

Tentamen i Robotteknik MPR211, 13 mars 1998

Lärare: Rolf Berlin, ank 1286
Per-Åke Jansson ank 1527

Tillåtna hjälpmedel: Typgodkända kalkylatorer och alla formelsamlingar.
OBS bifogat finns några formler på sista sidan!

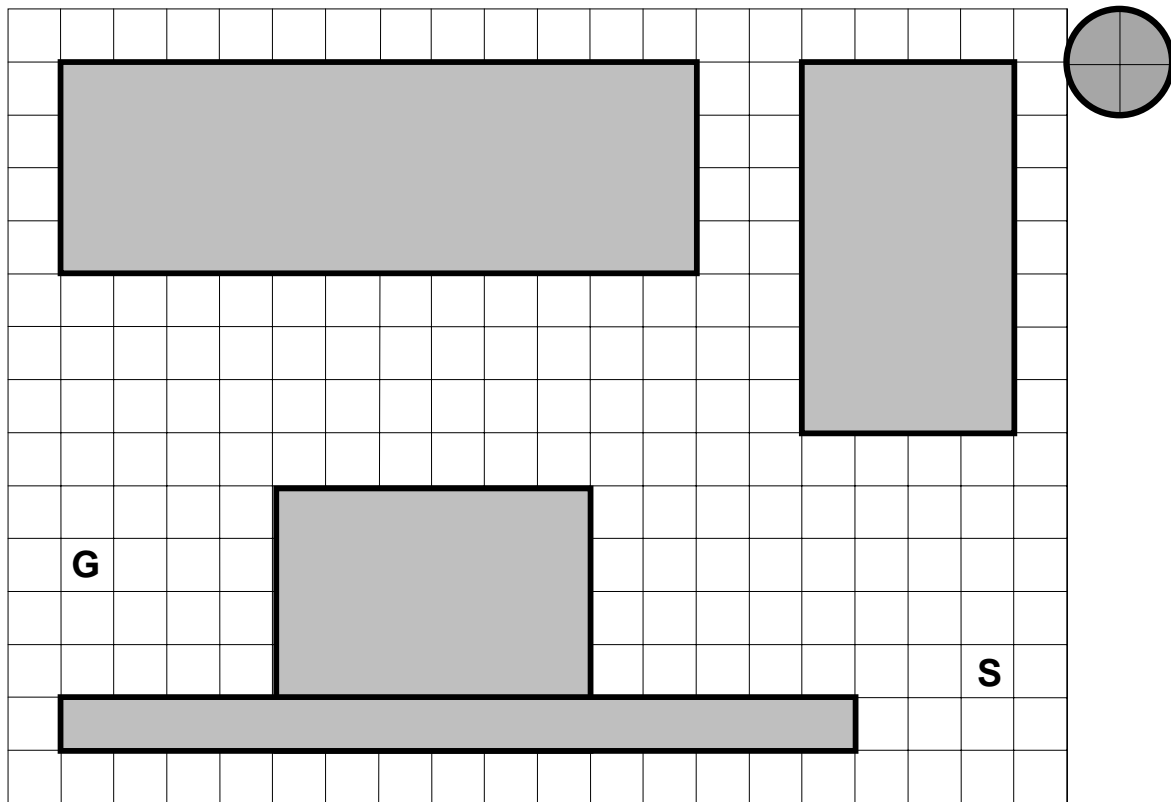
Betygslista anslås fredag 27/3 på Robotlaboratoriets anslagstavla. Granskning sker hos Gunvor Johansson måndag 30/1 kl 13-15 på inst. för prod teknik.

Betygsgränser: 30-39p=betyg 3, 40-49p =betyg 4 50-60p=betyg 5

- 1a Vilken industrisektor har främst utnyttjat industrirobotar och därmed drivit utvecklingen framåt. 1p
- 1b Beskriv tre faser av utvecklingen av bågsvetsning med robot som visar hur mycket utvecklingsarbete som behövs för att robotisera vissa applikationer. 3p
- 1c Beskriv hur en "articulated" robotarm är uppbyggd (typ av länkar och placering i förhållande till varandra). 1p
- 1d Beskriv arbetsvoymen för en "articulated" robotarm. 1p
- 1e Nämn två fördelar och två nackdelar med elektriskt drivna robotar. 4p
- 2a Förklara vad som menas med "error recovery" i samband med robotprogrammering och ge ett exempel. 2p
- 2b Nämn två fördelar med ON-LINE resp OFF-LINE programmering av robotar 2p
- 2c Vad innebär transportabilitet och spegling i samband med robotprogram? 2p
- 2d Hur påverkar man vägval för olika truckar i en AGV-installationer med induktiv slingnavigering? 1p
- 2e Beskriv detaljerat hur en 6-dimensionell kraftgivare är konstruerad, dels mekaniskt, elektriskt och programvarumässig. 3p

