



Patentdirektoratet
TAASTRUP

(21) Patentansøgning nr.: 5269/84

(51) Int.Cl.5

G 06 F 11/22

(22) Indleveringsdag: 06 nov 1984

(41) Alm. tilgængelig: 14 jun 1985

(44) Fremlagt: 22 mar 1993

(86) International ansøgning nr.: -

(30) Prioritet: 07 nov 1983 US 549612

(71) Ansøger: *DIGITAL EQUIPMENT CORPORATION; 146 Main Street; Maynard; Massachusetts 01754, US

(72) Opfinder: Tryggve *Fossum; US, Milton L. *Shively; US

(74) Fuldmægtig: Hofman-Bang & Boutard A/S

(54) Apparat til selv-afprøvning af flydende kommberegningssenheder

(56) Fremdragne publikationer

(57) Sammendrag:

5269-84

Der beskrives en fremgangsmåde og et apparat til kontinuerlig afprøvning af en flydende kommatalsberegningssenhed (FPAP), eller en anden beregningssenhed i et flerberegningssenhedssystem. Mindst to beregningssenheder, såsom instruktionsudførelsesberegningssenhed (EU) og en FPAP, er forbundet til en fælles indgangsbuss, hvorved de samtidig modtager den samme information (instruktionskoder og operander). Både EU'en og FPAP'en dekode instruktionskoderne. I stedet for at blive i "tomgang", mens EU'en arbejder, udfører FPAP'en en diagnosefunktion. FPAP'en vælger en diagnosefunktion til udførelse fra et antal tilgængelige diagnosefunktioner. Valget af diagnosefunktion afhænger af den instruktion, der skal udføres af EU'en. For ikke at sinke systemets udførelsestider, vælges der en diagnosefunktion, hvis udførelsestid er tilpasset udførelsestiden for den instruktion, der udføres af EU'en. Der vælges således en diagnostisk funktion af en sådan længde, at FPAP'en vil have afsluttet denne

fortsættes

funktion, før EU'en afslutter udførelsen af sin instruktion. Operanddata, der tilføres EU'en over indgangsbussen, anvendes af diagnosefunktionerne for således at tilføre en grad af tilfældighed til testsignalerne og gøre det muligt at detektere bits, der er tvunget til en fast værdi af enten nul eller et. For visse diagnostiske funktioner kan der anvendes én eller flere variable fra de generelt anvendelige registre.

EU'en modtager ikke information om en fejl fundet ved en FPAP selvdiagnosefunktion, indtil der optræder en instruktion, der skal udføres af FPAP'en, på hvilket tidspunkt en sådan instruktion kan afbrydes eller fuldføres i afhængighed af, om kilden til fejlen vil påvirke instruktionen eller ej.

På denne måde undersøger FPAP'ens dataveje og styrelogik ved udførelsen af hver ikke flydende kommatalsinstruktion. Systemets svartider forøges ikke, og der tilføres kun en ubetydelig omkostningsforøgelse ved opbygning af systemet, og endelig er diagnosesystemet fuldstændigt transparent for programmører.

