

NORGE



**STYRET
FOR DET INDUSTRIELLE
RETTSVERN**

Utlegningsskrift nr. 118944

Int. Cl. G 06 f 11/04 Kl. 42m³-11/04

Patentsøknad nr. 165.572 Inngitt 14.XI 1966

Løpedag -

Søknaden alment tilgjengelig fra 1.VII 1968

Søknaden utlagt og utlegningsskrift utgitt 2.III 1970

Prioritet begjært fra: 16.XI-65 Sverige,
nr. 14.761/65

TELEFONAKTIEBOLAGET L M ERICSSON,
L M Ericssons väg 4 - 8, Fack, Stockholm 32, Sverige.

Oppfinner: Bengt Erik Ossfeldt, Ringdansvägen 5C,
Hägersten, Sverige.

Fullmektig: Ingeniør Fr. W. Münster.

Fremgangsmåte for å lokalisere en feil i et system som
består av minst to parallelt arbeidende datamaskiner.

Oppfinnelsen vedrører en fremgangsmåte for å lokalisere en feil i et system som består av minst to parallelt arbeidende datamaskiner, som er bygget opp av hver sin sentralenhet og av i respektive datamaskiner identiske funksjonsenheter med hver sin bestemte lagringsfunksjonstype og hvilke datamaskiner utfører samme regneoperasjon hver for seg og synkront for å muliggjøre at ved forekomst av en feil blir i det minste den ene datamaskins resultat korrekt.

Ved et datamaskinsystem av nevnte type, hvor forskjellige typer av lagringsfunksjoner utføres av fysisk adskilte funksjonsenheter, er det vesentlig hurtigst mulig å kunne bestemme ikke bare hvilken av datamaskinene som er feilaktig, men også hvilken av funksjonsenhetene som har forårsaket feilen i den feilaktige maskin for at denne del skal kunne bortkobles og de øvrige deler skal kunne fortsette det normale arbeide også i den tid, da den feilaktige enhet undersøkes og repareres resp. byttes ut.

For i en datamaskin av nevnte type ikke å risikere at en deloperasjon blir feilaktig utført innen man har rukket å tilveiebringe kontrollresultat er det nødvendig på bitnivå å foreta en stadig sammenligning av signalene som mates til og fra funksjonsenheter i de resp. systemer. Man bør nemlig få kjennskap til feil så snart denne har oppstått i en deloperasjon, og ved den minste avvikelse mellom to signaler i to parallelle enheter er det således ønskelig å avbryte det pågående arbeide slik at informasjon som er lagret i en av maskinene, ikke forstyrres. Forskjellen mellom en vanlig datamaskin og den nedenfor beskrevne er at man ved maskinen ifølge oppfinnelsen normalt ikke har noe overvåkningspersonale og heller ikke har tid eller mulighet til å avbryte pågående koblingsarbeide for å foreta reparasjonsforanstaltninger eller for å utføre feilsøkning. Først når det ved stadig sammenligning mellom de to systemers signaler på bitnivå er fastslått hvilken av maskinene som er feilaktig og denne umiddelbart frakobles, utføres sammenlignings- og feillokaliseringsarbeide, hvorfor den feilfrie datamaskin samt de feilfrie funksjonsenheter i den feilaktige maskin kan fortsette å arbeide sammen.

Ifølge oppfinnelsen tilveiebringes dette derved, at de to maskiners regneresultat stadig sammenholdes i en sammenligningskrets, som ved avvikelse ved hjelp av en kontrollkrets avbryter datamaskinens normale funksjon og setter igang en i begge maskiner identisk regneoperasjon, hvis resultatavvikelse fra en fastsatt verdi påvirker et frakoblingsorgan for respektive datamaskin, slik at den feilfrie datamaskin kan fortsette sitt normale arbeide, idet den feilfrie datamaskin i tur og orden innkobler en funksjonsenhet i den feilaktige datamaskin for samarbeide med den feilfrie maskin på en slik måte, at den utfører samme funksjon som tilsvarende funksjonsenhet i den feilfrie datamaskin, idet resultatene fra nevnte to funksjonsenheter stadig sammenlignes og ved avvikelse bortkobles den til den feilaktige datamaskin hørende funksjonsenhet, mens samtlige øvrige funksjonsenheter inklusive sentralenheten i den hittil feilaktige datamaskin igjen innkobles for parallellarbeide med den feilfrie datamaskin under felles anvendelse av den bare i den ene datamaskin feilfrie funksjonsenhet.

Oppfinnelsen skal nedenfor forklares nærmere under henvisning til et på tegningen vist utførelseseksempel.

Fig. 1 viser skjematisk et system bestående av to parallelt og synkront arbeidende datamaskiner, for hvilket system oppfinnelsens prinsipp kan tilpasses.

Fig. 2 viser kontrollkretsen ifølge fig. 1 mere detaljert i form av et logisk skjema.

Fig. 3 viser et blokkdiagram over en fiktiv datamaskin for lettere å kunne belyse oppfinnelsens prinsipp.

Fig. 4 viser skjematisk en logisk enhet i datamaskinen ifølge fig. 3.

Fig. 5 og fig. 6 viser skjematisk styreenhetens funksjon i datamaskinen ved utførelse av et normalt programavsnitt.

Fig. 7 og 8 viser skjematisk styreenhetens funksjon i datamaskinen ved utførelse av et avsnitt av et testprogram.

Fig. 9 viser skjematisk tidsforløpet for datamaskinens arbeide på forskjellige prioritetsnivåer.

Fig. 10 viser skjematisk datalagringsfeltet med et antall delfelt.

Fig. 11 viser overgangen fra et lavere til et høyere prioritetsnivå under datamaskinens normale arbeide, og

Fig. 12 viser funksjonen av datamaskinens styreenhet ved utførelse av et avsnitt i feillokaliseringsprogrammet.

Den etterfølgende beskrivelse inneholder mere enn det som har direkte forbindelse med oppfinnelsen, men den utførligere beskrivelse kan på den annen side være nødvendig for å kunne forklare oppfinnelsestanken. I beskrivelsen gis også noen eksempler på programmeringsavsnitt som har en viss tilknytning til oppfinnelsens tilpasning, men det må fremholdes, at oppfinnelsen ikke vedrører noe program i og for seg men den vedrører en fremgangsmåte for å lokalisere og etter lokaliseringen å fraskille feilaktige, fysiske funksjonsenheter i et datamaskinsystem bestående av to eller flere parallelt arbeidende datamaskiner.

For nærmere å kunne forklare oppfinnelsestanken beskrives først i korte trekk en fiktiv datamaskin, som er bygget opp på en slik måte, at den kan utføre de for styring av en automatisk telefonstasjon nødvendige operasjoner eller i prinsipp styringen av et vilkårlig system, som består av flere med hverandre samvirkende organ, f. eks. verktøysmaskiner. En datamaskin, som anvendes i praksis, har en mere komplisert organisasjon, som er valgt på en slik måte, at et optimum skal oppnåes hva angår antall anvendte organ (d. v. s. anleggsmkostningene) og antall trinn, som er nødvendige for å kunne utføre en operasjon. Den prinsipielle funksjon av datamaskinen er imidlertid uavhengig av nevnte optimum, og for å lette forklaringen er det gunstig om antall inngående organ i datamaskinen er minst mulig.

Datamaskinen kan deles opp i to hoveddeler, en lagringsdel MD omfattende et antall lagringsorgan og en sentral enhet CE omfattende et antall registre, en aritmetisk enhet og en styreenhet for mikroprogrammet (se fig. 3).

Lagringsdelen omfatter et instruksjonslagringsorgan IM (fig. 3), i hvilket instruksjonene, som skal utføres av datamaskinen, er lagret på hver sin bestemte adresse i form av f. eks. 16-sifrede binære ord. Disse instruksjoner avleses i tur og orden eller i annen ved programmet foreskrevet rekkefølge og hver instruksjon innebærer utførelse av et antall bestemte operasjoner, som tilhører denne instruksjon og er bestemt

av datamaskinens mikroprogram. Mikroprogrammet kan innebære avlesning og innskrivning av oppgaver i de forskjellige organer, overføring av oppgaver fra et organ til et annet, utførelse av logiske operasjoner i den aritmetiske enhet o. s. v. i den rekkefølge og i et sådant antall trinn som den spesielle instruksjon innebærer. Instruksjonslagringsdelen IM er forsynt med et adresseregister IA, i hvilket den adresse innskrives under hvilken den tilsiktede instruksjon befinner seg i instruksjonslagringsorganet, og med et instruksjonsregister IR, til hvilket en instruksjon, som er blitt utpekt ved hjelp av adresseregisteret IA, kan overføres fra instruksjonslagringsorganet IM for videreforsendelse til de øvrige organer. Alternativt kan en instruksjon innmattes fra et ytre organ til instruksjonsregisteret IR og samtidig mates den adresse til adresseregisteret IA, under hvilken instruksjonen skal plasseres i instruksjonslagringsorganet IM. Det sistnevnte forløp pågår som regel ikke ved maskinens normale forløp men bare ved endring av programmet, og under det normale forløp skjer bare avlesing. Muligheten for såvel innskriving som avlesing er symbolisert med S og L i blokkdiagrammet i fig. 3.

Med DM betegnes datalagringen, i hvilken noteringer av tilfeldige oppgaver skjer under maskinens arbeide i f. eks. 16-sifrede binære ord. Disse oppgaver kan vedrøre eksempelvis de forskjellige organers momentane tilstand i telefonnettet, lagring av siffersignaler o. s. v. Datalagringen DM har på samme måte som instruksjonslagringsorganet et adresseregister DA, i hvilket den adresse innskrives i datalagringen DM til resp. fra hvilken den tilsiktede informasjon skal innskrives resp. avleses. Datalagringen har også et resultatregister DR, til hvilket den tilsiktede informasjon innmattes for innskriving i den av adresseregisteret DA utpekte adresse eller til hvilket en ved adresseregisteret utpekt informasjon overføres for videreforsendelse til de øvrige organ avhengig av at skrive- eller leseinstruksjon fåes slik som antydnet med S resp. L. I den prinsipielle oppbygning er det således ingen forskjell mellom instruksjonslagringsorganet IM og datalagringen DM, men forskjellen ligger i deres anvendelsesmåte.

Med FE betegnes et ytterligere lagringsorgan, formidlingsenheten, hvis oppgave er å styre organ beliggende utenfor selve datamaskinen, f. eks. koblingsorgan i en automatisk telefonstasjon, resp. å avsøke tilstanden for disse organ. Med hensyn til forskjellen i funksjonshastighet for datamaskinen og de f. eks. elektromekaniske organ, som inngår i en telefonstasjon, er en lagringsfunksjon nødvendig for dels å lagre de fra datamaskinen i form av f. eks. 16-sifrede binære ord mottatte manøvreringsinstruksjoner inntil de relativt langsomme elektromekaniske organ har rukket å bli påvirket, dels å lagre de fra de elektromekaniske organ mottatte tilstandsinformasjoner inntil disse kan avsøkes av datamaskinen. I et binært ord, som innebærer man-

øvreringsinstruksjon, betyr "1" at i en utpekt 16-gruppe av organ skal de til respektive sifferposisjon hørende organ påvirkes, mens "0" betyr, at de til respektive sifferposisjon hørende organ ikke skal påvirkes. På tilsvarende måte betyr ved tilstandsavsøkning "1" at i en utpekt gruppe er de til respektive sifferposisjon hørende organ opptatte, og "0" betyr, at de er ledige. Som det fremgår er formidlingsenhetens FE funksjon sett fra datamaskinens synspunkt meget lik instruksjonslagringsorganets IM og datalagringens DM funksjon, da formidlingsenheten på samme måte som de sistnevnte i sitt adresseregister FA får en adresse fra sentralenheten og kan ved sitt resultatregister FR enten tilveiebringe en manøvrering av det ved resultatregisterets FR innhold bestemte organ i den ved adresseregisteret FA bestemte 16-gruppe av organ i telefonnettet TN eller kan alternativt i resultatregisteret FR innskrive tilstanden for de organ i telefonnettet TN, som inngår i en av adresseregisteret FA utpekt 16-gruppe. Nevnte to alternative muligheter er på samme måte som for IM og DM symbolsk antydnet med L resp. S.

Datamaskinens sentralenhet CE inneholder ifølge eksemplet tre registre RA, RB, RC (fig. 3), i hvert av hvilke et 16-sifret binært ord kan innføres, lagres der og mates ut derifra. En vesentlig del er logikkenheten LE, som kan utføre forskjellige aritmetiske operasjoner, f. eks. addisjon, subtraksjon, sammenligning, logisk eksklusiv-ELLER-funksjon o. s. v. Logikkenheten LE er forsynt med et innmatningsregister AA og et resultatregister AR for innskrivning av den ene eller annen av to operander, idet resultatet ved addisjon eller subtraksjon fåes i resultatregisteret ved at det i det sistnevnte innskrevne binære ord endres til regneresultatet. Ved logiske sammenligningsoperasjoner fåes utslag fra en indikator eller flaggvippe SEF, som ved overensstemmelse indikerer "0", mens den ved avvikelse indikerer "1". Det finnes dessuten et bitadresseregister LB, som i tilfelle ulikhet ved sammenligning mellom to 16-sifrede binære ord er inntruffet, angir sifferposisjonen for eksempelvis den laveste sifferposisjon, i hvilken ulikhet er blitt funnet.

