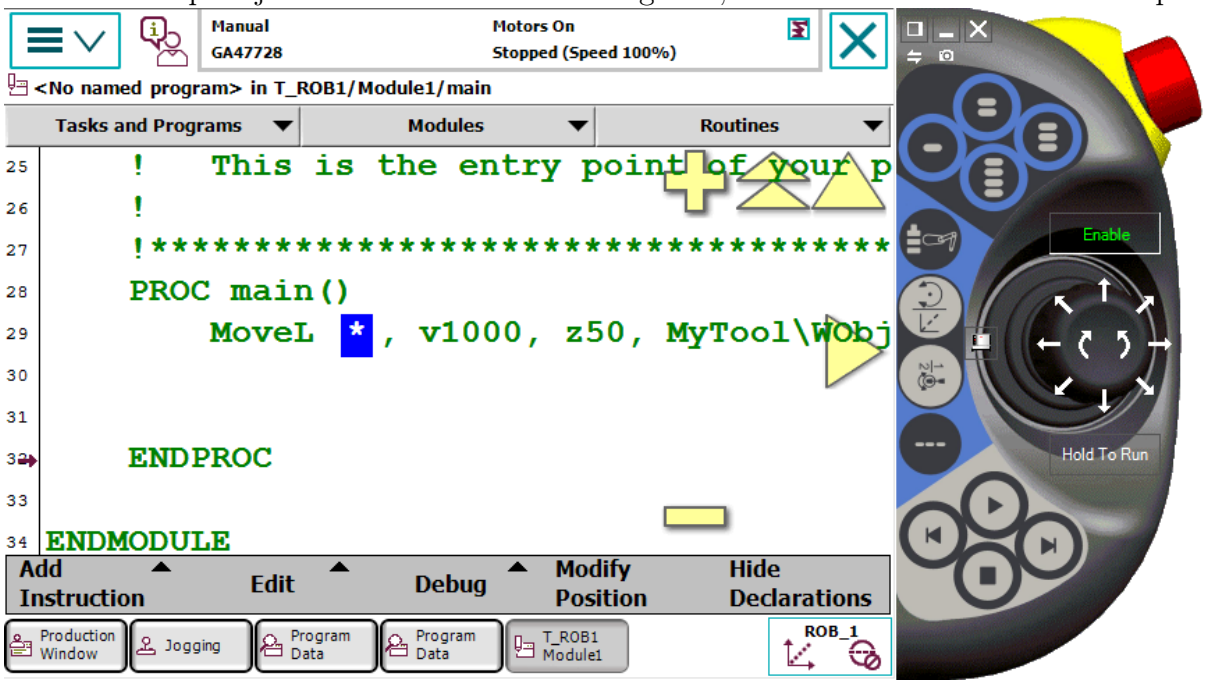
Jog robot til startposisjon, og sørg for at både tKulepenn og wobjArk er aktivt.
Velg Program editor i hovedmeny. Marker <smt> og trykk Add instruction



