

**AMC10B / AMC10C
AMC11B
AMC12B / AMC12C**

**AC-Servo Motor Controller
Bruger Manual**



JVL Industri Elektronik A/S

Copyright 1996-1997, JVL Industri Elektronik A/S. Der tages forbehold for ændringer af indholdet i denne brugermanual, uden forvarsel.
Ligeledes tages forbehold for trykfejl eller fejl og mangler i denne brugermanual.

MotoWare er et registreret varemærke

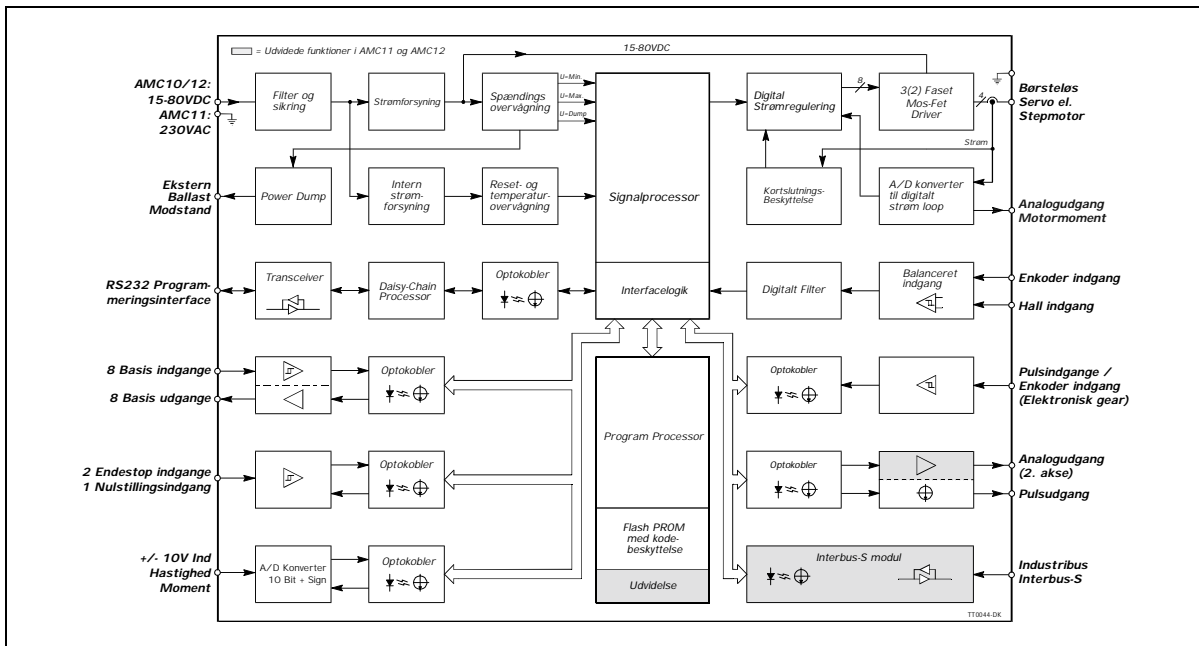
JVL Industri Elektronik A/S
Blokken 42
DK-3460 Birkerød
Denmark
Tlf. 45 82 44 40
Fax. 45 82 55 50
e-mail: jvl@jvl.dk
Internet: <http://www.jvl.dk>

Indholdsfortegnelse

| | | |
|----------|--|------------|
| 1 | Indledning | 1 |
| 1.1 | Features | 2 |
| 1.2 | Controller Forplade | 3 |
| 1.3 | Oversigt over modes | 4 |
| 1.4 | Kom i gang - Gearmode (Mode 1) | 5 |
| 1.5 | Kom i gang - Positioneringsmode (Mode 2) | 6 |
| 1.6 | Kom i gang - Registermode (Mode 3) | 7 |
| 1.7 | Kom i gang - Hastighedsmode (Mode 4) | 8 |
| 1.8 | Kom i gang - Momentmode (Mode 5) | 9 |
| 2 | Installation og justering | 11 |
| 2.1 | Generelt om installation | 12 |
| 2.2 | Valg af motortype | 13 |
| 2.3 | Justering af servoregulering | 16 |
| 2.4 | Justering af BIAS | 17 |
| 3 | Hardware | 19 |
| 3.1 | Stikforbindelser | 20 |
| 3.2 | Motortilslutning | 21 |
| 3.3 | Brugerindgange | 24 |
| 3.4 | Endestopindgange | 25 |
| 3.5 | Nulstillingsindgang | 26 |
| 3.6 | Brugerudgange | 27 |
| 3.7 | Enkoderindgang | 28 |
| 3.8 | Hall-indgang | 30 |
| 3.9 | Strømforsyning | 31 |
| 3.10 | Pulsindgange | 33 |
| 3.11 | Pulsudgange | 35 |
| 3.12 | Analogindgang | 36 |
| 3.13 | Power dump udgang | 37 |
| 3.14 | RS232 Interface | 38 |
| 3.15 | RS485 Interface | 41 |
| 3.16 | Modulinterface | 42 |
| 4 | Software | 45 |
| 4.1 | Brug af RS232 kommandoer | 46 |
| 4.2 | Gearmode (MO=1) | 47 |
| 4.3 | Positioneringsmode (MO=2) | 48 |
| 4.4 | Registermode (MO=3) | 49 |
| 4.5 | Hastighedsmode (MO=4) | 52 |
| 4.6 | Momentmode (MO=5) | 53 |
| 4.7 | Programafvikling i AMC12 | 54 |
| 4.8 | Mekanisk nulstilling | 65 |
| 4.9 | Justering af analogindgang | 66 |
| 4.10 | Kommando beskrivelser | 67 |
| 4.11 | Fejlmeddelelser | 114 |
| 4.12 | Alfabetisk oversigt over kommandoer | 119 |
| 5 | Appendix | 123 |
| 5.1 | Tekniske data | 124 |
| 5.2 | Fysiske mål | 125 |
| 5.3 | Servo loop | 128 |
| 5.4 | Fejlindikation | 129 |
| 5.5 | Almindelige fejl | 130 |
| 5.6 | Tilslutning af ukendt motor | 131 |
| 5.7 | Eksempler på tilslutning af motor | 143 |
| 5.8 | Typiske anvendelser | 145 |
| 5.9 | Connectorboard | 146 |
| | Index..... | 138 |

1.1

Features



AMC10, AMC11 og AMC12 er en serie kompakte programmerbare AC servomotor controllere.

Controllerne udmærker sig ved, at de kan styres via RS232 interface, eller en analogindgang ($\pm 10V$). Endvidere kan controllerne styres som et stepmotorsystem via pulsingange.

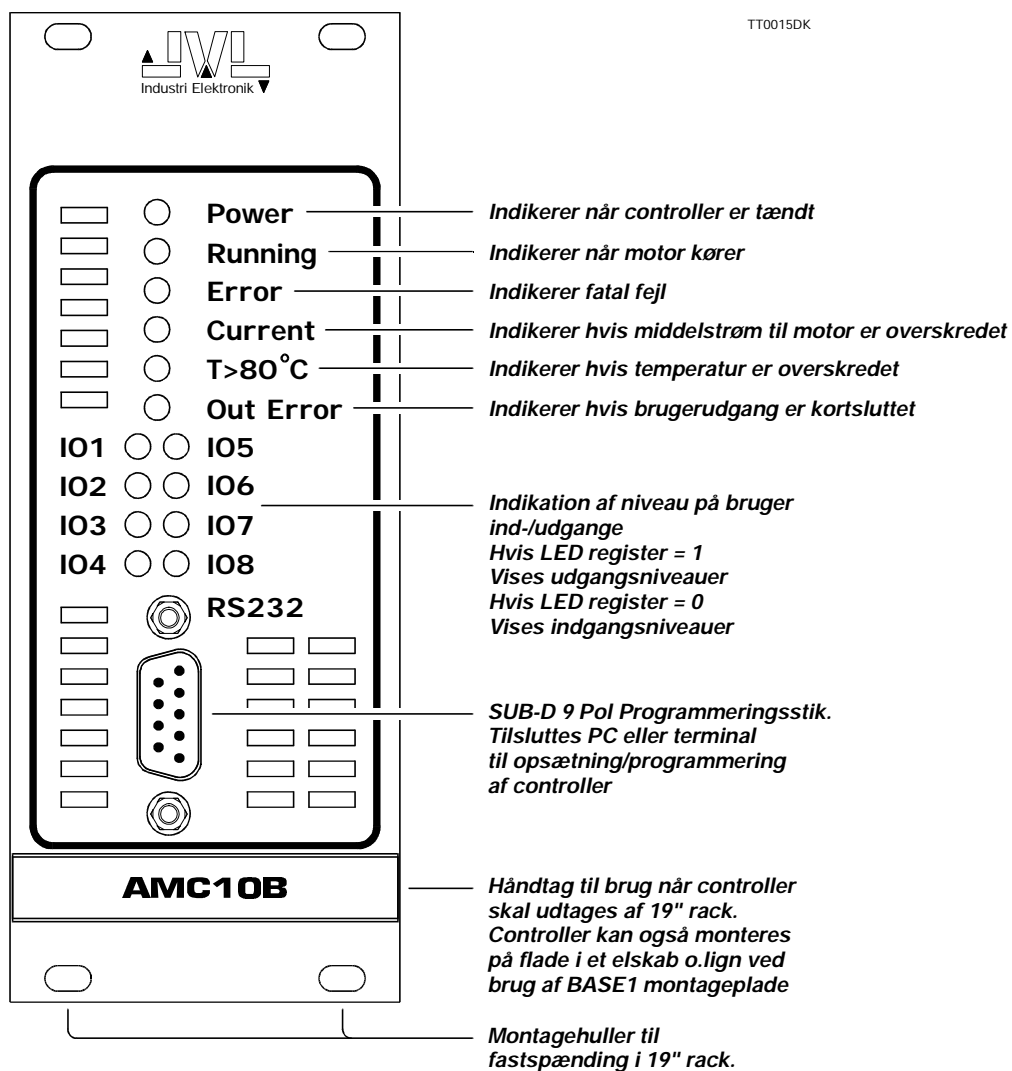
Controllerne kan konfigureres til absolut/relativ positionering via 6 binære indgange. Controllerne accepterer et balanceret eller ubalanceret signal fra en standard 2-kanals inkremental enkoder.

Alle ind- og udgange er galvanisk isolerede og over-spændingsbeskyttede.

Controllerne har 8 generelt anvendelige udgange. De kan f.eks. konfigureres til at give klarmelding, når motoren har nået sin position eller give fejlmelding, hvis der opstår en forhindring, som gør at motoren ikke kan køre. Controllerene kan monteres i 19" rack eller på en flade.

Hoved-features:

- Digital reguleringsloop
- Ekstrem præcis positionering
- Små fysiske mål
- Strøm 6A kont., 12A spids (AMCxxB)
- Strøm 12A kont., 25A spids (AMCxxC)
- Kortslutnings- og temperaturbeskyttet
- Absolut/Relativ positionering
- EMC korrekt udformning - CE mærket
- Overstrømsbeskyttelse
- Følgende indgangsmuligheder:
 - Analog $\pm 10V$
 - Steppuls og retning
 - Puls op - puls ned
 - Inkremental-enkoder
 - Valg af position via tabel
- Grafisk overvågning af hastighed, moment, fejlposition m.m
- Endestopindgange
- RS232 Interface
- Husker set-up i EEPROM
- Kan håndtere motorer op til 1kW
- Preprogrammerede hastighedsprofiler
- Automatisk nulpunktssøgning
- Programmering i simpelt sprog
- Hvilkårlig AC-motor kan benyttes



1.3

Oversigt over modes

1.3.1 Controller modes

Controlleren rummer mange enkeltfeatures, som er generelt anvendelige, men overordnet kan følgende grundfunktioner (*modes*) vælges via controllerens modekommando *MO*, eller via *Motoware*'s parameter opsætnings-menu.

1. Gearmode

Controlleren vil fungere som et stepmotorsystem, idet motoren vil bevæge sig et trin, hver gang pulsindgangene tilføres en spændingspuls. Hastigheden samt acceleration/deceleration vil være bestemt af den eksterne frekvens.

Formatet på disse pulsindgange muliggør følgende:

- Tilslutning af en inkrementalenkoder således, at motoren kører i et valgbart gearforhold til denne inkrementalenkoder, også kaldet elektronisk gear.
- Tilslutning af et puls- og retningssignal på de 2 pulsindgange. Dette er et typisk stepmotorformat.
- Tilslutning af et pulssignal på én af de 2 pulsindgange. Hvis motoren skal køre frem tilføres pulser på den ene indgang, og der tilføres pulser på den anden indgang hvis motoren skal køre baglæns.

2. Positioneringsmode

Controlleren vil positionere via kommandoer, der sendes på RS232 interfacet.

Denne mode kan primært benyttes i systemer, hvor controlleren indgår i et system med permanent kontakt til en PC via RS232 interfacet. Endvidere kan det anbefales at indkøre et vilkårligt system via denne mode.

3. Registermode

Controllerens parametersæt (X0-X63) vil indeholde de positioner og hastigheder m.v., der kræves i det aktuelle system. Disse registre kan udpeges via brugerindgangene og udføres ved at aktivere en startindgang. Denne mode er særdeles slagkraftig, idet controlleren selv klare hele positioneringsforløbet.

4. Hastighedsmode

Controlleren styrer motorens hastighed iflg. analogindgangen.

Denne mode benyttes typisk i simple opgaver, eller opgaver hvor der indgår en anden enhed som overordnet styring af hastighed og position eksempelvis et PC-kort eller en PLC med aksekort.

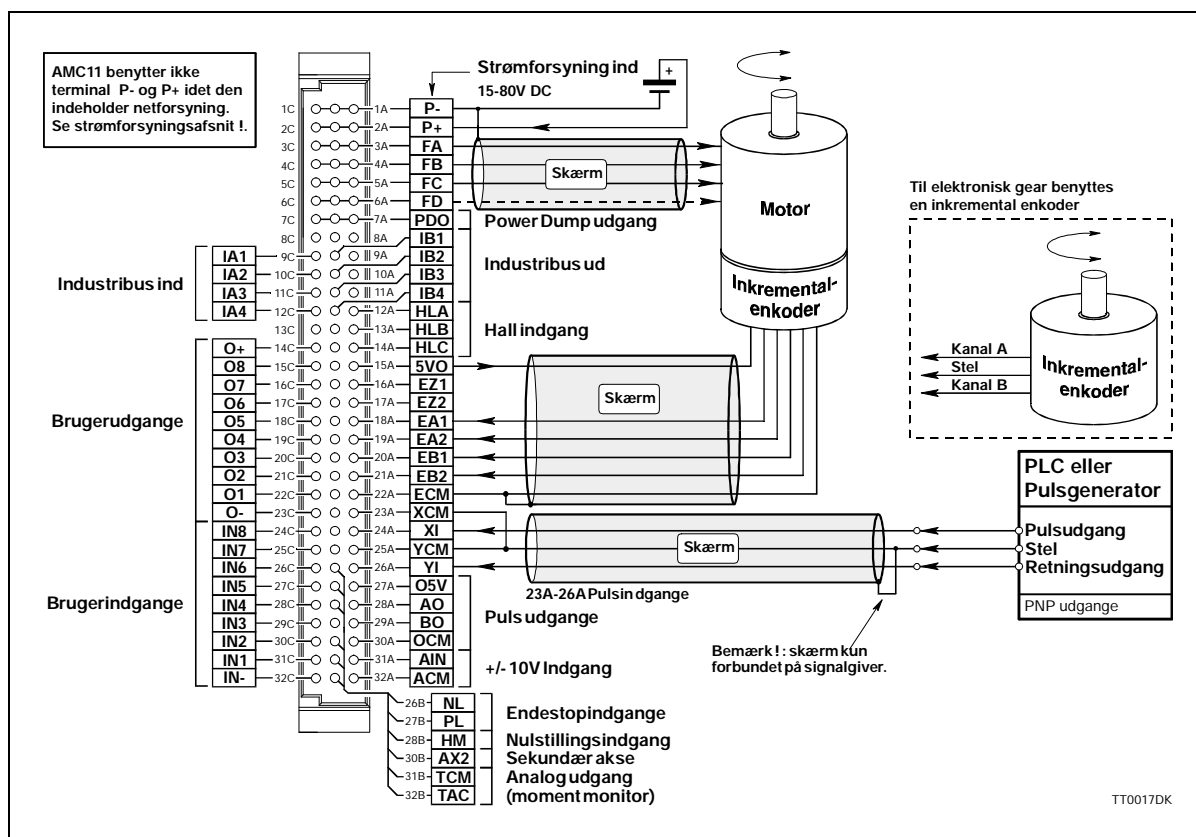
5. Momentmode

Controlleren styrer motorens drejningsmoment iflg. analogindgangen.

Typiske opgaver hvor denne mode kan benyttes, er f.eks. til opspoling eller tilspænding af folie eller ledning m.v.

De enkelte modes er illustreret på de følgende sider. Disse sider er en hurtig måde at få et funktionsdygtigt system i gang, men ønskes mere uddybende dokumentation, er de enkelte ind- og udgange samt kommandoer beskrevet i afsnittene *Hardware* side 19 og *Software* side 47.

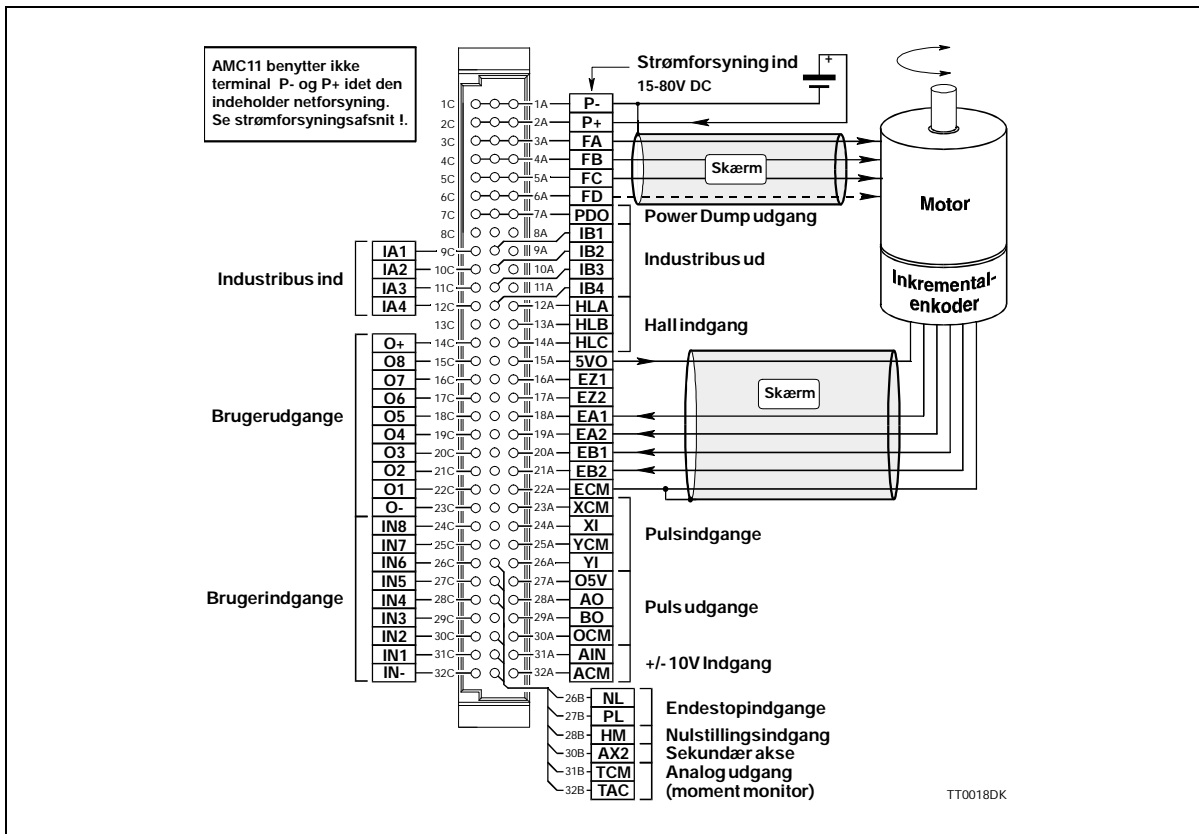
1.4 Kom i gang - Gearmode (Mode 1)



Følg nedenstående procedure, hvis controlleren skal benyttes i mode 1 (Gearmode)

1. Tilslut controller som vist ovenfor. Se evt. følgende afsnit : *Motortilslutning* side 21 / *Enkoderindgang* side 29 / *Strømforsyning* side 33 / *Pulsindgange* side 35.
2. Tilslut PC med et terminalprogram (F.eks. JVL's *MotoWare*), følg evt. beskrivelse af RS232 interface i afsnittet *RS232 Interface* side 41.
3. Tænd controlleren, men sørg for at alle indgange er logisk "0". Det er kun lysdioden *Power* og eventuelt *Out 1*, der må lyse. Hvis en eller flere af de røde lysdioder lyser eller blinker, er controlleren sandsynligvis indstillet til den forkerte motor. Følg instruktionerne i afsnittet *Generelt om installation* side 12
4. Send kommando ? (enter) til controller og afvent, at controlleren sender statusoversigt. Vises statusoversigt, er RS232 interface og strømforsyning korrekt forbundet.
5. Sæt nu controlleren i gearmode ved at sende kommandoen *MO=1* (enter). Controlleren skulle gerne give svaret *Y*, som tegn på at gearmode er valgt.
6. Reguleringsparametrene *KD*, *KP*, og *KI*, er som default indstillet moderat, hvilket vil sige, at motoren kan køre, uden at justere yderligere. Dog er det påkrævet at justere parametrene, hvis systemet skal arbejde optimalt. Se også afsnittet *Justering af servoregulering* side 16.
7. Controlleren er nu i gearmode.

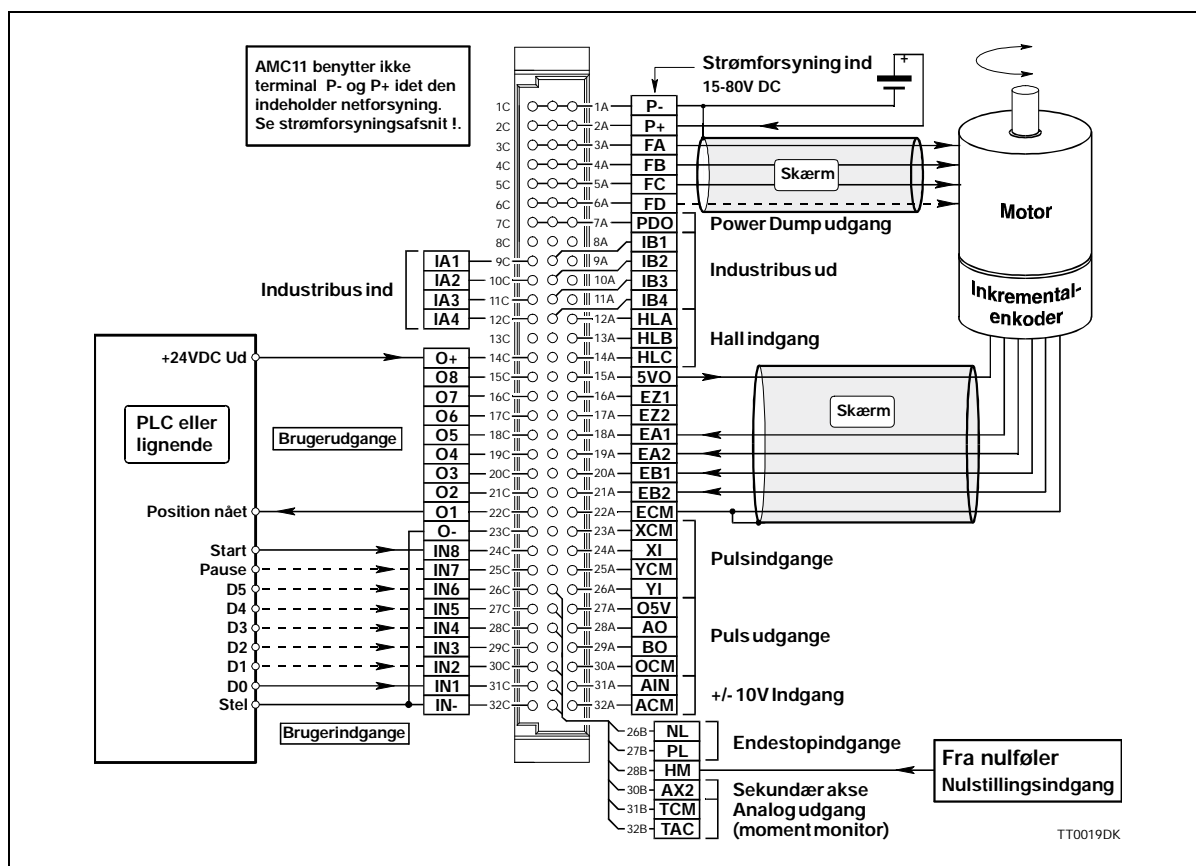
1.5 Kom i gang - Positioneringsmode (Mode 2)



Følg nedenstående procedure, hvis controlleren skal benyttes i mode 2 (positioneringsmode)

1. Tilslut controller som vist ovenfor. Se evt. følgende afsnit : *Motortilslutning* side 21 / *Enkoderindgang* side 29 / *Strømforsyning* side 33.
2. Tilslut PC med et terminalprogram (F.eks. JVL's *MotoWare*), følg evt. beskrivelse af RS232 interface i afsnittet *RS232 Interface* side 41 .
3. Tænd controlleren, men sørg for at alle indgange er logisk "0". Det er kun lysdioden *Power* og eventuelt *Out 1*, der må lyse. Hvis en eller flere af de røde lysdioder lyser eller blinker, er controlleren sandsynligvis indstillet til den forkerte motor. Følg instruktionerne i afsnittet *Generelt om installation* side 12
4. Send kommandoen ? (enter) til controlleren, og afvent at controlleren sender statusoversigt. Vises statusoversigt, er RS232 interface og strømforsyning korrekt forbundet.
5. Sæt nu controlleren i positioneringsmode ved at sende kommandoen *MO=2* (enter). Controlleren skulle gerne give svaret *Y*, som tegn på at positionerings-mode er valgt.
6. Reguleringsparametrene *KD*, *KP*, og *KI*, er som default indstillet moderat, hvilket vil sige, at motoren kan køre uden at justere yderligere. Dog er det påkrævet at justere parametrene, hvis systemet skal arbejde optimalt. Se afsnit *Justering af servoregulering* side 16.
7. Controlleren er nu i positionerings-mode. Som test af dette kan motoren køres til den absolutte position 1000 ved at sende kommandoen *SP=1000* (enter). Som følge af dette skulle motoren gerne køre til denne position. Ved at sende kommandoen *SP=-1000* (enter) skal motoren gerne køre den anden vej mod position -1000. Sker dette ikke eller kører motoren i meget lang tid, kan det skyldes, at positionstælleren enten stod på 1000, eller at den forrige position lå langt væk fra 1000. Se afsnittet *Positioneringsmode (MO=2)* side 50 eller *Kommando beskrivelser* side 70, vedr. øvrige kommandoer.

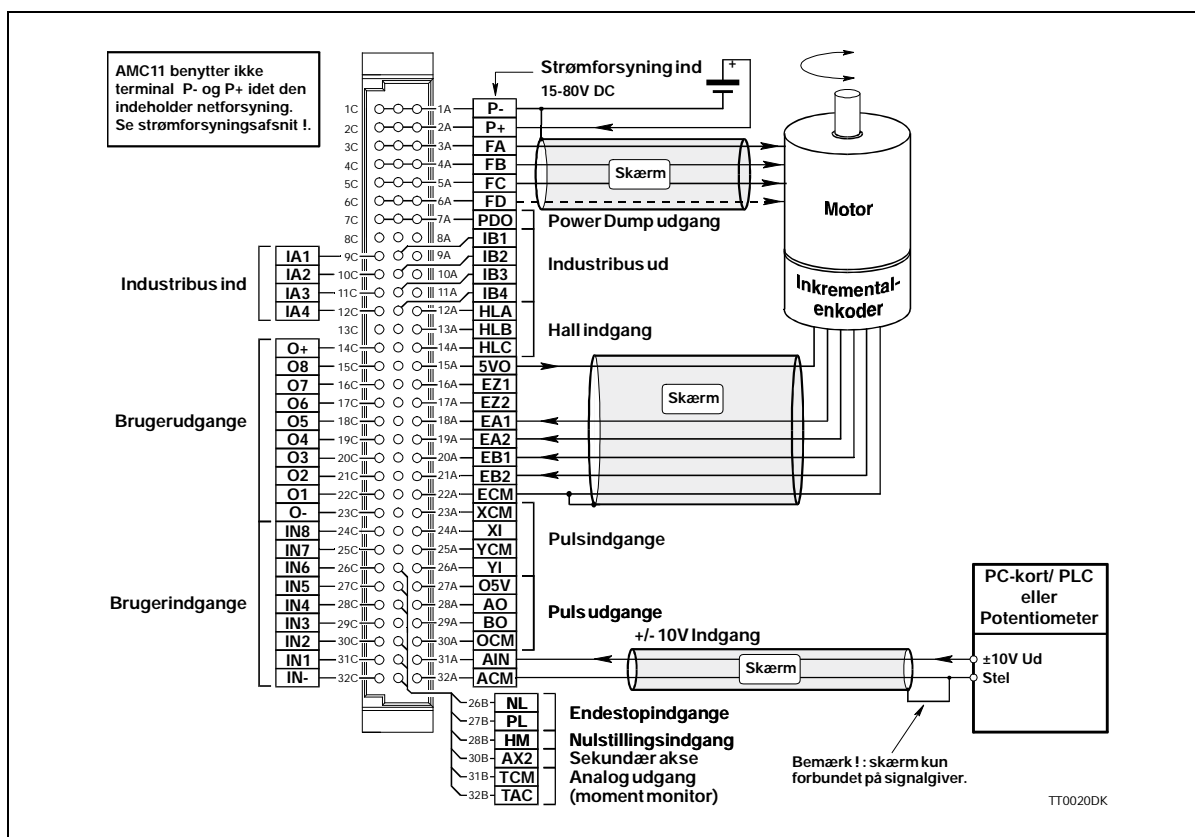
1.6 Kom i gang - Registermode (Mode 3)



Følg nedenstående procedure, hvis controlleren skal benyttes i mode 3 (Registermode)

1. Tilslut controller som vist ovenfor. Se evt. følgende afsnit *Motortilslutning* side 21 / *Brugerindgange* side 24 / *Brugerudgange* side 27 / *Enkoderindgang* side 29 / *Strømforsyning* side 33.
2. Tilslut PC med et terminalprogram (F.eks. JVL's *MotoWare*), følg evt. beskrivelse af RS232 interface i afsnit *RS232 Interface* side 41.
3. Tænd controlleren, men sørg for at alle indgange er logisk "0". Det er kun lysdioden *Power* og eventuelt *Out 1*, der må lyse. Hvis en eller flere af de røde lysdioder lyser eller blinker, er controlleren sandsynligvis indstillet til den forkerte motor. Følg instruktionerne i afsnittet *Generelt om installation* side 12
4. Send kommandoen ? (enter) til controlleren og afvent, at controlleren sender statusoversigt. Vises statusoversigt, er RS232 interface og strømforsyning korrekt forbundet.
5. Sæt nu controlleren i registermode ved at sende kommandoen *MO=3* (enter). Controlleren skulle gerne give svaret *Y*, som tegn på at register-mode er valgt.
6. Reguleringsparametrene *KD*, *KP*, og *KI* er som default indstillet moderat, hvilket vil sige, at motoren kan køre uden at justere yderligere. Dog er det påkrævet at justere parametrene, hvis systemet skal arbejde optimalt. Se afsnit *Justering af servoregulering* side 16.
7. Controlleren er nu i register-mode. Prøv at tilslutte en spænding på indgang 1 og 8 (startindgang). Motoren skulle gerne køre til position 1000. Denne værdi er indlagt som default i *XPI* ved levering. For yderligere information om mode 3 se afsnittet *Registermode (MO=3)* side 51

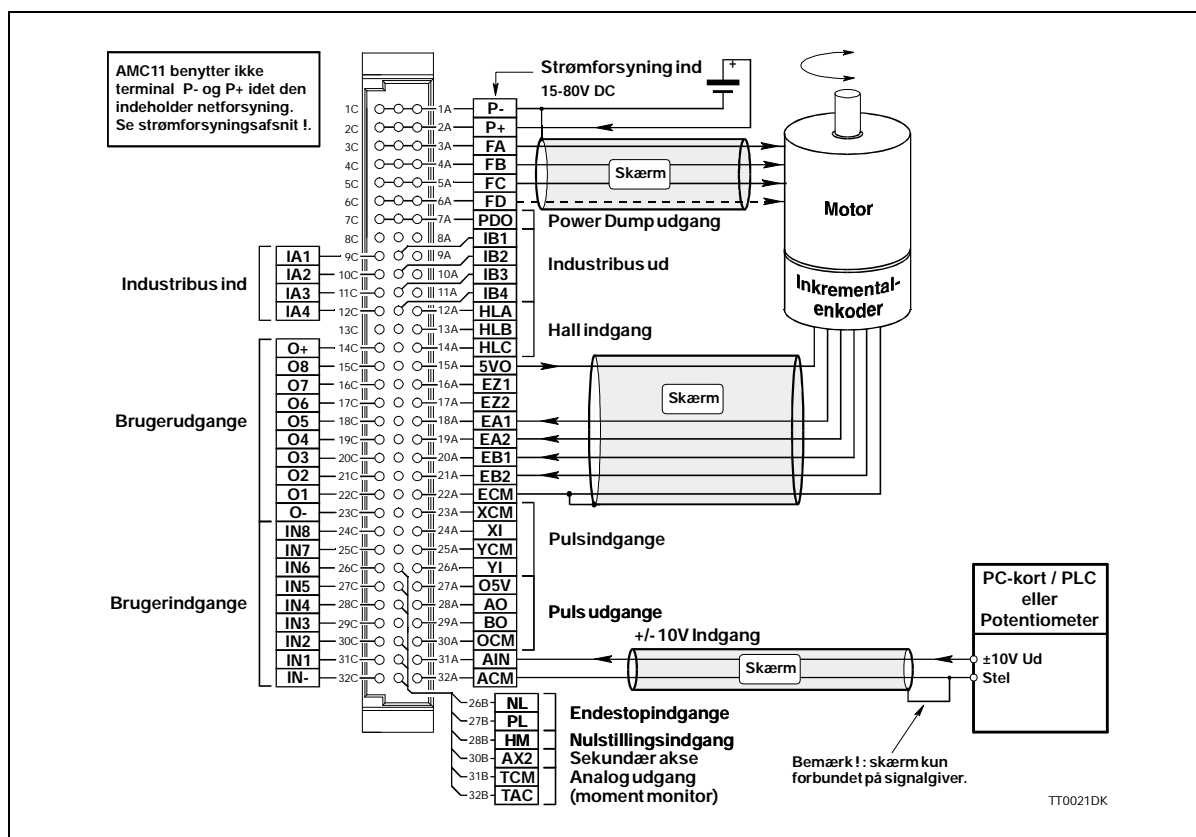
1.7 Kom i gang - Hastighedsmode (Mode 4)



Følg nedenstående procedure, hvis controlleren skal benyttes i mode 4 (hastigheds-mode)

1. Tilslut controller som vist ovenfor. Se evt. følgende afsnit *Motortilslutning* side 21 / *Enkoderindgang* side 29 / *Strømforsyning* side 33 / *Analogindgang* side 38.
2. Tilslut PC med et terminalprogram (F.eks. JVL's *MotoWare*), følg evt. beskrivelse af RS232 interface i afsnit *RS232 Interface* side 41.
3. Tænd controlleren, men sørg for at analogindgangen er 0 volt. Det er kun lysdioden *Power* og eventuelt *Out 1*, der må lyse. Hvis en eller flere af de røde lysdioder lyser eller blinker, er controlleren sandsynligvis indstillet til den forkerte motor. Følg instruktionerne i afsnittet *Generelt om installation* side 12.
4. Send kommandoen ? (enter) til controlleren og afvent, at controlleren sender statusoversigt. Vises statusoversigt, er RS232 interface og strømforsyning korrekt forbundet.
5. Sæt nu controlleren i hastighedsmode ved at sende kommandoen $MO=4$ (enter). Controlleren skulle gerne give svaret *Y* som tegn på, at hastigheds-mode er valgt.
7. Reguleringsparametrene *KD*, *KP*, og *KI* er som default indstillet moderat, hvilket vil sige, at motoren normalt kan køre uden at justere yderligere. Dog er det påkrævet at justere parametrene, hvis systemet skal arbejde optimalt. Hvis motoren ikke kører på dette tidspunkt, prøv da først at sæt *KI* til en høj værdi (100-1000). Se også afsnit *Justering af servoregulering* side 16.
8. Controlleren er nu i hastigheds-mode. Når spændingen på indgangen er større end 0V, vil motoren køre den ene vej med en hastighed, der er proportional med spændingen. Når indgangsspændingen er under 0V (negativ), vil motoren køre den anden vej.
For yderligere information se afsnittet *Hastighedsmode (MO=4)* side 54.

1.8 Kom i gang - Momentmode (Mode 5)



Følg nedenstående procedure, hvis controlleren skal benyttes i mode 5 (moment-mode)

1. Tilslut controller som vist ovenfor. Se evt. følgende afsnit : *Motortilslutning* side 21 / *Strømforsyning* side 33 / *Analogindgang* side 38.
2. Tilslut PC med et terminalprogram (f.eks. JVL's *MotoWare*), følg evt. beskrivelse af RS232 interface i afsnit *RS232 Interface* side 41.
3. Tænd controlleren, men sørg for at analogindgangen er 0 volt. Det er kun lysdioden *Power* og eventuelt *Out 1*, der må lyse. Hvis en eller flere af de røde lysdioder lyser eller blinker, er controlleren sandsynligvis indstillet til den forkerte motor. Følg instruktionerne i afsnittet *Generelt om installation* side 12
4. Send kommandoen ? (enter) til controlleren, og afvent at controlleren sender statusoversigt. Viser statusoversigt, er RS232 interface og strømforsyning korrekt forbundet.
5. Sæt nu controlleren i momentmode ved at sende kommandoen *MO=5* (enter). Controlleren skulle gerne give svaret *Y* som tegn på, at moment-mode er valgt.
6. Reguleringsparametrene *KD*, *KP*, og *KI* er som default indstillet moderat, hvilket vil sige, at motoren kan køre uden at justere yderligere. Dog er det påkrævet at justere parametrene, hvis systemet skal arbejde optimalt. Se afsnit . Hvis motoren ikke kører på dette tidspunkt, prøv da først at sæt *KI* til en høj værdi (100-1000). Se også afsnittet *Justering af servoregulering* side 16.
7. Controlleren er nu i moment-mode. Når spændingen på indgangen er større end 0V, vil motoren yde et positivt drejningsmoment, der er proportional med spændingen. Når indgangsspændingen er under 0V (negativ), vil motoren yde et negativt drejningsmoment proportional med spændingen.

For yderligere information se afsnittet *Momentmode (MO=5)* side 55.

2 Installation og justering

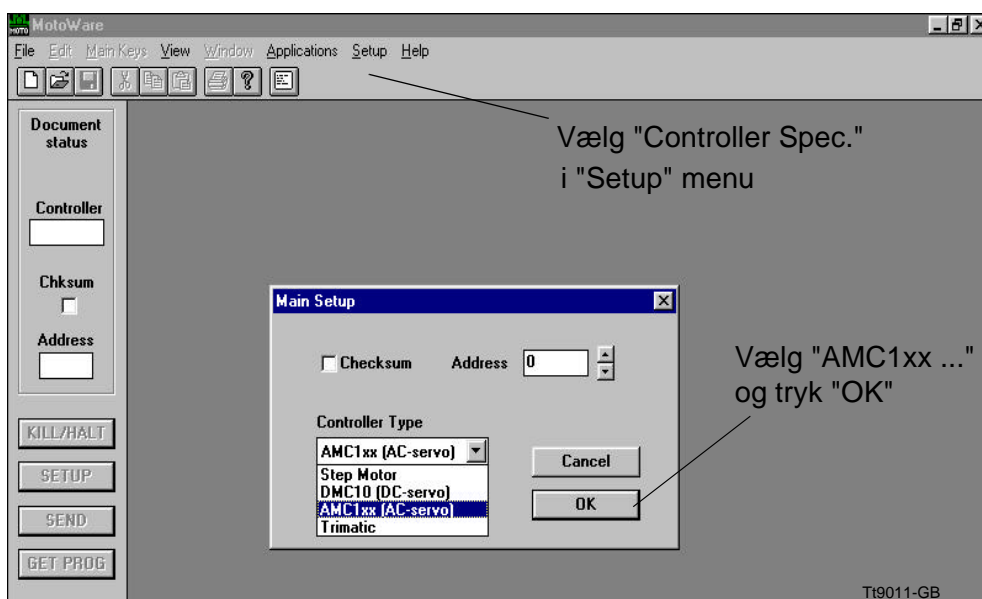
Det anbefales at læse dette afsnit omhyggeligt i forbindelse med installation af AC-servocontrolleren.

Når controlleren første gang er installeret bør denne checkliste følges:

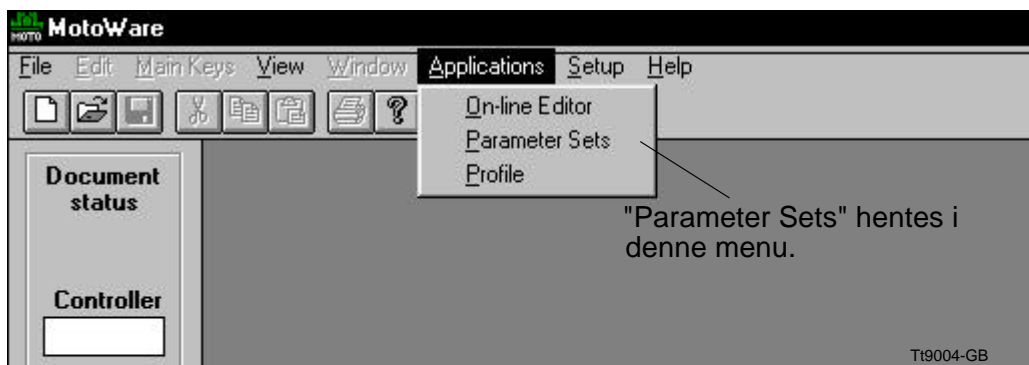
1. Skab klarhed over hvilken mode (1-5) der ønskes benyttet. Se eventuelt *Oversigt over modes* side 4, hvor der står forklaret i hovedtræk hvorledes de enkelte modes benyttes.
2. Forbind motor, enkoder, evt. hall-føler, samt diverse endestop, ind- og udgange efter behov. Enkeltheder omkring motortilslutning og ind- /udgange, strømforsyning m.v. står beskrevet i afsnit om *Hardware* side 19.
Bemærk at vedr. forbindelse af motor samt enkoder m.v., at der i appendix - (*Eksempler på tilslutning af motor* side 145) er lavet konkrete diagrammer for et udvalg af AC-servo-motorer samt stepmotorer. Her er ligeledes opført tilhørende parametre som controlleren skal indstilles til for at motoren kører optimalt.
3. Tilslut nu spænding til controlleren. Det vil sandsynligvis vise sig at de default parametre controlleren indeholder ved levering ikke svarer til den tilsluttede motortype. Dette vil resultere i at controlleren melder fejl og motoren bliver strømløs.
Hvis den benyttede motor er en af de nævnte typer i appendix (*Eksempler på tilslutning af motor* side 145) eller Motoware's parameterliste skal disse parametre overføres iflg. afsnittet *Valg af motortype* side 13.
Er motoren kendt burde systemet fungere optimalt efter at have overført det tilhørende parametersæt. Der kan evt. finjusteres - se de efterfølgende sider i dette kapitel. Grundinstallationen er nu færdig og den egentlige funktion af controlleren kan afprøves og indstilles, se beskrivelse af mode1....mode5 i softwareafsnittet side 49....55 afhængigt af den ønskede funktion.
Ønskes en optimering af det samlede system følges afsnittet *Justering af servoregulering* side 16.
Er motoren derimod **ikke** kendt skal vejledningen i afsnittet *Tilslutning af ukendt motor* side 133 følges.

2.2

Valg af motortype



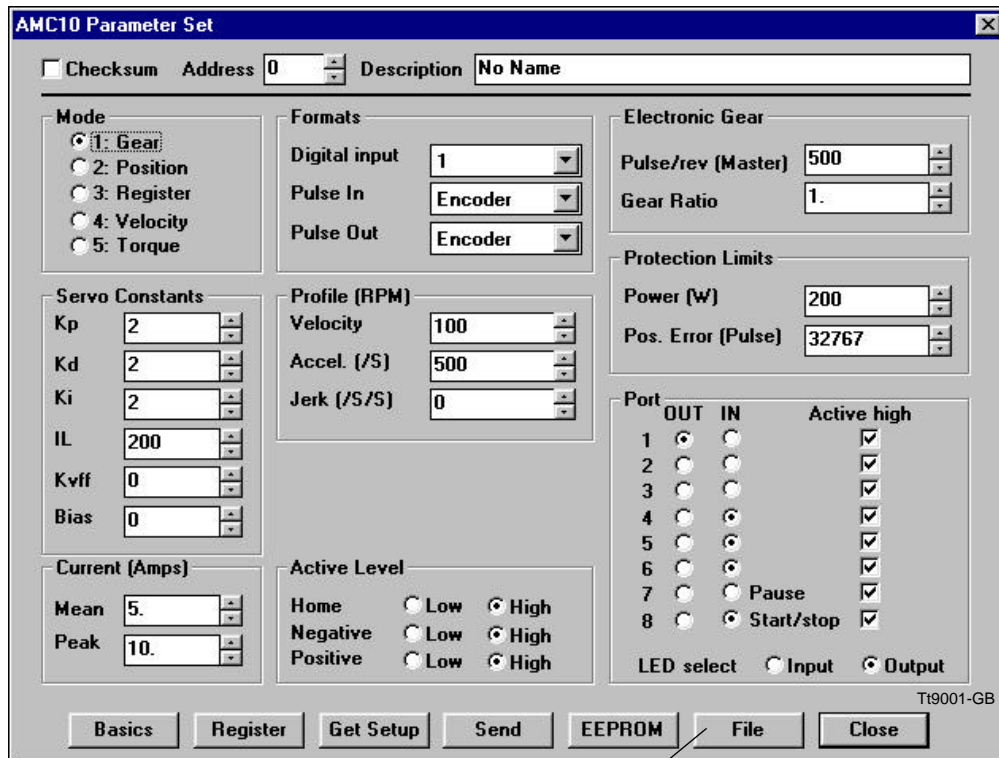
Når der skal overføres parametre til controlleren, kan det anbefales at benyttes JVL's programmeringssoftware *Motoware*. Programmet startes og RS232 kablet forbindes til controlleren. Indstil *Motoware* til at arbejde med AC-servocontroller ved at vælge AMC1xxx (AC Servo) i vinduet *Controller Spec.* under menuen *Setup*. Se ovenstående illustration. Denne opsætning omstiller *Motoware* til at arbejde med AMC10, 11 og 12, hvilket indebærer at der bliver adgang til forskellige nye vinduer med bl.a. grafisk visning af kørselsforløb m.v.