En tredje vesentlig del av sentralenheten er styreenheten SE, som bestemmer oppgavenes forflytting mellom de forskjellige registre, med andre ord mikroprogrammet, som for hver instruksjon er koblingsmessig innebygget i styreenheten. Styreenheten har et ordregister OR, i hvilket en ordre innføres fra instruksjonsregisteret IR. Styreenheten avkoder det i ordregisteret innskrevne binære ord, i hvilket f. eks. 4 biter angir 16 mulige operasjoner, slik at en av 16 ledere blir strømførende (se fig. 5). Den på denne måte utvalgte leder bestemmer sammen med et antall ledere, som aktiveres i tur og orden etter hverandre, inn- og utmatningen av oppgavene til og fra registrene resp. de logiske operasjoner i logikkenheten, slik som det vil bli forklart i forbindelse med et eksempel. Samtlige registre kan tilkobles en felles 16-trådet ledning, som i fig. 3 er antydnet med en enkelt ledning, over OG-kretser OK1-OK22, hvis

andre inngangsvilkår bestemmes av styreenhetens SE utganger. Som ovenfor nevnt aktiveres de utvalgte utganger i tur og orden etter hverandre, slik at i tur og orden åpnes samtidig minst to OG-kretser for å muliggjøre dels utmatning av et 16-sifret binært ord til de felles lederne og dels innmatning av dette ord til ett av registrene, hvis inngangskrets er åpen. Som det er antydnet i fig. 3 har en del av registrene både inn- og utmatningsport, over hvilke både innmatning og utmatning fra registrene skal skje, mens en del av registrene er forsynt med bare innmatningsport fra den felles 16-leder, fordi deres innhold ikke mates direkte til de felles ledere. Styreenhetens SE og hele den fiktive datamaskins funksjon er lettest å forstå i forbindelse med noen elementære operasjoner.

Det antas at det skal adderes to 16-sifrede binære ord som allerede under et tidligere trinn av beregningen er blitt innført i registrene RA resp. RB. Det antas dessuten at det siste trinn i den foregående operasjon har vært at en instruksjon, som er avlest ved hjelp av en i adresseregisteret IA innskrevet adresse, fra IR er blitt overført til ordregisteret OR (se fig. 3). Ved overføringen av det nye instruksjonsord til ordregisteret forsvinner det tidligere innskrevne ord. Den nye ordre kan således formuleres på følgende måte: adder innholdet i RA og RB. Fig. 5 viser skjematisk styreenheten sammen med ordregisteret med nevnte ordre uttrykt i kodeform. De første 4 biter i tegnene angir 1 av 16 mulige operasjoner, av hvilke det antas at addisjon angis med koden 0001. Bitene 5-8 fra venstre angir koden 0001 for registeret RA, i hvilket den ene operanden befinner seg, og bitene 9-12 fra venstre angir koden 0010 for registeret RB, i hvilket den andre operanden befinner seg. Styreenheten er forsynt med avkodere AVK1, 2 og 3, hver med 4 innganger og 16 utganger, av hvilke sistnevnte en aktiveres for hver avkoder; utgangen nr. 1 for avkoderen AVK1 angir da at det dreier seg om registeret RA med kodennummer 1, og utgangen nr. 2 for avkoderen AVK3 angir da at det dreier seg om registeret RB med kodennummer 2. Med EK betegnes en trinnforflytningskjede med et antall utganger, som aktiveres den ene etter den annen i tur og orden og som sammen med utgang nr. 1 fra AVK1 i tur og orden aktiverer et antall OG-kretser K1a, K2a o. s. v. Disse OG-kretser bestemmer, sammen med eventuelt fra AVK2 og AVK3 mottatte signaler, hvilke av OG-kretsene OK1-OK22 som skal åpnes, for at den ønskete overføring av binære ord fra et register til et annet register skal skje, og hvilke aritmetiske operasjoner i LE som skal utføres. Første trinn er, at innholdet i registeret RA må overføres til registeret AR i logikkenheten LE, hvilket kan skrives: $(AR) = (RA)$. Som det er antydnet i fig. 5, danner dels utgangen h1a fra OG-kretsen K1a, dels utgangen v1a fra AVK2 (tilsvarende registeret RA) inngangene til en OG-krets Ka, som aktiveres og i sin tur aktiverer OG-portene OK6 og OK14. Disse porters aktivering er en betingelse for at innholdet fra RA skal overflyttes til AR. Samtidig berøres ikke de

øvrige registre, som det lett vil innsees med veiledning av blokkdiagrammet i fig. 3. Ved mottagning av neste signal fra trinnforflytningskjeden EK aktiveres lederen h2a, hvorved OG-kretsen Kb i krysningspunktet mellom lederne h2a og v2b (tilsvarende registeret RB) blir ledende og åpner OG-kretsene OK11, OK12. For at addisjon skal kunne utføres må nemlig innholdet fra RB overføres til AA i logikkenheten LE, da som tidligere forklart, de to operander må befinne seg i registeret AA resp. AR for at man skal kunne utføre en regneoperasjon. Nevnte overføring kan uttrykkes som $(AA) = (RB)$, hvilket oppnåes ved OG-kretsene OK11 og OK12 aktivering, som det fremgår av fig. 3. Neste trinn blir selve addisjonen, og den styres av kjedens EK tredje trinn på en slik måte, at OG-kretsen K3a over lederen h3a aktiverer en inngang In2 for logikkenheten LE, hvilken inngang styrer adderingsfunksjonen.

Logikkenheten, som i og for seg er alment kjent, vises i fig. 4 i form av et blokkdiagram. De to operander mates i form av 16-sifrede binære ord til registrene AA og AR, som adskiller seg derved, at i det førstnevnte hvert nytt innmatet ord stryker det der allerede forekommende ord, mens i det sistnevnte hvert nytt innmatet ord adderes til det allerede i registeret forekommende. I AR kan både inn- og utmatning av det i registeret forekommende ord finne sted, mens registeret AA bare er beregnet på innmatning fra og ikke utmatning til andre organ. Slik som det er antydning symbolsk i fig. 4, kan man ved å påvirke de forskjellige innganger In1, In1, In3 o.s.v. med signaler fra styreenheten SE styre forskjellige operasjoner. Ved f.eks. å aktivere inngangen In1 adderes +1 til det i registeret AA innskrevne binære ord, og resultatet skrives i AR. Ved å aktivere inngangen In2 adderes innholdet i AA og AR, og resultatet skrives i AR. Hvis man aktiverer inngangen In3, skjer en logisk sammenligning mellom innholdet i AA og AR, og hvis noen forskjell i en eller annen av de binære tegnelementer foreligger, indikeres "1" av en flaggvippe SEF, og ved likhet mellom samtlige tegnelementer indikeres "0" av flaggvippen. Adressen til tegnelementet med f.eks. laveste siffervekt, som avvikelse, indikeres samtidig av et bitadresseregister LB, som har 4 sifferposisjoner og følgelig kan angi en av 16 sifferplasser.

Som ovenfor nevnt, aktiverer lederen h3a inngangen In2 til logikkenheten LE, slik at addisjonsresultatet i registeret AR fåes som tredje trinn under kjedens EK trinnforflytning. Addisjonsresultatet må nu overføres fra AR til et annet register for lagring, f.eks. til RA, hvilket kan uttrykkes: $(RA) = (AR)$. For at dette skal kunne skje, må portene OK15, OK5 åpnes, hvilket skjer under trinnforflytningskjedens neste trinn ved hjelp av lederen h4a. Det forutsettes ifølge eksemplet, at adressen til den nettopp utførte instruksjon under utførelsen har vært lagret i registeret RC, hvor den er blitt lagret samtidig som den er blitt anvendt for å utpeke instruksjonen i instruksjonslagringsorganet (sammenlign utgangsstillingen for den nu beskrevne prosess). En registrering av den nettopp under utførelse værende instruksjons adresse er nød-

vendig for at man skal kunne bestemme den adresse, ved hvilken neste instruksjon skal hentes fra instruksjonslagringsorganet IM. Som kjent kan instruksjonene utleses i nummerorden etter hverandre, d. v. s. etter en viss instruksjon med en bestemt adresse n utleses neste instruksjon med adressen $n+1$. Som et annet alternativ kan maskinen utføre hopp, idet adressen til neste instruksjon bestemmes av beregningsresultatet, som fåes på grunn av den nettopp utførte instruksjon. Ifølge det behandlede eksempel antas det at noe hopp ikke er nødvendig idet neste instruksjon har i rekkefølge derefter følgende adresse. Neste trinn blir følgelig at instruksjonsadressen hentes i registeret RC og økes med 1 i logikkenheten LE. Selve hentingen fra RC og overføringen til AA kan symboliseres som $(AA) = (RC)$ og finner sted ved at signalet over tråden h5a i styreenheten SE åpner portene OK21, OK12. Ved å aktivere inngangen In1, som forklart i forbindelse med fig. 4, hvilket skjer ved at tråden h6a aktiveres, adderes 1 til det i AA innskrevne binære ord og resultatet som utgjør den nye instruksjonsadresse, fåes i AR. Den nye instruksjonsadresse må nu dels lagres for å bli stående til instruksjonen er blitt utført og for å danne basis for bestemmelse av neste instruksjonsadresse, dels innskrives i instruksjonslagringsorganets IM adresseregister IA for å finne ut neste instruksjon. Lagringen av instruksjonsadressen finner igjen sted i RC, hvilket kan skrives $(RC) = (AR)$, ved at tråden h7a aktiverer portene OK15, OK20. Overføringen av instruksjonsadressen fra RC til instruksjonslagringsorganets adresseregister, hvilket kan skrives $(IA) = (RC)$, skjer ved at tråden h8a aktiverer portene OK21, OK7. Også leseinstruksjon sendes til instruksjonslagringsorganet ved at inngangen L aktiveres, slik at instruksjonen utleses og overføres til instruksjonslagringsorganets resultatregister IR. Neste trinn blir overføringen av denne instruksjon til ordregisteret OR: $(OR) = (IR)$ ved å aktivere portene OK2, OK16 ved hjelp av lederen h9a. Den tidligere innskrevne instruksjon raderes derved, trinnforflytningskjeden EK stilles tilbake til sitt begynnelsespunkt og man påbegynner et nytt forløp, som avhengig av den innskrevne instruksjon selvsagt kan være av helt annen karakter, f. eks. foreskrive en subtraksjon istedenfor den nettopp utførte addisjon.

Som tilknytning til det ovenfor beskrevne kan man gi et eksempel på hvorledes datamaskinen løser en i forbindelse med styringen av en automatisk telefonstasjon forekommende oppgave. En av de mange oppgaver, som forekommer, er å fastslå om en eller annen forandring i tilstanden for en bestemt abonnentledning har inntruffet, d. v. s. at en abonnent, som ved den siste avspøking av tilstanden har vært ledig, er blitt opptatt eller er blitt ledig etter tidligere å ha vært opptatt. Som ovenfor anført i forbindelse med formidlingsenheten FE, skjer avspøkingen av abonnentledningenes tilstand i grupper på 16 abonnenter, hvorved man får et 16-sifret binært ord i hvilket til hver av de 16 abonnenter i gruppen hører et binært siffer "0" eller "1" avhengig av deres

ledige resp. opptatte tilstand. Utpekningen av en 16-gruppe skjer ved hjelp av en i formidlingsenhetens FE adresseregister FA innskrevet adresse. Avsøkningen av abonnentledningstilstanden skjer syklisk med f.eks. 300 ms intervaller og resultatet noteres i datalagringen DM under den til resp. 16-gruppe hørende adresse, slik at det i datalagringen alltid er lagret noteringen vedrørende den siste avsøkning. For å kunne fastslå om noen forandring i ledningstilstanden er inntruffet, må man utføre en sammenligning mellom det i formidlingsenheten ved hjelp av en bestemt adresse oppnådde resultat og den i datalagringen under samme adresse forekommende notering, hvilket kan uttrykkes: sammenlign innholdet i FR med innholdet i DR. For enkelhets skyld antas, at den nu beskrevne operasjon direkte slutter seg til den ovenfor beskrevne addisjon, d.v.s. instruksjonen, som under det siste trinn er blitt overført til ordregisteret OR, foreskriver sammenligning mellom innholdet i FE og DM under en i instruksjonsordet angitt adresse. Fig. 6 viser ordregisteret OR og styreenheten SE på tilsvarende måte som fig. 5. Ordregisteret er antydnet med det innskrevne instruksjonsord, i hvilket tegnelementene 1-4 fra venstre angir 1 av 16 mulige operasjoner, ifølge eksemplet en sammenligningsoperasjon med kodenummer 2, sifferposisjonene 5-8 angir det organ, i hvilket den ene operand befinner seg, ifølge eksemplet registeret DR med kodenummer 9, sifferposisjonene 9-12 angir det organ, i hvilket den andre operand befinner seg, ifølge eksemplet FR med kodenummer 11, og sifferposisjonene 13-16 angir adressen til den 16-gruppe av abonnenter, som skal undersøkes og som ifølge eksemplet er antatt å ha adressen 7. Instruksjonsordet har således formen 0010100110110111.

