

GRAF CET

Tekniken grundades i Frankrike på 1970-talet och ligger till grund för ett standardiserat programspråk i enlighet med standard IEC 61131-3.

Tanken med systemet är att det ska vara oberoende av val av teknik för styrsystem d.v.s. man ska kunna använda vilken teknik som helst för styrningen (styrsystemet) som t.ex. relätekniska, mekaniska, pneumatiska, dator, PLC etc.

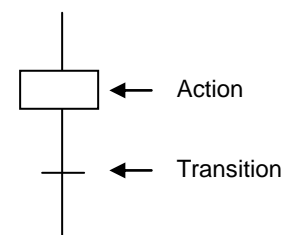
Grafcet används för att beskriva sekventiella händelseförlopp och där särskilda villkor ska vara uppfyllda för att styrsystemet ska utföra nästa händelse. Principen för Grafcet är att den tillämpar steg s.k. ”action” och mellan varje action övergångsvillkor s.k. ”transition”.

Action

Vid en action ska ny händelse ske. En ny händelse kan t.ex. vara att en motor startar eller att en pneumatisk cylinder går tillbaka. En action kan flera olika händelser.

Transition

En transition är ett villkor. För att en action ska aktiveras måste dess transition vara ”sann”. När villkoret i en transition är uppfyllt, säger man att transition är ”sann”. Då aktiveras efterföljande action. I en transition kan flera villkor ingå.



Fördelar med Grafcet

Det finns flera fördelar med Grafcet, bla.

- Full kontroll på varje enskild action.
- Endast en action (i enkel sekvens) kan vara aktiv åt gången
- Felsökning i en sekvens blir enkel och snabb
- Transition till deaktiverade action påverkar inte andra action, systemet blir ”felsäkert”
- Extrafunktioner som t.ex. jogging kan enkelt läggas till
- S.k. ”loopar” kan enkelt skapas.
- Det är lätt att modifiera med flera steg i ett senare skede
- Logiken följer en struktur och blir därmed enkel, även vid komplexa styrningar

Nackdelar med Grafcet

Man kan i stort säga att fördelarna överväger nackdelarna i många tillämpningar. Det finns dock en nackdel man bör känna till

- Den logiska lösningen blir något större än andra metoder

Grafcet och PLC

Programvara för programmering av PLC har ofta tilläggseditorer för just Grafcet och IEC-standard. Som exempel kan nämnas

- Mitsubishi kallas editorn för SFC (Sequential Function Chart) och även STL-instruktion
- Siemens kallas editorn för S7Graph

I följande avsnitt kommer Grafcet och IEC-standarderna att beskrivas utifrån ett PLC-system men lösningarna kan likväl realiserars med t.ex. reläteknik eller dator.

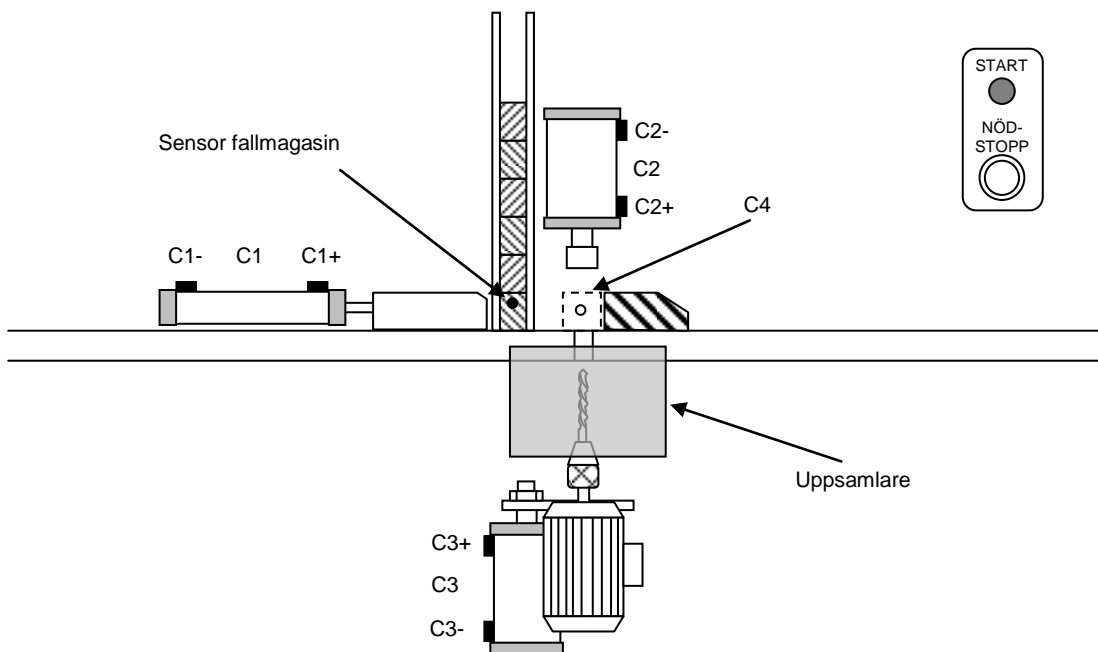
Exempel på en sekvens

En borrenhet ska borra ett hål i ett ämne. Flera ämnen laddas i ett fallmagasin. Ämnena går vidare till ett fixeringsläge som håller de på plats. Därefter ska borrning ske. När borrningen är avslutad ska ämnet putas ut och ned i en uppsamlare. Processen ska ske automatiskt och startas via en start-knapp.

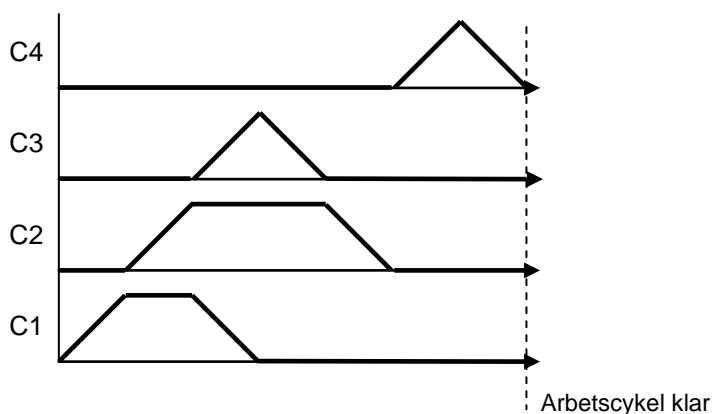
Funktion.