Vælg nu punktet *Parameter Sets* i menuen *Applications*. Dette punkt giver adgang til vinduet med alle grundparametre i controlleren.

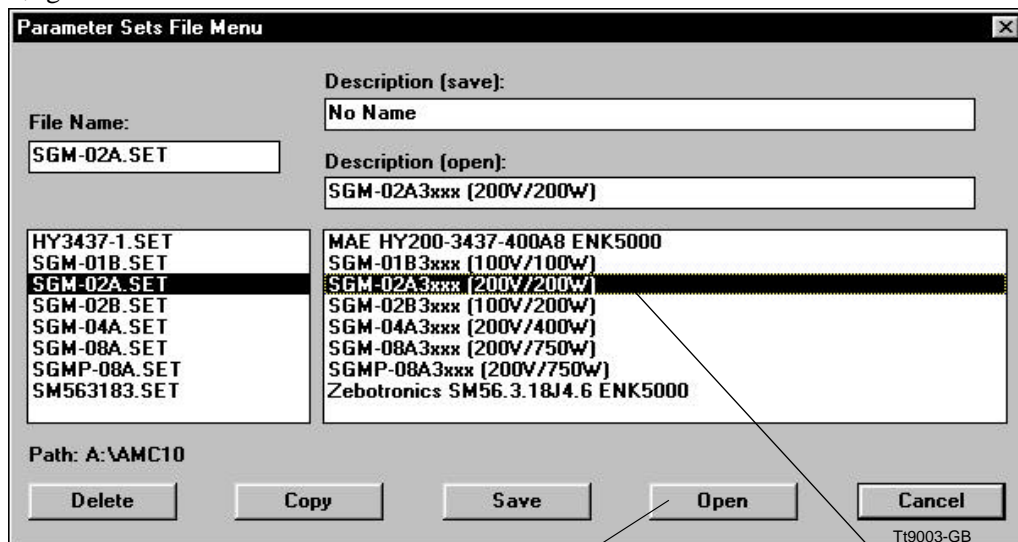
2.2

Valg af motortype



Vælg "File" for valg af motor type

Når der skal vælges en konkret motortype vælges *File*, hvorved der fremkommer efterfølgende billede.



Vælg "Open" for at hente parametre

Vælg motortype

Vælg den ønskede motortype og tast Open, for at hente parametrene.

2.2

Valg af motortype

| Port | OUT | IN | Active high |
|------|----------------------------------|----------------------------------|-------------------------------------|
| 1 | <input checked="" type="radio"/> | <input type="radio"/> | <input checked="" type="checkbox"/> |
| 2 | <input type="radio"/> | <input type="radio"/> | <input checked="" type="checkbox"/> |
| 3 | <input type="radio"/> | <input type="radio"/> | <input checked="" type="checkbox"/> |
| 4 | <input type="radio"/> | <input checked="" type="radio"/> | <input checked="" type="checkbox"/> |
| 5 | <input type="radio"/> | <input checked="" type="radio"/> | <input checked="" type="checkbox"/> |
| 6 | <input type="radio"/> | <input type="radio"/> | <input checked="" type="checkbox"/> |
| 7 | <input type="radio"/> | <input type="radio"/> | <input checked="" type="checkbox"/> |
| 8 | <input type="radio"/> | <input checked="" type="radio"/> | <input checked="" type="checkbox"/> |

Opsætning til controller

Gem opsætning i hukommelse

Parametrene for den valgte motor viser sig nu på skærmen. Overførsel af parametrene kan nu finde sted ved at trykke *Send*. Controlleren vil efter et lille stykke tid spørge om der ønskes at gemme opsætningen og/eller genstarte kontrolleren. I dette tilfælde skal der trykkes på *Save and Reset*.

Controlleren er nu indstillet til den valgte motor og grundopsætningen er afsluttet.

2.3 Justering af servoregulering

2.3.1 Justering af servo-parametre

Den anvendte regulator er af PID-typen og har derfor 3 parametre at justere. Regulatorens funktion er at sørge for, at motoren kører med en jævn og stabil bevægelse og stopper ved sit mål. De 3 parametre skal stilles på baggrund af det aktuelle system, idet motortype, belastning, forsyningsspænding m.v. har en afgørende betydning for parametrenes indstilling.

De 3 parametre har følgende benævnelse og funktion :

- KP Bestemmer systemets proportionalforstærkning. Denne parameter er den vigtigste, idet systemet vil kunne fungere alene med denne parameter.
- KD Bestemmer systemets differentialforstærkning. Denne parameter bestemmer, hvor aggressivt systemet skal reagere ved pludselige belastningsændringer eller en pludselig hastighedsændring.
- KI Bestemmer integrationen af positionsfejlen. Denne parameter bestemmer, hvor meget en vedvarende positionsfejl skal have at indflydelse på motorens position og hastighed.

De 3 parametre indstilles hurtigt efter følgende metode:

Start Motoware, og gå ind i *On-line* editoren. Her kan de forskellige parametre indtastes.

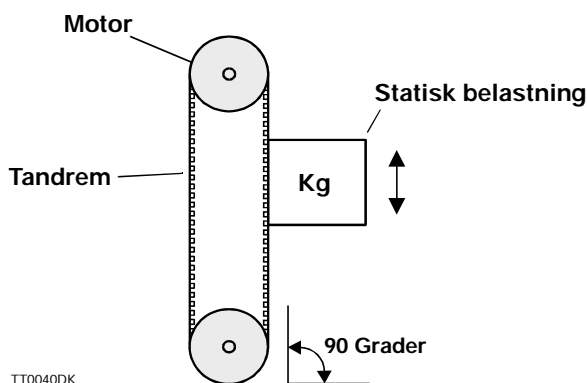
1. Alle 3 parametre sættes til 0, ved at skrive $KP=0$ (enter) og dernæst $KI=0$ (enter) og dernæst $KD=0$ (enter).
2. Der skrues langsomt op for KP, indtil systemet begynder at blive uroligt. KP stilles derefter på halvdelen af denne værdi.
3. For at gøre systemet hurtigere kan KD indføres.
KD skrues op, indtil systemet bliver uroligt, hvorefter der skrues ned til ca. 0,5-0,7 af denne værdi.
4. KI skrues op, indtil systemet er uroligt. KI sættes nu til ca. 0,5 - 0,75 af denne værdi.
5. Hvis der ønskes et system som reagerer hurtigt på en opstået positionsfejl, men den opsummerede fejl ikke vokser til uoverskuelige højder, benyt integral summeringsgrænsen IL.
6. Afslut evt. med at finjustere de enkelte parametre.
7. Husk at gemme parametrene i den permanente hukommelse ved at skrive MS (enter).

2.4

Justering af BIAS

Controlleren indeholder en parameter benævnt BIAS. Denne parameter kan benyttes til opgaver hvor motoren er udsat for en vedvarende belastning, som f.eks. en løfteenhed. BIAS funktionen muliggør en udbalancering af denne statiske belastning, hvadenten belastningen trækker eller hiver i motoren. Denne udbalancering kan som regel være en fordel idet belastningen for PID filteret er ensartet, uanset om motoren skal køre den ene eller anden vej, og i den sidste ende giver dette en lettere justering af det samlede system med en dertil hurtigere responstid.

Illustration af løfteenhed :



Justeringen af BIAS foretages under installation af systemet og foretages efter følgende procedure.

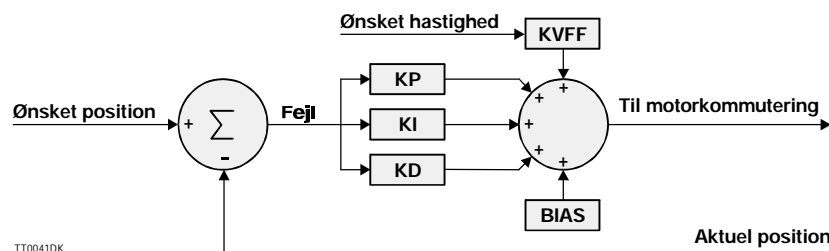
1. Start Motoware og controller. Gå ind i "On line editor".
2. Check at der er kontakt med controller ved at skrive ? (enter).
3. Sørg for at motoren har den ønskede belastning, som systemet skal fungere med.
4. Sæt controlleren i Mode 2 ved at skrive $MO=2$ (enter).
5. Sæt PID-filteret ud af funktion ved at skrive $KP=0$ (enter) $KI=0$ (enter) og $KD=0$ (enter).

Men vær opmærksom på at motoren bliver strømløs og dermed slipper sin belastning.

6. Juster BIAS til en passende værdi, således at motoren er istand til at holde belastningen i nogenlunde ro. Start med at sætte BIAS til 100 ved at skrive $BIAS=100$ (enter). Forøg BIAS i trin af 100 eller mindre indtil ligevægten er opnået.

Vær opmærksom på at controlleren kan gå i fejl under justeringen, hvis motorens positionsfejl overstiger det fastsatte i parameteret PE . Indstil eventuelt PE til 0 under denne justering, således at controlleren ignorerer positionsfejlen. Hvis belastningen er modsat positiv kørselsretning skal BIAS sættes i negativt område, f.eks. $BIAS=-100$ (enter). Vær opmærksom på at hvis BIAS sættes til en for høj værdi, vil motoren begynde at køre.

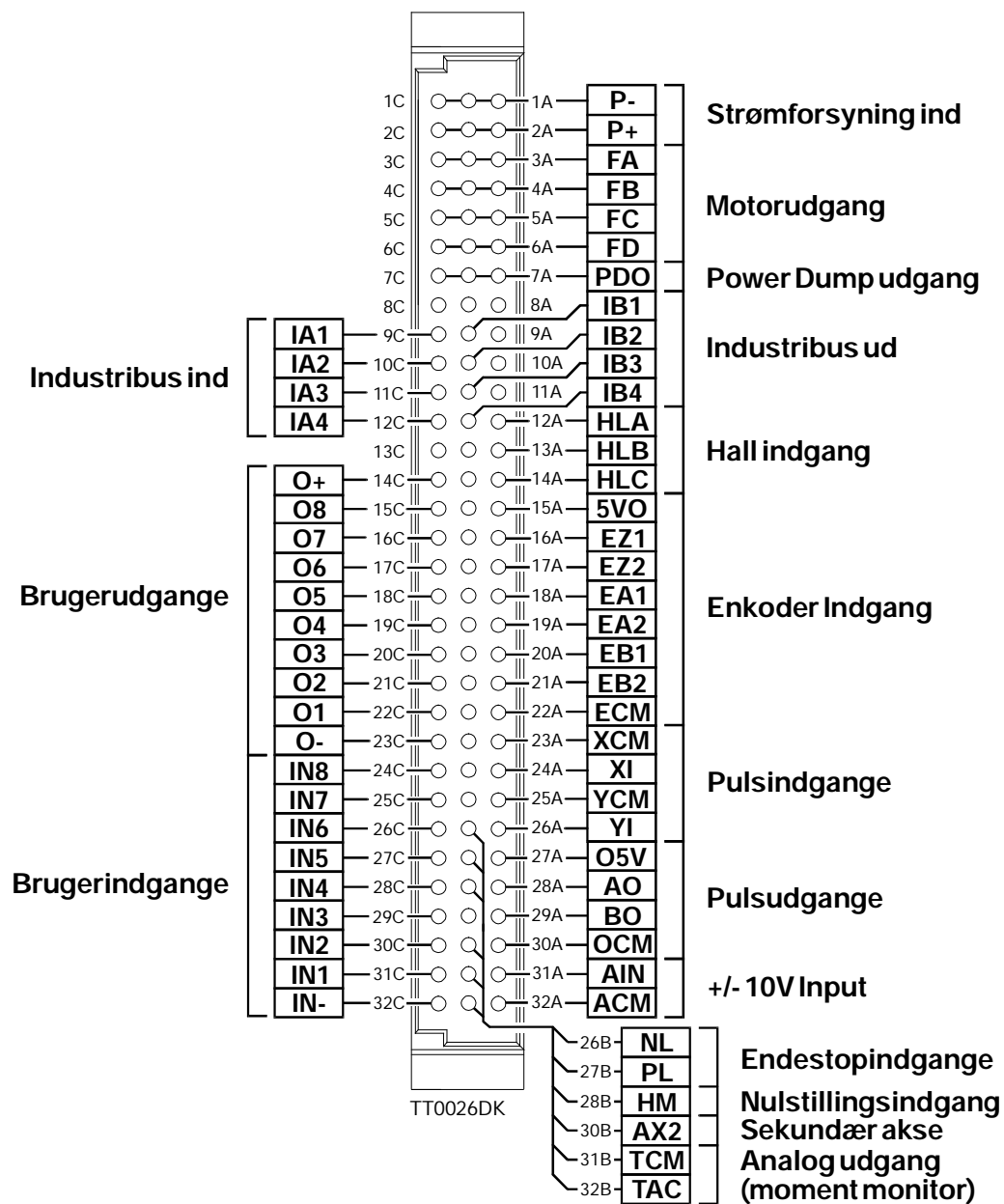
7. Til sidst indstilles filterkonstanterne tilbage (KP , KI , KD) hvor de stod før $BIAS$ justeringen og den fundne $BIAS$ værdi gemmes i controllerens permanente hukommelse, ved at sende kommandoen MS (enter). Filterkonstanterne skal evt. genjusteres efter $BIAS$ justeringen. Se afsnittet *Justering af servoregulering* side 16.



3.1

Stikforbindelser

(Stik DIN41612 ver. C)



3.2

Motortilslutning

3.2.1 Generelt om tilslutning af motor

Controlleren er beregnet for styring af almindelige AC-servomotorer (børsteløse), eller stepmotorer med vinkelenkoder. Controlleren kan levere op til 12 Amp kontinuerligt og 25 Amp. spids. Disse strømme skal indstilles via kommandoerne *CA* og *CP*.

Driveren i controlleren er opbygget med Mos-Fet transistorer, hvilket giver en særdeles god virkningsgrad. Spændingen til motoren bliver reguleret med en frekvens på 24,3kHz, hvilket sikrer, at motoren ikke frembringer hørbare lyde som følge af reguleringen.

Driverens skiftetid er meget lille (<200nS), hvilket kan medføre, at der opstår højfrekvente støjkomponenter på ledningerne mellem driveren og motoren.

Dette kan i visse tilfælde give anledning til uønsket påvirkning af andet elektronisk udstyr i nærheden af servomotorsystemet. For at imødegå denne problematik skal forbindelsen mellem controlleren og den tilsluttede motor foretages med skærmet kabel som vist på illustrationerne side 22 og 23. Det kan endvidere kraftigt anbefales, at motorens enkoderkabel skærmes på samme måde, for at undgå påvirkning fra motorledningerne til enkodersignalet.

3.2.2 Kortslutning af motorudgang

Motorudgangen kan tåle kortslutning mellem terminalerne FA, FB, FC og FD. Endvidere kan alle motor-terminaler tåle at blive kortslettet til stel eller den positive forsyning.

Hvis en kortslutning opstår, vil controlleren stoppe al aktivitet og melde fejl ved, at lysdioden *Current* lyser rødt. Endvidere vil controllerens fejlregister blive aktiveret, se kommandoerne ES og EST.

3.2.3 Tilladt motorinduktion

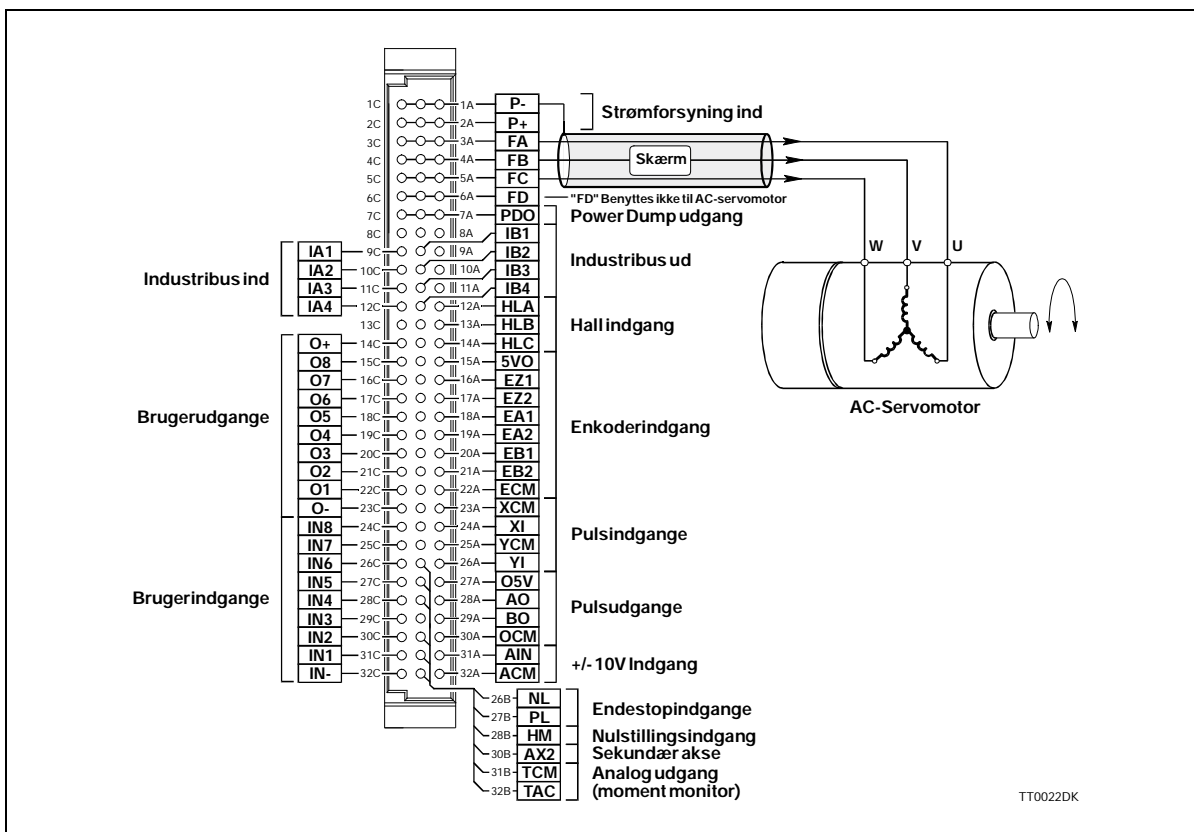
Driveren kan drive motorer der har en induktion pr. fase i området 0,5 til 20 mH.

Benyttes en motor med lavere induktion skal der placeres en spole på 0,5-1mH i serie med hver motorledning.

Denne spole vil fungere som integrator og sikre at controlleren kan kontrollere strømmen korrekt.

3.2

Motortilslutning



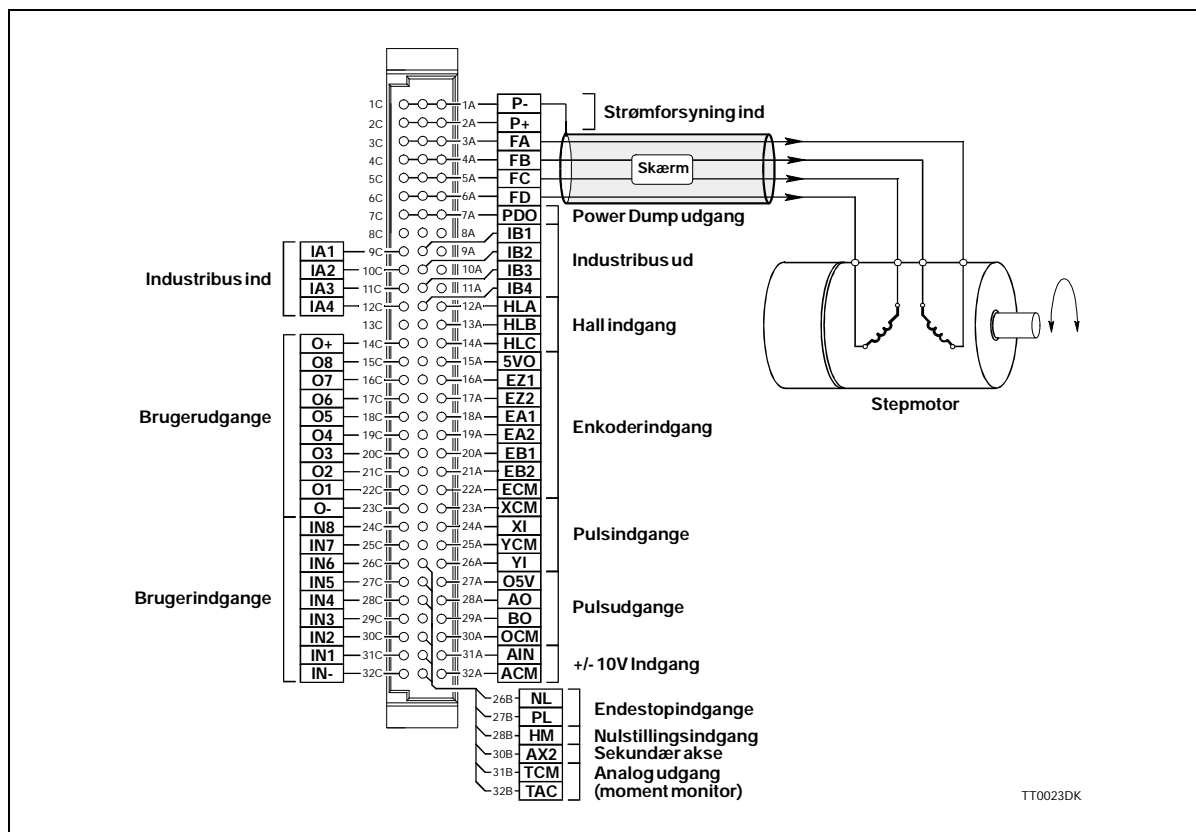
3.2.4 Tilslutning af 3-faset motor

Tilsluttes controlleren en 3 faset børsteløs motor, benyttes terminalerne FA, FB og FC. Der skal benyttes skærmet kabel mellem motoren og controlleren. Den aktuelle motors middelstrøm og spidsstrøm skal indstilles via de 2 controllerkommandoer CA og CP - se afsnittet *Justering af motorstrøm* side 139.

Se afsnittet *Eksempler på tilslutning af motor* side 145 vedr. tilslutning af diverse fabrikkemotorer.

3.2

Motortilslutning



3.2.5 Tilslutning af 2 eller 4-faset steppmotor

Tilsluttes controlleren en 2 eller 4 faset steppmotor, benyttes terminalerne FA, FB, FC og FD.

Der skal benyttes skærmet kabel mellem motoren og controlleren.

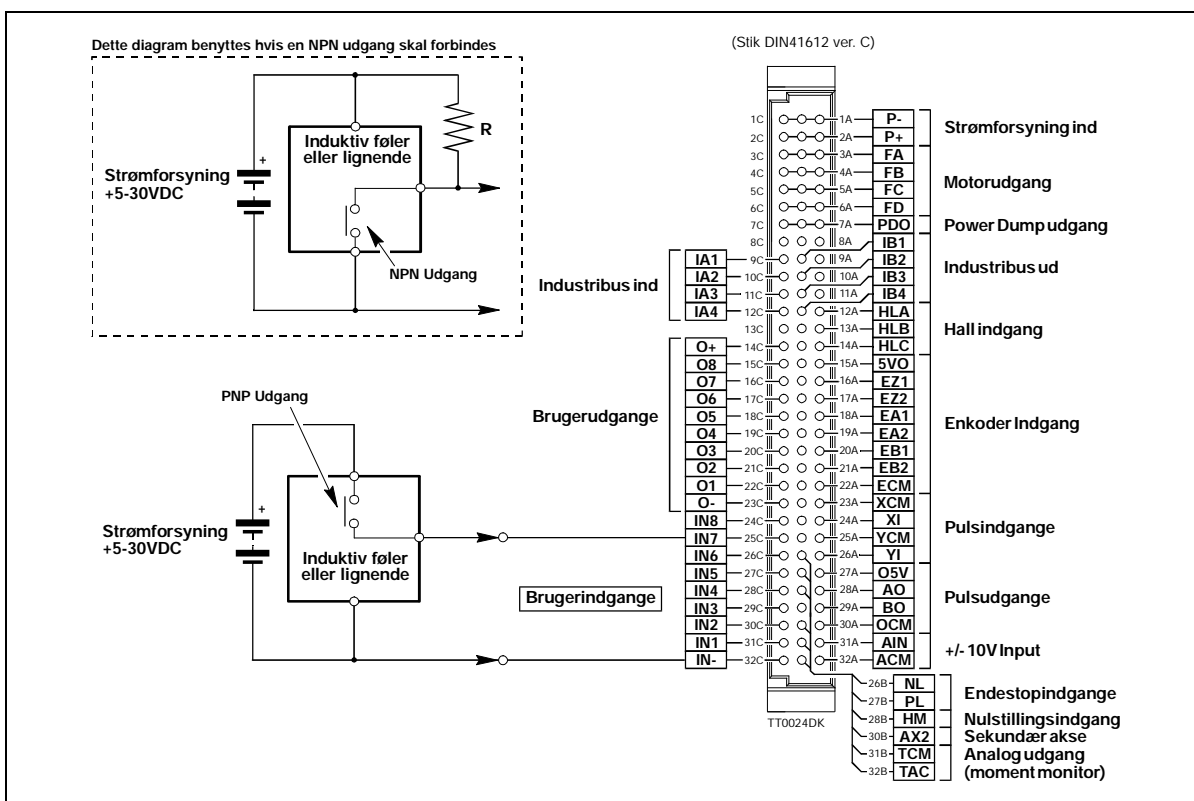
Den aktuelle motors middelstrøm og spidsstrøm skal indstilles via de 2 controllerkommandoer CA og CP - se afsnittet *Justering af motorstrøm* side 139 .

Når der benyttes en standard steppmotor med 200 step pr. omdrejning, skal den benyttede enkoder have en opløsning på mindst 4000 pulser pr. omdrejning. Ligeledes anbefales det at den pågældne enkoder har en index puls. Se også afsnittet *Enkoderindgang* side 29

Se afsnittet *Eksempler på tilslutning af motor* side 145 vedr. tilslutning af diverse fabrikkater motorer.

3.3

Brugerindgange



3.3.1 Generelt om brugerindgange

Controlleren indeholder ialt 8 digitale indgange. Hver indgang kan benyttes til forskellige formål afhængigt af den grundmode, som controlleren er indstillet til at arbejde efter. Indgangene er galvanisk isoleret fra de øvrige kredsløb i controlleren. Alle indgangene har en fælles stelterminal benævnt *IN-*. Bemærk at denne terminal også bliver anvendt med endestop og nulstillingsindgang. Hver indgang kan operere med spændinger i området 5 til 30VDC. Bemærk, at indgangene normalt skal modtage signal fra en PNP udgang, idet indgangene skal tilføres en positiv strøm for at blive aktiveret.

3.3.2 Tilslutning af NPN udgang

Hvis en given indgang kobles til en NPN udgang, skal der forbindes en såkaldt Pull-Up modstand mellem indgangen og + forsyningen - se ovenstående illustration. Modstandens størrelse afhænger af den benyttede forsyningsspænding. Følgende modstande kan anbefales :

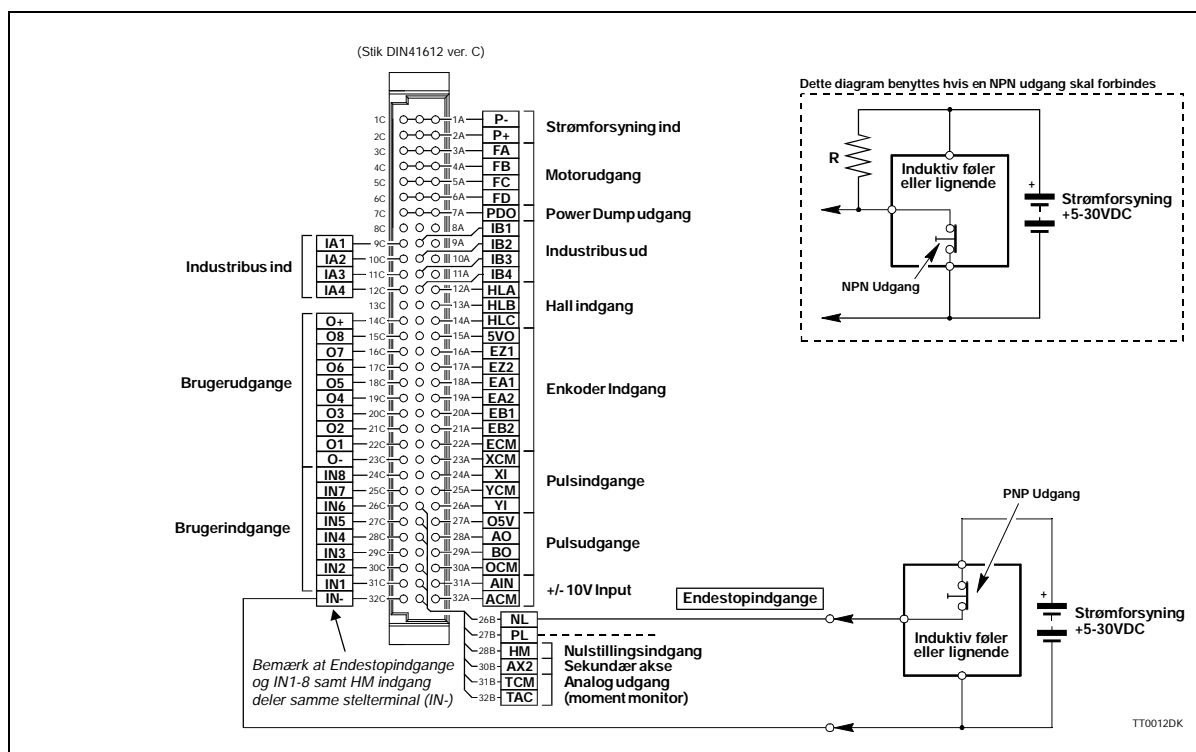
| Forsyningsspænding | Anbefalet modstandsværdi |
|--------------------|--------------------------|
| 5-12VDC | 1kOhm / 0,25W |
| 12-18VDC | 2,2kOhm / 0,25W |
| 18-24VDC | 3,3kOhm / 0,25W |
| 24-30VDC | 4,7kOhm / 0,25W |

3.3.3 Indikation af indgangsniveau

Til indikation af hver indgangs niveau er der en lysdiode på controllerens forplade benævnt IO1, IO2,..... IO8. Disse lysdioder lyser, når den tilhørende indgang er aktiveret. Bemærk at lysdioderne både kan vise status på digitale ind- eller udgange. Kommandoen *LED* benyttes til at vælge hvorvidt ind- eller udgangsstatus ønskes vist.

3.4

Endestopindgange



3.4.1 Generelt om endestopindgange

Controlleren indeholder en øvre og nedre endestopindgang benævnt *NL* (nedre endestop, og *PL* øvre endestop. Indgangene er galvanisk isoleret fra de øvrige kredsløb i controlleren, med undtagelse af *IN1 - IN8*, samt *HM* (nulstillingsindgang). Alle disse indgange har en fælles stelterminal benævnt *IN-*. Endestopindgangene kan operere med spændinger i området 5 til 30VDC. Bemærk, at indgangene normalt skal modtage signal fra en PNP udgang, idet indgangene skal tilføres en positiv strøm for at blive aktiveret. En aktivering af *PL* indgangen, vil stoppe motorkørslen hvis motoren køre i positiv retning. Motoren kan dog køres i negativ retning, selvom *PL* indgangen er aktiveret. En aktivering af *NL* indgangen, vil stoppe motorkørslen hvis motoren køre i negativ retning. Ligesom med *PL* indgangen kan motoren køres i positiv retning selvom *NL* indgangen er aktiveret. Det niveau der skal til for at aktivere endestopindgangene kan indstilles via parametrene *NLS* og *PLS* - se afsnittet *Negativ endestop (NLS)* side 95 eller *Positiv endestop (PLS)* side 99.

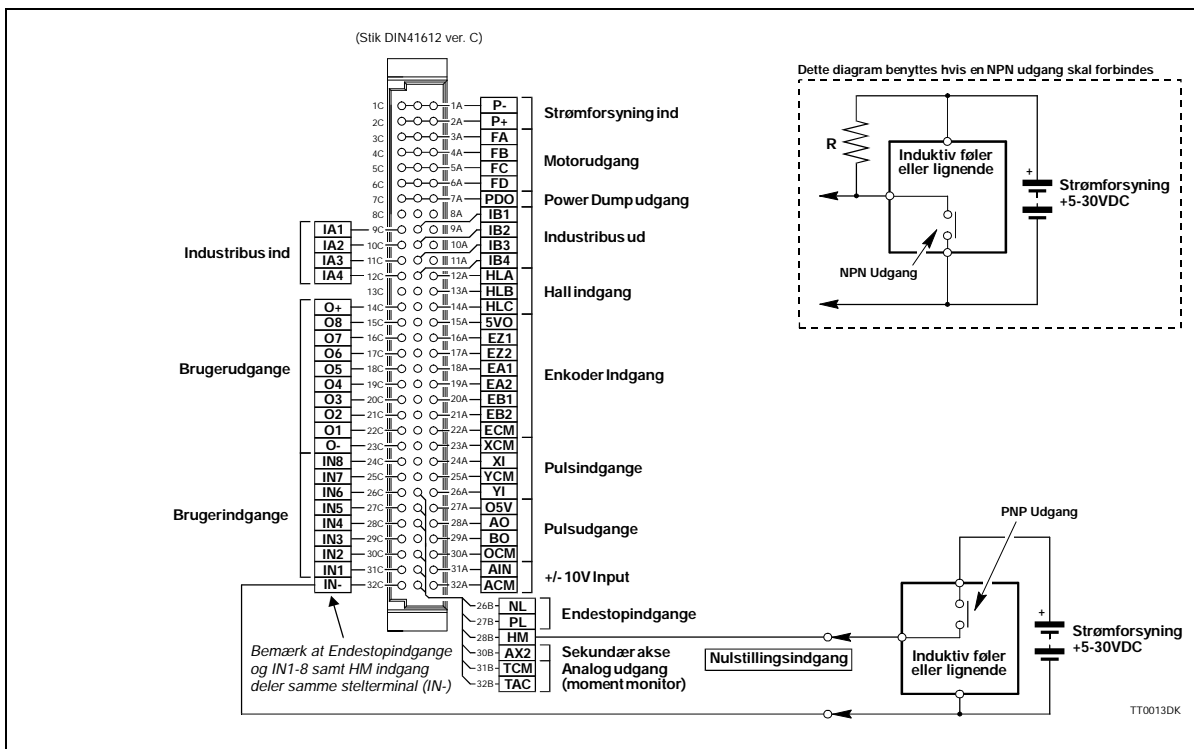
Der vil optræde en fejlmeddelelse i controllerens fejlregister såfremt *NLS* eller *PLS* har været eller er aktiveret. Se afsnittet *Fejlmeddelelser* side 117

3.4.2 Tilslutning af NPN udgang

Hvis en endestopindgang skal kobles til en NPN udgang, skal der forbindes en såkaldt Pull-Up modstand mellem indgangen og + forsyningen - se ovenstående illustration. Modstandens størrelse afhænger af den benyttede forsyningsspænding. Følgende modstande kan anbefales :

| Forsyningsspænding | Anbefalet modstandsværdi |
|--------------------|--------------------------|
| 5-12VDC | 1kOhm / 0,25W |
| 12-18VDC | 2,2kOhm / 0,25W |
| 18-24VDC | 3,3kOhm / 0,25W |
| 24-30VDC | 4,7kOhm / 0,25W |

3.5 Nulstillingsindgang



3.5.1 Generelt om nulstillingsindgang

Nulstillingsindgangen *HM* (Home) bliver anvendt under nulpunktsøgning. Denne nulpunktsøgning vil finde sted efter en af følgende handlinger :

1. Controlleren modtager kommandoen SZ (nulstil). Se afsnittet *Find nulpunkt (SZ)* side 110
 2. Efter kontrolleren tændes (kun hvis XR=1). Se afsnittet *Mekanisk nulstilling* side 67
 3. Hvis kontrolleren er i Mode 3 og register 0 vælges. *Registermode (MO=3)* side 51
- Nulstillingsindgangen benyttes primært hvis kontrolleren bruges til positioneringsformål, dog kan der være specielle anvendelser i hastighedsmode eller momentmode hvor nulstilling også kan være hensigtsmæssig. Indgangen er galvanisk isoleret fra de øvrige kredsløb i kontrolleren, med undtagelse af *IN1 - IN8*, samt *NL* og *PL* (Endestopindgange). Alle disse indgange har en fælles stelterminal benævnt *IN-*. Nulstillingsindgangen kan operere med spændinger i området 5 til 30VDC. Bemærk, at indgangen er konstrueret til at modtage signal fra en PNP udgang, idet indgangene skal tilføres en positiv strøm for at blive aktiveret.

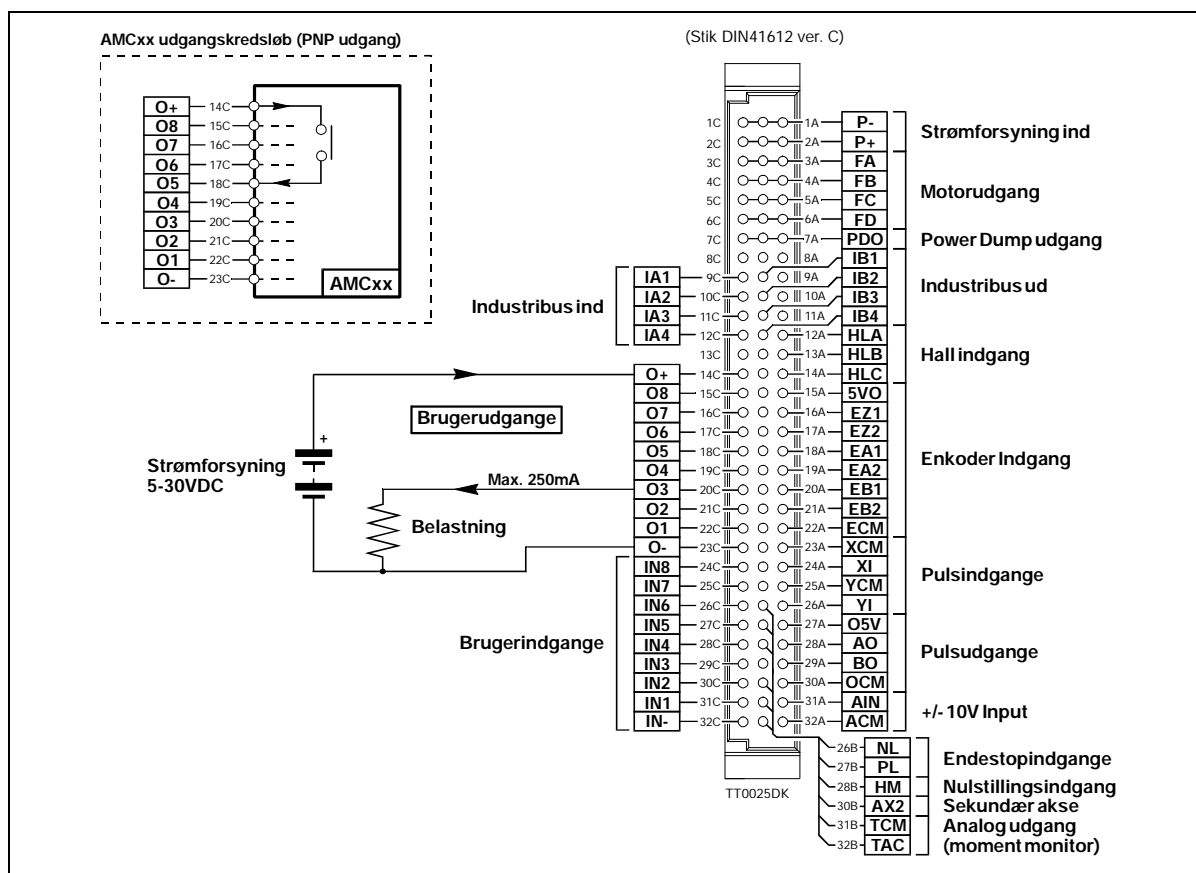
3.5.2 Tilslutning af NPN udgang

Hvis en nulstillingsindgangen skal kobles til en NPN udgang, skal der forbindes en såkaldt Pull-Up modstand mellem indgangen og + forsyningen - se ovenstående illustration. Modstandens størrelse afhænger af den benyttede forsyningsspænding. Følgende modstande kan anbefales :

| Forsyningsspænding | Anbefalet modstandsværdi |
|--------------------|--------------------------|
| 5-12VDC | 1kOhm / 0,25W |
| 12-18VDC | 2,2kOhm / 0,25W |
| 18-24VDC | 3,3kOhm / 0,25W |
| 24-30VDC | 4,7kOhm / 0,25W |

3.6

Brugerudgange



3.6.1 Generelt om brugerudgange

Controlleren indeholder ialt 8 digitale udgange. Hver udgang kan benyttes til forskellige formål afhængigt af den grundmode, som kontrolleren er indstillet til at arbejde efter. Udgangene er galvanisk isoleret fra de øvrige kredsløb i kontrolleren. Udgangskredsløbet skal forsynes af en ekstern strømforsyning for at fungere. Denne strømforsyning tilsluttes mellem terminalerne benævnt O+ og O-. Udgangskredsløbet kan arbejde med spændinger i området 5-30VDC. Hver udgang kan levere 250mA kontinuerligt. Udgangene er alle sourcedrivere, d.v.s. er en given udgang aktiveret, vil der dannes kontakt mellem +forsyningen (O+) og den pågældende udgangsterminal - se ovenstående illustration. Til indikation af hver udgangs niveau er der en lysdiode på kontrollerens forplade benævnt IO1, IO2,..... IO8. Disse lysdioder lyser, når den tilhørende udgang er aktiveret.

Bemærk at lysdioderne både kan vise status på digitale ind- og udgange.

Kommandoen *LED* benyttes til at vælge hvorvidt ind- eller udgangsstatus ønskes vist.

Bemærk ! Lysdioderne indikerer ikke det reelle niveau på udgangene, eftersom de er koblet direkte til den interne mikroprocessor, og ikke har forbindelse til selve udgangsterminalerne.

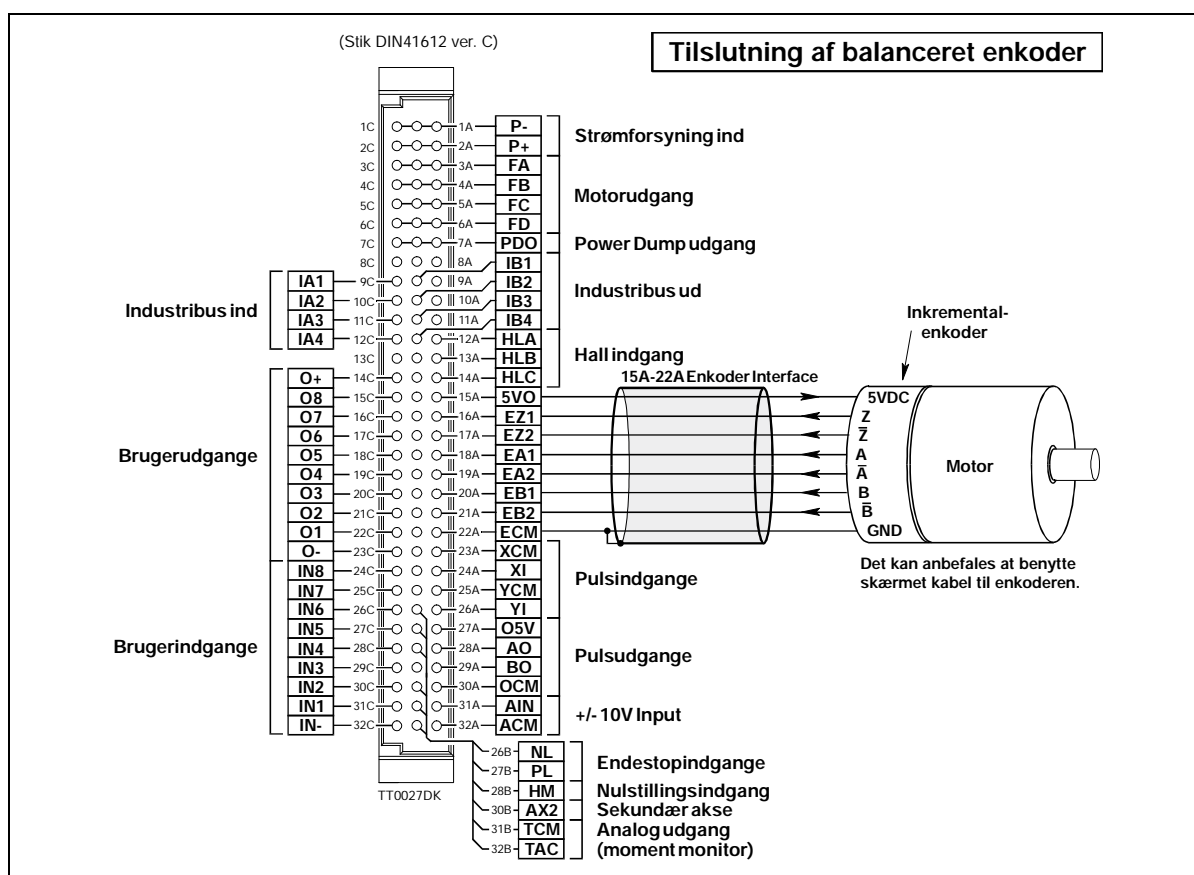
3.6.2 Overbelastning af brugerudgange

Samtlige udgange er kortslutningssikret, hvilket indebærer, at de afbrydes ved kortslutning. Udgangen fungerer først normalt igen, når kortslutningen ophører, og kontrolleren har været strømløs i mindst 5 sekunder. Lysdioden benævnt Out Error lyser, når en eller flere af udgangene er eller har været kortsluttet. Lysdioden vil ligeledes indikere, hvis udgangskredsløbet er blevet overophedet som følge af for stor belastning.

Denne lysdiode sidder placeret på controllerens forplade.

3.7

Enkoderindgang



3.7.1 Generelt om enkoderindgang

Det er obligatorisk at benytte en inkremental-enkoder ligegyldigt om controlleren benyttes med en AC-servomotor eller en stepmotor. Det kan anbefales at benytte en enkoder med index kanal, hvilket vil sige at enkoderen udover A og B kanalen har en 3 kanal der afgiver 1 impuls for hver motoromdrejning. Denne puls benyttes til nulstilling af controllerens kommuterings-elektronik og sikre at en manglende impuls på A eller B kanalen korrigeres. Uden index kanalen vil controlleren over længere tids drift gå i fejl som følge af en skæv kommutering af motoren. Alternativt kan systemets effektivitet falde.

Den tilsluttede enkoder detekterer den tilsluttede motors hastighed og position. Næsten alle enkodere kan tilsluttes, hvis de har en af følgende udgange - NPN-, PNP-, Push-Pull- eller Balanceret udgang.

Enkoderindgangen kan aftaste et enkodersignal op til 500kHz. Spændingen på enkodersignalerne skal ligge indenfor 0 til 5V.

Bemærk ! - Kablet mellem enkoder og controller skal altid være skærmet og skærmen må kun have forbindelse på controllerens enkoder-stelterminal (ECM).

Vedrørende generel enkoderopsætning se afsnittet *Indstilling af enkoderopløsning* side 134.