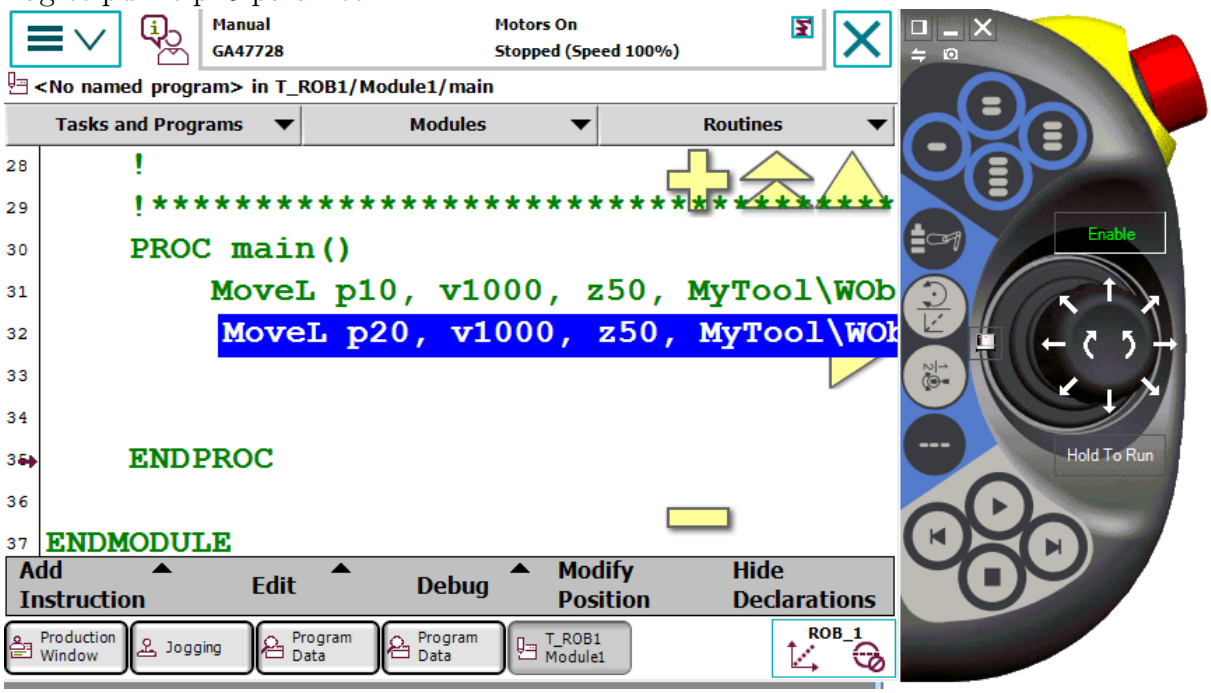
Velg instruksjonen MoveL



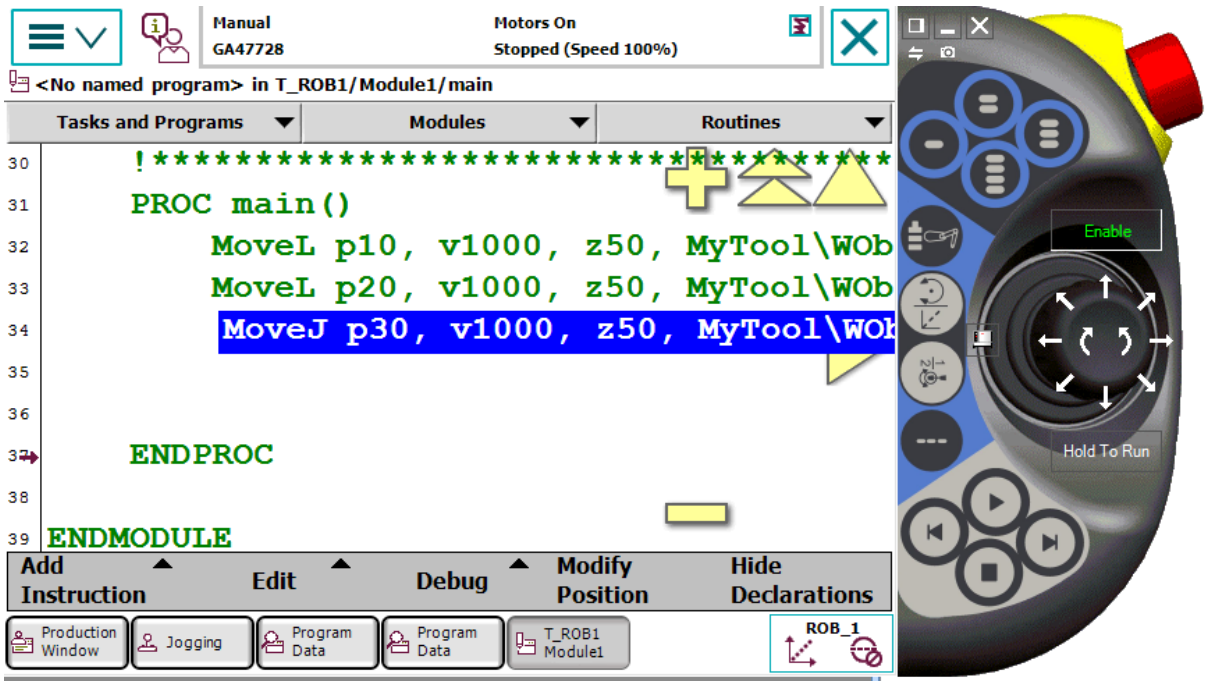
Dobbel klikk på stjernen for å endre navn. Velg new, roboten vil automatisk foreslå p10.



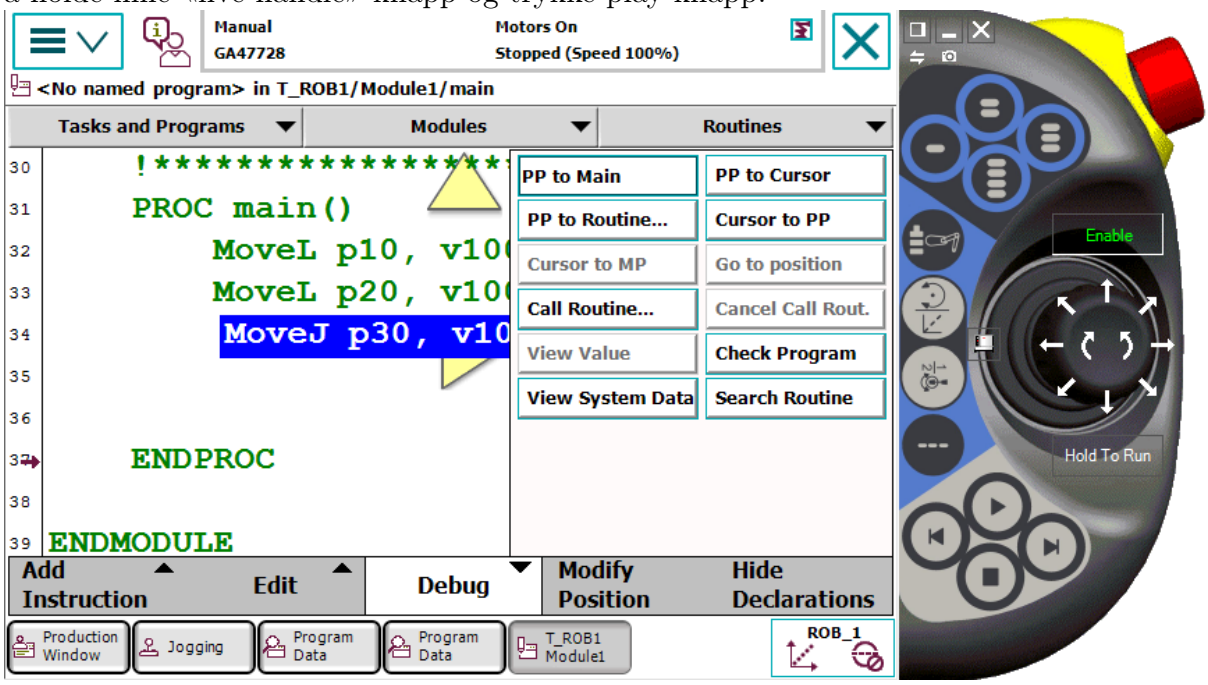
Trykk OK.
Jogg til et punkt 20cm fra p10.
Lag et punkt p20 på arket.



