



UNIVERSAL ROBOTS

Bruksanvisning för PolyScope



Version 3.5.0

Översättning av originalinstruktionerna (sv)

Den information som ingår häri tillhör Universal Robots A/S och får inte återges, i sin helhet eller delvis, utan i förväg inhämtat skriftligt tillstånd från Universal Robots A/S. Den information som ingår häri kan komma att ändras utan föregående meddelande och skall inte anses vara en utfästelse från Universal Robots A/S. Den här handboken omarbetas och revideras med jämna mellanrum.

Universal Robots A/S påtar sig inget ansvar för eventuella fel eller utelämnad information i det här dokumentet.

Copyright © 2009–2017 av Universal Robots A/S

Universal Robots-logotypen är ett registrerat varumärke som tillhör Universal Robots A/S.

Innehåll

II Bruksanvisning för PolyScope	II-1
10 Säkerhetskfiguration	II-3
10.1 Inledning	II-3
10.2 Ändra säkerhetskfigurationen	II-5
10.3 Säkerhetssynkronisering och fel	II-5
10.4 Toleranser	II-6
10.5 Säkerhets-checksum	II-6
10.6 Säkerhetslägen	II-7
10.7 Frikörningsläge	II-7
10.7.1 Backdrivning	II-8
10.8 Lösenordslås	II-8
10.9 Använd	II-8
10.10 Allmänna gränser	II-9
10.11 Ledbegränsning	II-11
10.12 Gränser	II-12
10.12.1 Välja en gräns att konfigurera	II-13
10.12.2 3D-visualisering	II-14
10.12.3 Konfiguration av säkerhetsplan	II-14
10.12.4 Konfiguration av verktygsgräns	II-17
10.13 Säkerhets-I/O	II-19
10.13.1 Ingångssignaler	II-19
10.13.2 Utgångssignaler	II-21
11 Börja programmeringen	II-23
11.1 Inledning	II-23
11.2 Komma igång	II-24
11.2.1 Installera robotarmen och manöverskåpet	II-24
11.2.2 Sätta på och stänga av manöverskåpet	II-24
11.2.3 Sätta på och stänga av robotarmen	II-24
11.2.4 Snabbstart	II-25
11.2.5 Det första programmet	II-25
11.3 Programmeringsgränssnittet i PolyScope	II-26
11.4 Välkomstkärmbilden	II-28
11.5 Initieringsskärm bild	II-29
12 Redigeringsprogram på skärmen	II-31
12.1 Redigeringsprogram för uttryck	II-31
12.2 Skärmbilden Poseredigering	II-31

13 Robotstyrning	II-35
13.1 Fliken Flytta	II-35
13.1.1 Robot	II-35
13.1.2 Funktioner och verktygsposition	II-36
13.1.3 Verktyget Flytta	II-36
13.1.4 Flytta leder	II-36
13.1.5 Frikörning	II-36
13.2 Fliken I/O	II-37
13.3 I/O för MODBUS-klient	II-38
13.4 Fliken AutoFlytta	II-39
13.5 Installation → Ladda/spara	II-40
13.6 Installation → TCP-konfiguration	II-41
13.6.1 Lägga till, ändra och ta bort TCP	II-41
13.6.2 Standard-TCP och aktiv TCP	II-42
13.6.3 Lära ut TCP-position	II-42
13.6.4 Lära ut TCP-orientering	II-43
13.6.5 Nyttolast	II-43
13.6.6 Masscentrum	II-43
13.7 Installation → Montering	II-44
13.8 Installation → I/O-inställning	II-45
13.8.1 I/O-signaltyp	II-46
13.8.2 Använda användaredefinierade namn	II-46
13.8.3 I/O-åtgärder och I/O-flikkontroll	II-46
13.9 Installation → Säkerhet	II-46
13.10 Installation → Variabler	II-47
13.11 Installation → I/O-inställningar för MODBUS-klient	II-48
13.12 Installation → Funktioner	II-51
13.12.1 Använda en funktion	II-52
13.12.2 Ny punkt	II-54
13.12.3 Ny linje	II-54
13.12.4 Nytt plan	II-55
13.12.5 Exempel: Manuell uppdatering av en funktion för att justera ett program	II-56
13.12.6 Exempel: Dynamisk uppdatering av en funktions pose	II-57
13.13 Inställning av transportbandsspårning	II-57
13.14 Installation → Standardprogram	II-59
13.15 Fliken Logg	II-61
13.16 Skärmbilden Ladda	II-61
13.17 Fliken Kör	II-64
14 Programmering	II-65
14.1 Nytt program	II-65
14.2 Fliken Program	II-66
14.2.1 Programträd	II-66
14.2.2 Markering av programkörning	II-67
14.2.3 Söknapp	II-67
14.2.4 Knapparna Ångra/Gör om	II-68
14.2.5 Programinstrumentbrädan	II-68

14.3	Variabler	II-69
14.4	Kommando: Tom	II-69
14.5	Kommando: Flytta	II-70
14.6	Kommando: Fast vägpunkt	II-73
14.7	Kommando: Relativ vägpunkt	II-79
14.8	Kommando: Variabel vägpunkt	II-80
14.9	Kommando: Vänta	II-81
14.10	Kommando: Ställ in	II-81
14.11	Kommando: Popup	II-82
14.12	Kommando: Stopp	II-83
14.13	Kommando: Kommentar	II-83
14.14	Kommando: Mapp	II-84
14.15	Kommando: Loop	II-85
14.16	Kommando: SubProgram	II-86
14.17	Kommando: Tilldelning	II-87
14.18	Kommando: Om	II-88
14.19	Kommando: Skript	II-89
14.20	Kommando: Händelse	II-90
14.21	Kommando: Tråd	II-91
14.22	Kommando: Switch	II-91
14.23	Kommando: Mönster	II-92
14.24	Kommando: Kraft	II-93
14.25	Kommando: Pall	II-97
14.26	Kommando: Sök	II-98
14.27	Kommando: Transportbandsspårning	II-101
14.28	Kommando: Undertryck	II-102
14.29	Grafik, flik	II-102
14.30	Struktur, flik	II-103
14.31	Variabler, flik	II-104
14.32	Kommando: Initiera variabler	II-105
15	Skärmbilden Inställning	II-107
15.1	Språk och enheter	II-108
15.2	Uppdatera robot	II-109
15.3	Ange lösenord	II-110
15.4	Kalibrera skärm	II-111
15.5	Ställ in nätverk	II-111
15.6	Ange tid	II-112
15.7	Installationsprogram för URCaps	II-113

Del II

Bruksanvisning för PolyScope

10 Säkerhetskfiguration

10.1 Inledning

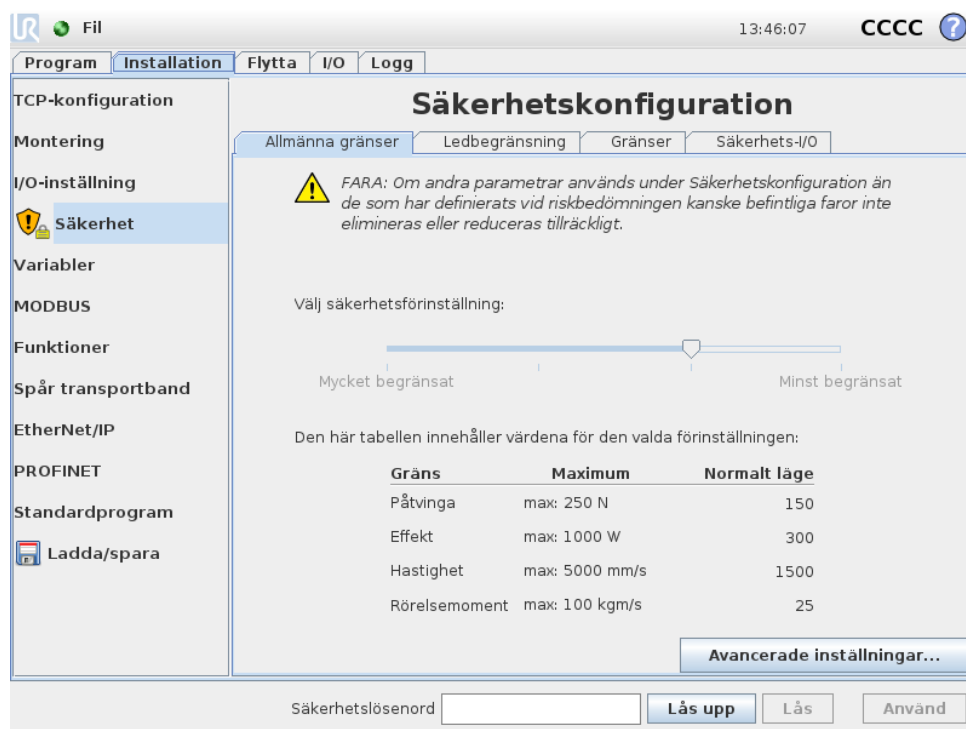
Roboten är utrustad med ett avancerat säkerhetssystem. Beroende på de exakta omständigheterna i robotens arbetsområde måste inställningarna för säkerhetssystemet konfigurerats så att de garanterar säkerheten för all personal och utrustning runt roboten. Det första som en integratör måste göra är att applicera de inställningar som har definierats genom riskbedömningen. Mer information om säkerhetssystemet finns i Bruksanvisning för installation av maskinvara.



FARA:

1. Användning och konfiguration av säkerhetsrelaterade funktioner och gränssnitt måste uppfylla villkoren för den riskbedömning som integratören utför för en specifik robottillämpning, se Bruksanvisning för installation av maskinvara.
2. Säkerhetskfigurationens inställningar för uppsättning och inläring måste appliceras i enlighet med den riskbedömning som integratören har utfört, och innan robotarmen startas för första gången.
3. Alla säkerhets- och konfigurationsinställningar som går att komma åt på den här skärmbilden och dess underflikar måste ställas in enligt integratörens riskutvärdering.
4. Integratören måste kontrollera att alla ändringar i säkerhetsinställningarna utförs i enlighet med integratörens egen riskutvärdering.
5. Integratören måste förhindra obehöriga personer från att ändra säkerhetskfigurationen, t.ex. genom att använda lösenordsskydd.

Du kan komma åt skärmen **Säkerhetskfiguration** från skärmen **Välkommen** (se 11.4) genom att trycka på knappen **Programmera robot**, välja fliken **Installation** och trycka på **Säkerhet**. Säkerhetskfigurationen är lösenordsskyddad. Se 10.8.



Säkerhetsinställningarna består av ett antal gränsvärden som används för att begränsa robotarmens rörelser, och säkerhetsinställningar för konfigurerbara in- och utgångar. De definieras i följande underflikar på säkerhetsskärmen:

- På underfliken *Allmänna gränser* definieras de maximala värdena för robotarmens *kraft*, *effekt*, *hastighet* och *rörelsemoment*. När det råder speciellt hög risk att för att en människa träffas eller robotarmen krockar med något i omgivningen, måste dessa inställningar ges låga värden. Om risken är låg gör högre allmänna inställningar att roboten kan röra sig snabbare och applicera mer kraft på omgivningen. Mer information finns i 10.10.
- Underfliken *Ledbegränsning* innehåller gränser för *ledhastighet* och *ledposition*. Gränserna för *ledhastighet* definierar den maximala vinkelhastigheten för enskilda leder, och innebär ytterligare begränsning av robotarmens hastighet. Gränserna för *ledposition* definierar det tillåtna positionsintervallet för enskilda leder (i ledrymden). Mer information finns i 10.11.
- På underfliken *Gränser* definieras säkerhetsplan (i kartesisk rymd) och ett gränsvärde för verktygsorienteringen för robotens TCP. Säkerhetsplanen kan konfigureras antingen som fasta gränser för positionen för robotens TCP eller som utlösare för att aktivera säkerhetsgränserna för läget *Reducerat* (se 10.6). Verktygsorienteringens gräns sätter en fast gräns för orienteringen för robotens TCP. Mer information finns i 10.12.
- På underfliken *Säkerhets-I/O* definieras säkerhetsfunktioner för konfigurerbara ini- och utgångar (se 13.2). Det går t.ex. att konfigurera *Nödstopp* som en ingång. Mer information finns i 10.13.

10.2 Ändra säkerhetskfigurationen

Säkerhets- och konfigurationsinställningar får bara ändras i enlighet med den riskbedömning som har utförts av integratören.

Den rekommenderade proceduren för att ändra säkerhetskfigurationen är följande:

1. Kontrollera att alla ändringar följer den riskbedömning som har utförts av integratören.
2. Justera säkerhetsinställningarna till rätt nivå enligt den riskbedömning som har utförts av integratören.
3. Verifiera att säkerhetsinställningarna har verkställts.
4. Lägg in följande text i driftshandböckerna: "Se till att säkerhetskfigurationen är enligt förväntan innan du arbetar i närheten av roboten. Detta kan kontrolleras t.ex. genom att granska checksum längst upp till höger i PolyScope (se 10.5 i Bruksanvisning för PolyScope)."

10.3 Säkerhetssynkronisering och fel

Tillståndet för säkerhetskfigurationen som används jämfört med den robotinstallation som det grafiska användargränssnittet har laddat visas med en sköldikon bredvid texten *Säkerhet* till vänster på skärmen. Dessa ikoner ger en snabb överblick över det aktuella tillståndet. De definieras nedan:



Konfiguration synkroniserad: Visar att det grafiska användargränssnittets installation är identisk med den säkerhetskfiguration som för närvarande används. Inga ändringar har gjorts.



Konfiguration ändrad: Visar att det grafiska användargränssnittets installation avviker från den säkerhetskfiguration som för närvarande används.

När säkerhetskfigurationen redigeras visar sköldikonen om de nya inställningarna har använts.

Om något av textfälten på fliken *Säkerhet* innehåller ogiltiga uppgifter är säkerhetskfigurationen i ett feltillstånd. Detta anges på flera sätt:

1. En röd felikon visas bredvid texten *Säkerhet* till vänster på skärmen.
2. De delflikar som innehåller fel markeras med en röd felikon överst.
3. Textfält som innehåller fel markeras med röd bakgrund.

Om det finns fel och du försöker lämna fliken *Installation* visas en dialogruta med följande alternativ:

1. Lös felen så att alla fel har eliminerats. Detta blir synligt när den röda felikonen inte längre visas bredvid texten *Säkerhet* till vänster på skärmen.
2. Återgå till säkerhetskfigurationen som användes tidigare. Detta ignorerar alla ändringar och låter dig fortsätta med någon annan uppgift.

Om det inte finns några fel och du försöker lämna fliken *Installation* visas en annan dialogruta med följande alternativ:

1. Använd ändringarna och starta om systemet. Ändringarna i säkerhetskfigurationen verkställs för systemet varefter systemet startas om. Obs: Detta innebär inte att några ändringar har sparats. Om roboten stängs ned nu går alla ändringar i robotinstallationen förlorade, inklusive ändringarna i säkerhetskfigurationen.
2. Återgå till säkerhetskfigurationen som användes tidigare. Detta ignorerar alla ändringar och låter dig fortsätta med någon annan uppgift.

10.4 Toleranser

Robotarmen använder inbyggda toleranser som förhindrar säkerhetsöverträdelser. En säkerhetstolerans är skillnaden mellan en säkerhetsgräns och ett maximalt driftvärde. Den allmänna hastighetstoleransen är t.ex. -150mm/s . Det här betyder att om användaren ställer in en hastighetsgräns på 250mm/s blir max. drifhastighet $250 - 150 = 100\text{mm/s}$. Säkerhetstoleranser förhindrar säkerhetsöverträdelser samtidigt som vissa variationer i programmets beteende tillåts. Vid hantering av en tung last kan det t.ex. finnas situationer där *robotarmen* kortvarigt måste arbeta över normal maximal drifhastighet för att kunna följa en programmerad bana. Ett exempel på en sådan situation visas i figuren 10.1.



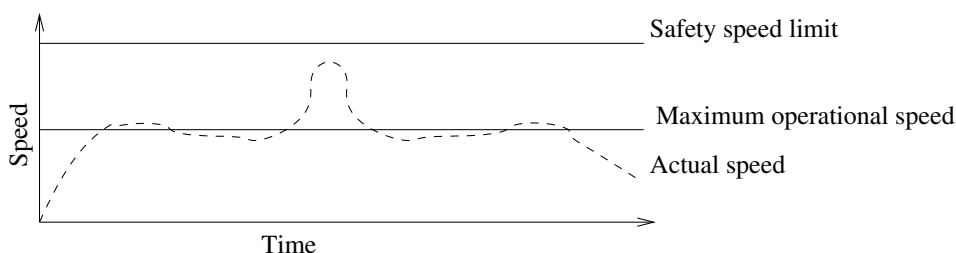
WARNING:

En riskutvärdering krävs alltid, och ska använda gränsvärdena utan toleranser.



WARNING:

Toleranser är specifika för en viss programvaruversion. Om programvaran uppdateras kan toleranserna ändras. I filen med viktig versionsinformation finns information om ändringar mellan versioner.



Figur 10.1: Exempel på säkerhetstolerans.

10.5 Säkerhets-checksum

Texten i skärmens övre högra hörn visar en kortfattad representation av den säkerhetskfiguration som roboten för närvarande använder. När texten ändras anger detta att även den aktuella säkerhetskfigurationen har ändrats. Om du klickar på checksum-värdet visas detaljerad information om den aktuella säkerhetskfigurationen.

10.6 Säkerhetslägen

Under normala omständigheter (d.v.s. när inget skyddsstopp är i kraft) är säkerhetssystemet i något av följande *säkerhetslägen*, vart och ett med en tillhörande uppsättning säkerhetsgränser:

Normalt läge: Det säkerhetsläge som är aktivt som standard.

Reducerat läge: Aktivt när robotens TCP finns bortom ett plan som angetts som *Utlös Reducerat läge* (se 10.12) eller när det utlösts med en konfigurerbar ingång (se 10.13).

Återställningsläge: När robotarmen brutit mot något av de andra lägena (dvs. lägena *Normal* eller *Reducerat*) och ett kategori 0-stopp har inträffat,¹ startas robotarmen i läget **Återställning**. Detta läge tillåter roboten att röra sig långsamt tillbaka till tillåtet område med knappen **Rörelse** eller **Frikörning**. Det går inte att köra program för roboten i det här läget.



WARNING:

Observera att gränser för *ledposition*, *TCP-position* och *TCP-orientering* är inaktiverade i läget *Återställning*, så var försiktig när robotarmen flyttas tillbaka innanför gränserna.

Med underflikarna på skärmen *Säkerhetskfiguration* kan användarna definiera separata uppsättningar med säkerhetsgränser för lägena *Normal* och *Reducerat*. För verktyget och lederna måste gränsvärdena för läget *Reducerat* avseende hastighet och rörelsemoment vara mer restriktiva än motsvarigheterna i läget *Normal*.

När en säkerhetsgräns överskrids i den aktiva gränsuppsättningen utför robotarmen ett kategori 0-stopp. Om en aktiv säkerhetsgräns, t.ex. en begränsning för *ledposition* eller en säkerhetsgräns, överskrids redan när robotarmen startas, startar den i läget *Återställning*. På så sätt kan robotarmen flyttas tillbaka innanför säkerhetsgränserna. I läget *Återställning* är robotarmens rörelse begränsad av en fast gränsuppsättning som användaren inte kan anpassa. Mer information om gränserna för läget *Återställning* finns i Bruksanvisning för installation av maskinvara.

10.7 Frikörningsläge

Om roboten är i läget *Frikörning* (se 13.1.5) och robotarmens rörelse närmar sig vissa gränser, kommer användaren att känna en bortstötande kraft. Kraften genereras för begränsningar för position, orientering och hastighet för robotens TCP samt ledernas position och hastighet.

Syftet med den bortstötande kraften är att informera användaren om att aktuell position eller hastighet är nära ett gränsvärde, och att förhindra att roboten överträder gränsvärdet. Det går dock att överträda gränsen, om användaren applicerar tillräckligt med kraft på robotarmen. Kraften blir starkare ju närmare gränsvärdet robotarmen kommer.

¹Enligt IEC 60204-1, se ordlista för mer information.

10.7.1 Backdrivning

I *Frikörningsläget* kan robotlederna flyttas med relativt låg kraft eftersom bromsarna är lossade. Under robotarmens initiering kan mindre vibrationer observeras när robotbromsarna lossas. I vissa situationer, till exempel när roboten är nära kollision, är sådana vibrationer oönskade och *Backdrivning* kan då användas för att flytta specifika leder till önskade lägen utan att lossa alla bromsar i robotarmen.

Aktivering av *Backdrivning*:

1. Tryck in PÅ-knappen för att aktivera drivningen till lederna. Robotens tillstånd ställs in på "Viloläge". Lossa **inte** bromsarna (dvs. tryck inte på START).
2. Tryck och håll in knappen *Frikörning*. Robotens läge ändras tillbaka till "Backdrivning".
3. Bromsarna lossas bara i de leder där betydande tryck har applicerats, så länge som knappen *Frikörning* är aktiverad / intryckt. Roboten känns tung att manövrera runt vid användning av *Backdrivning*.

10.8 Lösenordslås

Alla inställningar på skärmen är låsta tills rätt säkerhetslösenord (se 15.3) anges i det vita textfältet längst ned på skärmen och knappen Lås upp har tryckts in. Du låser skärmen igen genom att klicka på knappen Lås. Fliken Säkerhet låses automatiskt när du lämnar skärmen Säkerhetskfiguration. När inställningarna är låsta visas detta med låsikonen bredvid texten Säkerhet till vänster på skärmen. En upplåst ikon visas när inställningarna är upplåsta.



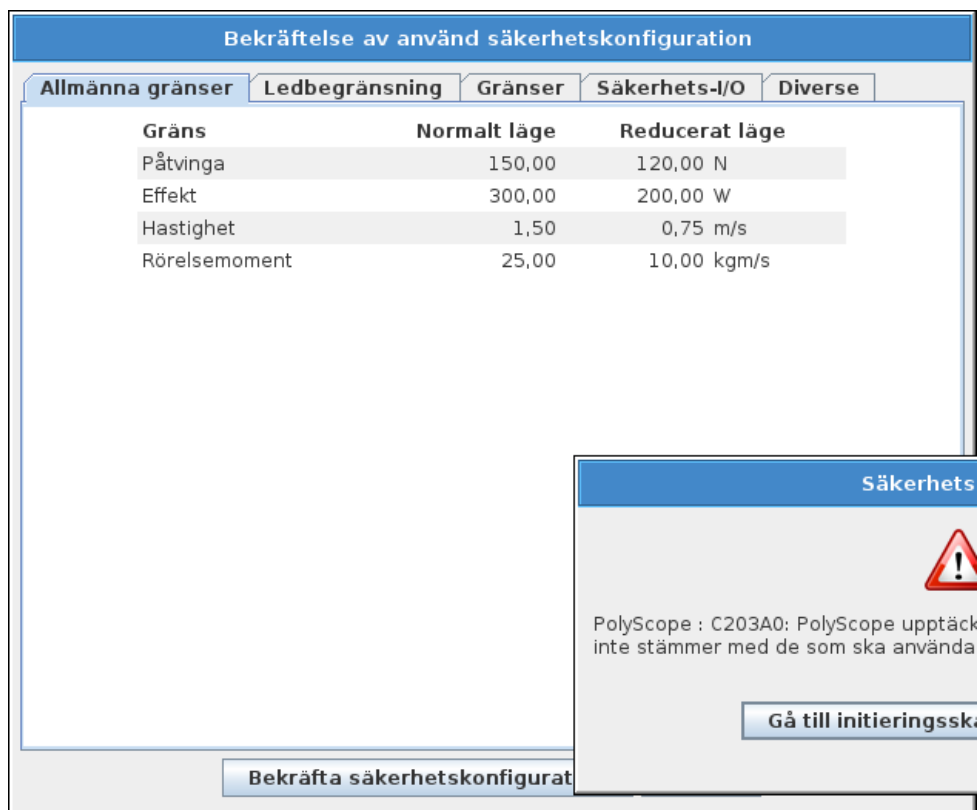
OBS:

Observera att robotarmen stängs av när skärmbilden Säkerhetskfiguration låses upp.

10.9 Använd

När säkerhetskfigurationen låses upp stängs robotarmen av medan ändringarna görs. Robotarmen kan inte sättas på förrän ändringarna har använts eller ångrats och en manuell uppstart har utförts från startskärmen.

Alla ändringar i säkerhetskfigurationen måste användas eller ångras innan du lämnar fliken Installation. Ändringarna träder *inte* i kraft förrän efter att knappen Använd tryckts in och bekräftelse getts. Bekräftelse kräver visuell granskning av de ändringar som getts till robotarmen. Av säkerhetsskäl visas informationen i SI-enheter. Ett exempel på dialogrutan för bekräftelse visas nedan.



Dessutom sparas ändringarna automatiskt som en del av den aktuella robotinstallationen vid bekräftelse. Mer information om att spara robotinstallationen finns i 13.5.

10.10 Allmänna gränser

De allmänna säkerhetsgränserna begränsar den linjära hastigheten för robotens TCP och den kraft den kan utöva på omgivningen. De består av följande värden:

Påtvunga: En gräns för den största kraft robotens TCP kan utöva på omgivningen.

Effekt: En gräns för det största mekaniska arbete som roboten kan utföra på omgivningen, med hänsyn till att nyttolasten är en del av roboten och inte en del av omgivningen.

Hastighet: En gräns för den högsta linjära hastigheten för robotens TCP.

Rörelsemoment: En gräns för det högsta rörelsemängdsmomentet för robotarmen.

Det finns två sätt att konfigurera de allmänna säkerhetsgränserna inom installationen: *Grundläggande inställningar* och *Avancerade inställningar*. De beskrivs närmare nedan.

Definition av de allmänna säkerhetsgränserna definierar endast en begränsning för verktyget, och inte den övergripande gränsen för robotarmen. Detta betyder att även om en hastighetsgräns anges, garanterar detta *inte* att andra delar av robotarmen lyder under samma begränsning.

Om läget är *Frikörning* (se 13.1.5) och den aktuella hastigheten för robotens TCP är nära gränsen för *Hastighet*, kommer användaren att känna en bortstötande kraft

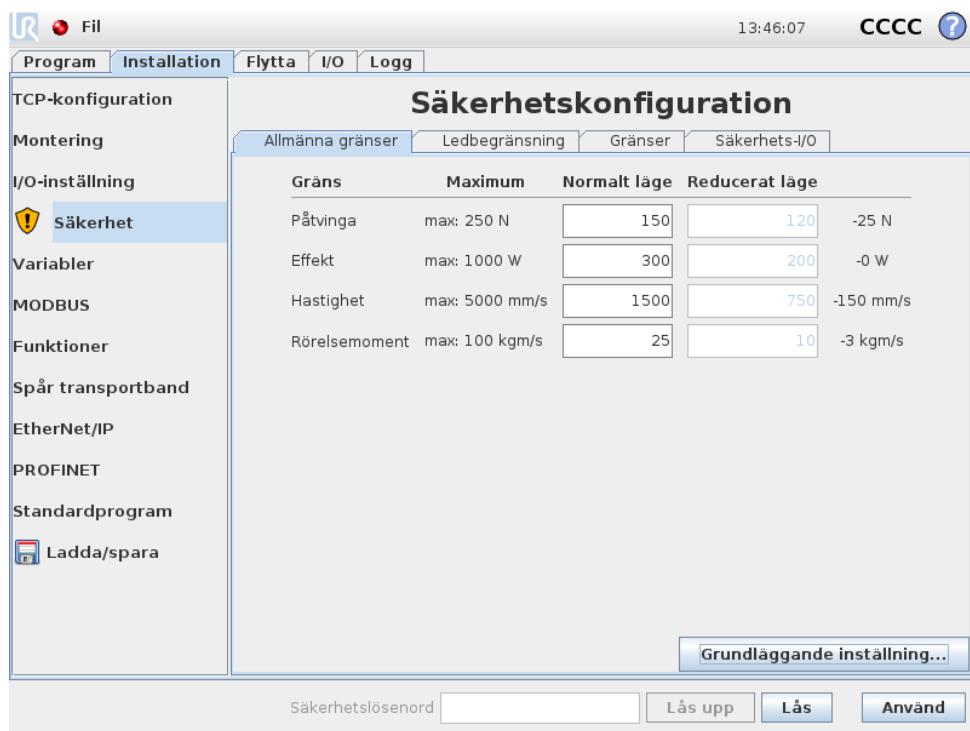
som ökar i styrka när hastigheten närmar sig gränsen. Kraften uppstår när den aktuella hastigheten är inom c:a 250 mm/s från gränsen.

Grundläggande inställningar Underpanelen för ursprungliga allmänna gränsvärden visas som standardskärm. Här finns ett skjutreglage med fyra fördefinierade uppsättningar värden för kraft, effekt, hastighet och momentgränsvärden i både *Normalt* och *Reducerat* läge.

De specifika värdeuppsättningarna visas i det grafiska användargränssnittet. Fördefinierade värden är endast förslag och de utgör inte någon riskbedömning.

Växla till Avancerade inställningar Om *ingen* av de fördefinierade uppsättningarna värden är lämpliga kan du trycka på knappen *Avancerade inställningar...* så att skärmen för avancerade allmänna inställningar öppnas.

Avancerade inställningar



Gräns	Maximum	Normalt läge	Reducerat läge	
Påtvinga	max: 250 N	150	120	-25 N
Effekt	max: 1000 W	300	200	-0 W
Hastighet	max: 5000 mm/s	1500	750	-150 mm/s
Rörelsemoment	max: 100 kgm/s	25	10	-3 kgm/s

Här kan var och en av de allmänna begränsningar som beskrivs i 10.10 ändras oberoende av de andra. Gör detta genom att trycka på motsvarande textfält och skriva ett nytt värde. Det högsta värde som accepteras för var och en av gränserna visas i spalten *Maximum*. Kraftgränsen kan anges till ett värde mellan 100 N och 250 N, och kraftgränsen kan anges till ett värde mellan 80 W och 1000 W,

Obs: Fälten för gränserna i läget *Reducerat* är inaktiverade om varken ett säkerhetsplan eller en konfigurerbar ingång har lagts upp för att utlösa det (se 10.12 och 10.13 om du vill ha mer information). Dessutom får värdena för *Hastighet* och *Rörelsemoment* i läget *Reducerat* inte vara högre än motsvarande värden för läget *Normal*.

Tolerans och enhet för varje gräns visas i slutet av motsvarande rad. När ett program körs justeras robotarmens hastighet automatiskt så att den inte överskrider

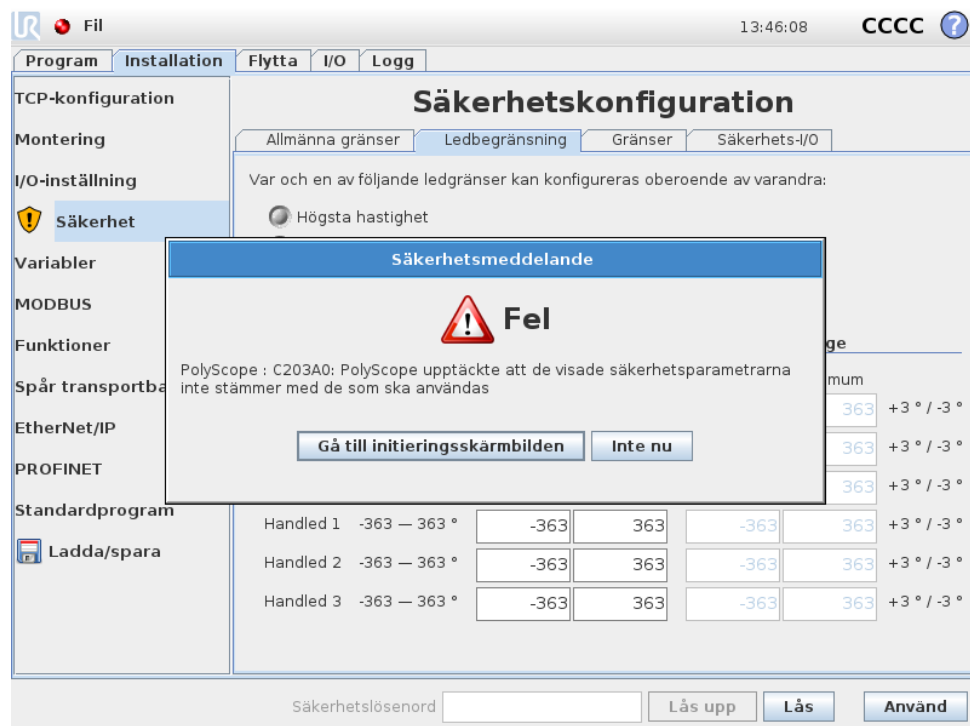
något av de värden som angetts, minus toleransen (se 10.4). Observera att det minustecken som visas med toleransvärdet bara anger att toleransen dras ifrån det verkliga värde som anges. Säkerhetssystemet utför ett Kategori 0-stopp om robotarmen överskrider den angivna gränsen (utan tolerans).

**WARNING:**

Hastighetsgränsen begränsar endast robotens TCP, så övriga delar av robotarmen kan röra sig snabbare än det värde som angetts.

Växla till Grundläggande inställningar Om du trycker på knappen Grundläggande inställningar... återgår du till skärmen för grundläggande allmänna begränsningar, och alla allmänna begränsningar återställs till sina förinställda *standardvärden*. Om detta gör att några anpassade värden går förlorade, visas en meddelanderuta där du får bekräfta åtgärden.

10.11 Ledbegränsning



Ledbegränsningar begränsar rörelsen för enskilda leder i ledrymden, dvs. de refererar inte till kartesisk rymd utan istället till den interna positionen (rotationspositionen) för lederna och deras rotationshastighet. Med växlingsknapparna i underpanelens övre del kan du ställa in *Högsta hastighet* och *Positionsintervall* för lederna oberoende av varandra.

Om läget är *Frikörning* (se 13.1.5) och den aktuella positionen eller hastigheten för en led är nära gränsen, kommer användaren att känna en bortstötande kraft som ökar i styrka när leden närmar sig gränsen. Kraften uppstår när ledhastigheten är

inom c:a 20°/s av hastighetsgränsen eller ledens position är inom c:a 8° av positionsgränsen.

Högsta hastighet Detta alternativ definierar den högsta vinkelhastigheten för varje led. Gör detta genom att trycka på motsvarande textfält och skriva ett nytt värde. Det högsta värde som accepteras visas i spalten *Maximum*. Inget av värdena kan anges till mindre än toleransvärdet.

Observera att fälten för gränserna i läget *Reducerat* är inaktiverade om varken ett säkerhetsplan eller en konfigurerbar ingång har lagts upp för att utlösa det (se 10.12 och 10.13 om du vill ha mer information). Dessutom får gränserna i läget *Reducerat* inte vara högre än motsvarande värden för läget *Normal*.

Tolerans och enhet för varje gräns visas i slutet av motsvarande rad. När ett program körs justeras robotarmens hastighet automatiskt så att den inte överskrider något av de värden som angetts, minus toleransen (se 10.4). Observera att det minustecken som visas med varje toleransvärde bara anger att toleransen dras ifrån det verkliga värde som anges. Säkerhetssystemet utför dock ett kategori 0-stopp om vinkelhastigheten för någon led överskrider det värde som angetts (utan tolerans).

Positionsintervall På denna skärm definieras positionsintervallet för varje led. Gör detta genom att trycka på motsvarande textfält och skriva nya värden för den övre och undre positionsgränsen för ledpositionen. Det intervall som anges måste falla inom de värden som anges i spalten *Intervall*, och den undre gränsen kan inte ligga högre än den övre gränsen.

Obs: Fälten för gränserna i läget *Reducerat* är inaktiverade om varken ett säkerhetsplan eller en konfigurerbar ingång har lagts upp för att utlösa det (se 10.12 och 10.13 om du vill ha mer information).

Toleranser och enhet för varje gräns visas i slutet av motsvarande rad. Det första toleransvärdet gäller för minimivärdet och det andra för maximivärdet. Programkörningen stoppas när positionen för en led håller på att överstiga det intervall som man får när den första toleransen läggs till det angivna minimivärdet och den andra toleransen subtraheras från det angivna maximivärdet, om den fortsätter röra sig längs den förutsedda banan. Observera att det minustecken som visas med toleransvärdet bara anger att toleransen dras ifrån det verkliga värde som anges. Säkerhetssystemet utför dock ett kategori 0-stopp om ledpositionen överskrider det intervall som angetts (utan tolerans).

10.12 Gränser

På den här fliken kan du konfigurera gränser, som består av säkerhetsplan och en begränsning av den högsta tillåtna avvikelser för robotverktygets orientering. Det går även att definiera plan som utlöser en övergång till *Reducerat* läge.

Säkerhetsplan kan användas för att begränsa robotens tillåtna arbetsområde, genom att framtvunga att robotens TCP håller sig på rätt sida om de definierade planen och inte korsar dem. Det går att konfigurera upp till åtta säkerhetsplan.

Begränsningen på verktygets orientering kan användas för att kontrollera att robotverktygets orientering inte avviker mer än vad som specifikt angetts från den orientering som önskas.

**WARNING:**

Definition av säkerhetsplan begränsar bara TCP, inte den övergripande gränsen för robotarmen. Detta betyder att även om ett säkerhetsplan anges, garanterar detta *inte* att andra delar av robotarmen lyder under samma begränsning.

Konfigurationen av varje gränsvärde baseras på någon av de egenskaper som definierats i den aktuella robotinstallationen (se 13.12).

**OBS:**

Vi rekommenderar starkt att du skapar alla de funktioner som behövs för konfigurerings av de önskade gränsvärdena och ger dem lämpliga namn innan du börjar redigera säkerhetskongfigurationen. Observera att eftersom robotarmen är strömlös när fliken *Säkerhet* har låsts upp är vare sig funktionen *Verktg* (som innehåller nuvarande position och orientering för robotens TCP) eller läget *Frikörning* (se 13.1.5) tillgängliga.

Om läget är *Frikörning* (se 13.1.5) och den aktuella positionen för robotens TCP är nära ett säkerhetsplan, eller om robotverktygets orientering har en avvikelse från den önskade orienteringen som ligger nära den angivna högsta avvikelsen, kommer användaren att känna en bortstötande kraft som ökar i styrka när TCP närmar sig gränsen. Kraften uppstår när TCP är inom c:a 5 cm från ett säkerhetsplan eller verktygsorienteringens avvikelse är c:a 3° från den angivna största avvikelsen.

När ett plan definierats som *Utlös reducerat läge* och TCP rör sig bortom denna gräns, går säkerhetssystemet över i *Reducerat läge*, som använder säkerhetsinställningarna för *Reducerat läge*. Utlösarplan följer samma regler som vanliga säkerhetsplan, förutom att robotarmen kan passera genom dem.

10.12.1 Välja en gräns att konfigurera

Panelen *Säkerhetsgränser* på flikens vänstra sida används för att välja vilken säkerhetsgräns som ska konfigureras.

Du lägger upp ett säkerhetsplan genom att klicka på någon av de åtta översta posterna i listan på panelen. Om det valda säkerhetsplanet redan har konfigurerats markeras motsvarande 3D-representation av planet i 3D-vyn (se 10.12.2) till höger om den här panelen. Säkerhetsplanet kan läggas upp i avsnittet *Egenskaper för säkerhetsplan* (se 10.12.3) längst ned på fliken.

Klicka på posten *Verktgsgräns* för att konfigurera gränsvärdet för robotverktygets orientering. Gränsens konfiguration kan anges i avsnittet *Egenskaper för verktgsgräns* (se 10.12.4) längst ned på fliken.

Klicka på knappen  /  för att växla 3D-visualiseringen av gränsvärdet. Om ett gränsvärde är aktivt anges *säkerhetsläge* (se 10.12.3 och 10.12.4) av någon av följande ikoner  /  /  / .

10.12.2 3D-visualisering

3D-vyn visar de konfigurerade säkerhetsplanen och gränsvärdet för robotverktygets orientering tillsammans med robotarmens aktuella position. Alla konfigurerade gränsposter där synlighetsomkopplaren valts (dvs. ikonen  visas) i avsnittet *Säkerhetsgränser* visas tillsammans med det gränsvärde som markerats.

Aktiva säkerhetsplan visas i gult och svart med en liten pil som representerar normalen mot planet, vilket anger den sida på planet där robotens TCP får placeras. Utlösarplan visas i blått och grönt. En liten pil markerar den sida av planet som *inte* utlöser en övergång till *Reducerat* läge. Om ett säkerhetsplan har markerats i panelen på flikens vänstra sida, markeras motsvarande 3D-representation.

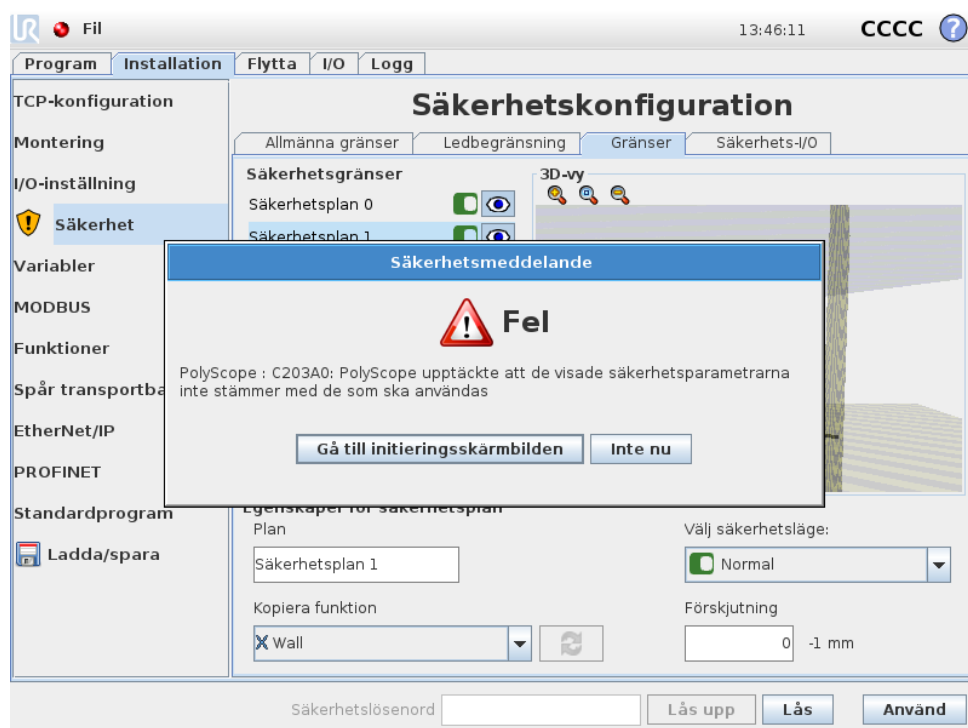
Verktygsorienteringens gränsvärde visas som en sfärisk kon tillsammans med en pil som anger robotverktygets nuvarande orientering. Konens insida representerar verktygsorienteringens tillåtna yta (vektor).