- 3a Förklara vad som menas med "reactive multiagent robotic societies" citatet är klippt ur en forskningsrapport om mobila robotar. 2p
- 3b Beskriv skillnaden mellan kartor för mobilla robotar av typen: 3p
- "composite map"
 - "object oriented map"
 - "freespace oriented map"
- 3c **Expandera objekten** med hänsyn till den runda roboten med radie en ruta. Utför därefter en **komplett banplanering** från start **S** till målet **G**. Använd en "8-connected distance transform", och "8-connected path finding algorithm" baserad på verkligt avstånd mellan rutcentrum. Rutorna är 10 enheter i kvadrat. 5p

OBS det finns en kopia av figuren för inlämning längst bak!



4. En gammal robotcell (IBM7545 SCARA), som tidigare användes för montering av duplo- och legofigurer, ska nu montera elektroniska komponenter på kretskort och har därför fått lite modernare kringutrustning:

- 1 st sorteringsmaskin för sortering av två sorters komponenter ("DIP" och "RAM"), vilka placeras i respektive position och där hämtas av roboten. Manuell påfyllning behöver utföras när antalet komponenter endast räcker till ytterligare 5 minuters produktion. Detta indikeras genom att ingång 55 ställs i robotens styrsystem. Om komponenterna tar slut ska utgång 99 ställas (se nedan).
- 1 st 2-positioners vridbord med automatisk tömningsfunktion.
- 1 st matare för kretskort som matar fram kretskorten direkt till vridbordet. Tiden för matningen tar knappt 1 sekund.

Styrsystemet har dessutom uppgraderats för att hantera multi-tasking (parallella processer). Med kommandot IOT_CREATE (se nedan) kan en sk I/O-task definieras för att sedan köras parallellt med huvudprogrammet. När första raden exekveras i huvudprogrammet, som av syntaxskäl ska heta MAIN, startas också körningen av alla I/O-tasks.

Fel i kringutrustning innebär att produktionen ska stoppas genom att ställa utgång 99. Detta innebär stoppad exekvering av huvudprogrammet och alla I/O-tasks samt aktivering av larm för att tillkalla personal.

Uppgift. Implementera de I/O-tasks som behövs för att hantera den nya kringutrustningen. Skriv dessutom ett huvudprogram med lämpliga subrutiner för att kontinuerligt bygga kretskort med de 2 komponenterna ("DIP" och "RAM"). Positionerna finns redan definierade (se nedan).

Gripdonsbyte behöver inte utföras utan roboten använder hela tiden ett permanent gripdon som kan gripa både en "DIP" och en "RAM". Vidare kan antas att roboten från början står i en lämplig beredskapsposition med alla utgångar nollställda, öppet gripdon, tomt vridbord och inga fel har indikerats.

Lösningen presenteras med en programlistning (se programexempel nedan). Skriv tydligt och använd stora bokstäver! För full poäng krävs en fullständig lösning. Avdrag görs för felaktiga sekvenser, brister i funktionalitet - inklusive optimering, logiska fel, strukturella fel, syntaxfel mm. (10p)

Befintliga kommandon och syntax.