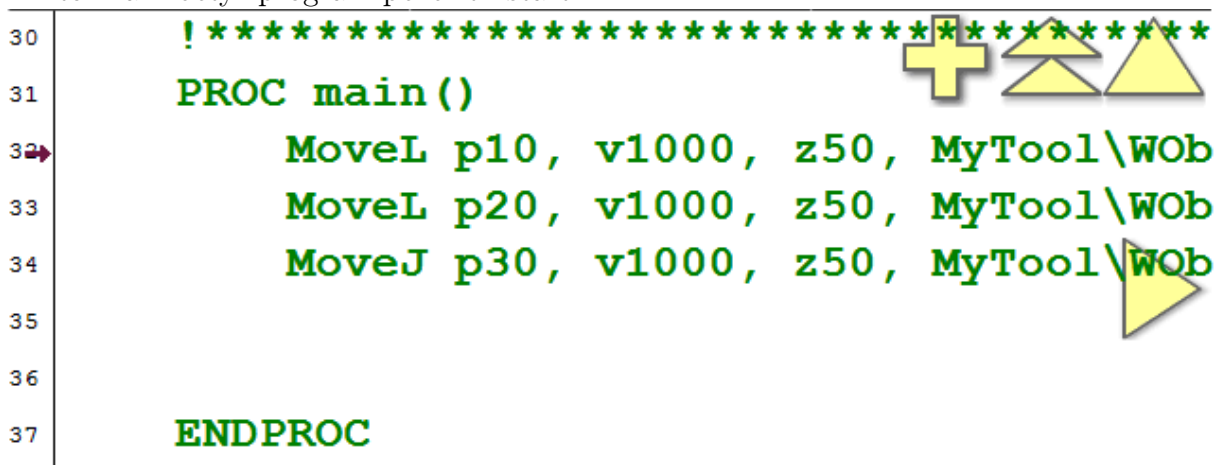
Kjør tilbake til punkt p10, men denne gang med instruksjonen MoveJ



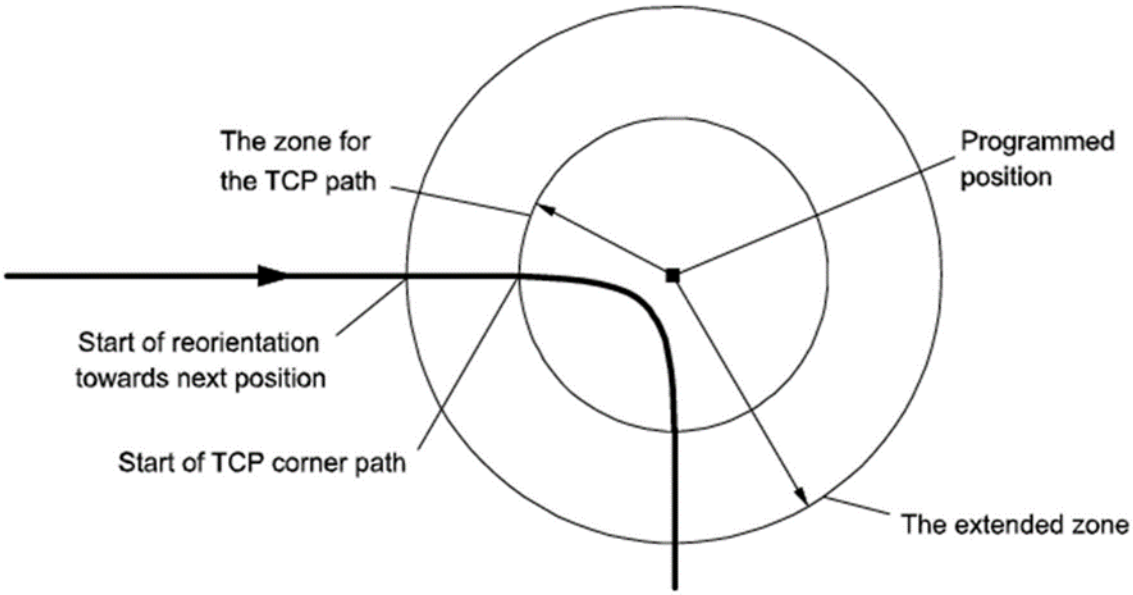
For å teste programmet trykk «debug», deretter «PP to Main». Nå kan robot startes ved å holde inne «live handle» knapp og trykke play knapp.