5269-84

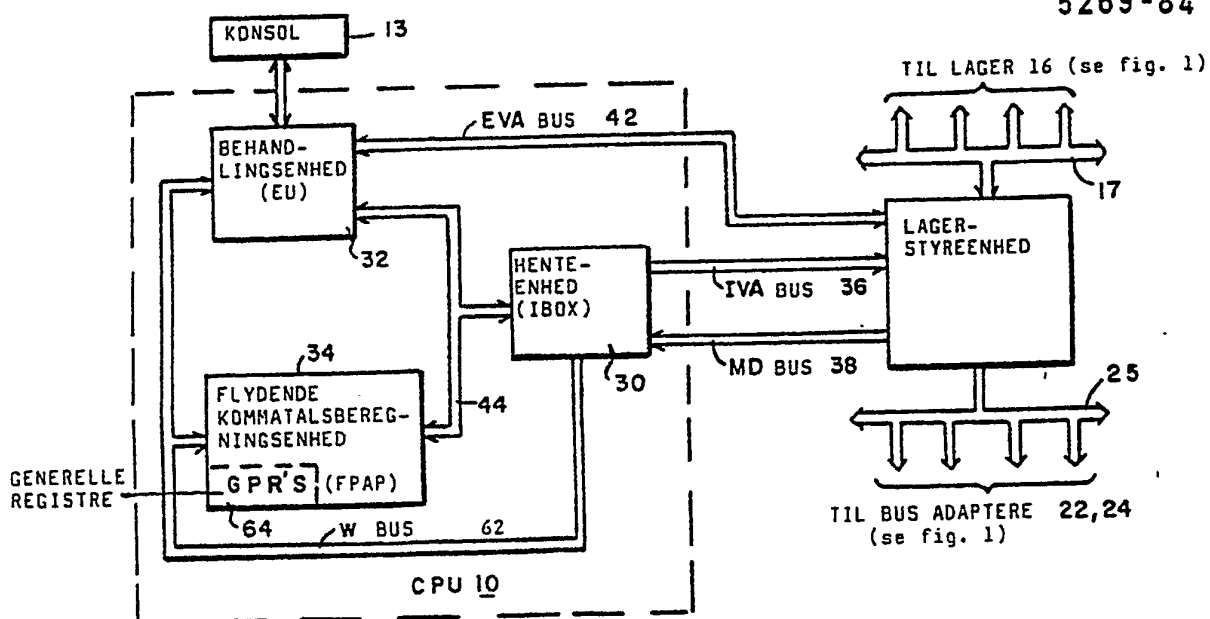


FIG.2

Den foreliggende opfindelse omhandler digitale data-behandlingssystemer, og mere specifikt "on-line" afprøvning af beregningsenheder i databehandlingssystemer.

5 Specifikt omhandler den foreliggende opfindelse diagnostisk afprøvning af en beregningsenhedsdel i en flydende kommaberegningssenhed i et databehandlingssystem. Denne afprøvning udføres, uden at systemets svar- eller udførelsestider ødelægges og fordyrer så at sige ikke systemet. Apparatet anvender systemer af busser, der ellers ville
10 være ledige når afprøvningen foregår. Derudover anvendes der "rigtige" data som afprøvningssekvensens operander med henblik på at forøge detektionssandsynligheden af dataafhængige fejl.

Digitale databehandlingssystemer indeholder sædvanligvis
15 tre grundlæggende dele, en lagerenhed, en indgangs/udgangs-
enhed og en beregningsenhed, idet alle disse enheder forbindes til hinanden med én eller flere busser. Lager-
enheden lagrer data i adresserbare lagerlokationer.
Disse data indeholder både operander og instruktioner
20 til behandling af operanderne. Beregningsenheden varetager, at data overføres til eller hentes fra lager-
enheden, og fortolker de indkomne data enten som instruktioner eller operander, og behandler operanderne i overensstemmelse med instruktionerne. Resultaterne lagres
25 da i adresserede lokationer i lagerdelen. En indgangs/udgangsdel kommunikerer også med lagerdelen og beregnings-
enheden, med henblik på at overføre data til systemet, og videregive de behandlede data fra systemet. Indgangs/-
udgangsdel arbejder sædvanligvis i overensstemmelse
30 med styreinformation, der overføres til den fra beregningsenheden. Indgangs/udgangsdel kan f.eks. være skrivere, fjernskrivere eller tastaturer og skærmterminaler, men kan også omfatte sekundære datalagringsenheder såsom pladelagre og båndstationer.

Databehandlingssystemer anvender ofte flere beregningsenheder til at udføre og overvåge de forskellige opgaver, som de er indrettet til at udføre. F.eks. anvendes der undertiden særskilte beregningsenheder til at udføre indgangs/udgangsfunktioner, til at styre perifere enheder, og til at udføre andre udskillelige opgaver. Derudover kan databehandlingsfunktioner også uddeles blandt flere beregningsenheder. Nogle gange anvendes en særlig art beregningsenhed, der betegnes som en flydende kommaberegningsenhed, til at udføre aritmetiske operationer med flydende komma. Flydende kommaberegningsenheder er specielt konstrueret til at forøge den hastighed, ved hvilken flydende kommaberegninger kan udføres. Når der skal udføres en beregning med flydende komma, udføres den i eller af flydende kommaberegningsenheder i stedet for en anden beregningsenhed.

Brugere og konstruktører af databehandlingssystemer kræver en høj pålidelighed og en nøjagtig funktion. På grund af dette forefindes der overalt i moderne databehandlingsenheder fejldektions- og -rettelsesmekanismer. Imidlertid kan sådanne mekanismer sædvanligvis ikke detektøre eller korrigere data, der er fejlbehæftede men ikke ødelagt logisk set. Et område hvor data med disse karakteristika kan opstå, er ved udførelse af aritmetiske operationer, specielt ved udførelsen af flydende kommaoperationer. På grund af dette har det længe været praksis i databehandlingsindustrien, at der i programmer, hvor der anvendes flydende kommaoperationer, at indbygge nogle trin der undersøger om resultatet af sådanne operationer ligger indenfor et numerisk værdiområde, i hvilket korrekte resultater sædvanligvis optræder. F.eks. vil et lønsystem for fabriksarbejdere, hvis ugentlige løn forventes at ligge indenfor et forudbestemt område på 300 til 500 \$ pr. uge, kunne forsynes