När ett plan eller gränsvärdet för verktygsorientering har konfigurerats men inte är aktivt är visualiseringen grå.

Tryck på förstöringsglaset för att zooma in/ut eller dra med ett finger över skärmen för att ändra vy.

10.12.3 Konfiguration av säkerhetsplan

Avsnittet *Egenskaper* för säkerhetsplan längst ned på fliken definierar konfigurationen för det markerade säkerhetsplanet i panelen *Säkerhetsgränser* på flikens övre vänstra del.



Namn Med textfältet `Namn` kan användaren tilldela ett namn till det markerade säkerhetsplanet. Ändra namnet genom att trycka på textfältet och skriva ett nytt namn.

Kopiera funktion Säkerhetsplanets position och normal anges med en funktion (se 13.12) från den aktuella robotinstallationen. Använd kombinationsrutan längst ned till vänster i avsnittet `Egenskaper för säkerhetsplan` för att välja en funktion. Endast funktioner av punkt- och plantyp är tillgängliga. Om du väljer `<Odefinierat>` rensas planets konfiguration.

Den valda funktionens z-axel kommer att peka mot det område som inte är tillåtet, och planets normal pekar i motsatt riktning, förutom när funktionen `Bas` valts, vilket gör att planets normal pekar i samma riktning. Om planet konfigurerats som *Utlös reducerat läge* (se 10.12.3) anger planets normal den sida av planet som *inte* utlöser en övergång till *Reducerat* läge.

Observera att när säkerhetsplanet har konfigurerats genom att en funktion väljs, blir positionsinformationen endast *kopierad* till säkerhetsplanet. Planet är *inte* länkat till funktionen. Detta betyder att om det görs ändringar i positionen eller orienteringen för en funktion som har använts för att konfigurera ett säkerhetsplan, kommer säkerhetsplanet inte att uppdateras automatiskt. Om funktionen har ändrats anges detta med ikonen  ovanför funktionsväljaren. Klicka på knappen  bredvid funktionsväljaren för att uppdatera säkerhetsplanet med funktionens aktuella position och orientering. Ikonen  visas också om den markerade funktionen har tagits bort från installationen.

Säkerhetsläge Kombinationsrutan på höger sida av panelen `Egenskaper för säkerhetsplan` används för att välja *säkerhetsläge* för säkerhetsplanet. Följande lägen finns:

Inaktiverad	Säkerhetsplanet är <i>aldrig aktivt</i> .
 Normal	När säkerhetssystemet är i läget <i>Normal</i> är ett plan i läget <i>Normal aktivt</i> och fungerar som en <i>strikt begränsning</i> för positionen för robotens TCP.
 Reducerat	När säkerhetssystemet är i läget <i>Reducerat</i> är ett plan i läget <i>Reducerat aktivt</i> och fungerar som en <i>strikt begränsning</i> för positionen för robotens TCP.
 Normal och Reducerat:	När säkerhetssystemet är i något av lägena <i>Normal</i> eller <i>Reducerat</i> är ett plan i läget <i>Normal</i> och <i>Reducerat aktivt</i> och fungerar som en <i>strikt begränsning</i> för positionen för robotens TCP.
 Utlös Reducerat läge	När säkerhetssystemet är i något av lägena <i>Normal</i> eller <i>Reducerat</i> är ett plan i läget <i>Utlös Reducerat läge aktivt</i> och gör att säkerhetssystemet växlar till läget <i>Reducerat</i> när robotens TCP finns bortom det.

Det *säkerhetsläge* som valts visas med en ikon i motsvarande post i panelen *Säkerhetsgränser*. Om *säkerhetsläget* angetts till *Inaktiverat* visas ingen ikon.

Förskjutning När en funktion har valts i kombinationsrutan i den nedre vänstra delen av panelen *Egenskaper för säkerhetsplan* kan säkerhetsplanet förskjutas genom att du trycker på textfältet *Förskjutning* nere till höger i den här panelen och anger ett värde. Om du skriver ett positivt värde ökas robotens tillåtna arbetsområde genom att planet flyttas i motsatt riktning mot planets normal, medan ett negativt värde minskar det tillåtna arbetsområdet genom att planet flyttas i normalens riktning.

Toleransen och enheten för gränsplanets förskjutning visas till höger om textfältet.

Effekt av plan som är strikt begränsning Programkörningen avbryts när TCP-positionen håller på att korsa ett aktivt säkerhetsplan med strikt begränsning, minus toleransen (se 10.4), om den fortsätter röra sig längs den förutsedda banan. Observera att det minustecken som visas med toleransvärdet bara anger att toleransen dras ifrån det verkliga värde som anges. Säkerhetssystemet utför ett Kategori 0-stopp om TCP-positionen överskrider den angivna gränsen för säkerhetsplanet (utan tolerans).

Effekt av plan i läget *Utlös Reducerat läge* När inget skyddsstopp är i kraft och säkerhetssystemet inte är i det speciella läget *Återställning* (se 10.6), är det i något av lägena *Normal* eller *Reducerat*, och robotarmens rörelse begränsas av respektive uppsättning gränser.

Som standard är säkerhetssystemet i läget *Normal*. Det går över i läget *Reducerat* när något av följande inträffar:

- Robotens TCP finns bortom något plan som angetts som *Utlös Reducerat läge*, d.v.s. den finns på den sida av planet som är *motsatt* riktningen på den lilla pilen i planets visualisering.
- Säkerhetsingångsfunktionen *Reducerat läge* har konfigurerats och signalerna är låga (mer information finns i 10.13).

När inget av ovanstående längre gäller går säkerhetssystemet tillbaka till läget *Normal*.

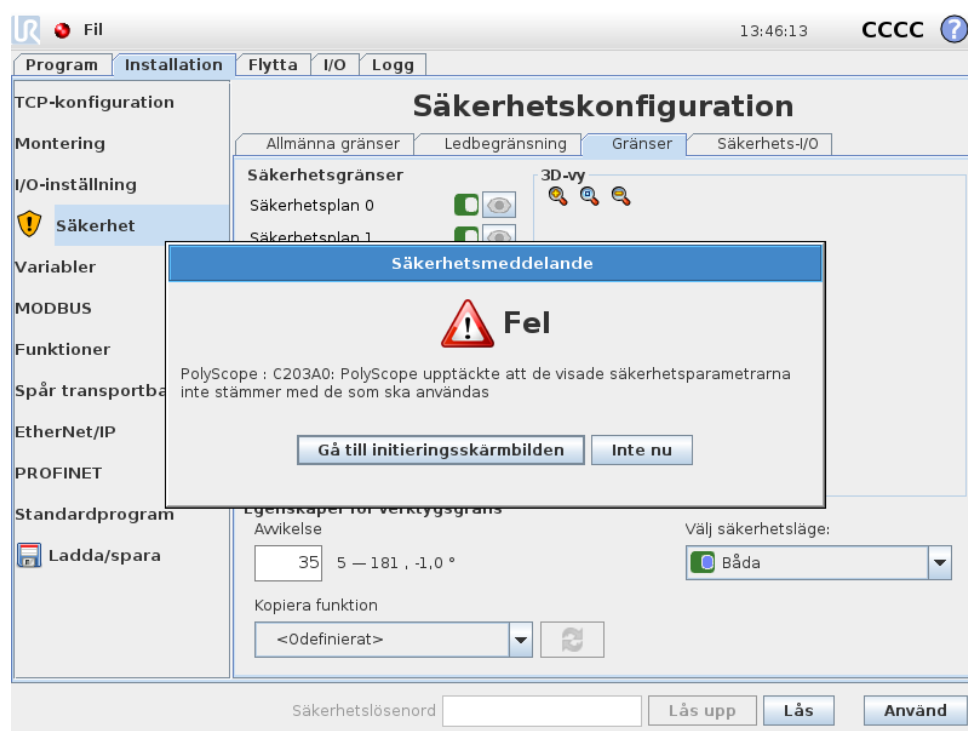
När övergången från läget *Normal* till läget *Reducerat* orsakas av passage genom ett plan som angetts som *Utlös Reducerat läge* inträffar en övergång från gränsuppsättningen för läget *Normal* till gränsuppsättningen för läget *Reducerat*. Så snart robotens TCP finns 20 mm eller närmare från planet som angetts som *Utlös Reducerat läge* (men fortfarande på sidan för läget *Normal*), används det mest tillåtande av gränsvärdena för lägena *Normal* och *Reducerat* för varje gränsvärde. När robotens TCP passerar genom planet som angetts som *Utlös Reducerat läge* är gränsuppsättningen för läget *Normal* inte längre aktiv, och gränsuppsättningen för läget *Reducerat* påtvingas.

När övergången från läget *Reducerat* till läget *Normal* orsakas av passage genom ett plan som angetts som *Utlös Reducerat läge* inträffar en övergång från gränsuppsättningen för läget *Reducerat* till gränsuppsättningen för läget *Normal*. Så snart robotens TCP

passerar genom planet som angetts som *Utlös Reducerat läge* används det mest tillåtande av gränsvärdena för lägena *Normal* och *Reducerat* för varje gränsvärde. När robotens TCP finns 20 mm eller mer från planet som angetts som *Utlös Reducerat läge* (på sidan för läget *Normal*) är gränsuppsättningen för läget *Reducerat* inte längre aktiv, och gränsuppsättningen för läget *Normal* påtvingas.

Om den förutsedda banan tar robotens TCP genom ett plan som angetts som *Utlös Reducerat läge* börjar robotarmen minska farten redan innan den passerar genom planet om den skulle överskriva ledhastigheten, verktygshastigheten eller rörelsemomentgränsen i den nya gränsuppsättningen. Observera att eftersom gränserna måste vara mer restriktiva i gränsuppsättningen för läget *Reducerat* kan sådan inbromsning i förväg endast inträffa vid övergång från läget *Normal* till läget *Reducerat*.

10.12.4 Konfiguration av verktygsgräns



Panelen *Egenskaper för verktygsgräns* längst ned på fliken definierar en gräns för robotverktygets orientering, bestående av en önskad orientering för verktyget och ett värde för den maximala tillåtna avvikelsen från denna orientering.

Avvikelse Textfältet *Avvikelse* visar värdet för den största tillåtna avvikelsen från den önskade orienteringen för robotverktygets orientering. Ändra värdet genom att trycka på textfältet och skriva ett nytt värde.

Det accepterade värdesintervallet tillsammans med avvikelsens tolerans och enhet visas bredvid textfältet.

Kopiera funktion Robotverktygets orientering anges med en funktion (se 13.12) från den aktuella robotinstallationen. Den valda funktionens z-axel används som vektor för önskad verktygsorientering för detta gränsvärde.

Använd kombinationsrutan längst ned till vänster i avsnittet *Egenskaper* för verktygsgräns för att välja en funktion. Endast funktioner av punkt- och plantyp är tillgängliga. Om du väljer <Odefinierat> rensas planets konfiguration.

Observera att när gränsen har konfigurerats genom att en funktion väljs, blir orienteringsinformationen endast *kopierad* till gränsen. Gränsen är *inte* länkad till funktionen. Detta betyder att om det görs ändringar i positionen eller orienteringen för en funktion som har använts för att konfigurera gränsvärdet, kommer gränsvärdet inte att uppdateras automatiskt. Om funktionen har ändrats anges detta med ikonen  ovanför funktionsväljaren. Klicka på knappen  bredvid funktionsväljaren för att uppdatera gränsvärdet med funktionens aktuella orientering. Ikonen  visas också om den markerade funktionen har tagits bort från installationen.

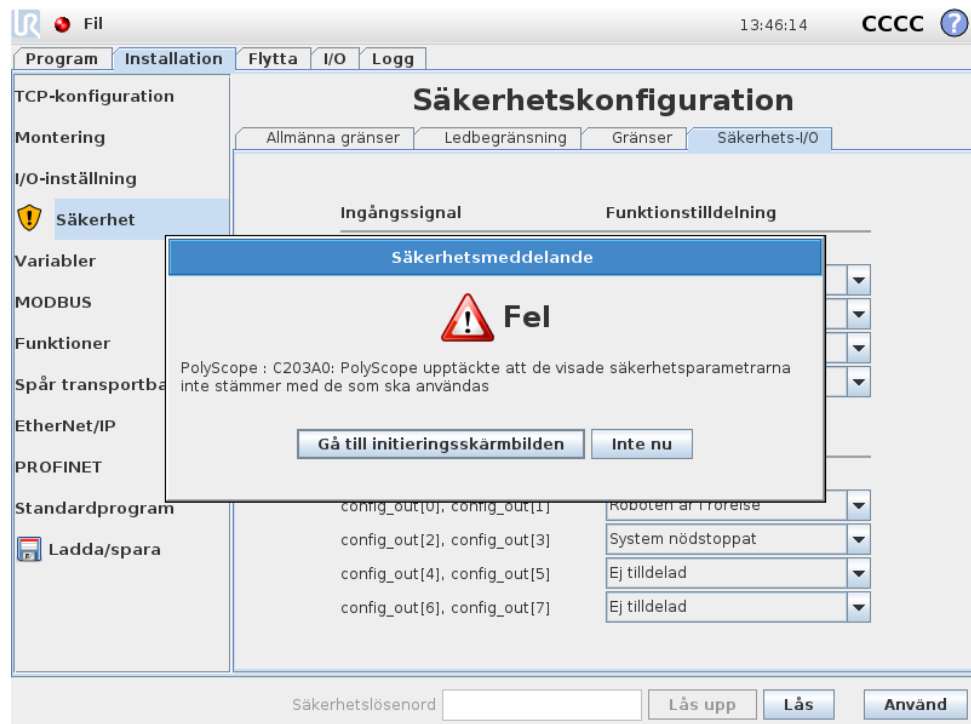
Säkerhetsläge Kombinationsrutan på höger sida av panelen *Egenskaper* för verktygsgräns används för att välja *säkerhetsläge* för verktygets orienteringsbegränsning. Följande lägen finns: Tillgängliga alternativ är:

Inaktiverad	Verktygets gränsvärde är aldrig aktivt.
 Normal	När säkerhetssystemet är i läget <i>Normal</i> är verktygets gränsvärde aktivt.
 Reducerat	När säkerhetssystemet är i läget <i>Reducerat</i> är verktygets gränsvärde aktivt.
 Normal och Reducerat:	När säkerhetssystemet är i läget <i>Normal</i> eller <i>Reducerat</i> är verktygets gränsvärde aktivt.

Det *säkerhetsläge* som valts visas med en ikon i motsvarande post i panelen *Säkerhetsgränser*. Om *säkerhetsläget* angetts till *Inaktiverat* visas ingen ikon.

Effekt Programkörningen avbryts när verktygsorienteringens avvikelse håller på att överstiga den maximala avvikelsen, minus toleransen (se 10.4), om den fortsätter röra sig längs den förutsedda banan. Observera att det minustecken som visas med toleransvärdet bara anger att toleransen dras ifrån det verkliga värde som anges. Säkerhetssystemet utför ett Kategori 0-stopp om verktygsorienteringens avvikelse överskrider den angivna gränsen (utan tolerans).

10.13 Säkerhets-I/O



Den här skärmen definierar *Säkerhetsfunktioner* för konfigurerbara in- och utgångar (I/O). Dessa I/O är uppdelade i in- och utgångar, och sätts ihop parvis så att varje funktion har en kategori² 3 och PLd I/O.

Varje *säkerhetsfunktion* kan bara styra ett par I/O. Om du försöker välja samma säkerhetsfunktion en gång till tas den bort från det första paret I/O där den först definierats. Det finns 5 *Säkerhetsfunktioner* för ingångssignaler och 4 för utgångssignaler.

10.13.1 Ingångssignaler

För ingångssignaler går det att välja följande *Säkerhetsfunktioner*: Systemnödstopp, Reducerat läge, Återställ skyddsstopp, Aktiveringsenhet för 3-position och Driftläge.

Systemnödstopp Efter konfigurering finns möjlighet att ha en extra Nödstoppsknapp bredvid Nödstoppsknapp på manöverenheten. Dessa funktioner kräver användning av utrustning som uppfyller ISO 13850.

Reducerat läge Alla säkerhetsgränser kan användas i två lägen: Läget *Normal*, som anger standardsäkerhetskonfigurationen, och läget *Reducerat* (mer information finns i 10.6). När den här ingångssäkerhetsfunktionen väljs orsakar en låg insignal att säkerhetssystemet övergår i läget *Reducerat*. Om det behövs minskar robotarmen farten så att den uppfyller gränsuppsättningen för läget *Reducerat*. Om robotarmen fortfarande bryter mot någon begränsning för läget *Reducerat* utförs

²Enligt ISO 13849-1, se ordlista för mer information.

ett kategori 0-stopp. Övergången tillbaka till läget *Normal* sker på samma sätt. Observera att säkerhetsplan också kan orsaka en övergång till läget *Reducerat* (mer information finns i 10.12.3).

Återställ skyddsstopp Om Skyddsstopp finns dragen i säkerhets-I/O används den här ingången för att se till att skyddsstoppet fortsätter tills en återställning utlöses. Robotarmen rör sig inte när den är i skyddsstoppat läge.



VARNING:

Funktionen Återställ skyddsstopp är i standardutförande konfigurerad för ingångsstift 0 och 1. En fullständig inaktivering innebär att robotarmens skyddsstopp upphör så snart insignalen Skyddsstopp blir hög. Med andra ord: utan Återställ skyddsstopp, kommer ingångarna Skyddsstopp i SI0 och SI1 (se Bruksanvisning för installation av maskinvara) att fastställa helt om tillståndet Skyddsstopp är aktivt eller inte.

Aktiveringsenhet för 3-position och Driftläge Med dessa kan du använda en 3-lägesenhet som extra skyddsanordning under robotens inställning och programmering. När ingången Aktiveringsenhet för 3-position har konfigurerats är roboten antingen i "körläge" eller "programmeringsläge". Det aktuella läget visas med en symbol i övre högra hörnet:

-  **Körläge:** Roboten kan bara utföra i förväg definierade uppgifter. Fliken Flytta och Frikörningsläget är inte tillgängliga.
-  **Programmeringsläge:** Begränsningarna från körläget är borttagna. Om insignalen från Aktiveringsenhet för 3-position är låg skyddsstoppas dock roboten. Hastighetsreglaget ställs in på ett ursprungsvärde som motsvarar 250 mm/s och kan ökas stegvis tills systemet når högre hastighet. Hastighetsreglaget återställs till det låga värdet varje gång ingången Aktiveringsenhet för 3-position växlar från låg till hög.

Det finns två metoder för konfigurering av driftlägesval:

1. Om du vill välja driftläge med en extern lägesväljarenhet ska du konfigurera ingången Driftläge. Konfigureringsmöjligheten visas i rullgardinsmenyerna när ingången Aktiveringsenhet för 3-position har konfigurerats. Roboten kommer att vara i körläge när ingången Driftläge är låg och i Programmeringsläge när den är hög.
2. Om du vill välja driftläge från Polyscope behöver endast ingången Aktiveringsenhet för 3-position konfigureras och verkställas i säkerhetskongfigurationen. I det här fallet är standardläget Körs (körläge). Om du vill växla till programmeringsläge väljer du knappen "Programmera robot" på välkomstskärmbilden. När du vill växla tillbaka till körläget lämnar du helt enkelt skärmbilden "Programmera robot".

**OBS:**

- När du har bekräftat konfigurationen av säkerhets-I/O med Aktiveringsenhet för 3-position visas välkomstkärmbilden automatiskt. Välkomstkärmbilden visas också automatiskt när driftläget växlar från *programmeringsläge* till *körläge*.
- Väljaren för fysiskt läge måste, om den används, helt uppfylla ISO 10218-1: artikel 5.7.1 för val.
- 3-Lägesväljaren, med tillhörande beteende, prestandaegenskaper och utförande, måste helt uppfylla ISO 10218-1: artikel 5.8.3 för en aktiveringsenhet.

10.13.2 Utgångssignaler

För utgångssignaler går det att använda följande *Säkerhetsfunktioner*. Alla signaler återgår till låg när det tillstånd som utlöste den höga signalen har avslutats.

Systemnödstopp Låg signal ges endast när säkerhetssystemet har utlösts och växlat till tillståndet Nödstoppad via ingången Robotnödstopp eller Nödstoppsknapp. För att undvika låsningar om tillståndet Nödstoppad har utlösts av ingången Systemnödstopp kommer låg signal inte att ges.

**OBS:**

Extern maskinutrustning som läser in tillståndet Nödstopp från roboten genom utgången Systemnödstopp måste uppfylla ISO 13850. Detta är särskilt viktigt i system där ingången Robotnödstopp är ansluten till en extern nödstoppsenhet. I dessa fall kommer utgången Systemnödstopp att växla till hög när den externa nödstoppsenheten frikopplas. Detta innebär att nödstoppstillståndet hos den externa maskinutrustningen återställs utan att någon manuell åtgärd krävs från robotoperatören. För att uppfylla säkerhetsstandarderna måste den externa maskinutrustningen kunna återställas manuellt för att driften ska kunna fortsätta.

Roboten är i rörelse En låg signal ges när robotarmen är i rörelse. När robotarmen är stilla ges hög signal.

Roboten stoppar inte När en begäran att stoppa robotarmen har skickats går det en viss tid från begäran tills armen stoppats. Under denna tid är signalen hög. När robotarmen är i rörelse och inte har fått en begäran om att stanna, eller när robotarmen är stoppad, är signalen låg.

Reducerat läge Skickar en låg signal när robotarmen placeras i läget *Reducerat* eller om säkerhetsingången konfigurerats med en ingång för *Reducerat läge* och signalen för närvarande är låg. Annars är signalen hög.

Ej reducerat läge Detta är motsatsen till *Reducerat läge* enligt definitionen ovan.

11 Börja programmeringen

11.1 Inledning

Universal Robot-armen är en arm som består av rör och leder. Lederna och deras vanligaste benämningar visas i figur 11.1. **Basen** är platsen där roboten är monterad. Robotens verktyg sitter i den andra änden (**Handled 3**). Genom att koordinera rörelsen i var och en av lederna kan roboten flytta sitt verktyg fritt, med undantag för området rakt ovanför och rakt under basen.

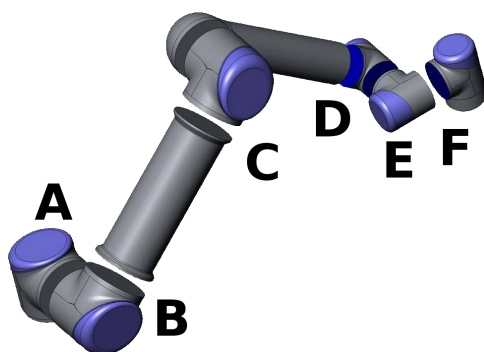
PolyScope är det grafiska användargränssnittet med vilket du kan manövrera robotarmen och manöverskåpet, kör robotprogram och skapar nya program på ett enkelt sätt.

Följande avsnitt hjälper dig att komma igång med roboten. Efteråt förklaras skärmarna och funktionen i PolyScope närmare.



FARA:

1. Kapitlet Bruksanvisning för installation av maskinvara innehåller viktig säkerhetsinformation, som måste läsas och förstås av den som integrerar UR-robotar innan roboten startas för första gången.
2. Integratören måste ställa in säkerhets- och konfigurationsparametrarna som har definierats i riskbedömningen innan robotarmen startas för första gången, se kapitlet 10.



Figur 11.1: Robotens leder. A: Bas, B: Skuldra, C: Armbåge samt D, E, F: Handled 1, 2, 3

11.2 Komma igång

Innan du använder PolyScope måste robotarmen och manöverskåpet installeras och manöverskåpet sättas på.

11.2.1 Installera robotarmen och manöverskåpet

Gör så här för att installera robotarmen och manöverskåpet:

1. Packa upp robotarmen och manöverskåpet.
2. Montera roboten på ett stabilt och vibrationsfritt underlag.
3. Placera manöverskåpet på den avsedda foten.
4. Sätt i robotsladden mellan roboten och manöverskåpet.
5. Anslut manöverskåpet till elnätet.



VARNING:

Risk att roboten välter. Om roboten inte monteras på en stabil yta kan den falla och orsaka personskada.

Detaljerade installationsinstruktioner finns i Bruksanvisning för installation av maskinvara. Observera att en riskutvärdering krävs innan robotarmen kan användas för att utföra något arbete.

11.2.2 Sätta på och stänga av manöverskåpet

Manöverskåpet startas genom att trycka på strömbrytaren på framsidan av panelen med pekskärmen. Panelen kallas vanligen *manöverenheten*. När manöverskåpet är på visas text från det underliggande operativsystemet på pekskärmen. Efter cirka en minut visas några knappar på skärmen, och ett meddelandefönster leder användaren till startskärmen (se 11.5).

Stäng av manöverskåpet genom att trycka på den gröna strömbrytaren på skärmen eller med knappen *Stäng av* på välkomstskärmen (se 11.4).



VARNING:

Om du stänger av roboten genom att dra ut kontakten ur vägguttaget kan robotens filsystem skadas, vilket kan leda till funktionsfel hos roboten.

11.2.3 Sätta på och stänga av robotarmen

Roboten kan startas om manöverskåpet har startats och om inga nödstoppsknappar är intryckta. Roboten startas från skärmbilden *Initiering* (se 11.5) genom att trycka in *PÅ*-knappen på skärmen och sedan trycka på *Start*. Observera att när roboten startas ger den ett ljud ifrån sig, och den rör sig lite när bromsarna släpps.

Strömmen till robotarmen kan stängas av genom att röra vid *AV*-knappen på skärmen *Initiering*. Robotarmen stängs också av automatiskt när manöverskåpet stängs av.

11.2.4 Snabbstart

Gör så här för att snabbt starta roboten efter installationen:

1. Tryck på nödstoppknappen på manöverenheten framsida.
2. Tryck på strömbrytaren på manöverenheten.
3. Vänta ett ögonblick medan systemet startar (text visas på pekskärmen).
4. När systemet är klart visas en ruta på pekskärmen med information om att roboten måste initieras.
5. Tryck på knappen i meddelandet. Du går till initieringsskärmen.
6. Vänta på dialogrutan *Bekräftelse av använd säkerhetskonsfiguration* och tryck på knappen *Bekräfta säkerhetskonsfiguration*. Detta tillämpar en ursprunglig uppsättning säkerhetsparametrar som behöver justeras baserat på en riskutvärdering.
7. Lås upp nödstoppsknappen. Robotens tillstånd växlar från *Nödstoppad* till *Ström av*.
8. Gå ut ur robotens arbetsområde.
9. Tryck på knappen *På* på pekskärmen. Vänta några sekunder tills robotens läge ändrats till *Viloläge*.
10. Kontrollera att nyttolastens massa och den montering som valts är korrekta. Du får ett meddelande om den montering som sensorerna känner av inte stämmer med den montering som valts.
11. Tryck på knappen *Start* på pekskärmen. Roboten ger nu ifrån sig ett ljud och rör sig en aning medan bromsarna lossas.
12. Tryck på knappen *OK*, så går du tillbaka till välkomstskärmen.

11.2.5 Det första programmet

Ett program är en lista över kommandon som berättar för roboten vad den ska göra. PolyScope gör det möjligt för personer som bara har lite programmeringserfarenhet att programmera roboten. För de flesta uppgifter görs programmeringen helt från pekskärmen utan att några kryptiska kommandon behöver skrivas in.

Eftersom verktygsrörelsen är en viktig del av ett robotprogram är det avgörande att det finns ett sätt att lära roboten hur den ska röra sig. I PolyScope anges verktygets rörelser genom en serie **vägpunkter**, d.v.s. punkter i robotens arbetsområde. En vägpunkt kan anges genom att flytta roboten till en viss position. Den kan även beräknas av programvaran. För att flytta robotarmen till en viss position använder du antingen fliken *Flytta* (se 13.1), eller drar bara robotarmen på plats medan du håller knappen **Frikörning** på manöverenhetens baksida intryckt.

Förutom förflyttning via vägpunkter kan programmet skicka I/O-signaler till andra maskiner vid vissa punkter längs robotens rörelsebana, och utföra kommandon som **om...så** och **loop**, baserat på variabler och I/O-signaler.

Så här startar du ett enkelt program på en robot som startats:

1. Tryck på knappen *Programmera robot* och välj *Tomt program*.

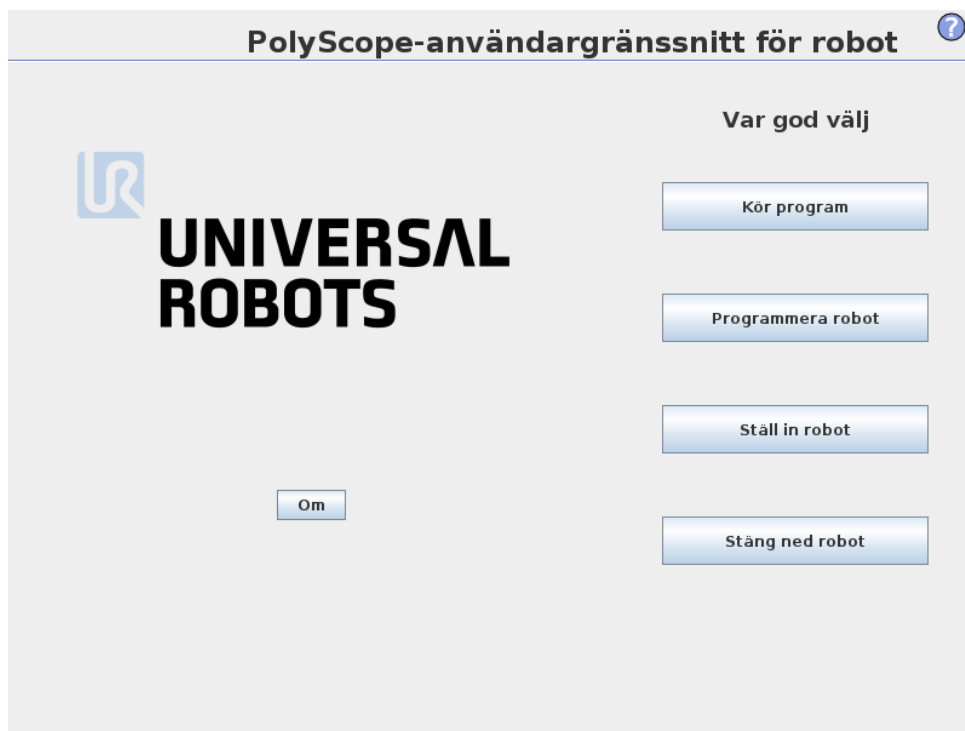
2. Tryck på knappen **Nästa** (nederst till höger) så att raden `<tom>` markeras i trädstrukturen till vänster på skärmen.
3. Gå till fliken **Struktur**.
4. Tryck på knappen **Flytta**.
5. Gå till fliken **Kommando**.
6. Tryck på knappen **Nästa** för att komma till inställningarna för **Vägpunkt**.
7. Tryck på knappen **Ange den här vägpunkten bredvid bilden “?”**.
8. Gå till skärmen **Flytta** och flytta roboten genom att trycka på de olika blå pilarna, eller flytta roboten genom att hålla in knappen **Frikörning** på baksidan av manöverenheten medan du drar i robotarmen.
9. Tryck på **OK**.
10. Tryck på **Lägg till vägpunkt före**.
11. Tryck på knappen **Ange den här vägpunkten bredvid bilden “?”**.
12. Gå till skärmen **Flytta** och flytta roboten genom att trycka på de olika blå pilarna, eller flytta roboten genom att hålla in knappen **Frikörning** medan du drar i robotarmen.
13. Tryck på **OK**.
14. Programmet är klart. Roboten kommer nu att röra sig mellan de två punkterna när du trycker på symbolen **“Spela upp”**. Ställ dig på behörigt avstånd, håll i nödstoppsknappen och tryck på **“Spela upp”**.
15. Grattis! Du har nu skapat ditt första robotprogram som förflyttar roboten mellan de två givna vägpunkterna.

**VARNING:**

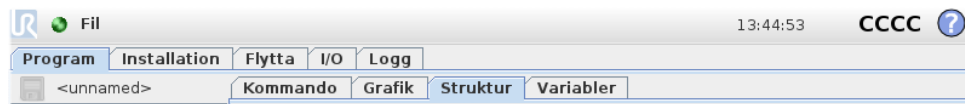
1. Kör inte in roboten i sig själv eller något föremål, eftersom detta kan skada roboten.
2. Håll ditt huvud och din torso utanför robotens räckvidd (arbetsområde). Placera inte fingrar där de kan fastna.
3. Detta är bara en snabbstartsguide, som visar hur enkelt det är att använda en UR-robot. Den förutsätter en ofarlig miljö och en mycket noggrann användare. Öka inte hastigheten eller accelerationen över standardvärdena. Utför alltid en riskvärdering innan roboten tas i drift.

11.3 Programmeringsgränssnittet i PolyScope

PolyScope körs från den beröringskänsliga pekskärmen som sitter på manöverskåpet.



Bilden ovan visar välkomstskärmbilden. De blåaktiga områdena på skärmen är knappar som kan tryckas in genom att trycka med fingret eller den trubbiga änden av en penna mot skärmen. PolyScope har en hierarkisk skärmstruktur. I programmeringsmiljön är skärmbilderna anordnade i *flikar*, för enkel åtkomst på skärmbilderna.

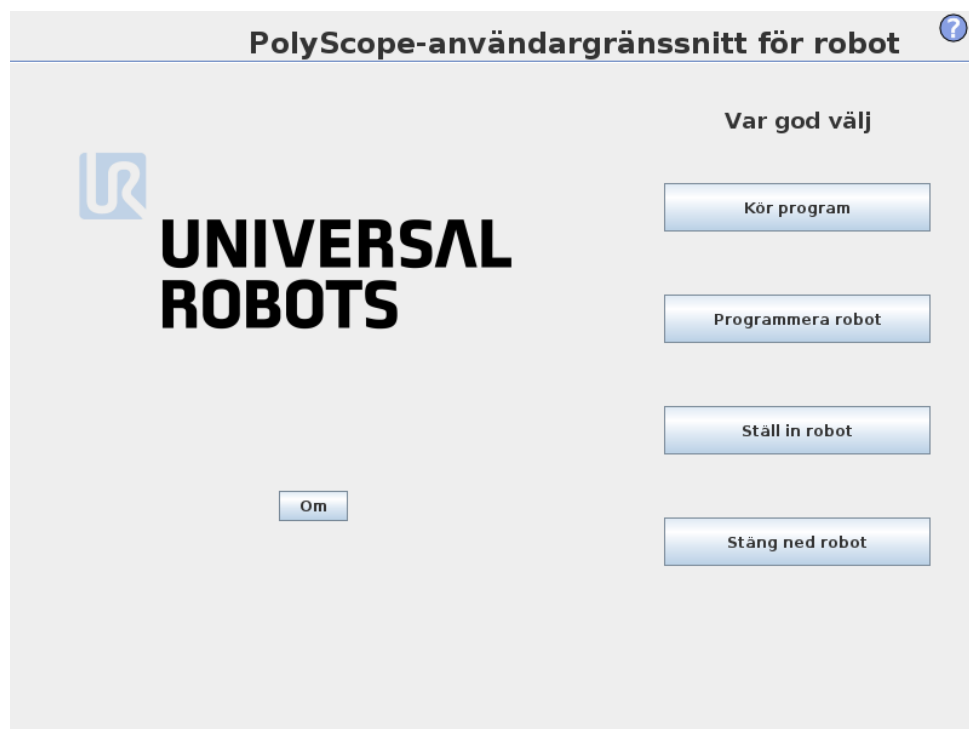


I det här exemplet är fliken **Program** markerad på översta nivån och under detta är fliken **Struktur** markerad. Fliken **Program** innehåller information om det program som laddats. Om du väljer fliken **Flytta**, växlar skärmbilden till skärmbilden **Flytta**, varifrån robotarmen kan flyttas. På samma sätt kan du välja fliken **I/O** för att se och redigera tillstånden för de elektriska I/O-enheterna.

Du kan ansluta en mus och ett tangentbord till manöverskåpet eller manöverenheten, men det är inte något krav. Nästan alla textfält har pekfunktion, så om du trycker på dem med fingret öppnas en knappsats eller ett tangentbord på skärmen.

De olika skärmbilderna i PolyScope beskrivs i följande avsnitt.

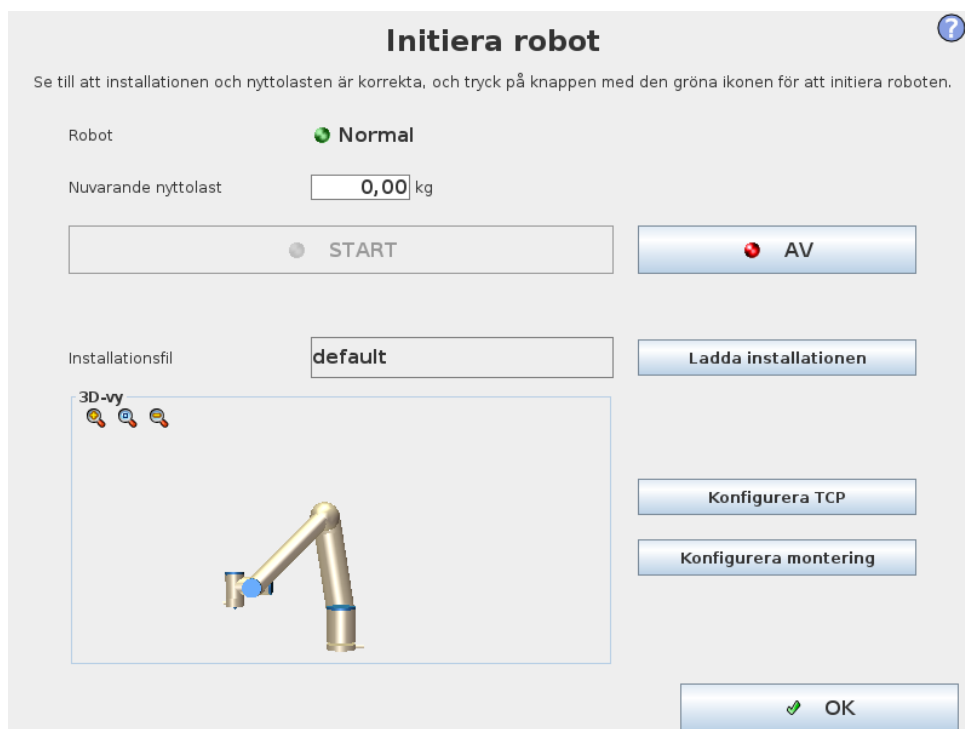
11.4 Välkomstskärmbilden



När manöverdatoren startats visas välkomstskärmbilden. Skärmbilden innehåller följande alternativ:

- **Kör program:** Välj och kör ett befintligt program. Detta är det enklaste sättet att hantera robotarmen och manöverskåpet.
- **Programmera robot:** Ändra ett program eller skapa ett nytt program.
- **Ställ in robot:** Ändra språk, ställ in lösenord, uppgradera programvara osv.
- **Stäng ned robot:** Stänger av robotarmen och manöverskåpet.
- **Om:** Här finns uppgifter om programversioner, värnhamn, IP-adress, serienummer samt juridisk information.

11.5 Initieringsskärm



På den här skärmbilden styr du robotarmens initiering.

Statusindikator för robotarmen

LED-lampan för status anger robotarmens körläge:

- En klarröd LED anger att robotarmen är i stoppat läge. Det kan finnas flera orsaker till detta.
- En klargul LED anger att robotarmen är strömsatt, men att den inte är klar för normal drift.
- En grön LED anger att robotarmen är strömsatt, och att den är klar för normal drift.

Den text som visas bredvid LED-lampan anger robotarmens aktuella tillstånd närmare.

Aktiv nyttolast och installation

När robotarmen är strömsatt visas i den lilla vita textrutan den nyttolastmassa som används av manöverenheten när robotarmen hanteras. Det går att ändra värdet genom att trycka på textfältet och skriva ett nytt värde.

Observera att inställningen av värdet inte ändrar nyttolasten i robotens installation (se 13.6), det anger bara den nyttolastmassa som används av manöverenheten.

På samma sätt visas i det grå textfältet namnet på den installationsfil som är inläst. Det går att läsa in en annan installation genom att trycka på textfältet, eller genom att använda knappen **Ladda** bredvid fältet. Det går även att anpassa den laddade installationen med knapparna bredvid 3D-vyn längst ner på skärmen.

Innan robotarmen startas är det mycket viktigt att verifiera att både den aktiva nyttolasten och den aktiva installationen motsvarar den verkliga situationen för robotarmen.

Initiera robotarmen



FARA:

Verifiera alltid att både den aktiva nyttolasten och den aktiva installationen är korrekta innan du startar robotarmen. Om inställningarna är felaktiga kommer inte robotarmen och manöverskåpet att fungera korrekt, och de kan utgöra en fara för personer och utrustning i närheten.