Operatören trycker på start-knappen. Om det finns bitar i fallmagasinet ska arbetscykeln starta. En sensor finns i fallmagasinets botten som känner av om det finns ämnen.

Om startvillkoret är uppfyllt går cylinder C1 ut och matar fram ett ämne mot ett fast stopp. När ämnet är fixerat går cylinder C2 ned och pressar fast det med stor kraft. Sedan går C1 tillbaka och när den nått sitt bakre läge faller det ned ett nytt ämne, klart för inmatning. Därefter startar motorn till borren och C3 matar upp borrenheten som borrar ett hål i ämnet. När borrenheten har nått sitt främre läge är borrning klar och då ska borrenheten gå tillbaka och därefter ska motorn till borren stanna. Sedan går C2 tillbaka och när den nått sitt bakre läge ska C4 gå ut snabbt och putta bort ämnet så det faller i en uppsamlare. C4 går därefter direkt tillbaka och arbetscykeln är klar.



Samtliga cylindrar har både bakre (-) och främre (+) sensor. Så här se följdidiagrammet ut.



Denna arbetscykel kan alltså delas in i 7 steg eller action.

I detta exempel ska vi använda ett PLC av typen FX. Vi får skapa en I/O-lista på de variabler vi behöver.

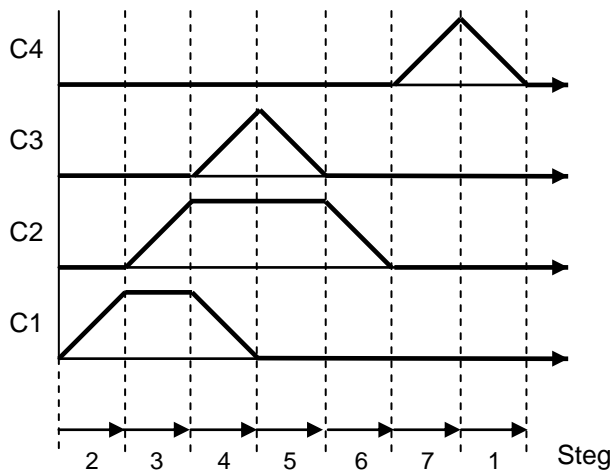
Ingångar

Operand (adress)	Identifier(benämning)
X0	STARTKNAPP
X1	ÄMNE I FALLMAGASIN
X2	C1-
X3	C1+
X4	C2-
X5	C2+
X6	C3-
X7	C3+
X10	C4-
X11	C4+

Utgångar

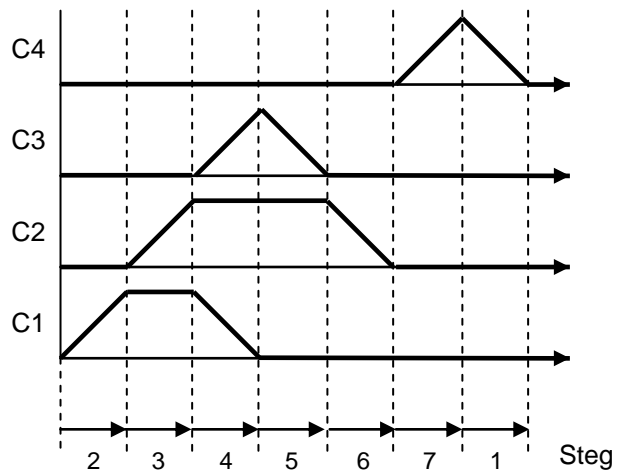
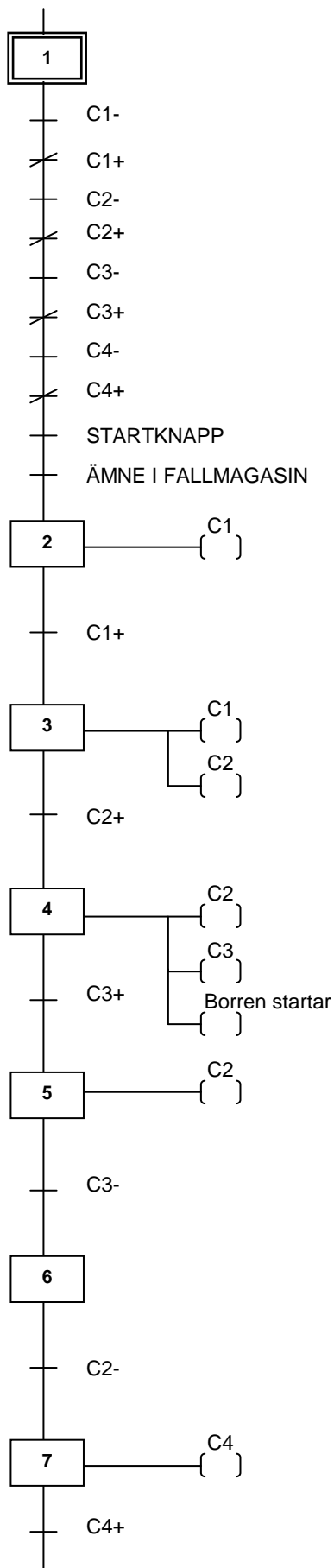
Operand (adress)	Identifier(benämning)
Y0	C1
Y1	C2
Y2	C3
Y3	C4
Y4	
Y5	
Y6	
Y7	BORRMOTOR

Sekvensen kan nu skapas i Grafcet. Först delar man in de olika händelserna i steg.