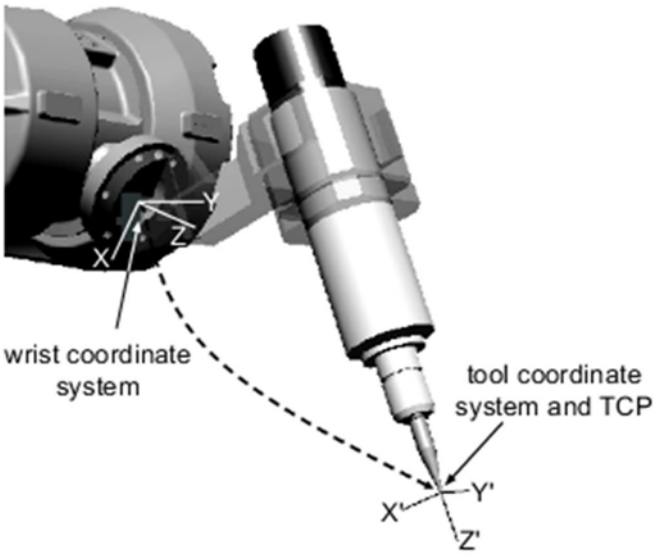
PP to Main betyr program peker til start



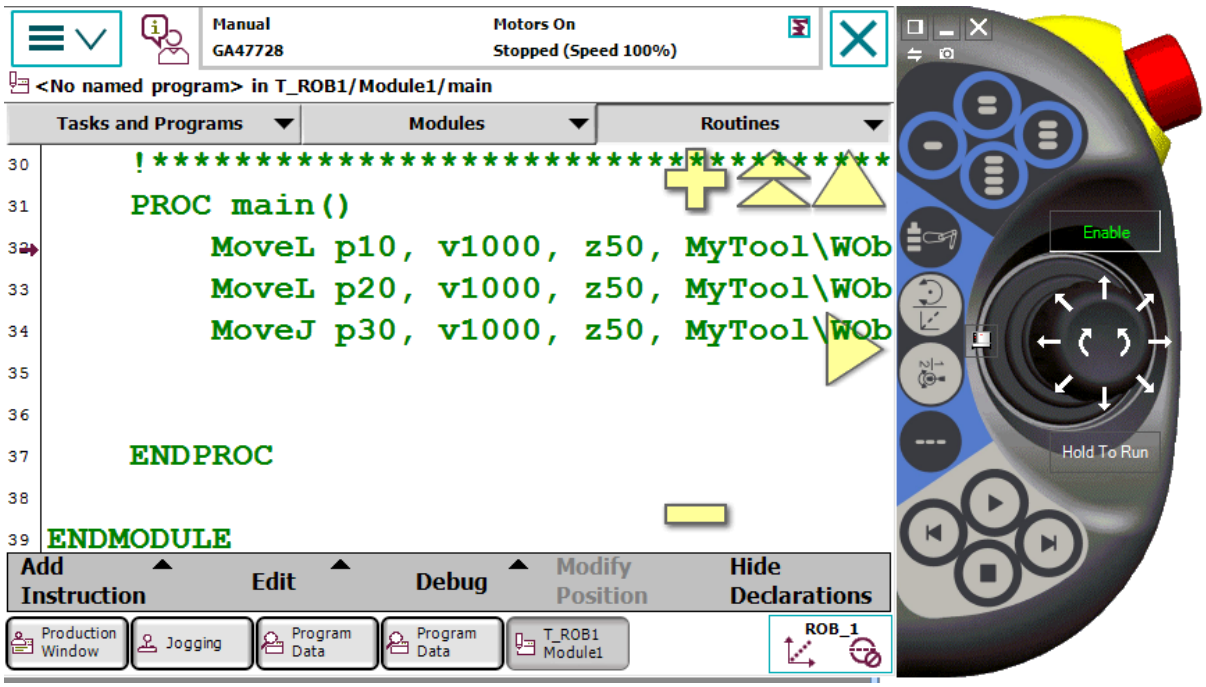
Pilen visualiserer hvor i programmet enn er.
Observer hva som skjer.
Hva består en MoveL og MoveJ instruksjon av?
Robtarget: består av X,Y,Z koordinat samt kvaternioner q1-q4 som oppgir orienteringen. Ett robtarget kan enten være ett «stjernepunkt» eller gis individuelt navn.
Speeddata: hastigheten roboten skal kjøre til punktet med. For eksempel V1000 som vil si 1000mm/s.
Zonedata: Avgjer når robot kan begynne på neste instruksjon og bane. Dersom du har en sone på z100, kan robot begynne å kjøre mot neste punkt 100mm før den har nådd punktet.



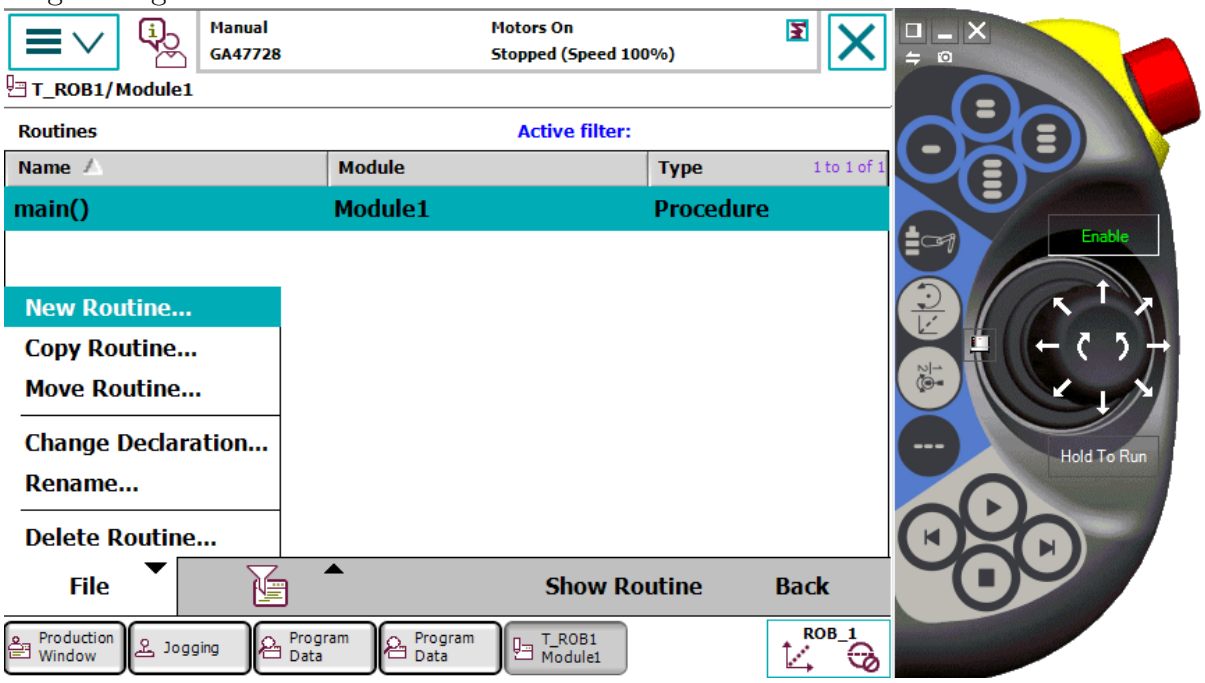
Tooldata: koordinat system ut fra flens på robot.