med instruktioner der sikrer, at der ikke udskrives nogen lønningscheck, der er større end et hvis givet maksimum, såsom f.eks. 500 \$. Selvfølgelig kan indgangsdata til lønprogrammet også undersøges på samme måde, for at sikre at alle parametre ligger inden for forventede områder, hvorved det f.eks. er muligt at sikre, at en arbejder ikke får udbetalt løn for en fiktiv arbejdsuge på tohundrede timer. Når man på denne måde har konstateret en fejl i beregninger med flydende komma, må man anvende diagnostiske midler til at analysere fejlen og lokalisere dens årsag. Hvis sagen er en periodisk fejl eller en programmeringsfejl, kan dette være vanskeligt.

En anden fremgangsmåde til kontrol af flydende kommaoperationer, der kan anvendes i tidsdelte systemer, er at tildele en system "bruger" en opgave, der kun indeholder en sekvens af flydende kommaberegnings-diagnosefunktioner, der gentages hele tiden. For at være virkningsfulde, kræver denne teknik sædvanligvis, at sådanne beregninger udføres med kendte data, hvorved det er muligt at sammenligne de aktuelle resultater med forudbestemte resultater. Nogle fejl kan imidlertid være dataafhængige, i hvilke tilfælde det ved afprøvningen kan være umuligt at finde fejlen, med mindre de indgående data varieres af og til. Derudover kan der tages mange fejlbehæftede flydende kommaberegninger mellem det tidspunkt, hvor der sker en fejl i den flydende kommaberegningsenhed, og det tidspunkt hvor den næste afprøvning, der kan konstatere fejlen, udføres. For ikke at forøge systemets belastning i væsentlig grad, og derved forøge svartiderne for andre brugere, er det nødvendigt, at diagnosefunktionerne kun optager systemets beregningsenheder i en lille brøkdelt af tid. Men herved kan der i flydende kommaberegningsenheden forårsage forkerte resultater, før fejlen detekteres af diagnosesystemet.

Disse andre begrænsninger ved den kendte teknik, er det formålet at overvinde ved den foreliggende opfindelse, der angiver et apparat til kontinuert afprøvning af en flydende kommaberegningsenhed, eller en anden
5 beregningsenhed i et passende beregningsenhedssystem. Den foreliggende opfindelse forringer ikke systemets svartider, tilføjer en ubetydelig marginalomkostning og er fuldstændig gennemsigtigt for programmøren.

Opfindelsen, som den er defineret i krav 1, angår et
10 databehandlingssystem med en første og en anden beregningsenhed, hvori den første beregningsenhed vælger en diagnostisk operation til udførelse i enheden, når den første beregningsenhed detekterer et tidsinterval, hvori den anden beregningsenhed er i gang med at udføre
15 en instruktion, og den første beregningsenhed vil være i tomgang. Den første beregningsenhed vælger en diagnostisk operation, som den kan færdiggøre, før den anden beregningsenhed er færdig med at udføre en instruktion. Opfindelsen tilbyder en fordel, idet diagnostiske operationer kan udføres i den ene beregningsenhed, når
20 den anden beregningsenhed udfører instruktioner, og diagnostiske operationer kan derfor udføres uden ekstra tidsforbrug i databehandlingssystemet.

Ifølge den foreliggende opfindelse er mindst to beregningsenheder, såsom en instruktionsudførelsesberegning
25 enhed (EU) og en flydende kommaberegningsenhed (FPAP) forbundet parallelt til fælles indgangsbuss, hvorved de samtidig modtager information, der anbringes på bussen af en anden enhed (en instruktions- og operandhentende enhed).
30 Informationen på bussen er pakket i et forudbestemt format, såsom bytes der indeholder instruktionskoder, for instruktioner, der skal udføres, og operanddata, der er de variable data, der skal anvendes ved udførelsen af disse instruktioner. Både instruktionsudførelsesbe-

regningsenheden og flydende kommaberegningsenheden de-
koder instruktionskoderne. Når EU'en dekode en instruk-
tionskode, der angiver, at den instruktion skal udføres
af EU, udfører den denne instruktion. Når FPAP'en de-
5 koder en instruktion, der skal udføres i FPAP'en, udfører
den den instruktion. Under normale omstændigheder har
FPAP'en ikke ansvaret for udførelse af nogen instruktioner
i en EU instruktionscyklus. Derfor vil FPAP'en, når
den dekode en instruktion, der skal udføres i EU'en,
10 ifølge den foreliggende opfindelse i stedet for at blive
i tomgangen, udføre diagnosefunktion, så længe EU'en
arbejder.