3.7.2 Enkodere med balanceret udgang

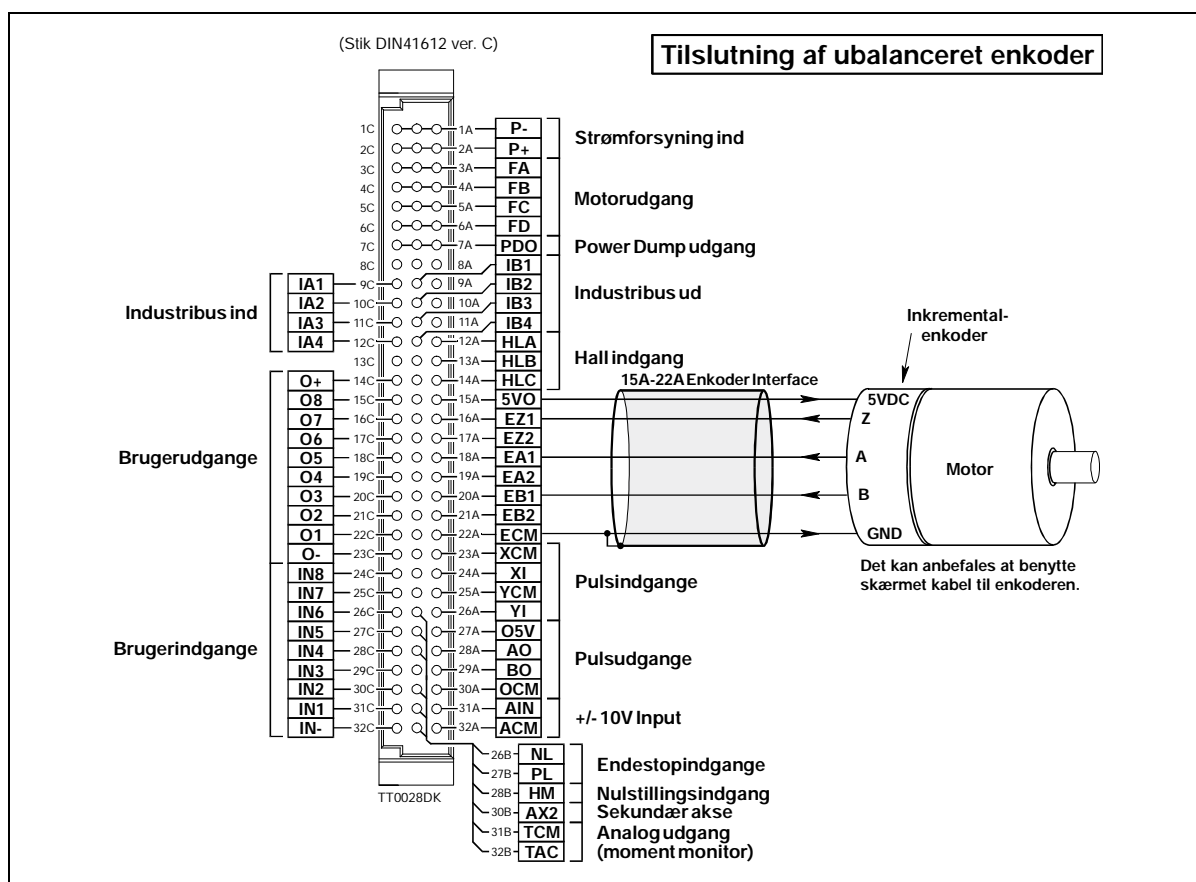
Hvis en enkoder med balanceret udgang skal tilsluttes, følges ovenstående illustration. Bemærk at det anbefales at benytte enkoder med balancerede udgange.

Det anbefales at bruge skærmet kabel med minimum 0,3mm² ledere.

Enkoderen må under ingen omstændigheder dele kabel med andre signalledninger, idet disse kan have fatal indflydelse på enkoderens signaler.

3.7

Enkoderindgang



3.7.3 Generelt om enkodere med ubalanceret udgang

Controlleren kan som nævnt køre med næsten alle typer enkodere, også enkodere der har ubalancerede udgange.

Visse type enkodere har en NPN eller en PNP udgang. Controlleren har til dette formål kommando *ET*, der benyttes til at indstille enkoderindgangen til den aktuelle enkodertype.

Benyttes enkodere med balancerede/ubalancerede udgange af typen *push-pull* eller benævnt *source/sink* kan der ses bort fra denne indstilling.

3.7.4 Specielle enkodere/sensore

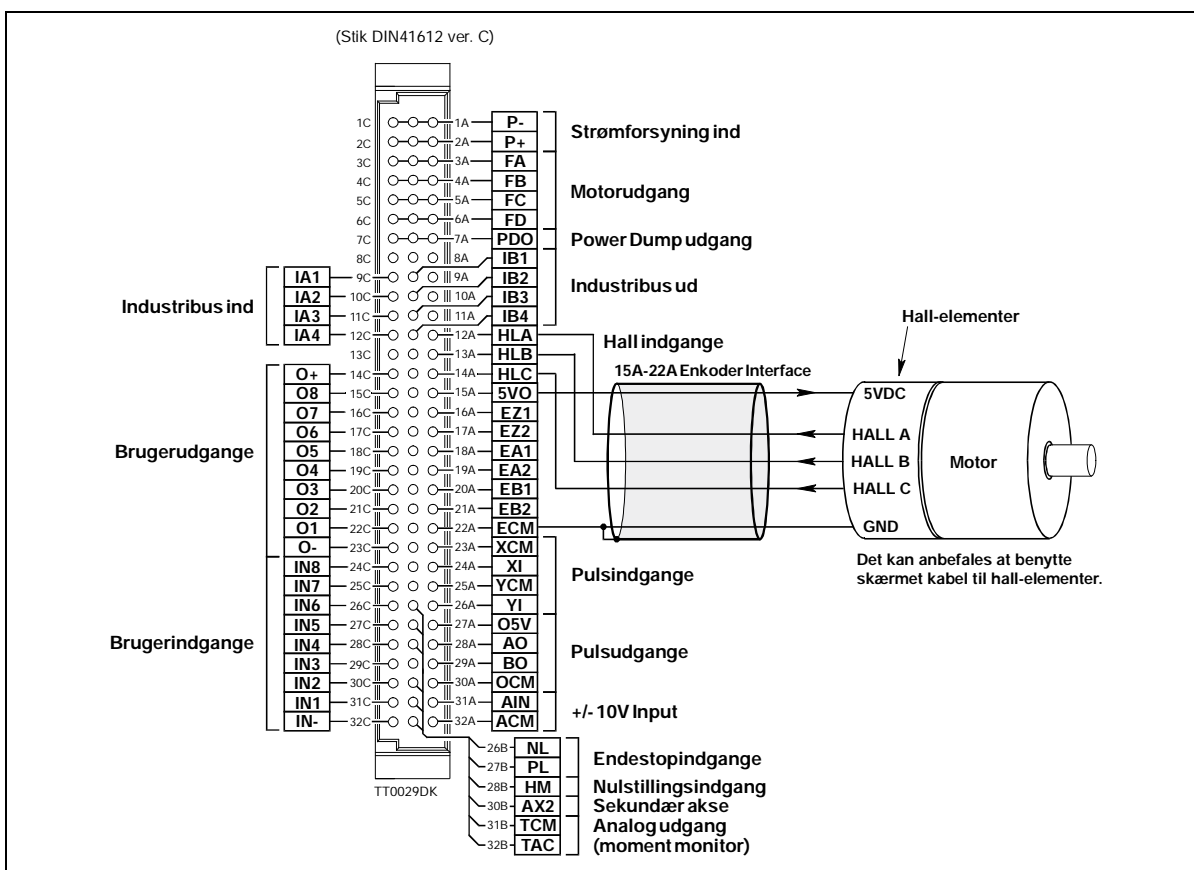
Der er planlagt lancering af flere typer adaptormoduler for andre typer enkodere/sensore. Kontakt evt. JVL Industri Elektronik angående tilslutning af andre typer enkodere/sensore.

Det anbefales at bruge skærmet kabel med minimum 0,3mm² parsnoede ledere.

Enkoderen må under ingen omstændigheder dele kabel med andre signalledninger, idet disse kan have fatal indflydelse på enkoderens signaler.

3.8

Hall-indgang



3.8.1 Generelt om Hall-indgang

Controlleren rummer 3 indgange for tilslutning af en Hall-sensor. Denne feature anvendes kun, hvis det ønskes at motoren under opstart af controlleren, ikke bevæger sig. Hall-indgangen kan kun benyttes med 3 fasede motorer og ikke stepmotorer.

Næsten alle Hall-følere kan tilsluttes, hvis de har en af følgende udgange - NPN-, PNP-, eller Push-Pull- udgang.

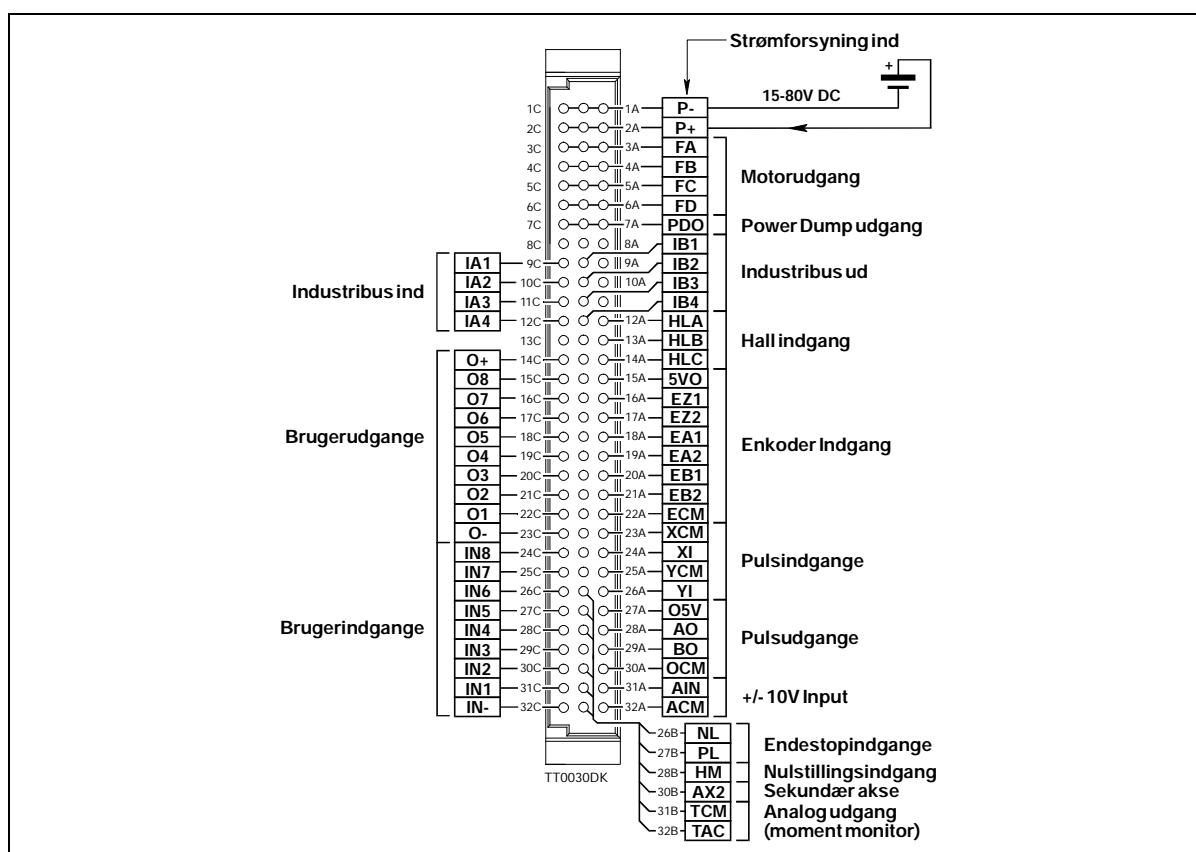
Hall-følernes signaler skal være indenfor spændingsområdet 0 til 5V.

Bemærk ! - Kablet mellem Hall-føler og controller skal altid være skærmet og skærmen må kun have forbindelse på controllerens enkoder/hall-stelterminal (ECM).

Vedrørende generel Hall-op sætning se afsnittet *Indstilling af Hall-element* side 141.

3.9

Strømforsyning



3.9.1 Strømforsyning generelt

Strømforsyning af controlleren er forholdsvis enkel idet der på controller type AMC10B, AMC10C og AMC12C kun kræves en forsyningspænding i området 15-80VDC. Controller type AMC11B og AMC12B har indbygget strømforsyning, og skal derfor tilsluttes 230VAC - se beskrivelse på side 34.

3.9.2 Strømforsyning af AMC10B, AMC10C og AMC12C

For at gøre strømforsyningen så enkel som mulig skal der kun tilsluttes én forsyningspænding. Den interne strømforsyning sørger for de nødvendige spændinger til driver og kontrolkredsløb m.v. Af hensyn til driverens virkemåde anbefales det, at den eksterne strømforsyning har en kondensator på mindst 2000-5000 μ F. Ligeledes kan det anbefales, at ledningerne mellem strømforsyning og controller er mindst 1,5mm. Hvis driverens forsyningspænding kommer under 12V, vil det interne reset-kredsløb nulstille driveren. Af denne grund bør det sikres, at forsyningspændingen forbliver minimum 12-15V, selvom f.eks. netspændingen falder.

3.9.3 Jordforbindelse

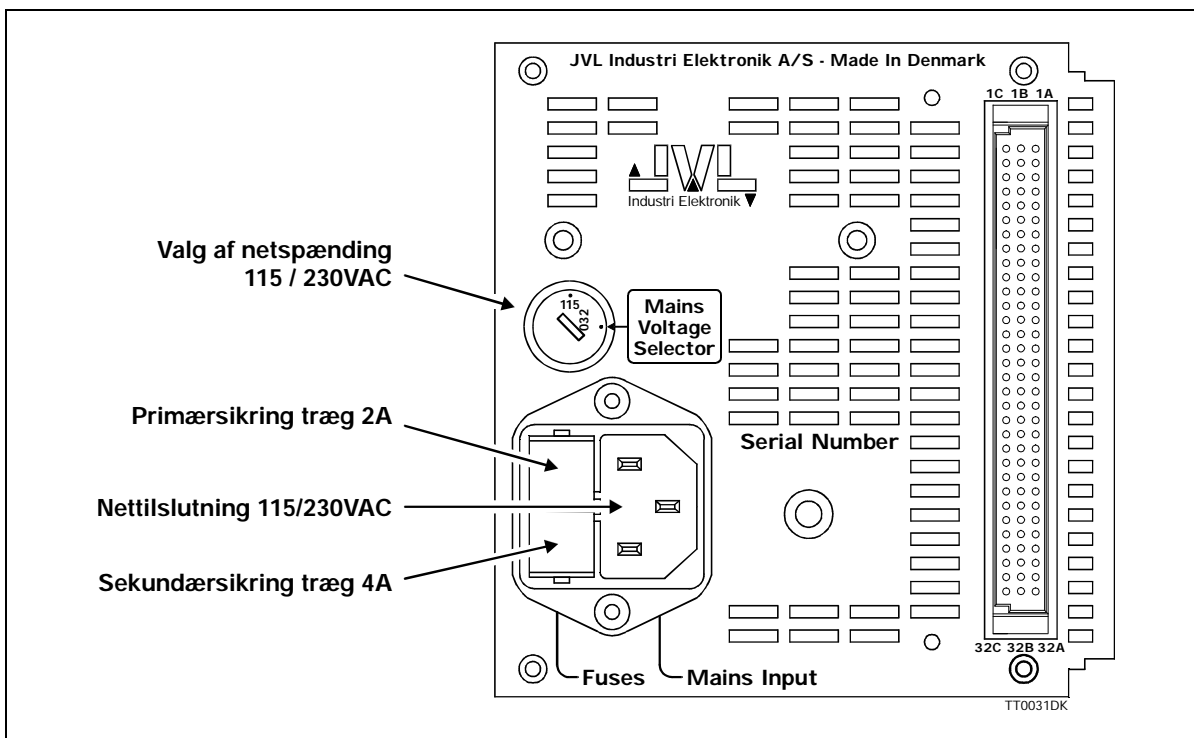
For at sikre ordentlig stel/jord-forbindelse er chassis, jord i netbrønd og P- (minus) forbundet sammen internt i controlleren.

3.9.4 Strømforsyningsfejl

Controlleren er sikret mod fejlpolarisering og overspænding. Opstår en overspænding på controllerens forsyning, eller fejlpolariseres forsyningen, vil controllerens interne sikring brænde af. Sikringen kan kun udskiftes på et autoriseret serviceværksted. Bemærk at AMC11B har ekstern sikring, se beskrivelse på side 34.

3.9

Strømforsyning



3.9.5 Strømforsyning af AMC11B og AMC12B

For at gøre strømforsyningen så enkel som mulig skal der kun tilsluttes én forsyningsspænding. Den interne strømforsyning sørger for de nødvendige spændinger til driver og kontrolkredsløb m.v. Denne strømforsyning kan levere 160W kontinuerligt, men tillader 300W i spidsbelastning i forbindelse med acceleration/deceleration af motor.

Controlleren kan forsynes med enten 115VAC eller 230VAC (+/-10%). Tilslutningen foretages på controllerens bagside hvor den ønskede netspænding også kan indstilles. Den interne strømforsyning er 80VDC nominelt.

3.9.6 Strømforsyningsfejl

Controlleren er sikret mod overspænding, idet et internt kredsløb kortslutter forsyningen og dermed brænder sekundær og evt. primærsikring af. Controlleren rummer et netstøjfilter der fjerner eventuelle transienter, der måtte opstå på forsyningen.

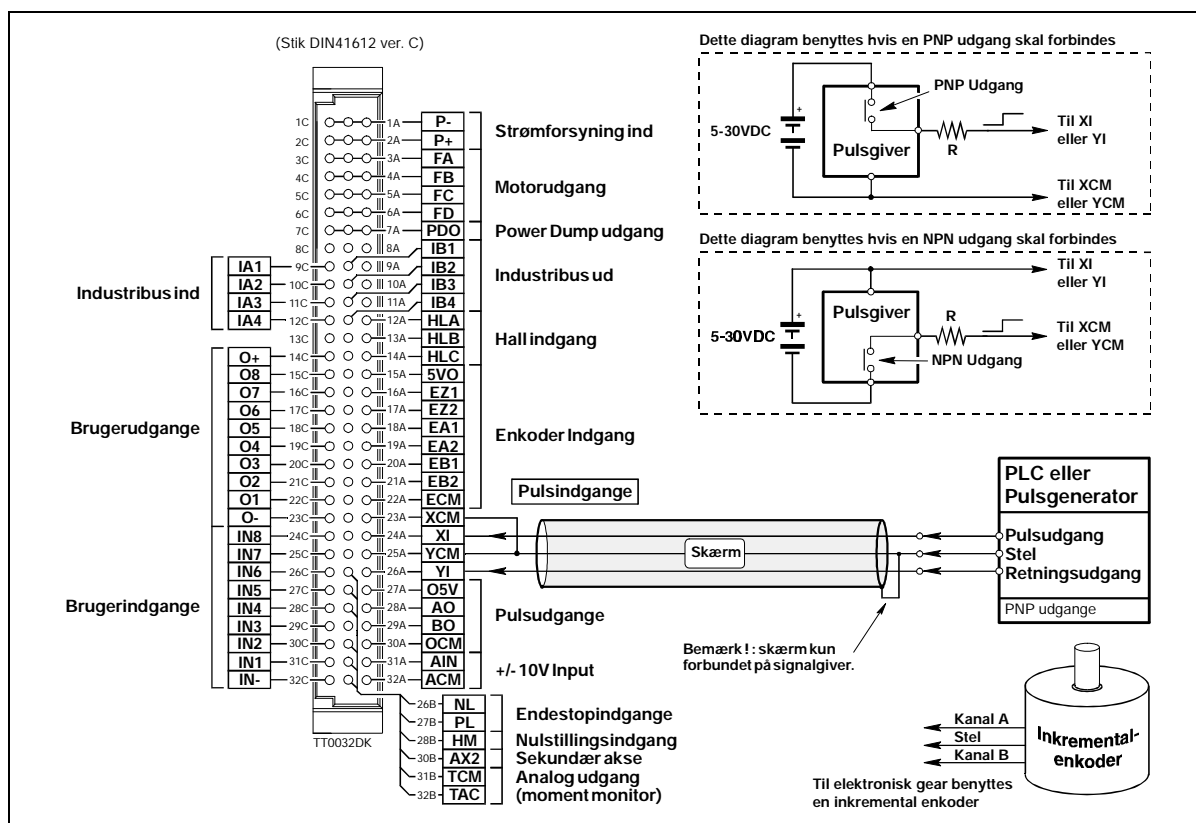
3.9.7 Jordforbindelse

For at sikre ordentlig stel/jord-forbindelse er chassis, jord i netbrønd og P- (minus) forbundet sammen internt i controlleren.

3.9.8 Udbygning af strømforsyning

Hvis den indbyggede strømforsyning på 160W ikke er tilstrækkelig kan der suppleres med en ekstern forsyning som tilsluttes iflg. tegning på side 33. Bemærk at den eksterne strømforsyning skal have samme spænding som den interne strømforsyning nemlig 80V nominelt.

3.10 Pulsindgange



3.10.1 Generelt om pulsindgange

Pulsindgangene benyttes i sammenhæng med mode 1. Hver gang indgangene tilføres en spændingsimpuls, drejer motoren en bestemt vinkel. Denne vinkel er bestemt med *GEAR* kommandoen samt den anvendte enkoder opløsning. Begge indgange har indbygget et støjfilter, der filtrerer alle frekvenser over 1MHz bort. Skitsen på næste side illustrerer minimumstider på de tilsluttede signaler. Bemærk, hvis den kilde, der afgiver puls og/eller retningssignalet, er af typen PNP, skal indgangene forbindes på en bestemt måde. Ligeledes skal indgangene forbindes på en bestemt måde, hvis der er tale om en signalgiver af NPN typen. Se ovenstående illustration. Det kan anbefales at benytte skærmet kabel.

3.10.2 Indgangsspænding

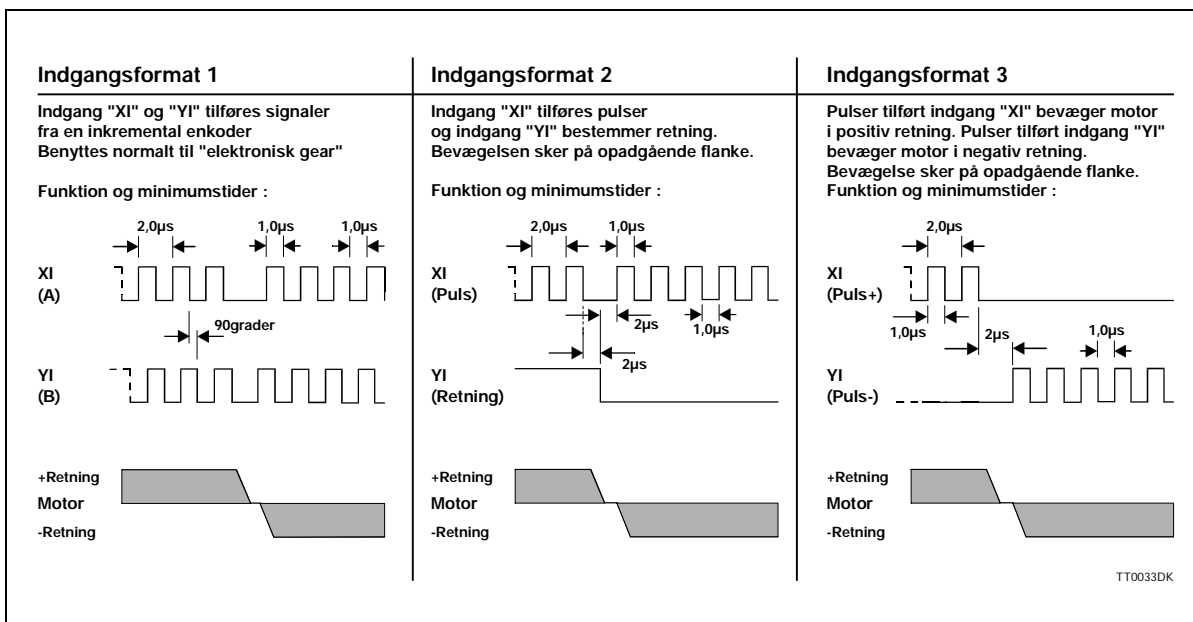
Indgangene er som standard dimensioneret til at operere med spændinger på 5V. Benyttes højere indgangsspænding, skal der indskydes en modstand R som vist på ovenstående illustration. Modstandens størrelse bestemmes af nedenstående skema.

| Spænding | Modstandsværdi |
|----------|---------------------|
| 5-8V | 0 Ohm (kortslettet) |
| 8-12V | 470 Ohm / 0,25W |
| 12-18V | 1,2 kOhm / 0,25W |
| 18-24V | 1,8 kOhm / 0,25W |
| 24-30V | 2,2 kOhm / 0,25W |

Se endvidere beskrivelse af mode 1 - *Kom i gang - Gearmode (Mode 1)* side 5 .

3.10

Pulsindgange



3.10.3 Pulsindgangenes format

Pulsindgangene kan indstilles til 3 formater - se ovenstående tegning.

De forskellige formater indstilles via kommandoen *PIF*. Se afsnittet *Pulsindgangsformat (PIF)* side 98. De 3 forskellige formater har følgende funktion.

For yderligere detaljer se afsnittet *Gearmode (MO=1)* side 49.

3.10.4 Indgangsformat 1

Dette format er normalt anvendt hvis controlleren skal benyttes i et system som elektronisk gear. På indgangen tilsluttes en inkremental enkoder som aftaster den bevægelse som motoren skal køre efter. *GEAR* kommandoen sættes til det ønskede gearingsforhold og *PIF* kommandoen indstilles til format 1 (*PIF=1*). Indgangskredsløbet i enkoderen vil herefter dekode de indkomne pulser iflg. ovenstående illustration. Se også kommandoen *PRM*.

3.10.5 Indgangsformat 2

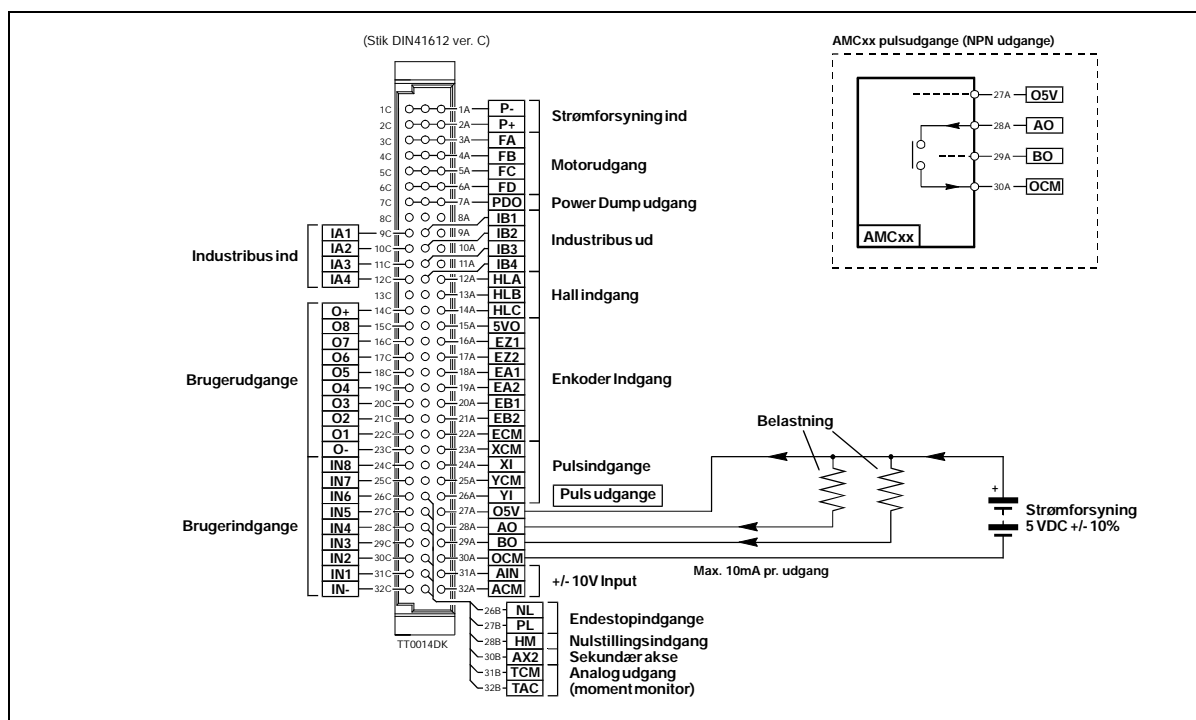
Dette format benyttes normalt hvis systemet skal modtage pulser fra et PLC eller PC aksekort. Controlleren vil fungere som et stepmotorsystem, og motoren vil derfor bevæge sig en given vinkel hver gang der modtages en puls på XI indgangen. Spændingsniveauet på YI bestemmer hvilken vej motoren skal bevæges.

3.10.6 Indgangsformat 3

Dette format svarer til format 2, dog er motorens retning bestemt af hvilken indgang (XI eller YI) der tilføres pulser.

3.11

Pulsudgange



3.11.1 Generelt om pulsudgange

De 2 pulsudgange *AO* og *BO* udsender 2 pulssignaler der kan konfigureres til enten at være motorenkoderen (*EA* og *EB*) eller det signal der bliver tilsluttet pulsindgangen (*XI* og *YI*). Pulsudgangen anvendes typisk til følgende anvendelser :

1. Master/slave system hvor master-controllerens pulsudgange forbindes til slavecontrollerens pulsindgange. Slavecontrolleren vil således følge master-controllerens bevægelser.
2. PC-system. En controller der kobles til et PC-kort via analogindgangen eller pulsindgangen, og udelukkende fungerer som hastighedsstyring. Pulsudgangen forbindes til PC-kortet, og sørger for at information om aktuel hastighed og position løbende bliver sendes til PC-kortet.

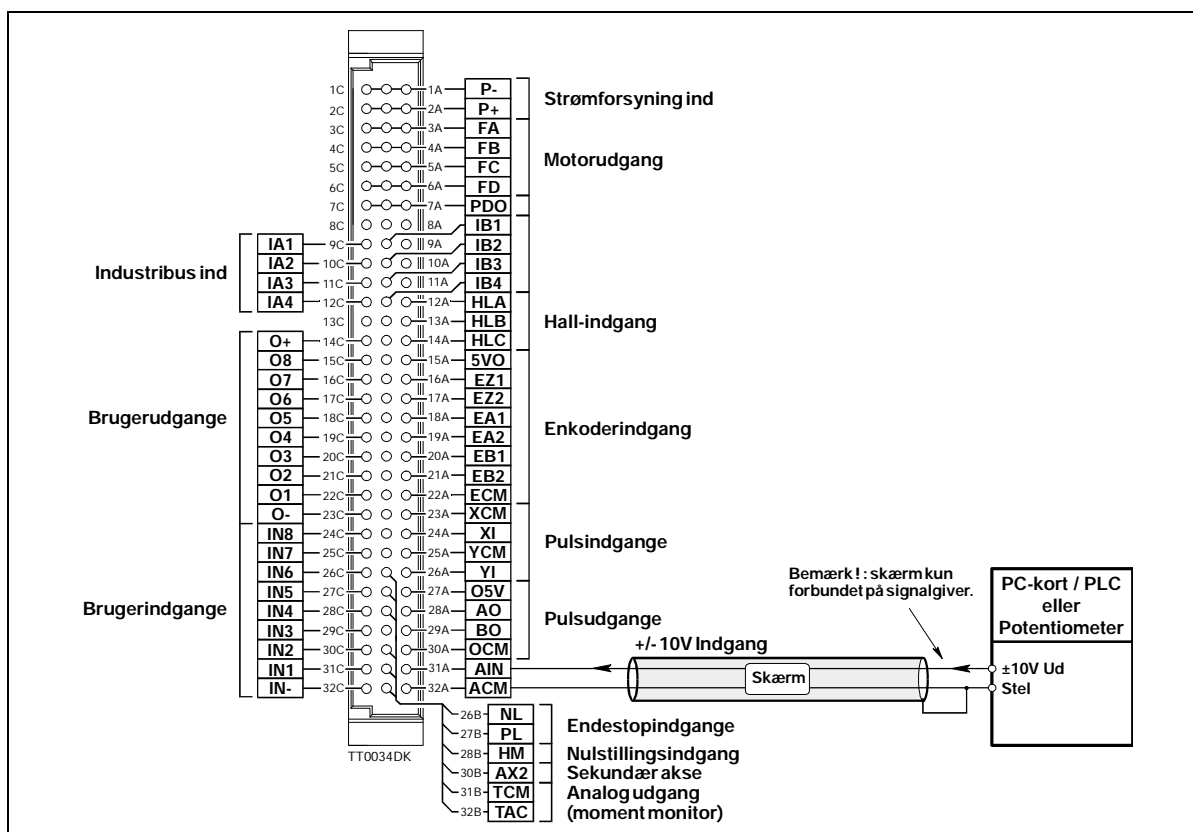
Udgangene er galvanisk isoleret fra de øvrige kredsløb i controlleren. Udgangskredsløbet skal forsynes af en ekstern 5V strømforsyning for at fungere. Denne strømforsyning tilsluttes mellem terminalerne benævnt *O5V* og *OCM* (se illustration).

Hver udgang kan levere op til 10mA og arbejde ved frekvenser op til 500kHz. Begge udgange er af NPN typen, d.v.s. er en given udgang aktiveret, vil der dannes kontakt mellem -forsyningen (*OCM*) og den pågældende udgangsterminal - se ovenstående illustration.

Bemærk at pulsudgangenes format skal indstilles med kommandoen *POF*, se afsnittet *Pulsudgangsformat (POF)* side 101

3.12

Analogindgang



3.12.1 Generelt om analogindgang

Analogindgangen benyttes bl.a., når controlleren arbejder i hastighedsmode (mode 4) eller momentmode (mode 5).

I disse modes styres motoren til den hastighed, eller det moment som indgangsspændingen på analogindgangen angiver.

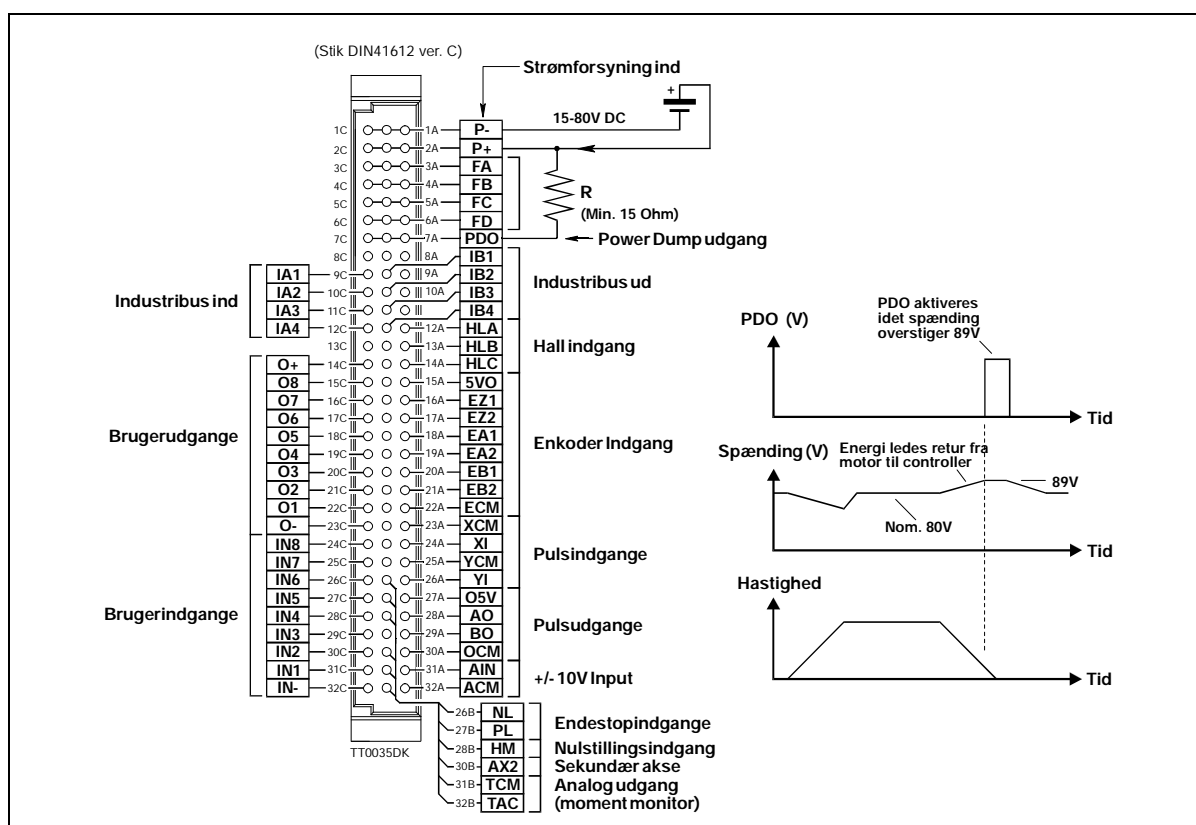
Analogindgangen accepterer indgangsspændinger mellem -10V og +10V og er galvanisk isoleret fra alle andre ind- og udgange, inkl. forsyningsterminaler. Bemærk dog, at den deler intern forsyning med RS232 interfacet, og derfor ikke er galvanisk isoleret fra dette. Analogindgangen er beskyttet imod overspænding op til 100V spids og har indbygget et filter, der fjerner overlejet støj på indgangssignalet.

Benyt altid skærmet kabel fra den kilde, der styrer analogindgangen, idet motor m.v. nemt kan influere det analoge signal og give ustabilitet.

I controlleren sidder en analog til digital konverter (ADC), der konverterer den målte værdi til den interne mikroprocessor. Konverteren har en opløsning på 11 bit, hvilket giver et samlet arbejdsområde på 2048 trin.

3.13

Power dump udgang



3.13.1 Generelt om "Power Dump" udgang

Hvis controlleren benyttes i systemer, hvor der skal drives meget store inertibelastninger (svinghjul etc.), kan det være et problem, at der ved deceleration bliver sendt energi tilbage fra motoren til controllerens forsyning. Dette vil medføre, at forsyningsspændingen vokser til et kritisk højt niveau over controllerens tilladte arbejdsområde. Som løsning på dette kan "Power Dump" udgangen (PDO) benyttes. Denne udgang kan lede overskydende energi ud til en ekstern modstand, og derved undgå, at controlleren lukker ned og melder fejl. Bemærk at en sænkning af hastighed VM, acceleration AC, eller peakstrøm CP, kan mindske den returnerede energimængde fra motoren.

3.13.2 Detaljeret beskrivelse af "Power Dump".

PDO Modstandens størrelse vil afhænge af mange parametre bl.a.: motorens maksimale omdrejningstal, forsyningsspændingen, hvor hurtigt der decelereres m.v. Det kan dog anbefales, at shuntmodstanden minimum vælges til 15 Ohm / 50W. Modstandens mærkeeffekt kan dog være mindre eller højere afhængigt af belastningen.

1. Når controlleren registrerer at forsyningsspændingen vokser over 89V, aktiveres PDO-udgangen, og lysdioden *Error* lyser. Controlleren sender automatisk fejlmeddelelsen *E24: Supply Voltage exceeds 89 V*.
2. Hvis indkoblingen af PDO udgangen og dermed PD-modstanden ikke stopper væksten af forsyningsspændingen sker følgende; Når forsyningsspændingen passerer 95V, lukker controlleren helt ned og motoren kortsluttes, således at den bringes til øjeblikkelig standsning. Controlleren sender fejlmeddelelsen *E29: Supply Voltage exceeds 95 V*. PDO udgangen aktiveres indtil spændingen falder under 89V, og controlleren vil forblive i denne fejltilstand indtil den får kommandoen *RESET* - se afsnittet *Genstart controller (RESET)* side 107. Hvis forsyningsspændingen stadig vokser p.g.a. andre omstændigheder vil controllerens interne overspændingskredsløb

træde i funktion og kortslutte forsyningen således at den interne sikring brænder af.

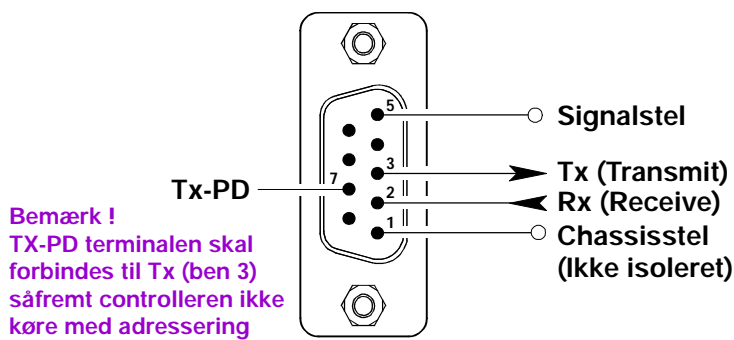
3.14

RS232 Interface

3.14.1 Interfaceforbindelser

Interfacet kører efter den meget anvendte RS232C standard, hvilket er en stor fordel, idet alle Personal Computers og standard terminaler har mulighed for at køre efter denne standard. De 3 ledere Rx, Tx og stel anvendes. Kabellængden bør ikke overstige 10 meter.

Controller interface:



3.14.2 Kommunikationsformat

Controlleren kører med følgende format: (1 startbit), 7 databit, Odd paritet, 1 Stop bit
Bemærk, at der altid er en startbit i RS232C/V24 protokollen.

3.14.3 Kommunikationshastighed

Controlleren kører med en fast kommunikationshastighed (Baud rate) på 9600 Baud. Hastigheden skal sættes på den terminal eller PC'er, som bruges til at kommunikere med controlleren.

3.14.4 Kommandoforamt

Når der sendes kommandoer til controlleren, er det en forudsætning, at der følges et bestemt format:

[Adresse] Kommando [=Argument] ; Kommando [=Argument]] [Checksum] <CR>

Tekst der er i [] kan, afhængig af opsætning, undværes eller medtages.

Adresse: Adresse skal bruges, når der anvendes flere controllere på samme kommunikationslinje. Se kommandoen også ADDR.

Kommando:Selve kommandoen.

Argument: Den efterfølgende talværdi for kommandoen. Et argument skal altid startes med en "=". Visse kommandoer benytter ikke argumenter. F.eks. kommandoer der viser opsætning.

;

Der kan sendes flere kommandoer på samme kommandolinje. De enkelte kommandoer skal adskilles af et ";;".

Checksum: I tilfælde af lange kommunikationslinier kan der benyttes checksum for ekstra sikkerhed for, at kommandoen er modtaget korrekt. I tilfælde af fejl modtages fejlmeddelelse E9, og kommandoen må derefter sendes igen. Se også kommandoen CHS.

<CR>: Har værdien 13 (ASCII). Dette er afslutningen af en kommandolinje.

3.14

RS232 Interface

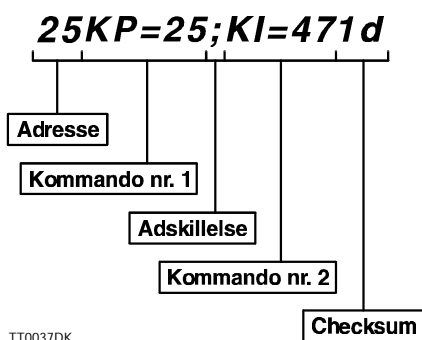
3.14.5 Synkronisering

Når der kommunikeres med controlleren, skal hver kommando afsluttes med et <CR>, som har værdien 13 (ASCII) eller et “;”. Dette fortæller controlleren, at kommandostrengen er fuldendt og en oversættelse kan påbegyndes. I det tilfælde, hvor der benyttes checksum, vil fortolkningen ikke begynde før hele kommandolinien er modtaget, dvs. afsluttet med et <CR>. Der må ikke være mere end 80 karakterer på en kommandolinie.

Hvis controlleren er indstillet til at kører med adressering (*ADDR>0*), kan strengen afsluttes med “;”

3.14.6 Checksum

I industriel sammenhæng vil der ofte optræde elektrisk støj fra f.eks. elektromotorer. Denne støj kan komme helt vilkårligt, og selv en effektiv elektrisk filtrering vil ikke kunne fjerne denne støj 100%. Der er således mulighed for at benytte checksum på kommandostrengen. En typisk kommandolinie kan se ud som følgende:



Der kommunikeres med adressering i dette tilfælde adresse 25. To kommandoer sendes adskilt af “;” og der sendes tilsidst en checksum. Checksummen består af to karakterer. Checksummen udregnes som en ‘simpel’ checksum og udregningen sker på følgende måde. Først findes ASCII værdien for hver af de karakterer, der indgår i kommandostrengen. Disse lægges sammen og af resultatets hexadecimale værdi benyttes de to mindst betydende cifre (mindst betydende byte).

De to mindst betydende cifre laves om til ASCII værdier og sendes med. Det egentlige regnestykke ser således ud:

$$50+53+75+80+61+50+53+59+75+73+61+52+55 = 797 \text{ (decimant)} = 31d \text{ (hexadecimant)}$$

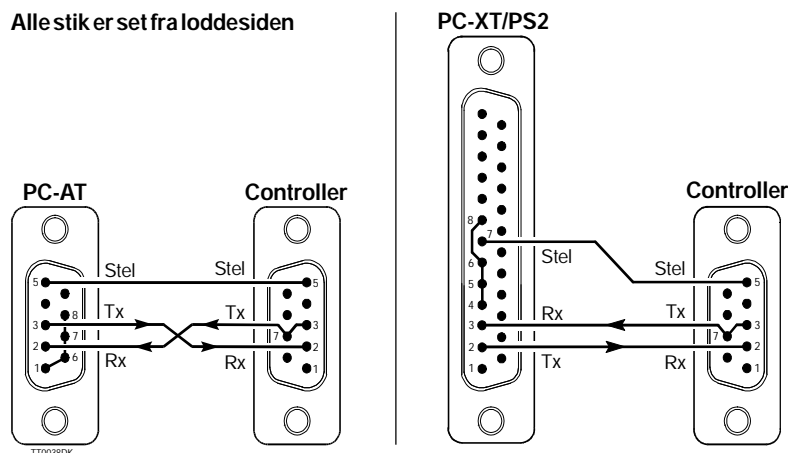
Checksummen bliver således 1d der sendes som ASCII 49 (decimant) og 100 (decimant). Hex. karakterne a-f kan også sendes som store bogstaver, dvs. d kan også sendes som ASCII 68 (decimant).

I tilfælde af, at kommandostrengen bliver påvirket under transmissionen, vil checksummen ikke passe, og controlleren vil melde “E9”, hvilket betyder checksumfejl. Kommandostrengen skal i så fald sendes en gang til. Checksumsfunktionen aktiveres med kommandoen CHS.