I likhet med hva som er beskrevet i det foregående eksempel, åpner kjedens EK trinnforflytning de forskjellige porter i tur og orden med den forskjell at idette tilfelle blir det utgang nr. 2 fra dekoderen AVK1 som aktiverer de forskjellige OG-kretser K1b, K2b o. s. v. i tur og orden i takt med kjedens trinnforflytning. Første trinn blir at instruksjonsordet fra ordregisteret overføres til såvel DA som FA. Dette skjer for at tegnelementene 13-16 fra venstre skal kunne anvendes som adresse for den abonnentgruppe som skal undersøkes. Portene OK22, OK13, OK19 åpnes og samtidig skjer avlesning i såvel FE som DM, slik at tilstandsnoteringen vedrørende den tilsiktede abonnentgruppe med adresse 7 (111) skrives inn såvel i FR som i DR. Dette utføres ved aktivering av lederen h1b i fig. 6. Neste trinn blir at innholdet i DR må overføres til AA ved åpning av portene OK9, OK12. Dette skjer ved at en OG-krets Ka, hvis aktivering er avhengig av at et signal fåes såvel fra lederen h2b for trinnforflytningskjeden EK som fra lederen v9a for dekoderen AVK2 (tilsvarende kodenummeret 9 for DR). Derefter overføres innholdet i FR til AR, for hvilket formål portene OK18, OK14 må åpnes, og dette skjer ved en OG-krets Kb, som aktiveres avhengig av samtidig forekommende signaler på lederen h3b og lederen v11b for dekoderen

AVK3 (tilsvarende kodennummer 11 for FR). Neste trinn blir sammenligning mellom innholdet i AA og AR, hvilket skjer ved aktivering av inngangen In3 for LE. Er en forskjell blitt påtruffet i en eller annen av sifferposisjonene, indikeres dette ved at flaggvippen SEF inntar "1"-posisjon, er der ingen forskjell, befinner flaggvippen seg i "0"-posisjon. Ved neste trinn, når tråden h5b aktiveres, er der to alternativ, og neste instruksjon kan hentes på to forskjellige steder: er forskjell påtruffet, d. v. s. sammenligningsresultatet fra SEF er "1", blir adressen til neste instruksjon den i nummerorden følgende. Adressen til den nettopp utførte instruksjon finnes i registeret RC, og denne adresse må økes med 1. Først åpnes portene OK21, OK12 for at innholdet i RC skal overføres til AA, hvilket skjer ved aktivering av lederen h6b. Derefter aktiveres lederen h7b for å aktivere inngangen In1 til logikkenheten og addere 1 til innholdet i AA, og resultatet lagres i AR. Fra AR overføres den nye adresse til RC for lagring ved å aktivere lederen h8b og åpne portene OK15, OK20. Fra RC overføres den nye instruksjonsadresse til IA ved at portene OK21 og OK7 åpnes og utlesning skjer, hvorved instruksjonen med den angitte adresse overføres til IR. Deretter overføres instruksjonen til OR ved at portene OK2, OK16 åpnes. Denne nye instruksjon er en følge av at forskjell ved sammenligningen er blitt påtruffet og kan f. eks. innebære, at et oppkoblingsforløp til respektive abonnentutstyr skal påbegynnes. Hvis på den annen side ingen forskjell ved sammenligningen er blitt påtruffet og SEF indikerer 0, må en annen instruksjon utføres som innebærer at f. eks. den i nummerorden etterfølgende 16-gruppe av abonnenter undersøkes.

Adressen til den abonnentgruppe som er blitt undersøkt, ble funnet notert i sifferposisjonene 13-16 fra venstre. Denne adresse må nu økes med 1. På grunn av at SEF indikerer 0, aktiveres en strømkrets, som på samme måte som beskrevet i forbindelse med det første alternativet sammen med trinnforflytningskjedens utganger aktiverer OG-kretsene K6c, K7c o. s. v., hvorved signaler fåes efter hverandre på utgangene h6c, h7c o. s. v. For at den nye adresse skal kunne beregnes, må først ordregisterets OR innhold overføres til AA, hvorfor aktiveringen av tråden h6c åpner portene OK22, OK12. Neste trinn blir at i LE adderes 1 til det i AA forekommende ordets adressedel, d. v. s. tegnposisjonene 13-16, hvilket skjer ved aktiveringen av en spesiell inngang In5 i logikkenheten. Av hensyn til at det er samme instruksjon som fortsetter med den forskjellen, at den i instruksjonsordet forekommende adresse er blitt øket med 1, behøver man ikke benytte seg av instruksjonslagringsorganet IM, men resultatregisterets AR innhold kan direkte overføres til ordregisteret OR, hvilket skjer ved at lederen h8c åpner portene 15, 16. Derved er samme stilling inntruffet som da den forrige instruksjon ble påbegynt, trinnforflytningskjeden 0-stilles og dens utganger begynner igjen i tur og orden å aktivere lederne h1b, h2b o. s. v. inntil likhet resp. ulikhet mellom den ved adressen utpekte abonnentgruppes

momentane og tidligere registrerte tilstand er blitt påtruffet.

Når styringen av telefonanlegget skjer ved hjelp av en datamaskin, avses tilstanden for samtlige i anlegget inngående organ regelmessig med en viss periodisitet og avspøringsperiodene må være så korte, at man med sikkerhet avspører hver forandring i tilstanden, d.v.s. at ikke noen forandring går tapt. Når det gjelder f. eks. pulsmottagende reléer, hvis kontakter nøyaktig følger forandringene i de innkommende signaler, må en avspøringsperiode være relativt kort, f. eks. 10 ms. Det finnes på den annen side slike organ, som ikke behøver å avspøreres resp. manøvreres i så rask takt. Ved f. eks. oppkobling av velgere kan en funksjonsperiode av 100 ms være tilstrekkelig og ved f. eks. avspørning av abonnentledningene for å avspør tilstanden kan en periode av 300 ms være fullt tilfredsstillende. Disse verdier tas i betraktning ved programmeringen av datamaskinen, som kan arbeide i f. eks. 3 adskilte prioritetsnivåer, hvilket innebærer, at funksjoner med kort avspøringstid, d.v.s. høyere prioritet, utføres først, mens funksjoner med lavere prioritet får vente inntil samtlige funksjoner, som har høyere prioritetsnivå, er blitt utført.

Fig. 9 viser skjematisk et tidsforløp for de 3 nivåer ved et helt vilkårlig valgt eksempel. Som det fremgår er tidsaksen oppdelt i 10 ms intervaller. Det høyeste prioritetsnivå A, i hvilket f. eks. test og styring av signalmottagende og signalstyrende organ skjer, påbegynnes uvilkårlig hver 10. ms, hvilket innebærer, at hvis den til nivå A hørende funksjon ikke er blitt fullført i løpet av en 10 ms-periode, fortsetter bearbeidelsen under den eller de neste 10 ms-perioder, mens funksjonene med prioritetsnivå B og C får vente. Hver gang arbeidet i A-nivået er blitt avsluttet før 10 ms-periodens slutt, påbegynnes prioritetsnivået B, som hører til mindre viktige funksjoner enn A-nivået, f. eks. oppkobling og nedkobling av velgere eller til- og frslag av mindre viktige reléer. På samme måte må prioritetsnivået C vente inntil arbeidet på prioritetsnivået B helt er avsluttet. De tidspunkt, i hvilke B- og C-nivåenes arbeide påbegynnes, bestemmes dels av programmet og dels av trafikken i telefonstasjonen, som kan medføre en forskyvning av dette tidspunkt.

Av det ovenfor anførte fremgår at arbeidet på B- og C-nivåene må avbrytes, når arbeidet på A-nivået påbegynnes hvert 10. ms. For at informasjon vedrørende arbeidet, som pågår i nettopp det øyeblikk, da B- eller C-nivået avbrytes, ikke skal gå tapt, må informasjonen ved avbruddstidspunktet lagres under i for dette formål reserverte delfelt i datalagringen DM (skjematisk antydnet i fig. 10) for at den igjen skal kunne anvendes, når maskinen igjen kan arbeide på resp. nivå. Lagringen innebærer, at samtlige registre i sentralenheten må tømmes og deres innhold må lagres på en slik måte, at det ved anvendelse umiddelbart kan tilbakeføres til riktig plass i sentralenheten CE. Det er således innholdet i registrene RA, RB, RC, SEF og LB, som må lagres i delfeltet f. eks. nivå B hvis det var dette nivå som ble avbrutt. Overføringen

skjer på samme måte som forklart i forbindelse med operasjonene ovenfor, ved at man åpner en inngangs- resp. en utgangsport for de organ, mellom hvilke overflytting av informasjonen skal utføres i tur og orden. Dette skjer på samme måte som ved en vanlig operasjon ved hjelp av styreenheten SE og vil senere bli ytterligere belyst i forbindelse med en feilttest, hvor også A-nivået må lagres, men forløpet er det samme som ved lagringen av B- eller C-nivået.

På A-nivået utføres arbeider, som har høyeste prioritet, f. eks. signalmottagning og signalsending, slik som tidligere nevnt. Disse arbeider består av et antall delfunksjoner, som hver for seg lagres i et delfelt. Det finnes f. eks. delfelt for innregning av mottatte pulser, for tidsovervåkning under pågående sending vedrørende pulsens og pausens lengde, tilsvarende funksjoner ved mottagning o. s. v., slik det er skjematisk antydnet i fig. 10. Alle disse delfelt tilhører A-nivået, og så lenge de inneholder informasjon, betyr dette at det finnes arbeide på A-nivået. Dette indikeres ved at en bestemt sifferposisjon, en såkalt arbeidsbit, er 1-stilt i hvert av disse delfelt. På tilsvarende måte tilhører bestemte delfelt B-nivået, f. eks. informasjon angående velgertilstand, velgerinnstilling o. s. v., og C-nivået, f. eks. informasjon angående abonnentledningstilstand. Som symbolsk antydnet i fig. 10, er en vippe VA resp. VB 1-stilt så lenge arbeidsbiten i en eller annen av delfeltene for A-nivået resp. B-nivået er "1". Dette er i fig. 10 symbolisert med ELLER-kretser EA og EB, men kan i praksis utføres ved at datamaskinen avleser arbeidsbiten i hvert delfelt i tur og orden. For C-nivået behøves ikke noen lignende indikering, da maskinen går inn på C-nivået bare når det ikke finnes arbeide på A- eller B-nivået.

Maskinen undersøker først A-nivåets indikeringsvippe VA, og så snart denne er blitt 0-stilt, d. v. s. når arbeidet på A-nivået er blitt fullført resp. når ingen siffer-sending eller siffermottagning pågår, d. v. s. under lavtrafikk, påbegynner maskinen arbeidet på B-nivået. Når arbeidet på dette nivå er avsluttet resp. ved lavtrafikk, da ikke noen kobling skal oppstilles resp. nedtas, blir indikeringsvippen for B-nivået 0-stilt og hopp til C-nivået finner sted. På dette nivå skjer arbeide stadig også under lavtrafikk, da f. eks. gjennomspøkning av ledningstilstanden må pågå kontinuerlig selv om på grunn av lavtrafikk ikke noe arbeide utføres på A- eller B-nivået.

Som nevnt påbegynnes A-nivåets arbeide hvert 10. ms og for dette formål må maskinen hvert 10. ms undersøke A-nivåets indikeringsvippe VA. Viser denne at noteringer finnes i de til A-nivået hørende delfelt, må arbeidet på A-nivået pågå inn-til samtlige oppgaver i nevnte delfelt er blitt bearbeidet. Skulle dette arbeide ikke være avsluttet innen 10 ms, innebærer dette at maskinens belastning er for stor. På samme måte gjøres alt arbeide på B-nivået, innen hopp finner sted til C-nivået. Som tidligere nevnt må oppgavene angående B-nivået resp. C-nivået lagres i for dette for-

mål i datalagringen reserverte delfelt, hvis man ikke har rukket å utføre arbeidet innen neste 10 ms-periode påbegynnes, da man uvilkårlig må hoppe inn på A-nivået.

Dette forløp er skjematisk antydnet i fig. 11. Med KL betegnes en tidspulsgenerator, som frembringer tids-pulser med 10 ms mellomrom, og med OKA betegnes en OG-krets, hvis ene inngangssignal dannes av nevnte tids-pulser, og det andre inngangssignal fåes fra indikeringsvippens VA "1"-stilling. Utgangssignalet opptrer fra denne krets hvis vippen VA er "1"-stilt, og dette utgangssignal mates til to OG-kretser OKB og OKC, hvis andre inngangssignal fåes fra en indikeringskjede IND. Denne har tre trinn A, B, C som aktiveres i overensstemmelse med de 3 prioritetsnivå. Kjeden beveges frem til neste trinn av samme puls som tilveiebringer påbegynnelsen av et nytt prioritetsnivå (fig. 9), slik at alltid det trinn i kjeden er aktivert, som tilsvarende det nettopp pågående prioritetsnivå. Da utgangen fra trinn B resp. C danner den andre inngang til OG-kretsene OKB og OKC, fåes ved opptreden av tids-pulsen KL et utgangssignal bare fra den av OG-kretsene OKB resp. OKC, på hvis prioritetsnivå maskinen nettopp arbeider. Dette utgangssignal skal tilveiebringe lagringsinstruksjonen for de i sentralenheten CÉ forekommende oppgaver. Innen lagringen påbegynnes, må en ytterligere betingelse oppfylles, nemlig at instruksjonen, som nettopp holder på å utføres, er blitt avsluttet. Dette er nødvendig for at ved igjenopptagelse av programavsnittet samtlige oppgaver skal stå til disposisjon. Når kjeden EK (fig. 6 eller 7) har utført det siste trinn og tilbakestilles til 0-stilling, fås et signal på en utgang fra styreenheten som tegn på at den i ordregisteret OR innskrevne instruksjon er blitt utført. Dette signal aktiverer sammen med det fra kretsene OKB og OKC mottatte signal en OG-krets OKE resp. OKF, hvis utgang utpeker en bestemt instruksjonsadresse i instruksjonslagringsorganets IM adresseregister IA. Derved avleses den under nevnte adresse i instruksjonslagringsorganet forekommende instruksjon og innskrives i instruksjonsorganets resultatregister IR. Dette instruksjonsord inneholder de instruksjoner, som er nødvendige for å styre OG-kretsene OK1 - OK22 på en slik måte, at samtlige oppgaver angående det nettopp under bearbeidelse stående B- resp. C-nivåavsnitt overføres til datalagringen for dette formål reserverte felt. Dette skjer ved at instruksjonsordet innskrives i ordregisteret OR og styreenheten beveges frem trinnvis ved hjelp av trinnkjeden EK for i tur og orden å åpne de respektive porter, som beskrevet i forbindelse med fig. 5 og 6. Et slikt lagringsforløp vil bli beskrevet nærmere i forbindelse med lagringen av A-nivåets oppgaver ved feilsøking.