FPAP'en vælger i hvert tilfælde den aktuelle diagnos-
tiske funktion, der skal udføres, ud fra et stort antal
15 tilgængelige diagnostiske funktioner. Valget af diagnos-
tiske funktioner er afhængig af hvilken instruktion,
der skal udføres af EU'en. For ikke at forsinke hele
systemet, vælges der en diagnostisk funktion, hvis ud-
førelsestid er tilpasset udførelsestiden for den in-
20 struktion der udføres af EU'en. Dvs., at der vælges
en diagnostisk funktion, således at FPAP'en vil afslutte
denne funktion, før EU'en afslutter sin funktion. Sy-
stemkonstruktøren må da have et forudgående kendskab
til EU'en funktionsudførelsestider, FPAP diagnosefunk-
25 tionsudførelsestider og må da kortlægge forholdet mellem
EU funktioner og FPAP diagnosefunktioner på en sådan
måde, at dette resultat opnås. Som følge heraf er sam-
menhængen mellem EU funktioner (eller instruktionskoder)
og FPAP diagnosefunktioner et spørgsmål om konstruk-
30 tionsvalg og ikke begrænset af den foreliggende opfindelse.

Da FPAP'en modtager operanddata, ligesåvel som instruk-
tionskoder over instruktionsbussen, anvendes operanddata
i mange af de diagnostiske funktioner, hvorved der til-
føres en grad tilfældighed til afprøvningssignalerne.

5 Dette gør det muligt at detektere fejl, som er afhængige af de værdier, såsom fejl der forårsages af bits, der er tvunget til en fast værdi på nul eller et. Det skal bemærkes, at de operanddata, der på denne måde anvendes, kan udgøre operander der normalt anvendes af EU'en. Selv om disse operander ikke er tænkt som flydende kommatatal, kan de fortolkes af FPAP'en, som dette var tilfældet og kan anvendes i de diagnostiske funktioner som flydende kommatatal. En eller flere variable kan fremkomme fra EU over en skrivebus, eller fra generelt anvendelige registre.

15 På denne måde bliver både FPAP'ens dataveje og styrelogik undersøgt ved udførelsen af hver ikke flydende kommafunktion, hvorved det er muligt hurtigt at detektere en FPAP fejl, og derved gøre det muligt for systemet at deaktivere en defekt FPAP (i mange tilfælde) før at den har haft mulighed for at frembringe upålidelige udgangsdata.

20 Når der udføres en FPAP selv-diagnoserutine som ovenfor beskrevet, og der optræder en fejltilstand, eller den detekteres af fejlrutinen, frembringes der et FPAP fejlsignal. Dette FPAP fejlsignal lagres i FPAP'en indtil EU'en kan behandle dette, hvilket vil sige i det tidsrum, hvor FPAP'en behandler den næste flydende kommatalsberegning. Andre data kan også lagres, såsom en identifikation af den diagnosefunktion, der blev udført da fejlen optrådte, de operanddata som diagnosefunktionen anvendte, og de fejlagtige resultater, der blev frembragt af flydende kommberegningssenheden. Mens den næste flydende kommberegning udføres, rapporteres fejlen til EU'en, 25 der bestemmer årsagen til fejlen, for så vidt at den er i stand til det, hvilket vil sige i det mindste det modul, i hvilket fejlen optræder, identificeres. Hvis årsagen til fejlen er af en sådan art, at "næste" fly-

dende kammaberegning ikke vil blive udført korrekt, kan denne beregning standses. Hvis årsagen til uheldet er isoleret fra og ikke vil påvirke den næste beregning, kan denne fortsætte uforandret.

- 5 Ved at anvende pseudotilfældige operanddata, og til stadighed at udføre diagnoseundersøgelser i "baggrunden" i et langt tidsrum, udføres der flere diagnostiske undersøgelser af FPAP'en, end der ellers ville blive udført, hvis man brugte en af de tidligere kendte fremgangsmåder.
- 10 Den foreliggende opfindelse er også anvendelig ved afprøvning af FPAP'ens konstruktion, idet det er nemmere at skelne mellem transiente fejl og konstruktionsfejl. Dvs., at når en FPAP diagnosefunktion medfører at der
- 15 detekteres en fejl, kan fejlen analyseres med henblik på at afgøre, om den er udtryk for et systematisk problem. En metode til at afgøre dette, er at udføre den samme diagnosefunktion med de samme operanddata på et andet system af samme konstruktion. Hvis den samme fejl optræder,
- 20 kan årsagen antages at være en konstruktionsfejl snarere end en transient optrædende fejl.