3.14 RS232 Interface

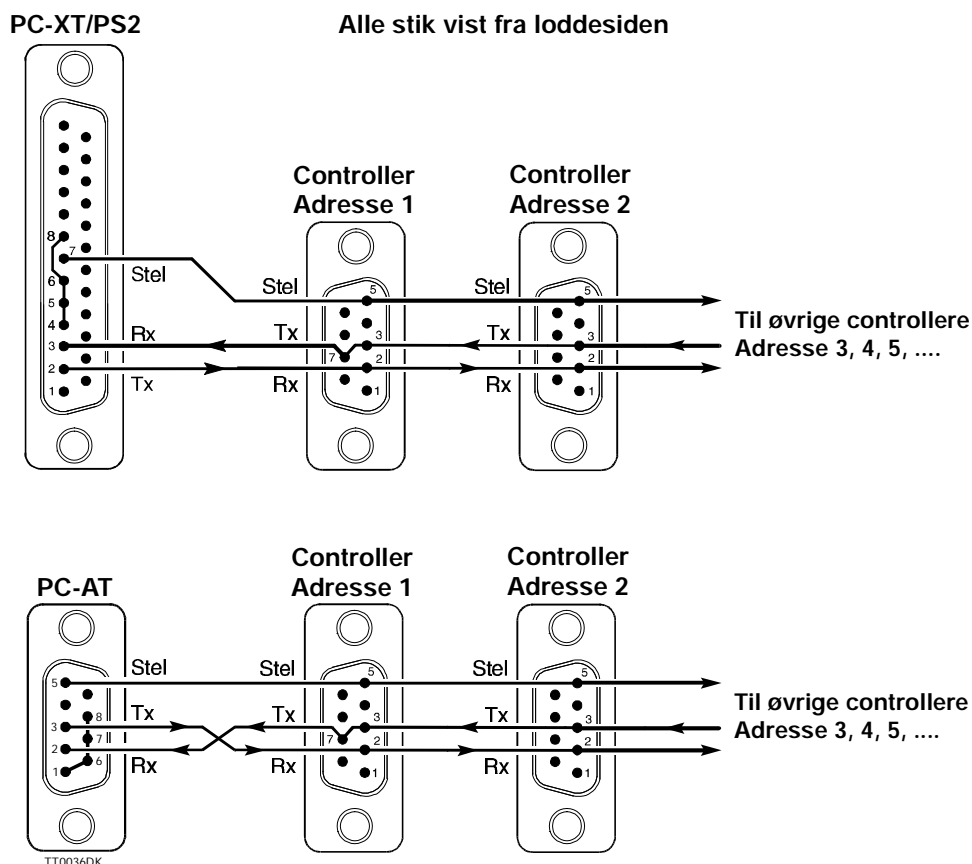
3.14.7 Forbindelse til PC

Skal man kommunikere fra en PC'er, kan disse skemaer anvendes. Tegningerne viser henholdsvis forbindelser mellem controller og IBM AT eller IBM-XT/PS2 :



3.14.8 Forbindelse til PC af flere controllere

Ved tilkobling af flere controllere (adressering) til PC, kan nedenstående forbindelses diagrammer anvendes. Bemærk at Tx (ben 3) skal forbindes til TX-PD (ben 7) på en af de controllere der indgår i systemet. Tegningerne viser henholdsvis forbindelser mellem controllere og IBM AT eller IBM-XT/PS2 :

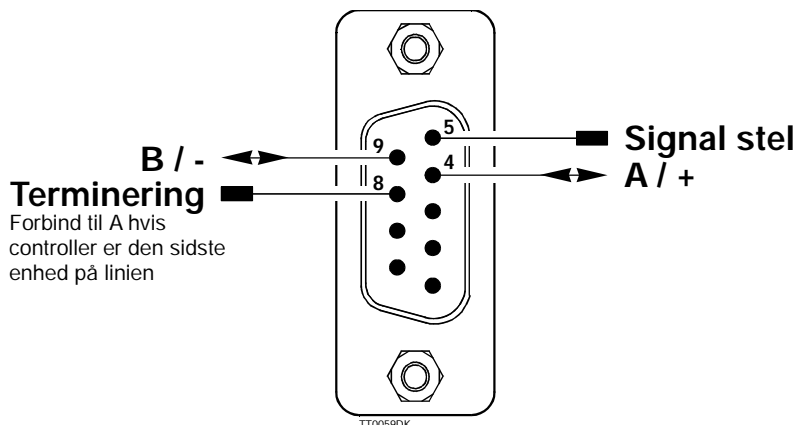


Bemærk at controllerenes adresse, skal ikke nødvendigvis være i nummerorden.

3.15

RS485 Interface

Indeholder udover det normale RS232 interface også RS485 interface. RS485 interfacet er primært tænkt anvendt i systemer hvor 1 til 32 kontrollere forbindes til den samme kommunikationslinie i elektrisk støjfyldte omgivelser.

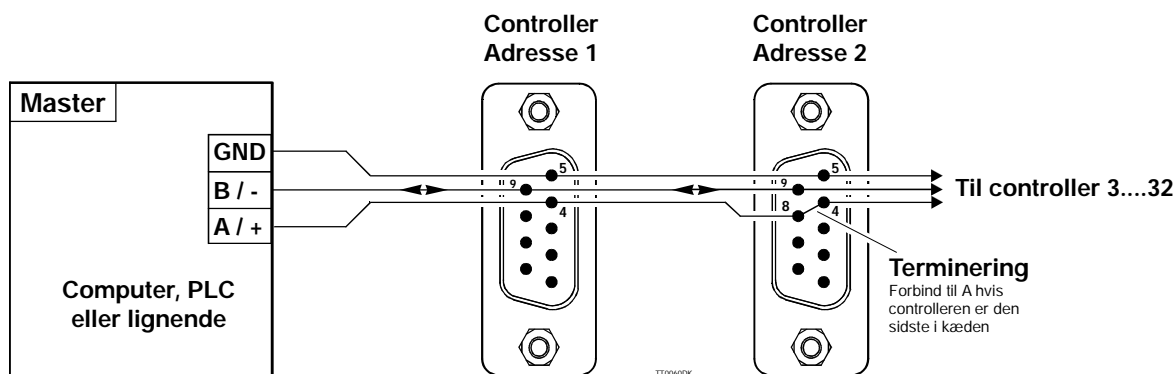


Kommunikations protokollen er præcis den samme som ved brug af RS232. Den eneste forskel ligger i at RS485 benytter balancerede signallinier og at kommunikationen derfor kun køre halv duples hvilket betyder at controlleren ikke kan sende og modtage samtidigt som ved brug af RS232.

RS485 interfacet gør det muligt at forbinde op til 32 enheder på de samme interface ledninger.

Den sidste controller der forbindes på linien skal have forbundet terminalen benævnt Terminering (ben 8) til terminal A (ben 4).

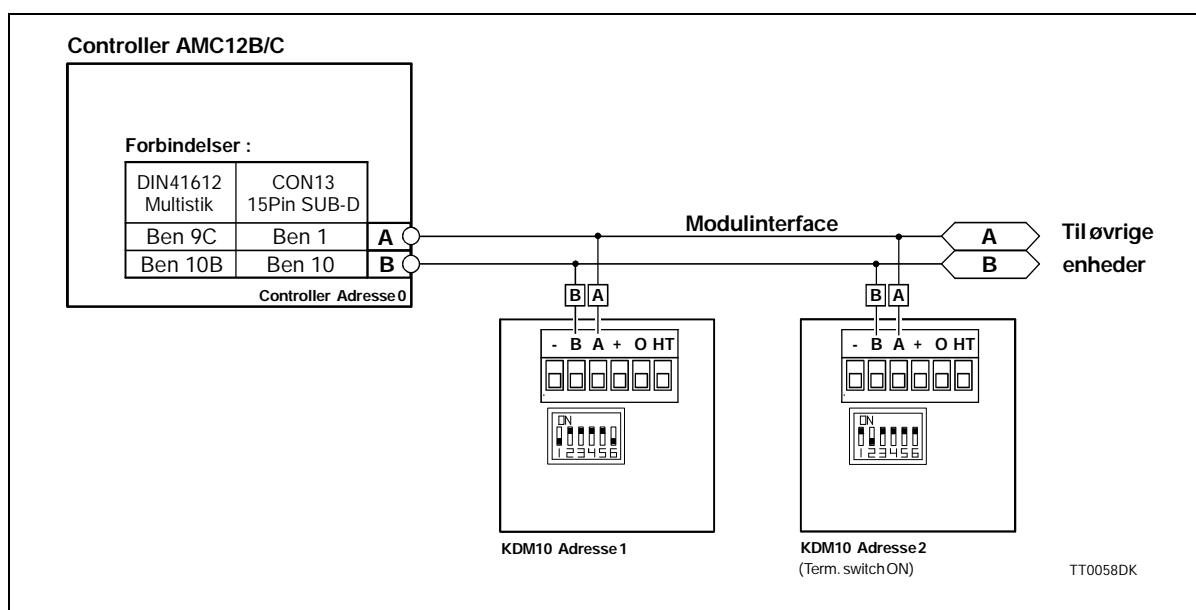
Følgende illustration viser sammenkoblingen i et typisk system med 2 eller flere enheder som er forbundet til en computer eller lignende.



RS485 Interface er kun tilgængeligt på kontrollere med serienummer højere end 5500.

3.16

Modulinterface



3.16.1 Modulinterface

Controlleren kan tilsluttes eksterne tilbehørsmoduler som f.eks. keyboard/display-modul, ind-/udgangs-moduler m.m.

Forbindelsen til disse eksterne moduler foretages via controllerens serielle modul interface. Interfacet udgøres af de 2 terminaler benævnt "A" og "B".

Via disse 2 terminaler styres alle de eksterne modulers funktioner. Op til 31 moduler og mindst 1 motorcontroller kan kobles på samme interface linie.

Modulinterfacet tilbyder flere fordele idet det kører balanceret og kommunikationslinien er lavimpedant. Envidere er de 2 interface-terminaler galvanisk isoleret fra de øvrige tilslutninger på controlleren.

Interfacet er beskyttet mod eventuelle transienter der kan forekomme på interfaceledningerne fra controller til modulet. Disse faktorer gør at der kan kommunikeres over store afstande, på trods af elektrisk støj.

Det kan anbefales at ledningsføringen mellem controlleren og de øvrige enheder på kommunikationslinien foretages med parsnoede ledninger.

I et system hvor kommunikationslængden overstiger 25 meter mellem 2 enheder skal dipswitchen mærket TERM stilles i position ON på de moduler der sidder isoleret mere end 25 meter væk fra de øvrige.

Se dipswitchens placering i brugermanualen for det aktuelle modul.

3.16.2 Adressering af moduler:

I et kommunikationssystem hvor flere enheder er koblet sammen, skal hver enhed indstilles til sin egen adresse. Denne adresse kan vælges i området 1-31.

Ovenstående skitse viser hvordan disse adresser i et typisk system er indstillet.

Det er ikke tilladt at lade flere moduler benytte den samme adresse. Hvis flere moduler benytter den samme adresse vil controlleren stoppe programafviklingen og melde fejl.

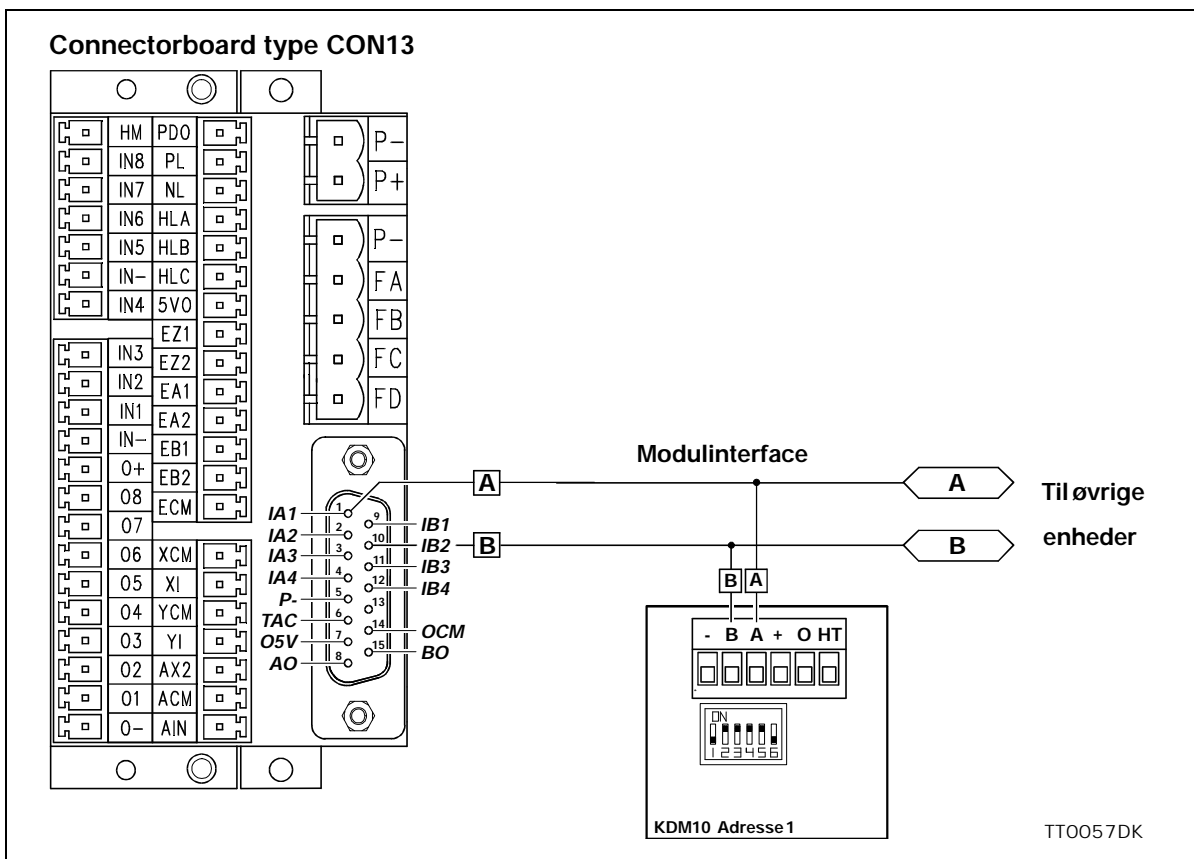
Bemærk at controllerens adresse er den samme som benyttes til RS232 interfacet.

Se afsnittet *Forbindelse til PC af flere controllere* side 43.

Det enkelte moduls adresse indstilles iflg. modulets brugermanual.

3.16

Modulinterface



3.16.3 Externe forbindelser via connectorboard CON13

Hvis de eksterne moduler tilsluttes via connectorboard JVL type CON13 følges ovenstående skitse. Bemærk at de øvrige ben på det 15 polede SUB-D stik, er benyttet til andet formål og derfor skal være uforbundne.

Ovenstående illustrere forbindelse til keyboard/display enhed KDM10.

Controlleren kan styres via RS232 interface. Kommandoerne sendes som ASCII karakterer og afsluttes med <CR> ASCII 13 (decimal) eller “;”. Se også *RS232 Interface* side 41.

Nogle kommandoer har tilhørende parametre andre har ikke. For kommandoer med parametre gælder det, at kommandoen alene får kontrolleren til at svare med kommandoen og den tilhørende værdi. Når der ikke benyttes adressering sender kontrolleren altid et svar tilbage, efter en kommando er modtaget. Hvis det er en kommando til visning af opsætning, vil den ønskede information sendes, ellers vil svaret være ‘Y’ som en indikation af, at kommandoen er modtaget. I de tilfælde, hvor der sendes noget forkert til kontrolleren - det kan være en kommando, der ikke findes, eller den ønskede opsætning ikke kan lade sig gøre, vil kontrolleren svare med en fejlmeddelelse. Der sendes først et ‘E’ og et nr. efterfulgt af en beskrivende tekst på engelsk. Se afsnittet *Fejlmeddelelser* side 117 .

| | |
|---------------------------------|----------------------|
| Eksempel: Sendes til controller | VM<CR> |
| Modtages fra controller | VM=500<CR> |
| Sendes til controller | VM=600<CR> |
| Modtages fra controller | Y<CR> |
| Sendes til controller | VM=-5<CR> |
| Modtages fra controller | E2: Out of range<CR> |

Når der benyttes adressering, vil kontrolleren ikke kvittere for modtagelse af en kommando. Eventuelle fejl i kommunikationen bliver opsamlet i fejlstatus register 0. Dette register kan aflæses med kommandoen *EST0 (enter)* - se også afsnittet *Fejlstatus i tekst (EST)* side 80

Kommandoer kan sendes som både store og små bogstaver. Med undtagelse af fejlmeddelelser er svaret fra kontrolleren altid med store bogstaver.

I det følgende beskrives alle RS232 kommandoerne. Alle kommandoer skal som før nævnt afsluttes med <CR> eller “;”, før kommandoen vil blive fortolket; dette fremgår ikke i beskrivelsen af de enkelte kommandoer.

Denne mode er primært tænkt anvendt som elektronisk gear. Pulsindgangen XI og YI tilsluttes en inkremental-enkoder, og motoren vil herefter følge denne enkoder. Systemet kan også styres som et steppermotorsystem via et steppuls- og retningsignal, idet motoren vil bevæge sig et trin, hver gang pulsindgangen tilføres en spændingspuls. Denne feature muliggør, at controlleren i mange applikationer kan erstatte et klassisk steppermotor-system uden enkoder. Hastigheden samt acceleration /deceleration vil være bestemt af de eksterne spændingspulser. MO skal sættes til 1 for denne mode. Se også afsnittet *Kom i gang - Gearmode (Mode 1)* side 5.

Eksempel på brug af gearmode:

Justér servoloopet, se evt. afsnittet *Justering af servoregulering* side 16 samt evt. øvrige parametre.

Vælg gearmode med $MO=1$

Vælg indgangsformat via *PIF* kommandoen. Se *Pulsindgangsformat (PIF)* side 98

Motoren kan nu styres via pulsindgangene XI og YI.

Kommandoer med særlig interesse for denne mode:

PIF, POF, ET, PR, PE

4.3 Positioneringsmode (MO=2)

Controlleren vil positionere via kommandoer, der sendes på RS232 interfacet. Diverse køre-parametre kan løbende ændres via RS232 interfacet, mens motoren kører. Denne mode kan primært benyttes i systemer, hvor kontrolleren indgår i et system med permanent kontakt til en PC via RS232 interfacet. MO skal sættes til 2 for denne mode. Se afsnittet *Kom i gang - Positioneringsmode (Mode 2)* side 6 .

Positionen angives i pulser. Bemærk at kontrolleren ganger enkoderens pulsantal med 4. Hvis f.eks. enkoderen har 500 pulser pr. omdrejning vil det samlede system have 2000 pulser pr. omdrejning. Angives en kørelængde på 2000 pulser vil dette betyde, at motoren roterer 1 omgange. Motorens øjeblikkelige position kan aflæses, uanset om motoren bevæger sig eller ej. Når en ny position sættes op, kører motoren til den nye position efter den programmerede hastighedsprofil, se AC og VM.

Motoren kan køre efter en programmeret hastighedsprofil. Til dette skal en maksimal hastighed og acceleration programmeres. Når motoren i positioneringsmode sættes til at køre til en given position, vil det ske efter den programmerede hastighedsprofil. Profilen vil altid følge accelerationen/decelerationen, d.v.s. motoren vil ikke altid opnå maksimal hastighed, hvis afstanden er for kort. Under hastighedsprofilen kan motorens tilstand udlæses ved brug af kommandoen RS.

Motoren kan på et vilkårligt tidspunkt standses med kommandoerne H eller SH.

Bemærk: for at opnå korrekt hastighed og acceleration skal antallet af enkoder pulser/omdr. sættes op ved brug af kommandoen PR.

Eksempel på brug af positioneringsmode:

Vælg positioneringsmode med MO=2

Sæt en maksimal hastighed med VM

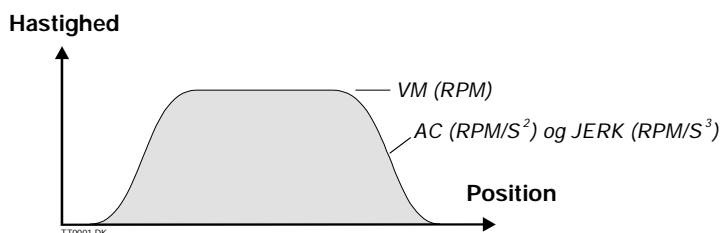
Sæt en acceleration med AC

Juster servoloopet, se evt. afsnittet *Justering af servoregulering* side 16

Motoren kan nu sættes til, at køre til forskellige positioner med kommandoerne SP eller SR.

Kommandoer med særlig interesse for denne mode:

ET, PR, SP, SR, VM, AC, PE



Figur -1 - Hastighedsprofil

4.4

Registermode (MO=3)

Controlleren kan også konfigureres til absolut eller relativ positionering via 8 binære indgange. Se også afsnittet *Kom i gang - Registermode (Mode 3)* side 7.

Controlleren har 64 programmerbare parametersæt. Hvert parametersæt kan indeholde information om acceleration, position (relativ eller absolut) og hastighed. Valg af parametersæt sker ved hjælp af indgangene IN1-IN6. Indgangen IN8 er en start/stop indgang. Hvis IN8 er høj, vælges et parametersæt, og motoren kører til den nye position med den ønskede hastigheds profil. Hvis IN8 sættes lav, inden den ønskede position er opnået, vil motoren standse med den programmerede deceleration (acceleration). Når IN8 igen går høj, fortsætter motoren mod den ønskede position. Når den ønskede position er opnået, går O1 høj, som indikation af at motoren er på plads. Se afsnittet *Kom i gang - Registermode (Mode 3)* side 7. Kommandoer med særlig interesse for denne mode: ET, PR, XR, XA, XP, XV, PE

Indgangene IN1-IN6 vælger hvilket parametersæt der skal benyttes til den pågældende kørsel.

Tabel -1 - Registersæt

| Register sæt | Binære indgange | | | | | | Funktion | | | |
|--------------|-----------------|-----|-----|-----|-----|-----|--------------|-----------|----------|---------|
| | IN6 | IN5 | IN4 | IN3 | IN2 | IN1 | Acceleration | Hastighed | Position | Relativ |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | XA0 | XV0 | XP0* | XR0** |
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | XA1 | XV1 | XP1 | XR1 |
| 2 | 0 | 0 | 0 | 0 | 1 | 0 | XA2 | XV2 | XP2 | XR2 |
| 3 | 0 | 0 | 0 | 0 | 1 | 1 | XA3 | XV3 | XP3 | XR3 |
| 4 | 0 | 0 | 0 | 1 | 0 | 0 | XA4 | XV4 | XP4 | XR4 |
| 5 | 0 | 0 | 0 | 1 | 0 | 1 | XA5 | XV5 | XP5 | XR5 |
| 6 | 0 | 0 | 0 | 1 | 1 | 0 | XA6 | XV6 | XP6 | XR6 |
| 7 | 0 | 0 | 0 | 1 | 1 | 1 | XA7 | XV7 | XP7 | XR7 |
| 8 | 0 | 0 | 1 | 0 | 0 | 0 | XA8 | XV8 | XP8 | XR8 |
| 9 | 0 | 0 | 1 | 0 | 0 | 1 | XA9 | XV9 | XP9 | XR9 |
| 10 | 0 | 0 | 1 | 0 | 1 | 0 | XA10 | XV10 | XP10 | XR10 |
| 11 | 0 | 0 | 1 | 0 | 1 | 1 | XA11 | XV11 | XP11 | XR11 |
| 12 | 0 | 0 | 1 | 1 | 0 | 0 | XA12 | XV12 | XP12 | XR12 |
| 13 | 0 | 0 | 1 | 1 | 0 | 1 | XA13 | XV13 | XP13 | XR13 |
| 14 | 0 | 0 | 1 | 1 | 1 | 0 | XA14 | XV14 | XP14 | XR14 |
| 15 | 0 | 0 | 1 | 1 | 1 | 1 | XA15 | XV15 | XP15 | XR15 |
| 16 | 0 | 1 | 0 | 0 | 0 | 0 | XA16 | XV16 | XP16 | XR16 |
| 17 | 0 | 1 | 0 | 0 | 0 | 1 | XA17 | XV17 | XP17 | XR17 |
| 18 | 0 | 1 | 0 | 0 | 1 | 0 | XA18 | XV18 | XP18 | XR18 |
| 19 | 0 | 1 | 0 | 0 | 1 | 1 | XA19 | XV19 | XP19 | XR19 |
| 20 | 0 | 1 | 0 | 1 | 0 | 0 | XA20 | XV20 | XP20 | XR20 |
| 21 | 0 | 1 | 0 | 1 | 0 | 1 | XA21 | XV21 | XP21 | XR21 |
| 22 | 0 | 1 | 0 | 1 | 1 | 0 | XA22 | XV22 | XP22 | XR22 |
| 23 | 0 | 1 | 0 | 1 | 1 | 1 | XA23 | XV23 | XP23 | XR23 |
| 24 | 0 | 1 | 1 | 0 | 0 | 0 | XA24 | XV24 | XP24 | XR24 |
| 25 | 0 | 1 | 1 | 0 | 0 | 1 | XA25 | XV25 | XP25 | XR25 |
| 26 | 0 | 1 | 1 | 0 | 1 | 0 | XA26 | XV26 | XP26 | XR26 |
| 27 | 0 | 1 | 1 | 0 | 1 | 1 | XA27 | XV27 | XP27 | XR27 |
| 28 | 0 | 1 | 1 | 1 | 0 | 0 | XA28 | XV28 | XP28 | XR28 |
| 29 | 0 | 1 | 1 | 1 | 0 | 1 | XA29 | XV29 | XP29 | XR29 |
| 30 | 0 | 1 | 1 | 1 | 1 | 0 | XA30 | XV30 | XP30 | XR30 |
| 31 | 0 | 1 | 1 | 1 | 1 | 1 | XA31 | XV31 | XP31 | XR31 |
| 32 | 1 | 0 | 0 | 0 | 0 | 0 | XA32 | XV32 | XP32 | XR32 |

* XP0 giver retning af nulpunktssøgningsfunktionen -1=negativ, 1=positiv

** XR0 fortæller om der skal ske automatisk nulpunktssøgning

4.4

Registermode (MO=3)

Tabel -1 - Registersæt

| Register sæt | Binære indgange | | | | | | Funktion | | | |
|--------------|-----------------|-----|-----|-----|-----|-----|--------------|-----------|----------|---------|
| | IN6 | IN5 | IN4 | IN3 | IN2 | IN1 | Acceleration | Hastighed | Position | Relativ |
| 33 | 1 | 0 | 0 | 0 | 0 | 1 | XA33 | XV33 | XP33 | XR33 |
| 34 | 1 | 0 | 0 | 0 | 1 | 0 | XA34 | XV34 | XP34 | XR34 |
| 35 | 1 | 0 | 0 | 0 | 1 | 1 | XA35 | XV35 | XP35 | XR35 |
| 36 | 1 | 0 | 0 | 1 | 0 | 0 | XA36 | XV36 | XP36 | XR36 |
| 37 | 1 | 0 | 0 | 1 | 0 | 1 | XA37 | XV37 | XP37 | XR37 |
| 38 | 1 | 0 | 0 | 1 | 1 | 0 | XA38 | XV38 | XP38 | XR38 |
| 39 | 1 | 0 | 0 | 1 | 1 | 1 | XA39 | XV39 | XP39 | XR39 |
| 40 | 1 | 0 | 1 | 0 | 0 | 0 | XA40 | XV40 | XP40 | XR40 |
| 41 | 1 | 0 | 1 | 0 | 0 | 1 | XA41 | XV41 | XP41 | XR41 |
| 42 | 1 | 0 | 1 | 0 | 1 | 0 | XA42 | XV42 | XP42 | XR42 |
| 43 | 1 | 0 | 1 | 0 | 1 | 1 | XA43 | XV43 | XP43 | XR43 |
| 44 | 1 | 0 | 1 | 1 | 0 | 0 | XA44 | XV44 | XP44 | XR44 |
| 45 | 1 | 0 | 1 | 1 | 0 | 1 | XA45 | XV45 | XP45 | XR45 |
| 46 | 1 | 0 | 1 | 1 | 1 | 0 | XA46 | XV46 | XP46 | XR46 |
| 47 | 1 | 0 | 1 | 1 | 1 | 1 | XA47 | XV47 | XP47 | XR47 |
| 48 | 1 | 1 | 0 | 0 | 0 | 0 | XA48 | XV48 | XP48 | XR48 |
| 49 | 1 | 1 | 0 | 0 | 0 | 1 | XA49 | XV49 | XP49 | XR49 |
| 50 | 1 | 1 | 0 | 0 | 1 | 0 | XA50 | XV50 | XP50 | XR50 |
| 51 | 1 | 1 | 0 | 0 | 1 | 1 | XA51 | XV51 | XP51 | XR51 |
| 52 | 1 | 1 | 0 | 1 | 0 | 0 | XA52 | XV52 | XP52 | XR52 |
| 53 | 1 | 1 | 0 | 1 | 0 | 1 | XA53 | XV53 | XP53 | XR53 |
| 54 | 1 | 1 | 0 | 1 | 1 | 0 | XA54 | XV54 | XP54 | XR54 |
| 55 | 1 | 1 | 0 | 1 | 1 | 1 | XA55 | XV55 | XP55 | XR55 |
| 56 | 1 | 1 | 1 | 0 | 0 | 0 | XA56 | XV56 | XP56 | XR56 |
| 57 | 1 | 1 | 1 | 0 | 0 | 1 | XA57 | XV57 | XP57 | XR57 |
| 58 | 1 | 1 | 1 | 0 | 1 | 0 | XA58 | XV58 | XP58 | XR58 |
| 59 | 1 | 1 | 1 | 0 | 1 | 1 | XA59 | XV59 | XP59 | XR59 |
| 60 | 1 | 1 | 1 | 1 | 0 | 0 | XA60 | XV60 | XP60 | XR60 |
| 61 | 1 | 1 | 1 | 1 | 0 | 1 | XA61 | XV61 | XP61 | XR61 |
| 62 | 1 | 1 | 1 | 1 | 1 | 0 | XA62 | XV62 | XP62 | XR62 |
| 63 | 1 | 1 | 1 | 1 | 1 | 1 | XA63 | XV63 | XP63 | XR63 |

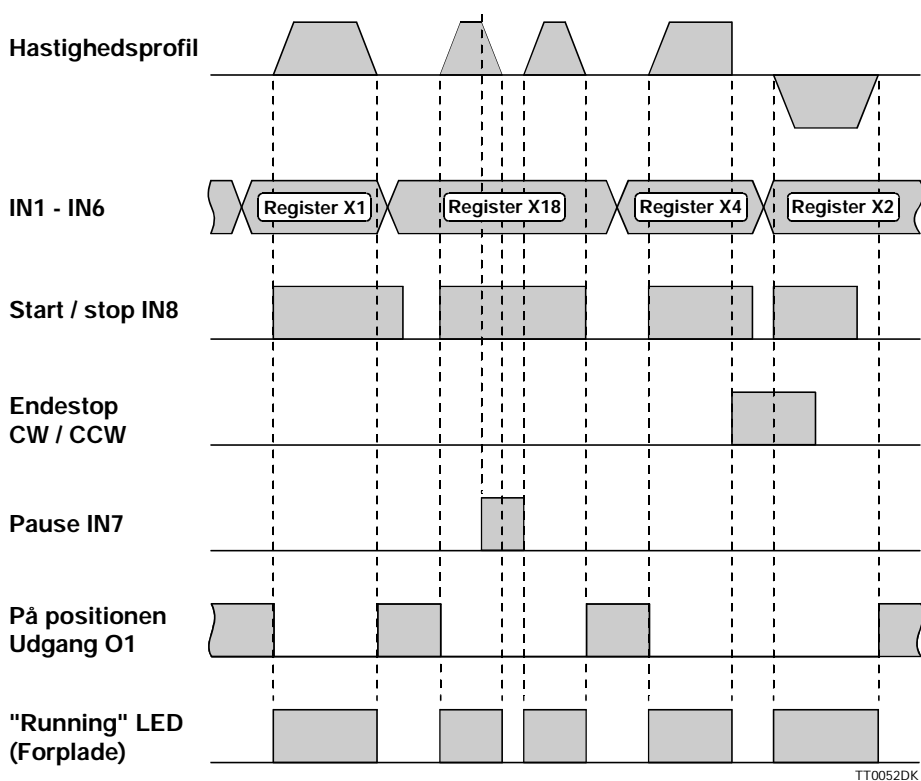
0 = Lav (Inaktiv)
1 = Høj (Aktiv)

0 = Nej
1 = Ja

4.4 Registermode (MO=3)

Set-up af parametersæt.

| | | |
|-----------------------------------|---|--|
| Eksempel 1: Sendes til controller | XV1=1000 | Sæt hastighed i parametersæt 1 til 1000 omdr./min. |
| Modtages fra controller | Y | |
| Eksempel 2: Sendes til controller | XV1 | Vis parametersæt 1 |
| Modtages fra controller | XV1=1000 | |
| Eksempel 3: Sendes til controller | XV | Vis alle parametersæt |
| Modtages fra controller | XV0=0 XV1=1000 XV2=200 XV63=0 | |



Figur -2 - Tidsforløb for valg af parametersæt og endestop

Bemærk at hvis endestop aktiveres indstilles den igangværende kørsel.

4.6

Momentmode (MO=5)

Motorens drejningsmoment kan styres analogt v.h.a. en analog indgang (AIN). Input spændingen skal ligge i området -10V til +10V, hvor en negativ spænding giver negativt drejningsmoment, og en positiv spænding giver positivt drejningsmoment. Momentet angives i Amp. CP angiver det maksimale moment d.v.s. det moment motoren skal yde ved fuld udstyring af analogindgangen.

Den numeriske værdi af momentet for fuld udstyring behøver ikke, at være den samme i positiv henholdsvis negativ retning. Benyt VVx kommandoerne til justering af analog indgangen.

Hvis f.eks. CP sættes til 6 Amp og den analoge indgangsspænding sættes til 5V, ydes der et moment svarrende til 3 Amp. Momentet er direkte proportional med motorstrømmen. Sammenhængen angives ved en momentkonstant, der normalt betegnes K_T eller K_A .

Brug af momentmode:

Vælg momentmode (MO=5)

Juster servoloop, se afsnit *Justering af servoregulering* side 16.

Juster evt. analogindgang, se afsnit *Justering af analogindgang* side 69.

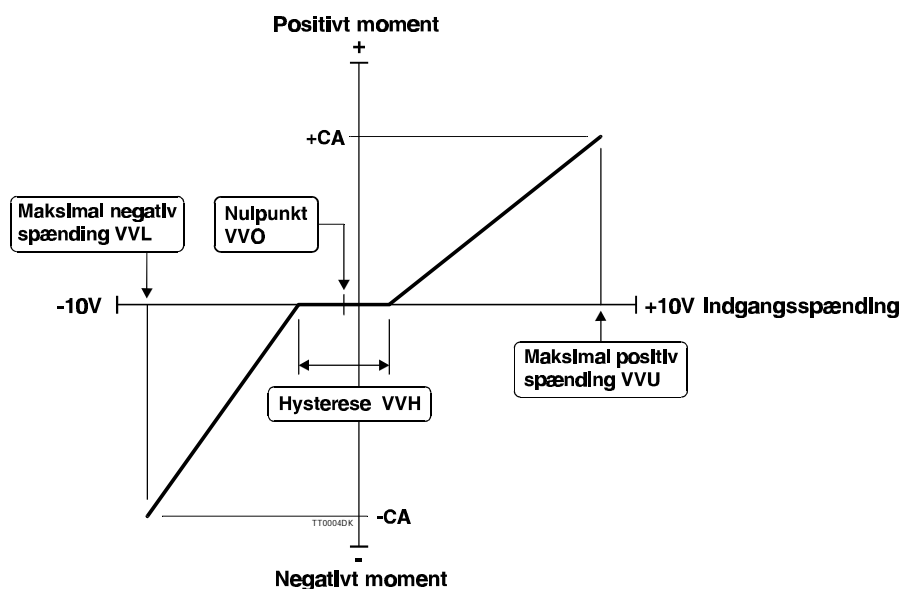
Sæt evt. maksimal hastighed med VM

Sæt maksimal moment med CP

Motoren kan nu styres med den analoge indgang (AIN). VM benyttes i denne mode til sikring af, at motoren ikke kommer så højt op i hastighed at mekanik samt motor overbelastes. Hastighedsbegrænsningen i denne mode er en sikkerhedsforanstaltning og ikke en præcis regulering.

Kommandoer med særlig interesse for denne mode:

CP, VM, VVH, VVL, VVO, VVU



Figur -5 - Momentstyring

4.7 Programafvikling i AMC12

4.7.1 Generel beskrivelse

AMC12 Servo controlleren har den udvidet feature, at den kan programmeres via et enkelt og fleksibelt programmeringssprog. Programmeringssproget er bygget op omkring interface kommandoerne. Således kan alle kommandoer benyttes i forbindelse med opbygning eller afvikling af programmer. Under programafviklingen kan alle parametre i controlleren aflæses og ændres. Alle opsætninger, der kan sættes og læses via en og samme kommando, kaldes registre, og kan benyttes i aritmetiske udtryk.

Programafviklingen er liniebaseret. Et program kan bestå af op til 500 linier, startende med linie 0. Der afvikles en linie for hver 2 msek. Controlleren kan således samtidig varetage alle de opgaver, som hører med til at være AC-servo controller. F.eks. overvåges effektforbrug og middelstrøm og ikke mindst er det muligt, at kommunikere via RS232 interface, når et program afvikles.

Selve programmeringssproget er meget simpelt, det kan sammenlignes med BASIC. Programmet oversættes ikke, men bliver fortolket under kørsel. Det giver den fordel, at der i princippet kun behøves et terminal program for at programmere controlleren.

4.7.2 Brug af kommandoer i programmet

En kommando indsat i programmet, som en "vis værdi" kommando, vil gøre at værdien sendes over RS232 interfacet. F.eks. kommandoen *AC* alene vil, hvis accelerationen er 100, give følgende streng på interfacet: *AC=100*. Kommandoen *AC=200* derimod vil ændre accelerationen til 200. Når en kommando indgår i et aritmetisk udtryk vil værdien af registret indgå i udtrykket. Linien *VM=AC+100* vil sætte den maksimale hastighed til værdien af accelerationen adderet 100. Når registerværdier indgår i udtryk tages der ikke hensyn til evt. enheder, som i dette tilfælde med hastighed og acceleration. Når f.eks. hastigheden ændres via kommandoen *VM* vil det have øjeblikkelig indvirkning på motorens bevægelse. Derfor skal ændring af motorparametre ske med stor forsigtighed.

Eksempel på brug af kommandoer i program:

```
AC=330           // Sæt accelerationen til 330 RPM/s
VM=500           // Sæt tophastigheden til 500 RPM
SR=100000        // Kør motoren 100000 pulser frem
AP               // Vis den aktuelle position via RS232 interface
```

4.7.3 Brugerregistre

Alle registre kan benyttes som midlertidigt lager af værdier. Da nogle registre, som før nævnt, har direkte indvirkning på motorbevægelsen, har controlleren 100 brugerdefinerede registre (R0-R99). Disse kan frit defineres og benyttes til at opbevare midlertidige resultater. I alle aritmetiske udtryk kan brugerregistrene indgå på lige fod med f.eks. servo-parametrene (KD, KI, KP) eller accelerationen (AC). Brugerregistrene kan indeholde værdier i området -2.147.483.647 til +2.147.483.647 og kan gemmes separat i den permanente hukommelse via kommandoen *MS2*. Når værdien af brugerregistrene er gemt i den permanente hukommelse, skal de hentes frem med kommandoen *MR2* før værdierne kan benyttes.

Eksempel på brug af brugerregistre:

```
R1=R2           // Sæt register 1 (R1) lig med register 2 (R2)
R1=-R1          // Vend fortegn på register 1
R1=-R2          // Vend fortegn på R2 og gem værdien i R1
R3=R1*-R2       // Vend fortegn på R2, gang med R1 og gem resultatet i R3
R1=KP*10        // Gang KP med 10 og gem resultatet i R1
```

4.7 Programafvikling i AMC12

Brugerregistrene kan også bruges til indirekte adressering ved brug af kantede parenteser [og]. R3 og R[3] giver samme resultat. [og] åbner muligheden for, at benytte et andet register eller et helt regneudtryk, som index til et register. Følgende er eksempler på indirekte adressering:

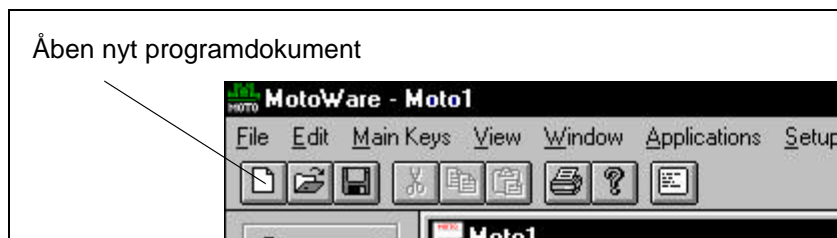
```
VM=R[R5]  
CA=R[R5+1]
```

4.7.4 Programmering af AMC12 ved hjælp af MotoWare

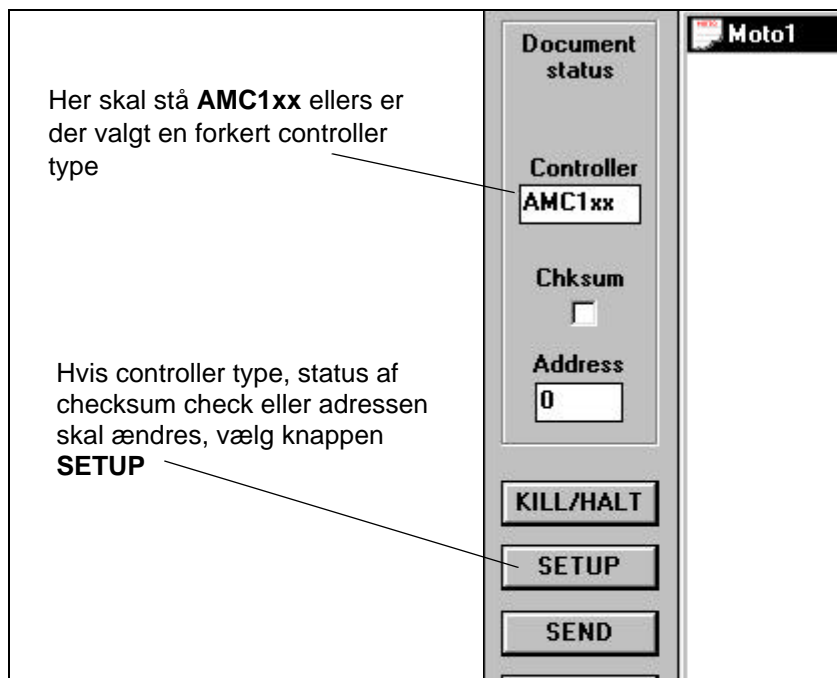
Ved brug af *MotoWare* kan programmerne nemt opbygges og lægges ned i controlleren.

Følg følgende vejledning første gang der skal laves et program:

- 1) Først åbnes et nyt programdokument. Enten ved at vælge FILE og derefter New... eller ved at trykke på nyt dokumentknappen.

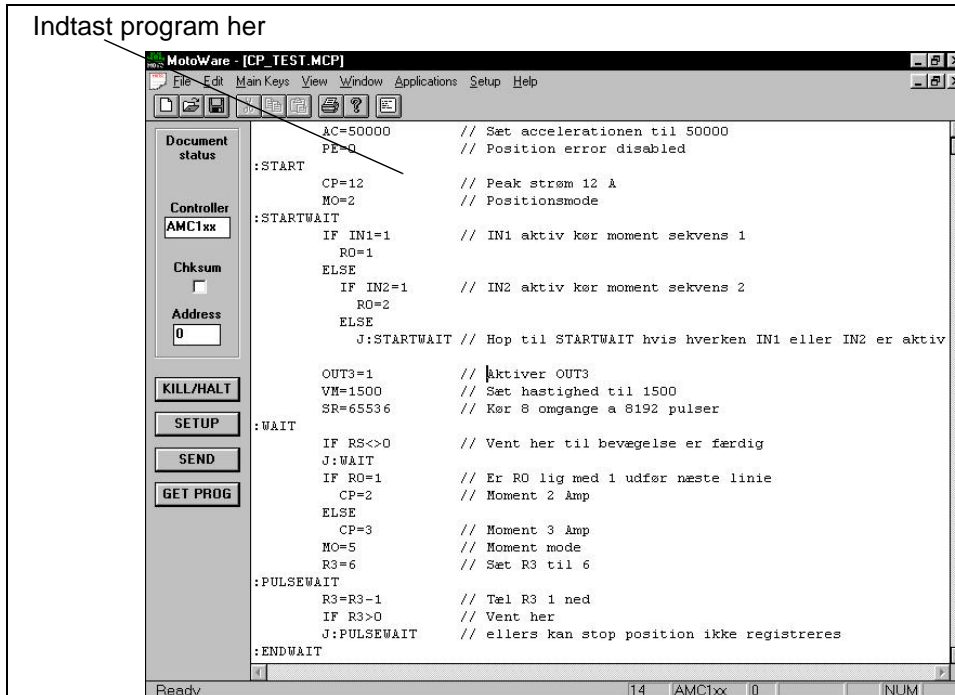


- 2) Vælg den rigtige controller type og evt. om der skal benyttes adressering og checksums check.



4.7 Programafvikling i AMC12

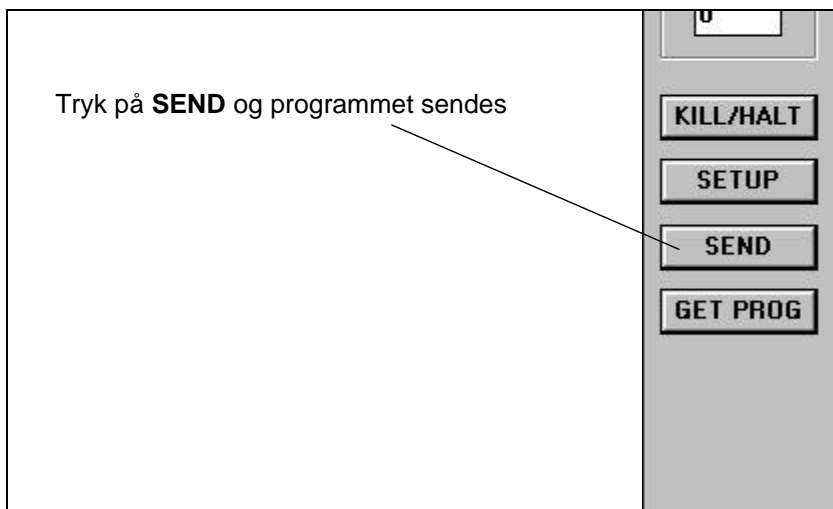
- 3) Indtast program i programdokument editoren



- 4) Når programmet er færdigt, kan det gemmes på harddisken.