Work object: Egen definert koordinatsystem.
Figurer
Mål: Lære å lage rutiner, bruke instruksjonen MoveC og endre hastighet /soner i instruksjoner.
Først og fremst skal vi opprette 3 nye rutiner. Firkant, trekant og sirkel.
Trykk på Routines.



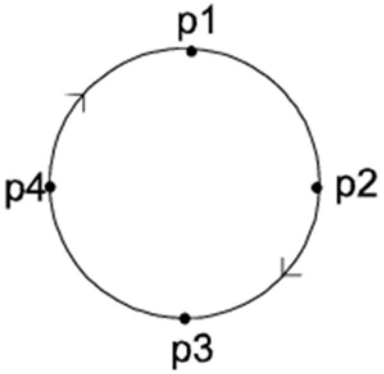
Velg File og new routine



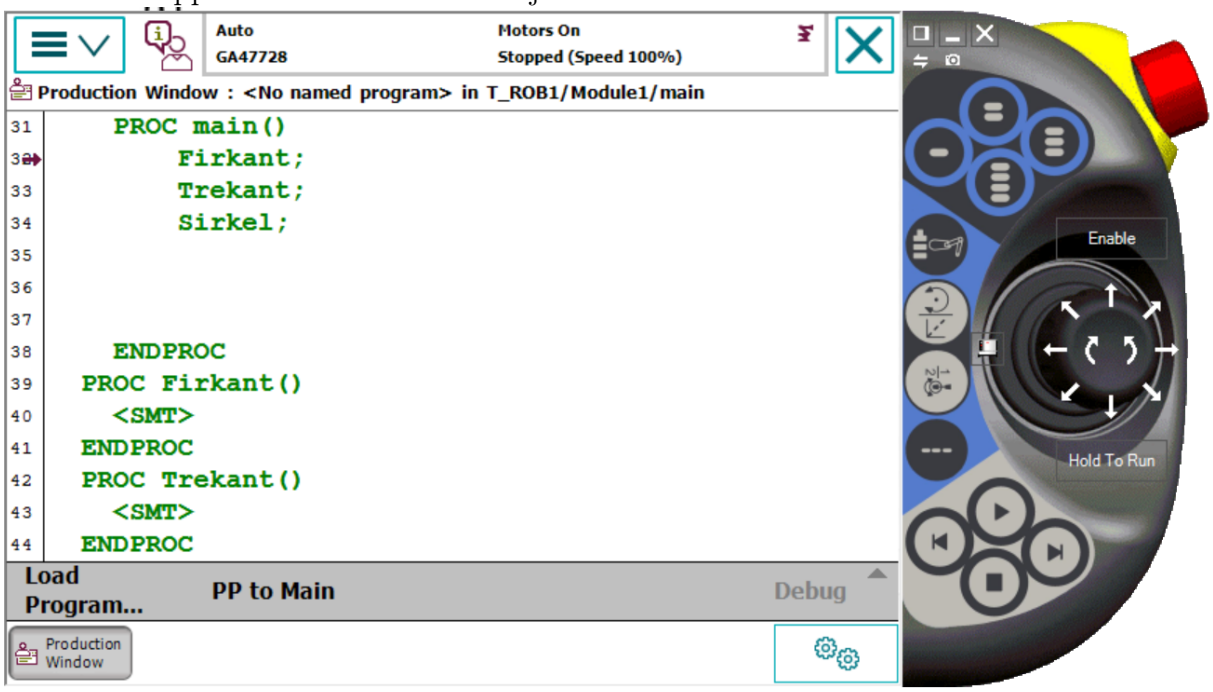
Opprett 3 rutiner, firkant, trekant og sirkel.
Lag program for å tegne firkant, trekant og sirkel. For å lage sirkel bruker vi instruksjonen MoveC. Se eksempel.

```
MoveL p1, v500, fine, tool1;
MoveC p2, p3, v500, z20, tool1;
MoveC p4, p1, v500, fine, tool1;
```

The figure shows how a complete circle is performed by two `MoveC` instructions.