Opfindelsen skal nu beskrives nærmere med henvisning til tegningen, af hvilken:

- 25 Fig. 1 er et generelt blokdiagram for et digitalt data-behandlingssystem, der indeholder flere beregningsenheder, såsom en centralberegningseenhed og udgør et eksempel for omgivelserne omkring den foreliggende opfindelse,

- 30 fig. 2 er et blokdiagram der viser opbygningen af den centrale beregningsenhed, hvori der anvendes flere beregningsenheder, såsom en instruktionsberegningseenhed og en flydende kommatalsberegningseenhed, hvori den foreliggende opfindelse er specielt anvendelig,

fig. 3 er en skematisk beskrivelse af en informations byte, der kan overføres til den flydende kommatalsberegningssenhed 34 fra fig. 2,

5 fig. 4 er et blokdiagram af en del af det i fig. 2 viste system, hvor den ændrer opbygning af den flydende kommatalsberegningssenhed, og

fig. 5 er en skematisk illustration af formatet for et flydende kommatalsdataord, der kan behandles af flydende kommatalsberegningssenheden fra fig. 2 og 4.

10 Som det er vist i fig. 1 er de grundlæggende dele i et databehandlingssystem i hvilke opfindelsen anvendes, den centrale behandlingssenhed 10, en lagerenhed 11, og en indgangs/udgangssenhed 12. Beregningssenheden 10 udfører de instruktioner, der er lagret i adresserbare
15 lagerlokationer i lagerenheden 11. Instruktionerne er identiske med operationer der skal udføres på operander, der også er lagret i adresserbare lokationer i lagerenheden. Beregningssenheden 10 henter instruktionerne og operanderne, efterhånden som der er behov for det,
20 og de behandlede data tilbageføres til lagerenheden. Beregningssenheden 10 overfører også styringsinformationer til enheder i indgangs/udgangsdelene, hvorved disse aktiveres til at udføre forudbestemte handlinger, såsom at overføre data til, eller at hente data fra lagerenheden
25 11. Sådanne data kan være instruktioner, eller behandlede data, der hentes fra lagret med henblik på lagring eller udlæsning.

En brugerkonsol 13 tjener som brugerens forbindelse til systemet. Den gør det muligt for brugeren at undersøge og afgive data, stoppe den centrale beregningssenhed
30 10's funktion, eller trinvis at styre den centrale beregningssenhed gennem en sekvens af instruktioner, og kontrol-

lere beregningsenhedens reaktion på disse instruktioner. Det er også muligt for en bruger at initialisere systemet ved hjælp af en "boot strap" procedure. Det er endelig
5 også muligt at udføre forskellige afprøvninger af hele databehandlingssystemet.

Den centrale beregningsenhed 10 er forbundet til lager-
enheden 11 gennem adskillige busser, der her betegnes
med henvisningstallet 14. Den centrale beregningsenhed
10 10 er forbundet direkte til en lagerstyreenhed 15, der
igen er forbundet til et antal lagerenheder 16 over
en enhedsbus 17. I en specifik udførelsesform indeholder
lagerstyreenheden 15 også et "cache" lager. Lagerstyre-
enheden 15 indeholder kredsløb, der er indrettet til
15 at udtage indholdet af adresserede lokationer fra enten
"cachen" eller enhederne 16, og til at lagre informationen
i disse på sædvanlig måde. "Cache"-lagerenheder er vel-
kendte indenfor teknikken og skal ikke beskrives yder-
ligere her.

20 Databehandlingssystemet kan indeholde adskillige arter
af indgangs/udgangsenheder, deriblandt pladelagre, bånd-
stationer, fjernskrivere, tastaturer og skærmterminaler
og lignende. Disse enheder 20 er gennem en indgangs/ud-
gangsbuss 21 forbundet til en busadapter 22. Indgangs/ud-
25 gangsbussen 21 kan være udformet som det er beskrevet
i US patentskrift nr. 4 232 366. Der kan også anvendes
andre arter af indgangs/udgangsbusser til at forbinde
lignende indgangs/udgangsenheder (ikke vist), der inde-
holder en indgangs/udgangsbuss 23, der er forbundet til
30 en busadapter 24, der kan være udformet som det er be-
skrevet i US patentskrift nr. 3 815 099.