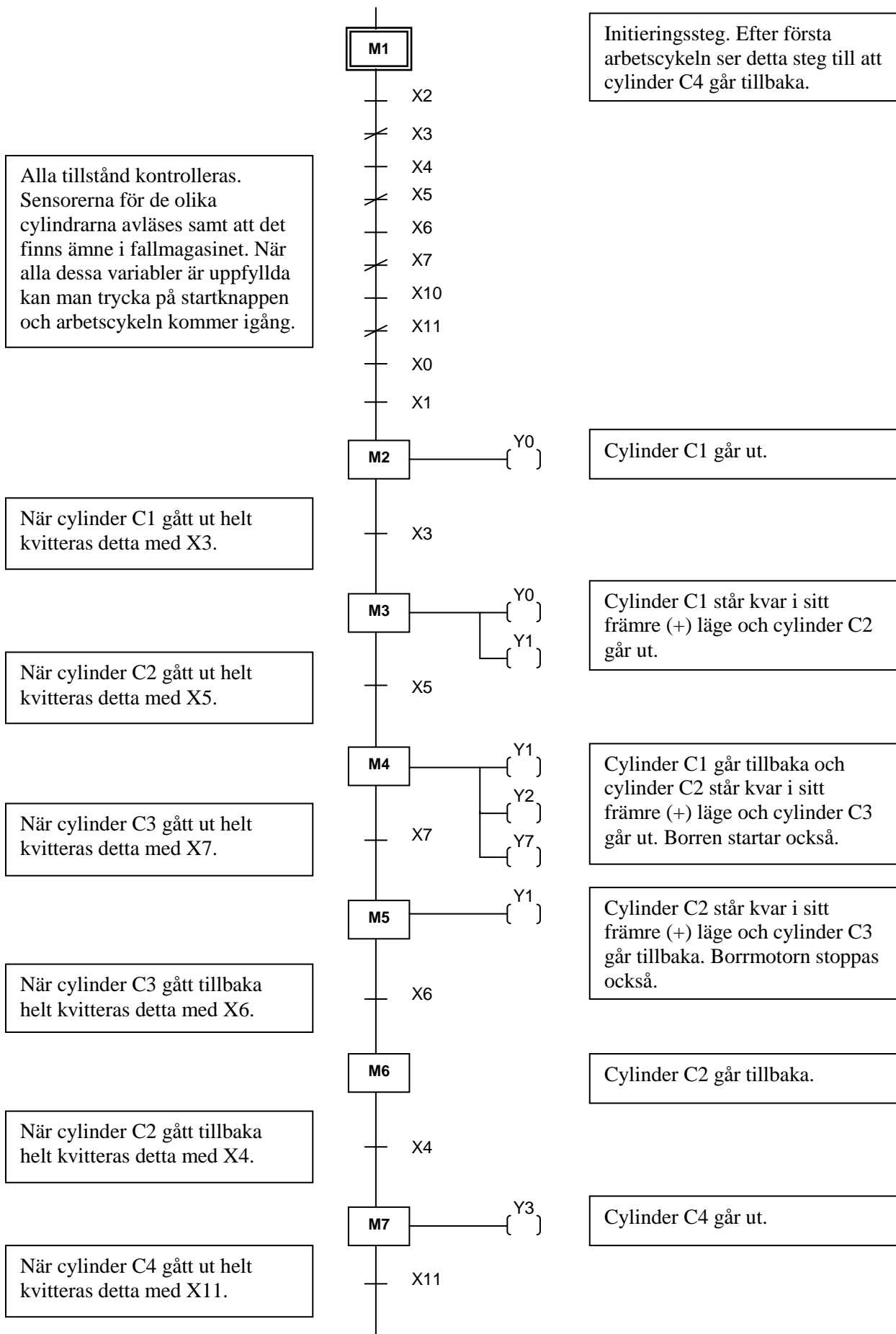


Här har vi de 7 stegen. Vissa steg som 3, 4 och 5 utför fler än en sak. Lägg märke till att steg 1 ligger som sista steg. Detta beror på att detta steg inte gör något "aktivt", lägger ut någon "1". Vi återkommer mer om detta längre fram.

Grafet nätverket för sekvensen.



Här har benämningarna byts ut mot PLC operander (I/O).