Kommando	Beskrivning
PMOVE(POS);	Roboten går till den globalt definierade positionen POS.
ZMOVE(ZPOS);	Roboten rör sig endast i z-led och går till det globalt definierade z-värdet ZPOS.
GRASP;	Gripdonet stängs - ingen fördröjning är nödvändig.
RELEASE;	Gripdonet öppnas - ingen fördröjning är nödvändig.
BRANCH(DEST);	Ovillkorligt hopp i befintlig subrutin till hoppadressen DEST - se programexempel.
TESTI(DI,VAL,DEST);	Villkorligt hopp i befintlig subrutin till hoppadressen DEST. Hoppet sker om ingången med nummer DI i robotens styrskåp = VAL (0 eller 1). Annars exekveras nästa rad.
WAITI(DI,VAL,TID,DEST);	Kommandot innebär att ingången med nummer DI måste anta värdet VAL (0 eller 1) för att nästa rad ska exekveras. Om detta inte inträffar inom väntetiden TID (sekunder)

	sker istället ett hopp till DEST (i befintlig subrutin).
WRITEO(DO,VAL);	Utgången med nummer DO får värdet VAL (0 eller 1).
NAMN(VAR,VAR1..);	Anropar subrutinen med namnet NAMN. Variabler med namnen VAR,VAR1.. blir lokala.
IOT_CREATE(NAMN,<VAR,VAR1..>);	Definierar en I/O-task - se programexempel. Denna utgörs av subrutinen med namnet NAMN, som kan användas för att skapa flera I/O-tasks med olika värden på variablerna VAR,VAR1..
END;	Markerar slutet på subrutin.

Deklarationer. Följande positioner är globalt definierade och kan anropas med kommandona PMOVE, ZMOVE och som variabler till en subrutin:

Deklaration	Beskrivning
UPP	Enda tillåtna z-värde innan roboten förflyttar sig i sidled med PMOVE
DIP	Grovposition för hämtning av en "DIP"
DIPZ	Absolutvärde i z-led för hämtning av en "DIP"
MTDIP	Grovposition för montering av en "DIP"
MTDIPZ	Absolutvärde i z-led för montering av en "DIP"
RAM	Grovposition för hämtning av en "RAM"
RAMZ	Absolutvärde i z-led för hämtning av en "RAM"
MTRAM	Grovposition för montering av en "RAM"
MTRAMZ	Absolutvärde i z-led för montering av en "RAM"

In- och utgångar.

Robotens styrskåp	Funktion
Ingång 1	=1 innebär att vridbordet är positionerat, =0 betyder att vridning pågår
Ingång 2	=1 innebär att ett kretskort ligger korrekt placerat på vridbordets monteringsposition, =0 betyder inget eller felplacerat kretskort
Ingång 55	=1 innebär att manuell påfyllning av komponenter indikeras, =0 garanterar minst 5 minuters produktion
Ingång 66	=1 innebär allmänt fel på sorteringsmaskinen, =0 innebär "OK".
Ingång 77	=1 innebär allmänt fel på vridbordet, =0 innebär "OK".
Ingång 88	=1 innebär allmänt fel på mataren av kretskorten, =0 innebär "OK".
Ingång 99	=1 innebär allmänt fel på tömningen av kretskorten, =0 innebär "OK".
Utgång 1	=1 innebär att vridbordet vrids till och fixeras i position 1, =0 för position 2
Utgång 5	=1 innebär att mataren aktiveras och försöker ladda ett kretskort på vridbordets monteringsposition, =0 innebär inaktivering.
Utgång 99	=1 stoppar all exekvering och larmar personal, =0 innebär "OK".

Programexempel: OBS! Detta program är endast syntaxmässigt riktigt.

--Först deklarerar alla subrutiner

```
SUB1:SUBR(VAR,VARZ);
PMOVE(VAR);
TESTI(99,0,DIT);
ZMOVE(VARZ);
GRASP;
RELEASE;
DIT;;
BRANCH(DIT);
END;
```

```
SUBMULTI1:SUBR;
WRITEO(5,1);
HOPP;;
WAITI(77,1,25,HOPP);
END;
```

```
SUBMULTI2:SUBR(VAR,VAR1);
WRITEO(VAR,VAR1);
END;
```