FÖRSIKTIGT:

Var mycket försiktig om robotarmen rör vid ett hinder eller bord, eftersom ledens växellåda kan skadas om robotarmen körs in i ett hinder.

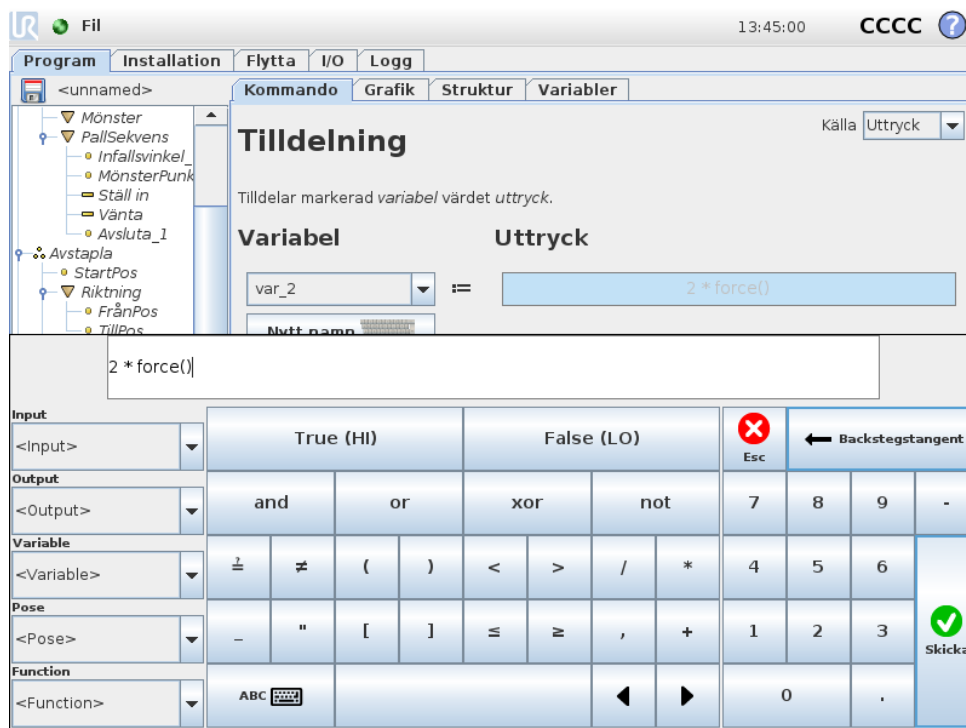
Den stora knappen med den gröna ikonen utför själva initieringen av robotarmen. Texten på knappen och den åtgärd som utförs ändras sig beroende på robotarmens aktuella tillstånd.

- När kontroll datorn startats måste du trycka på knappen en gång för att strömsätta robotarmen. Robotarmens tillstånd ändras då till **Ström på** och sedan till **Viloläge**. Observera att vid nödstopp går det inte att strömsätta robotarmen, så knappen visas som gråtonad.
- När robotarmens tillstånd är **Viloläge** måste du trycka på knappen igen för att starta robotarmen. Då kontrolleras sensordata mot robotarmens konfigurerade montering. Om de inte stämmer överens (inom toleransen 30°) inaktiveras knappen och ett felmeddelande visas under den.
Om monteringen verifieras, släpps alla ledbromsar när du trycker på knappen, och robotarmen är klar för normal drift. Observera att roboten ger ett ljud ifrån sig och rör sig lite när bromsarna släpps.
- Om robotarmen överträder någon av säkerhetsgränserna efter start arbetar den i ett speciellt **återställningsläge**. I det här läget kommer en tryckning på knappen att växla till en skärm för återställningsförflyttning, där det går att flytta robotarmen tillbaka inom säkerhetsgränserna.
- Om ett fel inträffar kan manöverenheten startas om med knappen.
- Om manöverenheten inte är igång, startas den vid tryckning på knappen.

Den mindre knappen med en röd ikon stänger av robotarmen.

12 Redigeringsprogram på skärmen

12.1 Redigeringsprogram för uttryck



Även om själva uttrycket redigeras som text har redigeringsprogrammet för uttryck ett antal knappar och funktioner som kan infoga särskilda uttryckssymboler, till exempel * för multiplikation och $\leq$ för mindre än eller lika med. Knappen med tangentbordssymbolen i skärmens övre vänstra hörn växlar till textredigering av uttrycket. Alla definierade variabler kan sättas med väljaren **Variabel** medan namnen på ingångs- och utgångsportarna kan ställas in med väljarna **Ingång** och **Utgång**. Vissa specialfunktioner finns under **Funktion**.

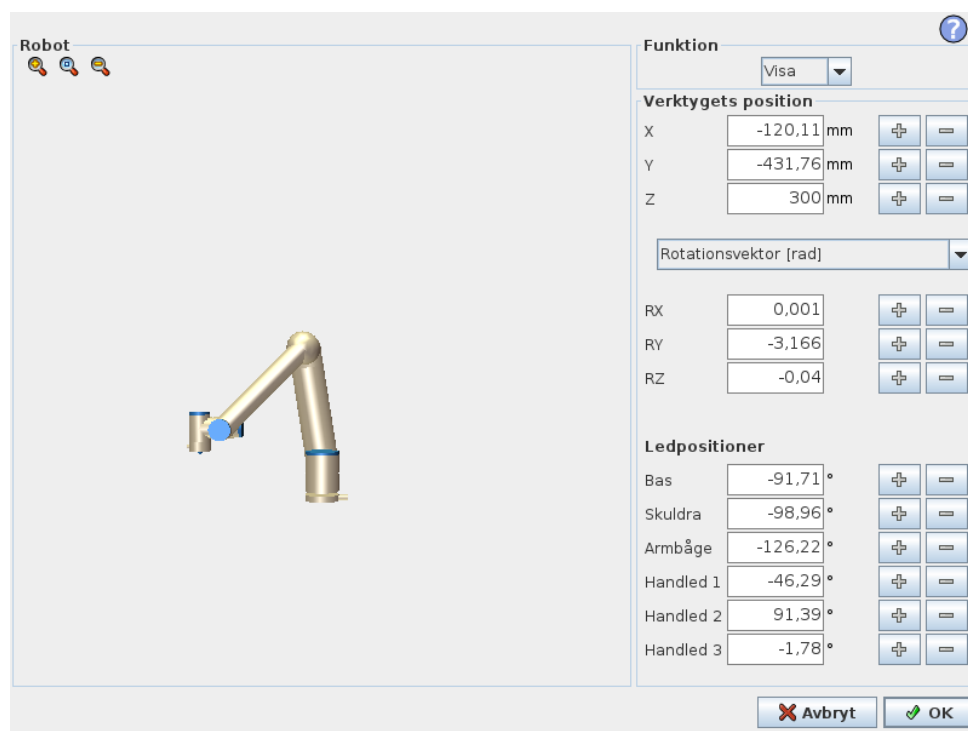
Uttrycket kontrolleras avseende grammatikfel när du trycker på OK. Om du trycker på Avbryt stängs skärmbilden och alla ändringar ignoreras.

Ett uttryck kan se ut så här:

```
digital.in[1] = Sant och analog.in[0] < 0.5
```

12.2 Skärmbilden Poseredigering

På den här skärmbilden kan du ange målpositioner för lederna eller en målpose (position och orientering) för robotverktyget. Den här skärmbilden är "offline" och styr inte robotarmen direkt.



Robot

Robotarmens nuvarande position och angiven ny målposition visas i 3D-grafik. 3D-ritningen över robotarmen visar robotarmens aktuella position, och "skuggan" av robotarmen visar målpositionen för robotarmen, som styrs av angivna värden till höger på skärmen. Tryck på förstöringsglaset för att zooma in/ut eller dra med ett finger över skärmen för att ändra vy.

Om den angivna målpositionen för robotens TCP ligger nära ett säkerhets- eller utlösningssplan, eller om robotverktygets orientering är nära verktygets orienteringsbegränsning (se 10.12), visas en 3D-representation av närhetsgränsen.

Säkerhetsplan visas i gult och svart med en liten pil som representerar normalen mot planet, vilket anger den sida på planet där robotens TCP får placeras. Utlösarplan visas i blått och grönt med en liten pil som pekar på planets sida, där begränsningarna i läget *Normal* (se 10.6) är aktiva. Verktygsorienteringens gränsvärde visas som en sfärisk kon tillsammans med en pil som anger robotverktygets nuvarande orientering. Konens insida representerar verktygsorienteringens tillåtna yta (vektor).

När målet för robotens TCP inte längre är i närheten av gränsvärdet försvinner 3D-representationen. Om mål-TCP överträder eller är mycket nära att överträda ett gränsvärde visas gränsvärdets visualisering som röd.

Funktioner och verktygsposition

Längst upp till höger på skärmen visas funktionsväljaren. Där definierar funktionsväljaren vilken funktion som ska styra robotarmen.

Under funktionsväljaren visas namnet på aktiv TCP. Med information om att konfigurera flera namngivna TCP finns i 13.6. Textrutorna visar fullständiga koordi-

natvärden för denna TCP jämfört med vald funktion. X, Y och Z styr verktygets position, medan RX, RY och RZ styr verktygets orientering.

Använd kombinationsrutan ovanför textrutorna RX, RY och RZ för att välja hur orienteringen representeras. Tillgängliga typer är:

- **Rotationsvektor [rad]** Orienteringen anges som en *rotationsvektor*. Axelns längd är den vinkel som ska roteras i radianer, och vektorn själv ger den axel runt vilken rotationen ska göras. Detta är standardinställningen.
- **Rotationsvektor [°]** Orienteringen anges som en *rotationsvektor*, där vektorns längd är den vinkel som ska roteras, angiven i grader.
- **RPY [rad]** Vinklar för *roll, tipp* och *gir* (RPY, Roll, Pitch, Yaw), där vinklarna angetts i radianer. RPY-rotationsmatrisen (X, Y', Z''-rotation) ges av:

$$R_{rpy}(\gamma, \beta, \alpha) = R_Z(\alpha) \cdot R_Y(\beta) \cdot R_X(\gamma)$$

- **RPY-vinklar [°]** Roll, tipp och gir (RPY) där vinklarna anges i grader.

Du kan redigera värdena genom att klicka på koordinaten. Om du klickar på knapparna + och – precis till höger om en ruta kan du lägga till eller dra bort ett belopp till eller från aktuellt värde. Om du trycker på en knapp och håller den nedtryckt ökas eller minskas värdet direkt. Ju längre knappen är nedtryckt, desto större blir ökningen eller minskningen.

Ledpositioner

Gör det möjligt att ange positionen för de enskilda lederna direkt. Varje ledposition kan ha ett värde i intervallet -360° till $+360^\circ$, vilket är *ledbegränsningarna*. Du kan redigera värdena genom att klicka på ledens position. Om du klickar på knapparna + och – precis till höger om en ruta kan du lägga till eller dra bort ett belopp till eller från aktuellt värde. Om du trycker på en knapp och håller den nedtryckt ökas eller minskas värdet direkt. Ju längre knappen är nedtryckt, desto större blir ökningen eller minskningen.

Knappen OK

Om den här skärmbilden aktiverades från fliken Flytta (se 13.1), kommer en klickning på knappen OK att ta dig tillbaka till fliken Flytta där robotarmen förflyttas till angivet mål. Om det senast angivna värdet var en verktygskoordinat, kommer robotarmen att förflytta sig till målpositionen med rörelsetypen FlyttaL, medan robotarmen flyttar sig till målpositionen med rörelsetypen FlyttaLed om det var en ledposition som senast angavs. De olika rörelsetyperna beskrivs i 14.5.

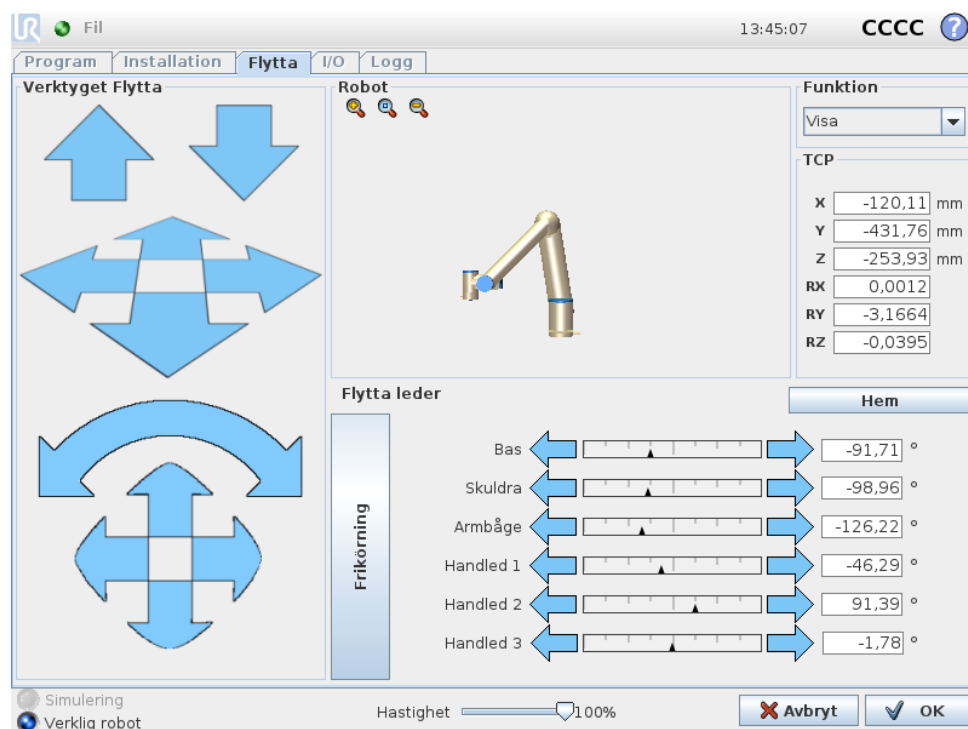
Knappen Avbryt

Om du trycker på Avbryt stängs skärmbilden och alla ändringar ignoreras.

13 Robotstyrning

13.1 Fliken Flytta

På den här skärmbilden kan du alltid flytta (knuffa) robotarmen direkt, antingen genom att flytta/rotera robotverktyget eller genom att flytta robotlederna var för sig.



13.1.1 Robot

Robotarmens nuvarande position visas i 3D-grafik. Tryck på förstoringsglasat för att zooma in/ut eller dra med ett finger över skärmen för att ändra vy. Du får den bästa känslan för att styra robotarmen om du väljer funktionen **Visa** och roterar visningsvinkeln i 3D-teckningen så att den matchar hur du ser den verkliga robotarmen.

Om den aktuella positionen för robotens TCP kommer nära ett säkerhets- eller utlösningssplan, eller om robotverktygets orientering är nära verktygets orienteringsbegränsning (se 10.12), visas en 3D-representation av närhetsgränsen. Observera att när roboten kör ett program är visualiseringen av gränsvärden inaktiverad.

Säkerhetsplan visas i gult och svart med en liten pil som representerar normalen mot planet, vilket anger den sida på planet där robotens TCP får placeras. Utlösarplan visas i blått och grönt med en liten pil som pekar på planets sida, där begränsningarna i läget **Normal** (se 10.6) är aktiva. Verktygsorienteringens

gränsvärde visas som en sfärisk kon tillsammans med en pil som anger robotverktygets nuvarande orientering. Konens insida representerar verktygsorienteringens tillåtna yta (vektor).

När robotens TCP inte längre är i närheten av gränsvärdet försvinner 3D-representationen. Om TCP överträder eller är mycket nära att överträda ett gränsvärde visas gränsvärdets visualisering som röd.

13.1.2 Funktioner och verktygsposition

Längst upp till höger på skärmen visas funktionsväljaren. Där definieras relativt vilken funktion robotarmen ska styras.

Namnet på aktiv TCP visas bredvid funktionsväljaren. Textrutorna visar fullständiga koordinatvärden för denna TCP jämfört med vald funktion. Mer information om att konfigurera flera namngivna TCP:er (finns i 13.6).

Du kan redigera värdena manuellt genom att klicka på koordinaten eller ledens position. Detta tar dig till skärmbilden för att pose-redigering (se 12.2) där du kan ange en målposition och orientering för verktyget eller målpositioner för lederna.

13.1.3 Verket Flytta

- Om du håller ned en *förflyttningspil* (överst) flyttas robotens verktygs-spets i den riktning som visas.
- Om du håller ned en *rotationspil* (nederst) ändras robotverktygets orientering i den riktning som visas. Rotationspunkten är verktygets centrumpunkt eller Tool Center Point (TCP), d.v.s. den punkt i änden av robotarmen som ger en karakteristisk punkt på robotens verktyg. TCP-punkten visas som ett litet blått klot.

Obs: Du kan när som helst släppa knappen för att stoppa rörelsen

13.1.4 Flytta leder

Gör det möjligt att styra de enskilda lederna direkt. Varje led kan förflyttas från -360° till $+360^\circ$, som är de standardledgränsvärden som illustreras av den vågräta stapeln för respektive led. Om en led når gränsvärdet kan den inte föras längre. Om gränserna för en led har konfigurerats med ett annat positionsintervall än det som är standard (se 10.11), anges intervallet med rött i den vågräta stapeln.

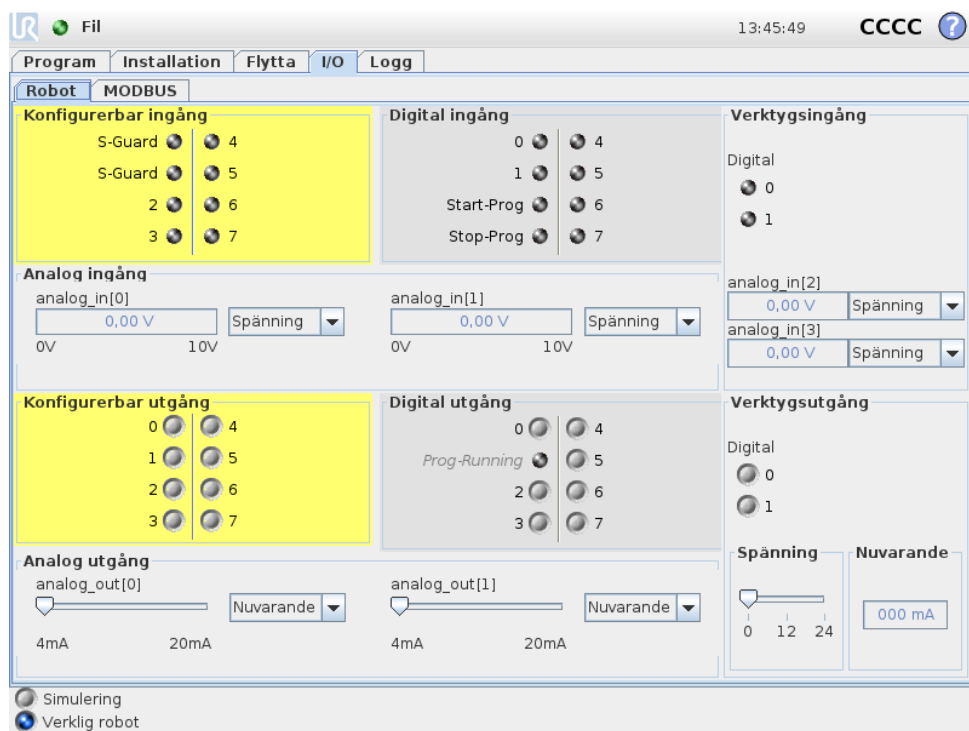
13.1.5 Frikörning

Om knappen *Frikörning* hålls ned går det att ta tag i robotarmen fysiskt och dra den till önskad plats. Om inställningen för gravitation (se 13.7) i **Installation** är fel, eller om robotarmen bär en tung last, kommer robotarmen att börja röra sig (falla) när knappen **Frikörning** trycks in. I så fall är det bara att släppa upp knappen **Frikörning** igen.

**VARNING:**

1. Se till att använda korrekt installationsinställningar (t.ex. Robotmontering och vinkel, vikt i TCP, TCP-förskjutning). Spara och ladda installationsfilerna tillsammans med programmet.
2. Se till att TCP-inställningarna och inställningarna för robotmonteringen är korrekta innan du använder knappen **Frikörning**. Om inställningarna inte är korrekta kommer robotarmen att röra sig när knappen **Frikörning** aktiveras.
3. Frikörningsfunktionen (**Impedans/Backdrivning**) ska bara användas i installationer där riskutvärderingen medger detta. Verktyg och hinder får inte ha vassa kanter eller klämpunkter. Se till att all personal är utom räckhåll för robotarmen.

13.2 Fliken I/O



På den här skärmen kan du alltid övervaka och ställa in I/O-signaler från/till robotens manöverskåp i realtid. Skärmen visar aktuellt I/O-tillstånd, även under programexekvering. Om något ändras under programexekveringen kommer programmet att stoppas. Vid programstopp kommer alla utsignaler att bibehålla sina tillstånd. Skärmen uppdateras bara vid 10 Hz, och en mycket snabb signal kanske inte visas korrekt.

Konfigurerbara in- och utgångar kan reserveras för speciella säkerhetsinställningar, som definieras i avsnittet för konfiguration av säkerhets-I/O för installationen (se 10.13);

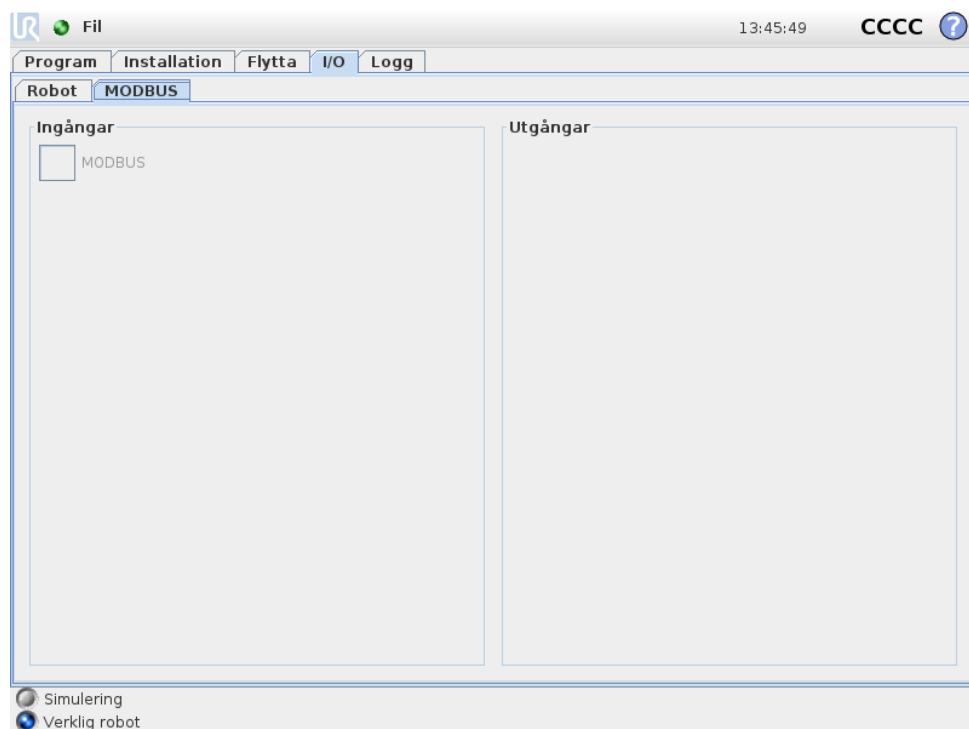
De som är reserverade kommer att ha säkerhetsfunktionens namn istället för standardnamnet eller det användardefinierade namnet. Konfigurerbara utgångar som är reserverade för säkerhetsinställningar kan inte växlas och visas endast som LED-lampor.

Signalernas elektriska egenskaper beskrivs i användarhandboken ??.

Inställningar för analog domän De analoga utgångarna kan ställas in på antingen ström [4–20 mA] eller spänning [0–10 V]. Systemet kommer ihåg inställningarna om de skulle behövas vid omstart av robotstyrenheten när ett program sparas.

13.3 I/O för MODBUS-klient

Här visas de digitala I/O-signalerna för MODBUS-klienten som angetts i installationen. Om signalanslutningen går förlorad inaktiveras motsvarande post på den här skärmbilden.



Ingångar

Visar status för MODBUS-klientens digitala ingångar.

Utgångar

Visar och växlar status för MODBUS-klientens digitala utgångar. Det går bara att växla en signal om detta tillåts av valet på flikkontrollen I/O (se 13.8).

13.4 Fliken AutoFlytta

Fliken AutoFlytta används när robotarmen måste förflytta sig till en specifik position inom sitt arbetsutrymme. Exempel på detta kan vara när roboten måste flyttas till ett programs startposition innan programmet körs, eller vid förflyttning till en punkt längs färdvägen när ett program modifieras.



Animering

Animeringen visar den rörelse som robotarmen ska utföra.

**FÖRSIKTIGT:**

Jämför animationen med den verkliga robotarmens position och kontrollera att roboten kan utföra rörelsen säkert utan att slå emot några hinder.

**FÖRSIKTIGT:**

AutoFlytta-funktionen flyttar längs roboten längs skuggbanan. Kollisioner kan skada roboten eller annan utrustning.

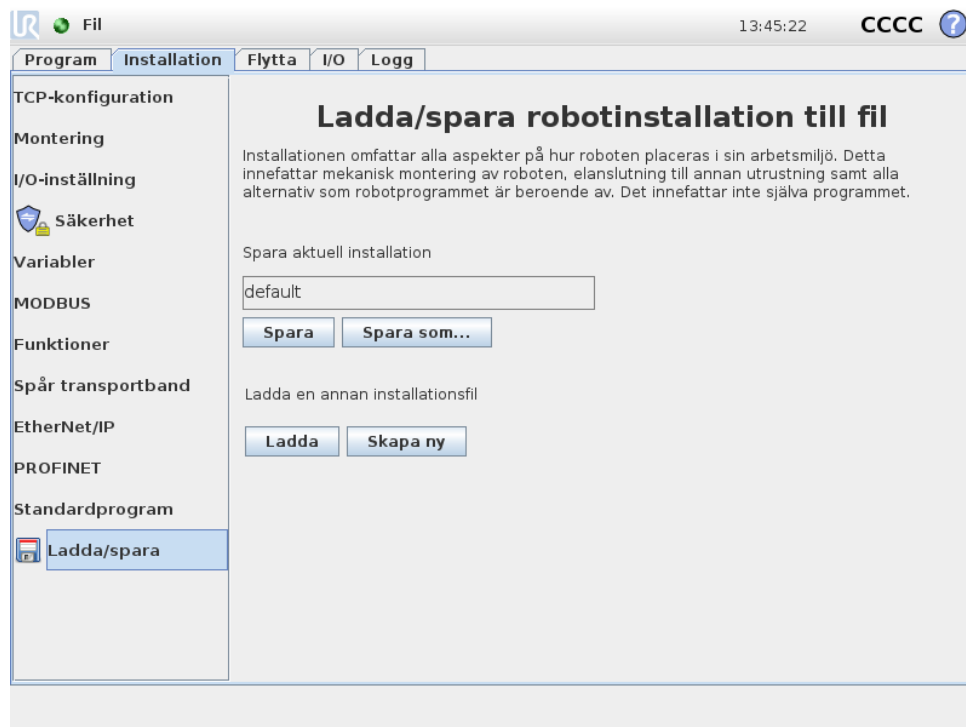
Auto

Håll ned knappen **Auto** för att flytta robotarmen enligt animationen.
Obs: Du kan när som helst släppa knappen för att stoppa rörelsen.

Manuell

Om du trycker på knappen **Manuell** kommer du till fliken **Flytta** där robotarmen kan flyttas manuellt. Detta behövs endast om rörelsen i animationen är olämplig.

13.5 Installation → Ladda/spara



Installationen omfattar alla aspekter på hur robotarmen och manöverskåpet placeras i arbetsmiljön. Detta innefattar mekanisk montering av robotarmen, elanslutning till annan utrustning samt alla alternativ som robotprogrammet är beroende av. Det innefattar inte själva programmet.

Dessa inställningar kan göras med de olika skärmarna under fliken **Installation**, förutom in- och utgångar, som ställs in i fliken **I/O** (se 13.2).

Det går att ha mer än en installationsfil för roboten. Program som skapas använder den aktiva installationen och installationen laddas automatiskt när den används.

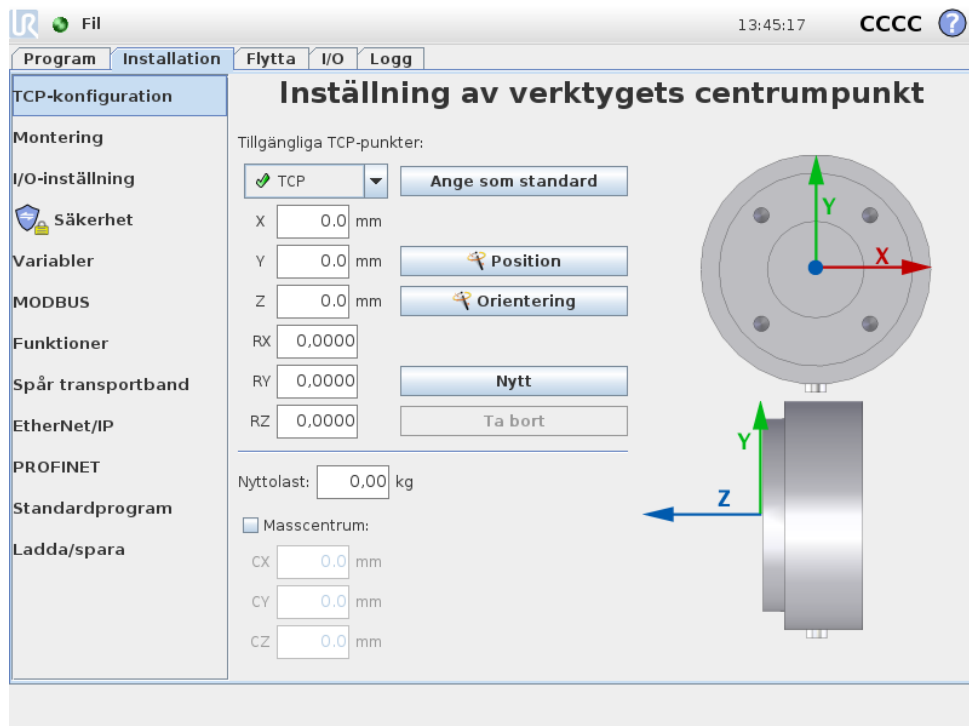
Alla förändringar av en installation måste sparas för att kunna behållas efter nedstängning. Om det finns ändringar i installationen som inte sparats, visas en diskettikon bredvid texten **Ladda/spara** på den vänstra sidan av fliken **Installation**.

En installation kan sparas genom att du trycker på knappen **Spara** eller **Spara som...** Om du sparar ett program sparas även den aktiva installationen. Om du vill ladda in en annan installationsfil använder du knappen **Ladda**. Knappen **Skapa ny** återställer alla inställningar i robotinstallationen till fabriksinställningarna.

**FÖRSIKTIGT:**

Vi rekommenderar inte att använda roboten med en installation som laddats från en USB-enhet. Om du vill använda en installation som lagrats på en USB-enhet laddar du den först och sparar den sedan i den lokala **program**-mappen med knappen **Spara som...**

13.6 Installation → TCP-konfiguration



Verktygets centrumpunkt (TCP) är en karakteristisk punkt på robotens verktyg. Flera namngivna TCP kan definieras på den här skärmen. Varje TCP innehåller en förflyttning och en rotation i förhållande till verktygets utgångsfläns, enligt skärmbilden. Positionskoordinaterna, X, Y och Z, anger positionen för TCP, medan RX, RY och RZ anger dess orientering. När alla angivna värden är noll, sammanfaller TCP med mittpunkten på verktygets utgångsfläns, varvid det koordinatsystem som avbildas till höger på skärmen används.

13.6.1 Lägga till, ändra och ta bort TCP

Du definierar en ny TCP genom att trycka på knappen **Ny**. Den TCP som skapas får automatiskt ett unikt namn och markeras i kombinationsrutan.

Förflyttning och rotation för markerad TCP kan ändras genom att trycka på respektive vita textfält och skriva in nya värden.

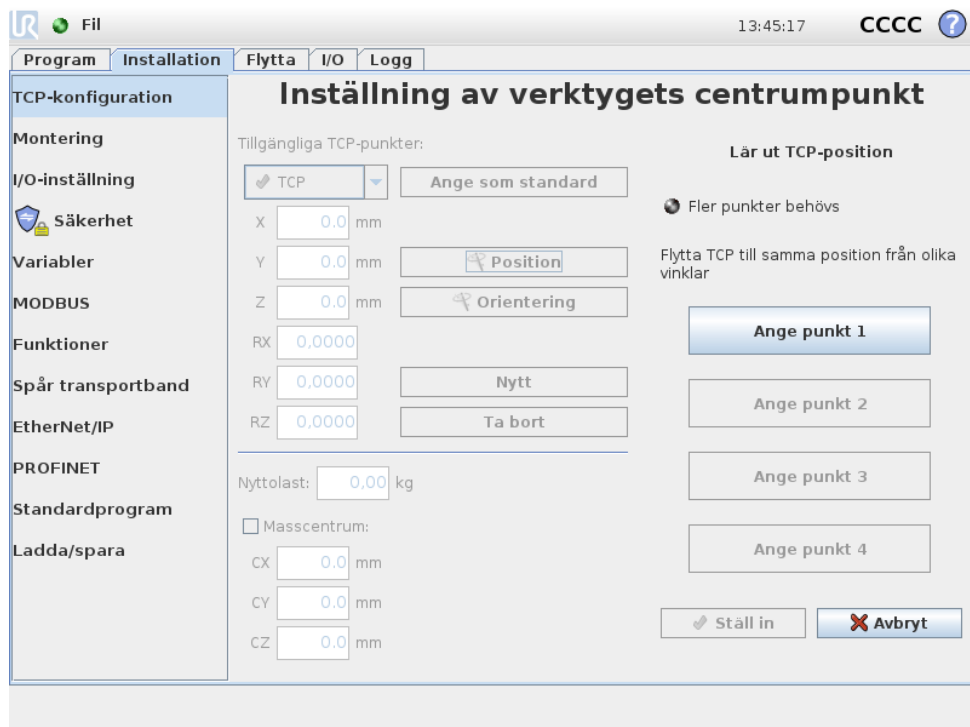
Du tar bort markerad TCP genom att trycka på knappen **Ta bort**. Den sista återstående TCP går inte att ta bort.

13.6.2 Standard-TCP och aktiv TCP

Exakt en av alla konfigurerade TCP är *standard*-TCP. Standard-TCP markeras med en grön ikon till vänster om namnet i kombinationsrutan TCP. Om du vill ange den TCP som är vald till standard-TCP trycker du på knappen **Ange som standard**.

En TCP-förskjutning används alltid som *aktiv* för att avgöra alla linjära rörelser i kartesisk rymd. Det är också den aktiva TCP-punktens rörelse som visas på grafikfiken (se 14.29). Innan något program körs och före programstart anges standard-TCP som aktiv TCP. Inom ett program kan vilken som helst av de aktiva TCP anges som aktiv TCP för en speciell robotrörelse (se 14.5 och 14.10).

13.6.3 Lära ut TCP-position



TCP-positionskoordinater kan beräknas automatiskt enligt följande:

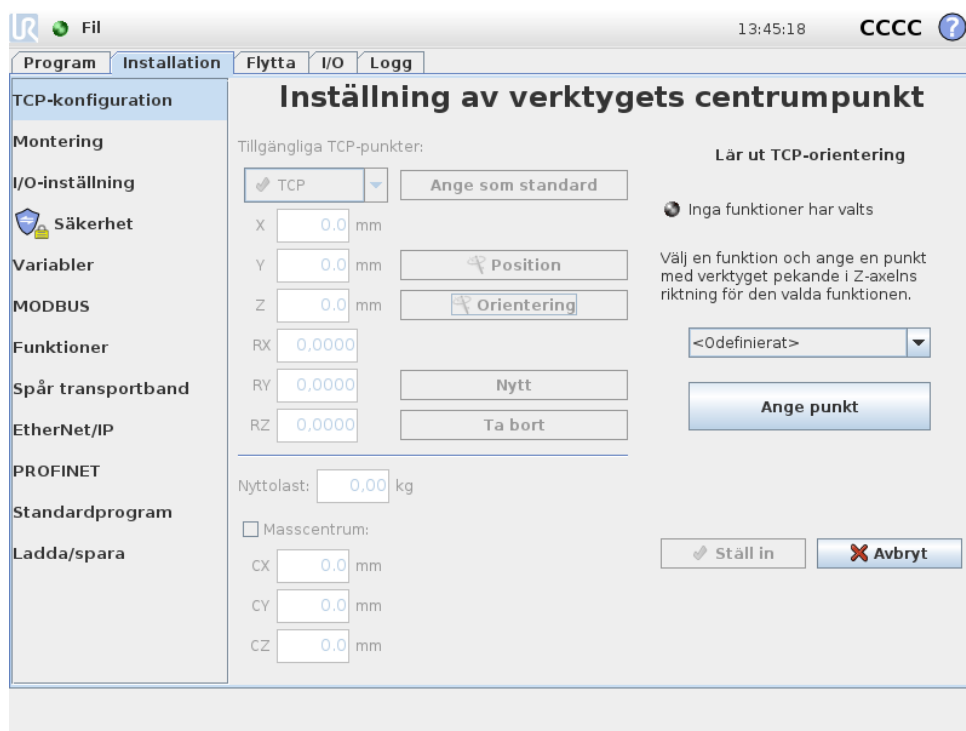
1. Tryck på knappen  **Position**.
2. Välj en fixpunkt i robotens arbetsutrymme.
3. Använd knapparna till höger på skärmen för att flytta TCP till valt punkt från minst tre olika vinklar och för att spara motsvarande positioner för verktygets utgångsflänsen.
4. Verifiera de beräknade TCP-koordinaterna och ange dem för vald TCP med knappen **Ställ in**.

Observera att positionerna måste vara tillräckligt olika för att beräkningen ska bli korrekt. Annars slår LED-statuslampan ovanför knapparna om till rött.

Dessutom gäller att även om tre positioner vanligtvis räcker för att avgöra korrekt TCP kan den fjärde positionen användas för att ytterligare verifiera att beräkningen

är korrekt. Kvaliteten för varje sparad punkt jämfört med beräknad TCP anges med grön, gul eller röd LED på respektive knapp.

13.6.4 Lära ut TCP-orientering



TCP-orientering kan beräknas automatiskt enligt följande:

1. Tryck på knappen  **Orientering**.
2. Välj en funktion i kombinationsrutan. Mer information om hur nya funktioner kan definieras finns i (se 13.12).
3. Använd knappen nedan för att flytta till en position där orienteringen för det verktyg som motsvarar TCP sammanfaller med koordinatsystemet för vald funktion.
4. Verifiera den beräknade TCP-orienteringen och ange den för vald TCP med knappen **Ställ in**.

13.6.5 Nyttolast

Vikten för robotens verktyg anges i skärmens nedre del. Du ändrar inställningen genom att trycka i det vita textfältet och skriva in en ny vikt. Inställningen gäller för alla definierade TCP. Information om den högsta tillåtna nyttolasten finns i installationshandboken för maskinvaran.

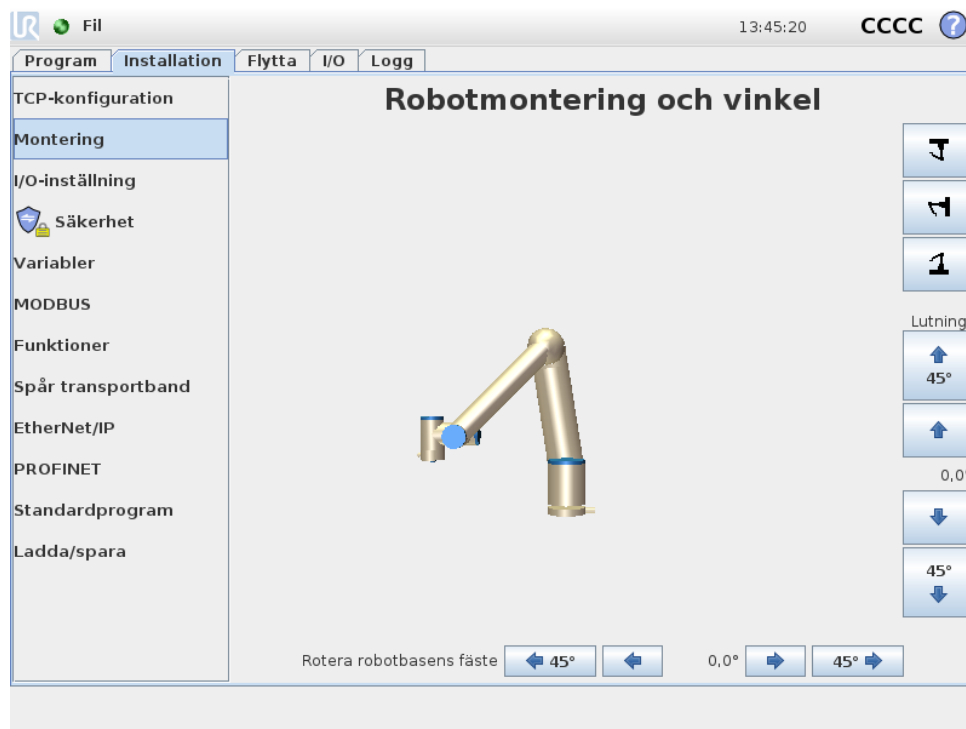
13.6.6 Masscentrum

Det går även att ange verktygets masscentrum (valfritt) med fälten CX, CY och CZ. Om inget annat anges antas verktygets mittpunkt sammanfalla med verktygets masscentrum. Inställningen gäller för alla definierade TCP.


VARNING:

Se till att använda rätt installationsinställningar. Spara och ladda installationsfilerna tillsammans med programmet.