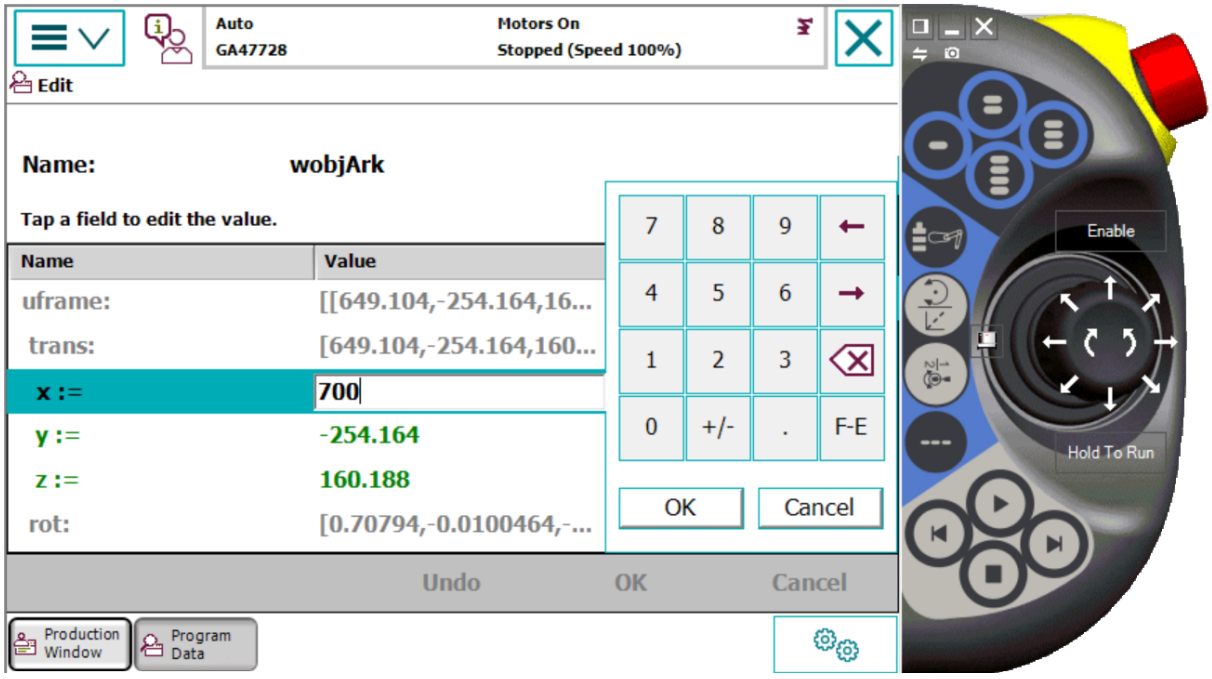


For å kjøre en enkel rutine velg debug, og PP to Routine.
Prøv å endre hastighet på de forskjellige instruksjonene. (under 250mm/s)
Prøv å endre zonedata.
Observer hva som skjer.
for å kjøre alle rutiner automatisk må de legges inn i Main rutinen. Vi sletter først alt gammelt i main.
For å kalle opp rutiner brukes instruksjonen ProcCall. Main rutinen skal se slik ut:

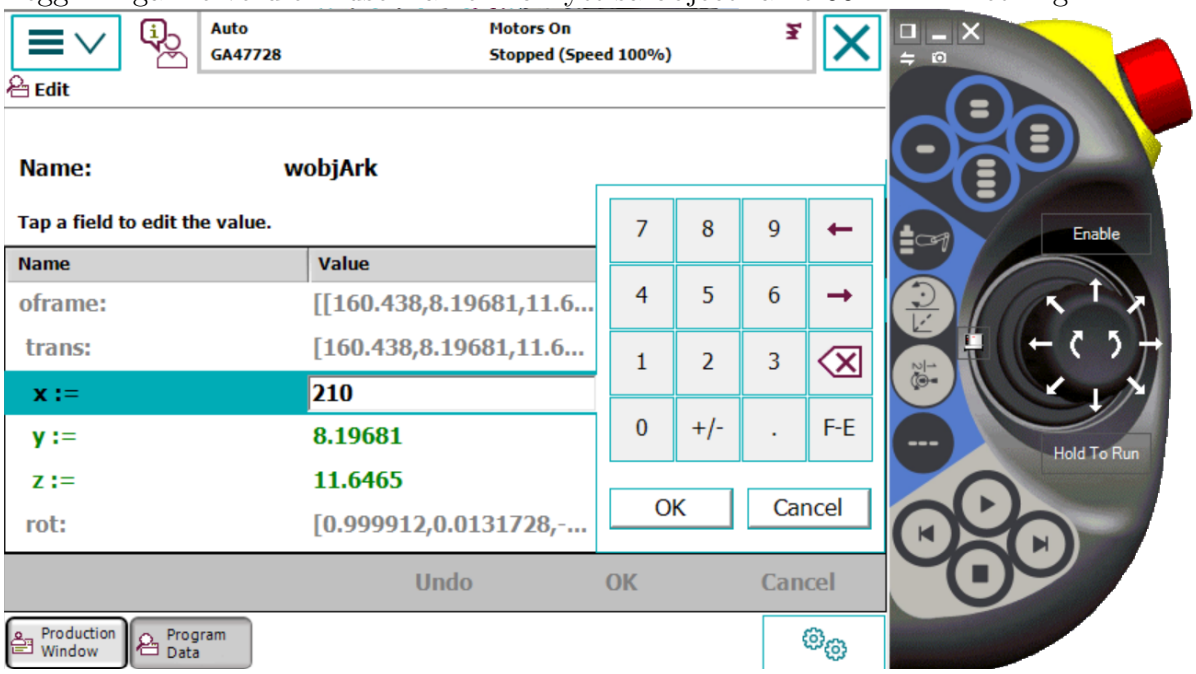


Flytte WorkObject

Mål: Lære forskjell på «userframe» og «objectframe» .
Bruk program fra oppg 5.
Gå til program data
Velg wobjdata
Dobbelklikk på wobjArk
Forflytt userframe 50 mm i X retning.