Busadapterne 22 og 24 er indrettet til at sende data
til og modtage data fra lagerstyreenheden 15 over en

adapterbus 25. Busadapterne er også forbundet til en
 "interrupt" forespørgsel/givet (request/grant) bus 26,
 over hvilken busadapterne kan afbryde behandlingen i
 den centrale behandlingsenhed 10, i det tilfælde, hvor
 5 der sker en ændring af status for en eller flere af
 indgangs/udgangsenhederne 20, hvilket sker på helt sæd-
 vanlig måde. Den centrale beregningsenhed 10 overfører
 på denne måde "interrupt" forespørgsels/givet signaler
 direkte til enhederne i indgangs/udgangsdelene, og over-
 10 fører styreinformation til og modtager statusinformation
 fra enhederne i indgangs/udgangsdelene 12 gennem lager-
 styreenheden 15. Lagerstyreenheden styrer således over-
 førelsen af data til og fra den centrale beregningsenhed
 10, og indgangs/udgangsdelene 12, og overførelsen af
 15 styre- og statusinformation mellem den centrale bereg-
 ningsenhed og indgangs- og udgangsdelene 12.

Den foreliggende opfindelse omhandler specielt den centrale
 beregningsenhed 10, der er vist mere detaljeret på fig.
 2. Den centrale beregningsenhed 10 indeholder 3 underen-
 20 heder, en instruktions- og operandhenteenhed (IBOX)
 30, en instruktionsberegningenhed (EU) 32 og en flydende
 kommatalseksekveringsenhed (FPAP) 34. IBOX'en 30 overfører
 gennem den virtuelle adresseinstruktionsbus 36 den virtu-
 elle adresse for hver instruktion (eller sekvens af in-
 25 struktioner) til lagerstyreenheden 15, således at disse
 hentes fra lagerenheden 16 og hvert resultat kan skrives
 i lagerenheden 16. IBOX skriver information til lagersty-
 reenheden 15 (der derefter overføres til lagerenheden
 16), og læser information fra lagerstyreenheden 15 gennem
 30 en lagerdatabus 38.

Beregningsenheden 10 er forbundet til lagerstyreenheden
 15 gennem en tredje bus 42. Formålet og virkemåden for
 denne bus er beskrevet yderligere nedenfor. Busserne
 36, 38 og 42 svarer til de med henvisningstal 14 benævnt-

te busser i fig. 1.

Instruktionshenteenheden 30 henter instruktioner der skal udføres i beregningsenheden 10 fra lageret, og de operanddata der skal anvendes ved udførelsen af disse instruktioner. Sådanne instruktions- og operanddata overføres til EU'en 32 og FPAP'en 34 samtidig over en instruktionsbus 44. Informationen, der overføres over instruktionsbussen 44, kan f.eks. være pakket i bytes på et forudbestemt antal bits. I en mulig udførelsesform, der er vist i fig. 3, angiver en enkelt byte 58 både en instruktionskode 54 og data 56 for en første operand "A". Naturligvis kan instruktionskode 54 og operanden 56 ligesåvel angives i særskilte bytes. Operanden 56 kan udgøre en operand der normalt tænkes anvendt af EU'en. Selv om en sådan operand ikke repræsenterer et flydende kommatall, kan den af FPAP'en fortolkes, som om dette var tilfældet, og kan anvendes i en diagnosefunktion på samme måde som et flydende kommatall.

Over instruktionsbussen 44 fremføres der to typer instruktioner. Typisk vil arten af instruktionen angives af en eller flere forudbestemte bits i instruktionskoden. F.eks. kan den første bit i instruktionskoden 54 anvendes til at angive instruktionstypen. Den første type instruktion (hvor bit 58 har værdien 0), angiver at instruktionen skal udføres i beregningsenheden 32, og en anden type instruktion (f.eks. hvor bit 58 har værdien 1), angiver en instruktion, der skal udføres i flydende kommageregningenheden 34. Beregningsenheden 32 og beregningsenheden 34 dekode samtidig instruktionskoderne 54 på bus 44, og hver af disse beregningsenheder vil, når de genkender en instruktion, der er rettet mod den selv, begynde at udføre instruktionen. Resultatet af de instruktioner, der udføres i beregningsenheden 32 og beregningsenheden 34 overføres til IBOX'en 40 gennem

en "Write" bus (WBUS) 62. Resultatet af registerændringer overføres af IBOX'en 30 og EU'en 32 via WBUS 62 til FPAP 34 og IBOX 40.