Programmet i instruktionslista (Mitsubishi)

```

    STEG 1
0 LD      M7      STEG_7
1 AND     X11     C4+
2 OR      M1      STEG_1
3 OR      M8002   INITPULS
4 ANI     M2      STEG_2
5 OUT     M1      STEG_1

    STEG 2
6 LD      M1      STEG_1
7 AND     X2      C1-
8 ANI     X3      C1+
9 AND     X4      C2-
10 ANI    X5      C2+
11 AND    X6      C3-
12 ANI    X7      C3+
13 AND    X10     C4-
14 ANI    X11     C4+
15 AND    X0      STARTKNAPP
16 AND    X1      ÄMNE_FALL_MAG
17 OR     M2      STEG_2
18 ANI    M3      STEG_3
19 OUT    M2      STEG_2

    STEG 3
20 LD     M2      STEG_2
21 AND    X3      C1+
22 OR     M3      STEG_3
23 ANI    M4      STEG_4
24 OUT    M3      STEG_3

    STEG 4
25 LD     M3      STEG_3
26 AND    X5      C2+
27 OR     M4      STEG_4
28 ANI    M5      STEG_5
29 OUT    M4      STEG_4

    STEG 5
30 LD     M4      STEG_4
31 AND    X7      C3+
32 OR     M5      STEG_5
33 ANI    M6      STEG_6
34 OUT    M5      STEG_5

    STEG 6
35 LD     M5      STEG_5
36 AND    X6      C3-
37 OR     M6      STEG_6
38 ANI    M7      STEG_7
39 OUT    M6      STEG_6

    STEG 7
40 LD     M6      STEG_6
41 AND    X4      C2-
42 OR     M7      STEG_7
43 ANI    M1      STEG_1
44 OUT    M7      STEG_7

    UTGÅNG Y0 (C1)
45 LD     M2      STEG_2
46 OR     M3      STEG_3
47 OUT    Y0      CYL_1

    UTGÅNG Y1 (C2)
48 LD     M3      STEG_3
49 OR     M4      STEG_4
50 OR     M5      STEG_5
51 OUT    Y1      CYL_2

    UTGÅNG Y2 (C3)
52 LD     M4      STEG_4
53 OUT    Y2      CYL_3

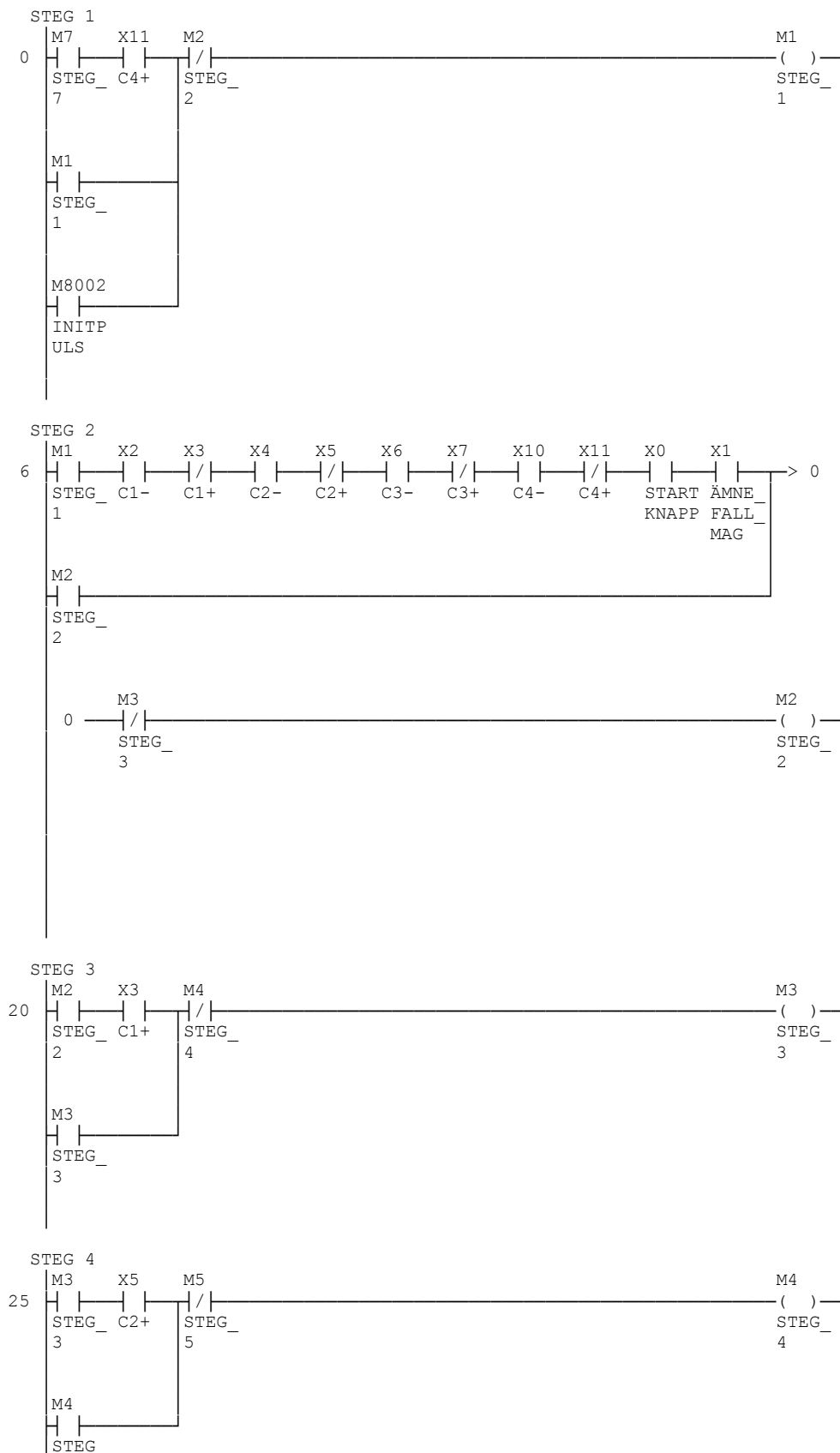
    UTGÅNG Y3 (C4)
54 LD     M7      STEG_7
55 OUT    Y3      CYL_4

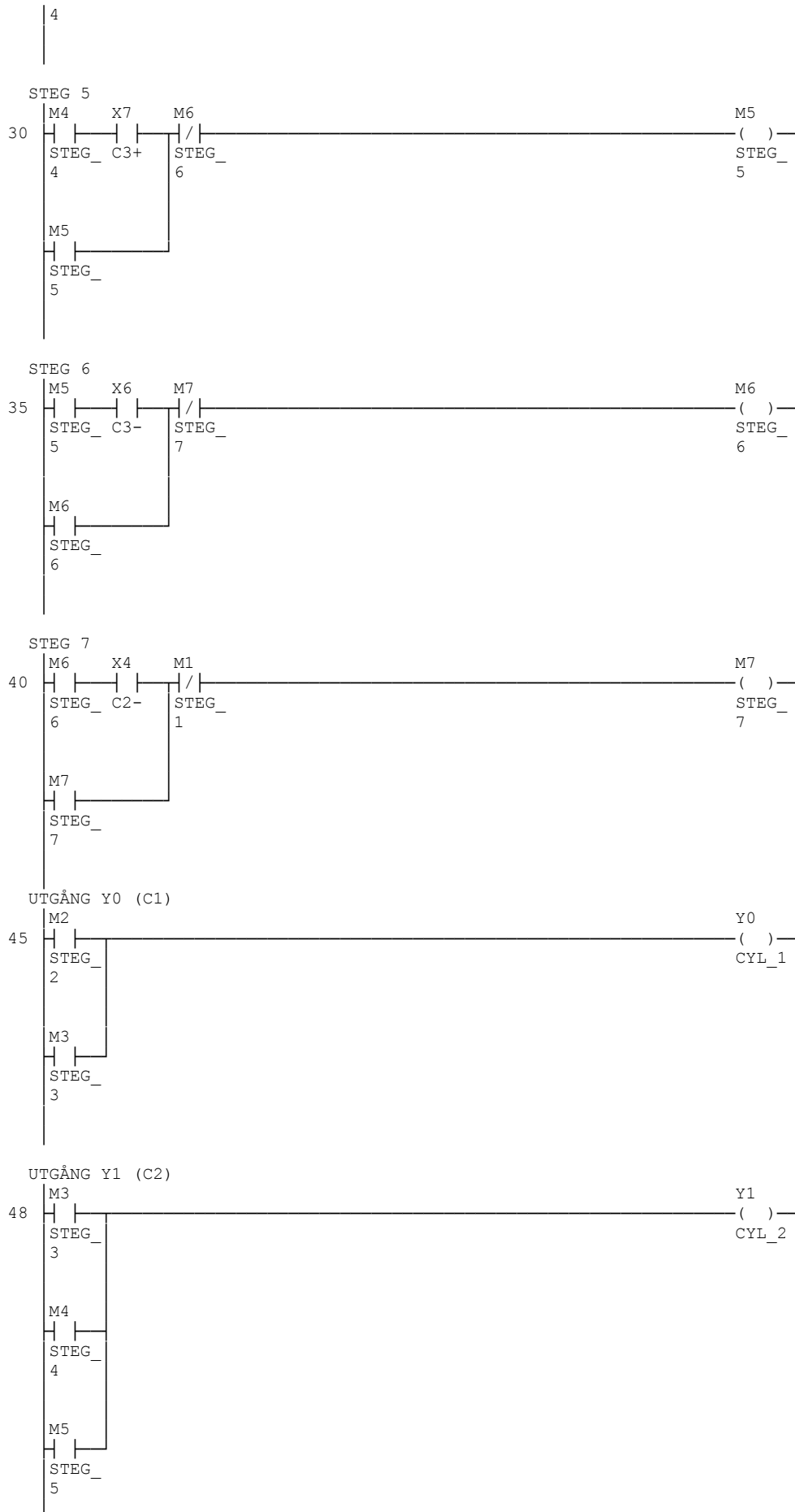
    UTGÅNG Y7 (BORRMOTOR)
54 LD     M4      STEG_4
55 OUT    Y7      BORRMOTOR

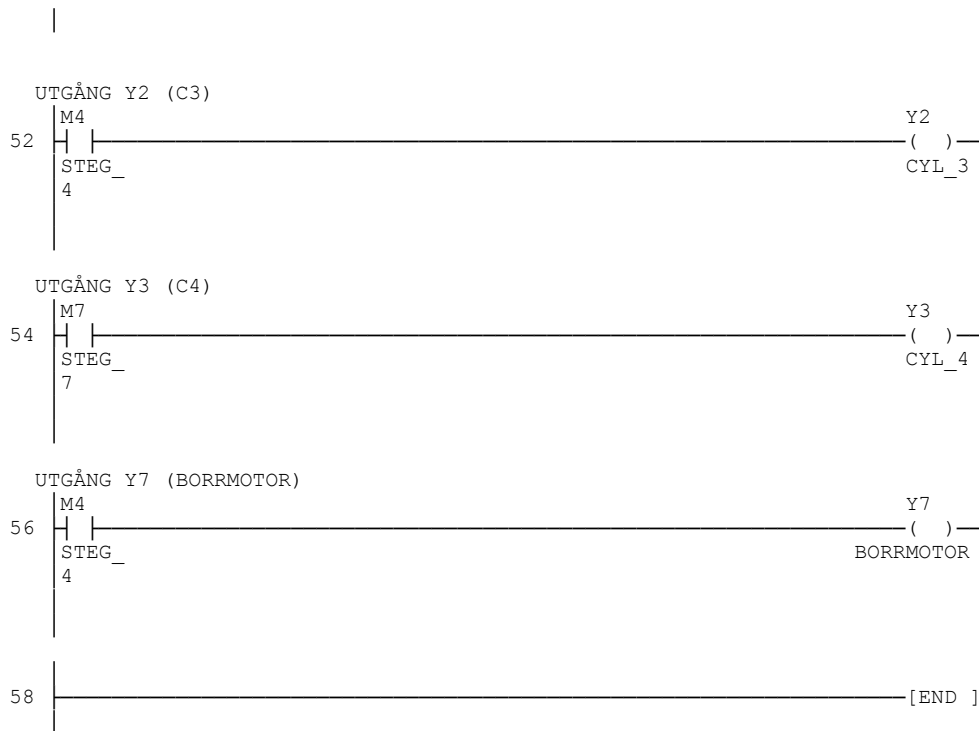
56 END