Kjør program og observer hva som skjer.
Legg inn gamle verdier i userframe. Forflytt så objectframe 50mm i X retning.



Kjør program og observer hva som skjer.

Arbeidsoppdrag 3 - Programmering I

IF instruksjonen

Mål: Forstå bruk av IF instruksjonen og lære å åpne/lukke griper.

1. Lag rutiner for henting og levering av Puck HentPuck og LeverPuck.

   Eks:

```
PROC HentPuck
        MoveJ p10,v1000,z50,tKlo\Wobj:=wobjArk;
        MoveL p20,v1000,fine, tKlo\Wobj:=wobjArk;
        Set CloseClaw;
        Reset OpenClaw;
        Waittime 0.2;
```

```
        MoveL p10,v1000,z50, tKlo \Wobj:=wobjArk;
ENDPROC
```

Instruksjonen Set vil aktivere ett utgangssignal. Reset deaktiverer utgangssignalet.

2. Lag ett program hvor du bruker bryterne til å bestemme hvilken figur som skal tegnes. Sett opp main rutinen ved hjelp av instruksjonen IF.

   diBr står for digital input Bryter. Ett digitalt inngangssignal er i bunn og grunn ett 24 V signal som er koblet til IO kortet til roboten. Dette signalet vil enten være 1 eller 0 ut ifra om bryteren er på eller av.

   Eks.

```
PROC main()
        IF diBr1=1 THEN
                Trekant;
        ELSEIF diBr2=1 THEN
                Sirkel;
        ELSEIF diBr3=1 THEN
                Firkant;
        ENDIF
ENDPROC
```

WHILE instruksjonen
Mål: Forstå bruk av WHILE instruksjon

1. Bruk WHILE til å velge mellom trekant, firkant og sirkel.

   Eks.

```
PROC main()
        WHILE diBr1=1 do
                Trekant;
        ENDWHILE
        WHILE diBr2=1 do
                Firkant;
        ENDWHILE
        WHILE diBr3=1 do
                Sirkel;
        ENDWHILE
ENDPROC
```

FOR instruksjonen
Mål: Forstå bruk av FOR instruksjon

1. Bruk FOR til å velge antall sykluser

   Eks.

```
PROC main()
        FOR i FROM 1 TO 5 DO
                Trekant;
        ENDFOR
        FOR i FROM 1 TO 3 DO
                Firkant;
```

```
        ENDFOR
        FOR i FROM 1 TO 2 DO
                Sirkel;
        ENDFOR
ENDPROC
```

i står for Identifier (iterator), og er en ren teller.

TEST instruksjonen
Mål: Forstå bruk av TEST instruksjon

1. Bruk TEST til å velge syklys.

   For å kunne bruke TEST instruksjonen må vi opprette ett register/num. Det er en type program data som husker tallet den har blitt gitt.

2. Gå til program data og velg num.

3. Trykk new og gi registeret navnet nFigur.

   For å endre verdi på nFigur, gå til program data num og velg nFigur. Du vil da få mulighet til å endre tallet.

   Dersom tallet er verken 1,2, eller 3 vil program peker hoppe til default.

   Eks.

```
PROC main()
TEST nFigur
        CASE 1:
                Trekant
        CASE 2:
                Firkant;
        CASE 3:
                Sirkel;
        DEFAULT:
                Stop;
        ENDTEST
ENDPROC
```

Operatørkommunikasjon
Mål: Lære å bruke TPReadFK og TPReadNum.
Du skal nå bruke det du har lært til å lage ett program som spør hvilken figur som skal tegnes, og hvor mange ganger den skal tegnes.

1. 1. Lag Init rutine der du nullstiller register og åpner griper. -

   Instruksjonen := kan brukes til å nullstille registeret, eller å gi det den verdien du ønsker.

```
PROC init()
        reg1 := 0;
        Reset CloseClaw;
        Set OpenClaw;
ENDPROC
```

2. Lag Meny rutine hvor instruksjonene TPReadFK skal brukes for å bestemme hvilken figur som skal tegnes, og TPReadNum for å bestemme antall ganger figuren skal tegnes.

   Beskrivelse av TPReadFK:

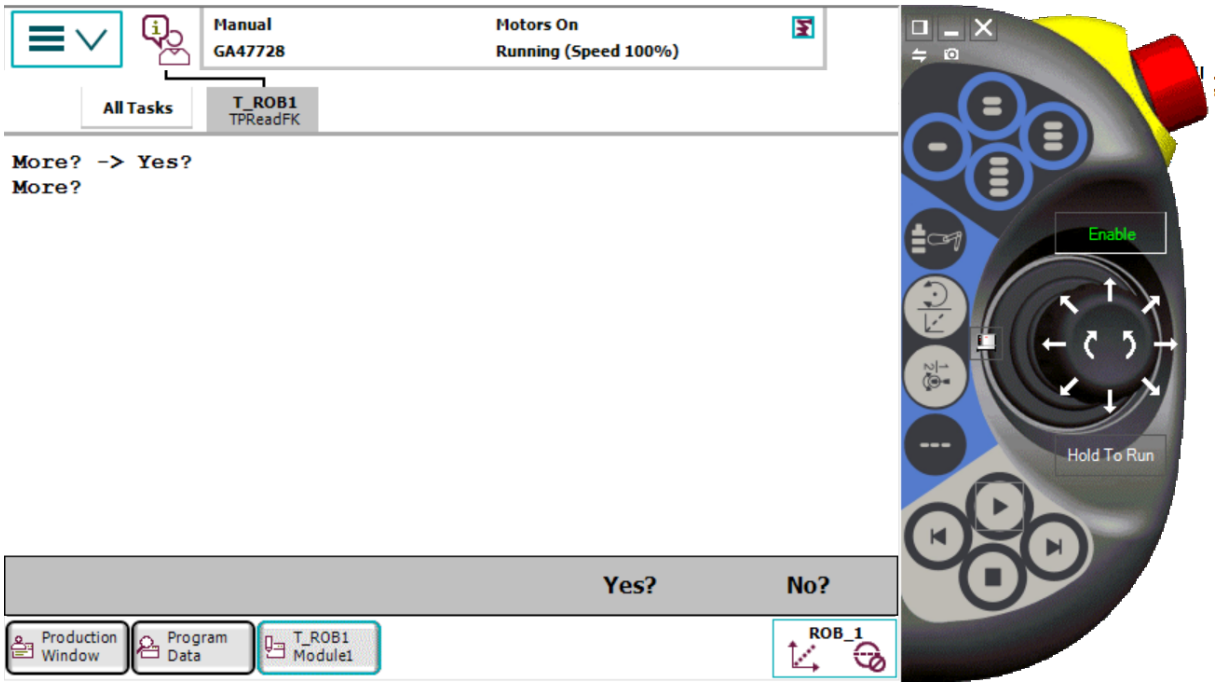
   TPReadFK (FlexPendant Read Function Key) er brukt til å skrive tekst på funksjons knappene, og til å finne ut hvilken knapp som trykkes på.

   Eksempel 1

   ```
   TPReadFK reg1, "More?", stEmpty, stEmpty, stEmpty, "Yes", "No";
   ```

   Teksten More? Blir skrevet på flexpendant skjermen, samt knappene 4 og 5 er aktivert med tekstene "Yes" og "No" (se bildet nedenfor). Programmet vil ikke gå videre før eventuelt knapp 4 eller 5 trykkes på. Med andre ord, reg1 vil få verdien 4 eller 5 avhengig av hvilken knapp som trykkes på.

   Bildet illustrerer hvordan instruksjonen vil se ut.



   Beskrivelse av TPReadNum: TPReadNum (FlexPendant Read Numerical) blir brukt til å lese ett nummer fra flexpendanten.

   ```
   TPReadNum reg1, "How many units should be produced?";
   ```

   Teksten "How many units should be produced?" vil bli skrevet på flexpendant skjermen. Programmet vil ikke fortsette før det har blitt lagt inn ett tall på det numeriske tastaturet. Tallet vil bli lagret i Reg1.

Arbeidsoppdrag 4 - Programmering II
Stabling av klosser.

1. Definer verktøy og workobjecter.
2. Lag rutiner for Init, Åpne- og Lukkegriper, Hent- og Leverpuck.
3. Lag Meny rutine til å velge fra hvilken stabel en skal hente.
4. Bruk navn på Robtarget f.eks pHentHvitKloss.

5. En skal hente fra 2 posisjoner og stable i høyden (1 stabel).

6. Bruk TPReadNum til å bestemme antall klosser levert i høyden.

Offs
Offs is used to add an offset in the object coordinate system to a robot position.
Example 1:

```
MoveL Offs(p2, 0, 0, 10), v1000, z50, tool1;
```

The robot is moved to a point 10 mm from the position p2 (in the z-direction).
Example 2

```
p1 := Offs (p1, 5, 10, 15);
```

The robot position p1 is displaced 5 mm in the x-direction, 10 mm in the y-direction and 15 mm in the z-direction.

1. Lever klosser i 4 stabler ved å bruke Offset i X og Y (et lag om gangen).

2. Endre henteposisjonene fra 1 i høyden til å plukke fra stabel, hentehøyden bestemmes ved å taste inn antall i meny rutinen.

3. Lag operatørkommunikasjon som fortell at det er tomt for klosser i hentepos og fullt i leveringsposisjoner.

file i04861

Oppgave 19

Feilsøing – Nivå 1

Arbidsoppdrag på Stasjon 5

Feilsøing I

Introdusjon

Teorioppgaver Start opp PLS programmet og sett deg inn i virkemåten.
PLS program til stasjonen

Skjema for anlegget finner du her.

Planlegging

Gjennomføring

Dokumentasjon

Beskriv hvordan du planlegger, gjennomfører og dokumenterer denne jobben.
file i04865

Oppgave 20

Feilsøing – Nivå 2

Arbidsoppdrag på Stasjon 5

Feilsøing II

Introdusjon

Teorioppgaver

Start opp PLS programmet og sett deg inn i virkemåten.
PLS program til stasjonen

Skjema for anlegget finner du her.

Planlegging

Gjennomføring

Dokumentasjon

Beskriv hvordan du planlegger, gjennomfører og dokumenterer denne jobben.
file i04866

Svar

Svar 1

Svar 2

Svar 3

Svar 4

Svar 5

Svar 6

Svar 7

Svar 8

Svar 9

Svar 10

Svar 11

Svar 12

Svar 13

Svar 14

Svar 15

Svar 16

Svar 17

Svar 18

Svar 19

Svar 20