- 5) Når programmet er gemt på harddisken, skal det sendes til controlleren. Vælg knappen **SEND** og programmet sendes. Hvis der opstår fejl vil en fejlmeddelelse vises - se: *Fejlmeddelelser under programmering og programafvikling side 64.*

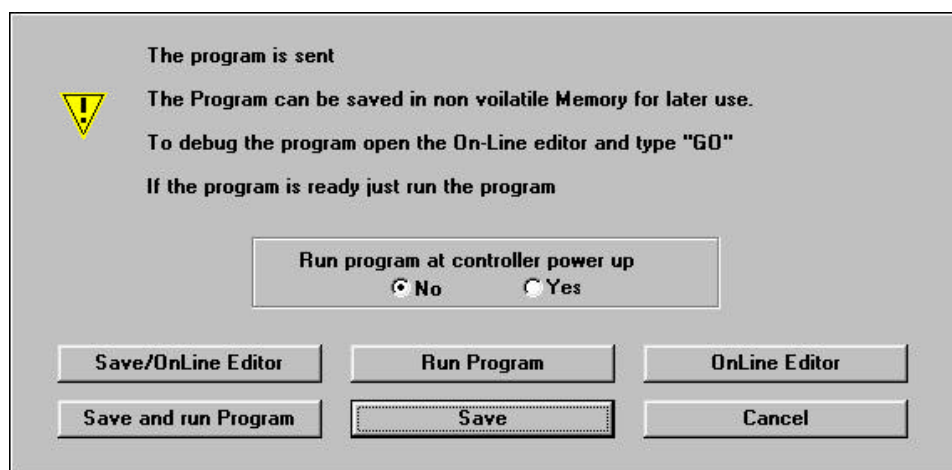


4.7

Programafvikling i AMC12

- 6) Når programmet er sendt, kommer der et vindue frem med forskellige muligheder. Der kan vælges om programmet, der er lagt ned i controlleren, skal startes automatisk, når controlleren tændes. Hvis programmet skal startes automatisk, dvs. *Yes* vælges, skal en af knapperne med *Save* vælges efterfølgende. Der er seks knapper med følgende betydning:

| | |
|-----------------------|--|
| Save/Online Editor: | Gem programmet i permanent hukommelse og åben OnLine Editoren. Først sendes kommandoen <i>MS</i> til controlleren. Dernæst åbnes OnLine Editoren og programmet kan startes herfra med kommandoen <i>GO</i> . Det er især vigtigt, at benytte OnLine Editoren, når programmer skal testes. Controlleren sender fejlmeddelelser i tilfælde af programfejl. Disse vises automatisk i OnLine Editoren. |
| Save and run Program: | Gem programmet i permanent hukommelse og start programmet. Kommandoerne <i>MS</i> og <i>GO</i> sendes i nævnte rækkefølge til controlleren. Hvorefter programmet gemmes og startes. |
| Run Program: | Start program. Kommandoen <i>GO</i> sendes til controlleren og programmet startes. |
| Save: | Gem program i permanent hukommelse. |
| OnLine Editor: | Start OnLine Editoren direkte herfra. OnLine Editoren åbnes og programmet kan startes herfra med kommandoen <i>GO</i> . Det er især vigtigt, at benytte OnLine Editoren, når programmer skal testes. Controlleren sender fejlmeddelelser i tilfælde af programfejl. Disse vises automatisk i OnLine Editoren. |
| Cancel: | Luk vinduet uden at gøre yderligere. |



4.7 Programafvikling i AMC12

4.7.5 Aritmetiske udregninger

Alle registre kan tillægges en værdi ved at efterfølge register navnet med et lighedstegn "=" efterfulgt af en værdi, et registernavn eller et matematisk udtryk. I et matematisk udtryk kan der benyttes absolutte værdier, registerværdier og de fire regningsarter:

De fire regningsarter der kan benyttes i matematiske udtryk:

| | |
|----------|-----------------------|
| + | addition |
| - | subtraktion |
| * | multiplikation |
| / | division |

Alle beregninger foregår enten som 32 bit heltal (-2.147.483.647 til +2.147.483.647) eller som 32 bit decimaltal også kaldet "float". Heltal benyttes med fortegn og har ca. 10 betydende cifre. De 32 bit for decimaltal er delt op som følgende: 1 bit fortegn, 8 bit eksponent og 23 bit mantisse. Decimaltal kan således beregnes med en nøjagtighed på 23 bit, hvilket giver ca. 7 betydende cifre. Når der laves beregninger hvori der indgår store tal bør der benyttes heltal. Som en general regel beregnes alle udtryk som heltal. Hvis der bare et sted i et udtryk indgår et decimaltal eller et register der udtrykkes med decimaler (f.eks. CP) bliver hele udregningen foretaget, som decimaltal. 3 vil indgå i et udtryk som heltal, hvor 3.0 vil gøre at hele udtrykket beregnes, som decimaltal. Når der regnes med heltal fjernes decimaldelen, dette gælder også for mellemregningerne. Udregningen sker ikke automatisk, som decimaltal, selvom registret på venstre side af lighedstegnet er et decimaltal. Konverteringen af værdien af udregningen på højre side, sker først når udregningen er tilendebragt. Udregninger der kun indeholder heltal går meget hurtigere end når der regnes med decimaltal. Brug derfor kun decimaltal, når det er nødvendigt. Følgende er eksempler på beregninger:

Resultatet forudsætter at IN1 er 1, R1 er 2, AC er 500, CP er 1.5 og VM er 100

| | |
|--------------------------------------|--------------------------------|
| R4=3/2+3/2 | // R4 får tildelt værdien 2 |
| R4=3.0/2+3/2 | // R4 får tildelt værdien 3 |
| CP=7/3+3/2 | // CP får tildelt værdien 3.0 |
| CP=7.0/3+3/2 | // CP får tildelt værdien 3.8 |
| R4=AC/VM*CP | // R4 får tildelt værdien 7 |
| CP=AC/VM*CP | // CP får tildelt værdien 7.5 |
| R4=IN1*35+CP*AC | // R4 får tildelt værdien 785 |
| R4=IN1*35+(R1-AC)*2--2*(7+3*(VM-50)) | // R4 får tildelt værdien -647 |

4.7.6 Rangen og rækkefølgen af evalueringer

Nedenstående tabel viser regler for rang og evalueringsretning for de enkelte operatører der kan indgå i et matematisk og/eller et logisk udtryk. Operatørerne på samme linie har samme rang, dvs. * og / har samme rang og udtrykket evalueres fra venstre mod højre; f.eks. $2*35/3$ giver værdien 23 og $35/3*2$ giver værdien 22, bemærk der regnes her kun med heltal. Linierne i tabellen er ordnet i faldende rangorden, så derfor har * og / højere rang end + og -. Det betyder, at de dele af udtrykket, hvor der indgår * eller / udregnes først. F.eks. giver $35+3*2$ resultatet 41. Parenteser "(" og ")" kan benyttes til at ændre rangordenen for de fire regningsarter; udtrykket $(35+3)*2$ giver resultatet 76.

4.7 Programafvikling i AMC12

Operatorer der kan benyttes i matematiske og logiske udtryk:

| Operator | Evaluerings retning |
|------------------------|---------------------|
| * / | venstre til højre |
| + - | venstre til højre |
| < > = <= >= <> | venstre til højre |
| AND | venstre til højre |
| OR | venstre til højre |
| = (tildeling af værdi) | højre til venstre |

4.7.7 Logiske ligninger

Logiske ligninger anvendes til at evaluere om en eller flere betingelser er opfyldt i forbindelse med IF sætninger. Formelt er syntaksen:

```
Logisk ligning ::= logisk udtryk { OR logisk udtryk }
logisk udtryk ::= logisk faktor { AND logisk faktor }
logisk faktor ::= værdi rel_op værdi    (hvor rel_op er <, >, =, <=, >= eller <>)
værdi ::= register eller regneudtryk
```

I logiske ligninger kan der benyttes almindelige regneudtryk, registre, relationelle (<, >, =, <=, >= eller <>) og logiske (AND og OR) operatorer. Rangen for OR og AND kan ikke ændres med paranteser "()". Der skal være et logisk udtryk på hver side af en AND eller en OR operator. I det logiske udtryk skal der indgå en relationel operator. Det er således ikke nok at skrive *AC OR VM* men *AC>0 OR VM>0* vil være legalt. Der kan benyttes så mange relationelle og logiske operatorer der ønskes så længe de formelle krav er opfyldt. I en logisk ligning kan der også indgå aritmetiske udregninger, hvor resultatet sammenlignes med en værdi, register eller en anden udregning. Følgende er eksempler på logiske ligninger:

```
IN1=1 OR IN2=1 OR IN3=1 AND IN4=1
// er sandt, hvis IN1 eller IN2 er 1 eller IN3 og IN4 er 1
```

```
AC>8*(4-3) AND IN1=IN2*IN3*IN4
// er sandt, hvis accelerationen er større end 8 og når IN1 er 1
// samtidig med at IN2, IN3 og IN4 er 1 eller IN1=0
// og bare en af IN2, IN3 eller IN4 er 0
```

```
AC<>VM*IN1// er altid sandt, når accelerationen er større end nul og forskellig
// fra hastigheden
```

Følgende er ikke legalt:

```
(AC>45 OR VM<67) AND AC<>VM
// paranteser må ikke benyttes til at ændre rangen af OR og AND
// den højre parantes forventes i dette tilfælde lige efter 45
```

```
IN1 OR IN2
// der vil her blive gjort opmærksom på at relationel operator mangler
```


4.7 Programafvikling i AMC12

4.7.8 IF (hvis) sætning

Logiske udtryk kan evalueres ved hjælp af en IF sætning. Sammen med ELSE kan IF sætningen bruges til at udtrykke beslutninger. Formelt er syntaksen:

```
IF udtryk
    handling1
ELSE
    handling2
```

hvor ELSE-delen er valgfri. Udtrykket beregnes, hvis det er sandt, udføres *handling1*. Hvis udtrykket er falsk og hvis der er en ELSE-del, da udføres *handling2*. If sætningen er liniebaseret, *handling1* skal stå på linien efter IF sætningen og hvis ELSE benyttes, skal ELSE og *handling2* stå på de følgende linier. *handling1* kan også bestå af flere kommando linier der bliver afsluttet af ELSE eller ENDIF. Hvis *handling2* består af flere linier skal sekvensen afsluttes med ENDIF. Ellers vil kun første linie indgå i IF ELSE-sætningen og de følgende linier vil altid blive udført. På grund af ovenstående vil følgende programstump ikke virke:

```
IF IN1=1          // NB bemærk denne programstump vil ikke virke
IF IN2=1
    AC
ELSE
    VM
```

hvis IN1 er lig med 1 vil det gå godt, da IF IN=2 vil blive evalueret. Hvis IN1 er 0 vil linien med IF IN2=1 blive sprunget over, hvorefter kommandoen AC udføres. Der kommer derefter en linie med ELSE. Linier efter ELSE må kun udføres, hvis der har været en forudgående IF sætning med falsk udsagn, hvilket der i dette tilfælde ikke har været. Løsning på ovenstående kan være følgende:

```
IF IN1=0          // Udfør næste linie hvis IN1 er 0
    J:NN          // Hop til NN
IF IN2=1          // Udfør næste line hvis IN2 er 1
    AC           // Vis accelerationen på RS232 interface
ELSE             // Udfør næste line hvis IN2 er 0
    VM           // Vis hastigheden på RS232 interface
:NN
```

løsningsen kan også være:

```
IF IN1=1          // NB bemærk denne programstump vil ikke virke
BEGIN
    IF IN2=1
        AC
    ELSE
        VM
END
```

4.7 Programafvikling i AMC12

Konstruktionen

```
IF udtryk
  handling
ELSE
BEGIN
  IF udtryk
    handling
  ELSE
  BEGIN
    IF udtryk
      handling
    ELSE
      handling
  END
END
```

forekommer så ofte, at der her gives en kort gennemgang af denne. Denne sekvens af IF sætninger er den mest generelle måde at lave beslutninger mellem mange muligheder. Udtrykkene beregnes i rækkefølge, hvis et af udtrykkene er sandt, udføres den tilhørende handling, og hele kæden afsluttes. Som altid, er koden til hver handling en linie med tilhørende kommando eller en række af kommando-linier omkranset af BEGIN END.

Den sidste ELSE håndterer tilfældet, hvor ingen af de ovenstående betingelser blev opfyldt. Af og til er der ikke nogen sidste mulighed, i disse tilfælde kan det afsluttende

```
ELSE
  handling
```

udelades. For at illustrere en tre-delt beslutning, viser følgende eksempel, hvorledes der kan ventes på input fra IN1 eller IN2. Når IN1 er aktiv (1) sættes accelerationen til 500 og programmet fortsættes. Hvis IN1 er inaktiv (0) og IN2 er aktiv (1) sættes accelerationen til 900 og programmet fortsætter herefter.

```
:START
  IF IN1=1          // IN1 aktiv sæt AC=500
    AC=500
  ELSE
  BEGIN
    IF IN2=1       // IN2 aktiv sæt AC=900
      AC=900
    ELSE
      J:START // Hop til START hvis hverken IN1 eller IN2 er aktiv
  END
```

Bemærk at hvis der benyttes flere IF ELSE sætninger i sammenhæng med hinanden, skal der benyttes BEGIN og END ('{' og '}') kan benyttes istedet for BEGIN og END henholdsvis).

4.7 Programafvikling i AMC12

4.7.9 Fejlmeddelelser under programmering og programafvikling

Der er tre typer af fejlmeddelelser: grammatisk, syntaktisk og fejl under kørsel (runtime error). Den første type fejl bliver der undersøgt for med det samme, når programmet overføres til controlleren. Der undersøges om de enkelte kommandoer og operatører findes, om absolutte tal ikke er for store. Der undersøges også om kommandoerne indgår i den rette sammenhæng f.eks. vil følgende programlinie:

```
AC=H
```

give fejlmeddelelsen: *Error: This command must not be included in an equation.* Kommando H er ikke af registertypen. Når programmet overføres via programeditoren i *MotoWare* og der opstår en fejl, standses overførslen. Linien hvori der er en fejl bliver herefter fremhævet.

Da programmet fortolkes under eksekveringen, bliver de syntaktiske fejl fundet, mens programmet er i brug. Derfor er det vigtigt under afprøvning af et program, at have *MotoWare* kørende med On-Line editor vinduet åbent. Controlleren vil under afvikling automatisk sende evt. fejlmeddelelser. Følgende er eksempler på fejl:

```
VM=500
AC=VM=CP // Denne linie giver inkorrekt syntaks
IF VM>600
VM=900
```

Ovenstående programstump vil give fejlmeddelelsen: *Error in line: 1 Des.: Syntax* Hvilket betyder at der er en syntaksfejl i linie 1.

```
VM=500
R4=14
AC=VM
IF (VM>600 OR AC<>800 // Der mangler en parentes efter 600
```

Denne programdel giver fejl meddelelsen: *Error in line: 3 Des.: Right parenthesis expected.* Dette betyder, at der mangler en afsluttende parentes i linie 3. Husk at linienummereringen starter med linie 0. Er der syntaktiske fejl i et program standses afviklingen af dette.

Den tredje type fejl, er fejl der opstår, under normal drift af et program, der virker. Det er fejl som ikke er programfejl, men fejl i brugen af f.eks. registre. Hvis et register tildeles en for stor (eller for lille) værdi vil der under online kontrol normalt gives fejl meddelelsen: *E2: Out of range.* Under programafvikling vil denne type fejl ikke generere fejlmeddelelser på RS232 interfacet. Information om forudgående fejl bliver gemt i et register, der kan læses med kommando ES. Disse typer fejl kan der således tages hånd om under programafviklingen, og giver derfor ikke grund til at stoppe afviklingen. Følgende er et eksempel på, hvordan fejl kan omgås:

```
R1=ES0 // Slet evt. fejlmeddelelser
AC=100000 // Sæt accelerationen til 100000
IF ES0>0 // Hvis der er fejl er ES0 større end 0
AC=50000 // Sæt accelerationen til 50000
```

resultatet bliver, at accelerationen sættes til 50000

4.7 Programafvikling i AMC12

4.7.10 Hop mellem programlinier og brug af labels

Med kommandoen *J* åbnes der mulighed for, at springe rundt mellem de enkelte linier i et givet program. Hop (Jump) kommandoen kan kun forstås af controlleren, når den gives sammen med en absolut værdi f.eks. *J50* (hop til linie 50). Dette kan give store problemer ved ændringer i programmer, hvis der benyttes absolutte værdier. Når man benytter *MotoWare* åbnes der muligheden for brug af labels. *MotoWare* fortolker og oversætter de enkelte labels, og sender en for controlleren forståelig kommando. Labels må stort set bestå af alle synlige karakterer, men det anbefales kun at benytte tal og bogstaver (a-z). Da der ellers kan opstå problemer, hvis programmerne flyttes mellem computere med forskellig opsætning. Der skelnes mellem store og små bogstaver.

```
:START  IF IN1=1           // Er IN1 lig med 1 udføres næste linie
        J:OK              // Hop til OK
        ELSE              // Når IN1 er 0 udføres linien efter ELSE
        J:FEJL           // Hop til FEJL
:OK      OUT5=1           // Sæt OUT5
        J:START          // Start forfra
:FEJL    OUT5=0           // Slet OUT5
        J:START          // Start forfra
```

oversættes til

```
IF IN1=1
J4
ELSE
J6
OUT5=1
J0
OUT5=0
J0
```

4.7.11 Kald af funktion (sub-rutine)

Hvis en sekvens af kommandoer bliver brugt ofte i et program, kan der med fordel oprettes en sub-rutine. En sub-rutine startes med en label og afsluttes med kommandoen RET. Når en sub-rutine skal udføres benyttes JS (Jump Subroutine) kommandoen. Når JS kommandoen udføres, hoppes der til linien angivet efter JS enten i form af et nummer eller en label. Eksekveringen fortsættes herfra indtil der mødes en retur kommand RET. Efter en RET kommando fortsættes programmet på linien lige efter JS kommandoen. Følgende er et eksempel på brug af sub-rutine kald:

```
R5=500
R6=1000
R1=5
JS:TEST // sæt accelerationen til 500
R1=6
JS:TEST // sæt accelerationen til 1000
J:END
:TEST
  AC=R[R1]
  RET
:END
```

4.7 Programafvikling i AMC12

4.7.12 Pause i programafviklingen (Delay)

Med kommandoen D kan der laves en pause i program afviklingen. Pausen angives i millisekunder enten ved at skrive $D=$ *pause* eller $D(\textit{pause})$. Da program-afviklingen sker med en linie for hver 2 msek. vil pausen altid være et lige antal msek. F.eks. hvis pausen sættes til 13 msek. vil der blive et ophold i programafviklingen på 14 msek.

```
R1=20      // sæt R1 til 20  
D=R1      // vent her i 20 msek.
```

4.8

Mekanisk nulstilling

4.8.1 Nulpunktssøgningsfunktion

Motoren kan bringes til et kendt mekanisk udgangspunkt. Dette gøres ved hjælp en føler forbundet til HM indgangen. Parametersæt 0 (IN1 IN2...IN6= 000000) afviger fra de andre parametersæt ved at indeholde information om, hvordan nulpunktssøgningsfunktionen skal udføres.

Parametersættet *XA0*, *XP0*, *XV0*, og *XR0* bestemmer hvorledes nulpunktssøgningen skal finde sted. Paramteret *ZL* bestemmer nulstillingindgangens (HM) aktive niveau.

Disse parametre har følgende funktion :

| Parameter | Funktion | |
|------------|---|---|
| <i>XA0</i> | Angiver acceleration/deceleration under nulstilling. Den specificerede værdi er udtrykt i omdr./min./sek. Hvis <i>XA0</i> sættes lig 0, vil controlleren benytte parameteret <i>AC</i> under nulstillingen. | |
| <i>XP0</i> | <i>XP0</i> =-1 vil medfører at nulpunktssøgningen foregår i negativ retning. | <i>XP0</i> =1 vil medfører at nulpunktssøgningen foregår i positiv retning. |
| <i>XV0</i> | Angiver nominel hastighed under nulstilling. Hvis <i>XV0</i> sættes lig 0, vil controlleren under nulstilling benytte parameteret <i>VM</i> . | |
| <i>XR0</i> | <i>XR0</i> =0 vil medfører at controlleren ikke udfører automatisk nulpunktssøgning efter opstart. | <i>XR0</i> =1 vil medfører at controlleren udfører automatisk nulpunktssøgning efter opstart. |
| <i>ZL</i> | <i>ZL</i> =0 HM aktiveres ved lav. | <i>ZL</i> =1 HM aktiveres ved høj. |

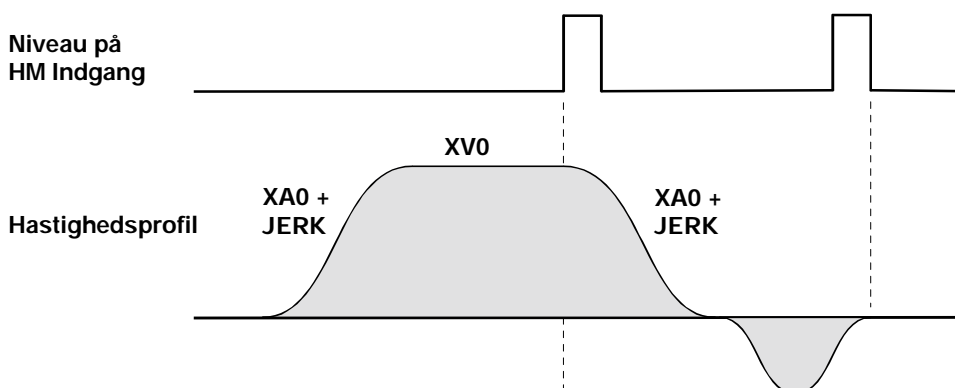
Nulstillingen vil blive udført efter en af følgende betingelser :

1. Efter opstart eller efter at controlleren har modtaget kommandoen *RESET*. Dette vil kun ske såfremt *XR0*=1 (se ovenstående tabel).
2. Hvis controlleren modtager nulstillingskommandoen *SZ*.
3. Hvis controlleren indstilles i Mode 3 (registermode), og registersæt *X0* bliver valgt.

4.8.2 Nulstillingssekvens

Når nulstillingen aktiveres vil motoren køre i den specificerede retning med den angivne hastighed, indtil indgangen *HM* bliver aktiv. Motoren vil herefter decelerer og standse, hvorefter den vil køre tilbage til den position hvor *HM* blev aktiveret.

Resultatet af søgningsprocessen bliver, at motoren vil stå lige ved kanten af nulpunktskontakten. Hermed er nulpunktet fundet, og motorens position *AP* (Actual Position) vil blive sat til 0.

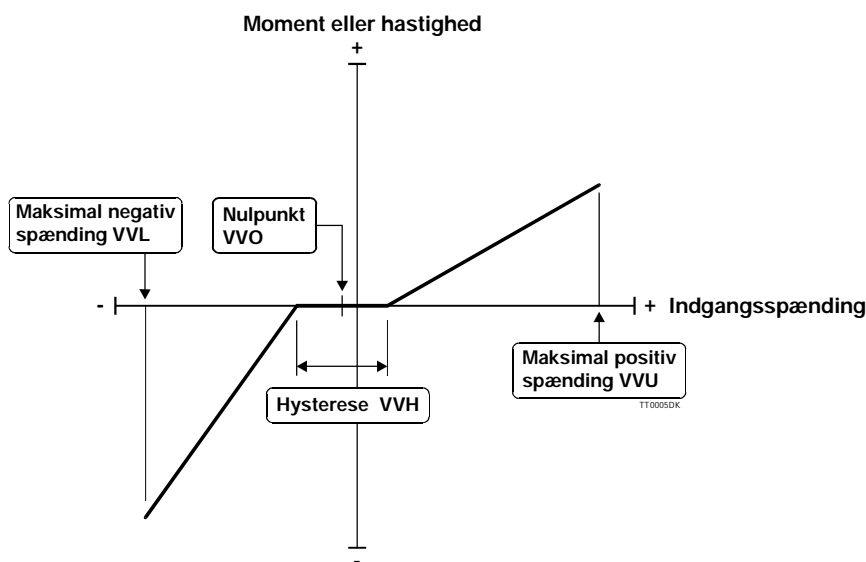


Motoren kan styres direkte ved hjælp af en analogindgang, hvor indgangsspændingen skal ligge i området ± 10 V. Analog indgangen benyttes i hastighedsmode (MO=4) og i momentmode (MO=5). Se afsnittet *Analogindgang* side 38 for mere information om analogindgangen.

Før den analoge indgang benyttes, skal den justeres til den aktuelle opgave. Denne justering er nødvendig idet den signalgiver der giver styresignalet til controlleren kan have offsetfejl eller kun kan afgive eksempelvis (9,5V eller mindre).

1. Vælg hastighedsmode (MO=4) eller momentmode (MO=5).
2. Fjern spændingen til motoren med kommandoen PO=1, så motoren ikke bevæger sig under justering.
3. Indstil nulpunkt ved at sætte indgang til 0V, og send kommandoen VVO.
4. Sæt indgangsspændingen maksimale negative værdi (maks. -10V) og send kommandoen VVL.
5. Sæt indgangsspændingens maksimale positive værdi (maks. +10V) og send kommandoen VVU.
6. Indstil en hystereseværdi ved hjælp af VVH. VVH sættes til det antal trin omkring 0V hvor motoren ikke må køre.
7. Nulstil indgangsspændingen (påtryk 0V).
8. Sæt spænding på motoren med kommandoen PO=0.

Motoren kan nu styres indenfor grænserne sat ved VVL og VVU med et område omkring nulpunktet givet ved VVO og VVH, hvor motoren står stille. Motoren styres lineært i området maksimal negativ spænding til hystereseværdien fra nulpunktet og i området fra nulpunktet plus hystereseværdien til maksimal positiv spænding. Bemærk, at hvis nulpunktet ikke er 0V, og den negative spænding ikke er numerisk lige så stor som den positive spænding, vil kurven være forskellig for negativ og positiv område.



Figur -6 - Analog moment- eller hastighedsstyring

4.10 Kommando beskrivelser

4.10.1 Vis opsætning (?)

Kommando ?

Mode 1, 2, 3, 4, 5

Beskrivelse De vigtigste informationer om status og opsætning kan vises med denne ene kommando.

Brug ? Viser værdierne.

Eksempel Sendes til controller?
Modtages fra controller :

```
Max. Velocity (RPM):      VM=100
Acceleration (RPM/S):    AC=6000
Average current (AMP):   CA=3
Peak current (AMP):      CP=10
Constant KD:             KD=10
Constant KI:             KI=30
Constant KP:             KP=8
Constant IL:             IL=1500
Pulses/Revolution:      PR=5000
Mode:                   MO=2
Encoder Type:           ET=1
Input (IN8-IN1):        IN=00000000
Output (8 LEDs - O8-O1): OUT=00000000
Actual Position (PULSES):AP=-1272
```

4.10.2 Controller type (!)

Kommando !

Mode 1, 2, 3, 4, 5

Beskrivelse Ved brug af denne kommando (udråbstegn) fås information om controllerens type og adresse. Controlleren vil give svar på denne kommando uanset om der benyttes adressering eller checksum. Der må således kun være en controller forbundet til interfacet, ved brug af denne kommando uden adresse. Kommandoen kan benyttes alene d.v.s. ! eller sammen med en adresse.

Brug ! Vis controllerens type og adresse.

Eksempel Sendes til controller !
Modtages fra controller AMC10C:ADDR=24

Bemærk at dette kun er et eksempel. Hvis controlleren er type AMC10B, AMC11B eller AMC12B, vil controlleren svare med AMC10B. Adressen (24) vil også afhænge af den aktuelle controller adresse.

4.10 Kommando beskrivelser

4.10.3 Acceleration (AC)

Kommando AC

Mode 2, 3

Interval 100 - 100000 omdr./min./sek.

Beskrivelse Denne kommando specificerer, med hvilken acceleration/deceleration profilen skal udformes. Hvis motoren er i bevægelse, når accelerationen ændres, vil accelerationen først blive ændret, når motoren standses. Bemærk at AC under ingen omstændigheder må benyttes i Mode 1, 4 og 5.

Brug AC = x Sæt accelerationen i omdr./min./sek.

AC Vis accelerationen.

4.10.4 Adresse (ADDR)

Kommando ADDR

Mode 1, 2, 3, 4, 5

Interval 0 - 255

Beskrivelse Controlleren kan konfigureres til at reagere på alt, hvad der kommer på interfacelinien, også kaldet Point to Point. Adressen skal være 0 for at opnå dette. Når adressen er 0 må adressen ikke sendes med, ved kommunikation med controlleren.

Der er også mulighed for at koble flere controllere på samme interfacelinien. For at opnå dette skal hver enkelt controller have sig egen adresse. Der kan benyttes adresser i området 1-255, det afhænger dog af hardwaren, hvor mange controllere der kan styres samtidig.

Bemærk: Hvis man har "glemt" controllerens adresse benyt kommandoen ! (udråbstegn).

Brug ADDR=x Sæt adressen til x.

ADDR Vis adressen.

4.10.5 Logisk OG operator (AND) - Kun AMC12

Operator AND

Mode Programmering

Beskrivelse AND operatoren benyttes i IF sætninger, når flere betingelser skal være opfyldt samtidig. Operatoren kan kun benyttes i en IF sætning. Operatoren kan kun benyttes i AMC12x.

Brug IF *udtryk* AND *udtryk*

4.10 Kommando beskrivelser

4.10.6 Aktiver flag i eksternt modul (AO) - Kun AMC12

Kommando AO

Mode Program

Interval Adresse 0-31, Flag 0 - 65535

Beskrivelse Kommandoen aktiverer et flag på modulet med adressen a. Flaget har nummer o. Flaget kan f.eks. være en udgang som på IOM11 modulet. Det vil medføre, at når et flag i IOM11 modulet bliver aktiveret, vil en udgang blive aktiveret. Hvis et flag i et andet modul bliver aktiveret, kan det betyde noget helt andet. F.eks. hvis man aktiverer flag 3 i KDM10, vil det medføre, at cursoren på LCD displayet begynder at blinke. D.v.s at flag med samme nummer på de forskellige moduler ikke har samme funktion. For at se funktionen af flagene, henvises til de respektive modulers brugermanual.

Brug AO{1<=a<=31}.{1<=o<=255}

Eksempel 1 Modulet KDM10 (Keyboard display modul) har adressen 4. Displayet ønskes slettes, så ny tekst kan udskrives. Nedenstående vil slette displayet og placere cursoren i øverste venstre hjørne.

```
AO4.1 // SLETTER DISPLAYET
```

Eksempel 2 IOM11 modulet og AMC12B er koblet sammen. IOM11 har adressen 10. Udgang 4 ønskes aktiveret.

```
AO10.4
```

4.10 Kommando beskrivelser

4.10.7 Aktuel Position (AP)

Kommando AP

Mode 1, 2, 3, 4, 5

Interval -1073741824 - 1073741823 pulser

Beskrivelse Motorens position kan på ethvert givet tidspunkt aflæses. Positionen angives i form af enkoder pulser i forhold til nulpunktet. Motorens position kan også “nulstilles” ved at give kommandoen en parameter.
Det anbefales kun at ændre positionen, når motoren står stille.

Brug **AP = x** Sæt motorens nuværende position til x.

AP Vis motorens position i pulser.

4.10.8 Aktuel Position for master akse (APM)

Kommando APM

Mode 1, 2, 3, 4, 5

Interval -1073741824 - 1073741823 pulser

Beskrivelse Position for master aksens i gear-mode kan på ethvert givet tidspunkt aflæses. Positionen angives i form af enkoder pulser i forhold til nulpunktet. Aksens position kan også “nulstilles” ved at give kommandoen en parameter.

Brug **APM = x** Sæt aksens nuværende position til x.

APM Vis aksens position i pulser.

4.10.9 Start program blok (BEGIN) - Kun AMC12

Kommando BEGIN

Mode Programmering

Beskrivelse BEGIN benyttes i IF sætninger, når flere kommando linier skal kædes sammen i en blok. BEGIN kan kun benyttes i en IF sætning. Se også *IF (hvis) sætning side 62*.

Brug IF *udtryk*
BEGIN
AC=500
VM=1000
END

4.10 Kommando beskrivelser

4.10.10 Bias efter PID filter (BIAS)

Kommando BIAS

Mode 1, 2, 3, 4, 5

Interval -32767 - 32767

Beskrivelse Controlleren indeholder denne parameter benævnt BIAS, som kan benyttes til opgaver hvor motoren er udsat for en vedvarende belastning, som f.eks. en løfteenhed. BIAS funktionen muliggør en udbalancering af denne statiske belastning, hvadenten belastningen trækker eller hiver i motoren. Denne udbalancering kan som regel være en fordel idet belastningen for PID filtret er ensartet uanset om motoren skal køre den ene eller anden vej, og i den sidste ende giver dette en lettere justering af det samlede system med en dertil hurtigere responstid.

Brug **BIAS=xx** Sæt BIAS lig xx.

BIAS Vis aktuelle BIAS indstilling.

4.10.11 Middelstrøm (CA)

Kommando CA

Mode 1, 2, 3, 4, 5

Interval 0.0 - 6.0 Amp. (AMC1xB)
0.0 - 12.0 Amp. (AMC1xC)

Beskrivelse Til beskyttelse af motoren mod overbelastning og forringelse af levetiden på motor kan en maksimal middelstrøm indstilles. Systemet vil automatisk lukke ned og give fejlmeddelelsen "E23 : Average Current limit exceeded", når den specificerede middelstrøm overskrides. Se også CP kommandoen til begrænsning af motorens spidstrøm.

Brug **CA=xx** Sæt middelstrøm i Amp.

CA Vis aktuelle middelstrømsindstilling.

4.10.12 Interface checksum (CHS)

Kommando CHS

Mode 1, 2, 3, 4, 5

Valg 0 = nej, 1 = ja

Beskrivelse Når der kommunikeres over interfacet kan der, som beskrevet, i afsnittet *Checksum* side 42 benyttes checksum.

Brug **CHS=x** 0=benyt ikke checksum, 1=benyt checksum.

CHS Vis checksum opsætning.

4.10 Kommando beskrivelser

4.10.13 Vis motorstrøm i % (CL)

Kommando CL

Mode 1, 2, 3, 4, 5

Interval 0 - 100 %

Beskrivelse Til overvågning af motorens belastning kan CL kommandoen benyttes. Hvis kommandoen CL sendes til controlleren vil den svare med et procentsats der angiver hvor høj den aktuelle middelstrøm til motoren er, sat i forhold til motorens maksimalt tilladte middelstrøm specificeret med CA kommandoen.

Brug CL Vis procentmæssig belastning af motor.

4.10.14 Deaktiver flag i eksternt modul (CO) - Kun AMC12

Kommando CO

Mode Program

Interval Adresse 0-31, Flag 0 - 65535

Beskrivelse I tilsluttede moduler er der nogle flag, som kan slettes med denne kommando. Antallet af flag i de forskellige moduler er forskelligt, men alle moduler har mindst et flag. I KDM10 (keyboard display modul) kan kommandoen feks bruges til at slette LCD displayet. I IOM11 (I/O modul) kan kommandoen bruges til at deaktivere en af udgangene m.m.
Kommandoen sletter et flag på modulet med adressen a.
Flaget har nummer o.

Brug CO{1<=a<=31}.{1<=o<=255}

Eksempel 1 AMC12B og KDM10 sidder forbundet sammen via modulinterfacet. AMC12B har adressen 1 og KDM10 har adressen 3. (Adressen bliver indstillet på de respektive dipswitch). Cursoren på LCD displayet skal slukke. Hvis cursor er tændt under tekst udskrivning med PRINT kommando, kan displayet flimre. Dette undgås ved at slette cursoren.

```
CO3.3 // SLETTER CURSOR
```

Eksempel 2 IOM11 modulet og AMC12B er koblet sammen . IOM11 har adressen 10. Udgang 4 ønskes aktiveret.

```
CO5.7 // DEAKTIVER UDGANG 7 PÅ IOM11 MED ADRESSE 5
```

4.10 Kommando beskrivelser

4.10.15 Spidsstrøm (CP)

Kommando CP

Mode 1, 2, 3, 4, 5

Interval 0.0 - 12.0 Amp. (AMC1xB)
0.0 - 25.0 Amp. (AMC1xC)

Beskrivelse Til beskyttelse af motoren mod overbelastning og forringelse af levetiden på motoren, kan en maksimal spidsstrøm indstilles. Systemet kan i kortere perioder acceptere strømme, der er højere end den maksimale tilladelige middelstrøm. Motoren kan dog beskyttes mod alt for høje strømpulser. CP benyttes til at sætte den maksimale spidsstrøm til motoren. Typisk skal CP indstilles 3-4 gange højere end middelstrømmen (CA).

Bemærk ! Den specificerede strøm er gældende for en enkelt motorfase. Se også afsnittet *Justering af motorstrøm* side 139

Brug CP=x.x Sæt spidsstrøm i Amp.

CP Vis aktuelle spidsstrømsindstilling.

4.10.16 Motorstrøm (CU)

Kommando CU

Mode 1, 2, 3, 4, 5

Beskrivelse Motorens strømforbrug i Amp kan aflæses med denne kommando.

Brug CU Vis motorens strømforbrug i Amp.

4.10.17 Aktuel hastighed (CV)

Kommando CV

Mode 1, 2, 3, 4, 5

Beskrivelse Motorens hastighed kan til enhver tid aflæses.

Brug CV Vis hastighed i omdr./min.

.

4.10 Kommando beskrivelser

4.10.18 Pause (D) - Kun AMC12

Kommando D

Interval 1 - 1073741823

Mode Programmering

Beskrivelse Med kommandoen D kan der laves en pause i program afviklingen. Pausen angives i millisekunder enten ved at skrive $D=pause$ eller $D(pause)$. Da program-afviklingen sker med en linie for hver 2 msek. vil pausen altid være et lige antal msek. F.eks. hvis pausen sættes til 13 msek. vil der blive et ophold i programafviklingen på 14 msek.

Brug **D=x** // vent i x msek.

D(x) // vent i x msek.

4.10.19 Digitalt Input Format (DIF)

Kommando DIF

Mode 3

Beskrivelse I mode registermode (MO=3) gives der mulighed for, at køre til en given position ved at sætte DIF=1 (default). Positionstælleren kan ignoreres ved, at sætte DIF=2. Fortegnet af værdien i XP angiver i, hvilken retning motoren skal køre. Motoren vil således køre så længe IN8 er aktiv.

Brug **DIF=x** Sæt Digitalt Input Format til værdien x

4.10.20 ELSE sætning (ELSE) - Kun AMC12

Sætning ELSE

Mode Programmering

Beskrivelse ELSE benyttes i forbindelse med en IF sætning. Hvis betingelsen efter IF ikke er opfyldt udføres linien efter ELSE. Sætningen kan kun benyttes i AMC12x. Se også *IF (hvis) sætning side 62*.

Brug IF *udtryk*
handling
ELSE
handling

Eksempel IF AC>(8+7)*2
AC=100
ELSE
AC=VM+98

4.10 Kommando beskrivelser

4.10.21 Afslut program blok (END) - Kun AMC12

Kommando END

Mode Programmering

Beskrivelse END benyttes i IF sætninger, når flere kommando linier skal kædes sammen i en blok. END kan kun benyttes i en IF sætning. Se også *IF (hvis) sætning side 62*.

Brug IF udtryk
BEGIN
AC=500
VM=1000
END

4.10.22 Afslut IF ELSE sætning (ENDIF) - Kun AMC12

Kommando ENDIF

Mode Programmering

Beskrivelse ENDIF benyttes i IF sætninger, når flere kommando linier skal kædes sammen i en blok. ENDIF kan kun benyttes i en IF sætning. Se også *IF (hvis) sætning side 62*.

Brug IF udtryk
AC=500
VM=1000
ENDIF

4.10.23 Execute Program Flag (EP) - Kun AMC12

Kommando EP

Mode 1, 2, 3, 4, 5

Valg 0 = Start ikke program, når der tændes for controlleren.
1 = Start program, når der tændes for controlleren.

Beskrivelse Når der i controlleren er indlagt et bruger program, kan dette startes automatisk. Når EP sættes til 1, hentes indlagt program fra permanent hukommelse, lægges i program hukommelse og startes. Når EP sættes til 0, starter controlleren som normalt. Herefter MR1 og GO kommandoerne kan benyttes. Kommandoen kan kun benyttes i AMC12x.

Brug **EP=x** Sæt Execute Program flag.

EP Vis status af Execute Program flag.

4.10 Kommando beskrivelser

4.10.24 Udskrift af fejlstatus (ES)

Kommando ES

Mode 1, 2, 3, 4, 5

Valg 0 og 1

Beskrivelse Under brug af styringen kan der opstå forskellige fejltilstande. Nogle fejl kan henføres til kommunikation og opsætning (fejlstatusregister 0) og andre fejl kan henføres til hardware og motorstyringsfejl. Fejlstatus kan udskrives ved hjælp af ES (Error Status) kommandoen. Kommandoen får controlleren til, at sende en række nuller (0) og ettaller (1). Der får herved et hurtigt overblik over fejlmeddelelserne, der kan forstås af et computerprogram. Ved brug af kommandoen EST fås en oversigt på engelsk, som er umiddelbart forståelig. Der findes to fejlstatusregistre. Register 0 giver information RS232 kommunikations- og opsætningsfejl. Registret opsamler alle fejl der er opstået siden registret sidst blev læst. Når registret læses slettes fejlinformationen automatisk.

Tabel -2- Fejl status bits, register 0

| Bit nr. | E nr. | Forklaring |
|---------|-------|---|
| 0 | E1 | Fejl |
| 1 | E2 | Værdi udenfor legalt område |
| 2 | E3 | Antallet af parametre er forkert |
| 3 | E4 | Kommandoen findes ikke |
| 4 | E5 | Det er ikke en kommando |
| 5 | E6 | Fejl i parameter eller værdi udenfor legalt område |
| 6 | E7 | Fejl i registernummer eller værdi udenfor legalt område |
| 7 | E8 | Data kan ikke blive gemt i EEPROM |
| 8 | E9 | Fejl i kommando checksum |
| 9 | E10 | Parameter vil blive trunkeret |
| 10 | E11 | Der er ingen program i hukommelsen |
| 15 | E16 | Se status register 1 |

Register 1 giver information om controller- og motorfejl. Nogle fejl kan være kortvarige, f.eks. kan maksimal spidsstrøm være overskredet i en kort periode og tilsvarende bit bliver sat i statusregistret. Fejlindikeringen fjernes efter læsning af fejl (Error) status. For de vitale fejl gælder det, at motoren afbrydes, og fejlinformationen vil blive stående, og O2 bliver sat høj (=1). Brugeren er herefter nødt til at slukke og tænde for systemet for at fjerne fejl tilstanden eller benytte RESET kommandoen.

4.10 Kommando beskrivelser

Tabel -3 - Fejl status bits, register 1

| Bit Nr. | Fejl status fjernes ved læsning | O2 sættes høj | Systemet skal genstartes | E nr. | Forklaring |
|---------|---------------------------------|---------------|--------------------------|-------|---------------------------------|
| 0 | Nej | Ja | Ja | E20 | Temperatur > 80°C |
| 1 | Nej | Ja | Ja | E21 | Strøm overload |
| 3 | Nej * | Ja | Ja | E23 | Middelstrøm overskredet |
| 4 | Ja | Nej | Nej | E24 | Fødespændingen er over 89 V |
| 5 | Nej | Nej | Nej | E25 | Negativ endestop aktiv |
| 6 | Nej | Nej | Nej | E26 | Positiv endestop aktiv |
| 7 | Nej | Ja | Ja | E27 | Motor er ikke forbundet rigtigt |
| 8 | Nej | Ja | Ja | E28 | Fejl på enkodersignal |
| 9 | Nej | Ja | Ja | E29 | Fødespændingen er over 95 V |
| 10 | Nej | Ja | Ja | E30 | Motoren er ikke forbundet |
| 11 | Nej | Nej | Nej | E31 | Middelstrøm kan ikke måles |
| 12 | Nej | Ja | Ja | E32 | Fejl på Hall signaler |
| 15 | Nej | Nej | Nej | E16 | Se status register 0 |

* Kun for software versioner højere end 2.4B.

Brug ES0Vis fejl (error) status register 0.

Eksempel Sendes til controller ES0
Modtages fra controller ES0=0000000001000101

Bemærk, bit 0 er sidste bit mod højre.

4.10.25 Fejlstatus i tekst (EST)

Kommando EST

Mode 1, 2, 3, 4, 5

Valg 0 og 1

Beskrivelse Kommandoen EST nøjagtig har samme funktion som kommandoen ES, fejlstatus udskrives dog i "klar" tekst. Ved brug af kommandoen fås en liste udskrevet på engelsk over de forskellige fejl. Hvis der ikke har været nogen fejl fås fejlmeddelelsen *E0: No errors*. En liste over fejlmeddelelserne kan ses i afsnittet *Fejlmeddelelser* side 117.

Brug **EST0** Udskriv fejlstatusregister 0 på engelsk.

EST1 Udskriv fejlstatusregister 1 på engelsk.

EST Udskriver både register 0 og register 1

4.10 Kommando beskrivelser

4.10.26 Enkoder type (ET)

Kommando ET

Mode 1,2, 3, 4, 5

Valg 0=PNP og 1=NPN

Beskrivelse For at opnå en korrekt positionering samt præcis hastighed og acceleration er det vigtigt, at enkoder opsætningen er rigtig. Enkoderen kan være af både PNP og NPN type. Desuden accepteres både et balanceret og ubalanceret signal fra en standard 2-kanals inkremental enkoder.

Ved enkoder tilslutning se afsnittet *Indstilling af enkoderopløsning* side 134.

Denne parameter bestemmer hvilken enkodertype der kan tilsluttes controlleren. Hvis der benyttes en enkoder med balanceret udgang kan der ses bort fra hvilken værdi ET sættes til.

Benyttes der derimod ubalanceret enkoder af NPN typen skal ET sættes til 1 (ET=1).

Benyttes en ubalanceret enkoder af PNP typen skal ET sættes til 0 (ET=0).

Brug **ET=x** Sæt enkoder type.

ET Vis enkoder type.

4.10.27 Stop programmering (EXIT) - Kun AMC12

Kommando EXIT

Mode Programmering

Beskrivelse Når et nyt program skal lægges ned i controlleren, startes sekvensen med kommandoen PROGRAM. Efter endt programmering sluttet af med kommandoen EXIT og programmet er klar til eksekvering (GO). Husk at gemme programmet i permanent hukommelse med kommandoen MS1. Kommandoen kan kun benyttes i AMC12x.

Brug **EXIT** Afslut programmeringssekvens.

4.10.28 Udveksling (GEAR)

Kommando GEAR

Mode 1

Interval 0.001 - 32767.999

Beskrivelse Denne kommando specificerer, hvilket forhold der skal være mellem antallet af pulser på pulsindgangen og antallet af pulser på motorens enkoder. Kommandoen kan kun benyttes i Mode 1 og er tiltænkt når controlleren bruges til såkaldt elektronisk gearing.

Brug **GEAR = x** Sæt udvekslingsforhold = x.

GEAR Vis udvekslingsforhold.

4.10 Kommando beskrivelser

4.10.29 Start program (GO) - Kun AMC12

Kommando GO

Mode 1, 2, 3, 4, 5

Beskrivelse Hvis der er et program i program hukommelsen, vil denne kommando starte eksekveringen. Kommandoen kan kun benyttes i AMC12x.

Brug **GO** Start program eksekvering.

4.10.30 Stop af motor (H)

Kommando H

Mode 2, 3

Beskrivelse Denne kommando standser motoren øjeblikkeligt, uanset hastighed deceleration etc.

Brug **H** Stop motor.

4.10 Kommando beskrivelser

4.10.31 Hall-element type (HALL)

Kommando HALL

Mode 1, 2, 3, 4, 5

Interval 0 - 3

Beskrivelse Controlleren kan initialiseres med eller uden Hall element i motoren. Normalt benyttes Hall elementet ikke hvis motoren gerne må bevæge sig under opstart. Er dette tilfældet sættes Hall registeret lig 0. Ønskes det derimod at motoren står fuldstændigt stille under opstart, skal der benyttes Hall element i motoren og Hall registeret sættes lig 1, 2 eller 3. Hall elementet benyttes under opstart til at fortælle controlleren hvor motoren befinder sig således at kommuteringselektronikken kan fastlåse det påtrykte magnetfelt ud for motorens aktuelle position uden at motoren flytter sig. Den information, der kommer fra motorens inkremental enkoder, kan ikke fortælle controlleren dette og kan derfor ikke benyttes.