-- Sedan definieras I/O-tasks

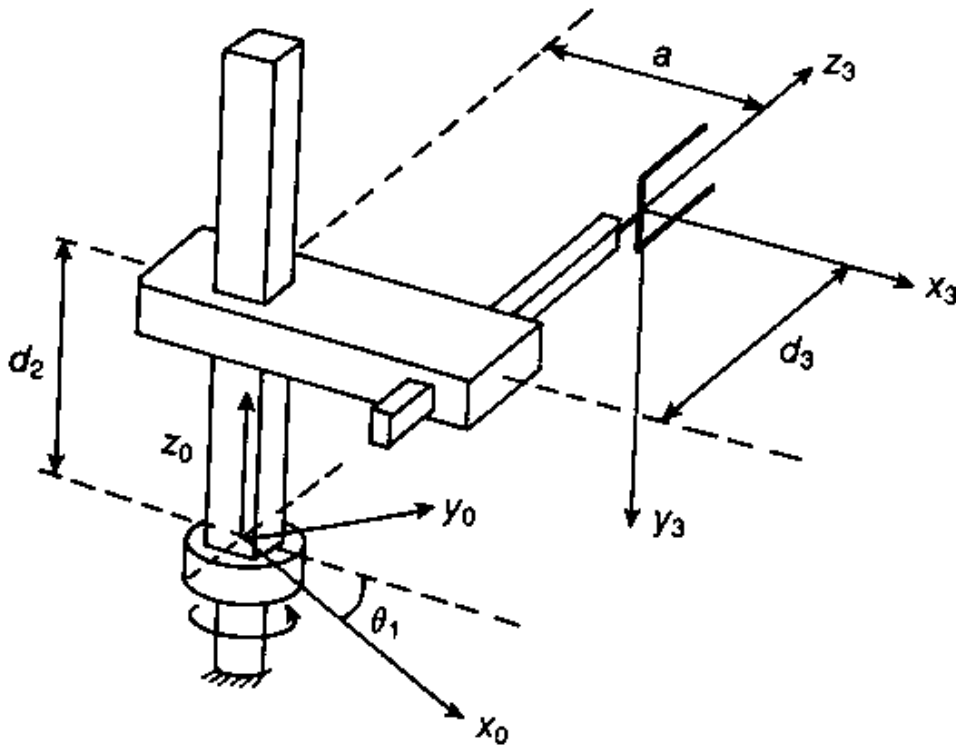
```
MULTI1: NEW IOT_CREATE(SUBMULTI1);
MULTI2: NEW IOT_CREATE(SUBMULTI2,<1,1>);
MULTI3: NEW IOT_CREATE(SUBMULTI2,<99,1>);
```

-- Sist deklarerar huvudprogrammet

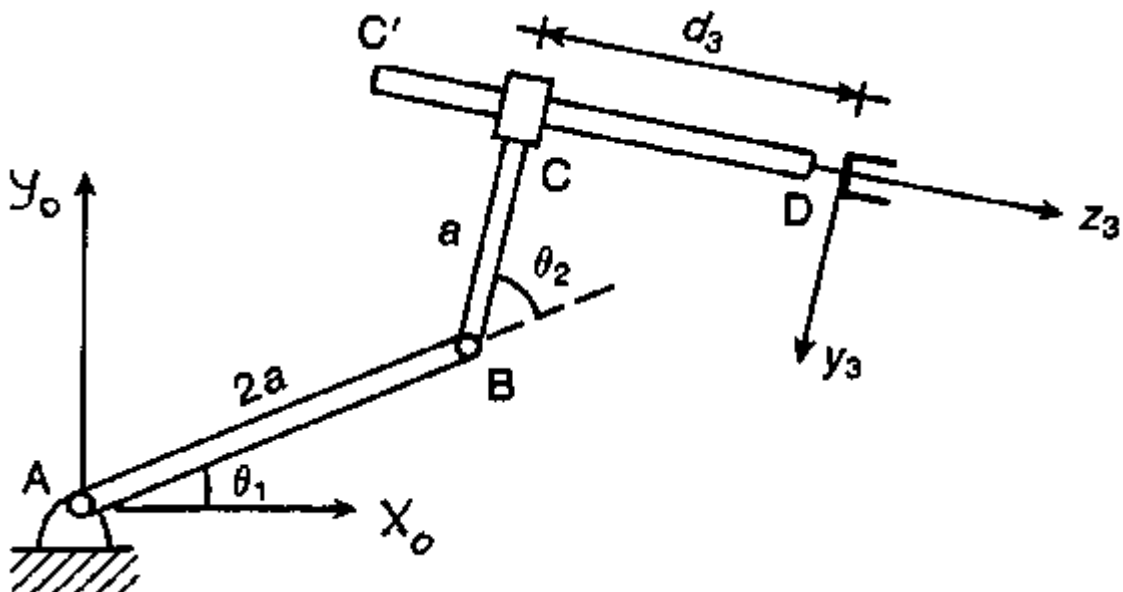
```
MAIN:SUBR;
SUB1(DIP,RAMZ);
END;
```

Roboten i figuren har tre länkvariabler θ_1 , d_2 och d_3

- 5a) Bestäm den homogena transformationsmatrisen. 5p
- 5b) Bestäm θ_1 , d_2 och d_3 då verktygets läge $(p_x, p_y, p_z) = (-a, 3a, 3a/2)$.
Det får förutsättas att d_2 och d_3 är positiva. 5p