13.7 Installation → Montering



Här kan robotarmens montering specificeras. Detta har två syften:

1. Att få robotarmen att se rätt ut på skärmen.
2. Att informera manöverenheten om gravitationsriktningen.

Manöverenheten använder en avancerad dynamikmodell för att ge robotarmen jämna och exakta rörelser, och för att få robotarmen att hålla emot i läget **Frikörning**. Därför är det viktigt att robotarmens montering ställts in korrekt.


VARNING:

Om robotarmens montering inte ställs in korrekt kan detta orsaka ofta återkommande skyddsstopp och en möjlighet att robotarmen rör sig när knappen **Frikörning** trycks ned.

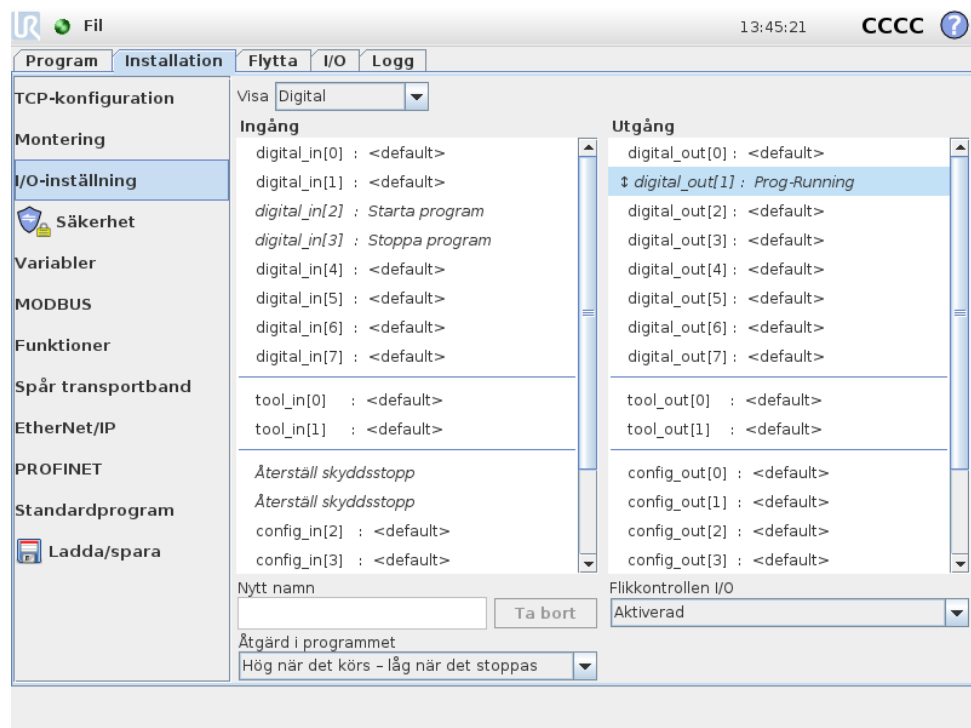
Robotarmen monteras vanligtvis på ett plant bord eller på golvet, och i sådana fall behöver inga ändringar göras på den här skärmbilden. Om robotarmen däremot **takmonteras**, **väggmonteras** eller **monteras i vinkel** måste detta justeras med tryckknapparna. Knapparna till höger på skärmen används för att ställa in vinkeln på

robotarmens montering. De tre knapparna på övre högra sidan ställer in vinkeln mot **taket** (180°), **väggen** (90°) och **golvet** (0°). **Luta**-knapparna kan användas för att ställa in en godtycklig vinkel. Knapparna på nedre delen av skärmen används för att rotera robotarmens fäste så att det stämmer överens med den faktiska monteringen.

**WARNING:**

Se till att använda rätt installationsinställningar. Spara och ladda installationsfilerna tillsammans med programmet.

13.8 Installation → I/O-inställning



På den här skärmen kan -signaler som är tillgängliga för roboten tilldelas användardefinierade namn samt konfigureras med åtgärder och flikkontrollen I/O.

Avsnitten **Ingång** och **Utgång** listar olika typer av I/O-signaler som:

Digital: (standard för allmänna ändamål, konfigurerbar och verktyg)

Analog: (standard för allmänna ändamål och verktyg)

MODBUS:

Allmänna: register (boolsk, heltal och flyttal)

De allmänna registeren kan nås ex. via en fältbuss (som Profinet och Ethernet/IP).

13.8.1 I/O-signaltyp

För att begränsa antalet signaler som listas i avsnitten **Ingång** och **Utgång**, använd kombinationsrutan **Visa** högst upp på skärmen för att ändra det visade innehållet baserat på signaltyp.

13.8.2 Använda användaredefinierade namn

Det är möjligt att associera namn till in- och utsignaler. Det gör det lättare att komma ihåg vad signalerna gör vid arbete med roboten. Välj en signal genom att klicka på den och klicka på textfältet i den nedre delen av skärmen för att ange namnet. Namnet kan återställas genom att radera det (använd knappen **Ta bort**).

För att göra ett allmänt register tillgängligt för programmet (t.ex. för kommandot **Vänta** eller ett villkorligt uttryck som ett **Om**-kommando), så måste registret ges ett användaredefinierat namn. Kommandona **Vänta** och **Om** beskrivs i (14.9) och (14.18) respektive. De namngivna allmänna registren finns i väljaren **Ingång** och **Utgång** på skärmbilden **Uttrycksredigerare**.

13.8.3 I/O-åtgärder och I/O-flikkontroll

Åtgärder för ingångar: De åtta allmänna standardingångarna och de två digitala verktygsingångarna såväl som de allmänna ingångsregistren av typen *boolsk* kan utlösa en åtgärd. En stigande kant kan leda till att:

- Det aktuella programmet startas
- Det aktuella programmet stoppas
- Det aktuella programmet pausas

Dessutom kan en åtgärd konfigureras så att den aktiverar eller desaktiverar läget **Frikörning** när ingången är hög/låg (som knappen **Frikörning** på baksidan av manöverenheten)).

Utgångsåtgärder och I/O-flikkontroll: Standarduppträdandet för utgångar är att deras värden bevaras efter att ett program har slutförts. Det går även att konfigurera en utgång med ett standardvärde som används så fort inget program körs.

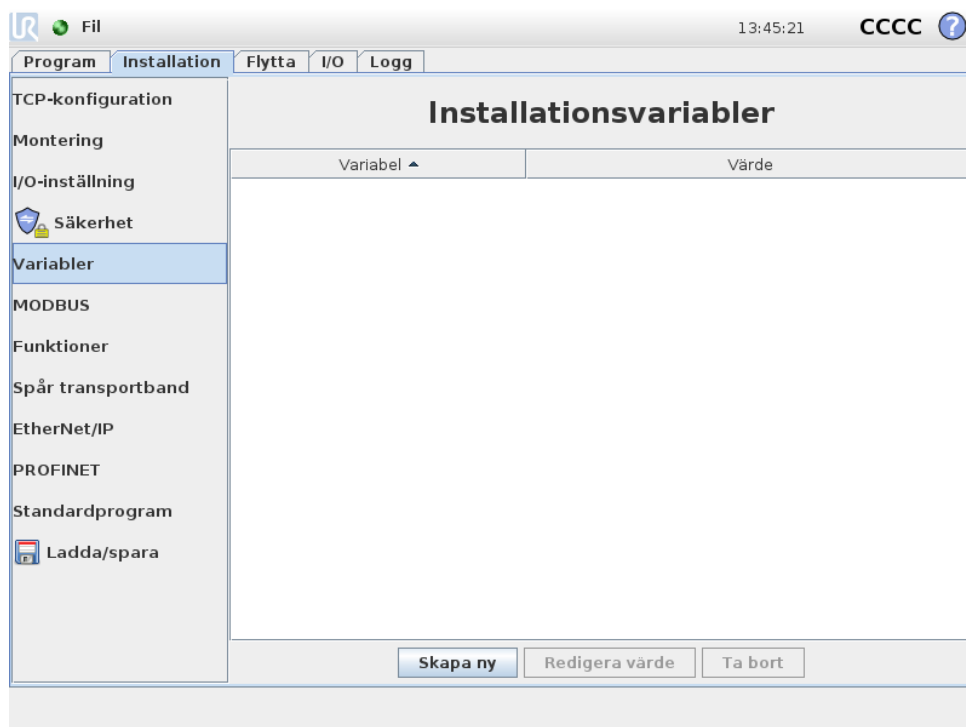
De åtta allmänna digitala standardutgångarna och de två verktygsutgångarna går dessutom att konfigurera så att de avspeglar om ett program körs, så att utgången är hög när ett program körs och annars låg. Även allmänna utgångsregister av typen *boolsk* och digitala MODBUS utgångssignaler stöder också det här.

Det går också slutligen att ange om en utgång kan styras via I/O-fliken (av programmerare eller av både operatörer och programmerare) eller om endast robotprogram kan ändra utgångsvärdet.

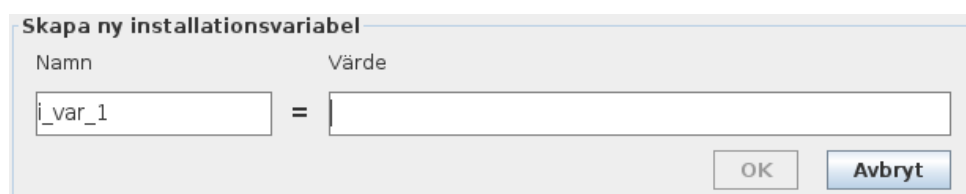
13.9 Installation → Säkerhet

Se kapitel 10.

13.10 Installation → Variabler



Variabler som skapas här kallas installationsvariabler och kan användas precis som normala programvariabler. Installationsvariabler är speciella, eftersom de behåller sina värden även om programmet stoppas och sedan startas om, och även om robotarmen och/eller manöverskåpet stängs av och sedan slås på igen. Deras namn och värden lagras med installationen, så det går att använda samma variabel i flera program.



Om du trycker på **Skapa ny** öppnas ett fönster med ett namnförslag för den nya variabeln. Namnet går att ändra och dess värde kan anges genom att röra vid något textfält. Det går bara att klicka på knappen **OK** om det nya namnet inte redan används i installationen.

Det går att ändra värdet på en installationsvariabel genom att markera den i listan och klicka på **Redigera värde**.

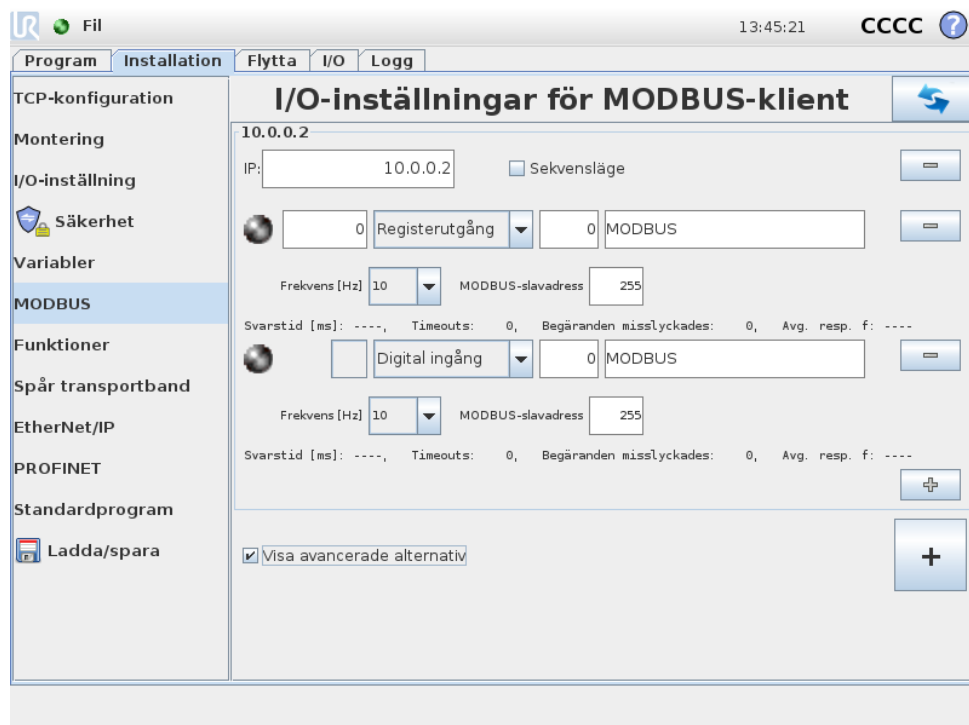
Du tar bort en variabel genom att markera den i listan och klicka på **Ta bort**.

När installationsvariablerna har konfigurerats måste själva installationen sparas för att konfigurationen ska behållas, (se 13.5). Installationsvariablerna och deras värden sparas också automatiskt var tionde minut.

Om ett program eller en installation laddats och en eller flera av programvariablerna har samma namn som installationsvariablerna får användaren två möjligheter

att lösa problemet: antingen används installationsvariabeln istället för programvariabeln, eller också får de variabler som krockar nya namn automatiskt.

13.11 Installation → I/O-inställningar för MODBUS-klient



Här ställs MODBUS-klientens (huvudenhetens) signaler in. Anslutningar till MODBUS-servrar (eller slavenheter) på angivna IP-adresser kan skapas med in- och utgångssignaler (register eller digitalt). Varje signal har ett unikt namn så att den kan användas i program.

Uppdatera

Tryck på den här knappen för att uppdatera alla MODBUS-anslutningar. Uppdatering kopplar ifrån och återansluter alla modbus-enheter. All statistik rensas.

Lägg till enhet

Tryck på den här knappen för att lägga till en ny MODBUS-enhet.

Ta bort enhet

Tryck på den här knappen för att ta bort MODBUS-enheten och alla signaler på den.

Ange enhetens IP-adress

Här visas MODBUS-enhetens IP-adress. Tryck på knappen för att ändra den.

Sekvensläge

Endast tillgängligt när Visa avancerade alternativ (se 13.11) är valt. Om den här kryssrutan markeras tvingas modbus-klienten vänta på ett svar innan nästa begäran skickas. Vissa fältbus-enheter kräver det här läget. Det kan hjälpa att aktivera det här alternativet om det finns flera signaler och ett större antal begäranden resulterar i signalförluster. Observera att den faktiska signalfrekvensen kan vara lägre än den begärda om flera signaler definieras i sekvensläge. Den faktiska signalfrekvensen kan observeras i signalstatistiken (se avsnitt 13.11). Signalindikatorn växlar till gult om den faktiska signalfrekvensen understiger halva det värde som valts i rullgardinsmenyn "Frekvens".

Lägg till signal

Tryck på den här knappen för att lägga till en MODBUS-signal på motsvarande MODBUS-enhet.

Ta bort signal

Tryck på den här knappen för att ta bort en MODBUS-signal från motsvarande MODBUS-enhet.

Ange signaltyp

Använd den här kombinationsrutan för att ange signaltypen. Tillgängliga typer är:

Digital ingång: En digital ingång (spole) är en kvantitet på en bit som läses från MODBUS-enheten på den spole som anges i signalens adressfält. Funktionskod 0x02 (Läs diskreta ingångar) används.

Digital utgång: En digital utgång (spole) är en kvantitet på en bit som kan anges till antingen hög eller låg. Innan värdet på den här utgången har angetts av användaren läses värdet från MODBUS-fjärrenheten. Detta betyder att funktionskod 0x01 (Läs spolar) används. När utgången har angetts med ett robotprogram eller genom att trycka på knappen **ange signalvärde** används funktionskoden 0x05 (Skriv enstaka spole) framgent.

Registeringång: En registreringång är en kvantitet på 16 bitar som läses från den adress som anges i adressfältet. Funktionskod 0x04 (Läs ingångsregister) används.

Registerutgång: En registerutgång är en kvantitet på 16 bitar som kan anges av användaren. Innan värdet på den här utgången har angetts, läses värdet från MODBUS-fjärrenheten. Detta betyder att funktionskod 0x03 (Läs hållregister) används. När utgången har angetts med ett robotprogram eller genom att ett signalvärde angetts i fältet **ange signalvärde**, används funktionskoden 0x06 (Skriv enstaka register) för att ange värdet på MODBUS-fjärrenheten.

Ange signaladress

Detta fält visar adresserna på MODBUS-fjärrservern. Använd knappsetsen på skärmen för att välja en annan adress. Giltiga adresser beror på MODBUS-fjärrenhetens tillverkare och konfiguration.

Ange signalnamn

Användaren kan ge signalen ett namn med tangentbordet på skärmen. Det här namnet används när signalen används i program.

Signalvärde

Här visas signalens aktuella värde. För registersignaler uttrycks värdet som ett heltal utan tecken. För utgångssignaler kan önskat signalvärde anges med knappen. Även för registerutgångar uttrycks det värde som ska skrivas till enheten som ett heltal utan tecken.

Signalanslutningsstatus

Denna ikon visar om signalen kan läsas och skrivas korrekt (grön), eller om enheten ger oväntade svar eller inte går att nå (grå). Om ett MODBUS-undantag tas emot visas svars-koden. Svar på MODBUS-TCP-fel är:

- E1: OGILTIG FUNKTION (0x01)* Den funktionskod som togs emot i frågan är inte en tillåten åtgärd för servern (eller slavenheten).
- E2: OGILTIG DATAADDRESS (0x02)* Den funktionskod som togs emot i frågan är inte en tillåten åtgärd för servern (eller slavenheten). Kontrollera att den signaladress som angavs svarar mot inställningen på MODBUS-fjärrservern.
- E3: OGILTIGT DATAVÄRDE (0x03)* Ett värde som finns i frågans datafält är inte ett tillåtet värde för servern (eller slavenheten). Kontrollera att det signalvärde som angavs är giltigt för den angivna adressen på MODBUS-fjärrservern.
- E4: SLAVENHETSFE (0x04)* Ett oåterkalleligt fel inträffade medan servern (eller slavenheten) försökte utföra den begärda åtgärden.
- E5: KVITTERA (0x05)* Speciell användning i samband med programmeringskommandon som skickas till MODBUS-fjärrenheten.
- E6: SLAVENHET UPPTAGEN (0x06)* Speciell användning i samband med programmeringskommandon som skickas till MODBUS-fjärrenheten. Slavenheten (servern) kan inte svara för tillfället.

Visa avancerade alternativ

Den här kryssrutan visar eller döljer de avancerade alternativen för varje signal.

Avancerade alternativ

Uppdateringsfrekvens: Den här menyn kan användas för att ändra signalens uppdateringsfrekvens. Detta är den frekvens med vilken förfrågningar skickas till MODBUS-fjärrenheten för att läsa eller skriva signalvärdet. Om frekvensen är inställd på 0, initieras modbus-begäranden på begäran med användning av skriptfunktionerna *modbus_get_signal_status*, *modbus_set_output_register*, och *modbus_set_output_signal*.

Slavadress: Detta textfält kan användas för att ange en speciell slavadress för de förfrågningar som motsvarar en specifik signal. Värdet måste vara i det öppna

intervallet 0-255, och standardvärdet är 255. Om du ändrar detta värde rekommenderar vi att du kontrollerar funktionen vid ändring av slavadress i dokumentationen till MODBUS-fjärrenheten.

Återanslut räkning: Antalet gånger TCP-anslutningen stängdes och återanslöt.

Anslutningsstatus: TCP-anslutningsstatus.

Svarstid [ms:] Tiden mellan att modbus-begäran skickas och svaret tas emot – uppdateras endast när kommunikation är aktiv.

Fel på modbuspaket: Antal mottagna paket som innehöll fel (d.v.s. ogiltig längd, felaktig data, TCP-uttagsfel).

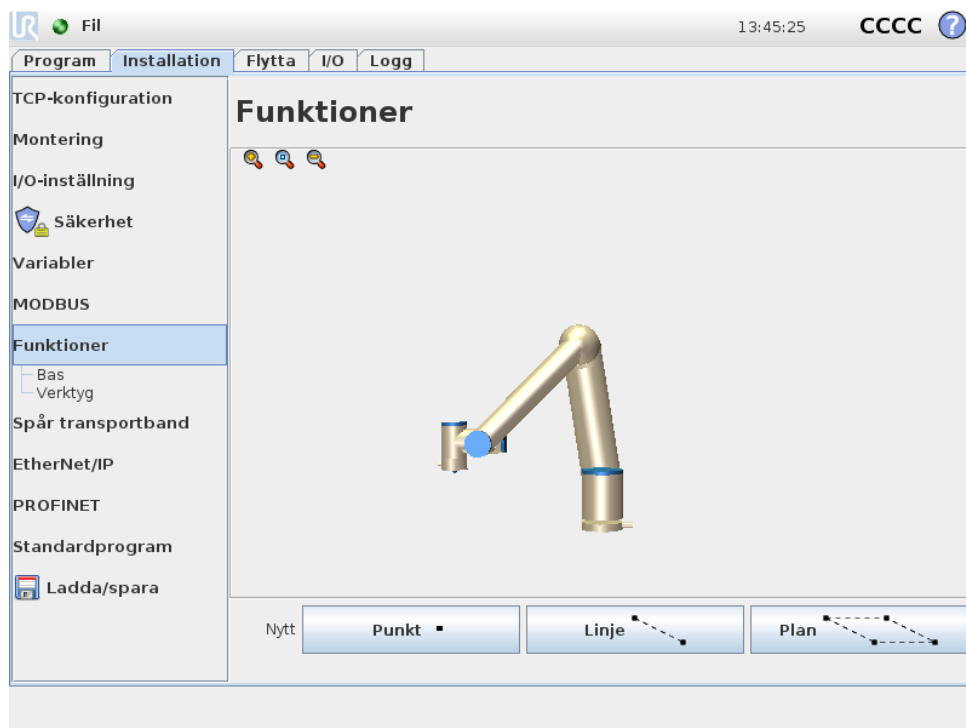
Timeouts: Antal modbus-begäranden som inte besvarats.

Begäranden misslyckades: Antal paket som inte kunde skickas p.g.a. ogiltig uttagsstatus.

Faktisk frekvo.: Genomsnittlig frekvens av klientens (huvudenhetens) signalstatusuppdateringar. Det här värdet beräknas om varje gång signalen tar emot ett svar från servern (eller en slav).

Alla räknare upp till 65535 och sedan omsluta tillbaka till 0.

13.12 Installation → Funktioner



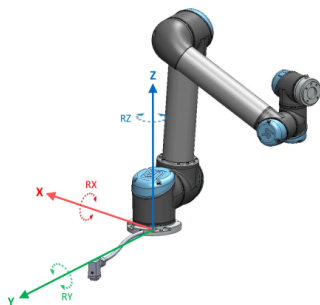
Det är vanligt att delar av ett robotprogram består av rörelser som måste utföras i förhållande till specifika föremål annat än robotarmens bas. Föremålen kan vara bord, andra maskiner, arbetsstycken, transportband, lastpallar, system för visuell avkänning, tomrum eller gränser som vanligen finns i robotarmens omgivning.

En **Funktion** är i grunden en representation av ett sådant föremål som har definierats med ett namn för framtida referens och en sexdimensionell pose (position och orientering) i förhållande till robotens fundament.

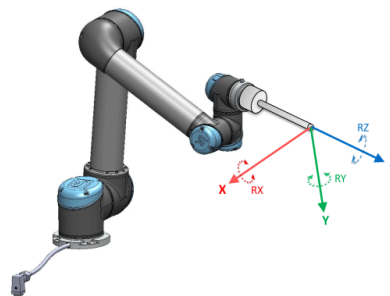
Det finns alltid två fördefinierade funktioner för roboten, var och en har sin pose definierad utifrån själva robotarmen:

Funktionen: Bas placerad med origo i robotbasens centrum (se figur 13.1)

Funktionen: Verktyg placerad med utgångspunkt i centrum av aktuell TCP (se figur 13.2)



Figur 13.1: Basfunktion



Figur 13.2: Verktyg (TCP-funktion)

Användardefinierade funktioner får sin pose definierad av användaren genom en unik och intuitiv metod som använder aktuell verktygpose (TCP) för att definiera platser i arbetsområdet. Det betyder att användaren kan lära in platser för funktioner med hjälp av ex. läget **Frikörning** eller knuffa (manuell styrning) till önskad pose.

Det finns tre olika strategier (**Punkt**, **Linje** och **Plan**) för att definiera funktionens pose. Den bästa strategin för en given tillämpning beror på typen av föremål och precisionskraven men allmänt sett är en funktion som bygger på fler punkter (**Linje** och **Plan**) att föredra om den är tillämplig för det specifika föremålet.

För att t.ex. definiera riktningen för ett linjärt transportband, är det möjligt att enbart använda funktionen **Punkt** men allmänt sett blir den linjära rörelsen noggrannare och bättre om den representeras av funktionen **Linje**. Vid användning av funktionen **Punkt** för att definiera transportbandets riktning är det nödvändigt för TCP att peka i samma riktning som transportbandet. Bättre noggrannhet uppnås med en definition av två punkter med funktionen **Linje** med så stor fysisk separation som möjligt.

En användning av fler punkter för att definiera t.ex. ett bords pose innebär att orienteringen är baserad på positionerna snarare än orienteringen hos en enda TCP, och en enda TCP-orientering är i allmänhet svårare att konfigurera med hög precision. En förklaring till hur man tillämpar olika metoder för att definiera en funktion finns i (avsnitten: 13.12.2), (13.12.3) och (13.12.4).

13.12.1 Använda en funktion

När en funktion har definierats vid installationen, är det möjligt att hänvisa till den från robotprogrammet, att relatera robotens rörelser (kommandona **FlyttaL** och

FlyttaP) till funktionen (se avsnitt 14.5). Detta medger en enkel anpassning av robotprogrammet, t.ex. vid flera olika robotstationer eller när ett föremål flyttas dynamiskt under programmets körning eller då det flyttas permanent runt scenen. Genom en enkel justering av den refererade funktionen för ett visst föremål sker alla programmets rörelser i förhållande till föremålet. Exempel på det här finns i avsnitten (avsnitten 13.12.5) och (13.12.6).

Funktioner som konfigureras som knuffbara är också användbara när roboten flyttas manuellt under fliken **Flytta** (avsnitt 13.1) eller skärmbilden **Pose-redigering** (se 12.2). När en funktion väljs som en referensram, kommer knapparna **Flytta verktyg** för förflyttningar och rotationer att fungera i den valda funktionsrymden (se 13.1.2) och (13.1.3), vilket även de aktuella TCP-koordinatvärdena gör som visas i TCP-delen av skärmen. Om t.ex. ett bord har definierats som en funktion och det har valts som referens i fliken **Flytta** kommer pilknapparna för förflyttningar upp/ner, vänster/höger och fram/bak, samt de visade TCP-koordinaterna också följa bordets ram.

Nytt namn

Med den här knappen går det att byta namn på en funktion.

Ta bort

Den här knappen tar bort den markerade funktionen och eventuella underfunktioner.

Visa axlar

Välj om den valda funktionens koordinataxlar ska synas i 3D-grafiken. Valet gäller den här skärmen och skärmen **Flytta**.

Kan knuffas

Välj om det ska gå att knuffa den valda funktionen. Detta bestämmer om funktionen ska visas i funktionsmenyn på skärmen **Flytta**.

Variabel

Välj om den valda funktionen kan användas som en variabel. Om detta alternativ är aktivt, kommer en variabel med samma namn som funktionen, avslutat av "_var", att vara tillgänglig vid redigering av robotprogram. Variabeln kan tilldelas ett nytt värde i ett program, vilket senare kan användas för att styra vägpunkter som beror av en funktions värde. En funktion måste också markeras som en variabel för att varje förflyttning som beror av den ska uppdateras när funktionen ändras. Om den inte markeras som en variabel kommer vägpunkten att vara relativ till posen när den definieras.

Ändra den här punkten

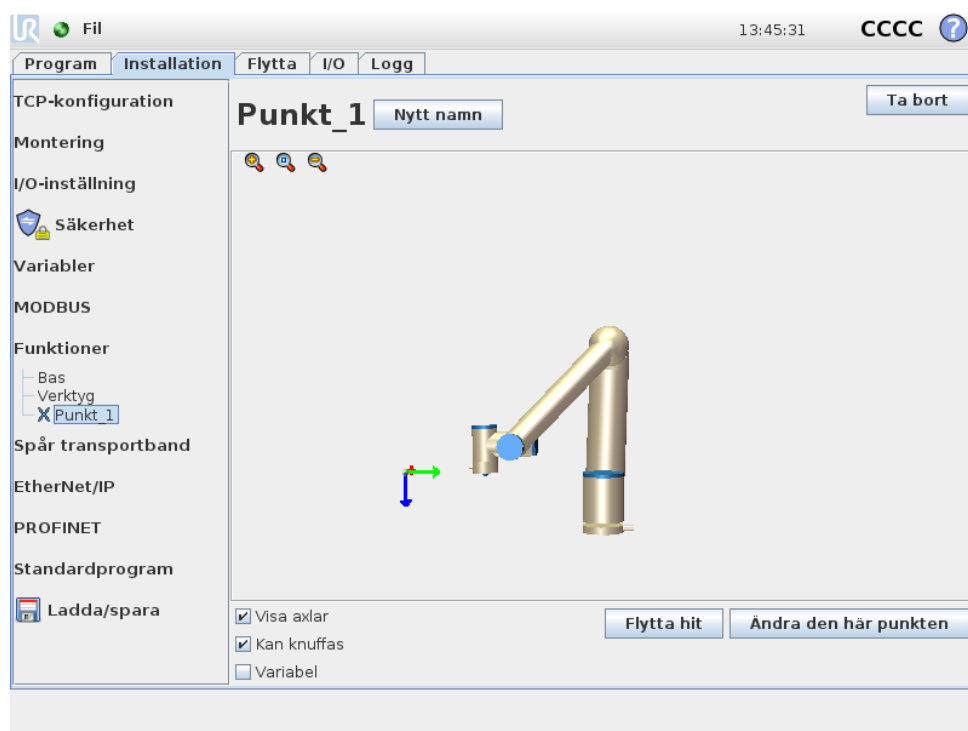
Använd den här knappen för att ändra den valda funktionen. Fliken **Flytta** (avsnitt 13.1) visas och en ny position kan anges för funktionen.

Flytta roboten hit

När den här knappen trycks ned flyttas robotarmen mot vald funktion. I slutet av förflyttningen kommer koordinatsystemen för funktion och TCP att sammanfalla.

13.12.2 Ny punkt

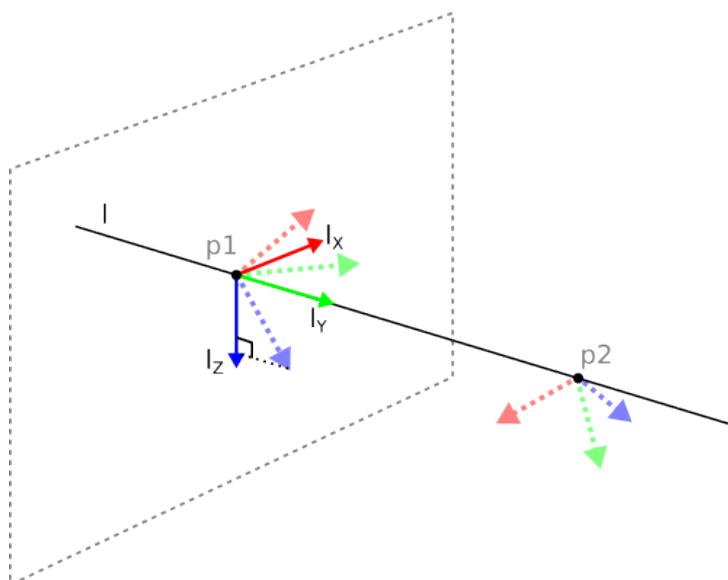
Tryck på den här knappen för att lägga till en punktfunktion till installationen. Punktfunktionen används vanligen vid definition av en säkerhetsgräns eller en global hemkonfiguration för robotarmen. Punktfunktionens pose definieras som position och orientering för TCP för den punkten.



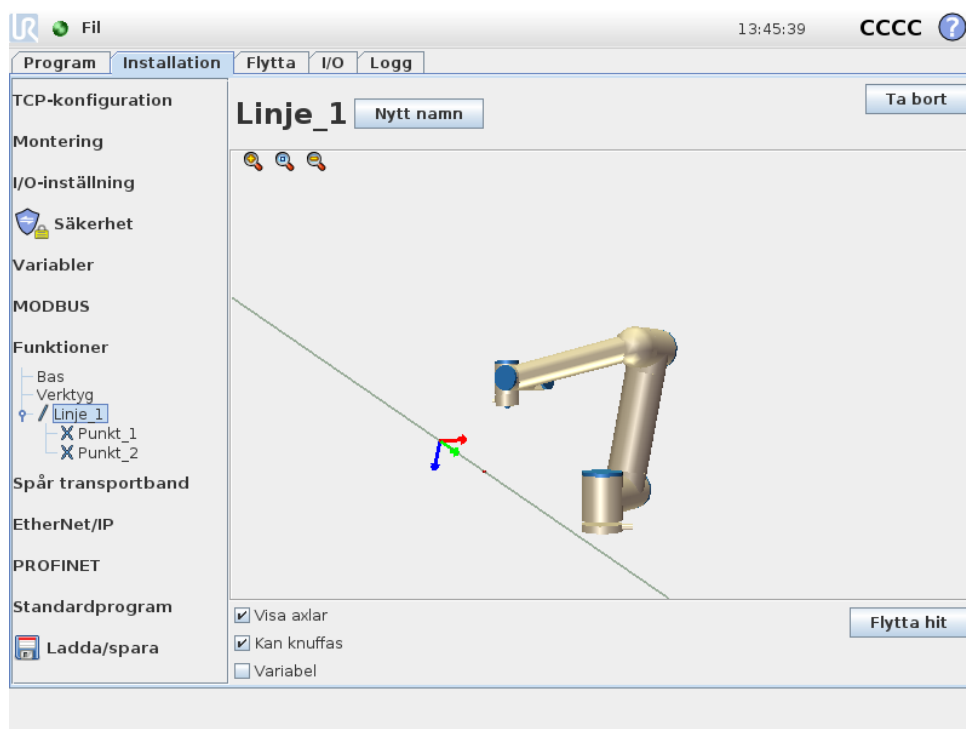
13.12.3 Ny linje

Tryck på den här knappen för att lägga till en punktfunktion till installationen. Linjefunktionen används vanligen för att definiera linjer som roboten behöver följa, t.ex. vid transportbandsspårning. En linje l definieras som en axel mellan två punktfunktioner $p1$ och $p2$ enligt figuren 13.3.

Denna axel, som är riktad från den första punkten till den andra, utgör y-axeln i linjens koordinatsystem. Z-axeln definieras som projektionen av z-axeln för $p1$ på det plan som är vinkelrätt mot linjen. Positionen för linjens koordinatsystem är samma som positionen för $p1$.

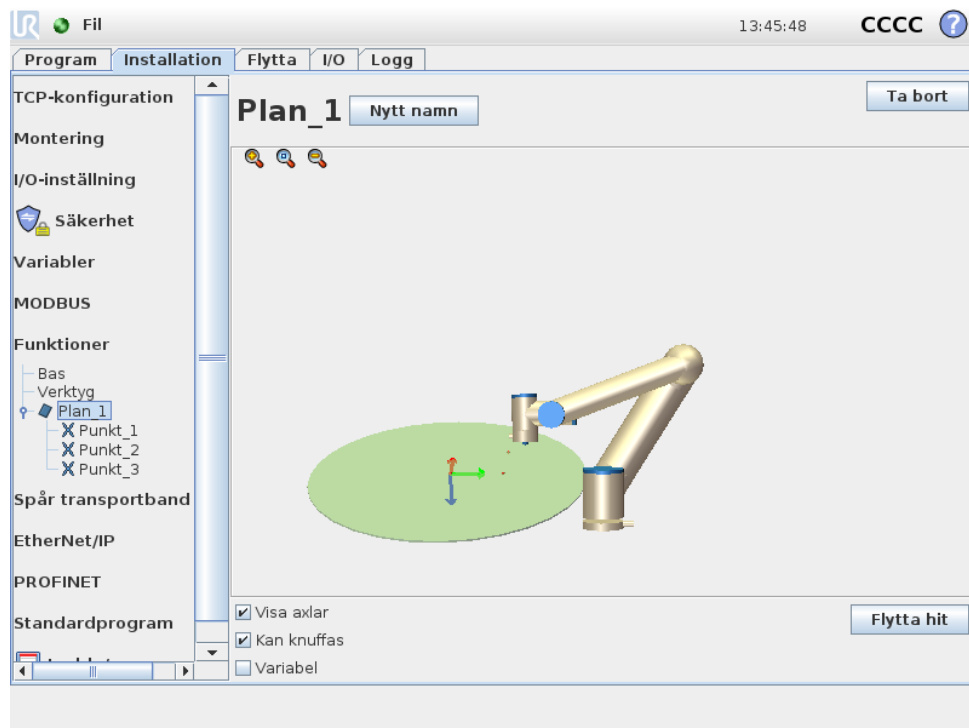


Figur 13.3: Definiera en linjefunktion



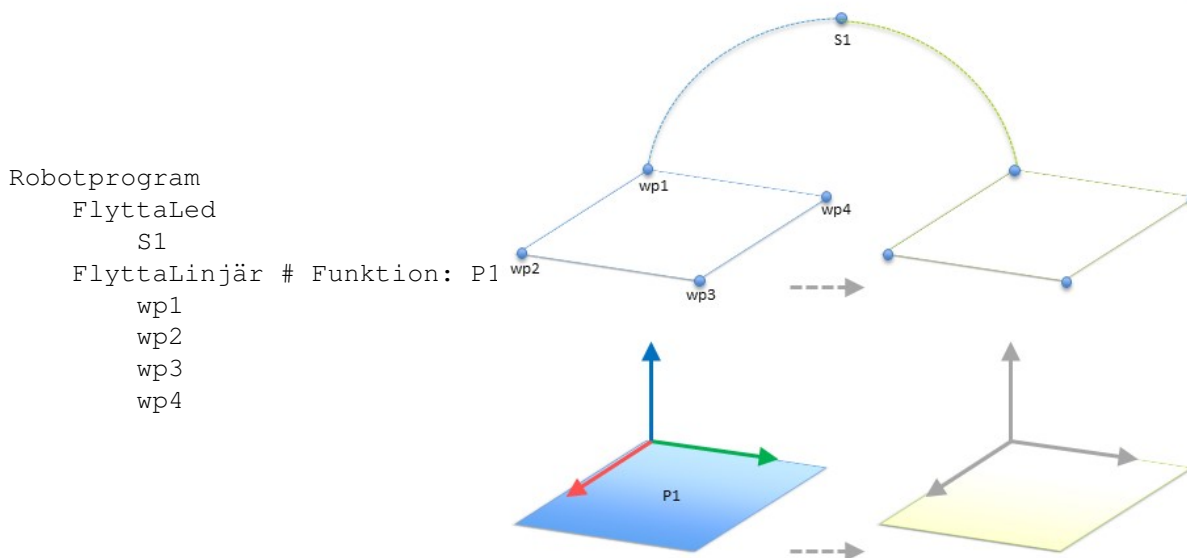
13.12.4 Nytt plan

Tryck på den här knappen för att lägga till en planfunktion till installationen. Planfunktionen används vanligen när det finns behov för en ram med hög precision, t.ex. vid arbete med ett system för visuell avkänning eller vid rörelser i förhållande till ett bord. Ett plan definieras av tre delpunktsfunktioner. Positionen för koordinatsystemet är samma som positionen för den första delpunkten. Z-axeln är planets normal, och y-axeln är riktad från den första punkten mot den andra. Z-axelns positiva riktning är definierad så att vinkeln mellan planets z-axel och den första punktens z-axel är mindre än 180 grader.



13.12.5 Exempel: Manuell uppdatering av en funktion för att justera ett program

Tänk dig en tillämpning där flera delar av ett robotprogram förhåller sig till ett bord. I figuren 13.4 visas detta som rörelsen från vägpunkterna wp1 till wp4.



Figur 13.4: Enkelt program med fyra vägpunkter relativt ett funktionsplan som uppdateras manuellt genom att ändra funktionen

Det är ett krav för tillämpningen att programmet bör återanvändas för flera robotinstallationer, där endast bordets position kan variera något. Rörelsen i förhållande till bordet är identiskt.

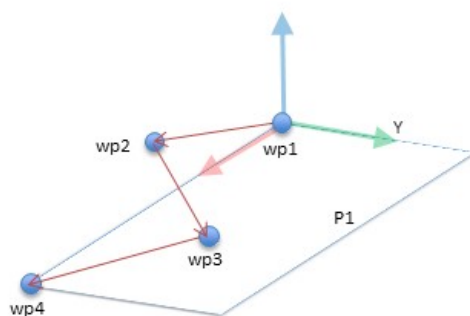
Genom att definiera bordets position som en funktion *P1* i installationen och se till att den är vald som **Variabel**, kan programmet genom att konfigureras i förhållande

till planet med kommandot *FlyttaLinjär* enkelt användas av ytterligare robotar bara genom att uppdatera installationen med bordets aktuella position.

Konceptet gäller valfritt antal funktioner i en tillämpning för att få ett flexibelt program som kan lösa samma uppgift på många robotar trots att t.ex. en plockning, placering eller annan platskritisk uppgift i arbetsområdet kan variera mellan olika anläggningar.

13.12.6 Exempel: Dynamisk uppdatering av en funktions pose

Tänk dig en liknande tillämpning där roboten även måste röra sig i ett specifikt mönster ovanpå ett bord för att kunna lösa en särskild uppgift, enligt figuren 13.5.