Når oppgavene vedrørende det nettopp pågående B- eller C-program er blitt overført til datalagringen, utpekes neste instruksjonsadresse i instruksjonslagringsorganets adresseregister IA. Denne adresse bestemmer neste instruksjon, som skal overføres til ordregisteret OR, og hvilken instruksjon innebærer at de opp-

gaver, som under de 10 siste ms er blitt samlet i de til A-nivået hørende delfelt, skal overføres til sentralenheten og der bearbeides på den ifølge programmet bestemte måte. Når samtlige oppgaver i A-nivåets delfelt er blitt bearbeidet, og disse felt er blitt tømt, blir A-nivåets indikeringsvippe 0, hvorved en bestemt adresse i instruksjonslagringsorganets adresseregister utpekes. Denne adresse angir en instruksjon for tilbakeføring til sentralenheten av de oppgaver som er lagret i B-nivåets delfelt. Skulle det vise seg at indikeringsbiten for B-nivåets delfelt er 0, - d. v. s. ved påbegynnelsen av A-nivået er ikke noen funksjon blitt avbrutt på B-nivået, og følgelig har ikke noen oppgaver for B-nivået behovt å lagres, - utpekes en instruksjonsadresse for et nytt instruksjonsord. Dette innebærer overflytting av de oppgaver som er lagret i C-nivåets delfelt i datalagringen, til sentralenheten CE og utførelse av de for C-nivået ved programmet bestemte forløp. Neste tids-signal etter 10 ms-periodens slutt bryter C-nivåets arbeide og tilveiebringer lagring av de til C-nivået hørende og i sentralenheten CE forekommende oppgaver i de for dette formål beregnete delfelt i datalagringen DM.

Fig. 1 viser et blokkdiagram over et system bestående av to datamaskiner A og B som er bygget opp av i de to datamaskiner identiske funksjonsenheter bestående av sentralenhet CE, instruksjonslagringsorgan IM, datalagring DM og formidlingsenhet FE. Datamaskinene A og B arbeider synkront med hverandre og utfører samme arbeidsoppgave for ved sammenligning mellom resultatene å muliggjøre en kontroll av at resultatet har vært korrekt. For dette formål er det anordnet en sammenligningskrets JK, som stadig sammenligner de i de to maskiner pågående forløp og som ved den minste avvikelse avgir et signal til en kontrollkrets KK. Som det fremgår av den tidligere beskrivelse av en datamaskin, står sentralenheten CE og lagringsorganene IM, DM, EF i forbindelse med hverandre over en 16-trådet forbindelsesledning f1a resp. f1b, over hvilken hver informasjon føres, som mates ut fra et organ og mates inn til et annet organ. Disse 16-trådede ledninger for resp. datamaskin er ført til sammenligningskretsen JK, som gir et utslag så snart en avvikelse oppstår i de to 16-sifrede binære ord, som mates over de to maskiners ledninger. Kontrollkretsens KK ene oppgave er å avspøke de signaler fra funksjonsenhetene, som oppstår på grunn av visse definerbare feil på grunn av f. eks. kortslutningsalarm, paritetskontrollsignal o. l. Det kan imidlertid oppstå feil som ikke medfører noe bestemt feilsignal, og kontrollkretsens KK oppgave er ved den minste forskjell mellom de to datamaskiners forløp, hvilken forskjell angis ved et signal fra sammenligningskretsen JK, å sette igang et feilsøkningsprogram for å lokalisere feilen. Ved at datamaskinen er delt opp i nevnte funksjonsenheter, som har hver sin bestemte funksjon og i størrelse ikke avviker altfor meget fra hverandre, lettes feillokaliseringen.

Som antydnet i blokkdiagrammet i fig. 1 kan hvilken som helst av funksjonsenheterne i begge datamaskiner frakobles og kobles til den andre datamaskin. Dette er symbolisert med reléene R1a, R2a, R2b o. s. v., men skjer fortrinnsvis ved elektroniske organ som styres av kontrollkretsen KK.

Ifølge oppfinnelsens grunntanke skal kontrollkretsen KK passe på ved tilfeldig mottagelse av et signal fra JK, f. eks. i en tid av høyst 10 ms. at begge datamaskiners normale program opphører og at begge datamaskiner hver for seg utfører samme bestemte regneoperasjon. Den av datamaskinene, hvis resultat ikke stemmer overens med det på forhånd fastslåtte resultat, er feilaktig og kobles umiddelbart bort fra kontrollkretsen, mens den feilfrie datamaskin fortsetter sitt normale arbeide. Dette vil senere bli belyst med et eksempel. Bestemmelsen av den feilaktige datamaskin skjer med høyeste prioritet som er ennå høyere enn prioritetsnivået A. Dette innebærer, at selv om datamaskinene ved oppdagelse av feil skulle arbeide på A-nivået, må arbeidet avbrytes og de i sentralenheten forekommende oppgaver må lagres i et for dette formål i datalagringen spesielt beregnet delfelt. Den eneste forsinkelse, som oppstår, er at mikroprogrammet må fullføres i overensstemmelse med det i ordreregisteret innskrevne instruksjonsord for at adressen til neste instruksjon skal stå til disposisjon ved gjenopptagelse av normalt arbeide. Dette er blitt forklart i forbindelse med lagringen av B- og C-nivåets oppgaver, når datamaskinen hver 10 ms går over til A-nivået. Nevnte forsinkelse er av betydelig mindre størrelse, f. eks. 10^{-7} s, og forsvinnende liten i sammenligning med A-Perioden (10^{-2} s).

Efter at det er blitt fastslått hvilken datamaskin som var den feilaktige, og denne er blitt koblet bort, tilbakeføres den lagrede informasjon til sentralenheten og arbeidet på A-nivået fortsetter. Hvis datamaskinene ved oppdagelse av en feil har arbeidet på B- eller C-nivået, skjer lagringen på samme måte som beskrevet i forbindelse med overgang til A-nivået. Efterat den feilaktige datamaskin er blitt koblet bort, begynner lokaliseringen av feilen i denne. Dette skjer imidlertid ikke med høyeste prioritet, da lokaliseringen av feilen ikke har slik hast, at den behøver å skje på det normale programs bekostning. Følgelig får hverken nivået A eller nivået B brytes for at feillokaliseringen skal påbegynnes, men feillokaliseringen skjer på nivået C. Kontrollkretsen KK kobler de enkelte funksjonsenheter i den feilaktige datamaskin hver for seg i tur og orden til den feilfrie datamaskin og lar hver slik funksjonsenhet sammen med tilsvarende funksjonsenhet i den feilfrie datamaskin utføre en testberegning. Det fremgår av fig. 1 at hvis f. eks. datamaskinen B har vist seg å være feilaktig, kan den kobles bort ved å påvirke reléet R1b. Hvis nu ved feillokaliseringen f. eks. IM i datamaskinen B først skal undersøkes, påvirkes dels reléet R3b, hvorved instruksjonslagringsorganet IM med sitt adresseregister IA kobles til data-

maskinens A leder fl_a, og dels reléene R4b og R6b for at DR og FR skal skilles fra datamaskinens B leder fl_b. Følgen blir at bare IA i dette tilfelle kan få adresseinformasjon fra datamaskinen A, og at bare IR kan mate en ved nevnte adresse utpekt informasjon til den 16-trådede leder, som fører til sammenligningskretsen JK. Et ulikhetssignal fåes fra JK, når signalene fra datamaskinens A og datamaskinens B instruksjonslagringsorgan IM avviker fra hverandre, hvilket innebærer, at i den feilaktige datamaskin er det IM som har vært feilaktig. Hvis ikke noe ulikhetssignal fåes, innebærer dette, at IM har vært feilfri og som neste trinn innkobles DM i den feilaktige datamaskin til den feilfrie på samme måte som har funnet sted i forbindelse med IM. Hvis ikke ulikhetssignal heller nu er blitt mottatt, innkobles FE i den feilaktige datamaskin på samme måte som forut har funnet sted, og er ikke heller nu ulikhetssignal blitt mottatt, må sentralenheten CE i datamaskinen B være feilaktig. Er det blitt fastslått at en eller annen av funksjonsenhetene IM, DM eller FE har vært feilaktig, kobles denne bort i fortsettelsen og datamaskinen B settes igjen i funksjon med den forskjell i forhold til den opprinnelige tilstand, at istedenfor den bortkoblede feilaktige funksjonsenhet, f. eks. IM, anvender datamaskinen B det i datamaskinen A forekommende instruksjonslagringsorgan IM felles med den sistnevnte. Det blir på denne måte også i fortsettelsen mulig å utføre en fortløpende kontroll ved stadig å sammenholde de to datamaskiners beregningsresultat, idet kontrollen selvfølgelig ikke omfatter den felles anvendte del. Et eksempel på feillokalisering på den ovenfor beskrevne måte vil senere bli beskrevet.

Fig. 2 viser skjematisk kontrollkretsens KK funksjon. Med Ax betegnes en binær vippe, som ved forskjell mellom de to datamaskiners resultat får et fc1-signal fra JK og overføres til 1-tilstand. Fra Ax fåes et signal, som direkte peker ut en instruksjonsadresse for avlesning av underlagringsinstruksjonen, med hvis hjelp de i sentralenheten CE i registrene RA, RB, RC, SEF og LB lagrete oppgaver overføres til et underlagringsfelt i DM på samme måte som er blitt forklart i forbindelse med underlagringen av B- og C-nivået. Når feil er blitt signalisert, må også A-nivået avbrytes og oppgavene lagres under i den tid da feilbestemmelsen skjer. Det nettopp pågående mikroprogram må selvfølgelig avsluttes innen underlagringen skjer for at adressen til neste instruksjon skal stå til disposisjon, når det normale program fortsetter. Dette er antydnet med en OG-krets OK6, hvis ene inngangssignal fåes fra vippen Ax og det andre inngangssignal fåes fra styreenheten SE, når denne nullstilles etter å ha fullført en instruksjon. Signalene fra OG-kretsen OK6 tilveiebringer dels utpekning av en bestemt adresse i adresseregisteret IA, dels utlesningen av den på respektive adresse i instruksjonslagringsorganet IM forekommende informasjon, slik at denne overføres til IR og dels åpningen av OG-kretsene OK2, OK16, slik at instruk-

sjonsordet fra IR overføres til ordregisteret OR.

Denne tilstand er antydnet i fig. 7, som viser styreenheten SE, når denne utfører det ved instruksjonsordet bestemte mikroprogram. Nu er utgang nr. 4 fra avkoderen AVK1 aktivert og ved fremtrinningen av kjeden EK aktiveres lederne h1d, h2d o. s. v. i tur og orden etter hverandre. Som første trinn åpnes kretsene OK22, OK13, hvorved instruksjonsordet, som inneholder adressen til et underlagingsfelt, innskrives i DA. På denne adresse skal lagres den første oppgave som overføres fra sentralenheten, og denne oppgave blir innholdet i registeret RA. For dette formål må under neste trinn i mikroprogrammet OG-kretsene OK6, OK8 åpnes, og samtidig skriveinstruksjonen sendes til datalagringen DM, hvorved det i DR innførte ord innskrives i den i DA angitte adresse. For at neste underlagring skal kunne utføres, må adressen bestemmes og denne fås ved at den forrige lagringsadresse økes med 1. Adressen, som finnes i DA, overføres til AA ved at OG-kretsene OK3, OK5 åpnes under neste trinn. Adressen i AA økes nu med 1 ved at inngangen In1 for LE aktiveres, idet den økte adresse åpenbarer seg i AR. Denne adresse overføres til DA ved at OG-kretsene OK15, OK13 åpnes. Neste trinn blir lagring av registerets RB innhold, hvilket overføres til DR ved at OG-kretsene OK11, OK8 åpnes og skriveinstruksjon samtidig sendes, slik at innholdet i RB lagres under den angitte adresse. Neste trinn blir, at innholdet i DA overføres til AA ved at portene OK3, OK5 åpnes, innholdet i AA økes med 1 ved at inngangen In1 aktiveres og resultatet, som fåes i AR, overføres til DA ved at OG-kretsene OK15, OK13 åpnes. Som neste trinn overføres innholdet i registeret RC til DR ved at OG-kretsene OK21, OK8 åpnes og skriveinstruksjon sendes til DM, hvorved innholdet fra RC lagres i den angitte adresse. Neste trinn blir at innholdet i DA overføres til AA ved at OG-kretsene OK3, OK12 åpnes, og økes med 1 ved at inngangen In1 aktiveres, idet resultatet fås i AR. Neste trinn blir at portene OK15, OK7 åpnes, hvorved innholdet i AR overføres til IA, hvor det anvendes som adresse for avlesning av en ny instruksjon. Denne nye instruksjon overføres så fra IR til ordregisteret OR ved at portene OK2, OK16 åpnes. Dette var det siste trinn i mikroprogrammet, ved hvilket underlagringen av innholdene i registerne RA, RB, RC og i den logiske enhet er blitt fullført. Den nye instruksjon som er innmatet i ordregisteret, innebærer påbegynnelsen av selve testprogrammet, som ifølge eksemplet er kontroll på at en enkel addisjon utføres korrekt, og som er lagret på et bestemt sted i datalagringen DM. Oppgaver som hører til denne test, skal nu overføres til den logiske enhet LE. Dette mikroprogram er skjematisk antydnet i fig. 8, som på samme måte som fig. 7 viser det nye instruksjonsord innskrevet i ordregisteret. Dette instruksjonsord innebærer, at utgang nr. 5 nu aktiveres på dekodere AVK1. Ved det første trinn aktiveres tråden h1e, hvorved portene OK22, OK13