```

Programmet i ladderformat (Mitsubishi)





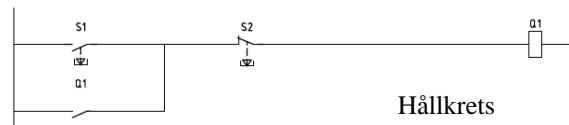


Stegövergången

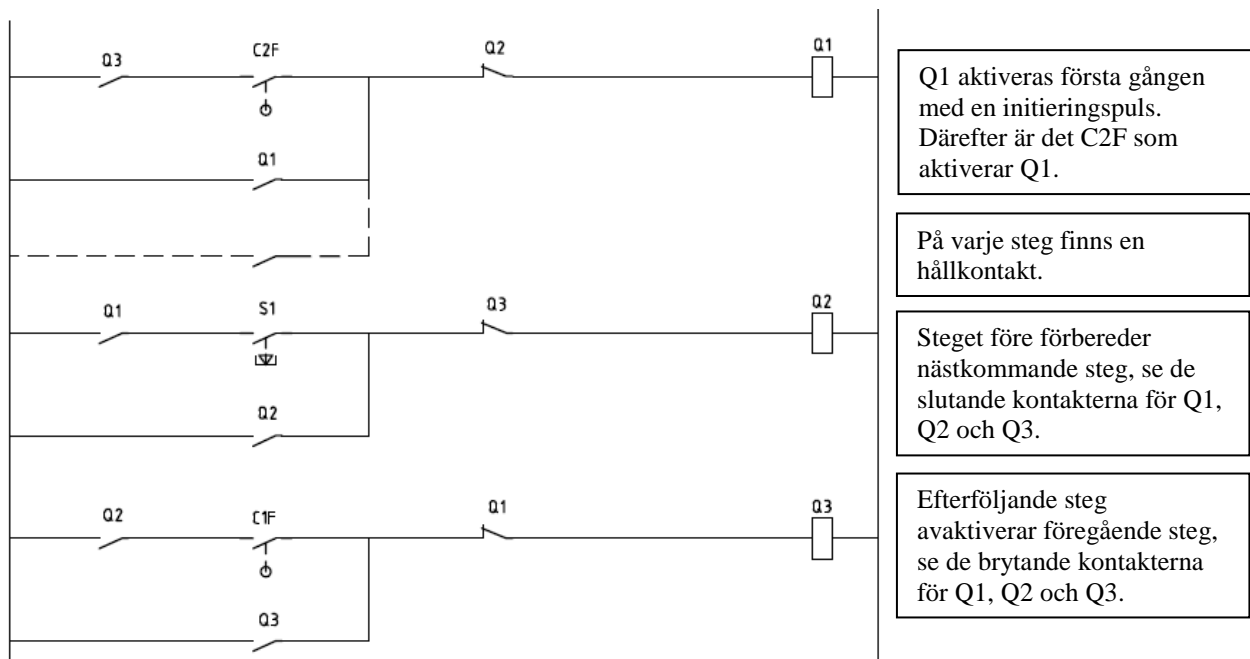
Varje steg bygger på en självhållningskrets eller set/reset funktion. Om tittar närmare på stegen kan man konstatera följande

- Föregående steg förbereder nästa steg
- Efterföljande steg slår ifrån (noll-ställer) föregående