5 Ifølge den foreliggende opfindelse vil FPAP 34 imidlertid når den detekterer en instruktionskode, for en instruktion der skal udføres af EU 32, vælge en diagnosefunktion, som FPAP'en udfører på sig selv. Valget af diagnosefunktionen afhænger af den specifikke EU instruktion der blev dekodet. Dvs., at hver EU instruktion
10 er parret sammen med en eller flere diagnostiske funktioner, der kræver mindre tid til udførelse af FPAP'en, end den minimale tid, EU'en anvender til at udføre den instruktion, der angives af instruktionskoden. Hvis
15 der kan vælges mere end én diagnostisk funktion til en given instruktionskode, da kan man anvende én eller flere bits i en operand, som retningsgivende for det endelige valg af diagnostisk funktion.

DA FPAP instruktioner sædvanligvis fordrer, at der udøves en aritmetisk operation på to operander, er det nødvendigt, at der findes en anden operandkilde ud over de
20 operander, der fremkommer over instruktionsbussen 44. Der findes to yderligere kilder til operander ud over instruktionsbussen 44. Den anden kilde er WRITE bussen 62, der kan overføre data fra beregningsenheden 32 og
25 instruktionshenteenheden 30. Som en tredje kilde findes der et antal generelle registre 64 i selve den flydende talberegneenhed.

Fig. 4 viser beregningsenheden 10, i hvilken opfindelsen er indeholdt. Den centrale beregningsenhed indeholder
30 en henteenhed 30, og en instruktionsberegneenhed 32, der overfører adresser over busserne 36 og 42 til et ikke vist lager. Henteenheden 30 modtager instruktioner fra lageret over en lagerdatabus 38. Henteenheden dekode

instruktionerne, henter operanderne og overfører dem til beregningsenheden 32, eller til en flydende kommandoberegningssenhed 34, idet disse overførelser sker over en operandbus 44. Beregningsenheden 32 udfører instruktionerne, undtagen de instruktioner, der tilhører den velkendte klasse af flydende kommandoinstruktioner, der udføres af flydende kommandoberegningssenheden 34. Beregningsresultaterne tilbageføres til henteenheden 30 over en WRITE bus 62, med henblik på lagring i lagerenheden over lagerdatabussen 38.

Beregningsenheden 32 er også forbundet til en ikke vist konsol over en konsolbus 13A. Konsollen tjener som brugermellemlid, og gør det muligt for brugeren at undersøge og anbringe instruktioner og operander, at standse den centrale beregningsenhed 10, eller trinvis at styre beregningsenheden gennem en instruktionssekvens og bestemme dens reaktioner på denne. Konsollen gør det også muligt for brugeren at initialisere systemet ved hjælp af en "bootstrap" procedure, og endeligt er det også muligt via konsollen at udføre forskellige diagnostiske undersøgelser på hele databehandlingssystemet.

Beregningsenheden 32 er også forbundet til en "interrupt" forespørgsels/bevilget bus 66, til enheder i et indgangs/udgangssystem (ikke vist), der kan omfatte skrivere, fjernskrivere, tastaturer og skærmterminaler, ligesåvel som pladelagre og båndstationer. Bussen 66 overfører "interrupt" forespørgselssignaler fra indgangs/udgangsenhederne til beregningsenheden 32, og "interrupt" bevilget signaler fra overførelsesenheden til indgangs/udgangsenhederne.

Som det er nævnt tidligere behandler flydende kommandoberegningssenheden 34 flydende kommandoinstruktioner. Disse instruktioner gør det muligt for beregningsenheden at

udføre instruktioner på operander, der er formatteret med flydende komma, hvilket er vist i fig. 5. En flydende komma operand indeholder en eksponentdel 68 og en decimalbrøk 70, der hver indeholder en fortegnsbid 72 og 74, der angiver fortegnet for de respektive dele 68 og 70. Flydende kommainstruktioner er sædvanligvis begrænset til aritmetiske instruktioner, såsom addition, subtraktion, multiplikation og division eller variationer af disse instruktioner. Flydende kommatalsberegningseenheden 34 indeholder to decimalbrøksbehandlingsveje, den ene, et additionsmodul 76, udfører addition og subtraktion, og den anden, et multiplikationsmodul 78, udfører multiplikationer og divisioner på operandens brøkdelt 70. Et eksponentbehandlingsmodul 80 behandler eksponentdelen 63 i flydende kommatalsoperanden ved en hver flydende kommatalsoperation.

Behandlingsmodulerne 76, 78 og 80 modtager flydende kommatalsoperander fra et antal kilder. En af disse kilder er operandbussen 44, og data modtages fra et operandbuslager 82. Andre kilder for operander er WRITE bussen 62, fra hvilken data modtages fra et WRITE buslager 84. Den tredje kilde for operander til behandlingslogikken 76, 78 og 80, er et antal generelle registre 64, der er indeholdt i flydende kommatalsberegningseenheden 34. Disse generelle registre 64 opdateres gennem WRITE buslageret 84, af beregningseenheden 32.