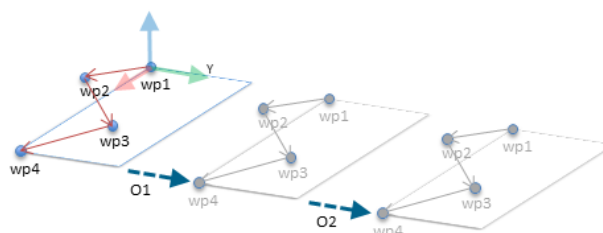


Figur 13.5: Ett *FlyttaV*-kommando med fyra vägpunkter jämfört med planfunktionen

Tänk dig nu att rörelsen i förhållande till *P1* måste upprepas ett antal gånger, varje gång med en förskjutning *o* som i exemplet är inställd till 10 cm i Y-led (se figur 13.6, förskjutningar *O1* och *O2*). Detta kan åstadkommas med t.ex. skriptfunktionerna *pose.ny()* eller *pose.flytta()* för att manipulera den variabla funktionen.

Robotprogram

```
FlyttaLed
  wp1\texttt{(se~\ref{gui:ins
y = 0,01
o = p[0,y,0,0,0,0]
P1_var = pose_flytta(P1_var, o)
FlyttaLinjär # Funktion: P1_var
  wp1
  wp2
  wp3
  wp4
```



Figur 13.6: Använda en förskjutning till planfunktionen

Det är även möjligt att växla till en helt annan funktion medan programmet körs istället för att bara lägga till en förskjutning. Detta kan ses i exemplet nedan (se figur 13.7) där referensfunktionen för kommandot *FlyttaLinjär P1_var* kan växla mellan de två planen *P1* och *P2*.

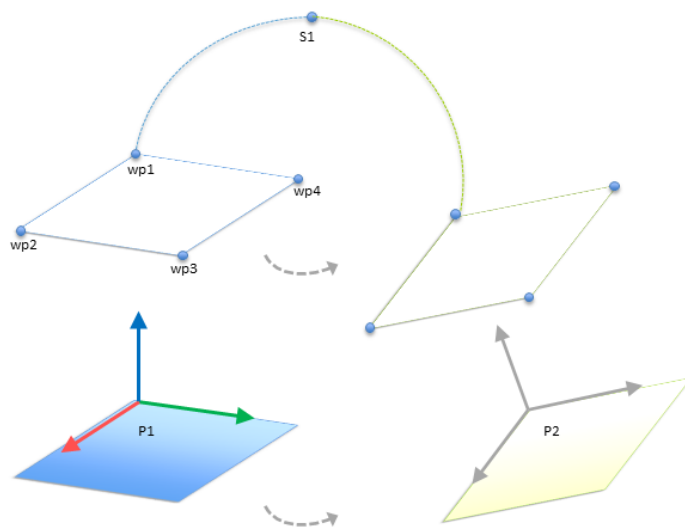
13.13 Inställning av transportbandsspårning

Vid användning av ett transportband kan roboten konfigureras för att spåra dess rörelse. Inställning av transportbandsspårning innehåller alternativ för att konfigu-

```

Robotprogram
  FlyttaLed
    S1
  om (digital_ingång[0]) så
    P1_var = P1
  annars
    P1_var = P2
  FlyttaLinjär # Funktion: P1
    wp1
    wp2
    wp3
    wp4

```



Figur 13.7: Växla mellan planfunktioner

rera roboten för arbete med absoluta och inkrementerande kodenheter plus linjära och cirkulära transportband.

Transportbandsparametrar

Inkrementella: kodenheter kan anslutas till digitala ingångar 0 till 3. Avkodningen av digitala signaler sker vid 40 kHz. Användning av en **Quadrature**-kodenhet (kräver två ingångar) kan roboten fastställa hastigheten tillsammans med transportbandets riktning. Om transportbandets riktning är konstant kan en enda ingång användas. Denna avkänner antingen stigande, fallande eller stigande och fallande i syfte att fastställa transportbandets hastighet.

Absolut: kodenheter kan användas vid anslutning genom en MODBUS-signal. Detta kräver förkonfigurering av ett digitalt MODBUS ingångsregister i (avsnitt 13.11).

Linjära transportband

Om du väljer ett linjärt transportband måste en linjefunktion konfigureras i **funktionsdelen** av installationen för att fastställa transportbandets riktning. Linjefunktionen ska vara parallell med transportbandet, och det ska finnas ett stort avstånd mellan de två punkter som definierar linjefunktionen. Du bör konfigurera linjefunktionen genom att placera verktyget fast mot transportbandets sida vid inläring av två punkter. Om linjefunktionen har motsatt riktning än transportbandets rörelse, tryck på knappen **Omvänd riktning**.

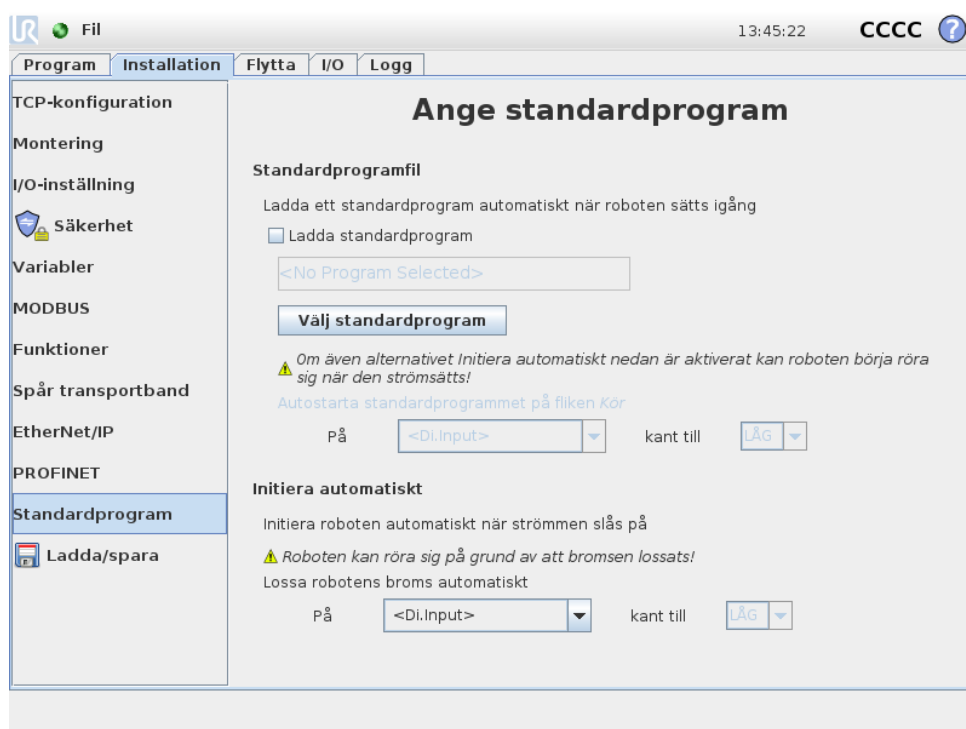
Fältet **Klick per meter** är det antal klick som kodenheten genererar när transportbandet flyttas en meter.

$$\text{Klick per meter} = \frac{\text{klick per varv för kodenheten}}{2\pi \cdot \text{radie för kodenhetens skiva[m]}} \quad (13.1)$$

Cirkelformade transportband

Vid spårning av ett cirkulärt transportband måste (cirkel-) transportbandets cent-rumpunkt vara definierad. Gör detta i installationens del **Funktioner** genom att definiera en punktfunktion. Värdet för **Klick per meter** måste vara det antal klick som kodenheten genererar när transportbandet roteras ett helt varv. Markera kryssrutan **Rotera verktyg med transportband** om verktygets riktning ska ta hänsyn till transportbandet (t.ex. om verktyget står vinkelrätt till transportbandet, kommer den att förbli vinkelrätt under rörelsen). Avmarkera, om orienteringen skal kontrolleras av banan.

13.14 Installation → Standardprogram



Den här skärmen innehåller inställningar för att automatiskt ladda och starta ett standardprogram samt för att initiera robotarmen automatiskt vid start.



WARNING:

När automatisk laddning, start och initiering alla är aktiverade, kommer roboten att köra programmet så snart manöverskåpet startas, om signalen matchar vald signalnivå, dvs. då en kantövergång till den valda signalnivån inte kommer att behövas i detta fall. Var försiktig när signalnivån är inställd till **Låg**, eftersom signalerna är låga som standard i typiska inställningar, vilket leder till att programmet körs automatiskt utan att utlösas av en extern signal.

Ladda standardprogram

Det går att ange att standardprogrammet ska laddas när manöverskåpet startas. Dessutom laddas standardprogrammet även automatiskt när skärmbilden **Kör Program** visas (se 11.4) och inget program har laddats.

Starta standardprogram

Standardprogrammet kan startas automatiskt på skärmen **Kör program**. När standardprogrammet laddats och kantövergången för den angivna externa insignalen upptäcks, startas programmet automatiskt.

Obs: vid start är signalnivån för den aktuella ingången odefinierad. Om en övergång väljs som matchar signalnivån vid start, startas programmet omedelbart. Dessutom inaktiveras funktionen för automatisk start om du lämnar skärmen **Kör program**, eller trycker på stoppknappen på **instrumentbrädan**, tills körningsknappen har tryckts in igen.

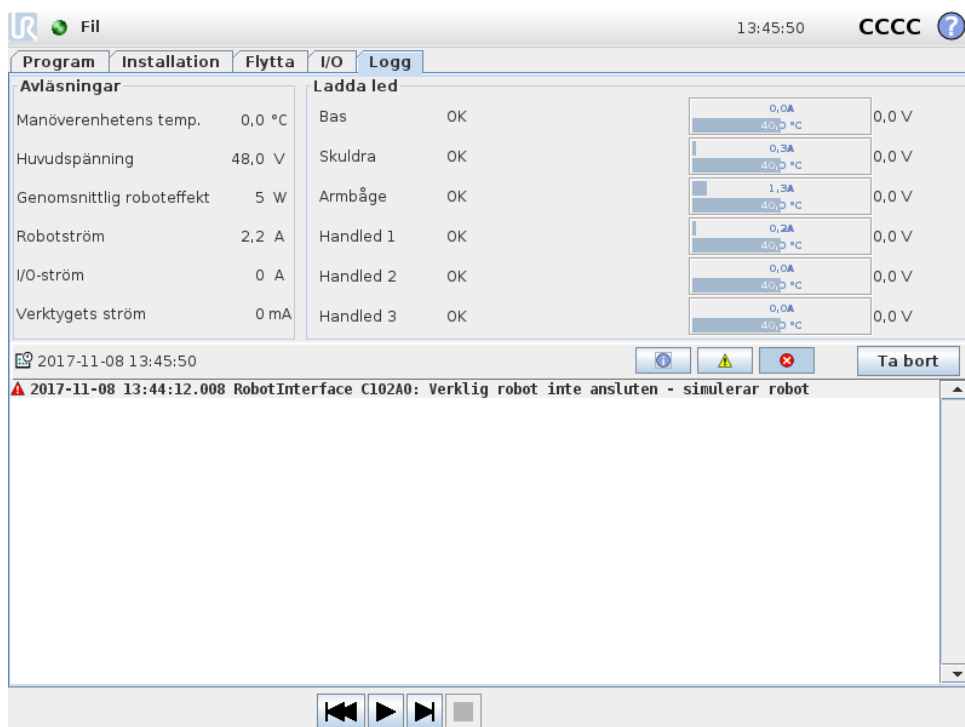
Automatisk initiering

Robotarmen kan initieras automatiskt, t.ex. när manöverskåpet startas. Vid den kantövergången för den angivna externa insignalen kommer robotarmen att initieras helt, oavsett vilken skärm som visas.

Initieringens sista fas är *Lossa broms*. När robotens broms lossa, rör den sig lite och ger ljud ifrån sig. Dessutom går det inte att lossa bromsarna automatiskt om inte den konfigurerade monteringen stämmer med den montering som sensorerna känner av. I så fall måste roboten initieras manuellt på initieringsskärmen (se 11.5).

Obs: Obs: vid start är signalnivån för den aktuella ingången odefinierad. Om en övergång väljs som matchar signalnivån vid start, initieras robotarmen omedelbart.

13.15 Fliken Logg



Robotens tillstånd Skärmens övre halva visar skicket på robotarmen och manöverskåpet.

Den vänstra delen visar information om robotens manöverskåp, och den högra delen visar information om respektive robotled. Varje robotled visar information om temperaturen i motorn och elektroniken, belastningen på leden och spänningen vid leden.

Robotlogg Loggmeddelanden visas på skärmens nedre halva. Den första spalten visar hur allvarig loggposten är. Den andra spalten visar den tidpunkt då meddelandet ankom. Nästa spalt visar meddelandets avsändare. Den sista kolumnen visar själva meddelandet. Det går att filtrera meddelanden genom att välja de växlingsknappar som motsvarar deras allvarlighetsgrad. Figuren ovan visar nu att fel kommer att visas, medan informations- och varningsmeddelanden filtreras. Vissa loggmeddelanden har utformats för att ge ytterligare information. Du kommer åt denna genom att markera loggposten.

13.16 Skärmbilden Ladda

På den här skärmbilden väljer du vilket program som ska laddas. Det finns två versioner av skärmbilden: en som ska användas när du bara vill ladda ett program och köra det, och en som används när du vill redigera ett program.

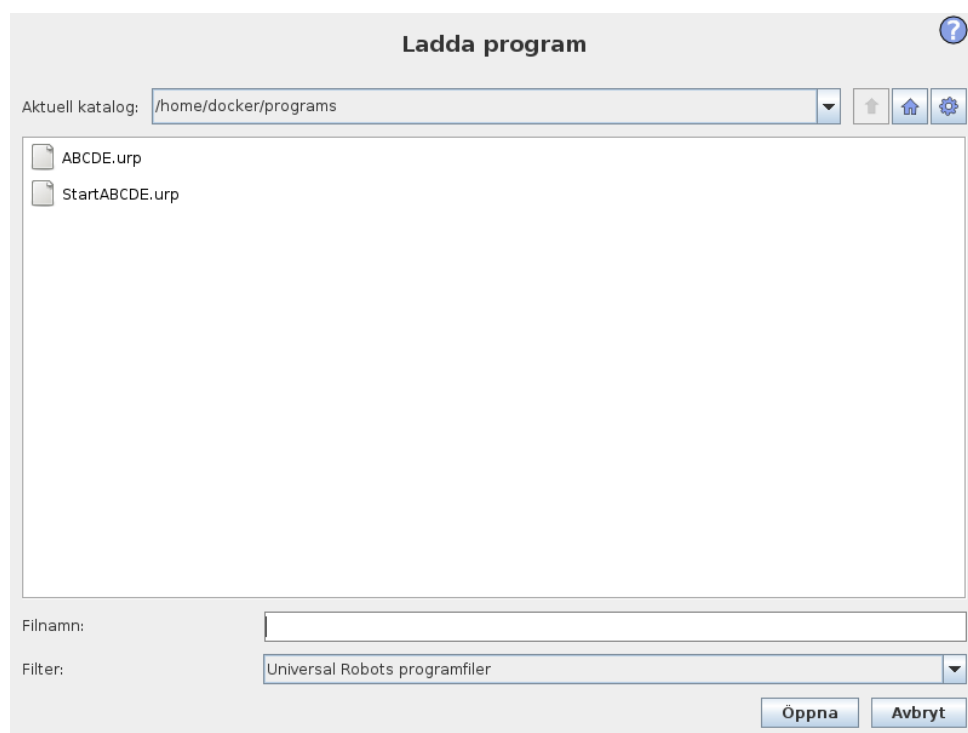

OBS:

Vi rekommenderar inte att program körs från en USB-enhet. Om du vill köra ett program som lagrats på en USB-enhet laddar du det först och sparar det sedan i den lokala `program`-mappen med kommandot **Spara som...** på **Arkiv**-menyn.

Den huvudsakliga skillnaden ligger i vilka åtgärder som är tillgängliga för användaren. På den grundläggande laddningsskärbilden får användaren bara läsa filer – inte ändra eller ta bort dem. Användaren får dessutom inte lämna den katalogstruktur som utgår från `program`mappen. Användaren kan gå till en underordnad katalog, men kan inte gå högre än mappen `program`.

Därför ska alla program placeras i `program`mappen och/eller undermappar under mappen `program`.

Skärmens layout



Den här bilden visar den faktiska laddningsskärbilden. Den består av följande viktiga områden och knappar:

Sökvägshistorik Sökvägshistoriken visar en lista över de vägar som lett fram till den aktuella platsen. Detta betyder att alla överordnade kataloger upp till datorns rotkatalog visas. Här kommer du att märka att du kanske inte har tillgång till alla kataloger ovanför `program`mappen.

När du väljer ett mappnamn i listan växlar dialogrutan till den valda katalogen, och visar den i området för val av fil (se 13.16).

Område för val av fil I den här delen av dialogrutan visas innehållet i det verkliga området. Här kan användaren markera en fil genom att enkelklicka på dess namn eller öppna filen genom att dubbelklicka på dess namn.

Mappar väljs med ett långt tryck på ca 0,5 sek. Tryck en gång för att öppna en mapp och visa dess innehåll.

Filfilter Genom att använda filfiltret kan du begränsa vilka filer som visas till önskade filtyper. Genom att välja **Säkerhetskopiera filer** kan området för filval visa de senaste 10 sparade versionerna av varje program, där .old0 är den senaste och .old9 är den äldsta.

Fältet Fil Här visas den fil som är markerad för närvarande. Användaren kan ange filnamnet manuellt genom att klicka på tangentbordsikonen till höger om fältet. Ett tangentbord visas då på skärmen, där användaren kan ange filnamnet direkt på skärmen.

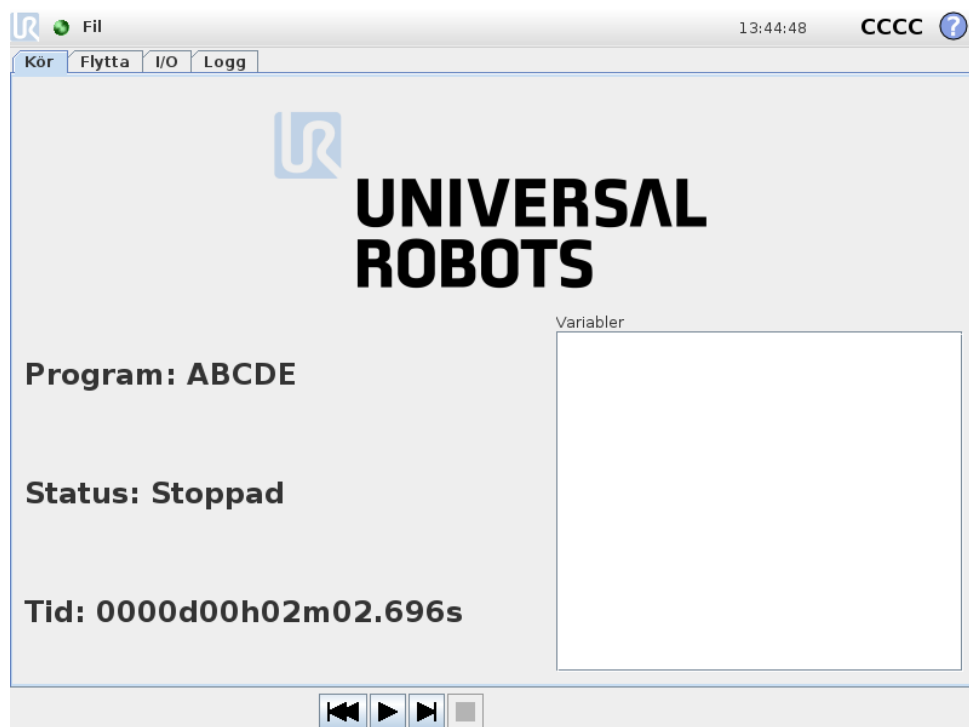
Knappen Öppna Om du klickar på knappen Öppna kommer den markerade filen att öppnas, varefter systemet återgår till föregående skärm.

Knappen Avbryt Om du klickar på Avbryt avbryts den pågående laddningsprocessen och skärmen återgår till föregående bild.

Åtgärdsknappar En rad knappar ger användaren möjlighet att utföra vissa åtgärder som brukar utföras genom att högerklicka på ett filnamn i en vanlig dialogruta. Dessutom finns möjlighet att gå upp i katalogstrukturen och gå direkt till programmappen.

- **Överordnad:** Gå upp i katalogstrukturen. Knappen är inte aktiverad i två fall: när den aktuella katalogen är den högsta nivån eller om skärmen är i begränsat läge och den aktuella katalogen är programmappen.
- **Gå till programmappen:** Gå hem
- **Åtgärder:** Åtgärder som exempelvis skapa katalog, ta bort fil osv.

13.17 Fliken Kör

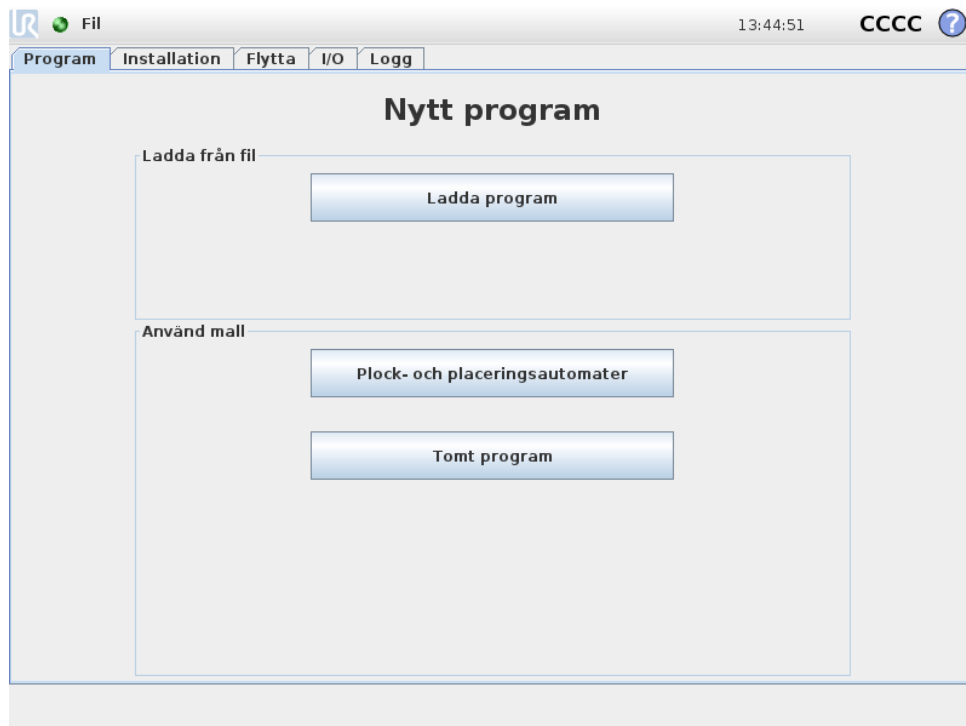


Den här fliken tillhandahåller ett mycket enkelt sätt att manövrera robotarmen, med så få knappar och alternativ som möjligt. Detta kan vara användbart i kombination med lösenordsskydd av programmeringsdelen i PolyScope (se 15.3), som kan göra roboten till ett verktyg som enbart kan köra specialskrivna program.

Dessutom kan på den här fliken ett standardprogram laddas och startas automatiskt, baserat på en kantövergång för en extern ingångssignal (se 13.14). Kombinationen av automatisk laddning och start av ett standardprogram och automatisk initiering vid start kan t.ex. användas för att integrera roboten i andra maskiner.

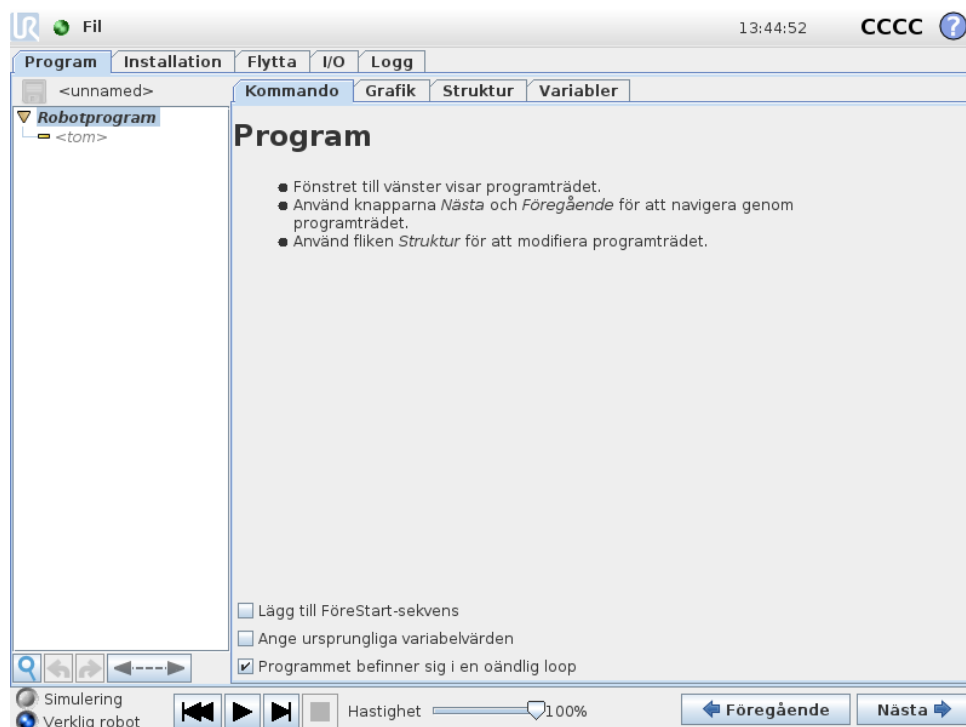
14 Programmering

14.1 Nytt program



Ett nytt robotprogram kan startas från en *mall* eller från ett befintligt (sparat) robotprogram. En *mall* kan ge en övergripande programstruktur, så att bara programuppgifterna behöver fyllas i.

14.2 Fliken Program



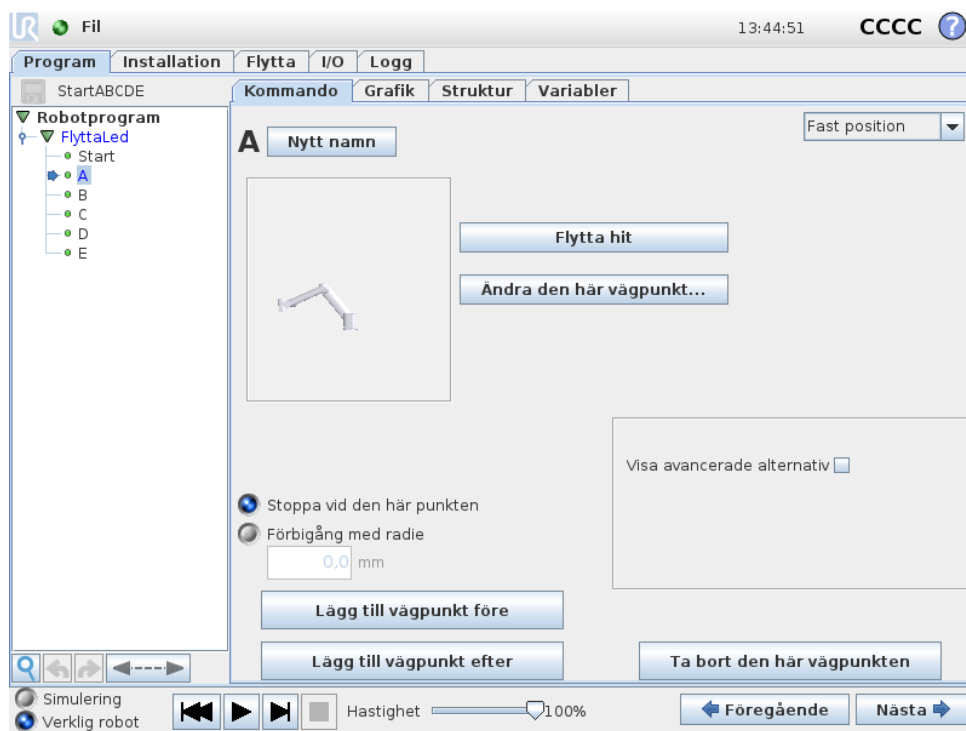
Programfliken visar det aktuella programmet som redigeras.

14.2.1 Programträd

Programträdet till vänster på skärmen visar programmet som en lista över kommandon, medan området till höger på skärmen visar information avseende det aktuella kommandot. Det aktuella kommandot väljs genom att klicka på kommandolistan, eller genom att använda knapparna **Föregående** och **Nästa** nederst till höger på skärmen. Kommandon kan infogas eller tas bort med hjälp av fliken **Struktur**. Programmets namn visas rakt ovanför kommandolistan, tillsammans med en liten disksymbol som kan klickas för att snabbt spara programmet.

I programträdet markeras det kommando som utförs enligt beskrivningen i 14.2.2.

14.2.2 Markering av programkörning



Programträdet innehåller visuella tecken om det kommando som utförs av robotstyrenheten. En liten -indikatorikon visas till vänster om kommandoikonen, och namnet på det kommando som körs samt eventuella kommandon som det här kommandot är ett underkommando till (identifieras normalt av  / -kommandoikoner) markeras med blått. Det gör det lättare för användaren att hitta det kommando som körs i trädet.

Om t.ex. robotarmen rör sig mot en vägpunkt markeras motsvarande vägpunktsunderkommando med -ikonen, och namnet, tillsammans med namnet på det Flytta-kommando (se 14.5) som det tillhör, visas i blått.

Om programmet pausas markerar programkörningsindikatorn det kommando som senast höll på att köras.

Om du klickar med knappen med -ikon en under programträdet hoppar du till det program i trädet som senast kördes eller som håller på att köras. Om du klickar på ett kommando medan programmet körs kommer fliken **Kommando** att visa information om det markerade kommandot. Om du trycker på knappen  visar fliken **Kommando** åter kontinuerligt information om de kommandon som körs.

14.2.3 Söknapp

Knappen med symbolen  kan användas för textsökning i programträdet. När du klickar här kan du sedan mata in en söktext och de programnoder som matchar sökningen markeras i gult. Tryck på  för att avsluta sökläget.

14.2.4 Knapparna Ångra/Gör om

Knapparna med ikonerna  och  under programträdet ångrar och upprepar ändringar som gjorts i programträdet och de kommandon trädet innehåller.

14.2.5 Programinstrumentbrädan

Den nedersta delen av skärmen är *instrumentbrädan*. *Instrumentbrädan* har en uppsättning knappar och liknar en gammaldags bandspelare. Härifrån kan programmen startas och stoppas, köras i enstaka steg samt startas om. Med *hastighetsreglaget* kan du när som helst justera programmets hastighet, och detta påverkar robotens rörelsehastighet. Dessutom visar *hastighetsreglaget* i realtid den relativa hastighet som robotarmen rör sig med, med hänsyn till säkerhetsinställningarna. Den angivna procentandelen är den högsta hastighet som kan uppnås för det program som körs utan att säkerhetsgränserna överskrids.

Till vänster om *instrumentbrädan* finns knapparna *Simulering* och *Verklig robot*, som växlar mellan att köra programmet i en simulering eller på den verkliga roboten. Vid körning i en simulering kommer robotarmen inte att röra sig, och den kan då inte heller skada sig själv eller annan utrustning i närheten. Använd simulering för att testa programmen om du är osäker på vad robotarmen kommer att göra.



FARA:

1. Se till att vara utanför robotens arbetsområde när knappen *Spela upp* trycks ned. Den rörelse du programmerat kanske inte är den du förväntade dig.
2. Se till att vara utanför robotens arbetsområde när knappen *Stega* trycks ned. Det kan vara svårt att förstå hur knappen *Stega* fungerar. Använd den bara när det är absolut nödvändigt.
3. Minska alltid hastigheten med *skjutreglaget* när du testar ditt program. Om integratören gjort programlogikfel kan detta orsaka att robotarmen rör sig på ett oväntat sätt.
4. När ett nödstopp eller skyddsstopp har inträffat stannar robotprogrammet. Det kan fortsätta förutsatt att ingen led har förflyttat sig mer än 10°. Om du trycker på uppspelningsknappen går roboten långsamt tillbaka till rörelsebanan och programexekveringen fortsätter.

Medan programmet skrivs visas robotarmens motsvarande rörelse som en 3D-ritning på fliken *Grafik* som beskrivs i 14.29.

Bredvid respektive programkommando finns en liten ikon som antingen är röd, gul eller grön. En röd ikon betyder att det finns ett fel i det aktuella kommandot, gul betyder att kommandot inte är klart och grönt betyder att allt är i sin ordning. Ett program kan bara köras när alla kommandon är gröna.

14.3 Variabler

Ett robotprogram kan använda variabler för att lagra och uppdatera olika värden under körning. Det finns två tillgängliga typer av variabler:

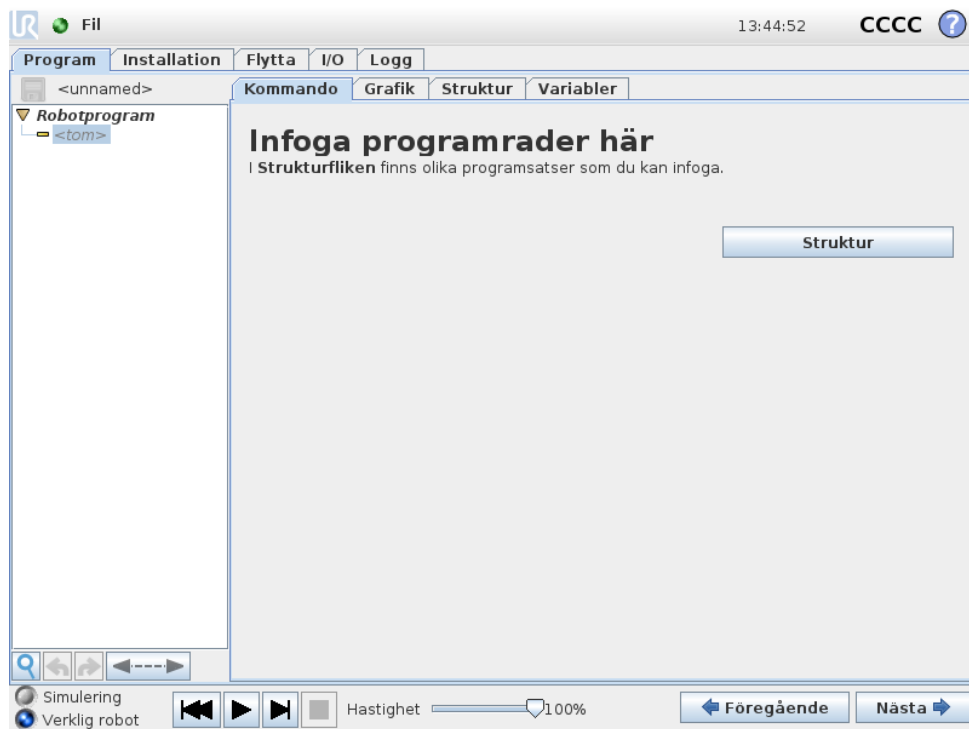
Installationsvariabler: Dessa kan användas av flera program, och deras namn och värden bibehålls tillsammans med robotinstallationen (se 13.10). Installationsvariablerna behåller sina värden efter att roboten och manöverskåpet har startats om.

Normala programvariabler: Dessa är endast tillgängliga för det program som körs, och deras värden går förlorade när programmet stoppas.

Följande typer av variabler är tillgängliga:

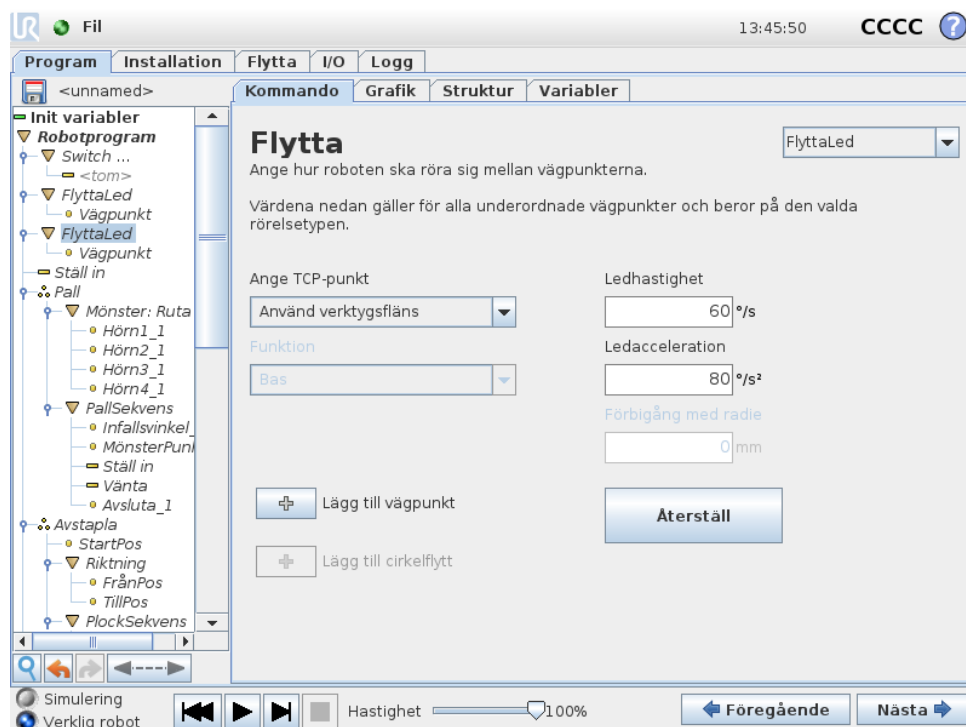
<i>bool</i>	En boolsk variabel som bara kan ha värdena <code>Sant</code> och <code>Falskt</code> .
<i>int</i>	Ett heltal i intervallet från <code>-2147483648</code> till <code>2147483647</code> (32-bit).
<i>float</i>	Ett tal med flytande decimal (ett vanligt decimaltal) (32-bit).
<i>string</i>	En sekvens med tecken.
<i>pose</i>	En vektor som beskriver plats och orientering i kartesisk rymd. Det är en kombination av en positionsvektor (x, y, z) och en rotationsvektor (rx, ry, rz) som representerar orienteringen, och den skrivs <code>p[x, y, z, rx, ry, rz]</code> .
<i>list</i>	En sekvens med variabler.

14.4 Kommando: Tom



Programkommandon ska infogas här. Tryck på knappen **Struktur** för att gå till strukturfliken. Där finns de olika programrader som kan väljas. Ett program kan inte köras innan alla rader specificerats och definierats.

14.5 Kommando: Flytta



Kommandot **Flytta** styr robotrörelsen genom de underliggande vägpunkterna. Vägpunkter måste ligga under ett Flytta-kommando. Kommandot Flytta definierar den acceleration och den hastighet med vilken robotarmen rör sig mellan dessa vägpunkter.

Rörelsetyper

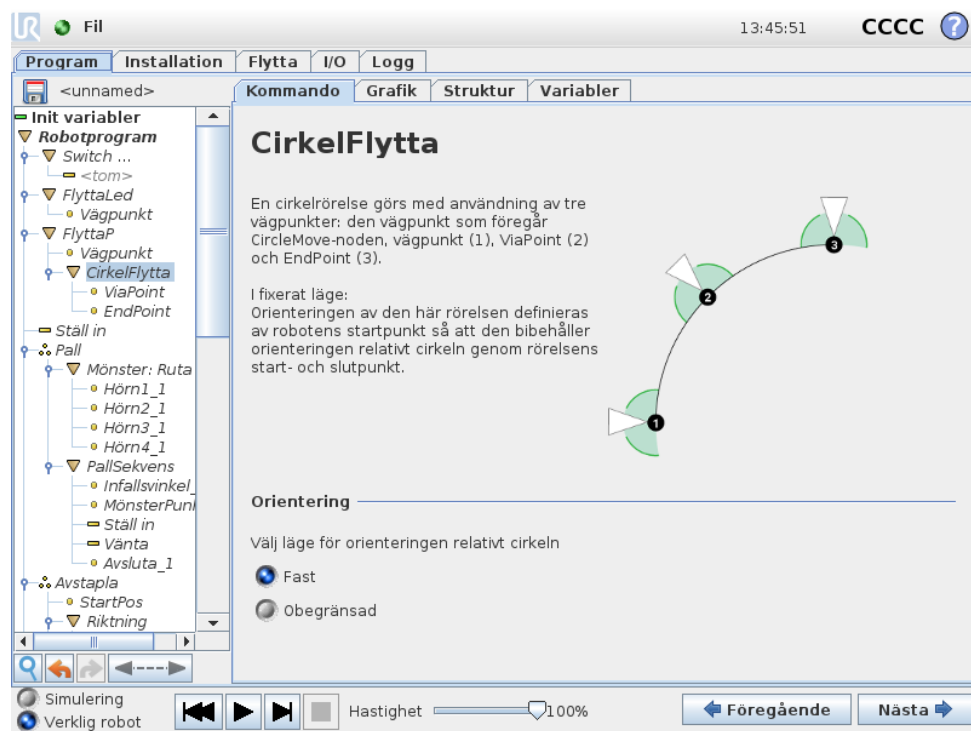
Du kan välja mellan tre olika typer av rörelser: **MoveJ**, **MoveL** and **MoveP**, vilka förklaras nedan.

- **moveJ** genererar rörelser som beräknas i robotarmens **ledutrymme**. Varje led styrs för att nå önskad slutposition samtidigt. Denna rörelsetyp gör att verktyget rör sig i en böjd kurva. De delade parametrar som gäller för den här rörelsetypen är den högsta ledhastigheten och ledaccelerationen som ska användas för rörelseberäkningarna, angivet i *grader/s* och *grader/s²*. Om robotarmen ska röra sig snabbt mellan vägpunkterna, och verktygets bana mellan vägpunkterna är oviktig, är detta den bästa rörelsetypen.
- **FlyttaL** flyttar verktyget linjärt mellan vägpunkterna. Det betyder att varje led utför en mer komplicerad rörelse för att hålla verktyget i rörelse längs en rät linje. De delade parametrar som kan anges för den här rörelsetypen är den önskade verktygshastigheten och verktygsaccelerationen, angivet i *mm/s* och *mm/s²*, och även en funktion. Den funktion som väljs bestämmer i vilken funktionsrymd verktygets positioner i vägpunkterna ska representeras. Av speciellt intresse när det gäller funktionsrymder är variabla funktioner och variabla vägpunkter. Variabla funktioner kan användas när verktygets position i en vägpunkt måste bestämmas av det verkliga värdet på den variabla funktionen när robotprogrammet körs.