åpnes og innholdet i OR overføres til DA og sifferposisjonene 5-8 anvendes som felt-adresse i DM. Innholdet på denne adresse skal avleses og fra DR overføres til LE, nærmere bestemt til RA. For dette formål sendes først leseinstruksjon og under trinnforflytningskjedens EK neste trinn åpnes portene OK9, OK5 og innholdet i DR overføres til RA. Derefter overføres adressen i DA, som har angitt delfeltet, til AA ved at portene OK3, OK12 åpnes i kjedens tredje trinn. Adressen må økes med 1, for hvilket formål inngangen In1 i LE aktiveres og resultatet fås i AR. Dette blir den nye adresse, som må overføres til DA ved at portene OK15, OK13 åpnes og leseinstruksjon sendes til DM, slik at oppgaven overføres til DR. Denne oppgave må nu overføres til RA ved at portene OK9, OK5 åpnes. Innholdet i RA overføres nu til AA ved at portene OK6, OK12 åpnes og innholdet i RB overføres til AR ved at portene OK11, OK14 åpnes, hvorefter addisjon skjer ved at inngangen In2 på den logiske enhet LE aktiveres. Resultatet i AR lagres i RA ved at portene OK15, OK5 åpnes. Adressen til neste oppgave i DM må nu bestemmes ved at innholdet i DA, som har bestemt adressen til den sist mottatte informasjon, overføres til AA for å økes med 1. Først åpnes portene OK3, OK12, hvorefter inngangen In1 aktiveres for å addere 1 til innholdet i AA. Resultatet fra AR overføres til DA ved at portene OK15, OK13 aktiveres, hvorpå innholdet på den gitte adresse i DM avleses og lagres i DR. Nevnte adresse har inneholdt nettopp det sluttresultat, med hvilket det tidligere mottatte addisjonsresultat skal sammenlignes. Innholdet fra DR overføres først til RB ved at OG-kretsene OK9, OK10 åpnes. Nu er addisjonsresultatet lagret i RA og kontrollresultatet er lagret i RB. Neste trinn blir dels at addisjonsresultatet overføres fra RA til AA ved at portene OK6, OK12 åpnes, dels at kontrollresultatet overføres fra RB til RA ved at portene OK11, OK14 åpnes. Derefter skjer sammenligning i LE ved at inngangen In3 aktiveres.

Det er to muligheter. Hvis sammenligningen har gitt resultatet 0, aktiveres flaggvippens SEF 0-inngang, hvilket innebærer at programmet fortsetter på en slik måte, at hopp tilbake til lederen h11e finner sted, hvor EK igjen begynner trinnforflytningen, d. v. s. adressen til neste delfelt bestemmes på samme måte som forut har funnet sted. Adressen i DA overføres til AA ved at portene OK3, OK12 åpnes, denne adresse økes med 1 ved at inngangen In1 påvirkes, hvorved adressen til neste testoppgave fås. Hvis det skulle vise seg etter hele testprogrammets fullføring, at begge maskiner har løst samtlige problem korrekt, innkobles igjen begge datamaskiner, da forskjellen mellom datamaskinenes resultat må ha berodd på en tilfeldig feil. Om sammenligningen derimot viste, at forskjell mellom det beregnete resultat og kontrollresultatet foreligger, 1-stilles en vippe Vf, og feil indikeres til kontrollkretsen KK (fig. 8, fig. 2). Denne inneholder en OG-krets OA resp. OB for resp. datamaskiner, hvilke OG-kretsers ene inngang består av feilsignalet fra resp. datamaskin, og den

andre inngang består av feilsignalet fra vippen Ax, som ble påvirket, når sammenligningskretsen JK indikerte forskjell mellom de to datamaskiner. Hvis det har vist seg, at det av f.eks. datamaskinens B utregnete resultat ikke stemte med kontrollresultatet, føres vippen Vfb til 1-stilling og reléet R1b tiltrekker, hvilket, som det fremgår av fig. 1, innebærer frakobling av datamaskinen B. I fortsettelsen arbeider nu bare datamaskinen A. Samtidig med datamaskinens B frakobling er vippen Vfc blitt overført til 1-stilling, hvilket innebærer at ventetilstand råder for at man i et hensiktsmessig øyeblikk skal kunne påbegynne feillokaliseringen, d.v.s. bestemme hvilken funksjonsenhet som er feilaktig i den feilaktige datamaskin. Som tidligere nevnt i forbindelse med fig. 10, er det en vippe for feltene for hvert av A- og B-nivåene VA resp. VB, som er 1-stilt så lenge det er arbeide for resp. nivå. En av betingelsene for at test av funksjonsenhetene skal kunne påbegynnes er at arbeide ikke lenger finnes være seg på A-nivået eller på B-nivået, da disse ikke får brytes for detaljtesten. På den annen side skal testprogrammet fortrinnsvis utføres når C-nivåets program kan påbegynnes, og da har det prioritet foran det vanlige C-nivåprogram. Betingelsen for at det vanlige C-nivåprogram skal kunne påbegynnes, er at B-feltets indikeringsvippe VB er 0-stilt, d.v.s. at samtlige arbeidsbiter i B-feltet er 0-stilt. Samme betingelse gjelder for påbegynnelse av detaljtestprogrammet, men detaljtestprogrammet har fortrinnsrett foran det vanlige C-nivåprogram. Er det således feilindikering fra vippen Vfc, påbegynnes testprogrammet istedenfor det normale C-nivåprogram.

Dette er symbolisert i fig. 2 ved hjelp av en logisk kobling. OG-kretsene OP og OS har hver to innganger, av hvilke den ene er koblet for begge OG-kretser til vippens VB 0-stilling, da begges aktivering er en følge av at det ikke lenger finnes arbeide igjen på B-nivået. Den andre inngang for kretsen OP er utgangen fra vippens Vfc 0-stilling, hvilket innebærer at OG-kretsen OP trer i funksjon, når B-nivåets program ble avsluttet og en eller annen feil ikke foreligger. Derved utpekes adressen, hvor C-nivåets neste oppgave er lagret i datalagringen DM, oppgaven avleses og overføres fra registeret DR til sentralenheten, slik at C-nivåets normale arbeide fortsetter. Kretsens OS andre inngang er koblet til vippens Vfc 1-stilling, d.v.s. den blir aktiv, når feil foreligger, mens kretsen OP ikke kan bli ledende. Følgelig settes datamaskinens feillokaliseringsprogram igang, hvilket er skjematisk antydnet i fig. 2. Ved hjelp av lederen OS utpekes en adresse i instruksjonslagringsorganets IM adresseregister IA for å bestemme en testinstruksjon, og samtidig kobles et signal til en dekode AVK5, som ved mottagelse av dette signal påvirker en viss relékombinasjon, f.eks. reléene R2b, R5b, R6b, som ved sin tiltrekning bevirker, at DM i datamaskinen B kobles til datamaskinens A felles ledning, mens IM og FE helt frakobles. Dessuten skjer avlesing i den feilfrie maskin av den på den angitte adresse forekommende infor-

masjon og overføring til IR samt overføring av innholdet i IR til ordregisteret OR ved at OG-kretsene OK2, OK16 åpnes. Samme adresse, som er blitt innskrevet i IA, overføres også til sentralenheten CE og lagres der i registeret RC ved at OG-kretsene OK3, OK20 åpnes (se fig. 12). Nu er instruksjonsordet for feillokaliseringstesten innskrevet i ordregisteret OR som antydnet i fig. 12, og testen utføres av mikroprogrammet på lignende måte som er blitt forklart i forbindelse med fig. 7 og 8. For feillokaliseringen aktiveres utgang nr. 6 fra dekoderen AVK1 og utgangene fra trinnforflytningskjeden EK aktiveres i tur og orden som tidligere forklart. Ved lederens hlf aktivering åpnes OK22, OK7, hvorved innholdet i OR overføres til IA for at en del av dette, f. eks. sifferposisjonen 5-8, skal anvendes som adresse, og leseinstruksjon sendes for å overføre innholdet på nevnte adresse til IR. Innholdet i IR er en ny adresse, som anvendes for test av DM, og den er valgt på en slik måte, at avlesning av den på adressen forekommende informasjon gir svar på at en feil av en bestemt type foreligger i resp. funksjonsenhet, ifølge eksemplet i DM. Adressen kan f. eks. være 0101010101010101 og den på denne adresse forekommende informasjon kan være 1010101010101010 og forbindelsen mellom adresse og den på adressen forekommende informasjon er valgt på en slik måte, at etter minst mulig antall operasjoner opptrer samtlige feiltyper i den undersøkte funksjonsenhet. Innholdet fra IR overføres nu til DA ved at OG-kretsene OK2, OK13 åpnes. Kretsen OK13 åpnes imidlertid også i datamaskinen B, hvis DM jo står i forbindelse med datamaskinens A 16-trådede ledning fl_{1a}, ved hjelp av datamaskinens A styreenhet SE. Avlesning skjer nu i begge datamaskiners DM, idet den avleste oppgave overføres til begges DR og sammenligning må gjøres, hvis det binære ord i de to registre DR er like. Dette kan skje på en slik måte, at såvel i datamaskinen A som i datamaskinen B, i hvilken DR fremdeles er koblet til maskinens B forbindelsesleder fl_{1b}, åpnes porten OK9, hvorved innholdet i begge DR mates til sammenligningskretsen og sammenholdes der. I tilfelle sammenholdelsen viser, at det ikke foreligger noen forskjell og at vippen Ax er i 0-stilling (fig. 2, fig. 12), fortsetter undersøkelsen og som det er symbolsk antydnet i fig. 12, skal nu informasjonen i IA anvendes for å få neste adresse ved at den økes med 1. For dette formål overføres innholdet i IA til AA ved at lederen h_{4f} aktiveres og portene OK4, OK12 åpnes. Neste trinn blir, at man adderer 1 til innholdet i AA, og det resultat som fås, overføres til IA ved at OK15, OK7 åpnes, og det danner der neste instruksjonsadresse. Avlesning skjer og den til adressen hørende instruksjon fås i IR. Herfra overføres den til DA ved at OK2, OK13 åpnes. Avlesning skjer, og innholdet i DM i datamaskinene A og B skal sammenlignes på samme måte som har funnet sted under den foregående test, ved at datamaskinen A påvirker OK9 i begge maskiner, hvorved innholdet i DM i resp. maskiner kommer til maskinens 16-trådede ledere, slik at det

kan sammenlignes. Hvis det ikke er noen avvikelse, fortsetter sammenligningen et bestemt antall ganger på samme måte, og er ikke noen avvikelse blitt påtruffet etter disse trinn, går datamaskinen over til testing av neste funksjonsenhet, d. v. s. IM eller EF, som testes på samme måte som ovenfor beskrevet. Det ovenfor nevnte forløp er skjematisk antydnet i fig. 2 og fig. 12. Ved avslutning av hver syklus (lederne h4f-h8f i fig. 12) fås en puls, når porten OK9 åpnes, og et skiftregister SK1 beveges frem et trinn. Etter et visst antall trinn, f. eks. 4, fås en utgangspuls fra skiftregisteret SK1 og et annet skiftregister SK2 mates frem et trinn (fig. 2, fig. 12). Dette innebærer at neste inngang på avkoderen AVK5 aktiveres, hvilket medfører at nu tiltrekker de reléer som tilveiebringer innkobling av instruksjonslagringsorganet IM i datamaskinen B til samarbeide med datamaskinen A. Forløpet blir lignende, når et visst antall sykler er blitt avsluttet uten at forskjell er blitt påtruffet, idet skiftregisteret SK2 beveges frem trinnvis til stillingen 3 og formidlingsenheten FE i datamaskinen B kobles til samarbeide med datamaskinen A. Hvis forskjell heller ikke nu er blitt påtruffet, innebærer dette, at sentralenheten i datamaskinen B var feilaktig resp. at feilen var av forbigående slag.

Hvis det skulle ha vist seg under test at en eller annen av funksjonsenhetene, f. eks. av DM, at forskjell finnes og vippen Ax har inntatt 1-stilling, påbegynnes et annet forløp som i fig. 12 er symbolisert med krysningspunktene mellom trinnforflytningskjedens utganger og en leder v2, som er koblet til vippens Ax "1"-utgang. Etter konstatering av feilen må adressen, hvor man fikk feil i DM, avleses for å påvirke manøvreringsorgan og lagres for anvendelse senere. For dette formål overføres den først fra IR til RB ved at portene OK2, OK12 åpnes (lederen h4g aktiveres), hvor den fastholdes inntil den er blitt anvendt for manøvreringen av til- og frakoblingsorgan, da den også definerer den funksjonsenhet, i hvilken feilen er blitt påtruffet. Adressen, hvor feilen er blitt påtruffet, må imidlertid også lagres i datalagringen for å kunne anvendes siden for feilens nærmere lokalisering, hvorfor innholdet i RB også overføres til DR ved at portene OK11, OK8 åpnes.