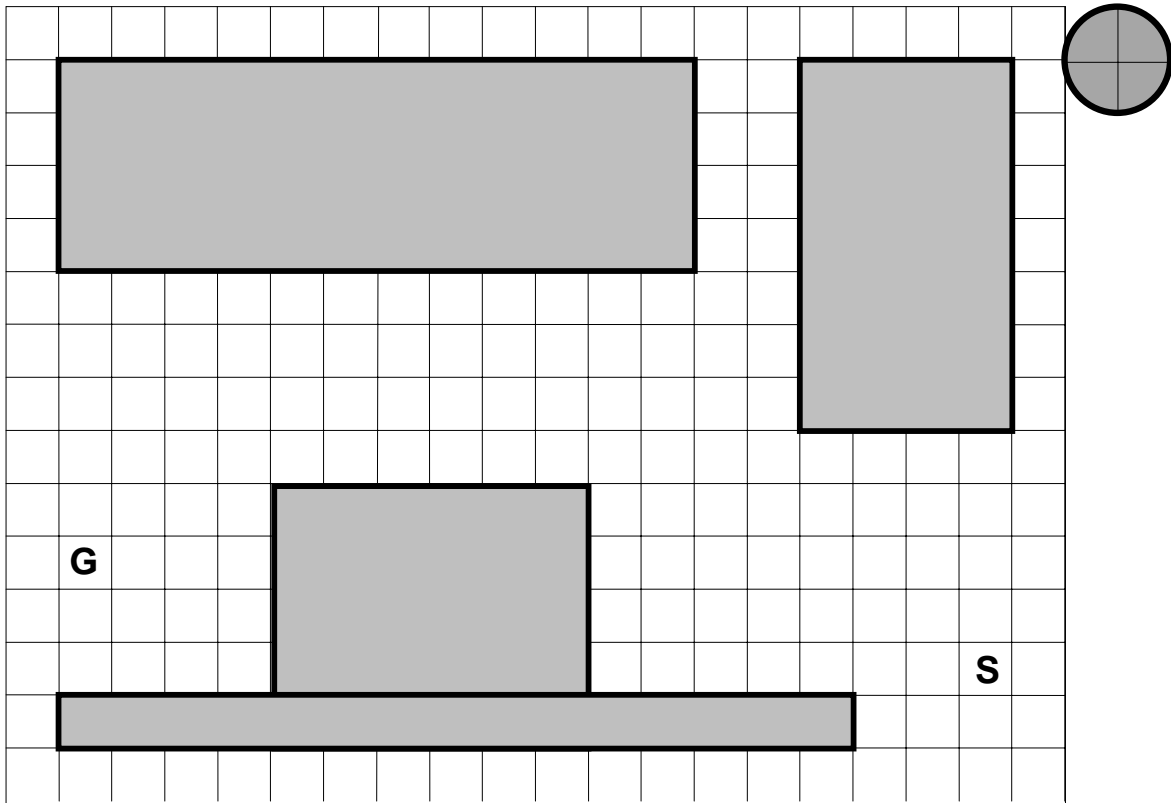


- 6a) Bestäm Jacobimatrisen (3 x 3) för den tvådimensionella roboten i figuren. 8p
- 6b) Vad krävs för att en singularitet skall föreligga? Finns det någon position där detta kan tänkas inträffa i det aktuella fallet? 2p



Svarsblankett till fråga 3c

Namn.....Personnummer.....



Rotationsmatriser

$$\mathbf{Rot}(x, \theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{bmatrix}$$

$$\mathbf{Rot}(y, \theta) = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix}$$

$$\mathbf{Rot}(z, \theta) = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$