- **FlyttaP** flyttar verktyget linjärt med konstant hastighet med cirkulära blandningar, och är avsett för vissa processoperationer, som limning och dispensering. Storleken på blandningsradien är som standard ett värde som delas av alla vägpunkter. Ett lägre värde gör att banan svänger skarpare, medan ett högre värde gör banan jämnare. När robotarmen rör sig mellan vägpunkterna med konstant hastighet kan robotens manöverskåp inte vänta på vare sig en I/O-operation eller en operatörsåtgärd. Detta skulle kunna stoppa robotarmens rörelse eller orsaka ett skyddsstopp.
- **Cirkulär rörelse** kan läggas till en **flyttaP** för att göra en cirkulär rörelse. Roboten startar rörelsen från dess nuvarande position eller startpunkt, går igenom en **ViaPoint** specificerad på den cirkulära bågen, samt en **EndPoint** som slutför den cirkulära rörelsen.

Ett läge används för att beräkna verktygsorienteringen genom den cirkulära bågen. Läget kan vara:

- Fast: endast startpunkten används för att definiera verktygsorienteringen
- Obegränsat: startpunkten övergår i **EndPoint** för att definiera verktygsorienteringen



Delade parametrar

De delade parametrarna längst ned i högra hörnet på skärmbilden Flytta gäller för rörelsen från robotens föregående position till dess första vägpunkt under kommandot, och därifrån till var och en av de efterföljande vägpunkterna. Inställningarna för kommandot Flytta gäller inte för rörelsesträckan som går *från* den sista vägpunkten under det aktuella Flytta-kommandot.

TCP-val

Den TCP som används för vägpunkterna under detta Flytta-kommando kan väljas i kombinationsrutan. Det är möjligt att välja mellan olika användardefinierade TCP från installationen, den aktiva TCP eller helt enkelt använda verktygsfläns. Om en användardefinierad TCP eller aktiv TCP väljs, kommer rörelsen under detta Flytta-kommando att justeras i förhållande till detta. Om Använd verktygsfläns väljs, kommer inget TCP att användas och rörelsen under detta Flytta-kommando kommer att vara i förhållande till verktygsflänsen (dvs. ingen justering till rörelsen).

Om aktiv TCP för den här rörelsen bestäms vid programkörning, måste den anges dynamiskt med kommandot Ställ in (se 14.10) eller med skriptkommandon. För ytterligare information om konfigurerings av namngivna TCP:er, (se 13.6).

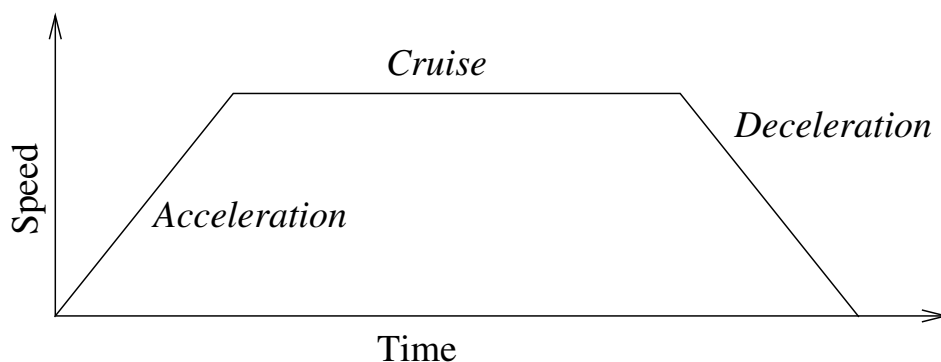
Funktionsval

För **FlyttaL** och **FlyttaP** går det att välja i vilken funktionsrymd vägpunkterna under kommandot Flytta ska representeras när vägpunkterna specificeras (se avsnitt 13.12). Detta betyder att när en vägpunkt anges, kommer programmet att komma ihåg verktygets koordinater i den valda funktionens funktionsrymd. Vissa omständigheter kräver en detaljerad förklaring:

Relativa vägpunkter: De valda funktionerna har ingen effekt på relativa vägpunkter. Den relativa rörelsen sker alltid relativt **Basens** orientering.

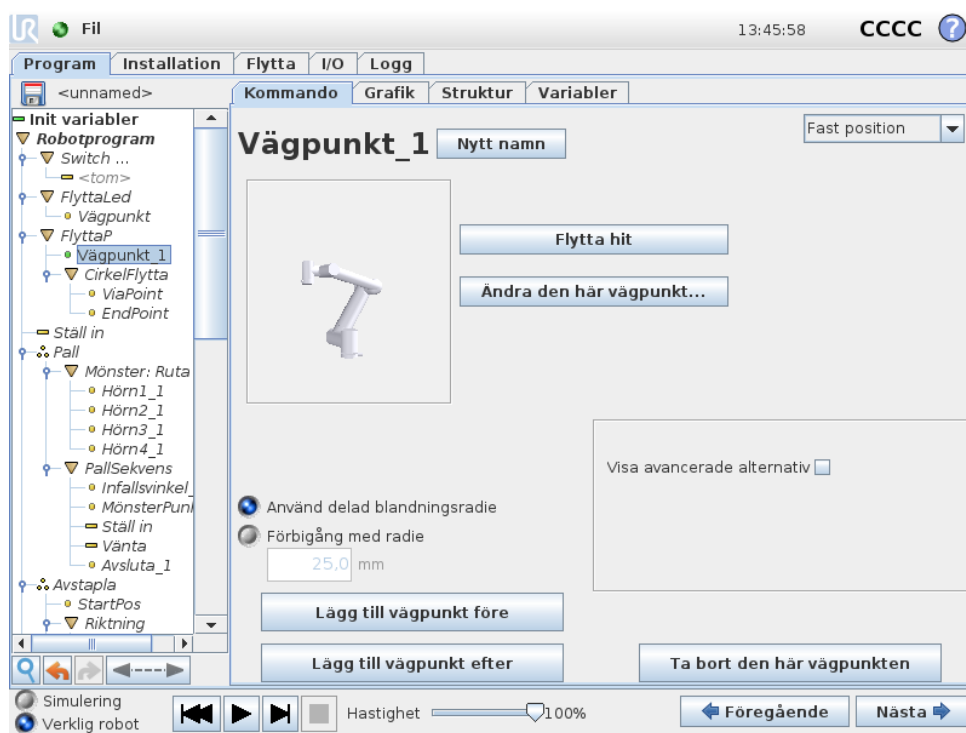
Variabla vägpunkter: När robotarmen flyttas till en variabel vägpunkt beräknas verktygets målposition alltid som koordinaterna för variabeln i den valda funktionens funktionsrymd. Därför kommer robotarmens rörelse för en variabel vägpunkt alltid att ändras om en annan funktion väljs.

Variabel funktion: Om någon av funktionerna i den inlästa installationen har valts som variabel, går det även att välja motsvarande variabler i funktionsvals-menyn. Om en funktionsvariabel (med samma namn som funktionen, föregånget av "_var") väljs, rör sig robotarmen (förutom mot **Relativa vägpunkter** relativt variabelns verkliga värde när programmet körs. Startvärdet på en funktionsvariabel är funktionens verkliga värde enligt konfigurationen i installationen. Om detta värde ändras, förändras även robotens rörelser.



Figur 14.1: Hastighetsprofil för en rörelse. Kurvan är indelad i tre segment: *acceleration*, *farthållare* och *inbromsning*. Nivån för fasen *farthållare* styrs av rörelsens hastighetsinställning, medan stigningen för faserna *acceleration* och *inbromsning* definieras av accelerationsparametern.

14.6 Kommando: Fast vägpunkt



En punkt längs robotens bana. Vägpunkter är den mest centrala delen i ett robotprogram, eftersom de informerar robotarmen om var den ska vara. En vägpunkt med fast position ställs in genom att flytta robotarmen fysiskt till den aktuella positionen.

Ange vägpunkten

Tryck på den här knappen för att gå till skärmbilden Flytta, där du kan ange robotarmens position för denna vägpunkt. Om vägpunkten är placerad under ett Flytta-kommando i ett linjärt rum (FlyttaLinjär eller FlyttaProcess) måste en giltig funktion ha valts vid det aktuella Flytta-kommandot, om knappen ska kunna tryckas in.

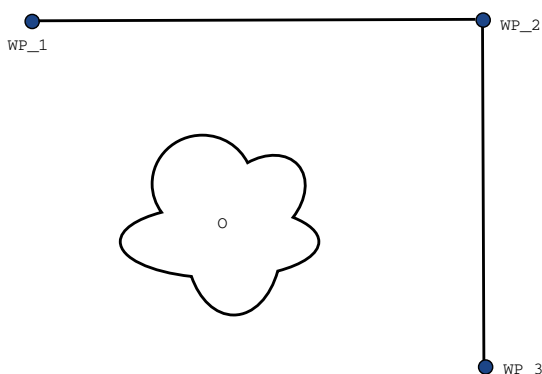
Vägpunkternas namn

Vägpunkterna får automatiskt ett unikt namn när de definieras. Namnet kan ändras av användaren. Vägpunkter med samma namn delar positionsinformationen. Således kommer en ändrad position för en vägpunkt att påverka alla andra vägpunkter med samma namn. Övrig vägpunktsinformation såsom förbigångsradien, verktyg-/ledhastighet och verktyg-/ledacceleration konfigureras för individuella vägpunkter även om de har samma namn.

Förbigång

Förbigång låter roboten smidigt övergå mellan två banor, utan att stanna vid vägpunkten mellan dem.

Exempel Tänk dig en plock- och placeringstillämpning som ett exempel (se figur 14.2), där roboten för närvarande är vid Vägpunkt 1 (WP_1) och den måste plocka upp ett föremål vid Vägpunkt 3 (WP_3). För att undvika kollisioner med föremål och andra hinder (O), måste roboten närma sig WP_3 i riktningen när den kommer från vägpunkt 2 (WP_2). Så tre vägpunkter införs för att skapa en väg som uppfyller kraven.



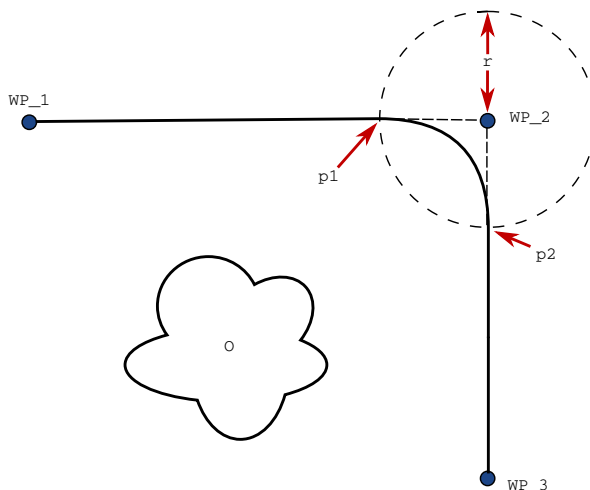
Figur 14.2: WP_1: utgångsläge, WP_2: via-punkt, WP_3: plockningsposition, O: hinder.

Utan att konfigurera andra inställningar, kommer roboten att göra ett stopp vid varje vägpunkt innan den fortsätter rörelsen. För denna uppgift är inte ett stopp vid WP_2 optimalt eftersom en mjuk sväng skulle kräva mindre tid och energi och samtidigt uppfylla kraven. Det är till och med godtagbart att roboten inte når exakt WP_2, så länge som övergången från den första banan till andra sker nära denna position.

Stoppet vid WP_2 kan undvikas genom att konfigurera en förbigång av den vägpunkten, vilket tillåter roboten att beräkna en mjuk övergång till nästa bana. Den primära parametern för en förbigång är en radie. När roboten befinner sig inom vägpunktens förbigångsradien kan den inleda förbigången och avvika från originalbanan. Detta möjliggör snabbare och mjukare rörelser, eftersom roboten inte behöver bromsa och accelerera igen.

Förbigångsparametrar Förutom vägpunkterna kommer flera olika parametrar att påverka förbigångsbanan (se figur 14.3):

- förbigångsradien (r)
- den ursprungliga och slutliga hastigheten för roboten (vid positionerna $p1$ respektive $p2$)
- rörelsetiden (t.ex. för att ställa in en specifik tid för en bana; detta kommer att påverka den ursprungliga och slutliga hastigheten för roboten)
- bantyperna att övergå från och till (FlyttaLinjär, FlyttaLed)

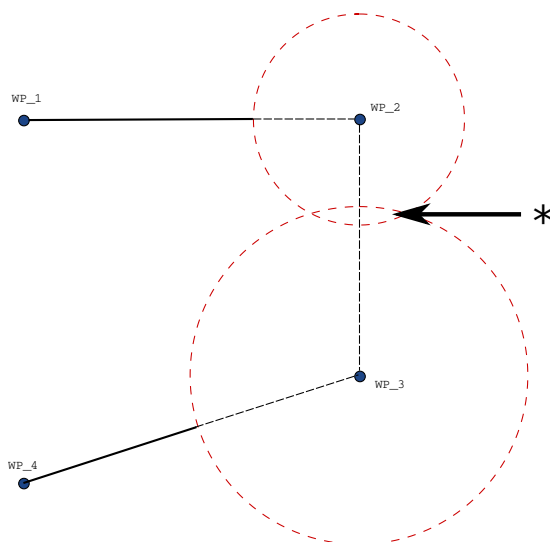


Figur 14.3: Förbigå WP_2 med radie r , ursprunglig förbigångsposition vid $p1$ och slutlig förbigångsposition vid $p2$. O är ett hinder.

Om en förbigångsradien anges kommer robotarmens rörelsebana att gå runt vägpunkten så att robotarmen inte stannar vid den punkten.

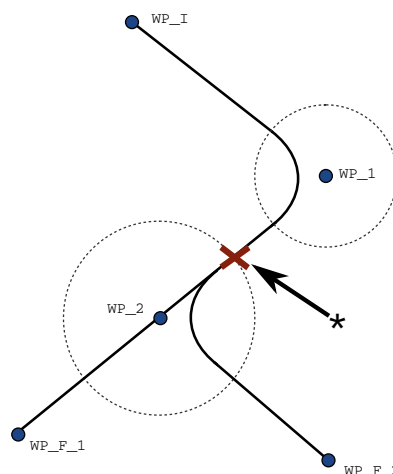
Förbigångar kan inte överlappa varandra, så det går inte att ställa in en förbigångsradien som överlappar en förbigångsradien för en tidigare eller senare vägpunkt enligt vad som visas i figur 14.4.

Villkorliga förbigångsbanor Förbigångsbanan påverkas både av den vägpunkt där förbigångsradien ställs in och följande punkt i programträdet. Det vill säga, i programmet i figuren 14.5 påverkas förbigången runt WP_1 av WP_2. Konsekvensen av detta blir mer uppenbar vid en förbigång runt WP_2 i detta exempel. Det finns två möjliga slutpositioner och för att avgöra vilken som är nästa vägpunkt att förbigå, måste roboten utvärdera nuvarande avläsning av `digital_ingång[1]` redan vid inträdet i förbigångsradien. Det innebär att uttrycket **om...så** (eller andra nödvändiga uppgifter för att fastställa följande vägpunkt, t.ex. variabla vägpunkter) utvärderas innan den faktiskt når WP_2 vilket är något krångligt när man tittar på programförloppet. Om en vägpunkt är en stoppunkt som följs av ett villkorligt uttryck för att avgöra nästa vägpunkt (t.ex. ett I/O-kommando) utförs detta när robotarmen stannar vid vägpunkten.



Figur 14.4: Överlappning av förbigångsradien är inte tillåten (*).

```
FlyttaLinjär
  WP_I
  WP_1 (förbigång)
  WP_2 (förbigång)
  om (digital_ingång[1]) så
    WP_F_1
  annars
    WP_F_2
```

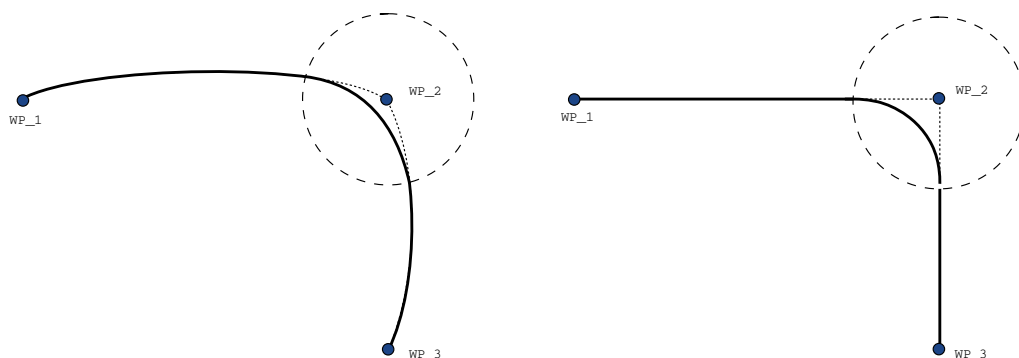


Figur 14.5: WP_I är den ursprungliga vägpunkten och det finns två potentiella slutliga vägpunkter WP_F_1 och WP_F_2, beroende på ett villkorligt uttryck. Det villkorliga om-uttrycket utvärderas när robotarmen går in en andra förbigång (*).

Kombinationer av bantyper Det är möjligt att övergå mellan fyra kombinationer av bantyper av **FlyttaLed** och **FlyttaLinjär** men den specifika kombinationen kommer att påverka den beräknade förbigångsbanan. Det finns fyra möjliga kombinationer:

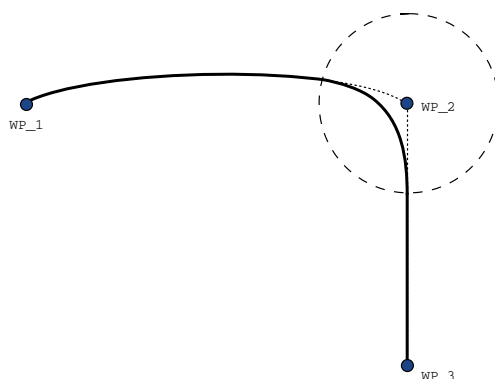
1. **FlyttaLed** till **FlyttaLed** (ren förbigång i ledrummet)
2. **FlyttaLed** till **FlyttaLinjär**
3. **FlyttaLinjär** till **FlyttaLinjär** (ren förbigång i kartesisk rymd)
4. **FlyttaLinjär** till **FlyttaLed**

Ren förbigång i ledrummet (punkt 1) mot ren förbigång i kartesisk rymd (punkt 3) jämförs i figur 14.6. Det visar två möjliga vägar för verktyget för identiska uppsättningar av vägpunkter.



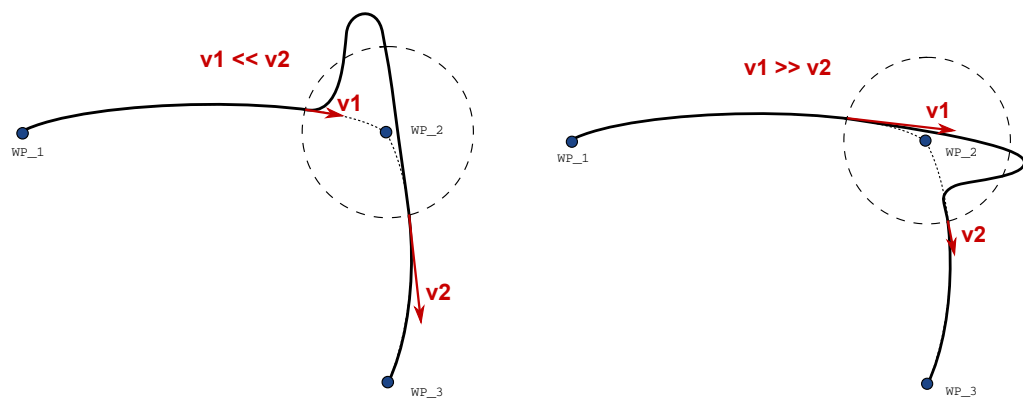
Figur 14.6: Förflyttning och förbigång i ledrummet (FlyttaLed) mot kartesisk rymd (FlyttaLinjär).

Av de olika kombinationerna, kommer punkt 2, 3 och 4 leda till banor som håller sig inom gränserna för den ursprungliga banan i den kartesiska rymden. Ett exempel på en blandning mellan olika bantyper (punkt 2) kan ses i figur 14.7.



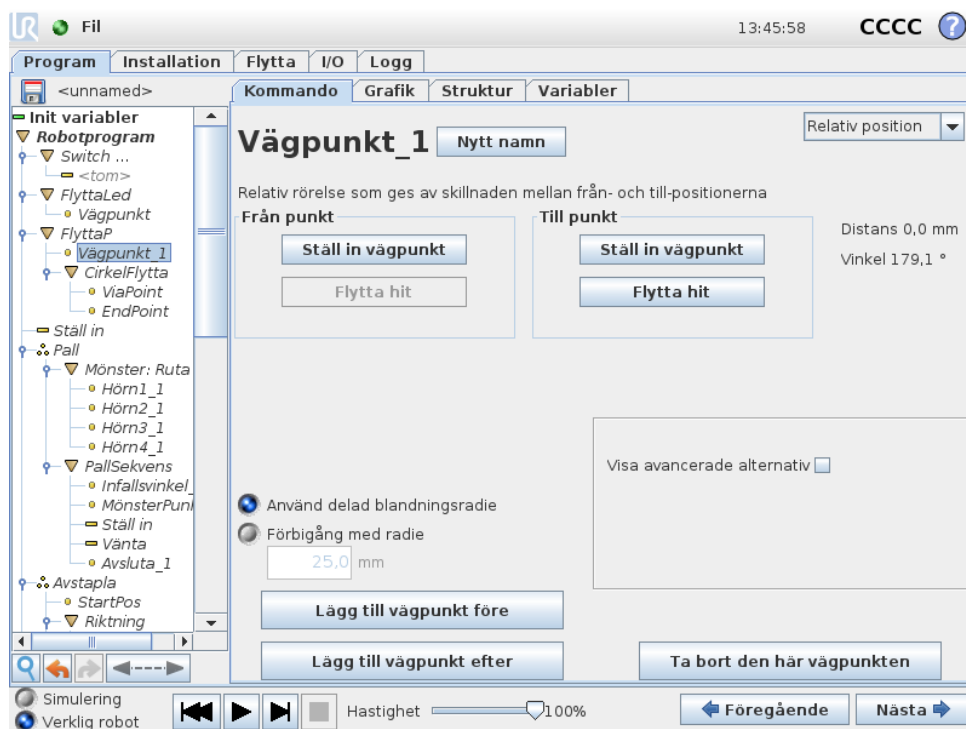
Figur 14.7: Förbigång från en rörelse i ledrummet (FlyttaLed) till linjär verktygsrörelse (FlyttaLinjär).

Rena förbigångar i ledrummet (punkt 1) kan däremot uppträda på ett sätt som är mindre intuitivt, eftersom roboten kommer att försöka uppnå mjukast möjliga bana i ledrummet med beaktande av hastigheter och tidskrav. På grund av detta kan de avvika från kursen som anges av vägpunkterna. Detta gäller särskilt om det finns betydande skillnader i ledens hastighet mellan de två banorna. *Varning:* om hastigheterna är mycket annorlunda (t.ex. genom att ange avancerade inställningar – antingen för hastighet eller tid – för en specifik vägpunkt) kan detta resultera i stora avvikelser från den ursprungliga banan som visas i figur 14.8. Om du behöver göra en förbigång mellan olika hastigheter och inte kan acceptera denna avvikelse kan du överväga en förbigång i den kartesiska rymden med hjälp av **FlyttaLinjär** istället.



Figur 14.8: Förbigång från en rörelse när utgångshastigheten $v1$ är betydligt lägre än sluthastigheten $v2$ eller tvärtom.

14.7 Kommando: Relativ vägpunkt

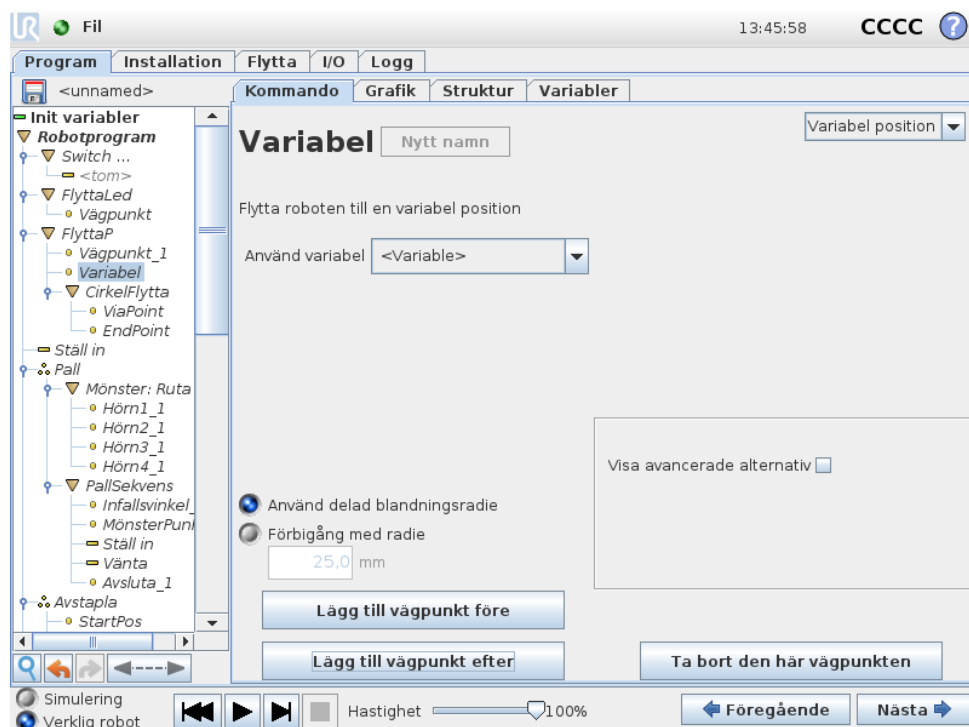


En vägpunkt där positionen ges i förhållande till robotarmens föregående position, till exempel "två centimeter åt vänster". Den relativa positionen definieras som skillnaden mellan de två givna positionerna (vänster till höger).

Observera att upprepade relativa positioner kan förflytta robotarmen ut ur dess arbetsområde.

Den här sträckan är den kartesiska sträckan mellan TCP-värdena i de två lägena. Vinkeln anger hur mycket TCP-orienteringen ändras mellan de två positionerna. Mer exakt, längden av den rotationsvektor som beskriver ändringen i orientering.

14.8 Kommando: Variabel vägpunkt



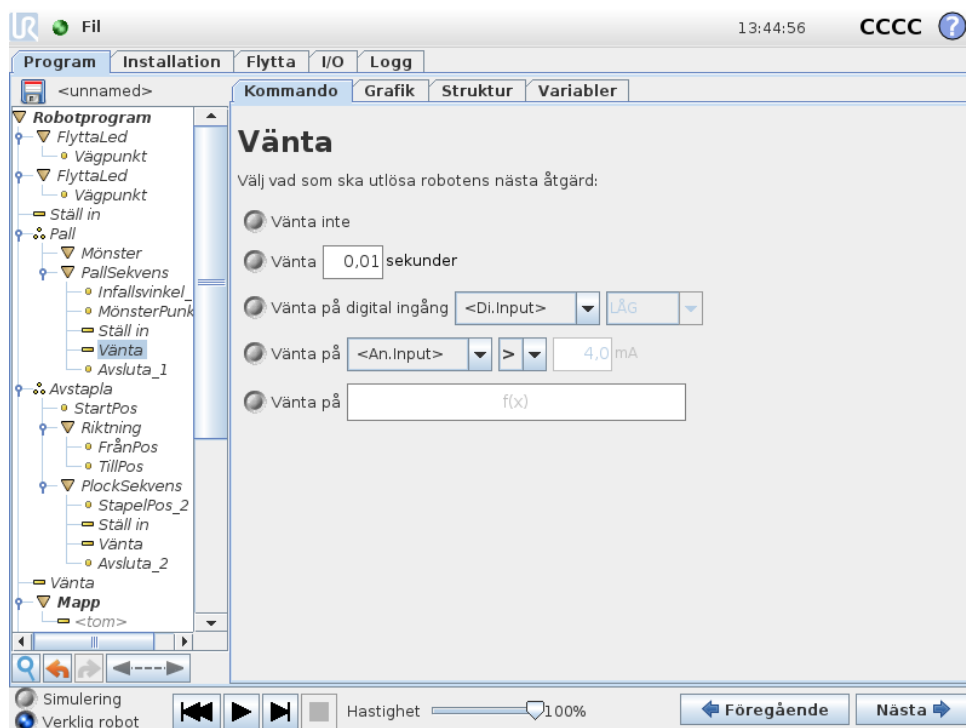
Detta är en vägpunkt där positionen ges av en variabel, i det här fallet `beräknad_position`. Variabeln måste vara en *pose*, t.ex.

`var=p[0.5,0.0,0.0,3.14,0.0,0.0]`. De första tre är *x,y,z* och de sista tre är orienteringen som en *rotationsvektor* på formen *rx,ry,rz*. Axelns längd är den vinkel som ska roteras i radianer, och vektorn själv ger den axel runt vilken rotationen ska göras. Positionen anges alltid i relation till en referensram eller ett koordinatsystem som definierats av vald funktion. Robotarmen förflyttas alltid linjärt till en variabel vägpunkt. Om en förbigångsradius sätts till en fast vägpunkt och de föregående och efterföljande vägpunkterna är variabla eller om en förbigångsradius sätts till en variabel vägpunkt, så kommer inte förbigångsradien att kontrolleras för eventuella överlappningar (se 14.6). Om en förbigångsradius överlappar en punkt vid körning av programmet, kommer roboten att bortse från den och gå vidare till nästa.

Så här flyttar man t.ex. roboten 20 mm längs verktygets z-axel:

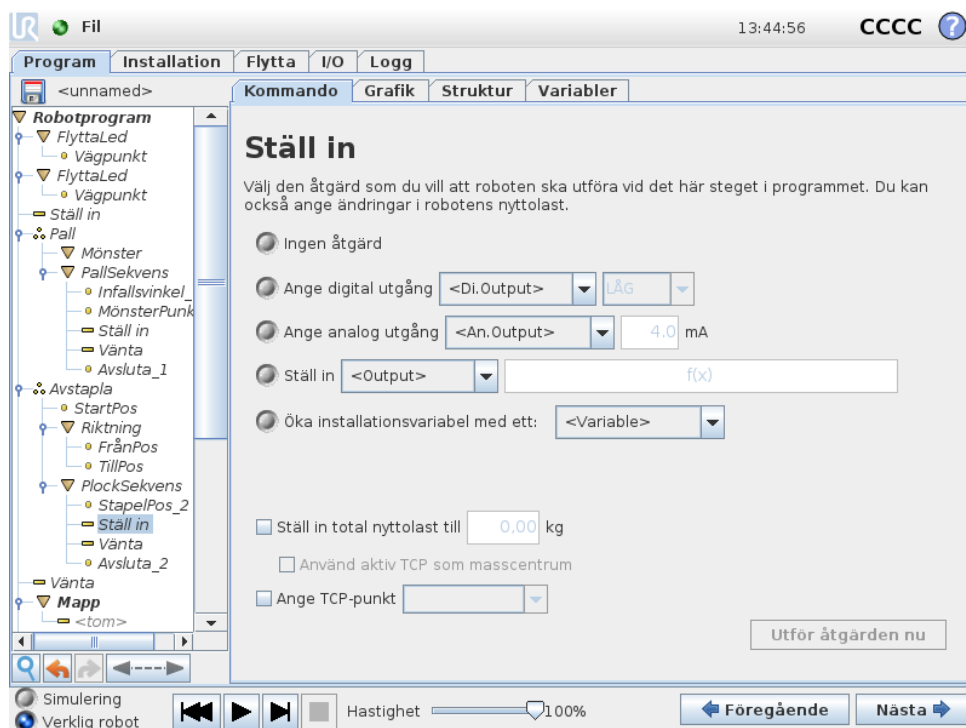
```
var_1=p[0,0,0.02,0,0,0]
FlyttaLinjär
  Vägpunkt_1 (variabel position):
    Använd variabel=var_1, Funktion=Verktyg
```

14.9 Kommando: Vänta



Vänta pausar en I/O-signal, eller uttryck under en viss tidsperiod. Om **Ingen väntetid** väljs utförs ingen åtgärd.

14.10 Kommando: Ställ in

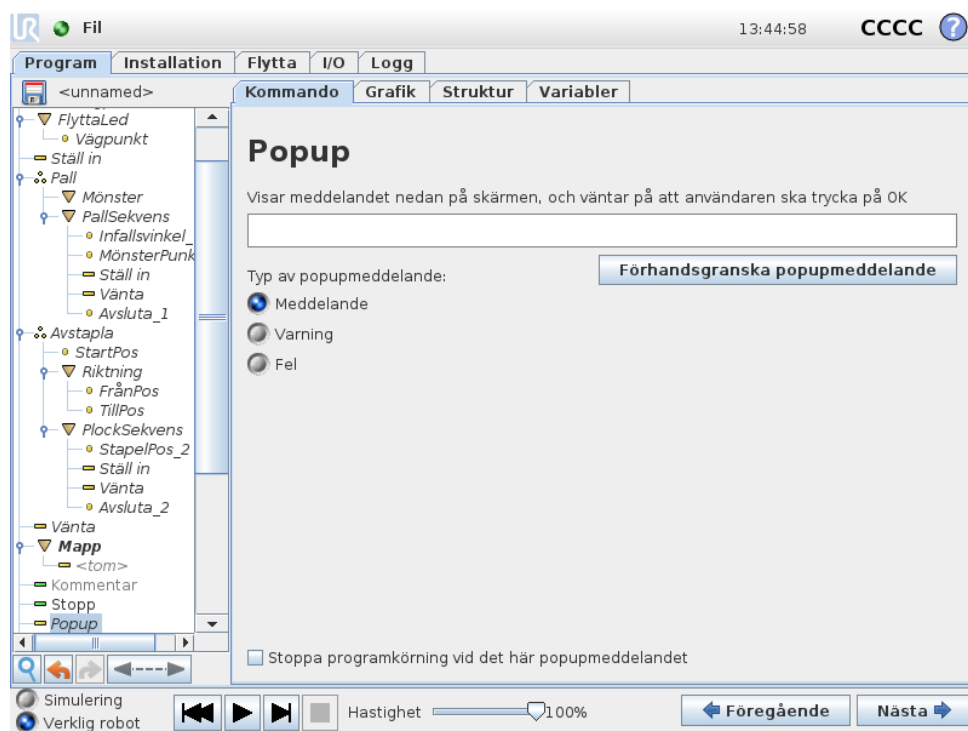


Väljer digitala eller analoga utgångar för ett givet värde.

Kommandot kan också användas för att ställa in robotarmens nyttolast. Det kan vara nödvändigt att justera nyttolasten för att förhindra roboten från att oväntat utlösa ett skyddsstopp, när ett verktygs vikt inte stämmer överens med förväntad nyttolast. Som standard används också den aktiva TCP:en som masscentrum. Om den aktiva TCP:en inte bör användas som masscentrum kan kryssrutan avmarkeras.

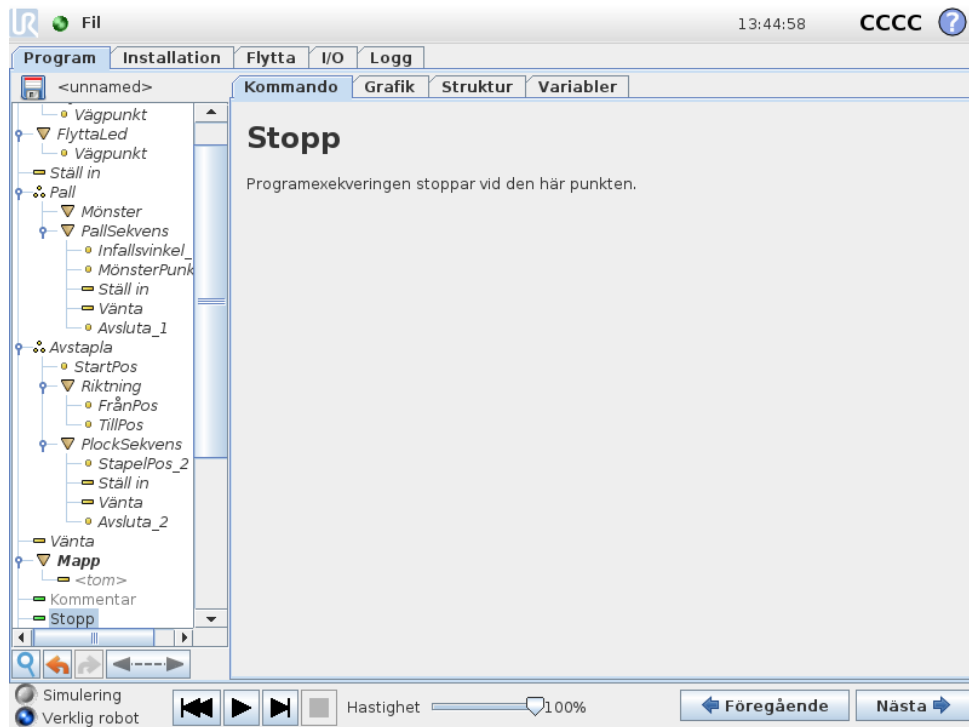
Aktiv TCP kan även ändras med ett **Ställ in**-kommando. Markera bara kryssrutan och välj en av TCP-förskjutningarna på menyn. Om aktiv TCP för en viss rörelse är känd när programmet skrivs, överväg istället att använda TCP-valet på kortet **Flytta** (se 14.5). För ytterligare information om konfigurering av namngivna TCP:er, (se 13.6).

14.11 Kommando: Popup



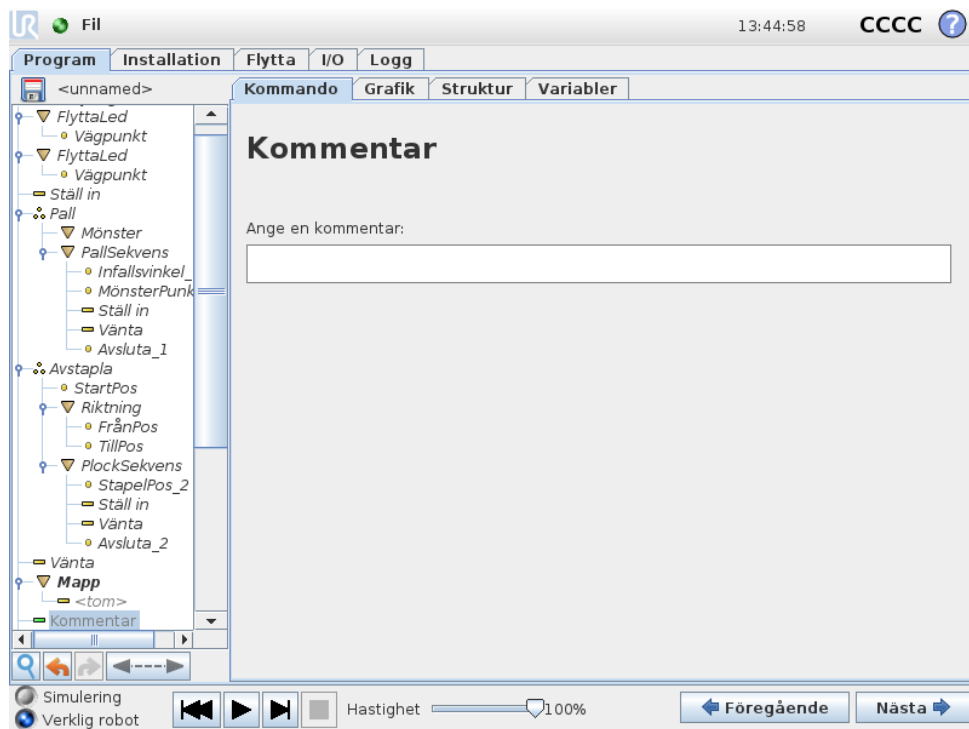
Ett popupmeddelande är ett meddelande som visas på skärmen när programmet kommer till det här kommandot. Du kan välja utseende för meddelandet och själva texten med tangentbordet på skärmen. Roboten väntar på att användaren/operatören ska trycka på "OK"-knappen under popupmeddelandet innan programmet fortsätter. Om alternativet "Stoppa programkörning" väljs stannar robotprogrammet vid det här popupmeddelandet.