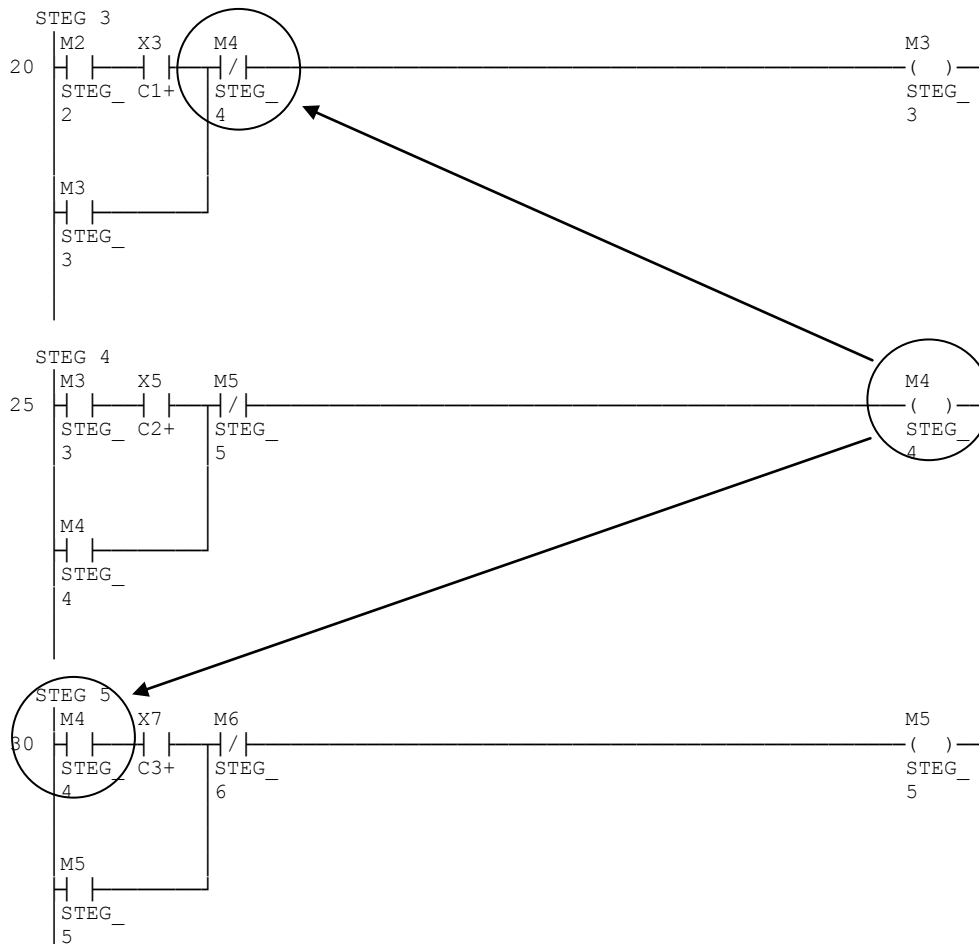
Varje steg arbetar med denna princip.



Exempel på en sekvens med 3 steg.



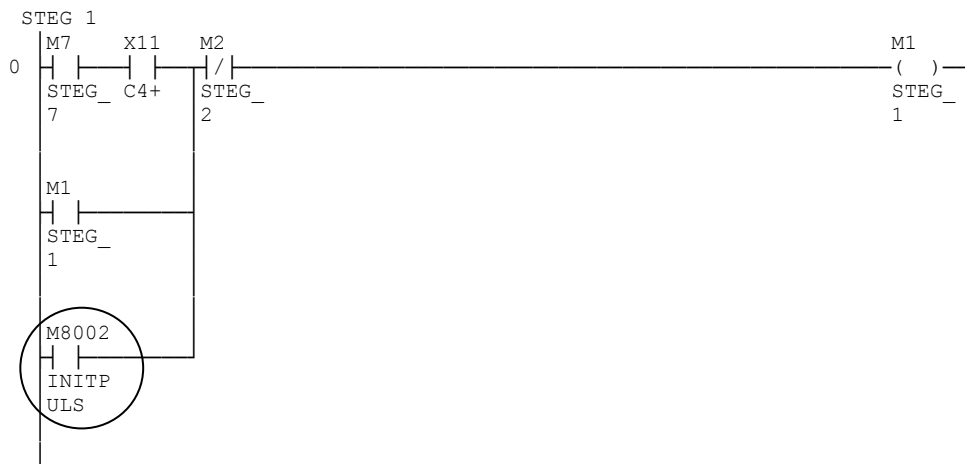
Om vi studerar 3 steg som ligger i följd kan man se detta. Det har ingen betydelse vilka 3 steg man studerar. Principen gäller för samtliga steg. Studera de inringade operanderna och hur de förhåller sig till steget före resp. steget efter.



Initieringssteget

Initieringssteget bör alltid vara det första steget i en sekvens. Initieringssteget har dubbla linjer runt sig. Initieringssteget måste 1-ställas vid start annars kommer sekvensen inte igång. Man skulle kunna säga att man skickar in en "1" i sekvensen som sedan skickas vidare till olika stegen i tur och ordning.

Initieringspulsen, som ska 1-ställa initieringssteget måste vara en signal som ger en "1" men bara vid uppstarten. I Mitsubishi's FX-system finns en operand som har just denna egenskap, den blir "1" precis i uppstartsskedet (någon msek) och blir sedan "0". Adressen är M8002 I första steget kan man se denna operand. Observera att operand M8002 (initieringspulsen) ligger parallellt med hållkontakten. M8002 är alltså endast aktiv i uppstartsskedet och går sedan låg.



Sekvensen

När väl initieringssteget är aktiverat är sekvensen igång och villkoren exekveras. När alla cylindrar står i sitt utgångsläge och det finns ämne i fallmagasinet är nästan alla villkor för steg 2 uppfyllda. Endast villkoret för tryckknappen saknas. Då tryckknappen påverkas är transition för steg 2 uppfyllt och steg 2 aktiveras. Samtidigt avaktiveras (noll-ställs) steg 1. Så arbetar sig sekvensen igenom varje steg för steg. När det sista steget är aktiverat går sekvensen tillbaka till det första steget d.v.s. initieringssteget. En ny tryckning på startknappen innebär en ny arbetscykel.

Sekvensen programmerad enligt IEC 61131-3

IEC 61131-3 beskriver 5 olika editorer eller språk för PLC-programmering och hur de ska tillämpas.

Regler för följande språk beskrivs i standarden

- SFC Sequential Function Chart
- LD Ladder Diagram
- IL Instuction List
- FBD Function Block Diagram
- ST Structured Text

När det gäller Grafcet-tekniken tillhör den SFC (Sequential Function Chart).