Hall elementet benyttes kun ved opstart.

Der kan vælges mellem følgende Hall typer.

| HALL register : | Funktion |
|-----------------|---|
| HALL = 0 | Opstart uden HALL |
| HALL = 1 | Normal HALL - benyt HLA, HLB og HLC indgange |
| HALL = 2 | Yaskawa HALL kodning type 1. Benyt kun enkoderindgange inkl. Index kanal. |
| HALL = 3 | Yaskawa HALL kodning type 2. Benyt kun enkoderindgange inkl. Index kanal. |

Bemærk at Yaskawa motorer har deres HALL signaler kodet sammen med enkoder signalerne inkl. indeks-signal. Dette minimerer antallet af ledninger mellem motor og controller. Se også afsnit *Hall-indgang* side 32

Brug **HALL=xx** Sæt HALL type.

HALL Vis aktuelle HALL type.

4.10 Kommando beskrivelser

4.10.32 Kommando oversigt (HELP)

| | |
|--------------------|--|
| <u>Kommando</u> | HELP |
| <u>Mode</u> | 1, 2, 3, 4, 5 |
| <u>Beskrivelse</u> | Man kan få en oversigt over de kommandoer systemet kan betjenes med. Help udskriver en alfabetisk liste over de kommandoer, der kan benyttes. |
| <u>Brug</u> | HELP Vis kommandoer. |
| <u>Eksempel</u> | Sendes til controller HELP Modtages fra controller Following Instructions can be used AC ADDR AP CHS CL |

4.10.33 HALL type (HL)

| | |
|--------------------|---|
| <u>Kommando</u> | HL |
| <u>Mode</u> | 1, 2, 3, 4, 5 |
| <u>Valg</u> | 0=PNP og 1=NPN |
| <u>Beskrivelse</u> | <p>For at opnå en korrekt dekodning af HALL elementet i motoren (såfremt det benyttes), er det vitalt at HALL opsætningen er rigtig. HALL elementer kan være af både PNP og NPN typen. Desuden accepteres både et balanceret og ubalanceret signal fra HALL elementet. Ved HALL element tilslutning se afsnit <i>Hall-indgang</i> side 32.</p> <p>Hvis der benyttes HALL element med balanceret udgang kan der ses bort fra hvilken værdi HL sættes til. Benyttes der derimod ubalanceret HALL element af NPN typen skal HL sættes til 1 (HL=1). Benyttes en ubalanceret HALL element af PNP typen skal HL sættes til 0 (HL=0).</p> <p>Hvis der benyttes Yaskawa motor er HL parametret ligegyldigt, idet HALL signalet ligger indkodet i selve enkodersignalet og HALL-indgangen derfor ikke benyttes.</p> |
| <u>Brug</u> | HL=x Sæt HALL type. HL Vis HALL type. |

4.10 Kommando beskrivelser

4.10.34 IF sætning (IF) - Kun AMC12

Sætning IF

Mode Programmering

Beskrivelse Det er muligt at kontrollere programafviklingen med betingede sætninger. Hvis betingelsen i IF sætningen er opfyldt (forskellig fra 0) eksekveres linien lige under. Hvis betingelsen ikke er opfyldt (lig med 0), springes næste linie over og programmet fortsættes herfra. Der kan også benyttes en ELSE sætning i forbindelse med IF sætningen. I en IF sætning kan alle registre eller kommandoer som returnere en værdi, benyttes. Sætningen kan kun benyttes i AMC12x. Se også afsnit: *IF (hvis) sætning side 62*.

Til sammenligning kan benyttes følgende operatore:

| Operator | Beskrivelse |
|----------|--------------------------|
| < | Mindre end |
| > | Større end |
| = | Lig med |
| <= | Mindre end eller lig med |
| >= | Større end eller lig med |
| <> | Forskellig fra |
| AND | Logisk OG operator |
| OR | Logisk ELLER operator |

Brug **IF sætning** { **OR sætning** }
sætning::= udtryk { **AND udtryk** }
udtryk::= værdi rel_op værdi (hvor rel_op er <, >, =, <=, >= eller <>)
værdi::= register eller regneudtryk

Eksempler **IF AC>56 AND IN1=1**
AC=789

IF IN1=1

IF IN2=1 OR IN3=0 AND IN4=1 OR IN5=1

IF IN5=IN6

IF AC>6+VM-IN1+3*9 OR IN7=1

4.10 Kommando beskrivelser

4.10.35 Integral summerings grænse (IL)

Kommando IL

Mode 1, 2, 3, 4, 5

Interval 0 - 32767

Beskrivelse K_i har samme effekt på øjeblikkelige fejl som K_p . K_i bestemmer normalt, hvordan systemet skal reagere på vedvarende fejl, men for at få systemet til at reagere hurtigt på ændringer skal K_i øges i værdi. For at den summerede fejl ikke overskrider uacceptable høje værdier, indsættes en integral summerings grænse (IL). Hvis man f.eks. har et system med stor masse, kan det være svært at få motoren til, at følge den ønskede hastighedsprofil. Det kan derfor være ønskeligt, at sætte K_i til en høj værdi, så der sker en hurtig respons på en positionsfejl. Når motoren ikke følger hastighedsprofilen, samtidig med der benyttes en høj værdi for K_i , skal den akkumulerede fejl begrænses. Hvis den akkumulerede fejl ikke begrænses bliver systemet ustabil, og motoren vil have alt for høj hastighed, når den ønskede position nås. Forsøg først at indstille systemet uden justering af I_L . I_L kan derefter, som en begyndelse sættes til 1500 i pulsmode (MO=1), positioneringsmode (MO=2) og registermode (MO=3) og til 10000 i hastighedsmode (MO=4) og momentmode (MO=5).

Brug **IL=x** Hvor x angiver integral summerings grænse.

IL Viser værdien af IL.

4.10.36 Motor-initialiseringsniveau (IMCL)

Kommando IMCL

Mode 1, 2, 3, 4, 5

Interval 0 - 25 Amp.

Beskrivelse IMCL angiver hvor stor en strøm motoren skal initialiseres med. Den specificerede værdi er udtrykt i ampere pr. motorfase. Hvis den anvendte motor ikke har HALL element og HALL registeret er indstillet til 0 (se *Hall-element type (HALL)* side 83) vil controlleren benytte følgende algoritme til initialisering af motoren.

1. Efter opstart vil motoren blive påtrykt en strøm som er specificeret via IMCL.
2. Strømmen vil blive påtrykt i den tid som PT registeret angiver. Se afsnittet *Motor-initialiseringstid (PT)* side 106
3. Efter denne tid som typisk skal sættes til 1-3 sekunder, vil motoren være kørt hen til en ligevægtsposition ud for det dannede magnetfelt, og controlleren vil herefter låse sin kommuteringselektronik til den aktuelle motorposition. Initialiseringen er afsluttet og controlleren er driftsklar.

Normalt skal IMCL indstilles til 80-100 % af motorens tilladte middelstrøm (CA), for at sikre at motoren placeres præcist i ligevægt ud for det dannede magnetfelt og at controlleren dermed efterfølgende kommutere motoren optimalt.

Brug **IMCL=x** Hvor x angiver den strøm (amp.) motoren skal påtrykkes.

IMCL Viser værdien af IMCL.

4.10 Kommando beskrivelser

4.10.37 Læs status af indgange (IN1 - IN8)

Kommando IN

Mode 1, 2, 3, 4, 5

Beskrivelse Controlleren har 8 indgange. Status af disse kan aflæses med IN kommandoen. Indgangene har visse predefinerede funktioner afhængigt af, hvilken mode controlleren arbejder i. Hver enkelt indgang kan aflæses med kommandoen INx, hvor x angiver port nummeret. Med kommandoen IN læses alle indgange på en gang.

Tabel -4 - Oversigt over indgange

| Indgang | Funktion | |
|---------|---------------------------|------------------|
| | Registermode (MO=3) | Alle andre modes |
| IN1 | D0 (Mindst betydende bit) | Generel indgang |
| IN2 | D1 | Generel indgang |
| IN3 | D2 | Generel indgang |
| IN4 | D3 | Generel indgang |
| IN5 | D4 | Generel indgang |
| IN6 | D5 (Mest betydende bit) | Generel indgang |
| IN7 | Pause indgang | Generel indgang |
| IN8 | Start / stop indgang | Generel indgang |

Brug IN Læs indgange.

INx Læs indgang x.

Eksempel Sendes til controller IN
Modtages fra controller IN=00010100

Bemærk IN8 er første ciffer fra venstre (MSB)

4.10.38 Aktive niveau på de digitale indgange (INAL)

Kommando INAL

Mode 3

Valg 0 - 1 (00000000 - 11111111)

Beskrivelse Det aktive niveau for de digitale indgange kan programmeres til, at være højt (1) henholdsvis lavt (0). Indgangene kan programmeres uafhængig af hinanden. Hvis f.eks. start/stop indgangen programmeres til, at være aktiv lav vil motoren køre, når indgangen er lav. Hvis IN1, IN2 og IN5 sættes til aktiv lav og IN3, IN4 og IN6 aktiv høj vil en kombination 111101 på indgangen vælge register 46 (101110).

Brug INAL Læs aktive niveau på alle indgangene
INAL=abcdefgh Sæt aktive niveau på alle indgangene (abc.. kan sættes til 0 eller 1)
INALx Læs aktive niveau på indgang x
INALx=n Sæt aktive niveau til n på indgang x

4.10 Kommando beskrivelser

4.10.39 Aktive niveau på index puls (INDEX)

Kommando INDEX

Mode 1,2,3,4,5

Valg 0 - 1

Beskrivelse Har den anvendte enkoder en index puls, skal controllerens index indgang indstilles til den pågældende enkoders index polaritet. Hvis index pulsen er aktiv høj d.v.s. at den kun optræder som høj en enkelt gang pr. omdrejning, skal index sættes til 1 (aktiv høj). Se også afsnit *Indstilling af index-indgang* side 136.

Brug **INDEX** Læs aktive niveau for index pulsen

INDEX=x Sæt aktive niveau for index pulsen

4.10.40 Læs data fra eksternt modul (INPUT) - Kun AMC12

Command INPUT

Mode Program

Beskrivelse INPUT kommandoen bruges til at hente data fra de eksterne moduler, som er koblet på modulinterfacet (RS485). Af eksterne moduler kan nævnes Keyboard, Display, thumbwheel, BCD cifre fra PLC, printer, ekstra indgange, digital til analog moduler m.m. Alle modulerne er intelligente og indeholder registre. Disse registres indhold kan via INPUT kommandoen overføres til Servo Controllerens registre. Registrenes størrelse og antal kan variere, men alle moduler har mindst et register.

Brug Kommando format : INPUTn1.n2
n1: Specificerer adressen på det modul, der ønskes input fra. Tallet skal være mellem 0-255. På modulinterfacet er der mulighed for at have op til 32 moduler koblet på samtidig. De enkelte moduler skal, via dipswitch på modulet, sættes til den adresse som er specificeret i n1.
n2: Specificerer, hvilket register i det eksterne modul servocontrolleren ønsker at få overført. n2 skal være mellem 0-255.

Eksempel IOM11 modulet med 16 indgange og 8 udgange benyttes. Modulet har adresse 5. Der ønskes at læse alle 16 indgange og teste, om den binære værdi er 255. Hvis det er tilfældet, skal tælleren læses, og programmet fortsættes. I IOM11's manual finder man, at tæller's registre hedder 3, og registret for alle 16 indgange er 2.

```
:LÆSINDG R10=INPUT5.2 //LÆS ALLE INDGANGE OG OVERFØR TIL R10.  
IF R10<>255 //HVIS R10 FORSKELLIG FRA 255 LÆS IGEN  
J:LÆSINDG // -  
J:LÆS_TÆLLER //ELLERS LÆS TÆLLER OG FORTSÆT PROGRAM  
:LÆS_TÆLLER R30=INPUT5.3 //LÆS TÆLLERVÆRDI OG OVERFØR TIL R30
```

4.10 Kommando beskrivelser

4.10.41 Hop til programlinie (J) - Kun AMC12

Kommando J

Mode Programmering

Interval 0 - 500

Beskrivelse Jump kommandoen medfører et ubetinget spring til den i kommandoen specificerede linie. Kommandoen kan kun benyttes i AMC12x. Se *Hop mellem programlinier og brug af labels side 65*.

Brug **Jx** Hop til linie x

4.10.42 Opsætning af S-kurve profil (JERK)

Kommando JERK

Mode 2, 3

Interval 0 - 65535

Beskrivelse JERK kommandoen benyttes sammen med AC kommandoen til at lave et S-kurve profil..

Brug **JERK** Vis jerk opsætning

JERK=x Sæt jerk til x RPM/s²

4.10.43 Hop til sub-rutine (JS) - Kun AMC12

Kommando JS

Mode Programmering

Interval 0 - 500

Beskrivelse Jump Sub-rutine kommandoen medfører et ubetinget spring til en sub-rutine på den i kommandoen specificerede linie. Fra en sub-rutine kan en sub-rutine kaldes der derefter kalder endnu en sub-rutine, dette kaldes nasted sub-rutine kald. Der kan laves op til 16 nastede kald i en række. Antallet af kald af forskellige sub-rutiner fra en given sub-rutine er ubegrænset. En sub-rutine afsluttes med RET kommandoen. JS kommandoen kan kun benyttes i AMC12x. Se *Kald af funktion (sub-rutine) side 65*.

Brug **JSx** Hop til sub-rutine startende på linie x

4.10 Kommando beskrivelser

4.10.44 Vis servo konstanter K

| | |
|--------------------|---|
| <u>Kommando</u> | K |
| <u>Mode</u> | 1, 2, 3, 4, 5 |
| <u>Beskrivelse</u> | For kontrol af systemets K og IL værdier m.v., kan man benytte kommandoen K. |
| <u>Brug</u> | K Viser alle tre K værdier og integral grænsen IL m.v. |
| <u>Eksempel</u> | Sendes til controller K Modtages fra controller KD=70 KI=10 KP=40 KVFF=10 IL=100 BIAS=0 |

4.10.45 Konstant K_d

| | |
|--------------------|--|
| <u>Kommando</u> | KD |
| <u>Mode</u> | 1, 2, 3, 4, 5 |
| <u>Interval</u> | 0 - 32767 |
| <u>Beskrivelse</u> | Differentiel konstanten har især indflydelse på systemet, når der opstår ændringer i systemet f.eks. ved en acceleration eller deceleration. |
| <u>Brug</u> | KD = x Sæt K_d til værdien x. KD Viser K_d set-up værdi |

4.10.46 Konstant K_i

| | |
|--------------------|---|
| <u>Kommando</u> | KI |
| <u>Mode</u> | 1, 2, 3, 4, 5 |
| <u>Interval</u> | 0 - 32767 |
| <u>Beskrivelse</u> | Integral konstanten har indflydelse på, hvor aggressivt systemet skal reagere på vedvarende fejl. |
| <u>Brug</u> | KI=x Sæt K_i til værdien x. KI Viser K_i set-up værdi. |

4.10 Kommando beskrivelser

4.10.47 Konstant K_p

Kommando KP

Mode 1, 2, 3, 4, 5

Interval 0 - 32767

Beskrivelse Proportional konstanten beskriver, hvordan systemet skal reagerer på den øjeblikkelige fejl. En høj K_p værdi får systemet til at ændre sig hurtigt ved en given fejl.

Brug **KP=x** Sæt K_p til værdien x.

KP Viser K_p set-up værdi.

4.10.48 Hastighedsafhængig kommuteringsoffset (KPHASE)

Kommando KPHASE

Mode 1, 2, 3, 4, 5

Interval 0 - 32767

Denne parameter er afgørende for hvor langt kommuteringen af motoren skal forskydes i forhold til motorens aktuelle position. KPHASE er hastighedsafhængig, hvilket vil sige at det har en voksende betydning desto hurtigere motoren kører.

Det er af vital betydning for systemets ydeevne at denne parameter bliver justeret korrekt, idet en misjustering vil afstedkomme at motoren ikke kan yde optimalt moment ved høje hastigheder.

I værste fald vil motoren ikke køre med fuld hastighed, og systemet vil gå i fejl når positionsfejlen overskrider det der er specificeret i PE registeret - se afsnittet *Maksimal pulsfejl (PE)* side 97.

Brug **KPHASE =x** Sæt KPHASE til værdien x.

KPHASE Viser KPHASE set-up værdi.

4.10 Kommando beskrivelser

4.10.49 Feed Forward konstant (KVFF)

Kommando KVFF

Mode 1, 2, 3, 4, 5

Interval 0 - 32767

Beskrivelse “Feed Forward” konstanten benyttes normalt når motoren skal meget hurtigt i gang. Konstanten medfører en øjeblikkelig tilførsel af strøm til motoren, hver gang der startes eller stoppes.

Normalt vil et startsignal skulle passere PID filteret og vil derfor også være under indflydelse af KI, KD, og KP. Dette vil påvirke responstiden i det samlede system. KVFF er speciel i forhold til disse “normale” reguleringskonstanter, idet KVFF afgør hvor stor del af startsignalet (fejlen), der skal ledes uden om PID filteret og direkte til motoren.

Brug **KVFF = x** Sæt KVFF til værdien x.

KVFF Viser KVFF set-up værdi

4.10.50 Lysdiode status (LED)

Kommando LED

Mode 1, 2, 3, 4, 5

Interval 0 / 1

Beskrivelse LED registeret afgør hvorvidt de 8 lysdioder på forpladen benævnt IO1- IO8, skal vise indgangsniver eller udgangsniveauer. Sættes LED = 0 vises indgange (IN1-IN8). Sættes LED = 1 vises udgange (O1 - O8)

Brug **LED = x** Sæt LED til værdien x.

LED Viser LED set-up værdi

4.10.51 Vis linie nummer (LINE) - Kun AMC12

Kommando LINE

Mode 1, 2, 3, 4, 5, Programmering

Beskrivelse LINE kommandoen viser, hvilken linie i programafviklingen der er nået til. Kommandoen kan benyttes under programafvikling, eller til at vise hvor et program blev afbrudt.

Brug **LINE** Vis linie nummer

4.10 Kommando beskrivelser

4.10.52 List program (LIST) - Kun AMC12

Kommando LIST

Mode 1, 2, 3, 4, 5, Programmering

Beskrivelse Med LIST kommandoen vises programmet der måtte være i program hukommelsen.

Brug LIST Vis program.

4.10.53 Valg af mode (MO)

Kommando MO

Mode 1, 2, 3, 4, 5

Interval 1 - 5

Beskrivelse Styringen af motoren kan ske på fem forskellige måder. I det følgende bliver de fem modes beskrevet. Ved brug af kommandoen MO kan styrings mode vælges.

Brug MO = x

| Mode nr. | Mode |
|----------|---------------|
| 1 | Gear |
| 2 | Positionering |
| 3 | Register |
| 4 | Hastighed |
| 5 | Moment |

MO Vis aktuel styringsmode.

4.10 Kommando beskrivelser

4.10.54 Hent opsætning (MR)

| | |
|--------------------|--|
| <u>Kommando</u> | MR |
| <u>Mode</u> | 1, 2, 3, 4, 5, Programmering |
| <u>Interval</u> | 0 - 2 |
| <u>Beskrivelse</u> | Set-up data kan gemmes i EEPROM, d.v.s. der behøves ingen strøm for at bevare data. Ved hjælp af MR kommandoen kan de gemte data hentes, og systemet sættes op efterfølgende med de nye set-up værdier. |
| <u>Brug</u> | MR Hent alt For AMC10x og AMC11x vil denne kommando hente opsætnings data. For AMC12x vil denne kommando hente opsætnings data, program og bruger registre. MR0 Hent opsætnings data. MR1 Hent program. MR2 Hent bruger registre. Ved brug af MR0-2 for AMC10x og AMC11x vil fejlmeddelelsen <i>E3: Number of parameters is wrong</i> gives. |

4.10.55 Gem opsætning (MS)

| | |
|--------------------|---|
| <u>Kommando</u> | MS |
| <u>Mode</u> | 1, 2, 3, 4, 5 |
| <u>Beskrivelse</u> | Set-up data kan gemmes i EEPROM, d.v.s. der behøves ingen strøm for at bevare data. Ved hjælp af MS kommandoen gemmes opsætnings værdierne, program og bruger registre. MS med index kan kun benyttes i en AMC12x. |
| <u>Brug</u> | MS Gem alt i hukommelsen For AMC10x og AMC11x gemmes set-up værdierne. For AMC12x gemmes opsætnings værdierne, program og bruger registre. MS0 Gem opsætnings værdier. MS1 Gem program. MS2 Gem bruger registre. Ved brug af MS0-2 for AMC10x og AMC11x vil fejlmeddelelsen <i>E3: Number of parameters is wrong</i> gives. |

4.10 Kommando beskrivelser

4.10.56 Negativ endestop (NLS)

Kommando NLS

Mode 1, 2, 3, 4, 5

Valg 0=lav og 1=høj

Beskrivelse Indgangene *PL* og *NL* fungerer som endestop. Hvis motoren bevæger sig i negativ retning, og *NL* aktiveres, standser motoren øjeblikkeligt. *PL* er endestop i positiv retning. De to endestop kan uafhængig af hinanden programmeres til at være aktiv høj (1=PNP føler) henholdsvis aktiv lav (0=NPN føler).
Ved tilslutning af endestop se afsnittet *Endestopindgange* side 25.

Brug NLS = x Sæt endestop i negativ retning (Negative Limit Switch) til level 0=lav
1=høj.

NLS Vis endestopniveau.

4.10.57 Logisk ELLER operator (OR) - Kun AMC12

Operator OR

Mode Programmering

Beskrivelse OR operatoren benyttes i IF sætninger, når bare en af flere betingelser skal være opfyldt. Operatoren kan kun benyttes i en IF sætning. Sætningen kan kun benyttes i AMC12x.

Brug IF udtryk **OR** udtryk

4.10 Kommando beskrivelser

4.10.58 Vis/sæt status af udgange (O1 - O8)

Kommando OUT

Mode 1, 2, 3, 4, 5

Beskrivelse Controlleren har 8 udgange. Status af disse kan aflæses eller sættes via følgende kommandoer. Når status af udgangene O1 - O8 læses, fås også information om status af de 8 kontrol-lysdioder.

Tabel -5 - Oversigt over udgange

| Bit nr. | Udgang | Funktion |
|---------|--------|--|
| 0 | O1 | 1 = på position (benyttes kun i mode 2 og 3) |
| 1 | O2 | 0 = ingen fejl , 1 = vital fejl |
| 2 | O3 | Udgang 3. Kan benyttes alment via OUT kommando |
| 3 | O4 | Udgang 4. Kan benyttes alment via OUT kommando |
| 4 | O5 | Udgang 5. Kan benyttes alment via OUT kommando |
| 5 | O6 | Udgang 6. Kan benyttes alment via OUT kommando |
| 6 | O7 | Udgang 7. Kan benyttes alment via OUT kommando |
| 7 | O8 | Udgang 8. Kan benyttes alment via OUT kommando |

Brug **OUT** Læs status af udgange

Eksempel Sendes til controller *OUT* Læs udgange
Modtages fra controller *OUT=00000000* Bemærk bit 0 er sidste ciffer mod højre (LSB)

Sendes til controller *OUT3=1* Sæt O3 til 1
Modtages fra controller *Y*

OUTn Læs status af udgang n

OUTn=x Sæt udgang n (On) til x (0 eller 1)

OUT=xxxxxxx Sæt alle udgange til x, hvor x er 0 eller 1 (Kun bit 2 til 7 ændres)

4.10 Kommando beskrivelser

4.10.59 Maksimale pulsfejl (PE)

Kommando PE

Mode 1, 2, 3

Interval 0 - 32767 pulser

Beskrivelse For en ekstra sikkerhed kan der sættes en maksimal tilladelig pulsfejl. Hvis fejlen mellem den ønskede position og den aktuelle position bliver for stor kan det betyde, at enkoderen er defekt eller at motoren er blokkeret. Hvis pulsfejlen overskrider grænsen standses motoren og 4 dioder vil blinke. PE kan benyttes i gearmode (MO=1), positionsmode (MO=2) og registermode (MO=3). Sættes PE lig 0, vil controlleren acceptere en uendelig høj fejl, uden at indstille kørslen og melde fejl. Lysdioderne på forpladen *Running*, *Error*, *Current*, og $T > 80^{\circ}\text{C}$, vil blinke synkront hvis den maksimale pulsfejl overskrides.

Brug **PE** = x Sæt pulsfejl

PE Vis pulsfejl grænse

4.10 Kommando beskrivelser

4.10.60 Pulsindgangsformat (PIF)

Kommando PIF

Mode 1

Valg 1, 2, 3, 5, 6, 7

Beskrivelse PIF registeret definerer hvorledes det indkomne pulssignal på pulsingang (XI og YI) skal dekodes. PIF registeret er kun relevant når controlleren er indstillet i mode 1 - gearmode. Der kan vælges mellem følgende formater:

| Opsætning | Funktion | Typisk anvendelse |
|-----------|--|---|
| PIF = 1 | Inkrementalenkoderformat Båndbredden på signal input er 2 MHz (se PIF = 5) Indgangen kan tilsluttes en standard inkremental enkoder med 2 kanaler der er 90 grader faseforskudt. | Elektronisk gear |
| PIF = 2 | Puls og retningsformat Båndbredden på signal input er 2 MHz (se PIF = 6) Der tilsluttes et pulssignal på XI som styre motorens position og hastighed. YI tilsluttes retningsignal der bestemmer kørselsretning. | Simulering af step-motorsystem. Styring fra PLC aksekort |
| PIF = 3 | Puls / Pulsformat Båndbredden på signal input er 2 MHz (se PIF = 7) Der tilsluttes et pulssignal på XI som styrer motorens position og hastighed i den positive kørselsretning. Hvis motoren skal køre i negativ kørselsretning tilsluttes dette pulssignal på YI. | Simulering af step-motorsystem. Styring fra PLC aksekort |
| PIF = 4 | Reserveret til fremtidige formål. Kan ikke vælges. | |
| PIF = 5 | Inkrementalenkoderformat Samme som PIF = 1, med et 200 kHz filter indsat i signal input. Indgangen kan tilsluttes en standard inkremental enkoder med 2 kanaler der er 90 grader faseforskudt. | Elektronisk gear |
| PIF = 6 | Puls og retningsformat Samme som PIF = 2, med et 200 kHz filter indsat i signal input. Der tilsluttes et pulssignal på XI som styre motorens position og hastighed. YI tilsluttes retningsignal der bestemmer kørselsretning. | Simulering af step-motorsystem. Styring fra PLC aksekort |
| PIF = 7 | Puls / Pulsformat Samme som PIF = 3, med et 200 kHz filter indsat i signal input. Der tilsluttes et pulssignal på XI som styrer motorens position og hastighed i den positive kørselsretning. Hvis motoren skal køre i negativ kørselsretning tilsluttes dette pulssignal på YI. | Simulering af step-motorsystem. Styring fra PLC aksekort |

Se også afsnittet *Pulsindgange* side 35

Brug **PIF = x** Sæt pulsingangsformat = x

PIF Vis pulsingangsformat.

4.10 Kommando beskrivelser

4.10.61 Aktuelt effektforbrug (PL)

Kommando PL

Mode 1, 2, 3, 4, 5

Beskrivelse Controllerens samlede effektforbrug kan til enhver tid aflæses. Den udlæste værdi er integreret over 1 sekund og er udtrykt i % af den maksimalt tilladte effekt. Den maksimalt tilladte effekt fastsættes med PM se *Power Management (PM)* side 99.

Brug **PL** Vis aktuelt effektforbrug i % af PM.

4.10.62 Positiv endestop (PLS)

Kommando PLS

Mode 1, 2, 3, 4, 5

Valg 0=lav og 1=høj

Beskrivelse Indgangene *PL* og *NL* fungerer som endestop. Hvis motoren bevæger sig i negativ retning, og *NL* aktiveres, standser motoren øjeblikkeligt. *PL* er endestop i positiv retning. De to endestop kan uafhængig af hinanden programmeres til at være aktiv høj (1=PNP føler) henholdsvis aktiv lav (0=NPN føler). Ved tilslutning af endestop se afsnittet *Endestopindgange* side 25.

Brug **PLS = x** Sæt endestop i positiv retning (Positive Limit Switch) til level 0=lav
1=høj.

PLS Vis endestopsniveau.

4.10.63 Power Management (PM)

Kommando PM

Mode 1, 2, 3, 4, 5

Interval 10 - 1000

Beskrivelse Denne kommando specificere det maksimalt tilladte effektforbrug. Hvis effektforbruget overstiger værdien specificeret i PM registret, vil controlleren gå i en fejltilstand hvor motoren er strømløs og fejlregister 1 vil indeholde meddelelsen "*E22 : Power consumption too high*".

Kommandoen *Reset* vil nulstille controlleren til normal tilstand.

Bemærk at effektforbruget bliver integreret over 1 sekund. Dette muliggør at et effektforbrug på 200-300 % er tilladt i kortere tid, f.eks. mens motoren accelerere.

Effektforbruget bliver målt med fortegn hvilket vil sige at retur-effekt fra motoren under deceleration bliver fratrukket den målte værdi.

Det faktiske effekt-forbrug kan måles på et hvilket som helst tidspunkt ved brug af PL kommandoen se *Aktuelt effektforbrug (PL)* side 99

Brug **PM = x** Sæt maksimum effekt forbrug til x Wat.
PM Vis den aktuelle værdi af PM registret.

4.10 Kommando beskrivelser

4.10.64 Antal Motorfaser (PN)

Kommando PN

Mode 1, 2, 3, 4, 5

Valg 2 - 3

Beskrivelse Denne kommando gør det muligt at indstille controlleren til at arbejde med 2 faser (stepmotor) eller 3 faser (AC servomotor).
Hvis PN sættes til 2 faser benyttes alle 4 motorudgange benævnt FA, FB, FC og FD. Hvis PN sættes til 3 faser benyttes kun motorudgangene benævnt FA, FB, og FC.
Se også afsnit *Motortilslutning* side 21 .

Brug **PN=x** Sæt PN lig antal faser

PN Vis det aktuelle antal faser

4.10.65 Motor strøm ON/OFF (PO)

Kommando PO

Mode 1, 2, 3, 4, 5

Valg 0 = ON (strøm på motoren) og 1 = OFF (ingen strøm på motoren)

Beskrivelse Denne kommando gør det muligt, at fjerne strømmen fra motoren. Dette kan man bl.a. have nytte af i hastighedsmode (MO=4) og momentmode (MO=5), når den analoge indgang skal justeres.

Brug **PO=x** Sæt spændingen on/off

PO Vis værdi

4.10 Kommando beskrivelser

4.10.66 Pulsudgangsformat (POF)

Kommando POF

Mode 1, 2, 3, 4, 5

Valg 1, 2

Beskrivelse POF registeret definerer, hvilket signal der skal optræde på pulsudgangene (AO og BO). Der kan vælges mellem følgende formater:

| Opsætning | Funktion | Typisk anvendelse |
|-----------|--|--------------------------------------|
| POF = 1 | Puls indgangen (XI og YI) sendes ud på henholdsvis AO og BO. | Overvågning. |
| POF = 2 | Moterens enkoder. Kanal A og B fra motorens enkoder sendes i udekodet form ud på AO og BO. | Til overordnet PC eller PLC aksekort |

Se også afsnittet *Pulsudgange* side 37

Brug **POF = x** Sæt pulsudgangsformat = x

POF Vis pulsudgangsformat.

4.10.67 Fase offset (POFFSET)

Kommando POFFSET

Mode 1, 2, 3, 4, 5

Interval 1 - 32767

Beskrivelse POFFSET kommandoen benyttes til at sætte en fase offset vinkelen der benyttes til at opretholde en passende motor kommutering. Værdien specificeres i enheder af enkoder pulser, og repræsenterer offset fra index pulsen til fase A's maksimale output værdi.

Brug **POFFSET = x** Sæt fase offset

POFFSET Vis fase offset

4.10 Kommando beskrivelser

4.10.68 Antal motorpoler (POL)

Kommando POL

Mode 1, 2, 3, 4, 5

Interval 2 - 100 poler

Beskrivelse For at kunne kommutere motoren korrekt er det vitalt at indstille POL registeret til præcis det antal poler motoren har. En typisk stepmotor med 200 step pr. omdrejning har 100 poler (50 pol sæt) og en typisk AC servo motor har 2 eller 4 poler. Hvis denne parameter bliver indstillet forkert vil controlleren gå i fejl. Bemærk dog at enkoderopløsningen PR også har samme effekt.

Brug **POL** = x Sæt antal poler.

POL Vis antal poler.

4.10.69 Enkoder pulser (PR)

Kommando PR

Mode 1, 2, 3, 4, 5

Interval (50 - 20000) - se tekst

Beskrivelse For at opnå korrekt hastighed og kommutering af motoren skal antallet af enkoderpulser pr. omdrejning programmeres. Der skal her benyttes det antal pulser, som er angivet for enkoderen.

Bemærk at controlleren internt multiplicerer dette antal med 4 således at eksempelvis en enkoder/motor med 500 pulser pr. omdrejning i realiteten får 2000 pulser pr. omdrejning. Skal motoren rotere 1 omdrejning skal positioneringskommandoen baseres på de 2000 pulser.

PR kan ikke sættes lavere end motorens poltal gange 128. Hvis PR forsøges indstillet lavere vil controlleren svare med fejlmeddelelsen : *E2 : Out of range*

Brug **PR** = x Sæt pulser pr. omdr.

PR Vis enkoder pulser pr. omdr.

4.10 Kommando beskrivelser

4.10.70 Udskriv til eksternt modul (PRINT) - Kun AMC12

Kommando PRINTn1.n2.n3

Mode 1, 2, 3, 4, 5

Interval Adresse 0-31, Register 0-65535, Værdi 0-65535 (eller tekst)

Beskrivelse Print kommandoen kan anvendes til at udskrive indholdet af registre til eksterne moduler. På nuværende tidspunkt er det muligt at udskrive til 5 forskellige typer eksterne moduler, nemlig til en PC'er el.lign. gennem RS232 og DIS10, KDM10, CMO10 og IOM11 moduler gennem modulinterfacet (RS485).

Brug PRINTn1.n2.n3

- n1 : Specificerer adressen på modulet, der skal skrive til (1-31).
Hvis RS232 interfacet benyttes, specificeres værdien 255.
- n2 : Specificerer, hvilket register eller cursorposition i modulet der skal skrives til.
- n3 : Specificerer, hvilket register, talværdi eller tekststreng i servocontroller der skal udskrives. Når n3 er en tekststreng indeholder strengen to typer af objekter: almindelige karakterer der bliver vist i displayet direkte, og konverterings-specifikationer, der hver især udfører konvertering og udskrift af de efterfølgende argumenter til PRINT. Hver konverterings-specifikation startes med '%' og afsluttes med en konverteringskarakter. Hvis der indgår et decimalt tal i det efterfølgende argument, skal dette omkranses med parenteser, f.eks. (CA*1.5).
Konverteringskaraktererne og deres betydning er:
- % For at udskrive et '%' skal der inkluderes to '%' f.eks. "%%"
 - c Argumentet benyttes som en enkelt karakter
 - i Argumentet benyttes som et 16-bit heltal (-32768 til 32767)
 - l Argumentet benyttes som et 32-bit heltal (-2.147.483.647 til 2.147.483.647)
 - f Argumentet benyttes som et decimaltal ("floating-point") med en decimal.
 - .nf Argumentet benyttes som et decimaltal ("floating-point") med n decimaler.

Eksempel 1 PRINT1.0.R23

Udskriver indholdet af register R23 til det modul, der har adressen 1. Da der sendes via RS485 balanceret, vil det være muligt at placere modulerne op til 500 meter fra controlleren.

Eksempel 2 PRINT255.0.R2

Udskriver indholdet af register R2 til PC'er gennem RS232 interface. Kommandoen kan bruges til at udskrive indholdet af registre under programafviklingen. Den er specielt velegnet under fejlfinding i program. Hvis man anvender JVL's terminalprogram "Editor2", kan man, efter at programmet er overført med F5, skifte til kommunikationsvinduet ved at trykke F6. Her vil register indholdet blive vist, når PRINT linien nåes. Adressen 255 er fast defineret til at være PC'er.

4.10 Kommando beskrivelser

Eksempel 3 PRINT3.41."Indtast Antal: "

I forbindelse med KDM10 (keyboard-display modul) er det ofte ønskeligt at udskrive information til brugeren. Ovenstående eksempel viser, hvordan man skriver tekst på LCD displayet. Adressen på modulet er 3. 41 står for cursorposition 41, som er 1 tegn på linie 2.

Eksempel 4 R1=5555 // LÆG VÆRDIEN 5555 I REGISTER R1
R30=333 // LÆG VÆRDIEN 333 I REGISTER R30
PRINT5.41.R1 // UDSKRIV INDHOLDET I REGISTER R1 TIL CURSOR POSI-
// TION 41 I KEYBOARD/DISPLAY MODUL KDM10 MED
// ADRESSEN 5
PRINT2.0.R30 // UDSKRIV INDHOLDET I REGISTER R30 TIL DISPLAY MODUL
// DIS10 MED ADRESSEN 2

I forbindelse med modulerne KDM10 eller DIS10 er det ofte ønskeligt at udskrive information til brugeren. Ovenstående eksempel viser, hvordan man skriver en talværdi til KDM10 eller til DIS10.

Eksempel 5 PRINT3.41."ACT.POSITION:%1".AP
// UDSKRIV STRENGEN ACT.POSITION TIL CURSOR
// POSITION 41 I KEYBOARD/DISPLAY MODUL KDM10 MED
// ADRESSEN 3
PRINT3.1."IN:%i%i%i%i%i%i%i%i".IN8.IN7.IN6.IN5.IN4.IN3.IN2.IN1
// UDSKRIV INDGANG 8-1 (IN8-IN1)

Ovenstående eksempel viser, hvordan man udskriver en tekststreng som indeholder konverteringsspecifikationer til KDM10 eller til DIS10.

Eksempel 6 PRINT3.41."CP=%1f".(CP*1.5)

Ovenstående eksempel viser, hvordan man udskriver et argument som indeholder et decimaltal.

4.10 Kommando beskrivelser

4.10.71 Enkoder pulser for master (PRM)

Kommando PRM

Mode 1

Interval 50 - 20000

Beskrivelse Hvis controlleren benyttes i mode 1 (elektronisk gear) benyttes PRM registeret til at definere hvor stor opløsningen er på den tilsluttede masterenkoder der tilsluttes pulsindgangen (XI og YI).
Som med PR skal angives det antal pulser, som er angivet for enkoderen.
Bemærk at controlleren internt multiplicerer dette antal med 4.
Controlleren benytter PRM registret til at udregne det korrekte gearingsforhold mellem de indkomne pulser på XI og YI og den bevægelse motoren skal udføre.
Bemærk at PRM ikke kun har indflydelse når der benyttes en enkoder på pulsindgangen men også har indflydelse hvis der tilsluttes pulsindgangen et puls og retningssignal (format 2 / $PIF=2$) eller puls og pulssignal (format 3 / $PIF=3$)

Brug **PRM = x** Sæt pulser pr. omdr. på master enkoder

PRM Vis enkoder pulser pr. omdr. på master enkoder

4.10 Kommando beskrivelser

4.10.72 Start programmering (PROGRAM) - Kun AMC12

| | |
|--------------------|--|
| <u>Kommando</u> | PE |
| <u>Mode</u> | 1, 2, 3, 4, 5 (Programmering) |
| <u>Interval</u> | 0 - 32767 pulser |
| <u>Beskrivelse</u> | Når et nyt program skal lægges ned i controlleren, startes skevensen med kommandoen PROGRAM. Efter endt programmering sluttes af med kommandoen EXIT og programmet er klar til eksekvering (GO). Husk at gemme programmet i permanent hukommelse med kommandoen MS1. Kommandoen kan kun benyttes i AMC12x. |
| <u>Brug</u> | PROGRAM Start programmering. |

4.10.73 Motor-initialiseringstid (PT)

| | |
|--------------------|--|
| <u>Kommando</u> | PT |
| <u>Mode</u> | 1, 2, 3, 4, 5 |
| <u>Interval</u> | 100 - 17694 |
| <u>Beskrivelse</u> | <p>PT angiver i hvor lang tid motoren skal initialiseres med den strøm, der er specificeret i IMCL registeret. Den specificerede tid er udtrykt i milisekunder.</p> <p>Hvis den anvendte motor ikke har HALL element og HALL registeret er indstillet til 0 (off) vil controlleren benytte følgende algoritme til initialisering af motoren.</p> <ol style="list-style-type: none">1. Efter opstart vil motoren blive påtrykt en strøm som er specificeret via IMCL.2. Strømmen vil blive påtrykt i den tid som PT registeret angiver.3. Efter denne tid som typisk skal sættes til 1000-3000 msek, vil motoren være kørt hen til en ligevægtsposition ud for det dannede magnetfelt, og controlleren vil herefter låse sin kommuteringselektronik til den aktuelle motorposition. <p>Initialiseringen er afsluttet og controlleren er driftsklar.</p> <p>Normalt skal PT indstilles til 1000-3000 hvilket svarer til 1000-3000 millisekunders initialiseringstid.</p> <p>Dette er normalt tilstrækkeligt til at motoren placeres præcist i ligevægt ud for det dannede magnetfelt og at controlleren dermed efterfølgende kommutere motoren optimalt.</p> <p>Hvis der benyttes HALL element kan der ses bort fra PT og IMCL parameteret.</p> |
| <u>Brug</u> | PT = x Hvor x angiver (i msek) hvor lang tid motoren skal initialiseres. |
| | PT Viser værdien af PT. |

4.10 Kommando beskrivelser

4.10.74 Bruger registre (R) - Kun AMC12

| | |
|--------------------|---|
| <u>Register</u> | R |
| <u>Interval</u> | 0 - 99 |
| <u>Mode</u> | Programmering |
| <u>Beskrivelse</u> | Systemet inkludere 100 bruger registre. Disse registre kan frit benyttes til at gemme værdier i, bruges i regneudtryk osv. Registrerne kan indeholde værdier i området -2147483648 - +2147483648. |
| <u>Brug</u> | Rx=y Sæt register x til værdien y. |

4.10.75 Genstart controller (RESET)

| | |
|--------------------|---|
| <u>Kommando</u> | Reset |
| <u>Mode</u> | 1, 2, 3, 4, 5 |
| <u>Beskrivelse</u> | Hvis der sker en overbelastning af systemet f.eks. for høj fødespænding (se ES) skal systemet genstartes, før motoren kan styres igen. Denne kommando afstedkommer samme funktion som at slukke og tænde for controlleren. Set-up værdierne kan gemmes (MS) før kommandoen benyttes. Advarsel ! - Når Reset kommandoen sendes til controlleren skal der altid holdes en pause på 1 sekund før næste kommando sendes. |
| <u>Brug</u> | RESET Genstart controller. |

4.10.76 Afslut sub-rutine (RET)

| | |
|--------------------|--|
| <u>Kommando</u> | RET |
| <u>Mode</u> | Programmering |
| <u>Beskrivelse</u> | RET benyttes til at afslutte en sub-rutine. Se <i>Kald af funktion (sub-rutine) side 65.</i> |
| <u>Brug</u> | RET // afslut sub-rutine |

4.10.77 Motor tilstand - run status (RS)

| | |
|--------------------|--|
| <u>Kommando</u> | RS |
| <u>Mode</u> | 1, 2, 3, 4, 5 |
| <u>Interval</u> | 0 - 3 |
| <u>Beskrivelse</u> | Systemet kan under operation informere om motorens tilstand, hvilket vil sige om motoren er stoppet, er i bevægelse etc. |
| <u>Brug</u> | RS Motor Status: 0=standset 1=accelerer 2=max hastighed 3=decelerer |

4.10 Kommando beskrivelser

4.10.78 Motor/program tilstand i tekst - run status (RST)

Kommando RST

Mode 1, 2, 3, 4, 5

Beskrivelse Systemet kan under operation informere om motorens tilstand, hvilket vil sige om motoren er stoppet, er i bevægelse etc. Denne kommando giver for AMC12 også information om program status.

Brug **RST** Motor standset (Motor Status: Stationary)
Motor accelererer (Motor Status: Accelerating)
Motor kører med max. hastighed (Motor Status: Running)
Motor decelererer (Motor Status: Decelerating)

Følgende er kun for AMC12x

Program klar (Program Mode: Standby)
Program kan indtastes (Program Mode: Programming)
Program kører (Program Mode: Running)

4.10 Kommando beskrivelser

4.10.79 Default opsætning (SD)

Kommando SD

Mode 1, 2, 3, 4, 5

Beskrivelse Controlleren har en default-opsætning, når den forlader fabrikken. *SD* kommandoen bringer controlleren tilbage til default-opsætning.
Bemærk dog at efter en default opsætning vil controlleren gå i fejl, eftersom motorparametre (*POL*, *PN*, *HALL*, m.v.) med stor sandsynlighed ikke svarer til den anvendte motor. De oprindeligt indtastede værdier kan reddes ved brug af *MR* (Memory Recall) kommandoen, under forudsætning af at de oprindelige parametre blev gemt.

Opsætningen bliver som følgende:

| | | |
|--------------|----------------|-------------|
| AC = 500 | KI = 2 | PR = 2048 |
| ADDR = 0 | KP = 2 | PRM = 500 |
| BIAS = 0 | KPHASE = 500 | PT = 2000 |
| CA = 5 | KVFF = 0 | VM = 100 |
| CHS = 0 | LED = 1 | VVH = 5 |
| CP = 10 | MO = 2 | VVL = -1024 |
| DIF = 1 | NLS = 1 | VVO = 0 |
| ET = 1 | OUT = 00000001 | VVU = 1023 |
| GEAR = 1.000 | PE = 32767 | XA0-63 = 0 |
| HALL = 2 | PIF = 1 | XP0 = 1 |
| HL = 0 | PLS = 1 | XP1 = 1000 |
| IL = 200 | PN = 3 | XP2-63 = 0 |
| IMCL = 2.0 | PO = 0 | XR0-63 = 0 |
| INDEX = 1 | POF = 1 | XV0-63 = 0 |
| KD = 2 | POL = 8 | ZL = 1 |

Brug **SD** Lav default opsætning

4.10.80 "Blødt" stop af motor (SH)

Kommando SH

Mode 2, 3

Beskrivelse Denne kommando stopper motoren efter den preprogrammerede deceleration (acceleration).

Brug **SH** Stop motor.