14.12 Kommando: Stopp



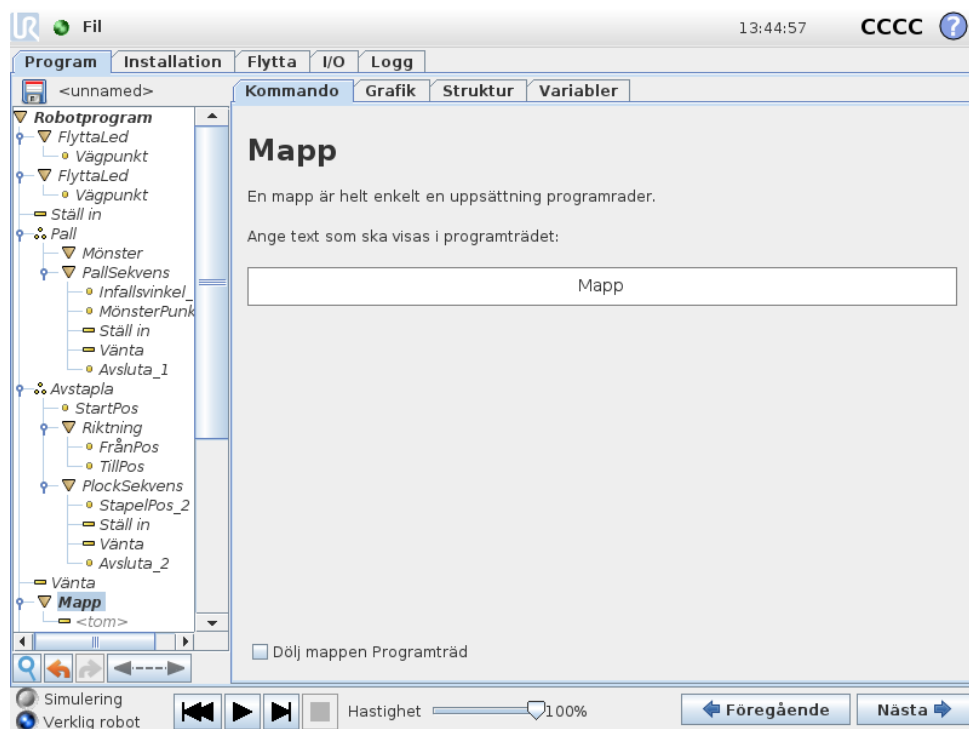
Programexekveringen stoppar vid den här punkten.

14.13 Kommando: Kommentar



Ger programmeraren möjlighet att lägga till en textrad i programmet. Den här textraden utför ingenting under programexekveringen.

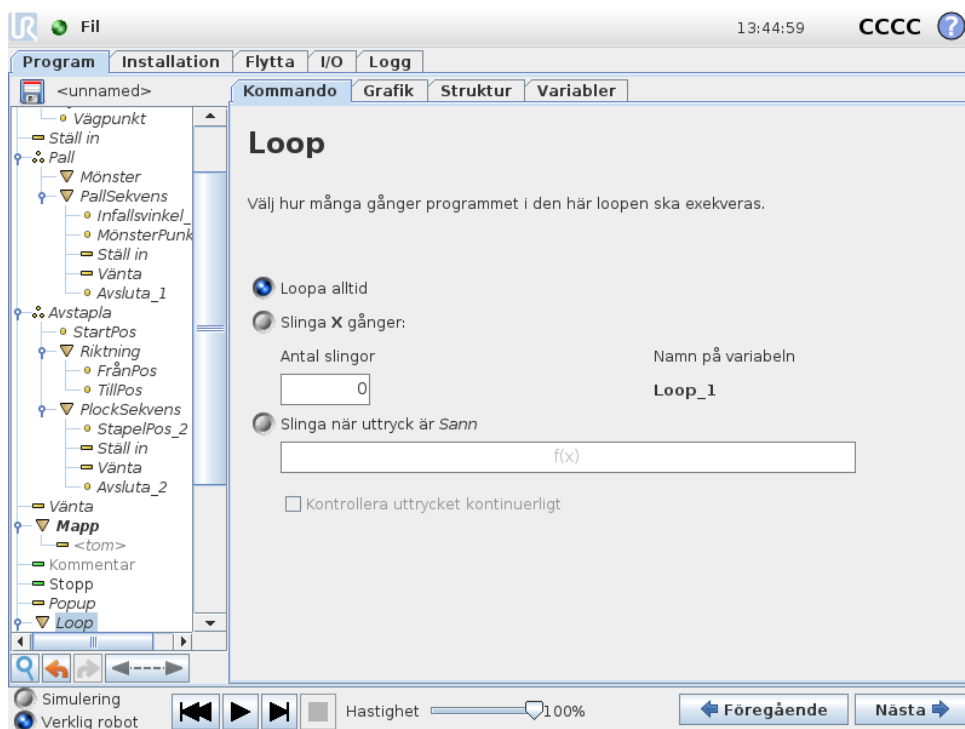
14.14 Kommando: Mapp



En **Mapp** används för att organisera och märka särskilda delar av ett program, för att rensa upp i programträdet och för att göra programmet lättare att avläsa och navigera i.

Själva **mappen** utför ingenting.

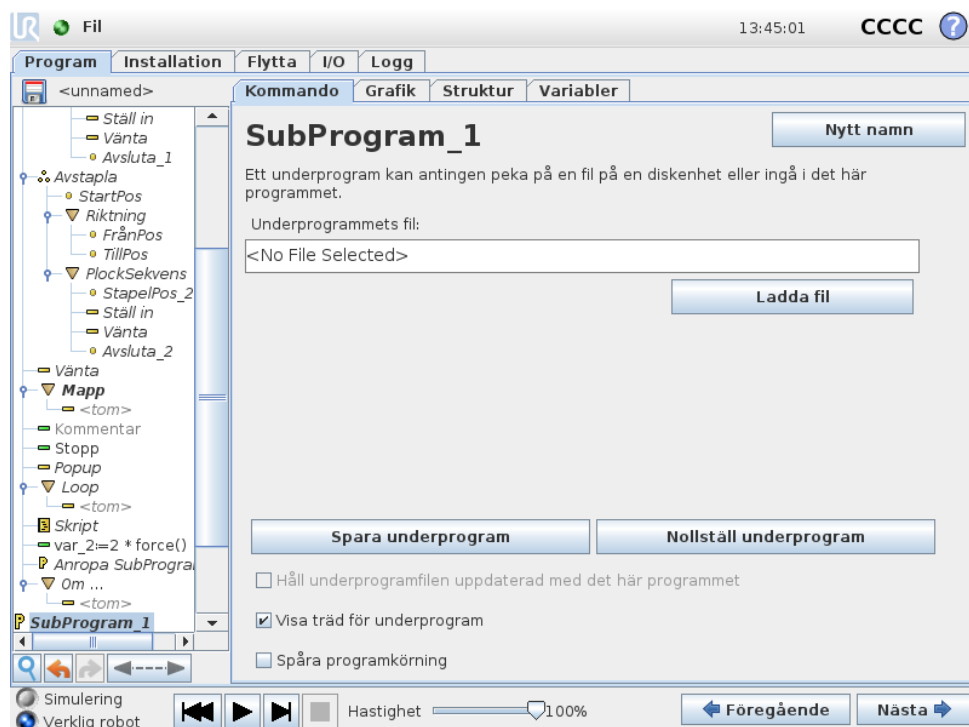
14.15 Kommando: Loop



Loopar underliggande programkommandon. Beroende på vilka val som gjorts kan de underliggande programkommandona antingen loopas oändligt, ett visst antal gånger eller så länge som det givna tillståndet är sant. Vid loopning ett visst antal gånger skapas en dedicerad loopvariabel (benämnd `loop_1` i skärmbilden ovan), och denna kan användas i uttryck inom loopen. Loopvariabeln räknar från 0 till $N - 1$.

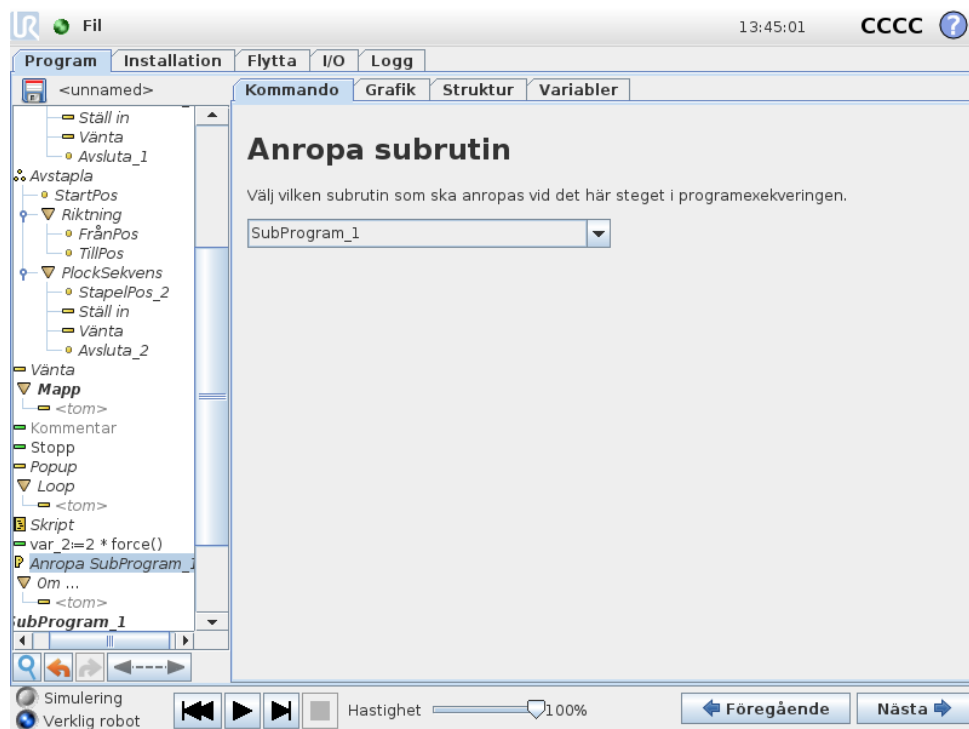
Vid loopning med användning av ett uttryck som sluttillstånd ger PolyScope möjlighet att utvärdera detta uttryck kontinuerligt, så att "loopen" kan avbrytas när som helst under exekveringen, och inte bara efter varje förekomst.

14.16 Kommando: SubProgram



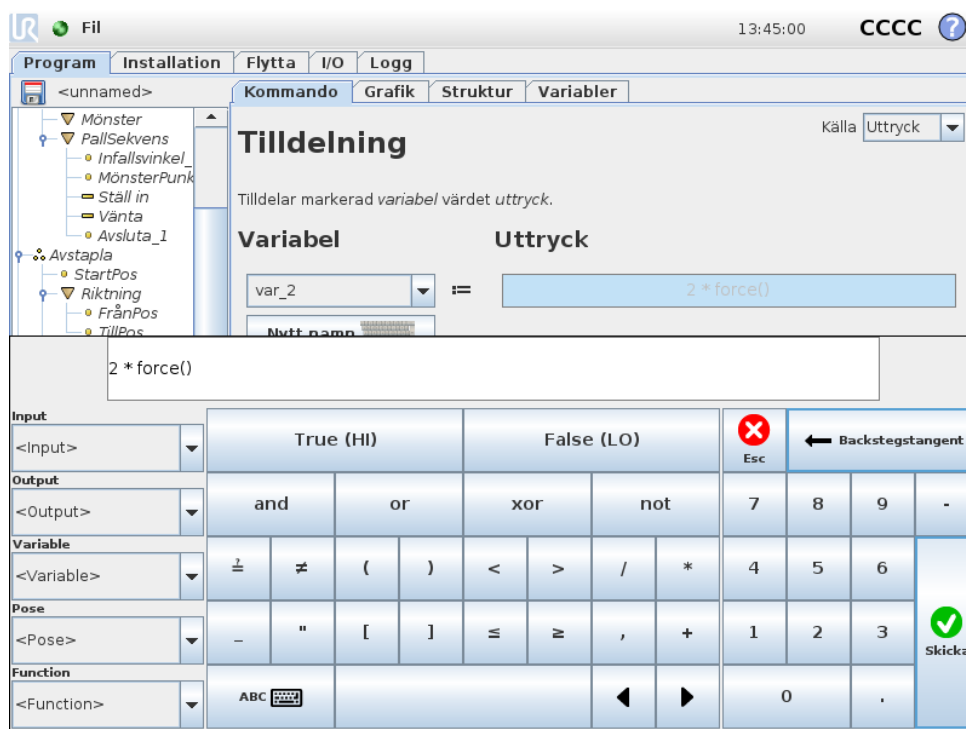
Ett underprogram kan innehålla programdelar som behövs på flera platser. Ett underprogram kan vara en separat fil på diskenheten och det kan också döljas som skydd mot oavsiktliga ändringar.

Kommando: Anropa SubProgram



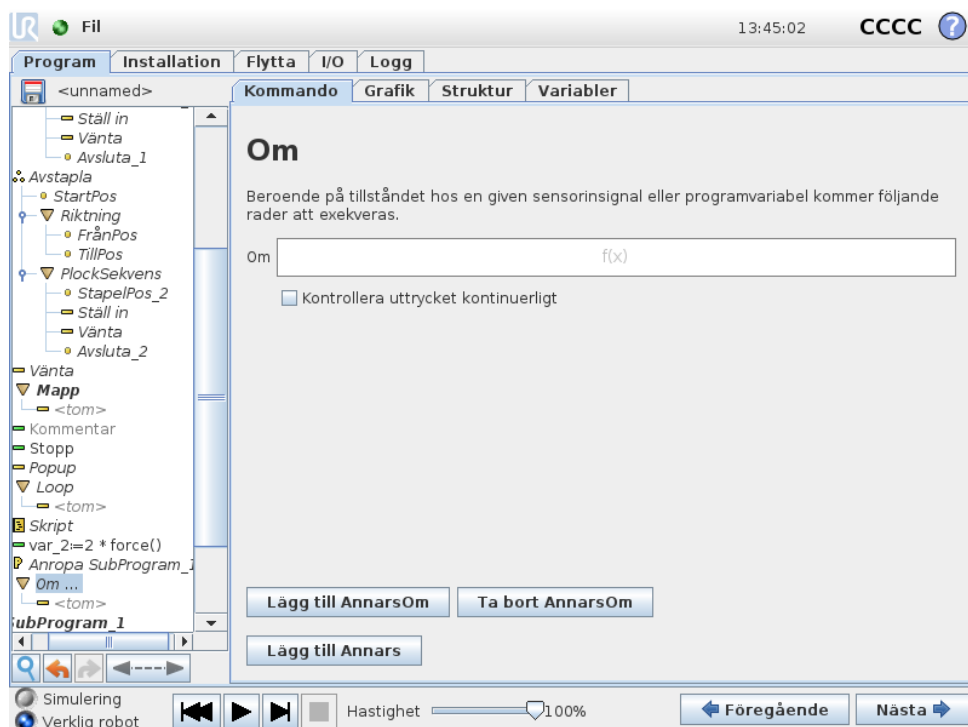
Vid ett anrop till ett underprogram körs programraderna i underprogrammet och sedan fortsätter programmet med nästa rad.

14.17 Kommando: Tilldelning



Tilldelar värden till variabler. Detta innebär att det beräknade värdet på höger sida tilldelas variabeln på vänster sida. Detta kan vara användbart vid komplicerade program.

14.18 Kommando: Om

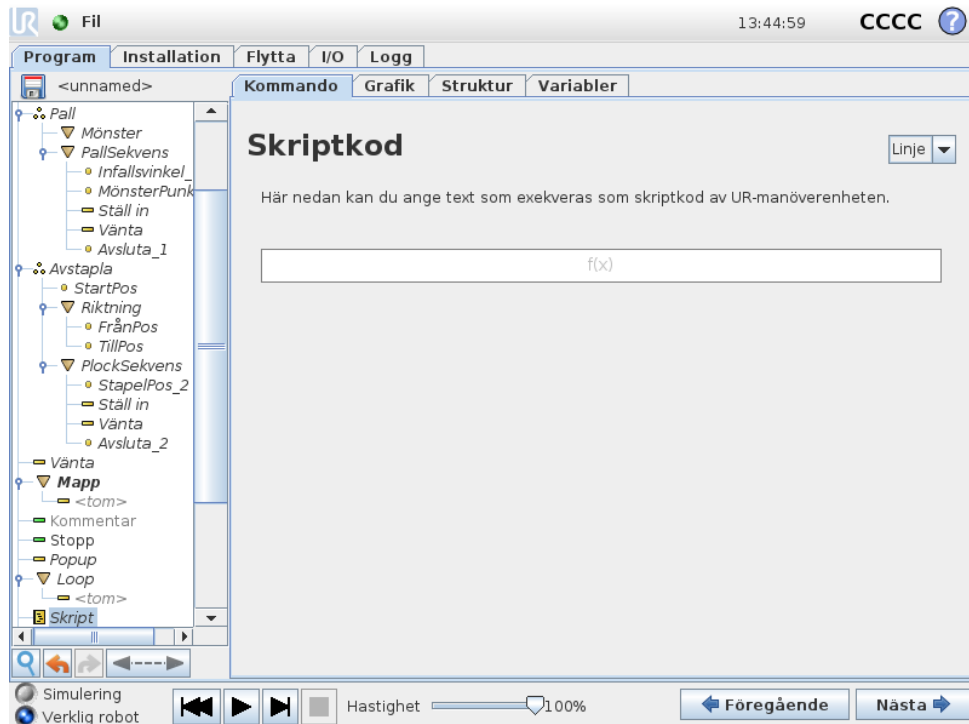


En konstruktion enligt formen **om...** så kan göra att roboten ändrar sitt beteende baserat på insignaler från sensorer eller värden på variabler. Använd **Redigeringsprogrammet för uttryck** för att beskriva under vilka förhållanden roboten ska gå vidare till underkommandona i denna **Om**-sats. Om tillståndet utvärderas till **Sant** körs raderna inom denna **Om**-sats.

Varje **Om** kan innehålla flera kommandosatser av typen **AnnarsOm** och en kommandosats av typen **Annars**. Dessa kan läggas till med knapparna på skärmen. Ett kommando av typen **AnnarsOm** kan tas bort från skärmbilden för det aktuella kommandot.

Den öppna satsen **Kontrollera uttrycket kontinuerligt** möjliggör utvärdering av villkoren i satserna **Om** och **AnnarsOm** medan de ingående raderna exekveras. Om ett uttryck utvärderas som *Falskt* när det befinner sig i huvuddelen av **Om**-delen kommer följande **AnnarsOm**- eller **Annars**-sats att nås.

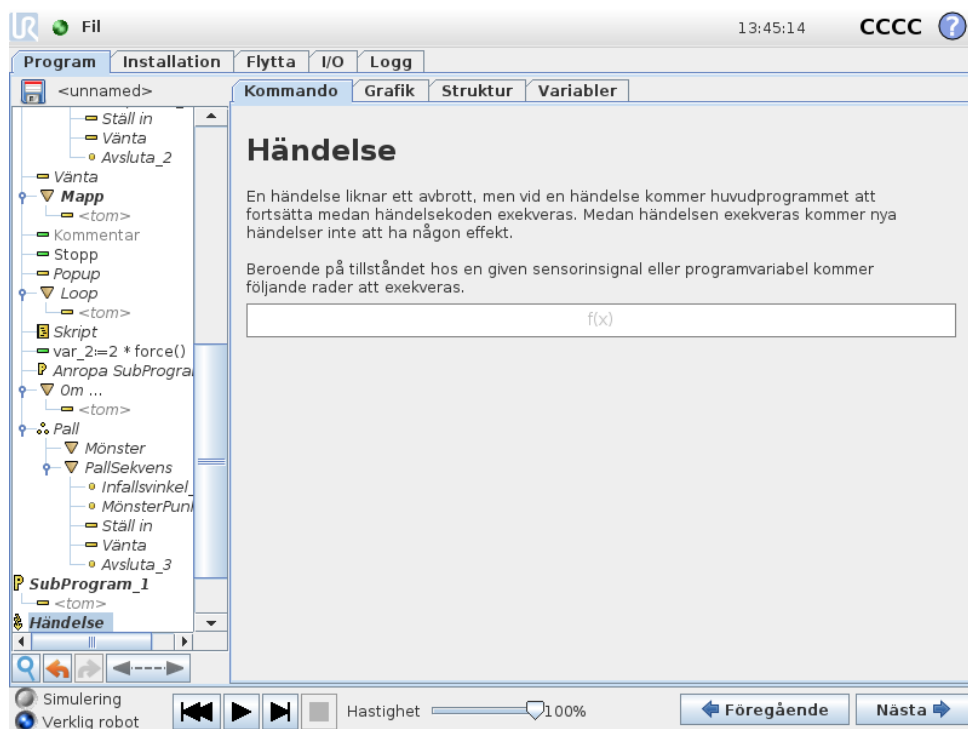
14.19 Kommando: Skript



Det här kommandot ger åtkomst till det underliggande skriptspråket i realtid (språket exekveras av robotens manöverenheter). Alternativet är endast avsett för avancerade användare. Instruktioner om hur det används finns i Skriptmanualen på supportwebbplatsen (<http://www.universal-robots.com/support>).

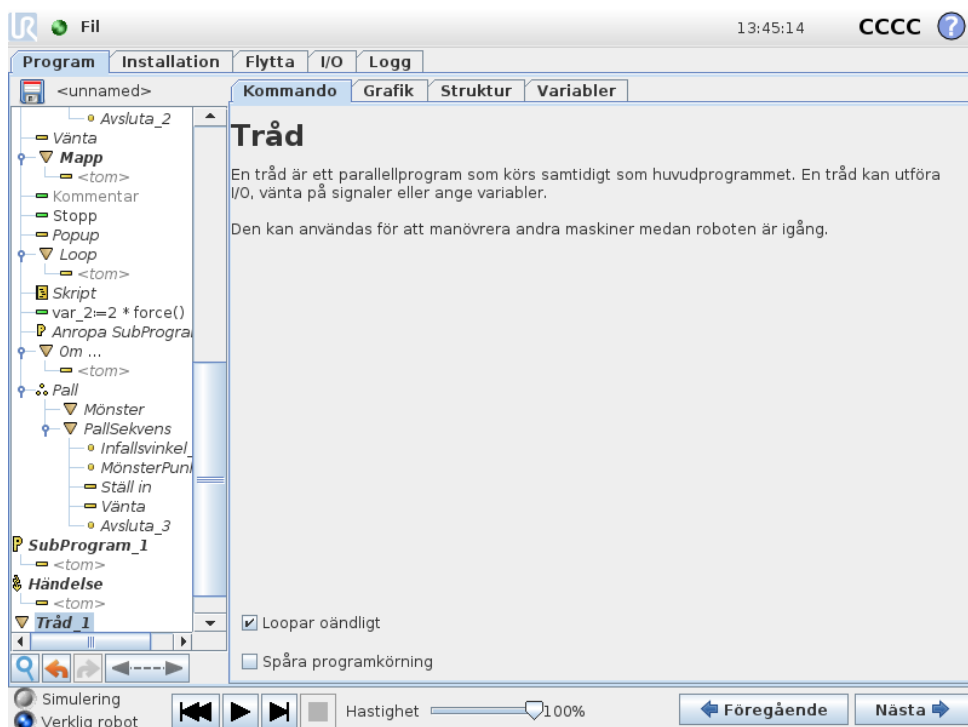
Om alternativet "Arkiv" längst upp till vänster väljs, går det att skapa och redigera skriptprogramfiler. På så sätt kan långa, komplexa skriptprogram användas tillsammans med den operatörsvänliga programmeringen i PolyScope.

14.20 Kommando: Händelse



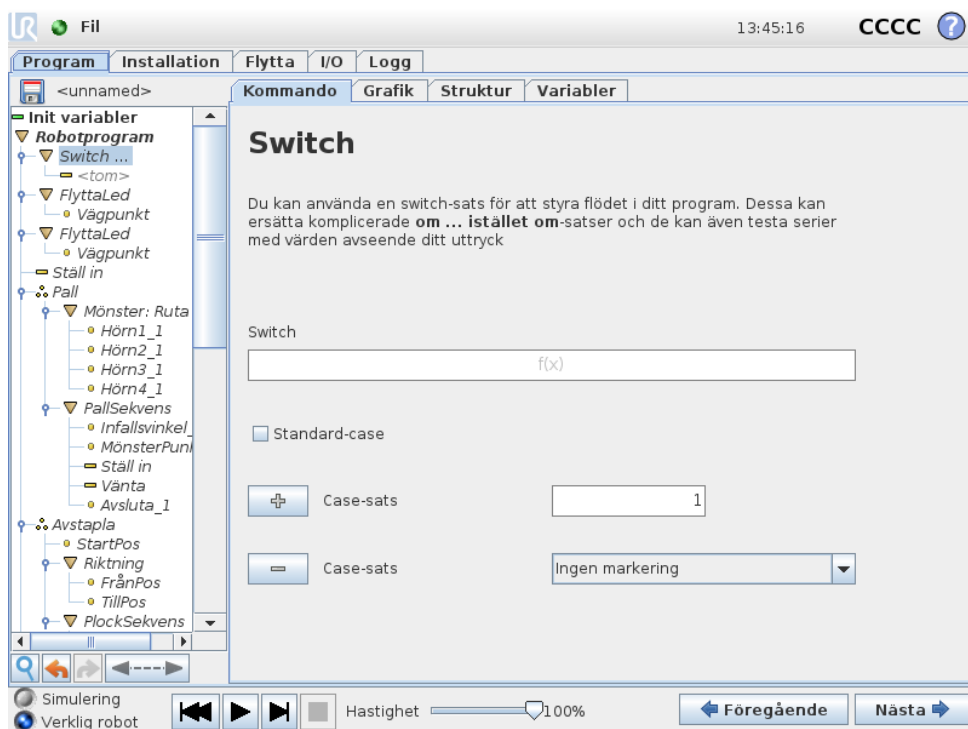
En händelse kan användas för att övervaka en insignal, utföra en åtgärd eller ställa in en variabel när den aktuella insignalen ger ett högt värde. Exempel: om en utsignal ger ett högt värde kan händelseprogrammet vänta 200 ms och därefter återställa värdet till ett lågt värde igen. Detta kan förenkla huvudprogrammets kod betydligt om en extern maskin triggar på en stigande flank i stället för på en hög insignalnivå. Händelserna kontrolleras en gång per styr-/reglercykel (8 ms).

14.21 Kommando: Tråd



En tråd är en parallellprocess till robotprogrammet. En tråd kan användas för att styra en extern maskin oberoende av robotarmen. En tråd kan kommunicera med robotprogrammet med variabler och utsignaler.

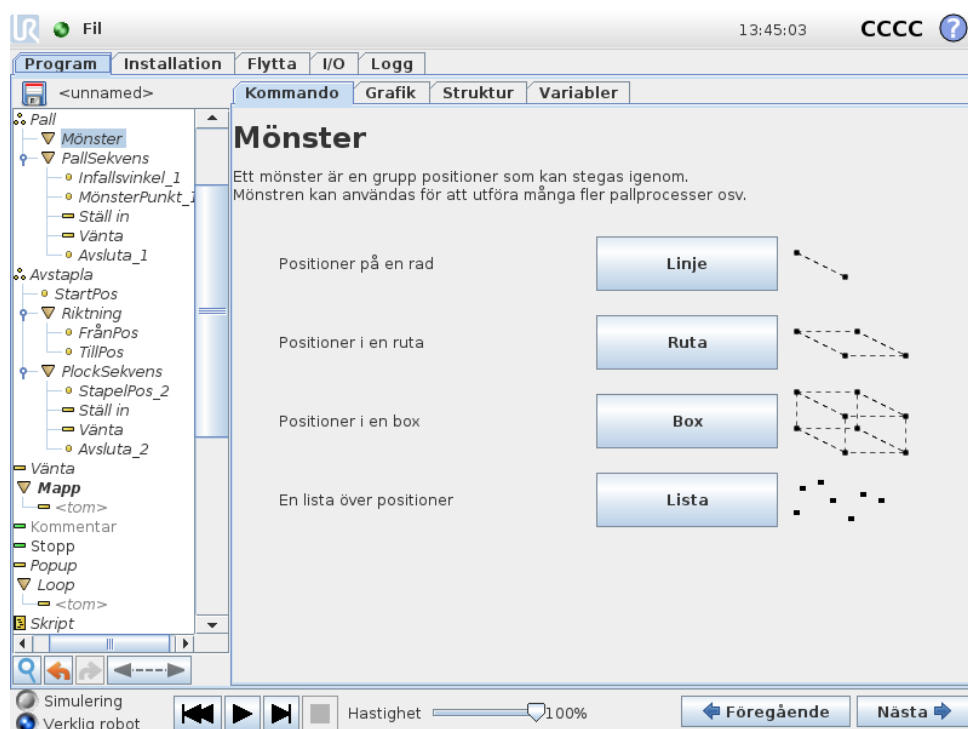
14.22 Kommando: Switch



En **Switch Case**-konstruktion kan göra att roboten ändrar beteende baserat på insignaler från sensorer eller variabelvärden. Använd **Uttrycksredigeraren** för att beskriva bastillståndet och definiera de fall då roboten ska gå vidare till underkommandona för denna Switch. Om tillståndet utvärderas och stämmer överens med något av fallet körs raderna inom case-satsen. Om en Default Case-sats har specificerats körs raderna bara om inga andra matchande case-tillstånd hittades.

Varje Switch kan ha flera Cases och ett Förvalt Case. Switch-satser kan bara ha en instans av något av de Case-värden som har definierats. Case-satser kan läggas till med hjälp av knapparna på skärmen. Ett case-kommando kan tas bort från skärmbilden för den aktuella switch-satsen.

14.23 Kommando: Mönster



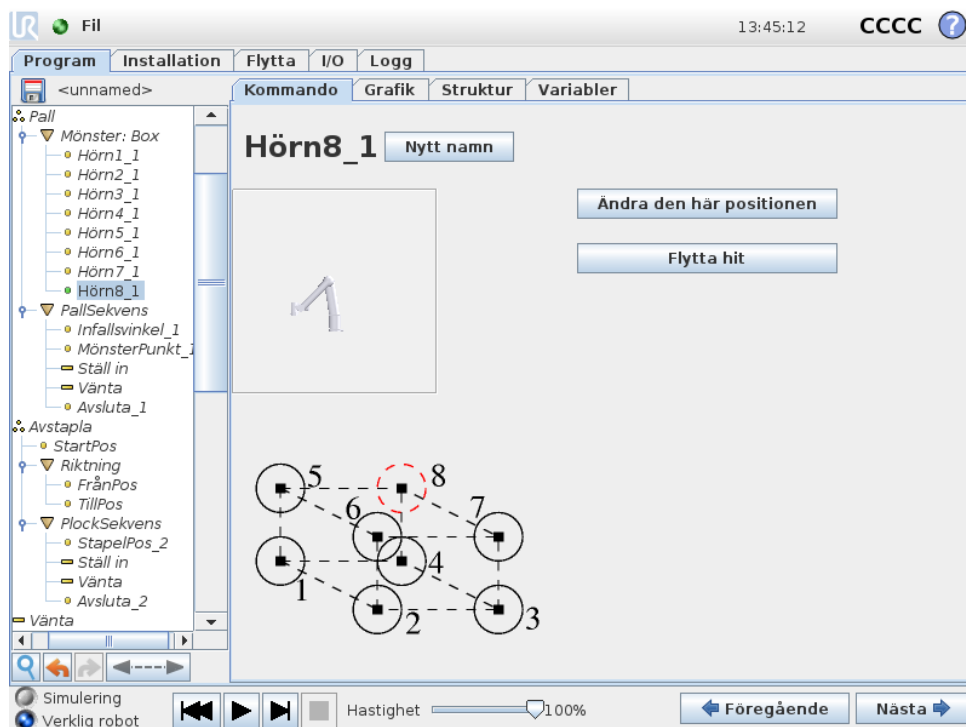
Kommandot **Mönster** kan användas för att stega igenom positionerna i robotprogrammet. Kommandot **Mönster** motsvarar en position vid varje exekvering.

Det finns fyra sorters mönster. De första tre, **Linje**, **Ruta** och **Box**, kan användas för positioner i ett regelbundet mönster. De regelbundna mönstren definieras av ett antal karakteristiska punkter, där punkterna definierar mönstrets kanter. För **Linje** är detta de två ändpunkterna, för **Ruta** är detta tre av de fyra hörnpunkterna och för **Box** är detta fyra av de åtta hörnpunkterna. Programmeraren skriver in antalet positioner längs var och en av kanterna på mönstret. Robotens styrenhet beräknar sedan de enskilda mönsterpositionerna genom att lägga ihop kantvektorena proportionellt.

Om positionerna som ska passeras inte ligger längs ett regelbundet mönster kan du använda alternativet **Lista**, där en lista över alla positioner ges av programmeraren. På så sätt kan alla typer av positionsarrangemang skapas.

Definiera mönstret

När mönstret **Box** valts växlar skärmbilden till den som visas nedan.



Ett **Box**mönster använder tre vektorer för att definiera boxens sidor. Dessa tre vektorer ges som fyra punkter, där den första vektorn går från punkt 1 till punkt 2, den andra vektorn går från punkt 2 till punkt 3 och den tredje vektorn går från punkt 3 till punkt 4. Varje vektor divideras med värdena från intervallräkningen. En specifik position i mönstret beräknas genom att lägga till intervallvektorena proportionellt.

Mönstren **Linje** och **Ruta** fungerar på liknande sätt.

En räknarvariabel används medan systemet förflyttar sig mellan mönstrets positioner. Variabelns namn visas på kommandoskrimen **Mönster**. Variabeln stegar igenom värdena från 0 till $X * Y * Z - 1$, antalet punkter i mönstret. Den här variabeln kan påverkas med hjälp av tilldelningar, och kan användas i uttryck.

14.24 Kommando: Kraft

Inom robotens arbetsområde möjliggör **Kraftläge** eftergivelse och kraft i valda axlar. Alla robotarmsrörelser under kommandot **Kraft** är i **Kraftläge**. När robotarmen rör sig i **Kraftläge** går det att välja en eller flera axlar där robotarmen ska vara eftergivlig. Robotarmen är eftergivlig i förhållande till omgivningen i enlighet med en eftergivlighetsaxel. Det här innebär att robotarmen justerar sin position automatiskt för att åstadkomma vald kraft. Det går även att göra så att robotarmen själv applicerar en kraft på sin omgivning, t.ex. ett arbetsstycke.

Kraftläge är lämpligt för tillämpningar där den faktiska TCP-positionen längs en fördefinierad axel inte är viktig, men där istället en viss kraft ska användas längs

aktuell axel. Om robotens TCP t.ex. ska rulla mot en böjd yta eller trycka eller dra tillbaka ett arbetsstycke. **Kraftläge** har också stöd för att applicera vissa vridmoment runt fördefinierade axlar.

Observera att om robotarmen inte möter något motstånd längs en axel där en kraft som inte är noll är inställd, försöker robotarmen accelerera längs axeln.

Även om en axel har angetts vara eftergivlig försöker robotprogrammet ändå flytta roboten längs aktuell axel. Kraftkontrollen säkerställer dock att robotarmen ändå närmar sig den angivna kraften.


VARNING:

1. Undvik kraftig acceleration precis innan kraftläget aktiveras.
2. Undvik kraftig acceleration i kraftläge, eftersom detta minskar noggrannheten i kraftkontrollen.
3. Undvik rörelser parallellt med eftergivliga axlar innan kraftläget aktiveras.



Funktionsval

Funktionsmenyn används för att välja vilket koordinatsystem (axlar) roboten ska använda när den arbetar i kraftläge. Funktionerna i menyn är de som definierats i installationen (se 13.12).

Kraftlägestyp

Det finns fyra olika typer av kraftläge, som avgör hur den valda funktionen ska tolkas.

- **Enkelt:** Endast en axel är fjädringsmjuk i kraftläge. Kraften längs denna axel går att justera. Den önskade kraften appliceras alltid längs z-axeln för den valda funktionen. För Linje-funktioner appliceras den dock längs y-axeln.
- **Ram:** Ram-typen tillåter mer avancerad användning. Här går det att välja fjädringsmjukhet och krafter oberoende i alla sex frihetsgraderna.
- **Punkt:** När Punkt är vald pekar y-axeln i uppgiftens koordinatsystem från robotens TCP mot den valda funktionens origo. Avståndet mellan robotens TCP och den valda funktionens origo måste vara minst 10 mm. Observera att uppgiftens koordinatsystem ändras vid körning allteftersom robotens TCP ändras. Uppgiftskoordinatsystemets x- och z-axlar beror på den valda funktionens ursprungliga orientering.
- **Rörelse:** Rörelse betyder att uppgiftens koordinatsystem ändras med TCP-rörelseriktningen. Uppgiftskoordinatsystemets x-axel blir projektionen av TCP-rörelseriktningen på det plan som spänns upp av den valda funktionens x- och y-axel. Y-axeln är vinkelrät mot robotarmens rörelse, och i den valda funktionens x-y-plan. Det här kan vara användbart vid avgradning längs en komplex bana där det krävs en kraft vinkelrätt mot TCP-rörelsen.
Observera att när robotarmen inte rör sig gäller följande: Om kraftläget startas när robotarmen står still finns det inga fjädringsmjuka axlar förrän TCP-hastigheten är större än noll. Om robotarmen senare står stilla, medan den fortfarande är i kraftläge, har koordinatsystemet för uppgiften samma orientering som senast TCP-hastigheten var större än noll.

För de sista tre typerna kan det faktiska koordinatsystemet för uppgiften visas vid körning under fliken Grafik (se 14.29) när roboten arbetar i kraftläge.

Val av kraftvärde

- Kraft- och momentvärden kan ställas in för eftergivliga axlar. Robotarmen justerar då sitt läge för att åstadkomma vald kraft.
- För axlar som inte är eftergivliga följer robotarmen banan som angetts av programmet.

För translationsparametrar anges kraften i Newton [N] och för rotation anges vridmomentet i Newtonmeter [Nm].



OBS:

Gör följande:

- Använd skriptet `get_tcp_force()` i en separat tråd för att minska den faktiska kraften och vridmomentet.
- Korrigera nyckelvektorn om den faktiska kraften och/eller vridmomentet är mindre än begärt.

Val av gränser

Ett gränsvärde kan anges för alla axlar, men effekterna skiljer sig beroende på om axlarna är eftergivliga eller inte.

- **Fjädringsmjuk:** Gränsen är den högsta hastighet som TCP tillåts uppnå längs (eller runt) axeln. Enheterna är [mm/s] och [grader/s].
- **Icke fjädrande:** Gränsen är den maximala avvikelse från den programmerade rörelsebanan som tillåts innan roboten skyddsstoppas. Enheterna är [mm] och [grader].

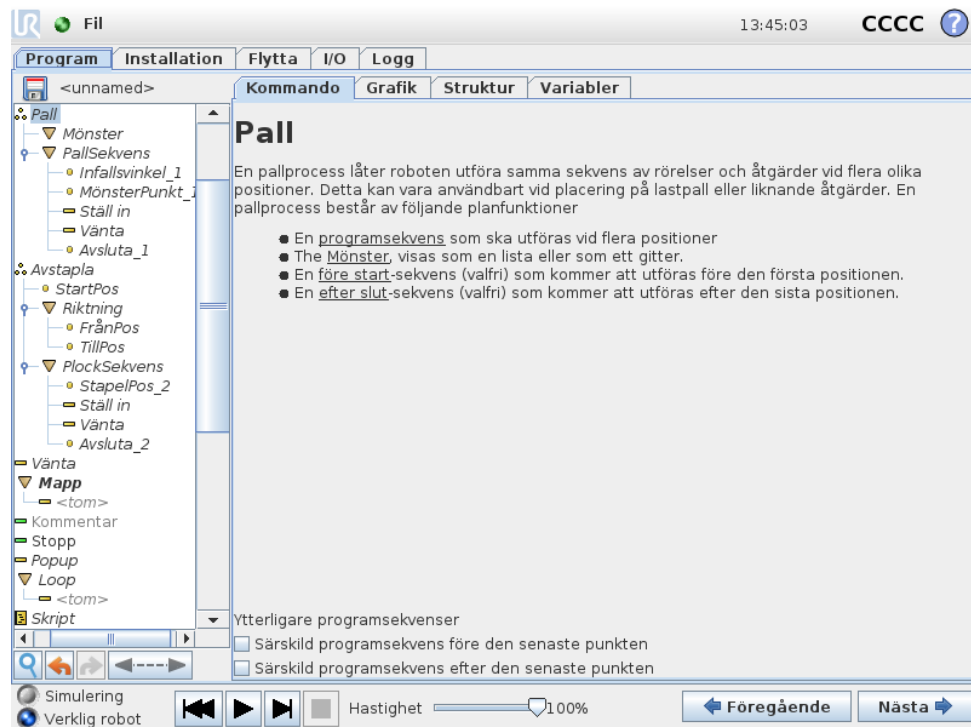
Testa kraftinställningarna

På/av-knappen märkt **Test** växlar funktionen för knappen **Frikörning** på baksidan av manöverenheten från normalt frikörningsläge till testning av kraftkommandot.

När **testknappen** är aktiverad och knappen **Frikörning** på baksidan av manöverenheten trycks in kommer roboten att uppföra sig som om programmet hade nått det här kraftkommandot. På så sätt kan inställningarna verifieras innan hela programmet körs. Detta är speciellt användbart för att verifiera att fjädringsmjuka axlar och krafter har valts på rätt sätt. Håll bara i robotens TCP med en hand och tryck på knappen **Frikörning** med den andra, och notera i vilka riktningar robotarmen kan/inte kan förflyttas. När du lämnar den här skärmbilden avaktiveras knappen **Test** automatiskt, vilket betyder att knappen **Frikörning** på baksidan av manöverenheten används för normalt **frikörningsläge** igen.

Obs: Knappen **Frikörning** är bara aktiv när en giltig funktion har valts för kommandot Kraft.

14.25 Kommando: Pall



En pallåtgärd kan utföra en rörelsesekvens på ett visst antal platser som beskrivs som ett mönster (se 14.23). Rörelsesekvensen kommer att köras i förhållande till mönsterpositionen vid varje position i mönstret.

Programmera en pallprocess

Följande steg ska finnas med:

1. Definiera mönstret.
2. Skapa en **PallSekvens** för plockning/placering vid varje enskild punkt. Sekvensen beskriver vad som ska göras vid varje mönsterposition.
3. Använd väljaren på skärmbilden för sekvenskommandot för att definiera vilken av vägpunkterna i sekvensen som ska motsvara mönsterpositionerna.