Som nevnt, er ved påbegynnelsen av testforløpet instruksjonsadressen for testen blitt lagret i registeret RC ved at portene OK3, OK20 er blitt aktivert (se fig. 2). Denne adresse økes med 1, hvorved man får en lagringsadresse hvor oppgaven vedrørende feilens plass kan lagres i DM. Adressen i DC mates til AA ved at portene OK21, OK12 åpnes, hvorefter innholdet i AA økes med 1 og resultatet fås i AR. Innholdet i AR overføres nu til datalagringens DM adresseregister DA ved at portene OK15, OK13 åpnes. Det skal fremholdes, at adressen, hvor man fikk feilen, stadig er i DR, hvor den ble overført fra RB. Når nu en skriveordre mates til DM samtidig med den fra AR til DA overførte underlagringsadresse, innskrives innholdet i DR til

denne underlagringsadresse. Nu må de manøvrer utføres, som frakobler den feilaktige funksjonsenhet IM, DM eller FE og kobler samtlige feilfrie funksjonsenheter samt CE i datamaskinen B til samarbeide med datamaskinen A. Datamaskinens funksjon fortsetter på samme måte som før feilen ble indikert, men den forskjell at istedenfor den feilaktige funksjonsenhet, f. eks. DM, anvendes den feilfrie DM felles av begge datamaskiner. Derved finnes stadig en feilkontrollmulighet ved sammenligning mellom de på resp. maskiners forbindelsesledere opptredende verdier bortsett fra naturligvis i det tilfelle, at feilen skulle oppstå i DM. For å tilveiebringe dette blir neste trinn i mikroprogrammet, at innholdet i RB avleses og mates til avkoderen AVK4 ved at man åpner portene OK11 og, f. eks. OK31 (fig. 2) hvis det dreier seg om bortkobling av den feilaktige DM. Derved fås et utgangssignal fra OG-kretsen OK31, og det mates til dekoderen AVK5, som i overensstemmelse med det oppnådde signal manøvrerer reléet R4b, som frakobler det feilaktige DM (fig. 12).

Oppfinnelsen er selvsagt ikke begrenset til det beskrevne utførelseseksempel og det er innlysende at hverken utformingen av datamaskinens normale program eller utformningen av deres testprogram har noe med selve oppfinnelsen å gjøre. Det vesentlige er at funksjonsenhetene i den feilaktige datamaskin kobles inn i tur og orden for samarbeide med den feilfrie datamaskin inntil en forskjell er blitt påtruffet, hvorefter den feilaktige funksjonsenhet bortkobles og de øvrige fortsetter sin normale funksjon.

Patentkrav

1. Fremgangsmåte for å lokalisere en feil i et system bestående av minst to parallelt og synkront arbeidende datamaskiner som er bygget opp av hver sin sentralenhet (CE) og av i respektive datamaskiner identiske funksjonsenheter (MD) med hver sin bestemte lagringsfunksjonstype, til og fra hvilke enheter signaler mates ved utførelsen av regneoperasjoner og hvilke datamaskiner hver for seg og synkront utfører samme regneoperasjon for å muliggjøre at ved forekomst av en feil blir i det minste den ene datamaskins resultat korrekt, karakterisert ved at nevnte signaler til og fra funksjonsenhetene i de to datamaskiner stadig sammenholdes i en sammenligningskrets, som ved avvikelse ved hjelp av en kontrollkrets avbryter datamaskinens normale funksjon og setter igang en i begge maskiner identisk regneoperasjon, hvis resultatavvikelse fra en fastsatt verdi påvirker et frakoblingsorgan for respektive datamaskin, slik at den feilfrie datamaskin kan fortsette sitt normale arbeide, idet den feilfrie datamaskin i tur og orden kobler inn en funksjonsenhet i den feilaktige datamaskin for samarbeide med den feilfrie maskin på en slik måte, at den utfører samme

funksjon som tilsvarende funksjonsenhet i den feilfrie datamaskin, idet resultatene fra nevnte to funksjonsenheter stadig sammenlignes og ved avvikelse bortkobles den til den feilaktige datamaskin hørende funksjonsenhet, mens samtlige øvrige funksjonsenheter inklusive sentralenheten i den hittil feilaktige datamaskin igjen kobles inn for parallellarbeide med den feilfrie datamaskin under felles anvendelse av den bare i den ene datamaskin feilfrie funksjonsenhet.

2. Fremgangsmåte som angitt i krav 1, karakterisert ved at ved sammenligning mellom funksjonsenheter i tur og orden i de to datamaskiner og ved oppnåelse av identisk regneresultat settes den som feilaktig indikerte datamaskin igjen igang i sin helhet, idet ved oppnåelse av avvikende resultat bortkobles den feilaktig arbeidende datamaskins sentralenhet, mens ved overensstemmende resultat fortsetter den normale funksjon av de to datamaskiner, da den ved den første resultatavvikelse indikerte feil bare var av tilfeldig karakter.

3. Fremgangsmåte som angitt i krav 1 eller 2, karakterisert ved at datamaskinene arbeider ved forskjellige prioritetsnivåer for utførelse av oppgaver av forskjellig viktighetsgrad, idet den anvendte regneoperasjon for utpekelse av den feilaktig arbeidende datamaskin har høyere prioritet enn den høyeste prioritet i datamaskinens normal funksjon, mens de for utpekelse av de enkelte feilaktige funksjonsenheter anvendte regneoperasjoner har lavere prioritet.