4.10 Kommando beskrivelser

4.10.81 Sæt ny position (SP)

| | |
|--------------------|---|
| <u>Kommando</u> | SP |
| <u>Mode</u> | 2, 3 |
| <u>Interval</u> | -1073741824 - 1073741823 pulser |
| <u>Beskrivelse</u> | Motoren kan i positioneringsmode (MO=2) og registermode (MO=3) sættes til at køre til en ny position angivet i pulser. Bemærk at pulstallet er referere til enkoderens opløsning gange 4. Eksempelvis vil en enkoder/motor med 500 pulser pr. omdrejning i realiteten få 2000 pulser pr. omdrejning. Skal motoren rotere 1 omdrejning skal <i>SP</i> kommandoen baseres på de 2000 pulser. |
| <u>Brug</u> | SP = x Kør til ny Position. SP Vis ny position. |
| <u>Eksempel</u> | Sendes til controller SP=-1000 Kør til absolut position -1000 Modtages fra controller Y |

4.10.82 Relativ positionering (SR)

| | |
|--------------------|---|
| <u>Kommando</u> | SR |
| <u>Mode</u> | 2, 3 |
| <u>Interval</u> | -1073741824 - 1073741823 pulser |
| <u>Beskrivelse</u> | Motoren kan i positioneringsmode (MO=2) og registermode (MO=3) sættes til at køre et angivet antal pulser i positiv eller negativ retning. Retningen angives med fortegnet på parameteren. Bemærk at pulstallet er referere til enkoderens opløsning gange 4. Eksempelvis vil en enkoder /motor med 500 pulser pr. omdrejning i realiteten få 2000 pulser pr. omdrejning. Skal motoren rotere 1 omdrejning skal <i>SR</i> kommandoen baseres på de 2000 pulser. |
| <u>Brug</u> | SR = x Sæt relativ position |
| <u>Eksempel</u> | Sendes til controller SR=5000 Kør 5000 pulser i positiv retning Modtages fra controller Y |

4.10.83 Find nulpunkt (SZ)

| | |
|--------------------|---|
| <u>Kommando</u> | SZ |
| <u>Mode</u> | 1, 2, 3, 4, 5 |
| <u>Beskrivelse</u> | Ønskes motorens position nulstillet til et kendt punkt, benyttes denne kommando. Se også afsnittet <i>Nulstillingsindgang</i> side 26. |
| <u>Brug</u> | SZ Begynd nulstilling. |

4.10 Kommando beskrivelser

4.10.84 Firmware version (VE)

| | |
|--------------------|---|
| <u>Kommando</u> | VE |
| <u>Mode</u> | 1, 2, 3, 4, 5 |
| <u>Beskrivelse</u> | Information om firmwares version og fremstillings dato. |
| <u>Brug</u> | VE Vis version og dato. |

4.10.85 Maksimal hastighed (VM)

| | |
|--------------------|--|
| <u>Kommando</u> | VM |
| <u>Mode</u> | 2, 3, 4, 5 |
| <u>Interval</u> | 0 - 65535 omdr./min. |
| <u>Beskrivelse</u> | VM benyttes til, at sætte den maksimale hastighed. |

I positioneringsmode (MO=2), benyttes VM til, at sætte hastigheden motoren skal accelerere til og følge indtil der skal deceleres. Bemærk at VM også bliver anvendt i registermode (MO=3) hvis et givent XV registeret er indstillet til 0.

I hastighedsmode (MO=4) sætter VM grænsen for den hastighed som maksimalt input på analogindgangen skal svare til. Hvis F.eks. VM er sat til 1000 og analog indgangen er justeret til indgangsspænding i området -10V og +10V, vil motoren, hvis der påtrykkes -5V, kører med 500 omdr./min. i negativ retning.

VM benyttes i momentmode (MO=5) til at sætte en begrænsning for motoren. Reguleringen af hastigheden i denne mode er ikke særlig præcis og bruges kun som en ekstra sikkerhed.

VM har ingen effekt i gearmode (MO=1).

| | |
|-------------|---|
| <u>Brug</u> | VM = x Sæt maksimal hastighed i omdr./min. |
| | VM Vis maksimal hastighed |

4.10.86 Forsyningens-spænding (VOL)

| | |
|--------------------|--|
| <u>Kommando</u> | VOL |
| <u>Mode</u> | 1, 2, 3, 4, 5 |
| <u>Interval</u> | 10 - 100 |
| <u>Beskrivelse</u> | For at kontrollere controllerens påtrykte spænding, benyttes denne kommando. |
| <u>Brug</u> | VOL Vis spænding i Volt |

4.10 Kommando beskrivelser

4.10.87 Læs analogindgang (VV)

Kommando VV

Mode 1, 2, 3, 4, 5

Beskrivelse Med denne kommando kan controllerens analogindgang aflæses direkte. Værdien der angives er i AD-konverter trin.

Brug VVLæs analogindgang i ADC trin

4.10.88 Analogindgang - hysteresese (VVH)

Kommando VVH

Mode 4, 5

Interval 0 - 200 ADC trin

Beskrivelse Omkring nulpunktet kan der defineres et område, hvor motoren ikke må køre. Området størrelse er lig med to gange den angivne værdi. Værdien der angives svarer til antal trin på AD-konverteren. Konverteren har en arbejdsområde på 2048 trin (11 bit). Dvs. med en justering til -10V og +10V fås ca. 10 mV pr. trin. Se afsnit *Justering af analogindgang* side 69 for mere information om brugen af denne kommando.

Brug VVH = x Hvor x angiver hysteresese værdien

VVH Vis hysteresese værdi, samt værdierne for de tre kalibrerings kommandoer (VVL, VVO og VVU).

4.10.89 Analogindgang - maksimale negative (-10V) værdi (VVL)

Kommando VVL

Mode 4, 5

Beskrivelse Kalibrer fuldscale - sæt negativ spænding (max. -10V) på analogindgangen og send kommandoen VVL. Controlleren vil herefter kalibrere analogindgangs negative værdi. Den negative inputspænding må ikke være mere positiv end spændingen for nulpunktet. Se afsnit *Justering af analogindgang* side 69 for mere information om brugen af denne kommando.

Brug VVL Maksimal negativ spænding kalibreres

4.10 Kommando beskrivelser

4.10.90 Analogindgang - nulpunktsspændingen (VVO)

Kommando VVO

Mode 4, 5

Beskrivelse Denne kommando bestemmer analogindgangens nulpunktsspænding. Tilslut nulpunktspændingen og send kommandoen. Controlleren vil herefter nulstille indgangen. I de fleste tilfælde vil nulpunktspændingen være 0 Volt, dette er dog ikke et krav. Nulpunktsspændingen skal ligge i området max negativ til max positiv spænding. Se afsnit *Justering af analogindgang* side 69 for mere information om brugen af denne kommando.

Brug VVO Nulpunktsspændingen kalibreres

4.10.91 Analogindgang - maksimale positive (+10V) værdi (VVU)

Kommando VVU

Mode 4, 5

Beskrivelse Kalibrer fuldscale - sæt positiv spænding (max. +10V) på analogindgangen, og send kommandoen VVL. Controlleren vil herefter kalibrere analogindgangs positive spænding. Den positive spænding må ikke være mere negativ end nulpunktsspændingen. Se afsnit *Justering af analogindgang* side 69, for mere information om brugen af denne kommando.

Brug VVU Maksimal positiv spænding kalibreres

4.10.92 Vis alle parametersæt værdier (X)

Kommando X

Mode 3

Beskrivelse For et hurtigt overblik over alle værdierne i de 64 parametersæt kan kommandoen X benyttes.

Brug X Vis alle parametersæt
Controlleren viser følgende :

X0 : A=0 , V=0 , P=1 , R=0
X1 : A=0 , V=0 , P=1000 , R=0
X2 : A=0 , V=0 , P=0 , R=0
.....
.....
X63 : A=0 , V=0 , P=0 , R=0

Bemærk at disse værdier er defaultværdier, og kan afvige hvis opsætningen er ændret.

4.10 Kommando beskrivelser

4.10.93 Acceleration i parametersæt (XA)

Kommando XA

Mode 3

Interval (0) 100 - 100000 omdr./min./sek.

Beskrivelse For hvert parametersæt kan der sættes en ønsket acceleration. Hvis accelerationen sættes til 0, vil accelerationen ikke blive ændret ved valg af det pågældende parametersæt, idet den forudgående acceleration bliver genbrugt.

Brug **XAn=xxxxx** Sæt acceleration i parametersæt n til xxxxx omdr./min./sek..

XAn Vis accelerationen i parametersæt n

XA Vis alle accelerations værdier

4.10.94 Position i parametersæt (XP)

Kommando XP

Mode 3

Interval -1073741824 - 1073741823 pulser
Positions register 0: -1=negativ retning, 1=positiv retning.

Beskrivelse For hvert parametersæt kan der sættes en ønsket position. Hvis positionen sættes til nul, vil der ikke ske nogen ændring af positionen, men acceleration og hastighed vil blive ændret. Bemærk! XP1-63 har ikke samme betydning som XP0, der benyttes ved nulpunktssøgning. Opsætning af positionsvariabel i parametersæt 0 (XP0) bestemmer, i hvilken retning nulpunktssøgningen skal foregå, -1=negativ retning, 1=positiv retning, se afsnittet *Mekanisk nulstilling* side 67.

Brug **XPn=xxxxx** Sæt Position parameter til xxxxx pulser for parametersæt n

XPn Vis position

XP Vis positions værdier for all parametersæt

4.10 Kommando beskrivelser

4.10.95 Relativ positionering i parametersæt (XR)

Kommando XR

Mode 3

Valg 0=absolut, 1=relativ, for register 0: 0=søg ikke, 1=søg nulpunkt

Beskrivelse Relativ positionerings-parametersæt (XR) indeholder information, om den ønskede position er relativ eller absolut. XR0 har anden betydning end de andre registre. XR0 bestemmer om der skal ske automatisk nulpunktssøgning, når der tændes for controlleren eller ej, 0=søg ikke, 1=søg. Hvis controleren står i registermode (MO=3) og XR0=1 vil der, således ske en automatisk nulpunktssøgning i positiv eller negativ retning specificeret af XP0, når der tændes, se afsnittet *Mekanisk nulstilling* side 67.

Brug **XRn** = xx xx angiver, om positionen er absolut (0) eller relativ (1)

XRn Vis relativ positioneringsopsætning i parametersæt n

XR Vis alle relativ værdier

4.10.96 Hastighed i parametersæt (XV)

Kommando XV

Mode 3

Interval 1 - 65535 omdr./min.

Beskrivelse For hvert parametersæt kan der sættes en ønsket maksimal hastighed. Hvis hastigheden sættes til nul, vil hastighed ikke blive ændret ved valg af parametersæt, idet den forudgående hastighed vil blive genbrugt.

Brug **XVn** = xxxxx Sæt maksimal hastighed i parametersæt n til xxxxx omdr./min.

XVn Vis hastighedsværdi i parametersæt n

XV Vis alle hastighedsværdier for alle parametersæt

4.10 Kommando beskrivelser

4.10.97 Nulpunktsindgang (ZL)

Kommando ZL

Mode 1, 2, 3, 4, 5

Valg 0 og 1

Beskrivelse Nulpunktskontakten forbindes til indgang HM. Nulpunktskontakten kan været aktiv høj (1), hvis det f.eks. er en PNP føler eller lav (0), hvis det er en NPN føler. Bemærk at der skal forbindes en modstand mellem HM og en spændingskilde hvis der benyttes en NPN føler.

Brug **ZL=x** Sæt level for nulpunktskontakt, 0 = lav, 1 = høj.

ZL Vis level.

4.10.98 Niveau på nulpunktskontakt (ZS)

Kommando ZS

Mode 1, 2, 3, 4, 5

Beskrivelse Denne kommando viser det aktuelle niveau på nulpunktskontakten. Der vises ikke om kontakten er aktiv eller ej, men om indgangen er høj (1) eller lav (0).

Brug **ZS** Vis level.

Når der opstår fejl i kommunikationen med controlleren, eller når der opstår interne fejl, sender controlleren en fejlmeddelelse. Meddelelsen består af først et 'E', dernæst et fejlnummer, efterfulgt af ':' og til sidst en beskrivende tekst på engelsk. Følgende er et eksempel på en fejlmeddelelse:

Eksempel: E2: Out of range

4.11.1 Forklaring af fejlmeddelelser

E0: No errors

Der har ikke været nogen fejl siden sidste forespørgsel.

E1: Error

Kommandostrengen kan ikke forstås.

Eksempel:

KP 8 K

Giver fejl E1.

Korrektion:

Undersøg hvad der sendes til controlleren og sammenlign med beskrivelsen i denne manual af den pågældende kommando.

E2: Out of range

Parameterværdien er udenfor legalt område.

Eksempel:

CP=100

Denne kommando vil sætte spidsstrømmen til 100 Amp, hvilket ikke kan lade sig gøre. Controlleren melder derfor fejl.

Korrektion:

Vælg en værdi indenfor det legale område.

E3: Number of parameters is wrong

Antallet af parametre er forkert.

Eksempel:

KP8 eller *ES0=9*

Begge disse kommandoer vil give denne fejlmeddelelse.

Korrektion:

Kommandoen KP har kun et tilhørende register og kan derfor kun kaldes med KP.

Kommandoen ES0 benyttes kun, til at vise information, specificering af en parameter giver således ingen mening.

E4: Instruction does not exist

Kommandoen findes ikke.

Eksempel:

ABCDEF

Korrektion:

Benyt en kommando der forstås af controlleren. Se den aktuelle kommando's detaljeret beskrivelse angående det korrekte format.

E5: It is not an instruction

Controlleren har ikke modtaget en rigtig kommando.

Eksempel:

4R

Hvis controlleren ikke benytter adressering vil denne kommando give fejlen E5.

Korrektion:

Benyt en rigtig kommando.

E6: Parameter error or out of range

Der er fejl i parameter eller parameterværdi er udenfor legalt område.

Eksempel:

SP=111111111111 eller *KP=8G7*

Korrektion:

Kontrolleren kan ikke arbejde med så store tal som 111111111111, benyt en værdi indenfor det legale område.

Og der må ikke være bogstaver i en parameterværdi.

E7: Register number error or out of range

Der er fejl i registernummer.

Eksempel:

XP7777 eller *XP4F*

Korrektion:

Benyt et registernummer indenfor det legale område.

Og der må ikke være bogstaver i et registernummer.

E8: Data can not be saved in EEPROM

Opsætningen kan ikke gemmes i EEPROM. Der er opstået en hardware fejl der gør, at mikroprocessoren ikke kan kommunikere med EEPROM.

E9: Checksum error

Controllerens (modtagerens) beregnede checksum er ikke den samme som afsenderens checksum.

Eksempel:

255KP=25F3

Korrektion:

Send kommandoen som 255KP=25DB.

E10: Parameter will be truncated

Controlleren har modtaget en parameterværdi som skal være et heltal.

Eksempel:

VM=1000.8

Korrektion:

Send kommandoen som heltal VM=1000.

E11: No Program available

Der er ingen program i program hukommelsen.

Eksempel:

GO

Korrektion:

Indlæs program eller hent program fra permanent hukommelse (MR1).

E16: Check other Status Register

Der er fejl registreret i det andet status register, læs dette.

E20: Temperature too high

Kontrollerens interne temperatur er for høj.

Korrektion:

Sluk for controlleren. Sørg for en bedre køling af controllerens omgivelser eller sænk den maksimale hastighed (VM).

E21: Current Overload

Controlleren har været overbelastet/kortsluttet.

Korrektion:

Benyt en anden motor eller indsæt spoler på ca. 1mH i serie med motorledningerne. Spolerne skal ved 3 fasede motorer placeres i ledningerne FA, FB, FC, og ved 2 fasede motorer (stepmotorer) placeres i ledningerne FA og FC.

E22 : Power consumption too high

Controlleren trækker for meget effekt fra strømforsyningen.

Korrektion:

Sænk hastigheden/belastningen på motoren eller hæv værdien i PM registeret. Bemærk at PM ikke må indstilles højere end 200W på AMC11B og AMC12B. Se også *Power Management (PM)* side 99

E23: Average Current limit exceeded

Den maksimale tilladelige middelstrøm er overskredet.

Eksempel:

Hastigheden er meget høj.

Korrektion:

Sænk hastigheden indtil fejlen forsvinder.

E24: Supply Voltage exceeds 89 V

Strømforsyningens spænding overskrider 89V.

Eksempel:

Strømforsyningens spænding er for høj, eller motoren decelereres for hurtigt.

Korrektion:

Hvis strømforsyningens spænding er for høj, så skru ned.

Motoren kan ved deceleration sende strøm tilbage til controlleren, hvilket kan give en forøgelse af forsynings spændingen. Decelerationen (AC) skal sænkes indtil fejlen forsvinder. Indsæt evt. en "Power Dump" modstand iflg. afsnit *Power dump udgang* side 39.

E25: Negative Limit Switch active

Negativ endestop er aktiv. Motorens bevægelse i negativ retning standses. Kun positiv bevægelse er nu mulig.

E26: Positive Limit Switch active

Positiv endestop er aktiv. Motorens bevægelse i positiv retning standses. Kun negativ bevægelse er nu mulig.

E27: The motor is not mounted correctly

Motoren er ikke forbundet rigtigt.

Eksempel:

Motoren kører i den forkerte retning.

Korrektion:

Læs afsnit omhandlende motorforbindelser.

E28: Encoder error or position error limit exceeded

Enkoderen er ikke forbundet eller motoren er blokeret.

Eksempel:

Motoren bliver blokeret af en bremse idet der tændes for controlleren. Der kan derfor ikke udføres kontrol af enkoderen. Enkoderen kan også være forbundet forkert eller slet ikke forbundet.

Korrektion:

Sørg for at motoren kan bevæge sig frit, når controlleren tændes. Enkoderens forbindelser skal også kontrolleres.

E29: Supply Voltage exceeds 95 V

Strømforsyningens spænding overskrider 95V.

Eksempel:

Strømforsyningens spænding er for høj, eller motoren decelereres for hurtigt.

Korrektion:

Hvis strømforsyningens spænding er for høj, så skru ned.

Motoren kan ved deceleration sende strøm tilbage til controlleren, hvilket kan give en forøgelse af forsynings spændingen. Decelerationen (AC) skal sænkes indtil fejlen forsvinder.

Problemet kan også afhjælpes ved brug af en power dump modstand benyttes, se afsnit *Power dump udgang* side 39.

E30: The motor is not connected

Motoren er ikke forbundet.

Eksempel:

Motoren bevæger sig ikke.

Korrektion:

Kontroller motorens forbindelse.

E31: Average Current cannot be measured correctly

Middelstrømmen kan ikke måles rigtigt.

Korrektion:

Sluk og tænd for controlleren og se om fejlen forsvinder. Hvis fejlen er vedvarende er der opstået en fejl i hardwaren.

Det er vigtigt at bemærke, at motoren må ikke være i bevægelse når der tændes for controlleren.

E32: HALL element is not connected properly

Hall elementets signaler er ikke forbundet eller er fejlbehæftet.

Korrektion:

Efterse forbindelserne fra Hall elementet og check at Hall registeret og HL er indstillet korrekt. Hvis det ikke ønskes at køre med hall-element skal Hall registeret sættes lig 0 (normal). Se afsnittet *Hall-element type (HALL)* side 83

E33: Position counter overflow

Positionstælleren har overskredet sit maksimumsinterval fra -1073741824 til +1073741823.

Korrektion:

Undgå gentagen brug af *SR* kommandoen eller foretag hyppig nulstilling af systemet. Brug eventuelt *SP* (absolut positionering istedet for *SR*)

E34: Motor controller communications error

Det har ikke været muligt for hovedprocessoren i controlleren at komme i forbindelse med motorprocessoren (Den processor der varetager motor kommuteringen).

Korrektion:

Kontakt JVL angående udbedring.

4.12 Alfabetisk oversigt over kommandoer

| Kommando | Beskrivelse | Grænser | | Mode | | | | | Enhed | Side |
|-------------------------|----------------------------------|--------------|---------------|------|---|---|---|---|----------|------|
| | | Min | Max | 1 | 2 | 3 | 4 | 5 | | |
| AC | Acceleration | 100 | 100000 | | x | x | | | Omdr/m/s | 71 |
| ADDR | Adresse | 0 | 255 | x | x | x | x | x | | 71 |
| AND | Logisk OG operator | | | | | | | | | 71 |
| AO | Aktiver flag i eksternt modul | | | x | x | x | x | x | | 72 |
| AP | Motorens nuværende position | -1073741824 | 1073741823 | x | x | x | x | x | Pulser | 73 |
| APM | Aktuel position for master akse | -1073741824 | 1073741823 | x | | | | | Pulser | 73 |
| BEGIN (f) | Begynd program blok | | | | | | | | | 73 |
| BIAS | Bias efter PID filter | -32767 | 32767 | x | x | x | x | x | | 74 |
| CA | Motorens tilladte middelstrøm | 1 | 12 (6) | x | x | x | x | x | Amp | 74 |
| CHS | Benyt Checksum | 0=nej | 1=ja | x | x | x | x | x | | 74 |
| CL | Vis motorstrøm i forhold til CA | 0 | 100 | x | x | x | x | x | % | 75 |
| CO | Slet flag i eksternt modul | | | x | x | x | x | x | | 75 |
| CP | Motorens maks. spidsstrøm | 1 | 25 (12) | x | x | x | x | x | Amp | 76 |
| CU | Vis motor strøm | | | x | x | x | x | x | Amp | 76 |
| CV | Vis Nuværende hastighed | | | x | x | x | x | x | Omdr/min | 76 |
| D | Pause i program afviklingen | 1 | 1073741823 | | | | | | | 77 |
| DIF | Digital input format | 1 (position) | 2 (hastighed) | | | x | | | | 77 |
| ELSE | ELSE sætning (IF sætning) | | | | | | | | | 77 |
| END (j) | Afslut program blok | | | | | | | | | 78 |
| ENDIF | Afslut IF ELSE sætning | | | | | | | | | 78 |
| EP | Start program flag (ved start) | 0=nej | 1=ja | | | | | | | 78 |
| ES | Fejl (Error) status | 0 | 1 | x | x | x | x | x | | 79 |
| EST | Fejl (Error) status i tekst | 0 | 1 | x | x | x | x | x | | 80 |
| ET | Enkoder type | 0=PNP | 1=NPN | x | x | x | x | x | | 81 |
| EXIT | Stop programmering | | | | | | | | | 81 |
| GEAR | Udveksling ml. master og slave | 0.001 | 32766.999 | x | | | | | | 81 |
| GO | Start program | | | x | x | x | x | x | | 82 |
| H | Stop motor (og program) | | | | x | x | | | | 82 |
| HALL | Motorinitialisering hall-baseret | 0 | 3 | x | x | x | x | x | | 83 |
| HELP | Vis kommandoer | | | x | x | x | x | x | | 83 |
| HL | Hall element type | 0=PNP | 1=NPN | x | x | x | x | x | | 84 |
| IF | IF sætning | | | | | | | | | 85 |
| IL | Integral summeringsgrænse | 0 | 32767 | x | x | x | x | x | | 86 |
| IMCL | Motor initialiserings-niveau | 0 | 100 | x | x | x | x | x | | 86 |
| Fortsættes næste side ! | | | | | | | | | | |

4.12 Alfabetisk oversigt over kommandoer

| Kommando | Beskrivelse | Grænser | | Mode | | | | | Enhed | Side |
|----------|---|------------|------------|------|---|---|---|---|--------------------|------|
| | | Min | Max | 1 | 2 | 3 | 4 | 5 | | |
| IN | Læs input port status | 00000000 | 11111111 | x | x | x | x | x | Bit | 87 |
| INAL | Aktive niveau på indgangene | 00000000 | 11111111 | x | x | x | x | x | Bit | 87 |
| INDEX | Aktiv-niveau på indexindgang | 0 | 1 | x | x | x | x | x | | 88 |
| INPUT | Læs data fra eksternt modul | - | - | x | x | x | x | x | | 88 |
| J | Hop til linie | 0 | 500 | | | | | | Linie | 89 |
| JERK | S-kurve profil | 0 | 65535 | | x | x | | | RPM/S ² | 89 |
| JS | Hop til sub-rutine | 0 | 500 | | | | | | Linie | 89 |
| K | Vis alle K og IL værdier | | | x | x | x | x | x | | 90 |
| KD | Konstant Kd | 0 | 32767 | x | x | x | x | x | | 90 |
| KI | Konstant Ki | 0 | 32767 | x | x | x | x | x | | 90 |
| KP | Konstant Kp | 0 | 32767 | x | x | x | x | x | | 91 |
| KPHASE | Hast. afh. Kommuteringsoffset | 0 | 32767 | x | x | x | x | x | | 91 |
| KVFF | "Feed forward" konstant | 0 | 32767 | x | x | x | x | x | | 92 |
| LED | Lysdiode status | 0=Indgange | 1=Udgange | x | x | x | x | x | | 92 |
| LINE | Vis program linie nummer | 0 | 500 | | | | | | Linie | 92 |
| LIST | Vis program | | | x | x | x | x | x | | 93 |
| MO | Mode: 1=puls, 2=position, 3=register, 4=hastighed, 5=moment | 1 | 5 | x | x | x | x | x | | 93 |
| MR | Hent data fra EEPROM | 0 (AMC12x) | 2 (AMC12x) | x | x | x | x | x | | 94 |
| MS | Gem opsætning i EEPROM | 0 (AMC12x) | 2 (AMC12x) | x | x | x | x | x | | 94 |
| NLS | Negativ endestops niveau | 0=lav | 1=høj | x | x | x | x | x | | 95 |
| OR | Logisk ELLER operator | | | | | | | | | 95 |
| OUT | Niveau på brugerudgange | 00000000 | 11111111 | x | x | x | x | x | Bit | 96 |
| PE | Maksimal Pulsfejl | 0 | 32767 | x | x | x | | | Pulser | 97 |
| PIF | Pulsindgangs format | 1 | 3 | x | | | | | | 98 |
| PL | Vis effektforbrug | 0 | 200 | x | x | x | x | x | % | 99 |
| PLS | Positiv endestops niveau | 0=lav | 1=høj | x | x | x | x | x | | 99 |
| PM | Sæt maks. tilladt effektforbrug | 10 | 2000 | x | x | x | x | x | W | 99 |
| PN | Antal motorfaser | 2 | 3 | x | x | x | x | x | Faser | 100 |
| PO | Motor strøm | 0=på (on) | 1=af (off) | x | x | x | x | x | | 100 |
| POF | Pulsudgangs format | 1 | 2 | x | x | x | x | x | | 101 |
| POFFSET | Fase offset | 1 | 32767 | x | x | x | x | x | Pulser | 101 |
| POL | Antal motorpoler | 2 | 100 | x | x | x | x | x | Poler | 102 |
| PR | Enkoder pulser pr omdr. | 50 | 20000 | x | x | x | x | x | Pulser/O. | 102 |
| PRINT | Udskriv tal/tekst til eksternt modul | | | x | x | x | x | x | | 103 |
| PRM | Enkoder pulser for master | 50 | 20000 | x | | | | | Pulser/O. | 105 |

Fortsættes næste side !

4.12 Alfabetisk oversigt over kommandoer

| Kommando | Beskrivelse | Grænser | | Mode | | | | | Enhed | Side |
|-------------|-------------------------------------|---------------|------------|------|---|---|---|---|----------|------|
| | | Min | Max | 1 | 2 | 3 | 4 | 5 | | |
| PROGRA M | Start program sekvens | | | x | x | x | x | x | | 106 |
| PT | Algoritmisk motor init. tid | 100 | 17694 | x | x | x | x | x | msek. | 106 |
| R | Bruger registre | 0 | 99 | | | | | | | 107 |
| RESET | Genstart controller | | | x | x | x | x | x | | 107 |
| RET | Returner fra sub-rutine | | | | | | | | | 107 |
| RS | Status: 0=stop, 1=acc, 2=max, 3=dec | 0 | 3 | x | x | x | x | x | | 107 |
| RST | Motor/program status i tekst | | | x | x | x | x | x | | 108 |
| SD | Default opsætning | | | x | x | x | x | x | | 109 |
| SH | "Blødt" stop af motor | | | | x | x | | | | 109 |
| SP | Sæt ny position | -1073741824 | 1073741823 | | x | x | | | Pulser | 110 |
| SR | Sæt relativ position | -1073741824 | 1073741823 | | x | x | | | Pulser | 110 |
| SZ | Find nulpunkt | | | x | x | x | x | x | | 110 |
| VE | Firmware version og dato | | | x | x | x | x | x | | 111 |
| VM | Maksimal hastighed | 0 | 65535 | | x | x | x | x | Omdr/m | 111 |
| VOL | Vis Forsyningsspænding | 12 | 100 | x | x | x | x | x | Volt | 111 |
| VV | Vis analog indgangs værdi | -1024 | 1023 | x | x | x | x | x | ADC trin | 112 |
| VVH | Hysteresese for analog indgang | 0 | 200 | | | | x | x | ADC trin | 112 |
| VVL | Neg. Spænding for analog indgang | -10V | Nulpunkt | | | | x | x | ADC trin | 112 |
| VVO | Nulpunkt for analog indgang | -10V | +10V | | | | x | x | ADC trin | 113 |
| VVU | Pos. Spænding for analog indgang | Nulpunkt | +10V | | | | x | x | ADC trin | 113 |
| X | Vis parametersæt | ingen eller 0 | 64 | | | x | | | | 113 |
| XA | Acceleration i parametersæt | 100 | 65535 | | | x | | | Omd/m/s | 114 |
| XP | Position i parametersæt | -1073741824 | 1073741823 | | | x | | | Pulser | 114 |
| XR | Relativ positionering | 0=nej | 1=ja | | | x | | | pulser | 115 |
| XV | Hastighed i parametersæt | 1 | 65535 | | | x | | | omdr/m | 115 |
| ZL | Niveau for nulpunktkontakt | 0 | 1 | | | x | | | | 116 |
| ZS | Status for nulpunktkontakt | 0 | 1 | x | x | x | x | x | | 116 |
| ! | Vis controller type og adresse | | | x | x | x | x | x | | 70 |
| ? | Vis opsætning | | | x | x | x | x | x | | 70 |

5.1

Tekniske data

| Beskrivelse | Min. | Typisk | Max. | Enhed |
|---|-------|--------|---------|-------|
| Forsyning | | | | |
| Forsyningsspænding (AMC10/12) (ben 1A, 2A / P+ og P-) | 15 | | 80 | V DC |
| Forsyningsspænding (AMC11B) | 200 | 230 | 246 | V AC |
| | 90 | 115 | 130 | V AC |
| Effektforbrug (uden tilslutninger) | | 8 | | W |
| Motorudgange FA, FB, FC og FD | | | | |
| Udgangsspænding (afhænger af forsyning) | 0 | | 85 | V RMS |
| Kontinuerlig motorstrøm | 0 | | (6) 12 | A |
| Spidsstrøm | 0 | | (12) 25 | A |
| Effekttab i driver (ved fuld motorstrøm) | | | (15) 25 | W |
| PWM Frekvens | | 24,5 | | kHz |
| Enkoder / Hall-indgang : | | | | |
| Forsyning til enkoder (ben 15A/5VO) | 4,8 | | 5,2 | V DC |
| Tilladt belastning af enkoderforsyning (ben 15A/5VO) | | | 200 | mA |
| Enkoderfrekvens (50% duty-cycle) | 0 | | 500 | kHz |
| Pulsindgange (ben 23A til 26A/XI, YI) | | | 500 | kHz |
| Tilladelig indgangsfrekvens (50% duty-cycle) | | | | kHz |
| Positiv pulsbredde | 1,0 | | | µs |
| Negativ pulsbredde | 1,0 | | | µs |
| Logisk "0" | | | 1,8 | V DC |
| Logisk "1" | 3,8 | | | V DC |
| Brugerindgange IN1-IN8 / CW, CCW, HM : | | | | |
| Indgangsimpedans | 3,2 | | 3,6 | kOhm |
| Logisk "0" | -1 | | 2,5 | V DC |
| Logisk "1" | 4,5 | | 30 | V DC |
| Logisk "0" | - | | 1,0 | mA DC |
| Logisk "1" | 2,0 | | - | mA DC |
| Brugerudgange O1 - O8 : | | | | |
| Spændingsforsyning | 6 | | 28 | V DC |
| Belastningsstrøm pr. udgang | | | 250 | mA DC |
| Analogindgang AIN : | | | | |
| Indgangsspænding (nominel) | -10,0 | | 10,0 | V DC |
| Indgangsimpedans | | 20 | | kOhm |
| Powerdump udgang PDO : | | | | |
| Spænding | 0 | | 100 | V DC |
| Tilsluttet shuntmodstand | 15 | | | Ohm |
| Diverse : | | | | |
| Arbejdstemperatur | 0 | | 45 | °C |
| Vægt (AMC10B og AMC12B) | | 720 | | gram |
| Vægt (AMC10C og AMC12C) | | 1100 | | gram |
| Vægt (AMC11B) | | 3100 | | gram |

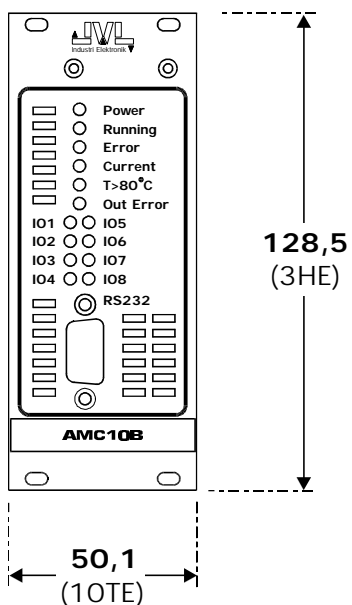
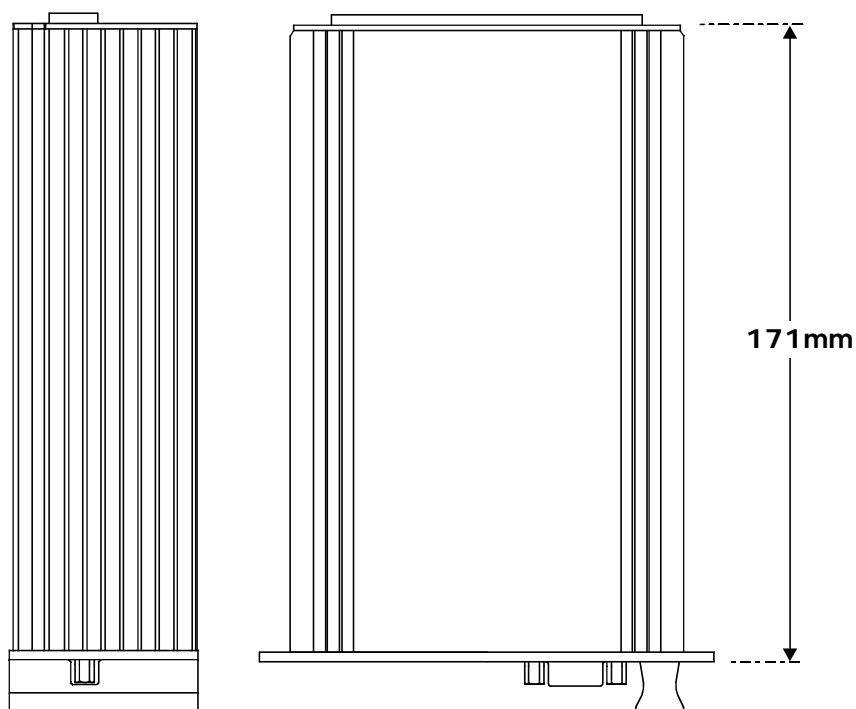
() = Værdier gældende for AMC1xB

5.2

Fysiske mål

5.2.1

Fysiske mål for AMC10B

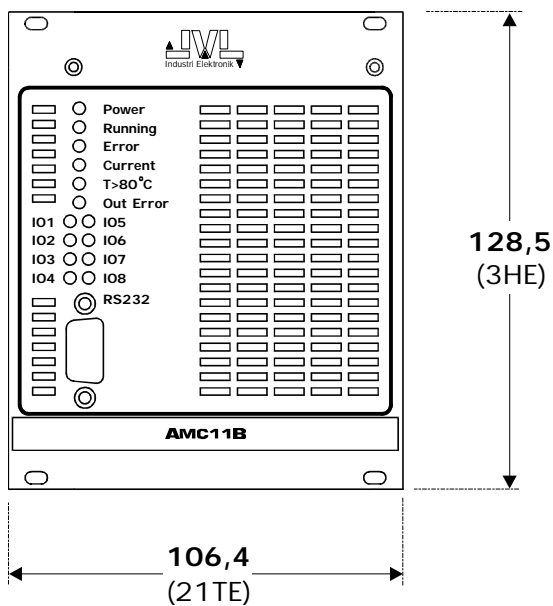
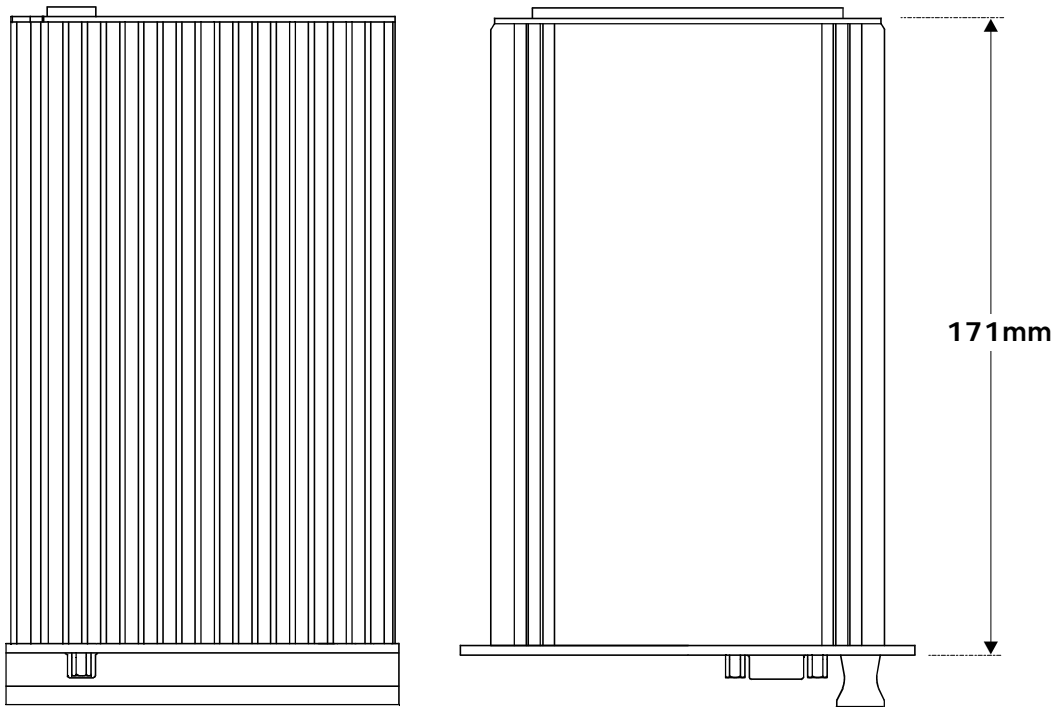


TT0010DK

5.2

Fysiske mål

5.2.2 Fysiske mål for AMC11B og AMC12B

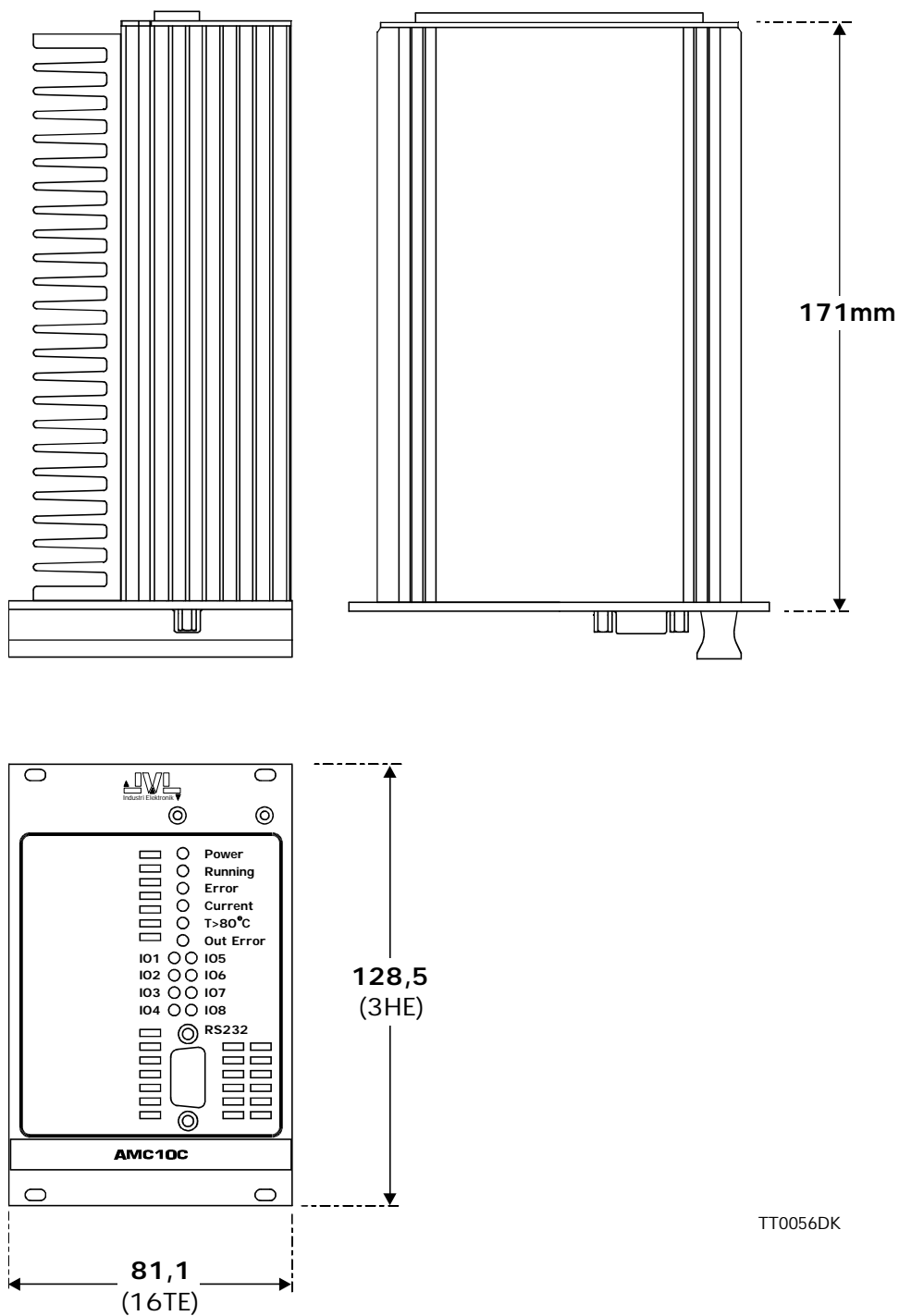


TT0055DK

5.2

Fysiske mål

5.2.3 Fysiske mål for AMC10C og AMC12C

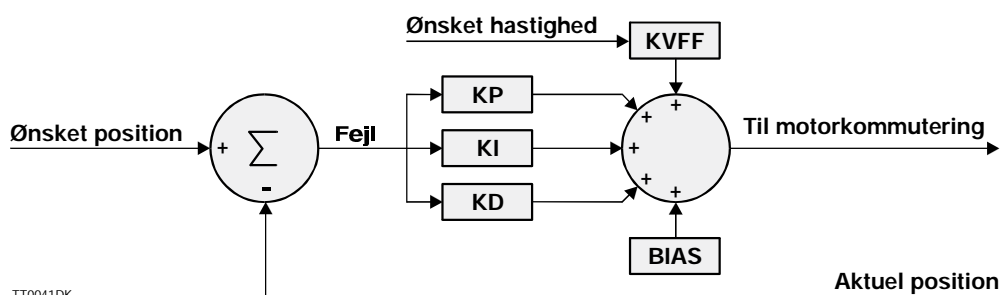


TT0056DK

5.3

Servo loop

Der benyttes et PID (Proportional-Integral-Differentiel) servo loop i servocontrolleren. Sammenhængen mellem de enkelte parametre kan ses af nedenstående illustration.



5.3.1

Formel for servo loop

Filteret kan matematisk beskrives med følgende formeler

$$\begin{aligned} P &= E_n * KP \\ I &= (S+E_n) * KI \\ S &= S+E_n \\ D &= (E_n-E(n-1))*KD \end{aligned}$$

Hvor:

$$\begin{aligned} E_n &= \text{Øjeblikkelig fejlværdi} \\ E &= \text{Forrige fejlværdi} \\ S &= \text{Sum af alle tidligere målinger} \end{aligned}$$

Bemærk at S begrænses af integral-summeringsparameteret IL .

Servoloopet kan justeres med parametrene IL , KP , KI , KD , $KVFF$, $BIAS$.

Dioderne (LED) på controlleren benyttes også udover de markerede funktioner til at indikere vitale fejl. I det følgende beskrives først standardfunktionerne dernæst de specielle funktioner. Se også afsnittet *Fejlstatus i tekst (EST)* side 80. Vedr. fejlmeddelelser fra controlleren

5.4.1 Error diode

Error dioden lyser, når der opstår en fatal fejl. Ved en fatal fejl forstås en fejl, der har en så stor indflydelse at motoren ikke kan køre. Dette kan f.eks. være fejl i enkoder kabel, blokering af motor, temperatur fejl, kortslutning af motorudgang, overspænding, middelstrøm overskredet.

5.4.2 Current diode

Current dioden lyser hvis den specificerede middelstrøm (CA) overskrides i længere tid. *Error* dioden tændes også. Dioden lyser også, når der er sket en overbelastning. Systemet skal genstartes efter en overbelastning. Se afsnittet *Genstart controller (RESET)* side 107 .

5.4.3 T>80°C diode

T>80°C dioden lyser, når controllerens interne temperatur overstiger 80°C. Controlleren skal herefter genstartes.

5.4.4 Out Error

Out Error dioden lyser, når der opstår en fejl på en af de otte udgange O1-O8.

5.4.5 Fire dioder blinker skiftevis

Hvis de fire dioder *Running*, *Error*, *Current* og *T>80°C* lyser skiftevis som et "løbelys", er det en indikation om PROM fejl. Når controlleren tændes, bliver checksummen i controllerens programhukommelse (PROM) kontrolleret. Hvis den preprogrammerede checksum ikke passer med den beregnede checksum vil controlleren ikke bevæge motoren. PROM'en kan være defekt prøv evt. en genstart af controller.

5.4.6 Fire dioder blinker samtidig

Hvis de 4 dioder *Running*, *Error*, *Current* og *T>80°C* blinker, er det en indikation om motor eller enkoder fejl. Når controlleren tændes, kontrolleres, om motor og enkoder er forbundet rigtigt. PWM signalet til motoren forøges gradvis, indtil der registreres en bevægelse eller PWM signaler er på 50%.

Controlleren kan således checke om:

1. Motoren er rigtig forbundet, d.v.s. køre i den rigtige retning.
2. Motoren er blokeret, dvs. bruger meget strøm, uden motor bevæger sig.
3. Enkoderen er ikke forbundet rigtigt.

Check, om motor eller enkoder er forbundet rigtigt. Benyt kommandoen *EST* afsnit *Fejlstatus i tekst (EST)* side 80 for yderligere information fra controller.

Ved installation og brug af controlleren kan der opstå forskellige fejl. Flere fejl kan controlleren selv give mere information om ved brug af kommandoen *EST* - se afsnittet *Fejlstatus i tekst (EST)* side 80. Nogle fejl "ligner" andre fejl. I det følgende beskrives nogle af de mest almindelige fejl og deres mulige løsning.

Enkoderen er ikke forbundet.
Benyt kommandoen *ET* og vælg den rigtige enkoder type.

Fire dioder blinker skiftevis.
Se forgående afsnit.

Hastigheden er ikke rigtig.
Der er vigtigt at indstille servo-konstanterne. Systemet har ikke mulighed for at holde den rigtige hastighed, hvis servo loopet ikke er justeret.

Hastigheden er ikke rigtig, selvom servo konstanter er indstillet.
Det er vigtigt, at enkoderens pulser/omdr. er sat rigtigt. Benyt kommandoen *PR*, se afsnittet *Enkoder pulser (PR)* side 102.