Pallsekvens/förankringsbar sekvens

I en **Pallsekvens**-nod är robotarmens rörelser relativa i förhållande till pallens position. Sekvensen fungerar så att robotarmen kommer att befinna sig vid den position som specificeras av mönstret vid **Ankarposition/Mönsterpunkt**. Samtliga återstående positioner kommer att flyttas och anpassas efter detta.

Använd inte kommandot **Flytta** inuti en sekvens, eftersom det inte har någon relation till ankarpositionen.

“FöreStart”

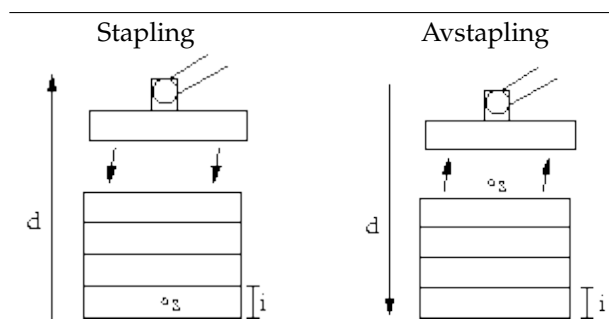
Den valfria sekvensen **Före start** körs precis innan processen startas. Denna kan användas för att vänta på klarsignaler.

“EfterSlut”

Den valfria sekvensen **Efter slut** körs när processen är klar. Denna kan användas för att signalera att ett transportband ska starta, och förbereda för nästa pall.

14.26 Kommando: Sök

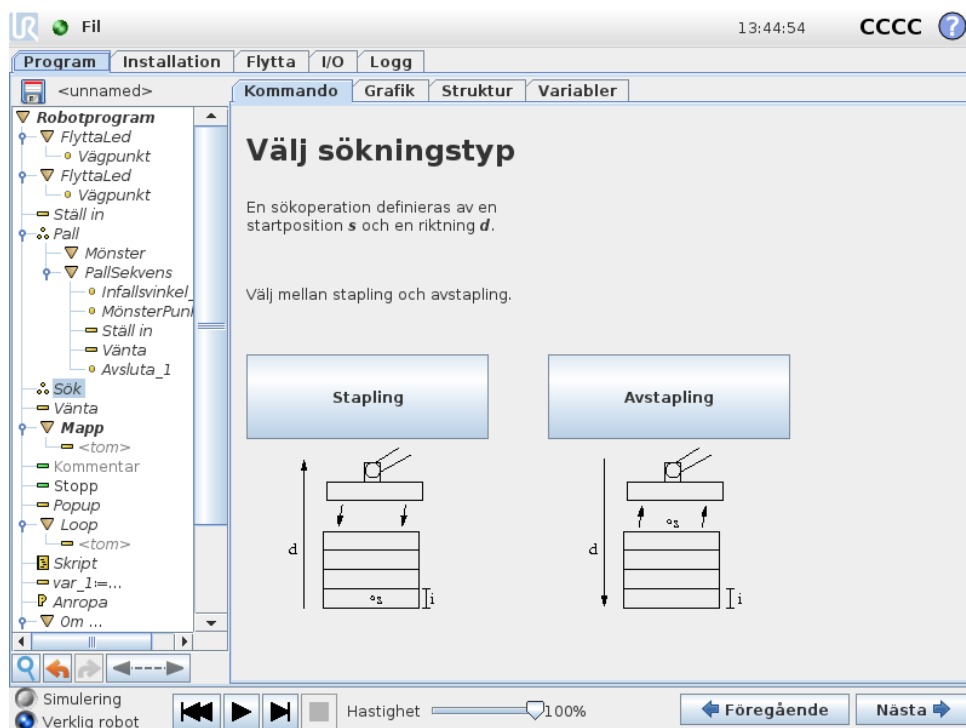
En sökfunktion använder en sensor för att fastställa när rätt position nåtts för att gripa eller släppa en artikel. Sensorn kan vara en tryckknappsomkopplare, en trycksensor eller en kapacitiv sensor. Funktionen är avsedd för arbete på artikelstaplar med varierande artikeltjocklek, eller där artiklarnas exakta positioner inte är kända eller svåra att programmera.



Vid programmering av en sökåtgärd för arbete på en stapel måste operatören definiera s – startpunkten, d – stapelns riktning och i – tjockleken på artiklarna i stapeln.

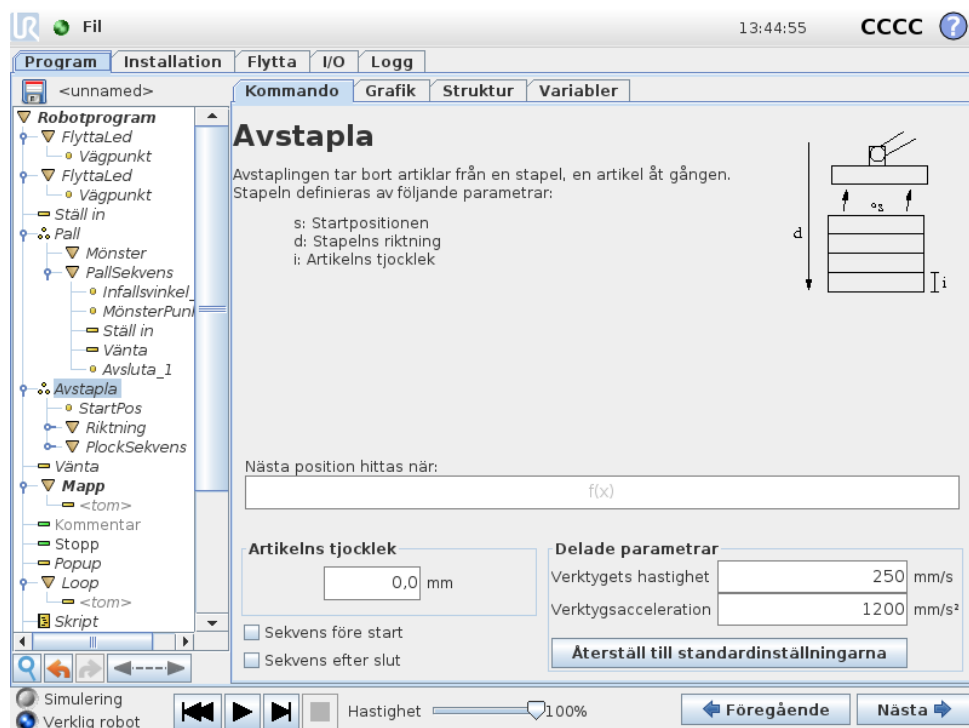
Dessutom måste operatören definiera villkoren för när nästa stapelposition uppnåtts, och en särskild programsekvens som ska utföras vid var och en av stapelpositionerna. Hastighet och accelerationer måste också anges för rörelsen som avser staplingen.

Stapling



Vid stapling rör sig robotarmen till startpositionen, och därefter går den vidare i *motsatt* riktning för att söka efter nästa stapelposition. När positionen hittas kommer roboten ihåg positionen och utför den särskilda sekvensen. Nästa gång startar roboten sökningen från den memorerade positionen, plus den artikeltjocklek som programmerats längs riktningen. Staplingen slutförs när stapelns höjd överstiger ett definierat värde eller när en sensor ger en signal.

Avstapling

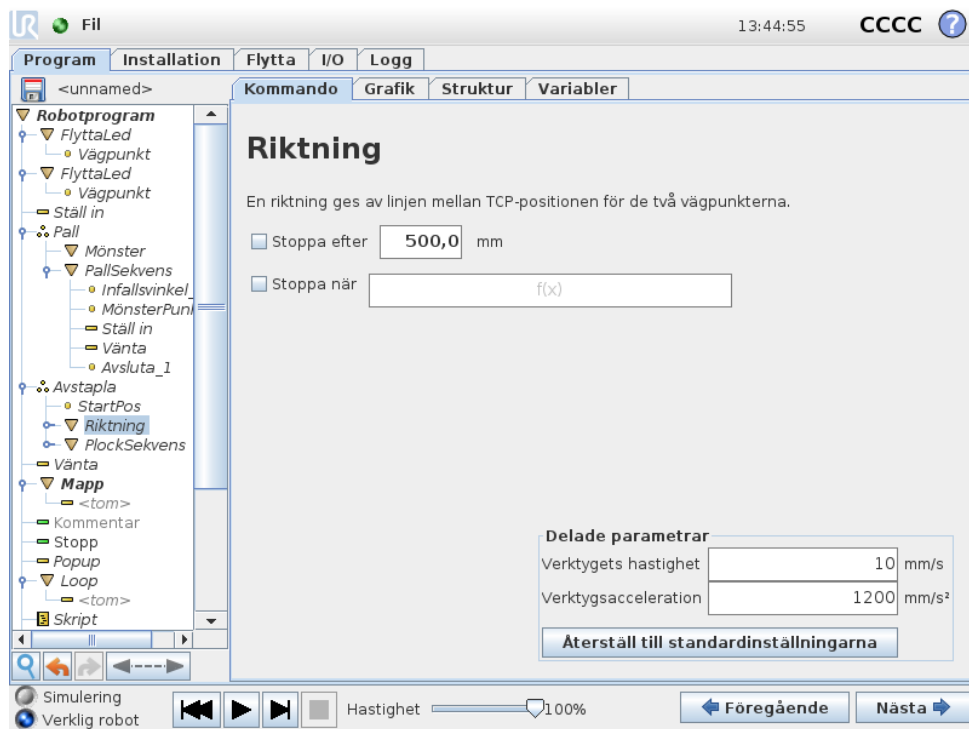


Vid avstapling går robotarmen från startpositionen i den angivna riktningen och söker efter nästa artikel. Villkoret på skärmen avgör när nästa artikel nås. När villkoret uppfylls kommer roboten ihåg positionen och utför den särskilda sekvensen. Nästa gång startar roboten sökningen från den memorerade positionen, plus den artikeltjocklek som programmerats längs riktningen.

Startposition

Startpositionen är där staplingen startar. Om startpositionen utelämnas startar staplingen vid robotarmens befintliga position.

Riktning



Riktningen anges av två positioner, och beräknas som positionsskillnaden mellan den första positionens TCP till den andra positionens TCP.

Obs: En riktning tar inte hänsyn till punkternas orientering.

Uttryck för nästa staplingsposition

Robotarmen rör sig längs riktningsektorn och utvärderar kontinuerligt om nästa stapelposition har nåtts. När uttrycket är `Sant` exekveras specialekvensen.

“FöreStart”

Den valfria sekvensen `FöreStart` körs precis innan processen startas. Denna kan användas för att vänta på klarsignaler.

“EfterSlut”

Den valfria sekvensen `EfterSlut` körs när processen är klar. Denna kan användas för att signalera att ett transportband ska starta, och förbereda för nästa stapel.

Sekvens för plockning/placering

Precis som för pallprocessen (14.25) utförs en särskild programsekvens vid varje stapelposition.

14.27 Kommando: Transportbandsspårning

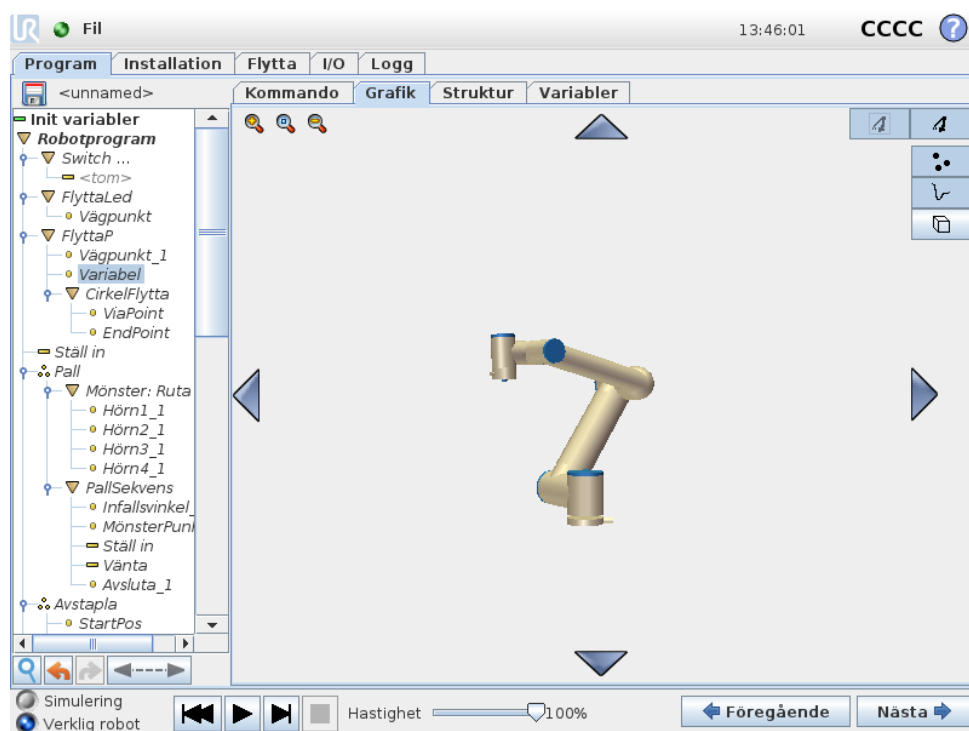
Vid användning av ett transportband kan roboten konfigureras för att spåra dess rörelse. Om `Transportbandsspårning` som är definierad i installationen är rätt kon-

figurerad anpassar roboten sina rörelser så att de följer transportbandet. Program-noden **Transportbandspårning** finns tillgänglig under fliken **Guider** under fliken **Struktur**. Alla rörelser under denna nod är tillåtna under transportbandspårning, men de är relativa till transportbandets rörelse. Inställningen **Transportbandspårning** under fliken **Installation** (se avsnitt 13.13) innehåller alternativ för att konfigurera roboten för arbete med absoluta och inkrementerade kodenheter plus linjära och cirkulära transportband.

14.28 Kommando: Undertryck

Undertryckta programrader hoppas över när programmet körs. En undertryckt rad kan återgå till att vara ej undertryckt vid en senare tidpunkt. Detta är ett snabbt sätt att göra ändringar i ett program utan att förstöra ursprungsinnehållet.

14.29 Grafik, flik



Grafisk representation av det aktuella robotprogrammet. Vägen för TCP visas i 3D-vy, med rörelsesegment i svart och förbigångssegment (övergångar mellan rörelsesegment) i grönt. De gröna punkterna anger positionerna för TCP vid var och en av vägpunkterna i programmet. 3D-ritningen över robotarmen visar robotarmens aktuella position, och *skuggan* av robotarmen visar hur robotarmen avser att nå den vägpunkt som valts till vänster på skärmen.

Om den aktuella positionen för robotens TCP kommer nära ett säkerhets- eller utlösningssplan, eller om robotverktygets orientering är nära verktygets orienteringsbegränsning (se 10.12), visas en 3D-representation av närhetsgränsen. Observera att när roboten kör ett program är visualiseringen av gränsvärden inaktiverad.

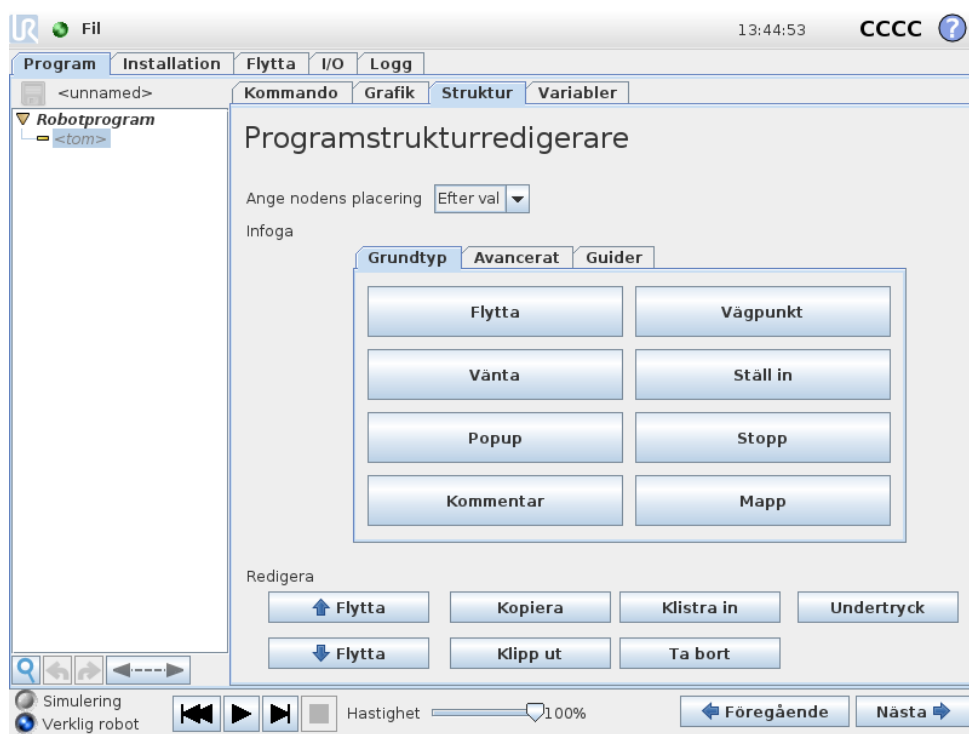
Säkerhetsplan visas i gult och svart med en liten pil som representerar normalen mot planet, vilket anger den sida på planet där robotens TCP får placeras. Utlösarplan visas i blått och grönt med en liten pil som pekar på planets sida, där gränserna **Normalt** läge (se 10.6) är aktiva. Verktygsorienteringens gränsvärde visas som en sfärisk kon tillsammans med en pil som anger robotverktygets nuvarande orientering. Konens insida representerar verktygsorienteringens tillåtna yta (vektor).

När målet för robotens TCP inte längre är i närheten av gränsvärdet försvinner 3D-representationen. Om TCP överträder eller är mycket nära att överträda ett gränsvärde visas gränsvärdets visualisering som röd.

Du kan zooma och rotera 3D-vyn för att få en bättre överblick över robotarmen. Knapparna på skärmens övre högra sida kan stänga av de olika grafiska komponenterna i 3D-vyn. Den nedre knappen växlar visningen av gränsvärdena för närhetsgränsen.

Vilka rörelsesegment som visas beror på den valda programnoden. Om du valt en **Flytta**-nod visas den rörelse som definierats av flyttparametern. Om en nod av typen **Vägpunkt** väljs, visas följande ~ 10 steg i rörelsen på skärmen.

14.30 Struktur, flik



På fliken Struktur under Program kan du infoga, flytta, kopiera och ta bort de olika typerna av kommandon.

Utför följande steg för att infoga nya kommandon:

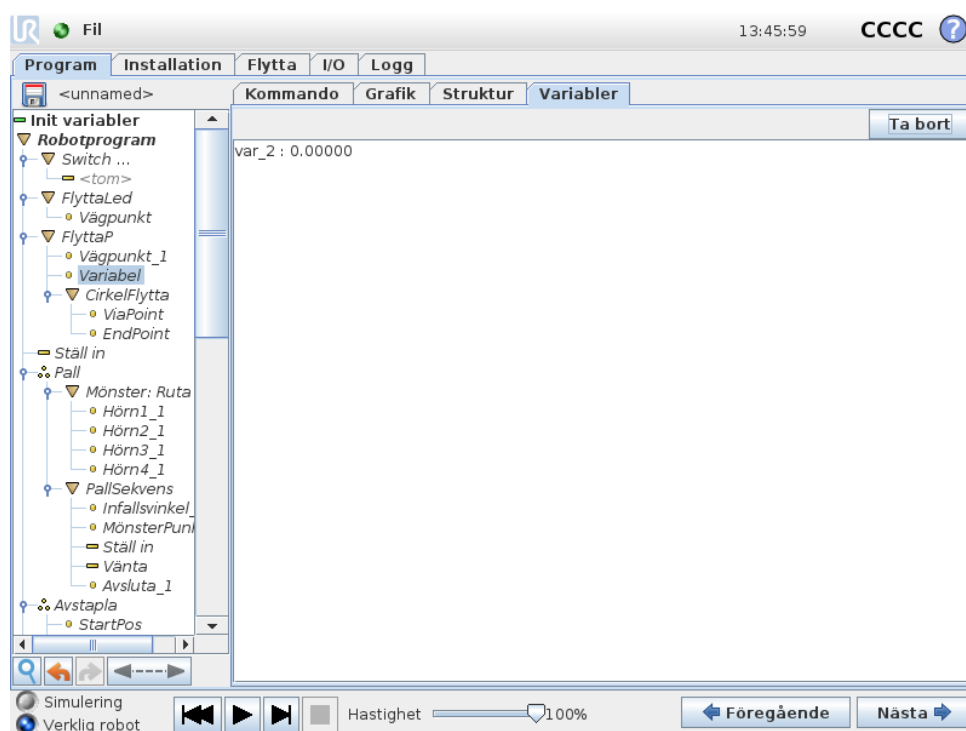
- 1) Markera ett befintligt programkommando.
- 2) Välj om det nya kommandot ska infogas ovanför eller nedanför det valda kommandot.

- 3) Tryck på knappen för den typ av kommando som du vill infoga. Du kan justera uppgifterna för det nya kommandot på fliken **Kommando**.

Kommandon kan flyttas/klonas/tas bort med knapparna i redigeringsrutan. Om ett kommando har underordnade kommandon (en triangel bredvid kommandot) kommer alla underordnade kommandon också att flyttas/klonas/tas bort.

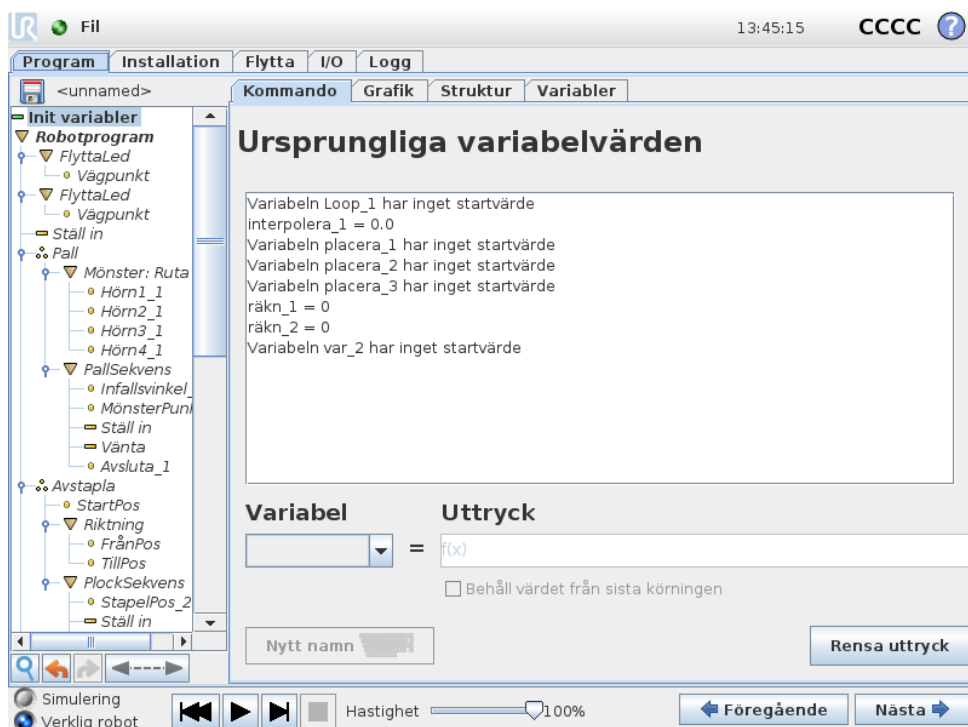
Vissa kommandon kan inte infogas var som helst i ett program. Vägpunkter måste ligga under kommandot **Flytta** (inte nödvändigtvis direkt under). Annars **Om** och Annars måste placeras efter kommandot **Om**. Det kan vara komplicerat när kommandon av typen **AnnarsOm** ska flyttas runt. Variabler måste tilldelas värden innan de används.

14.31 Variabler, flik



Fliken **Variabler** visar realtidsvärden för de variabler som körs i programmet, och sparar en lista över variabler och värden mellan programkörningar. Den visas bara när det finns information att visa. Variablerna är sorterade alfabetiskt på namn. Variabelnamnen på den här skärmbilden visas med högst 50 tecken, och variablernas värden visas med högst 500 tecken.

14.32 Kommando: Initiera variabler



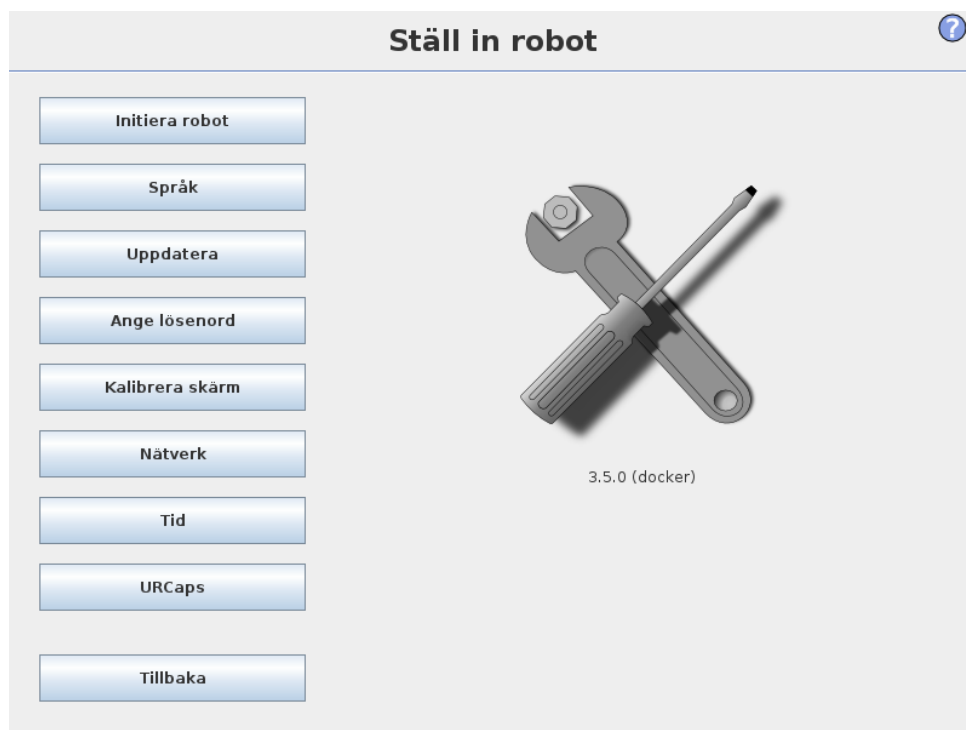
På den här skärmbilden kan du ställa in värden för variablerna innan programmet (eller några trådar) börjar exekveras.

Välj en variabel från listan över variabler genom att klicka på den eller genom att använda väljarrutan för variabler. När du valt en variabel kan du ange ett uttryck som kommer att användas för att ställa in variabelns värde när programmet startas.

Om du har markerat kryssrutan “Behåller helst värdet från sista körningen” kommer variabeln att initieras till det värde som står på fliken *Variabler* och som beskrivs i avsnittet 14.31. På så sätt kan variabler behålla sina värden mellan programkörningar. Variabeln får sitt värde från uttrycket om programmet körs för första gången eller om fliken med värdet har raderats.

En variabel kan tas bort från programmet genom att lämna fältet för dess namn tomt (endast blanksteg).

15 Skärmbilden Inställning



- **Initiera robot** Går till initieringsskärbilden. Se 11.5.
- **Språk och enheter** Konfigurerar språk och måttenheter för användargränssnittet. Se 15.1.
- **Uppdatera robot** Uppgraderar robotens programvara till en senare version. Se 15.2.
- **Ange lösenord** Ger möjlighet att låsa robotens programmeringsdel för personer utan lösenord. Se 15.3.
- **Kalibrera skärmen** Kalibrerar pekskärmens "känslighet". Se 15.4.
- **Ställ in nätverk** Öppnar gränssnittet för inställning av Ethernet-nätverket för robotens manöverskåp. Se 15.5.
- **Ange tid** Ange tid och datum för systemet och konfigurera klockans visningsformat. Se 15.6.
- **Konfiguration av URCaps** Översikt av installerade URCaps och alternativ för installation och avinstallation, se 15.7.
- **Bakåt** Återgår till välkomstskrmen.

15.1 Språk och enheter

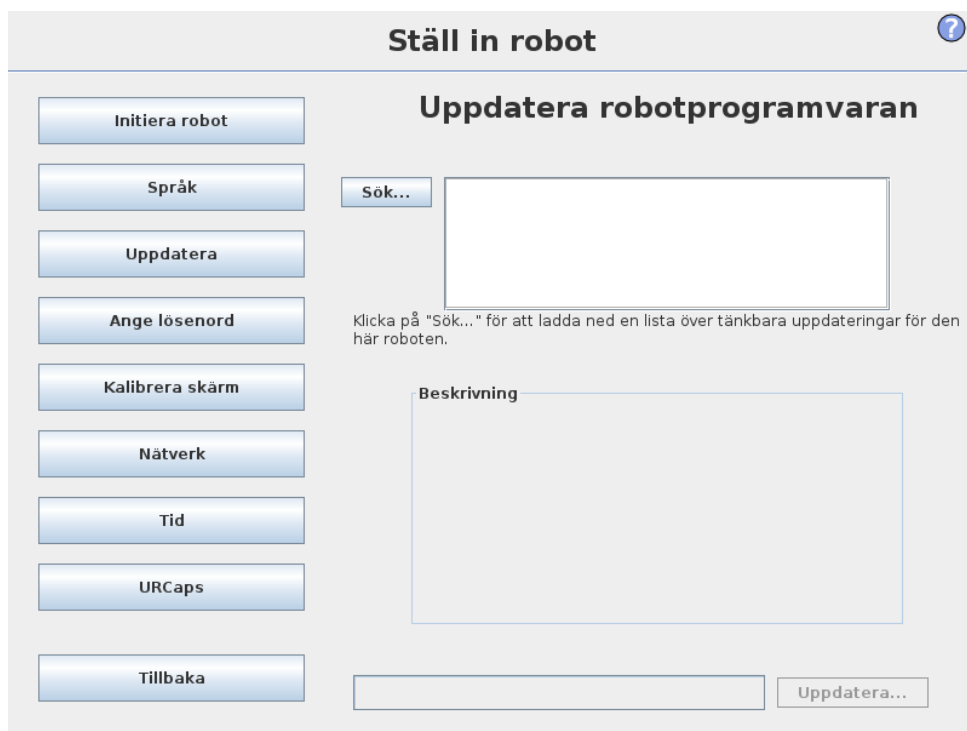


På den här skärmen går det att välja det språk och de enheter som används i PolyScope.

Det språk som väljs används både för den text som visas på skärmarna i PolyScope och för den inbäddade hjälpen. Markera "Engelsk programmering" om du vill att kommandonamnen i robotprogram ska skrivas på engelska. PolyScope måste startas om för att ändringarna ska träda i kraft.

Valt tangentbordsspråk kommer att användas för alla pop up-tangentbord i PolyScope.

15.2 Uppdatera robot



Det går att installera programvaruuppdateringar från ett USB-flashminne. Sätt i ett USB-minne och klicka på **Sök** för att visa dess innehåll. Du uppdaterar genom att välja en fil, klicka på **Uppdatera** och följa instruktionerna på skärmen.



WARNING:

Kontrollera alltid dina program när programvaran uppdaterats. Uppgraderingen kan ändra rörelsebanor i ditt program. Du kan hitta de uppdaterade programvaruspecifikationerna genom att trycka på knappen "?" längst upp till höger i gränssnittet. Maskinvarans specifikationer ändras inte. De finns i den ursprungliga manualen.

15.3 Ange lösenord

Initiera robot

Språk

Uppdatera

Ange lösenord

Kalibrera skärm

Nätverk

Tid

URCaps

Tillbaka

Ställ in robot

Systemlösenord

Nytt lösenord

Bekräfta det nya lösenordet

Använd

Säkerhetslösenord

Aktuellt lösenord

Nytt lösenord

Bekräfta det nya lösenordet

Använd

Två lösenord stöds. Det första är ett *valfritt* systemlösenord som hindrar icke auktoriserad ändring av robotens inställningar. När ett systemlösenord angetts kan programmen laddas och köras utan lösenordet, men användaren måste ange rätt lösenord för att skapa eller ändra i program.

Det andra är ett *obligatoriskt* säkerhetslösenord, som måste anges korrekt för att kunna ändra säkerhetskfigurationen.



OBS:

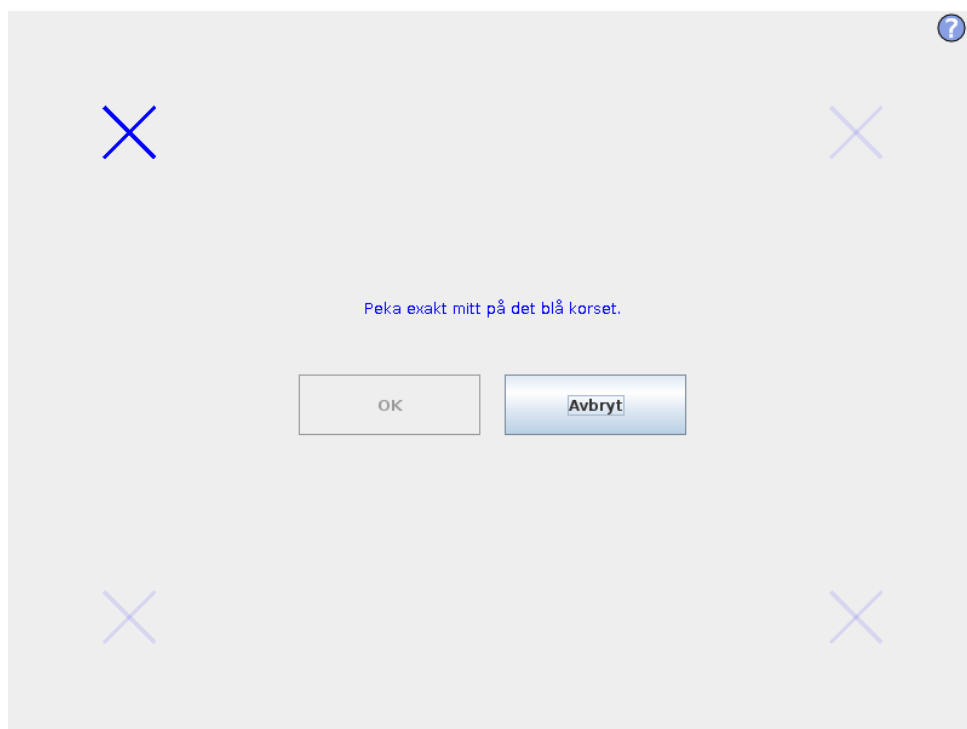
Säkerhetslösenordet måste ha ställts in för att det ska gå att ändra säkerhetskfigurationen.



VARNING:

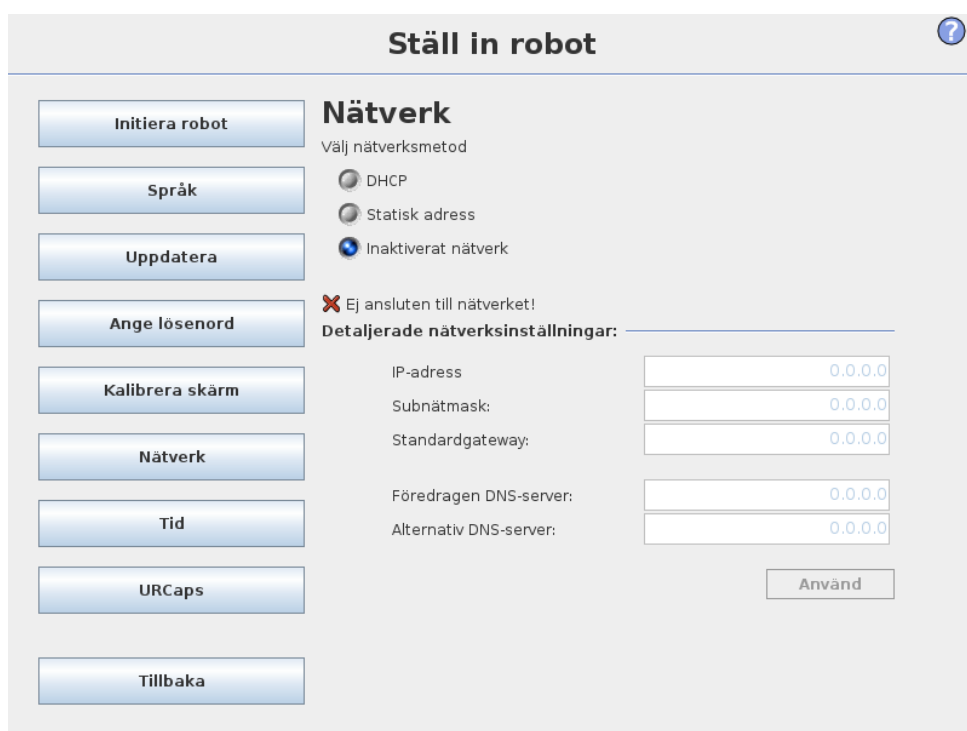
Du lägger till ett systemlösenord om du vill hindra icke auktoriserad personal att ändra robotinstallationen.

15.4 Kalibrera skärm



Kalibrera pekskärmen. Följ anvisningarna på skärmen för att kalibrera pekskärmen. Använd helst ett föremål som inte innehåller metall, t.ex. en penna med indragen spets. Tålmod och försiktighet bidrar till ett bättre resultat.

15.5 Ställ in nätverk



Panel för inställning av Ethernet-nätverket. En Ethernet-anslutning är inte nödvändig för de grundläggande robotfunktionerna, och denna är inaktiverad i standardläget.

15.6 Ange tid

Initiera robot

Språk

Uppdatera

Ange lösenord

Kalibrera skärm

Nätverk

Tid

URCaps

Tillbaka

Ställ in robot

Ange tid

Tidsformat:
☒ 24-timmars
☐ 12-timmars

Ange aktuell tid:

+

+

+

13

:

44

:

37

-

-

-

Ange datum

Ange dagens datum:

den 8 november 2017

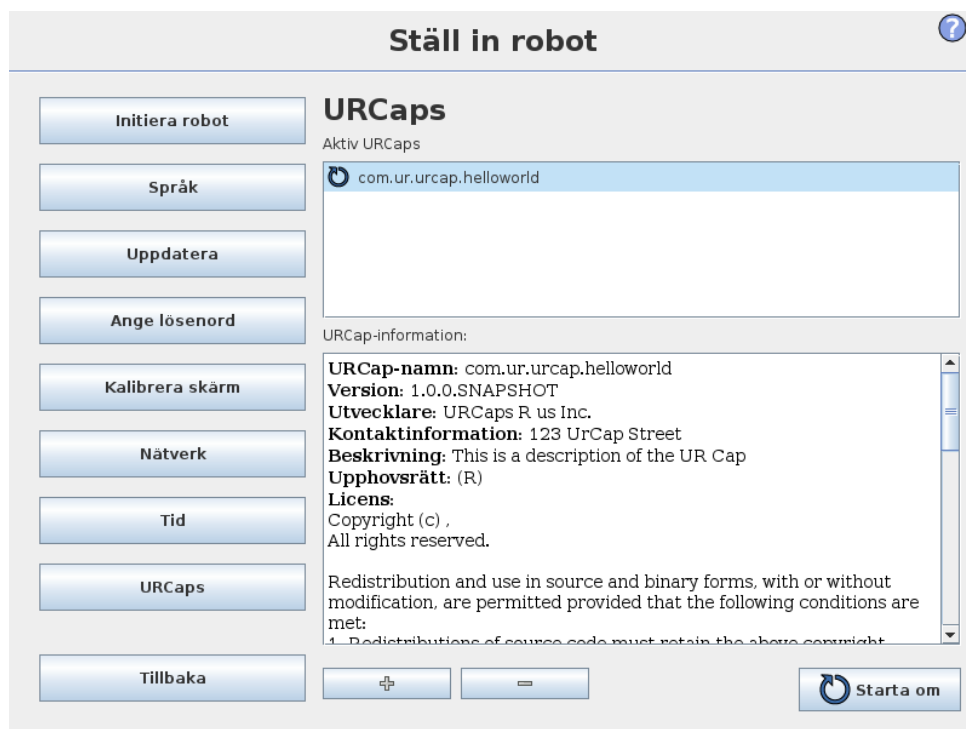
Datumformat:
☒ den 8 november 2017
☐ 2017-nov-08
☐ 2017-11-08

Starta om PolyScope för att använda de nya inställningarna

Starta om

Ange tid och datum för systemet och konfigurera klockans visningsformat. Klockan visas överst på skärmarna *Kör program* och *Programmera robot*. Om du klickar på den visas datumet kortvarigt. Gränssnittet måste startas om för att ändringarna ska träda i kraft.

15.7 Installationsprogram för URCaps



I den övre listan visas en översikt av alla *URCaps*. Klicka på en *URCap* för att se dess meta-information (inklusive dess namn, version, licens osv.) i fältet för *URCap-information* under listan.

Klicka på knappen + längst ned på skärmen för att installera en ny *URCap*. En filväljare visas där en *.urcap*-fil kan väljas. Klicka Öppna och PolyScope återgår till inställningsskärmen. Den valda *URCap* installeras och motsvarande information visas i listan kort efter. En nyligen installerad eller avinstallerad *URCap* kräver att PolyScope startas om och Omstart-knappen kommer därför att aktiveras.

För att avinstallera en *URCap*, välj denna *URCap* från listan och klicka på --knappen. Denna *URCap* försvinner från listan men en omstart krävs.

Ikonen bredvid en *URCap* i listan anger dess tillstånd. De olika tillstånden definieras nedan:

-  *URCap ok*: Denna *URCap* är installerad och körs normalt.
-  *URCap fel*: Denna *URCap* är installerad men kan inte starta. Kontakta utvecklaren av denna *URCap*.
-  *URCap omstart krävs*: Denna *URCap* har precis installerats men en omstart krävs.