Anførte publikasjoner: -

118944

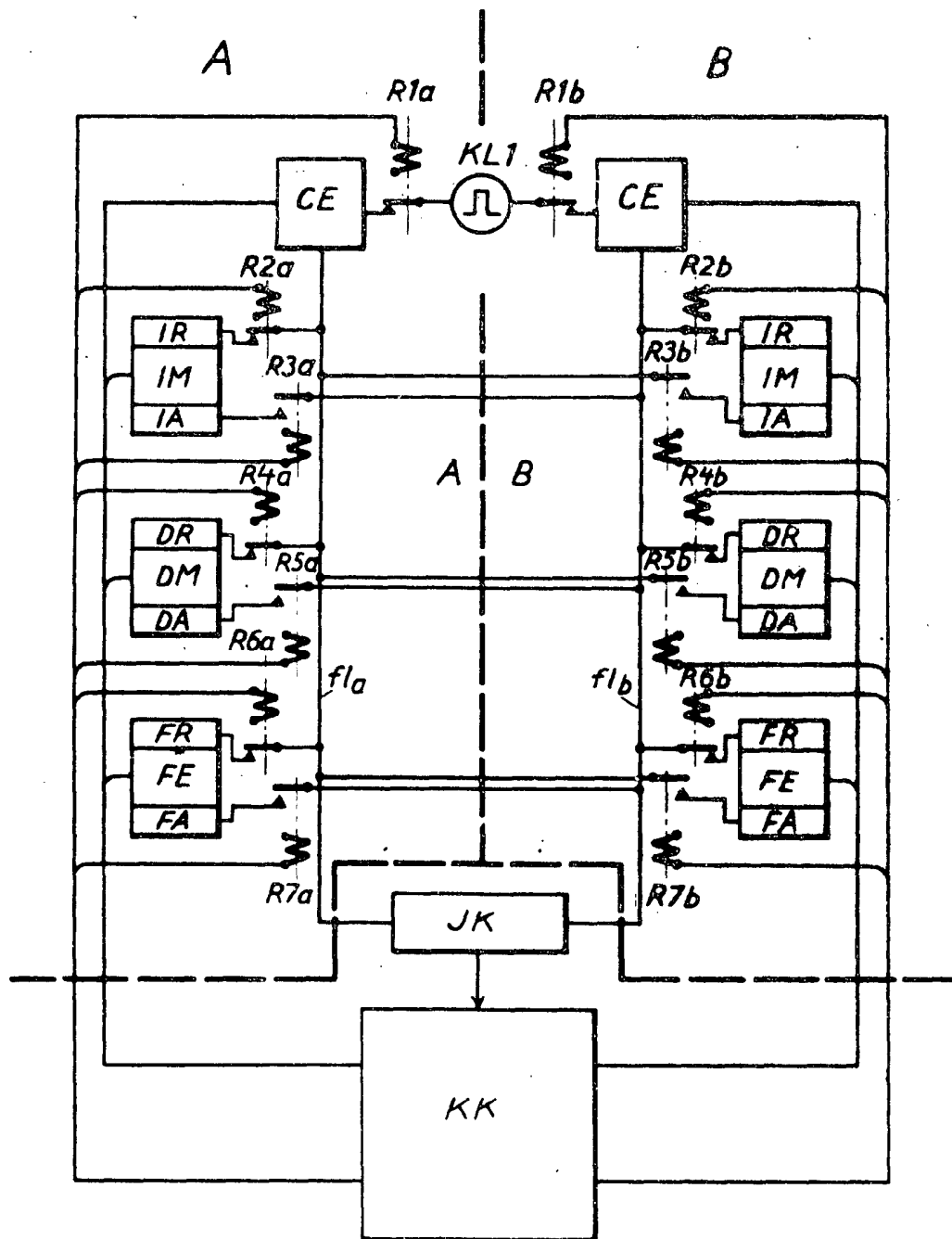


Fig. 1

118944



Fig. 2

118944

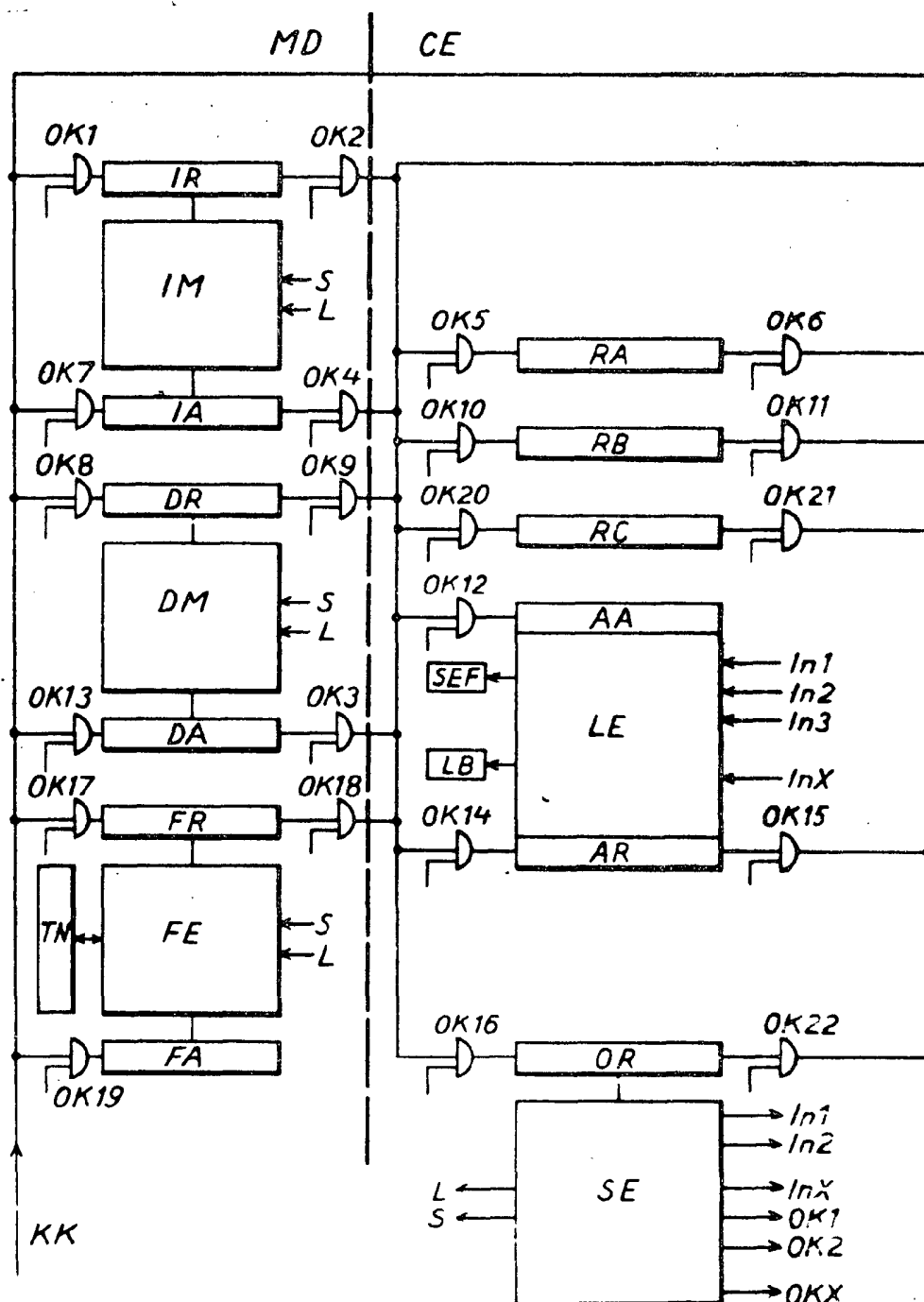


Fig.3

118944

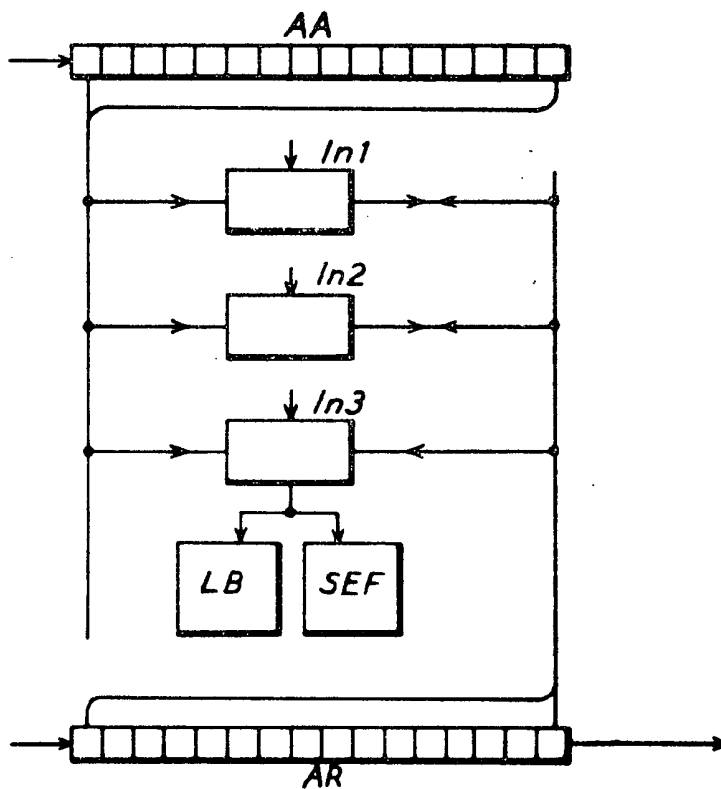


Fig. 4

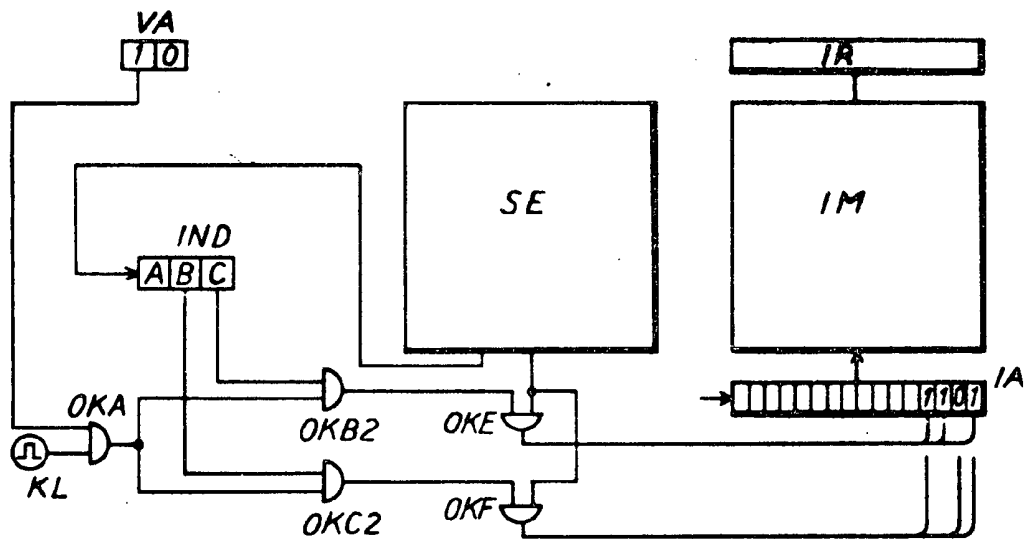
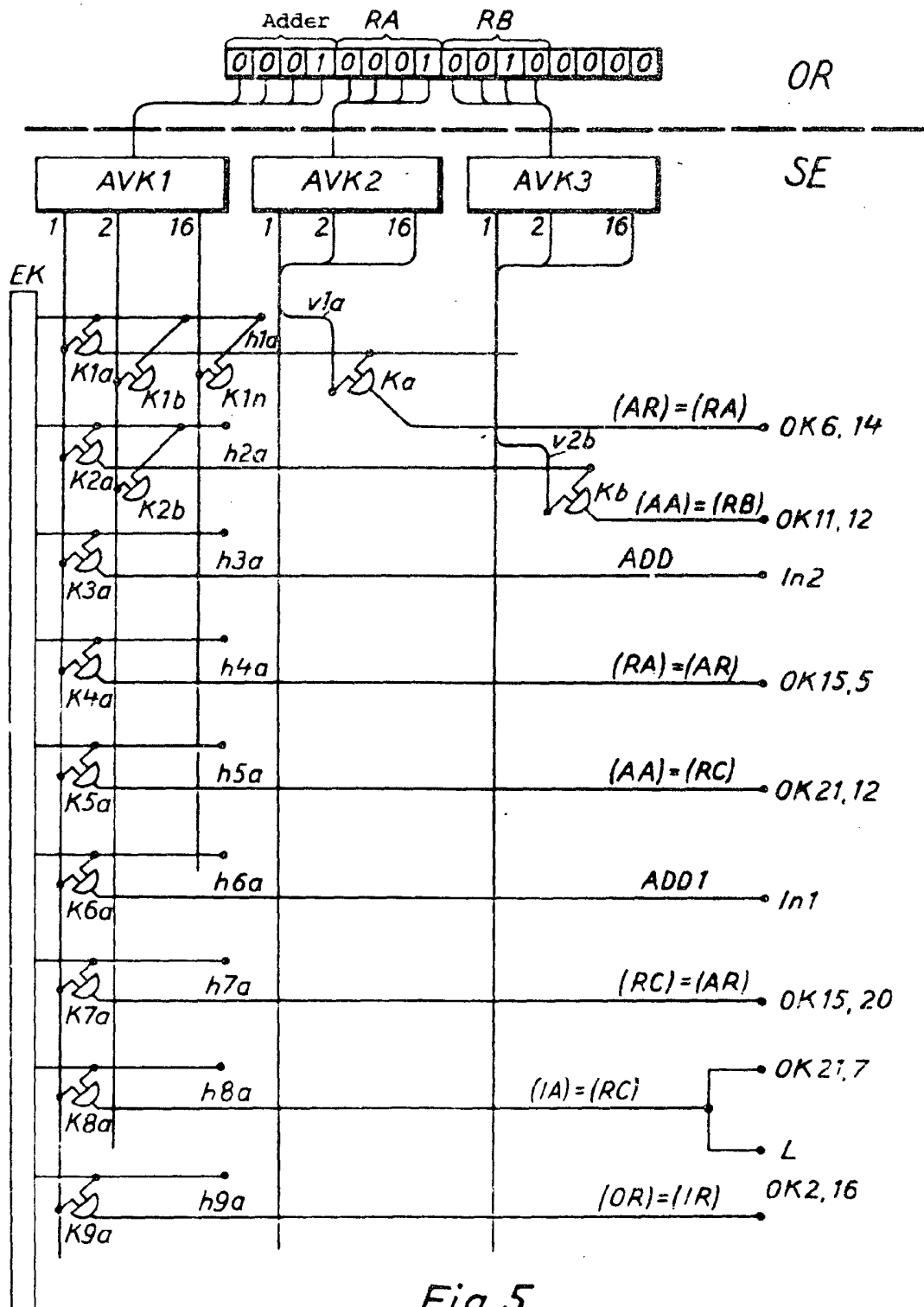
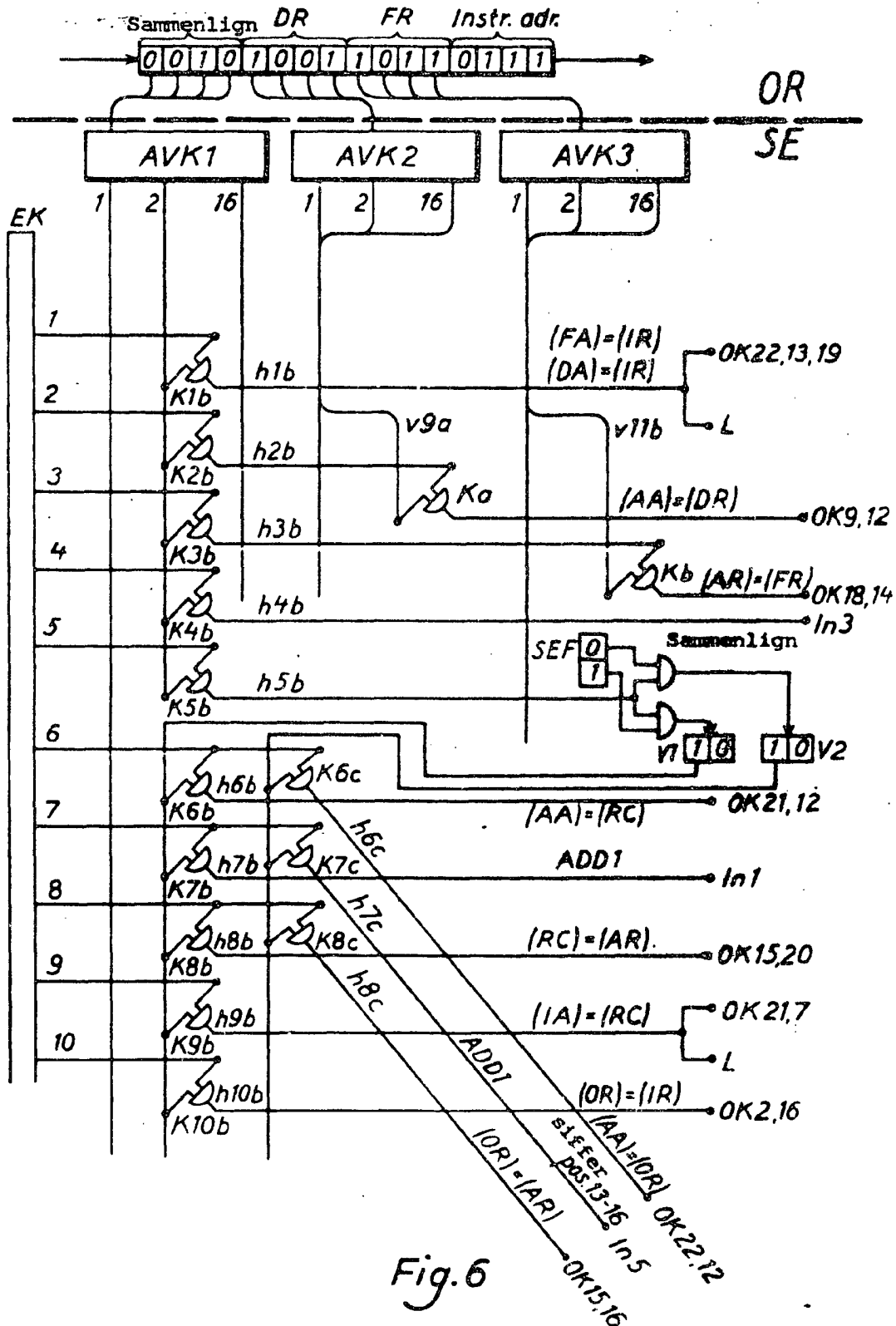