IEC 61131-3 tillämpar instruktionerna SET resp. RESET för de olika stegen. Här nedan visas motsvarande PLC-program i enlighet med standarden IEC 61131-3.

```

      STEG 1
0 LD      M2      STEG_2      32 LD      M4      STEG_4
1 RST     M1      STEG_1      33 AND     X7      C3+
                                     34 SET     M5      STEG_5

2 LD      M7      STEG_7
3 AND     X11     C4+
4 OR      M8002   INITPULS
5 SET     M1      STEG_1

      STEG 2
6 LD      M3      STEG_3
7 RST     M2      STEG_2

8 LD      M1      STEG_1
9 AND     X2      C1-
10 ANI    X3      C1+
11 AND     X4      C2-
12 ANI    X5      C2+
13 AND     X6      C3-
14 ANI    X7      C3+
15 AND     X10     C4-
16 ANI    X11     C4+
17 AND     X0      STARTKNAPP
18 AND     X1      ÄMNE_FALL_MAG
19 SET     M2      STEG_2

      STEG 3
20 LD     M4      STEG_4
21 RST    M3      STEG_3

22 LD     M2      STEG_2
23 AND    X3      C1+
24 SET    M3      STEG_3

      STEG 4
25 LD     M5      STEG_5
26 RST    M4      STEG_4

27 LD     M3      STEG_3
28 AND    X5      C2+
29 SET    M4      STEG_4

      STEG 5
30 LD     M6      STEG_6
31 RST    M5      STEG_5