Motoren kører ikke til den rigtige position ved valg af XP0.
XP0 benyttes til nulpunktsøgningsfunktionen, og har derfor en anden funktion end de andre positionsregistre.

Motor og enkoder er forbundet rigtigt, men melder stadig fejl.
Check, om enkoder typen er sat rigtigt med kommandoen *ET* (side 81).

Motoren leverer ikke det rigtige moment
Der er vigtigt at indstille servo konstanterne. Systemet har ikke mulighed for at give det rigtige moment, hvis servo loopet ikke er justeret.

Fire dioder blinker.
Der er problemer med enkoderen eller motoren. Enkoderen er faldet af eller motoren er blokeret. Hvis fejlen opstår når der tændes for systemet læs da afsnittet *Indstilling af enkoderopløsning* side 134 . Fejlen kan også opstå under kørsel. I det tilfælde hvor enkoderen og motoren tilsyneladende er forbundet rigtigt. Kontrollér med kommandoen *PE* afsnittet *Maksimal pulsfejl (PE)* side 97 den maksimale tilladelige pulsfejl. Kontrollér også enkodertypen med *ET*, se afsnittet *Enkoder type (ET)* side 81 .

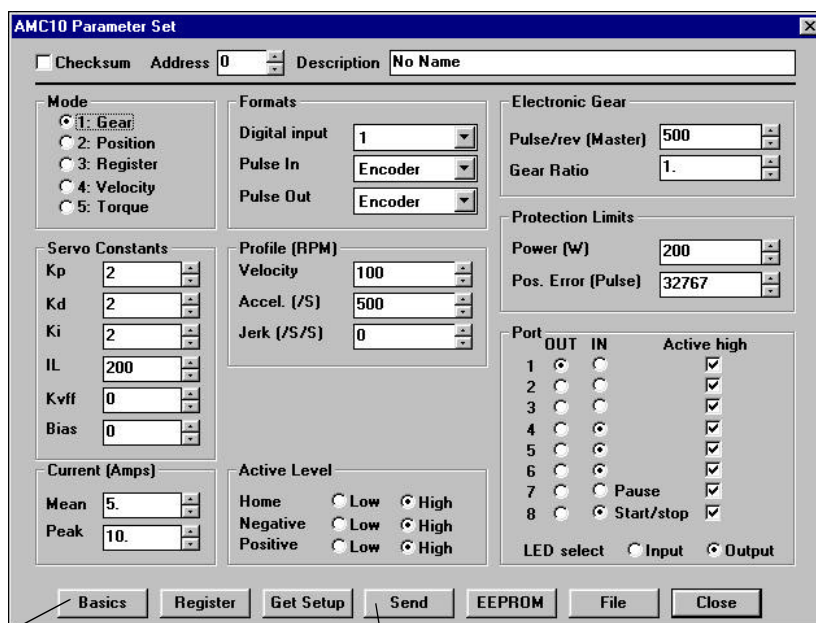
5.6 Tilslutning af ukendt motor

Dette afsnit følges hvis controlleren skal indstilles til en motor der ikke er omfattet af paramterlisten i Motoware.

Følgende checkliste følges:

1. Find følgende data for den pågældende motor og indstil controlleren derefter.
Antal poler - parameter *POL* - Se afsnittet *Indstilling af antal motorpoler* side 137.
Antal faser - parameter *PN*- Se afsnittet *Indstilling af antal motorfaser* side 138.
Antal enkoderpulser pr. omdrejning og enkodertype - parameter *PR*, *ET*, og *INDEX* - se afsnittet *Indstilling af enkoderopløsning* side 134.
Motorens tilladte middelstrøm/peakstrøm - parameter *CA* og *CP* - se afsnittet *Justering af motorstrøm* side 139
Det kan anbefales at justere controlleren uden at benytte Hall-elementer, selvom motoren har disse. Tilslut først Hall-elementer efter endt justering.
Hvis motoren ikke har hall-elementer, følges afsnittet *Opstart af motor uden hall element* side 144.
Hvis motoren har hall-elementer, kan disse evt. benyttes se afsnittet *Indstilling af Hall-element* side 141 .
2. Indstil nu de øvrige vitale parametre, der vedkommer motortype herunder :
KP, KD og KI - se afsnittet *Justering af servo-parametre* side 16.
Hastighedsafhængig kommuteringsoffset - parameter *KPHASE* - se afsnittet *Justering af KPHASE* side 143.
For opsætning af øvrige funktioner i controlleren se afsnittet *Software* side 47.

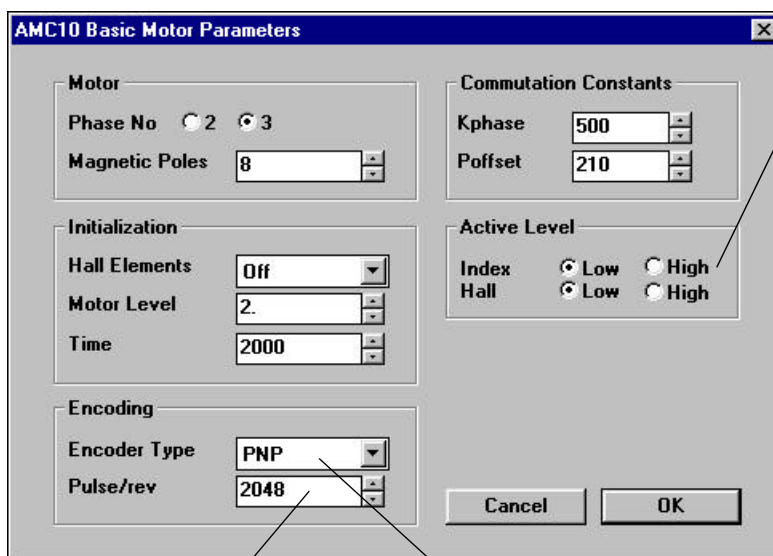
5.6 Tilslutning af ukendt motor



① Vælg basis parametre her

⑤ Send opsætning til controller

Indstilling af aktivt niveau på index indgang ④



② Enkoderopløsning indstilles her

Enkoderstype indstilles her ③

5.6.1 Indstilling af enkoderopløsning

For at opnå korrekt hastighed og kommutering af motoren skal antallet af enkoderpulser pr. omdrejning programmeres. Der skal her benyttes det antal pulser, som er angivet for enkoderen. Bemærk at controlleren internt multiplicerer dette antal med 4, således at eksempelvis en enkoder med 500 pulser pr. omdrejning i realiteten får 2000 pulser pr. omdrejning. Skal motoren rotere 1 omdrejning skal positioneringskommandoen baseres på de 2000 pulser. Enkoderopløsningen kan ikke sættes lavere end motorens poltal gange 128.

5.6 Tilslutning af ukendt motor

Hvis enkoderopløsningen forsøges indstillet lavere vil controlleren svare med fejlmeddelelsen : *E2 Out of range*. Det er derfor vigtigt i hvilke rækkefølge enkoderopløsning og pøtal indstilles.

Enkoderopløsningen skal indstilles i intervallet 256-20000 pulser pr. omdrejning. I AMC10 Parameter Set vinduet vælges BASICS knappen og AMC10 Basic Motor Parameters vinduet kommer frem. Indstil enkoderopløsningen i feltet *Pulse/rev (S)*. Tryk på OK og AMC10 Parameter Set vinduet kommer frem igen. Send informationen ved at trykke *Send*. Når den nye opsætning er sendt, gives muligheden for, at gemme opsætningen i permanent hukommelse og/eller genstarte controlleren.

Ændres enkoderopløsningen via on-line editoren, benyttes kommandoen *PR*.

Eksempel :

PR=2048 (enter) Sætter enkoderopløsningen lig 2048 pulser pr. omdrejning.

PR(enter) Viser den indstillede enkoderopløsning.

Afslut ved at gemme indstillingen i controllerens permanente hukommelse ved at skrive *MS (enter)*.

5.6.2 Indstilling af enkodertype

Enkoderen kan være af både PNP og NPN type. Desuden accepteres både et balanceret og ubalanceret signal fra en standard 2-kanals inkremental enkoder. Ved enkodertilslutning se afsnit *Enkoderindgang* side 29.

Feltet *Encoder Type* bestemmer hvilken enkodertype der kan tilsluttes controlleren. Hvis der benyttes en enkoder med balanceret udgang kan der ses bort fra denne indstilling.

Benyttes der derimod ubalanceret enkoder af NPN typen skal feltet indstilles til *NPN*. Er enkoderen af typen PNP indstilles vinduet til *PNP*.

I AMC10 Parameter Set vinduet vælges BASICS knappen og AMC10 Basic Motor Parameters vinduet kommer frem. Indstil enkodertypen i feltet *Encoder Type*. Tryk på OK og AMC10 Parameter Set vinduet kommer frem igen. Send informationen ved at trykke *Send*. Når den nye opsætning er sendt, gives muligheden for, at gemme opsætningen i permanent hukommelse og/eller genstarte controlleren.

Ændres enkodertype via on-line editoren, benyttes kommandoen *ET*.

Eksempel :

ET=0 (enter) Sætter enkodertype lig PNP.

ET=1 (enter) Sætter enkodertype lig NPN.

ET(enter) Viser den indstillede enkodertype.

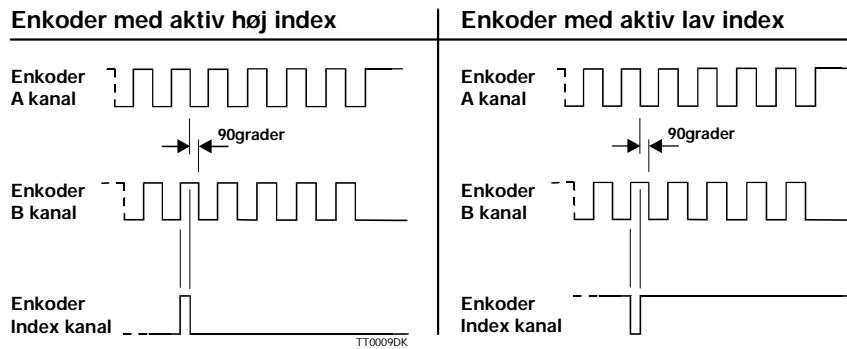
Afslut ved at gemme indstillingen i controllerens permanente hukommelse ved at skrive *MS (enter)*.

5.6 Tilslutning af ukendt motor

5.6.3 Indstilling af index-indgang

Det kan anbefales at benytte en enkoder med index puls. Har den anvendte enkoder en sådan index puls, skal controllerens index indgang (EZ1 og EZ2) indstilles til den pågældende enkoders index polaritet. Hvis index pulsen er aktiv høj d.v.s. at den kun optræder som høj en enkelt gang pr. omdrejning, skal feltet *High* afkrydses, ellers skal feltet *Low* afkrydses.

Illustration af aktive niveauer :



Ændres niveauet for index indgangen via on-line editoren, benyttes kommandoen *INDEX*.

Eksempel :

`INDEX=1 (enter)`

Sætter aktivt niveau til logisk høj.

`INDEX (enter)`

Viser det indstillede aktive niveau.

5.6 Tilslutning af ukendt motor

Polantal indstilles her.

AMC10 Basic Motor Parameters

Motor

Phase No 2 3

Magnetic Poles 8

Initialization

Hall Elements Off

Motor Level 2

Time 2000

Encoding

Encoder Type PNP

Pulse/rev 2048

Commutation Constants

Kphase 500

Poffset 210

Active Level

Index Low High

Hall Low High

Cancel OK

5.6.4 Indstilling af antal motorpoler

Den tilsluttede motors polantal skal kendes for at controlleren kan fungere. Hvis poltallet står forkert, vil controlleren gå i fejl efter opstart eller ved første kørsel, og give fejlmeddelelsen "E28: The encoder is not connected or the motor is blocked". Poltallet kan sættes i intervallet 2-100. De fleste 3 fasede servomotorer har et poltal på 2, 4, 6 eller 8 poler. En typisk stepmotor har 100 poler (200 step/omdr.).

Poltallet indtastes lettest i parametervinduet.

I AMC10 Parameter Set vinduet vælges BASICS knappen og AMC10 Basic Motor Parameters vinduet kommer frem. Indstil poltallet i feltet *Magnetic poles*. Tryk på OK og AMC10 Parameter Set vinduet kommer frem igen. Send informationen ved at trykke *Send*. Når den nye opsætning er sendt, gives muligheden for, at gemme opsætningen i permanent hukommelse og/eller genstarte controlleren.

Ændres poltallet via on-line editoren, benyttes kommandoen *POL*.

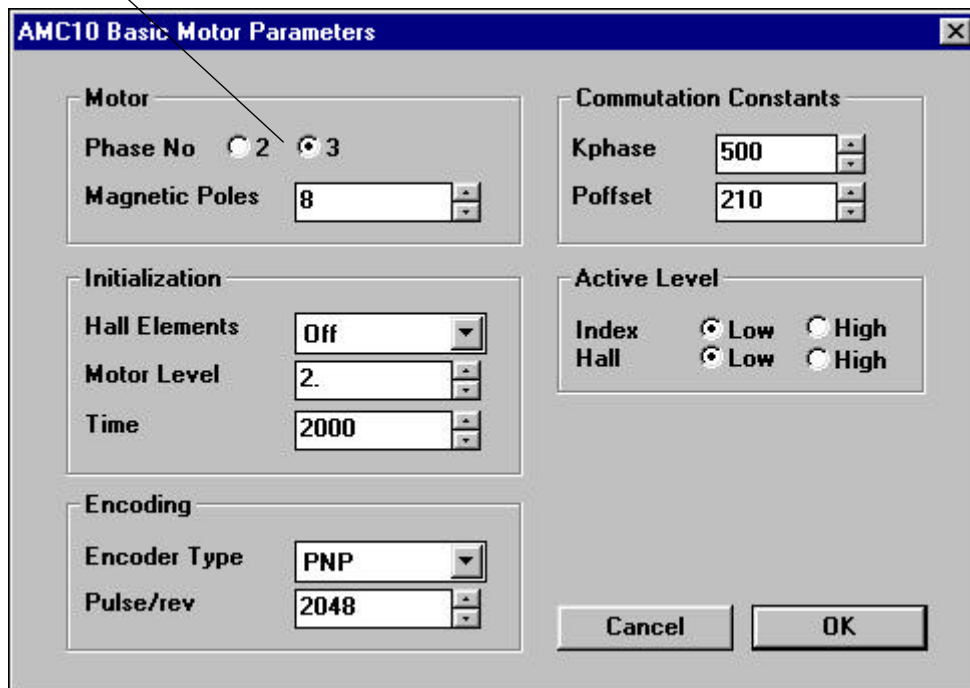
Eksempel :

POL=8 (enter) Sætter antal poler lig 8 (4 sæt).
POL (enter) Viser det indstillede antal poler.

Afslut ved at gemme indstillingen i controllerens permanente hukommelse ved at skrive *MS (enter)*.

5.6 Tilslutning af ukendt motor

Antal faser indstilles her



5.6.5 Indstilling af antal motorfaser

Controlleren muliggør tilslutning af motorer med 2 eller 3 faser. En stepmotor er typisk med 2 faser og en AC-servomotor er typisk med 3 faser.

Hvis kontrolleren indstilles til 2 faser benyttes alle 4 motorudgange benævnt FA, FB, FC og FD. Hvis kontrolleren indstilles til 3 faser benyttes kun motorudgangene benævnt FA, FB og FC.

Når antallet af faser skal indstilles vælges BASICS i AMC10 Parameter Set vinduet knappen og AMC10 Basic Motor Parameters vinduet kommer frem. Afkryd enten rubrikken 2 eller 3 i Phase No. rubrikken. Tryk på OK og AMC10 Parameter Set vinduet kommer frem igen. Send informationen ved at trykke *Send*. Når den nye opsætning er sendt, gives muligheden for, at gemme opsætningen i permanent hukommelse og/eller genstarte kontrolleren.

Se eventuelt også afsnittet *Motortilslutning* side 21, vedr. forbindelser.

Ændres antallet af faser via on-line editoren, benyttes kommandoen *PN*.

Eksempel :

PN=2 (enter) Sætter antallet af motorfaser lig 2 (stepmotor typisk).

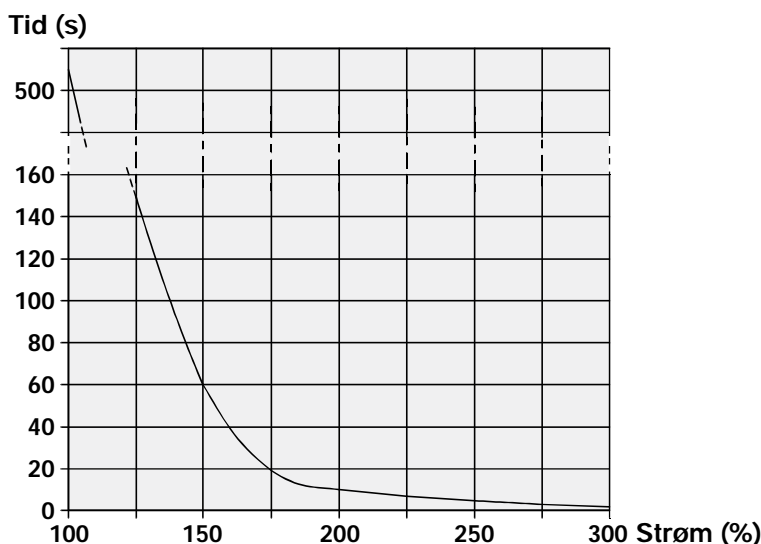
PN (enter) Viser det indstillede fase antal.

Afslut ved at gemme indstillingen i controllerens permanente hukommelse ved at skrive *MS* (enter).

5.6 Tilslutning af ukendt motor

5.6.6 Justering af motorstrøm

En AC-servomotor eller en stepmotor har 2 strømgrænser, der skal overholdes for ikke at overophede motoren eller mindske motorens levetid. Disse strømme er henholdsvis motorens tilladte middelstrøm og spidsstrøm. Hvis den indstillede middelstrøm overskrides vil controlleren gå i overstrømsfejl. Middelstrømmen bliver overvåget iflg. nedenstående illustration. Bemærk at den indstillede middelstrømsværdi CA , svarer til 100%.



5.6.7 Justering af motorstrøm for AC-servomotor

Justering af middelstrøm.

Se den aktuelle motors datablad og find middelstrømsværdien. Denne kan være benævnt som "Continuous Current", "Rated Current", eller "Nominal Current".

Middelstrømmen indstilles via controllerens parameter CA .

Eksempel :

Ønskes middelstrømmen indstillet til 1 Amp., skrives $CA=1$ (enter).

Herefter vil controlleren under ingen omstændigheder tillade, at motoren trækker en middelstrøm større end 1,0 Amp i længere tid.

Bemærk, at middelstrømmen kun kan indstilles med en opløsning på 0,1 Amp.

Justering af spidsstrøm.

Den anden strøm er motorens spidsstrøm, som indstilles på følgende måde.

Se den aktuelle motors datablad og find den tilladelige spidsstrømsværdi. Denne kan være benævnt som "Peak Current", "Instantaneous max. Current", eller "Stall Current".

De fleste motortyper kan tåle en spidsstrøm, som er 3-4 gange større end middelstrømmen.

Spidsstrømmen indstilles via controllerens parameter CP .

Eksempel :

Ønskes spidsstrømmen indstillet til 4 Amp., skrives $CP=4$ (enter).

Herefter vil controlleren under ingen omstændigheder tillade, at motoren trækker en spidsstrøm større end 4,0 Amp.

Bemærk, at spidsstrømmen kan indstilles med 1 decimal (xx.x).

5.6 Tilslutning af ukendt motor

5.6.8 Justering af motorstrøm for stepmotor

Justering af middelstrøm.

Se den aktuelle motors datablad og find middelstrømsværdien. Denne er som regel benævnt som *Rated Phase Current*.

Middelstrømmen indstilles via controllerens parameter *CA*.

Eksempel :

Ønskes middelstrømmen indstillet til 1Amp., skrives $CA=1$ (enter).

Herefter vil controlleren under ingen omstændigheder tillade, at motoren trækker en middelstrøm større end 1,0 Amp i længere tid.

Bemærk, at middelstrømmen kan indstilles med 1 decimal (xx.x).

Justering af spidsstrøm.

Den anden strøm er motorens spidsstrøm, som indstilles på følgende måde.

Spidsstrømmen skal indstilles 20% højere end *CA*.

Spidsstrømmen indstilles via controllerens parameter *CP*.

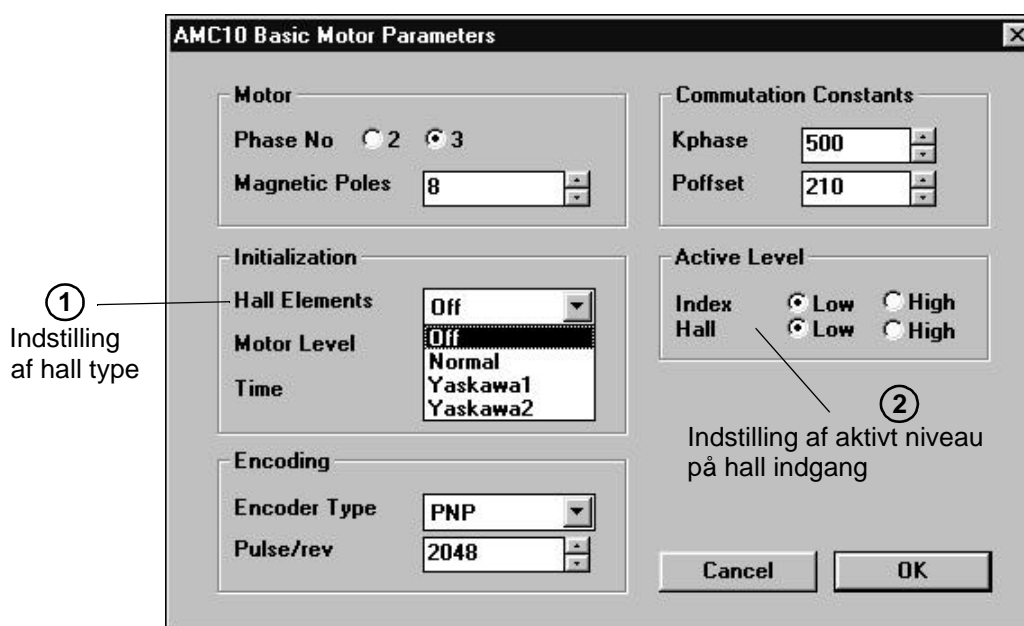
Eksempel :

Ønskes spidsstrømmen indstillet til 4 Amp., skrives $CP=4$ (enter).

Herefter vil controlleren under ingen omstændigheder tillade, at motoren trækker en spidsstrøm større end 4,0 Amp.

Bemærk, at spidsstrømmen kan indstilles med 1 decimal (xx.x).

5.6 Tilslutning af ukendt motor



5.6.9 Indstilling af Hall-element

Controlleren kan initialiseres med eller uden Hall element i motoren. Normalt benyttes Hall elementet ikke hvis motoren gerne må bevæge sig under opstart. Er dette tilfældet sættes Hall registeret lig 0. Ønskes det derimod at motoren står fuldstændigt stille under opstart, skal der benyttes Hall element i motoren og Hall registeret sættes lig 1, 2 eller 3. Hall elementet benyttes under opstart til at fortælle kontrolleren, hvor motoren befinder sig således at kommuteringselektronikken kan fastlåse det påtrykte magnetfelt ud for motorens aktuelle position uden at motoren flytter sig. Den information der kommer fra motorens inkremental enkoder kan ikke fortælle kontrolleren dette og kan derfor ikke benyttes. Hall elementet benyttes kun ved opstart.

Der kan vælges mellem følgende Hall typer.

| HALL register | Motoware felt | Funktion |
|---------------|---------------|--|
| HALL = 0 | Off | Opstart uden HALL |
| HALL = 1 | Normal | Normal HALL - benyt HLA, HLB og HLC indgange |
| HALL = 2 | Yaskawa 1 | Yaskawa HALL kodning type 1. Benyt kun enkoderindgange inkl. Index kanal. |
| HALL = 3 | Yaskawa 2 | Yaskawa HALL kodning type 2. Benyt kun enkoderindgange inkl. Indeks kanal. |

Bemærk at Yaskawa motorer har deres HALL signaler kodet sammen med enkoder signalerne inkl. indeks-signal. Dette minimerer antallet af ledninger mellem motor og controller. Se også afsnit *Eksempler på tilslutning af motor* side 145

Når hall type skal indstilles vælges BASICS i AMC10 Parameter Set vinduet knappen og AMC10 Basic Motor Parameters vinduet kommer frem. Indstil hall type i feltet *Hall elements*. Tryk på OK og AMC10 Parameter Set vinduet kommer frem igen. Send informationen ved at trykke *Send*. Når den nye opsætning er sendt, gives muligheden for, at gemme opsætningen i permanent hukommelse og/eller genstarte kontrolleren.

5.6 Tilslutning af ukendt motor

Ændres hall type via on-line editoren, benyttes kommandoen *HALL*.

Eksempel :

HALL=1 (enter) Sætter hall type til normal hall føler.

HALL (enter) Viser den indstillede hall type.

Afslut ved at gemme indstillingen i controllerens permanente hukommelse ved at skrive *MS (enter)*.

5.6.10 Indstilling af Hall type.

For at opnå en korrekt dekodning af Hall elementet i motoren (såfremt det benyttes), er det vitalt at Hall opsætningen er rigtig. Hall elementer kan være af både PNP og NPN typen. Desuden accepteres både et balanceret og ubalanceret signal fra Hall elementet. Ved Hall element tilslutning se afsnittet *Hall-indgang* side 32 .

Hvis der benyttes Hall element med balanceret udgang kan der ses bort fra indstillingen af hall type. Benyttes der derimod ubalanceret Hall element af NPN eller PNP typen skal der foretages en indstilling i *AMC10 Basic Motor Parameters* vinduet's *Hall* felt.

Hvis der er tale om et Hall element af NPN typen sættes feltet til *High*. Er det PNP typen sættes feltet til *Low*.

Hvis der benyttes en Yaskawa motor er Hall typen ligegyldig, idet Hall signalet ligger indkodet i selve enkodersignalet og Hall-indgangen derfor ikke benyttes.

Når valget er foretaget tryk da på OK og AMC10 Parameter Set vinduet kommer frem igen. Send informationen ved at trykke *Send*. Når den nye opsætning er sendt, gives muligheden for, at gemme opsætningen i permanent hukommelse og/eller genstarte controlleren.

Ændres Hall type via on-line editoren, benyttes kommandoen *HL*.

Eksempel :

HL=0 (enter) Sætter Hall type lig PNP (low).

HL=1 (enter) Sætter Hall type lig NPN (high).

HL (enter) Viser den indstillede Hall type.

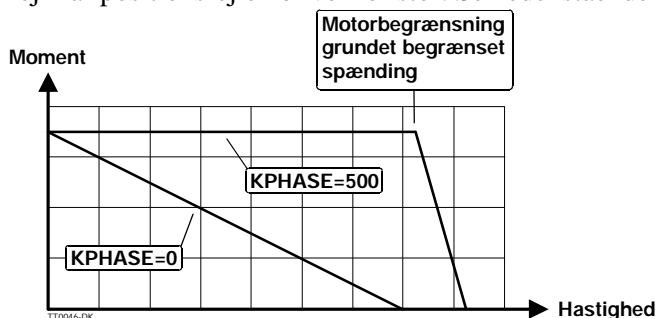
Afslut ved at gemme indstillingen i controllerens permanente hukommelse ved at skrive *MS (enter)*.

5.6 Tilslutning af ukendt motor

5.6.11 Justering af KPHASE

Controlleren indeholder en parameter benævnt KPHASE. Denne parameter er afgørende for hvor langt kommuteringen af motoren skal forskydes i forhold til motorens aktuelle position. KPHASE er hastighedsafhængig, hvilket vil sige at den har en voksende betydning desto hurtigere motoren kører.

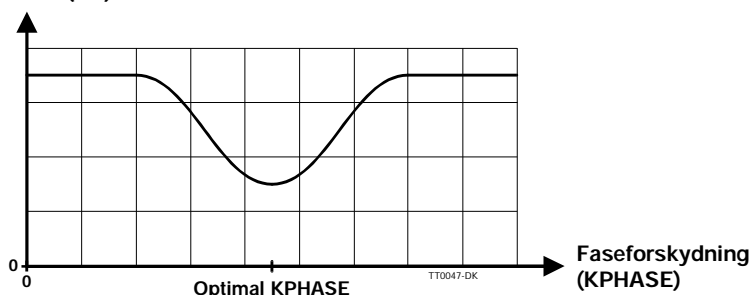
Det er af vital betydning for systemets ydeevne at denne parameter bliver justeret korrekt, idet en misjustering vil afstedkomme at motoren ikke kan yde nok moment ved høje hastigheder. I værste fald vil motoren ikke køre med fuld hastighed og systemet vil gå i fejl når positionsfejlen bliver for stor. Se nedenstående illustration.



Justeringen af KPHASE foretages under installation af systemet og foretages efter følgende procedure.

1. Start Motoware og controller. Gå ind i "On line editor".
2. Check at der er kontakt med controller ved at skrive ? (enter).
3. Sørg for at motoren kan køre med en vilkårlig hastighed og længde, uden at tilsluttet mekanik bliver beskadiget.
4. Sæt controlleren i Mode 2 ved at skrive $MO=2$ (enter).
5. Sæt tophastigheden i controlleren så den svarer ca. til det motorfabrikanten angiver som værende den maksimalt tilladte hastighed med belastning, typiske 3000 omdr./min.. Dette gøres ved at skrive $VM=3000$ (enter).
Indstil endvidere KPHASE til værdien 400 ved at skrive $KPHASE=400$ (enter)
6. Lad motoren køre en lang strækning ved at give kommandoen $SP=99999999$.
7. Motoren skulle nu gerne køre. Hvis controlleren går i fejl efter kort tids kørsel er KPHASE indstillet forkert eller forsyningsspændingen til systemet er ikke indstillet til samme værdi som motorens nominelle spænding. Gå evt. tilbage til pkt. 5 og angiv en lavere hastighed eller benyt en højere forsyningsspænding som svarer til motorens nominelle spænding, dog maksimalt 80VDC.
8. Når motoren kører med en så høj hastighed som muligt kan KPHASE indstilles. Undersøg motorstrømmen ved at sende kommandoen CU (enter). Controlleren vil derefter svare med beskeden (eksempel) $CU=1.0$, hvilket indikerer at den aktuelle motorstrøm er lig 1,0 ampere. Prøv nu at justere KPHASE op eller ned i værdi og stop når KPHASE har den værdi hvor motorstrømmen er lavest.
9. Til sidst gemmes den fundne værdi i controllerens permanente hukommelse, ved at sende kommandoen MS (enter).

Motorstrøm (CU)



5.6 Tilslutning af ukendt motor

Indstilling af hall type - vælges "OFF" - ingen hall element. ①

Indstilling af init. niveau. ②

Indstilling af init. tid. ③

5.6.12 Opstart af motor uden hall element

Controlleren kan initialiseres med eller uden Hall element i motoren. En stepmotor har f.eks. ikke noget hall element, og i dette tilfælde vil den eneste mulighed være at initialisere efter følgende metode.

1. Controllerens hall-indgang skal være slået fra. Indstil vinduet *Hall Elements* til *OFF*, eller send kommandoen *Hall=0*.
2. Efter opstart vil motoren blive påtrykt en strøm som er specificeret via vinduet *Init. Motor Level* eller kommandoen *IMCL*.
3. Strømmen vil blive påtrykt i den tid som der specificeres via vinduet *Init. Time* eller kommandoen *PT*. Den specificerede tid er udtrykt millisekunder.
4. Efter denne tid som typisk skal sættes til 1000-3000 msek, vil motoren være kørt hen til en ligevægtsposition ud for det dannede magnetfelt, og controlleren vil herefter låse sin kommuteringselektronik til den aktuelle motorposition. Initialiseringen er afsluttet og controlleren er driftsklar.

Indstil de omtalte parametre og tryk på OK og AMC10 Parameter Set vinduet kommer frem igen. Send informationen ved at trykke *Send*. Når den nye opsætning er sendt, gives muligheden for, at gemme opsætningen i permanent hukommelse og/eller genstarte controlleren.

Ønskes det at motoren står fuldstændigt stille under opstart, skal der benyttes Hall element i motoren og Hall registeret sættes lig 1, 2 eller 3. I dette tilfælde kan der ses bort fra *PT* og *IMCL* parameteren. Se afsnittet *Indstilling af Hall-element* side 141

5.7 Eksempler på tilslutning af motor

I dette afsnit er opstillet en række eksempler på tilslutning af 2 og 3 fasede motorer, herunder også hvorledes de vitale parametre i controlleren skal indstilles. Vedr. generel opsætning og finjustering se afsnit *Generelt om installation* side 12.

5.7.1 Eksempel 1

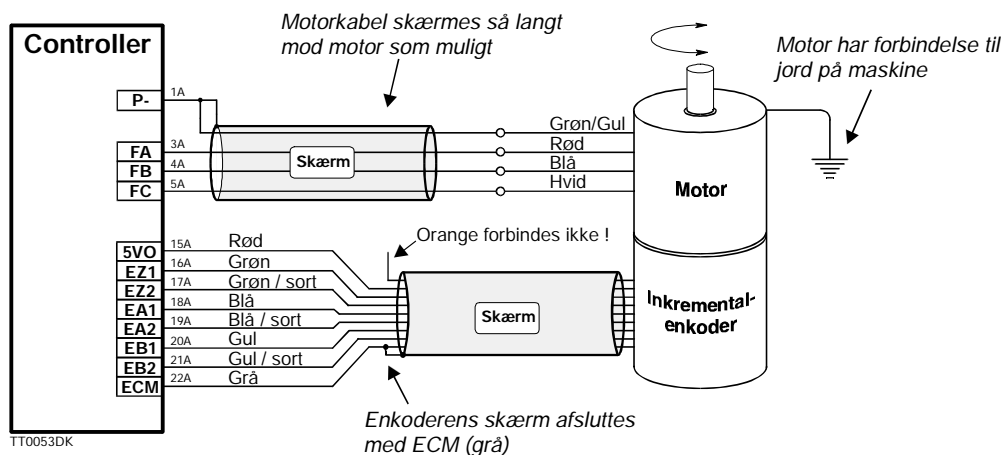
Yaskawa 3 faset motor : 200W/200V - Type betegnelse SGM-02A3xxx.
Filnavn i motorware under parametersetup : "SGM-02A3xxx (200V/200W)"
Hvis opsætningen udføres uden Motorware parametersetup, følges nedenstående parameter indstillinger.
Indstil alle parametre til default ved i "On Line Editor" at skrive *SD (enter)*.
Indlæs derefter følgende:

Controller parameter :

| | | |
|---------------|-----------|---------------|
| KP = 7 | KI = 20 | KD = 70 |
| KVFF = 0 | IL = 100 | KPHASE = 1500 |
| POFFSET = 220 | PR = 2048 | POL = 8 |
| PN = 3 | CA = 4 | CP = 13 |
| HALL = 2 | INDEX = 0 | HL = 0 |

Afslut ved at gemme de indtastede parametre - skriv *MS (Enter)*.

Genstart controller ved at skrive *Reset (enter)*.



5.7 Eksempler på tilslutning af motor

5.7.2 Eksempel 2

MAE stepmotor : Type betegnelse HY200-3437-460-A8

Der benyttes en 4000 pulser enkoder med indexkanal.

Filnavn i motorware under parametersetup : "MAE HY200-3437-460-A8"

Hvis opsætningen udføres uden Motorware parametersetup, følges nedenstående parameter indstillinger.

Indstil alle parametre til default ved i "On Line Editor" at skrive *SD (enter)*.

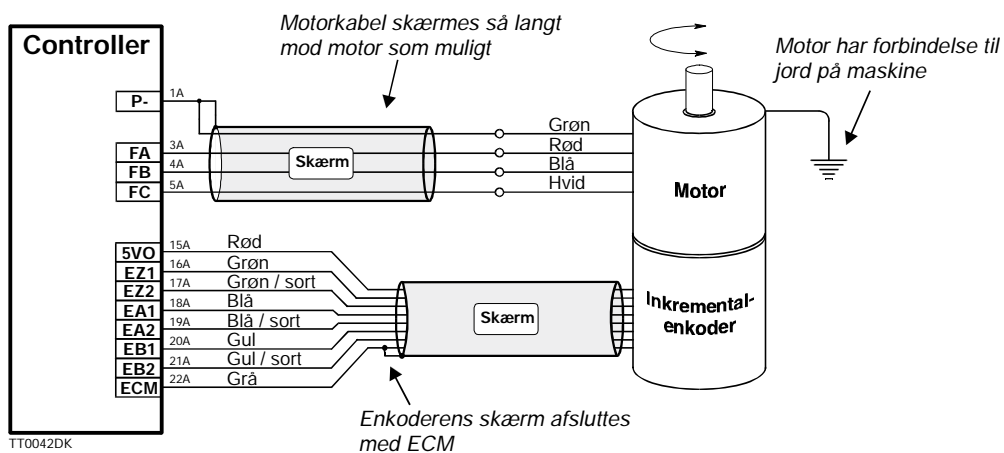
Indlæs derefter følgende:

Controller parameter :

| | | |
|-------------|-----------|---------------|
| KP = 7 | KI = 20 | KD = 70 |
| KVFF = 0 | IL = 100 | KPHASE = 1500 |
| POFFSET = 0 | PR = 2048 | POL = 100 |
| PN = 2 | CA = 4 | CP = 13 |
| HALL = 0 | INDEX = 0 | |

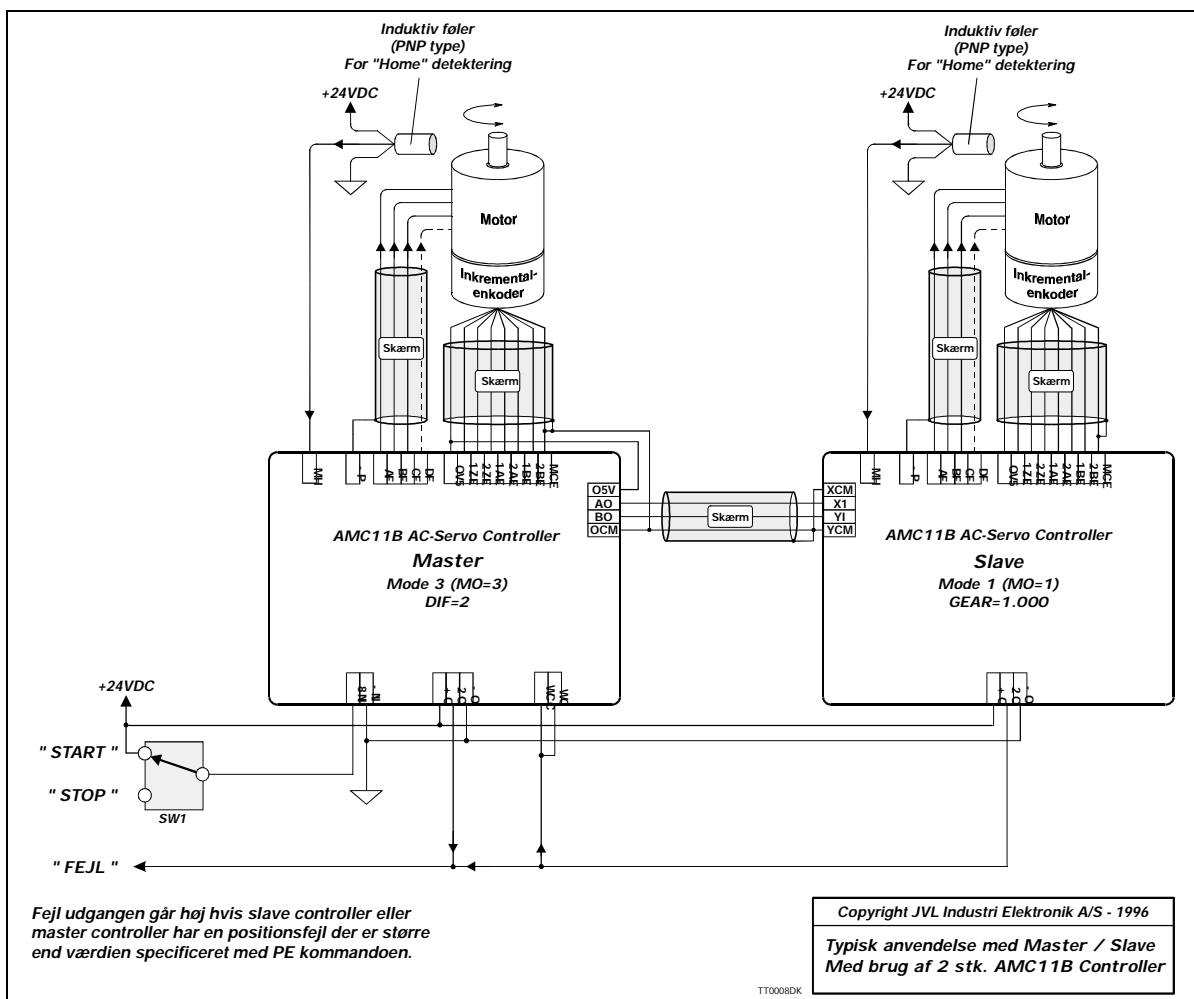
Afslut ved at gemme de indtastede parametre - skriv *MS (Enter)*.

Genstart controller ved at skrive *Reset (enter)*.



5.8

Typiske anvendelser



5.9

Connectorboard

Som tilbehør til controlleren kan JVL levere et connectorboard type CON13. Dette connectorboard muliggør tilslutning via stikbare skrueterminaler. Nedenstående illustrationer viser tilslutnings-mulighederne. Connectorboardet kan enten monteres bagest i et 19" rack eller på selve styringen.

Forbindelses-skema

Connectorboard til AMC10, AMC11 og AMC12

Type: "CON13"

Controller-forsyning

P+ +15-80V Ind
P- Stel

Motorudgang

P- Motorkabel skærm
FA Motor, fase A
FB Motor, fase B
FC Motor, fase C
FD Motor, fase D

Power dump

PDO Power dump udgang

Enkoder- og Hall-indgang

HLA Hall-indgang A
HLB Hall-indgang B
HLC Hall-indgang C
SVO +5V ud til enkoder/Hall-føler
EZ1 Enkoder, kanal Z1 (index)
EZ2 Enkoder, kanal Z2 (index)
EA1 Enkoder, kanal A1
EA2 Enkoder, kanal A2
EB1 Enkoder, kanal B1
EB2 Enkoder, kanal B2
ECM Enkoder-/Hall stel

Pulsindgange

XCM Stel for X-pulsindgang
XI X-Pulsindgang
YCM Stel for Y-Pulsindgang
YI Y-Pulsindgang

Pulsudgange

O5V +5V Ind til forsyning af udgange
AO Pulsudgang A
BO Pulsudgang B
OCM Stel til pulsudgange

Analog ind-/udgang

AIN +/-10V Analogindgang
ACM Stel til analog ind-/udgang
▲AX2 Analogudgang +/- 5V til 2. akse m.v.

Brugerudgange

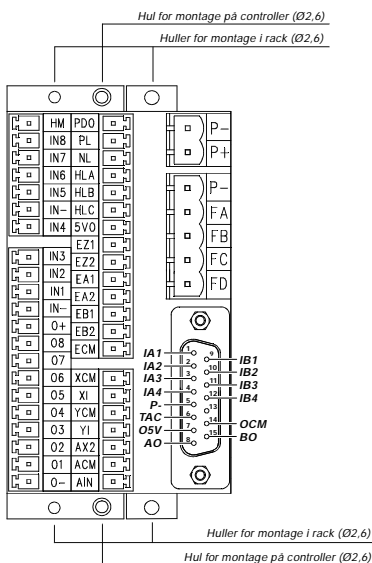
O+ +5-30V ind til forsyning af udgang
O1 Udgang 1
O2 Udgang 2
O3 Udgang 3
O4 Udgang 4
O5 Udgang 5
O6 Udgang 6
O7 Udgang 7
O8 Udgang 8
O- Udgangsstel

Bruger-/stopindgange

PL Positive endstop
NL Negative endstop
HM Nulstillingsindgang
IN1 Indgang 1
IN2 Indgang 2
IN3 Indgang 3
IN4 Indgang 4
IN5 Indgang 5
IN6 Indgang 6
IN7 Indgang 7
IN8 Indgang 8
IN- Indgangsstel

Diverse

▲IA1 Industribus, terminal IA1
▲IA2 Industribus, terminal IA2
▲IA3 Industribus, terminal IA3
▲IA4 Industribus, terminal IA4
▲IB1 Industribus, terminal IB1
▲IB2 Industribus, terminal IB2
▲IB3 Industribus, terminal IB3
▲IB4 Industribus, terminal IB4
P- Stel til Industribus m.v.
TAC Momentovervågning 0-5VDC



Terminaler mærket med "▲" er kun tilgængelige på controller type AMC12

Symbols

! 67
 ? 67
A
 AC 68
 Acceleration 68, 111
 ADDR 68, 92
 Adresse 38, 68
 AIN 52
 AND 68
 APM 70
B
 BEGIN 70
 Brugerindgange 24, 26
C
 CA 53
 Checksum 38, 39, 71
 CHS 72
 CLK terminal 33
 CP 80
 CR 38
D
 D 74
 D (pause) 74
 Deceleration 68
 DELAY 74
E
 ELSE 74
 END 74, 75
 Endestopindgange 25
 ENDIF 75
 ES 77
 EST 77
 ET 78
 Execute Program Flag (EP)
 75
 EXIT 78
F
 Fejlmeddelelser 115
G
 Galvanisk isoleret 24, 25, 26
 Gearmode 47
 GO 79
H
 Hastighed 108
 Hastighedsmode 8, 52, 66
 Hop til programlinie (J) 86

I
 IN 84
 INAL 84
 INDEX 85
 Indgange
 Brugers- 26
 Digitale- 24
 endestop- 25
 Retnings- 33
 Indgangsspænding 33
 INPUT 85
J
 J 84, 85, 86
 JERK 86
K
 K 87
 KD 89
 Kommando 38
 Kommandoforamt 38
 Kommunikationsforamt 38
 Kommunikations-hastighed
 38
 Kondensator 31
 KP 88
 KPHASE 88
L
 LINE 89
M
 Momentmode 9, 66
 Motorfaser 97
N
 Negative Limit Switch 92
 Niveau på nulpunktskontakt
 113
 NLS 92
 NPN 33
 NPN udgang 24, 25, 26
 Nulpunkts søgnings funktion
 65
O
 O1 49
 OUT 93
 Overspænding 31
P
 P- terminal 31
 P+ terminal 31
 PE 94
 PID 128

PIF 95
 PLS 96
 PNP 24, 25, 26
 PO 97
 POFFSET 98
 Positioneringsmode 6, 48
 Positive Limit Switch 96
 PROGRAM 103
 PROM fejl 129
 Pull-Up modstand 24, 25, 26
R
 Registermode 7
 RESET 104
 RET 104
 Retningsindgang 33
S
 Servo 128
 Spidsstrøm 80
 Start program (GO) 79, 82
 Start programmering (PRO-
 GRAM) 103
 Stelterminal 24, 25, 26
 Steppulsindgang 33
 Steppulsmode 5
 Stop programmering (EXIT)
 78
 Strømforsyning 31
U
 Udråbstegn 67, 68
V
 VM 52, 108
 VOL 108
 VVL 109
X
 XAn 111
 XPn 111
 XRn 112
 XV 112
Z
 ZL 113
 ZS 113

JVL Industri Elektronik A/S