Fig. 11

118944



118944



118944

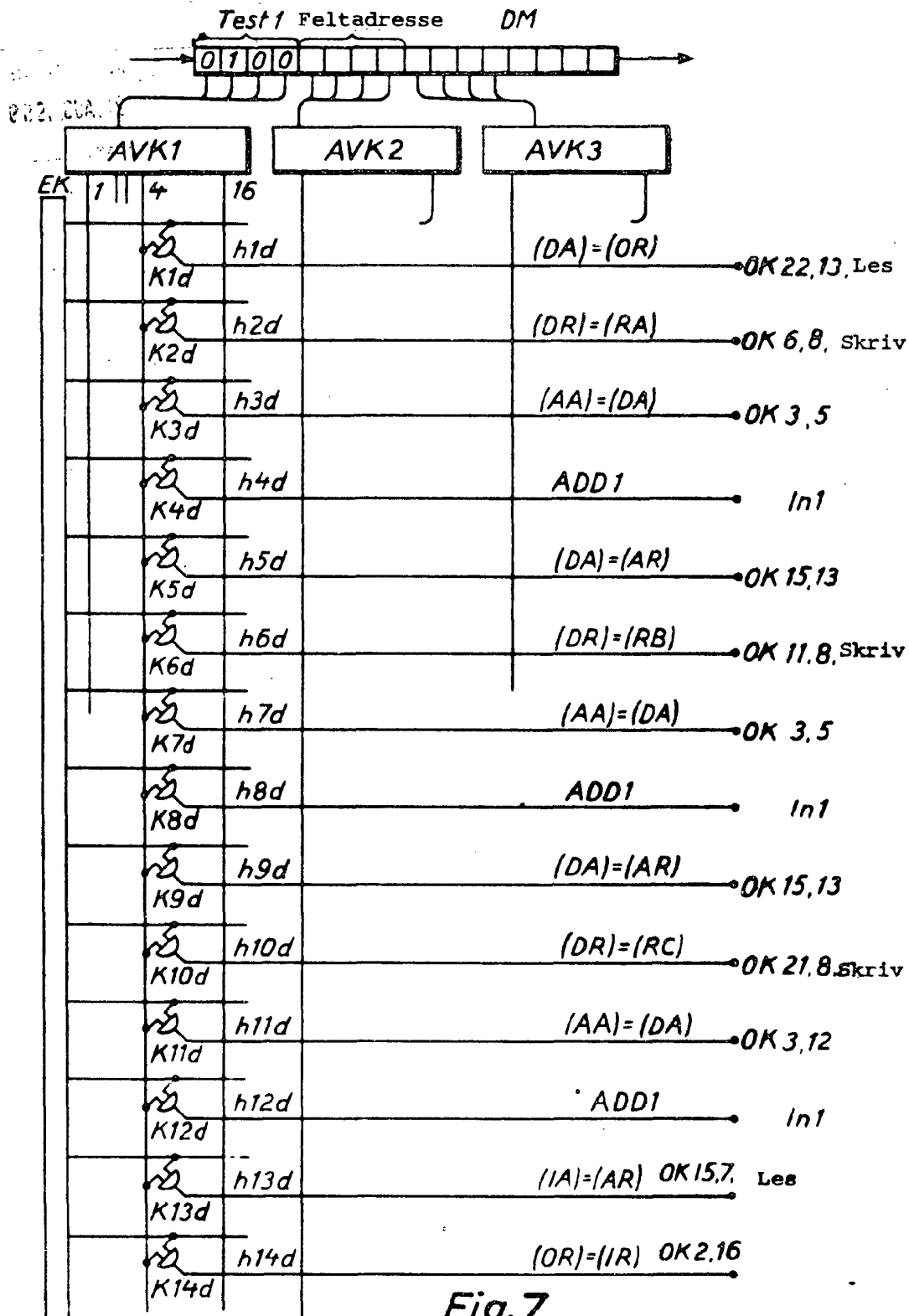


Fig. 7

118944

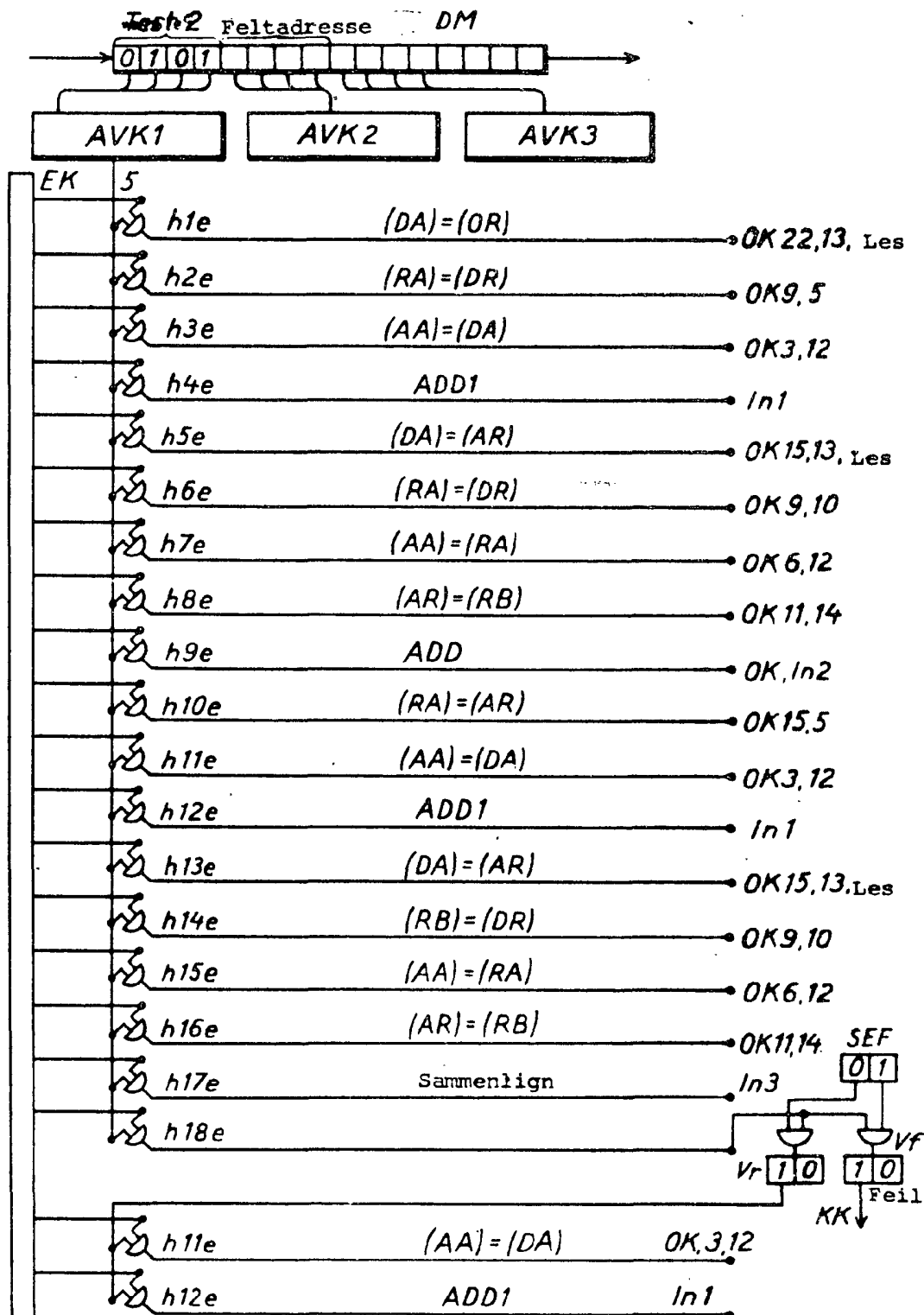


Fig. 8

DM

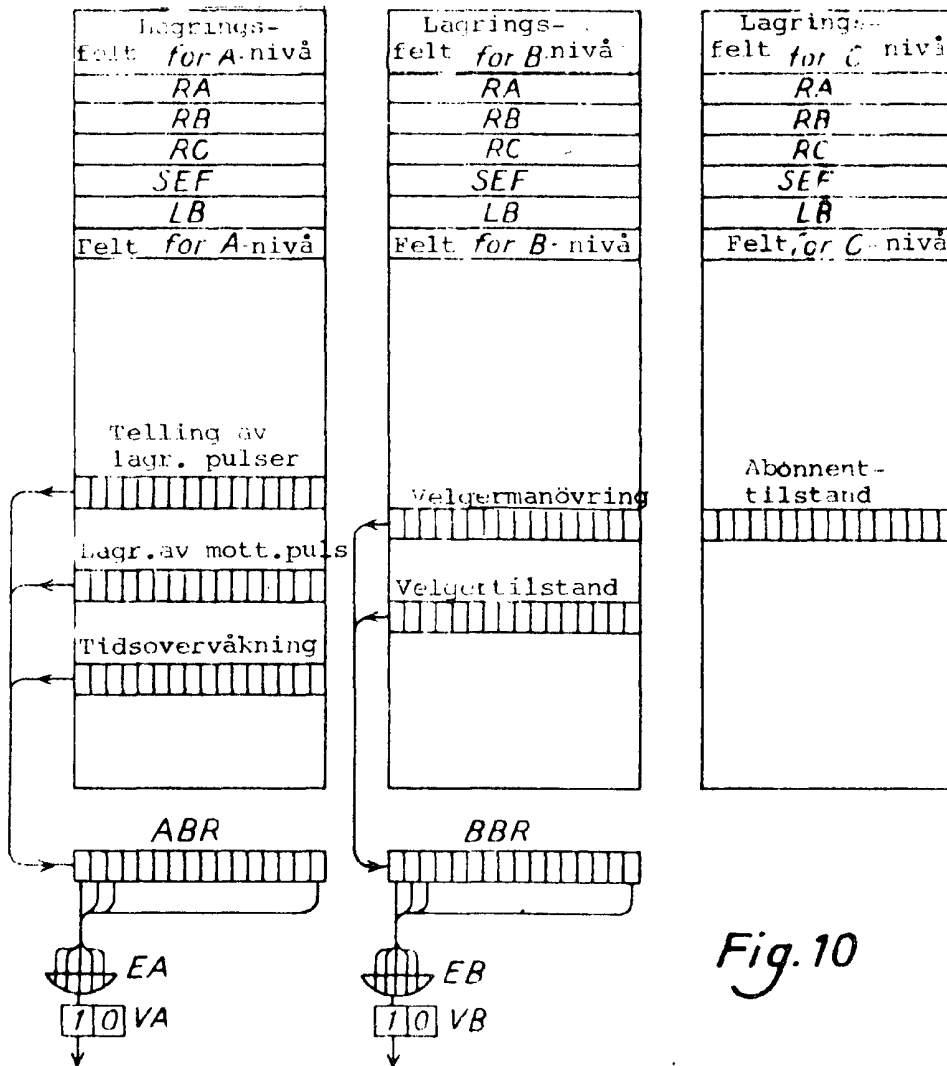


Fig. 10

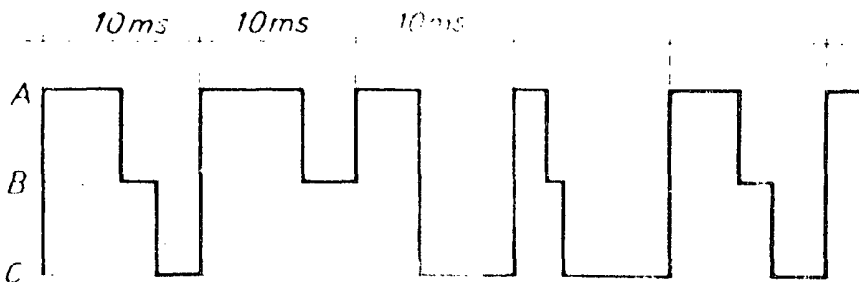


Fig. 9

118944

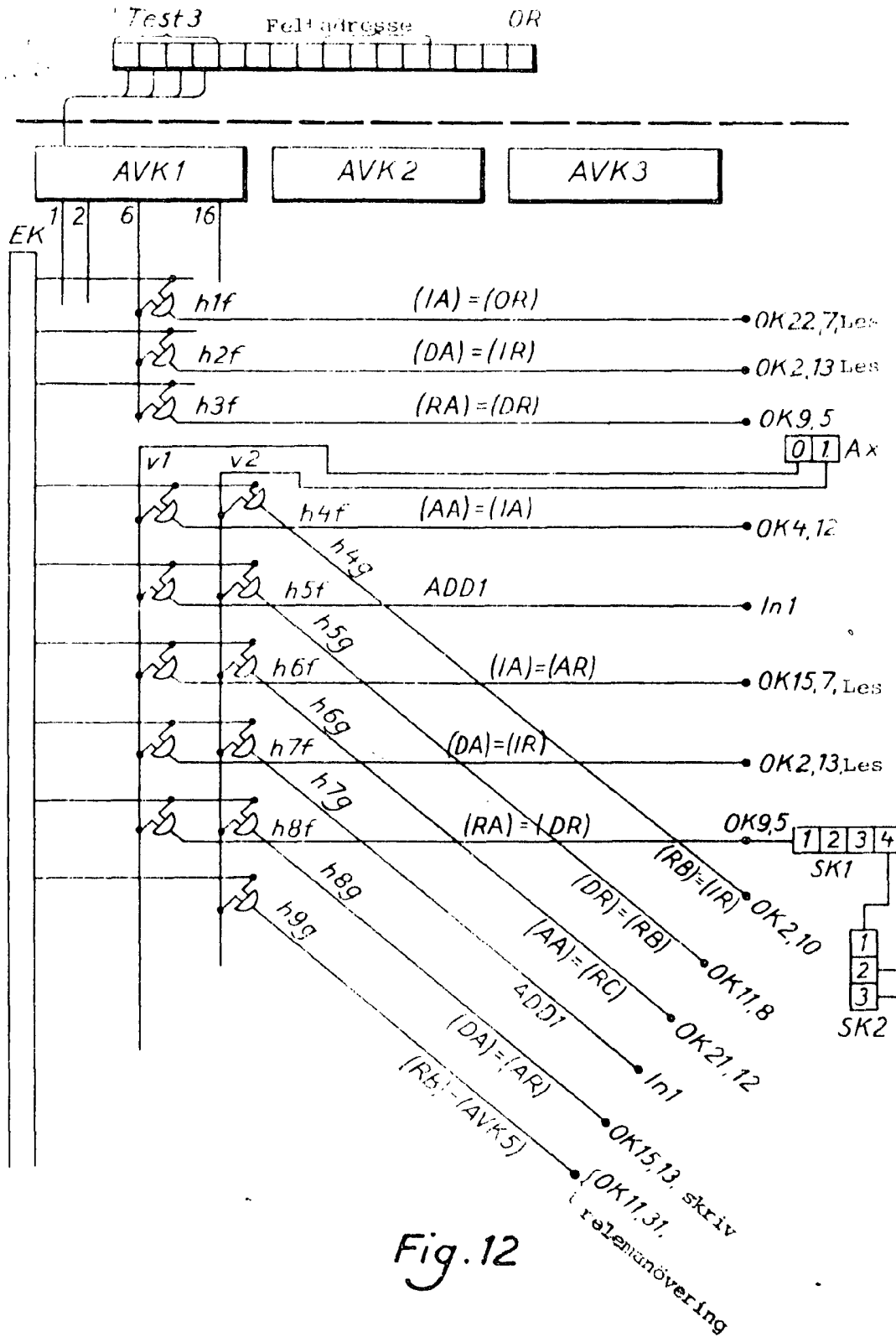


Fig. 12