      STEG 6
35 LD     M7      STEG_7
36 RST    M6      STEG_6

37 LD     M5      STEG_5
38 AND    X6      C3-
39 SET    M6      STEG_6

      STEG 7
40 LD     M1      STEG_1
41 RST    M7      STEG_7

42 LD     M6      STEG_6
43 AND    X4      C2-
44 SET    M7      STEG_7

      UTGÅNG Y0 (C1)
45 LD     M2      STEG_2
46 OR     M3      STEG_3
47 OUT    Y0      CYL_1

      UTGÅNG Y1 (C2)
48 LD     M3      STEG_3
49 OR     M4      STEG_4
50 OR     M5      STEG_5
51 OUT    Y1      CYL_2

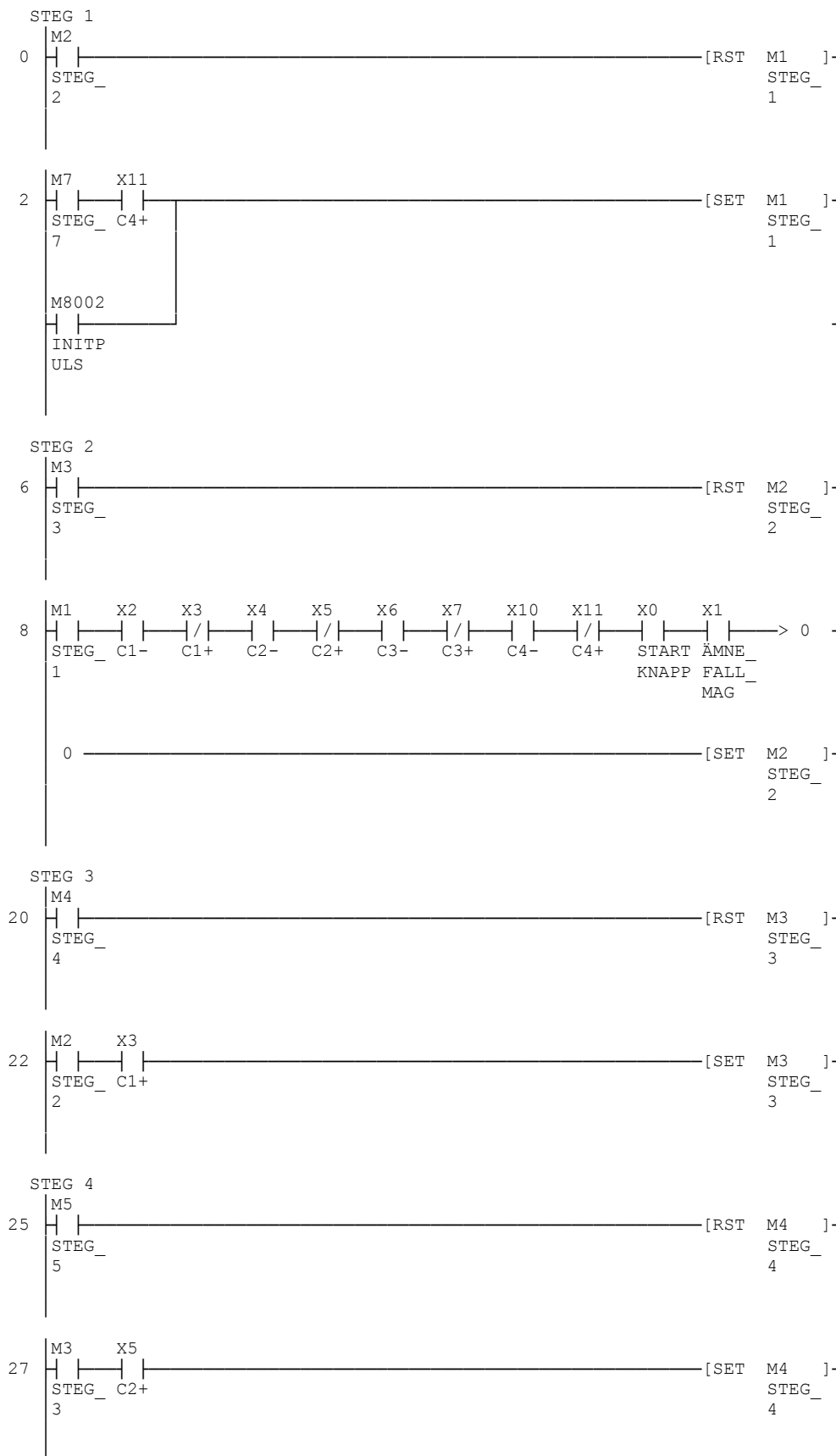
      UTGÅNG Y2 (C3)
52 LD     M4      STEG_4
53 OUT    Y2      CYL_3

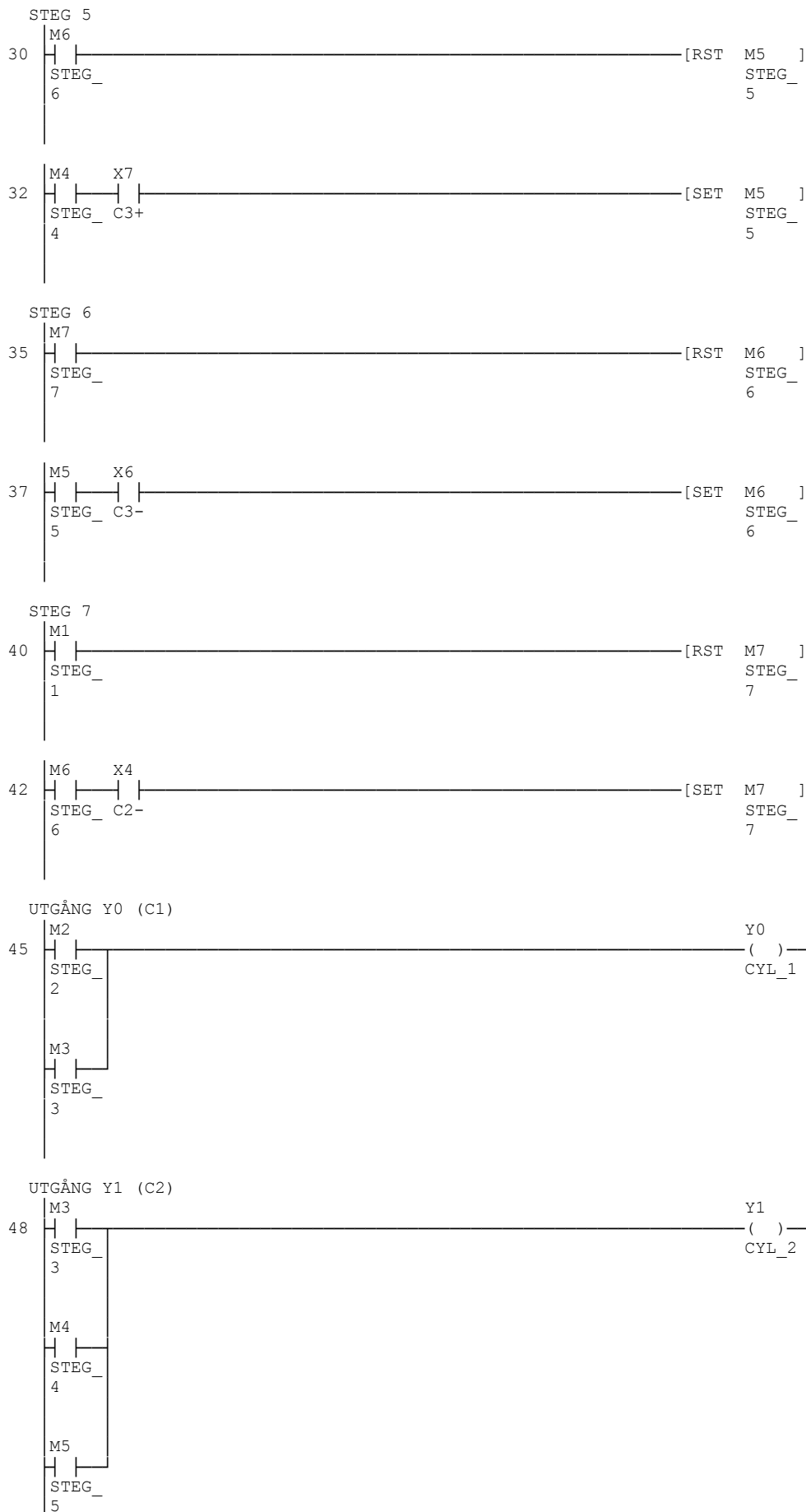
      UTGÅNG Y3 (C4)
54 LD     M7      STEG_7
55 OUT    Y3      CYL_4

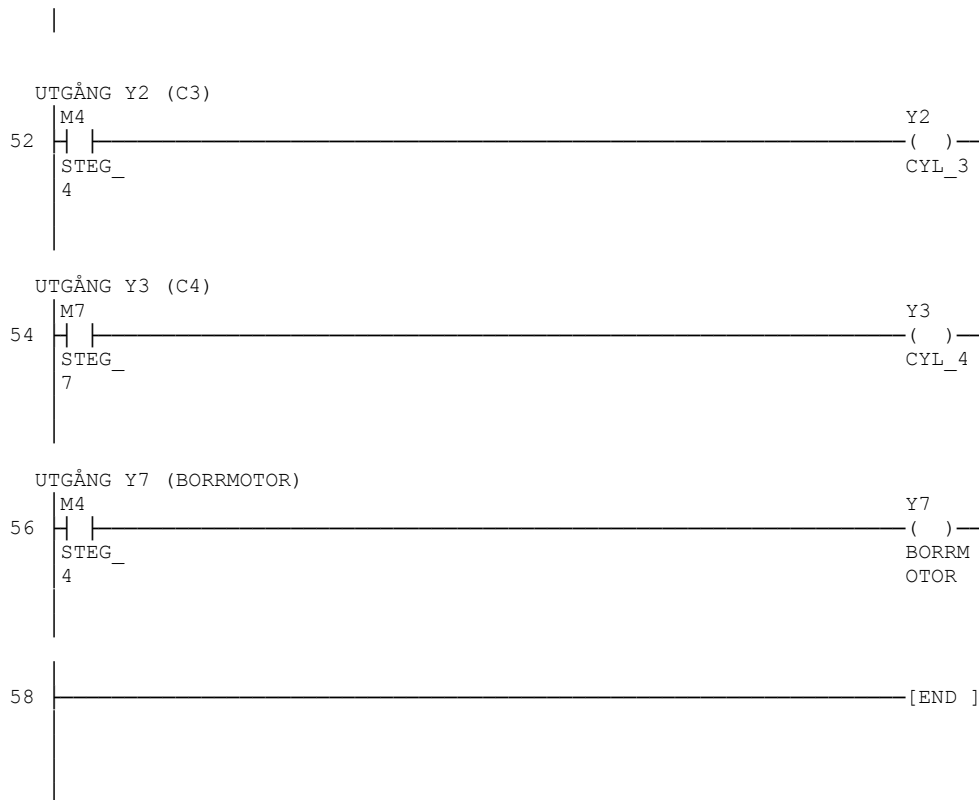
      UTGÅNG Y7 (BORRMOTOR)
56 LD     M4      STEG_4
57 OUT    Y7

58 END
```

Programmet i ladderformat (Mitsubishi)







Grafcet-tekniken går att tillämpa i en mängd olika fall och även i komplexa styrningar. Man kan t.ex. skapa sekvenser för

- Programval
- Parallella sekvenser
- Subrutiner
- etc.

Detta kompendie ska betraktas som en introduktion i Grafcet-tekniken.