Behandlingsmodulerne 76, 78 og 80 styres af mikroinstruktioner, der frembringes af et mikrokodestyrelager 86, hvorfra de vælges af en mikrosekvensstyreenhed 88 på sædvanlig måde.

Derudover styres mikrokodeseqvensstyreenheden 88 af et flydende kommatalsdiagnoselager (der er et læselager med tilfældig adgang) (FDRAM) 90, hvis funktion er som

følger. Et instruktionskodelager 92 overvåger de instruktionskoder, der overføres over instruktionsbussen 44, og lagrer de instruktioner, der fremkommer på denne. I afhængighed af værdien af instruktionskoden i lageret 5 92, overfører FDRAM'en tilsvarende forudbestemt adresse til FDRAM lageret 94. Herfra overføres denne adresse til mikroinstruktionsstyrekredsløbet 88, som starter adressen fra en sekvens af mikroinstruktioner. Når instruktionskoden i lageret 92 angiver en EU instruktion, 10 der af den adresse der frembringes af FDRAM'en 90 starten på FPAP diagnosesekvens.

En typisk FPAP diagnosesekvens der udføres på denne måde kan f.eks være udførelsen af beregningen AB-BA, og derefter følgende kontrol af om resultatet er nul. Ovenfor 15 angiver "A" en første variabel og "B" en anden variabel. Værdien for én af variablerne, som "A", kan som det er angivet ovenfor være en operandværdi fra instruktionsbuslageret 82. Værdien for den anden variabel (i dette tilfælde "B") kommer fra WBUS 62 eller fra GPR'erne 20 64.

Når en FPAP selv-diagnoserutine er udført, som det er beskrevet her, eller der optræder en fejltilstand, eller denne opdages af fejlrutinen, frembringes der et FPAP fejlsignal. Dette FPAP fejlsignal lagres i FPAP'en til 25 EU'en kan behandle det, hvilket vil være, når FPAP'en behandler den næste flydende kommainstruktion. Andre data kan også lagres, såsom identifikation af den diagnosesekvens der blev udført, da fejlen optrådte, de operanddata, der anvendtes ved diagnosefunktionen, og fejlagtige 30 resultater, der blev frembragt af flydende kommatalsberegningseenheden. Mens den næste flydende kommatalsinstruktion udføres, rapporteres fejlen til systemets "fejllog", der er af den type, der sædvanligvis vedligeholdes af databehandlingssystemets operativsystem. Før resultatet

af det næste flydende kommatalsberegning anvendes, vil EU'en, ved at anvende sædvanlige metoder til analyse af flydende kommatalsfejl, bestemme årsagen til fejltilstanden, for så vidt at det er muligt for beregningsenheden 32 at gøre det, idet man dog i det mindste vil identificere det modul i hvilket fejlen er opstået. Hvis årsagen til fejlen er af en sådan art, at den "næste" flydende kommatalsinstruktion ikke vil blive udført korrekt, kan denne instruktion standses. Hvis årsagen til fejlen er isoleret fra og ikke vil påvirke den næste instruktion, kan denne fortsætte og ved afslutningen returnere sine resultater. Hvis f.eks. fejlen kan bestemmes til at hidrøre fra brøkdelsadditionsmodulet 76, men dette modul ikke anvendes ved udførelsen af den "næste" instruktion, er der ikke nogen grund til at afbryde denne "næste" instruktion. Hvis instruktionen afbrydes, kan den udføres igen, eller den kan i stedet udføres i EU'en, da EU'en kan udføre flydende kommatalsberegninger, men dog ikke så hurtigt som FPAP'en. Nogle gange vil årsagen til fejlen naturligvis være transient, og en ny udførelse af instruktionen vil kunne gennemføres korrekt. Derfor kræver de yderligere handlinger, der sker efter detektion af en fejl, analyse af fejlen for at bestemme, om denne er permanent eller ej. Hvis fejlen ikke er permanent, eller ikke er særlig hyppig, er det ikke nødvendigt at foretage sig yderligere. Hvis imidlertid fejlen optræder for hyppigt, kan det være nødvendigt at lade beregningsenheden reparere.

P a t e n t k r a v :

1. Databehandlingssystem med to beregningsenheder, hvilket databehandlingssystem er indrettet sådan, at en første beregningsenhed (34) er tilnærmelsesvis i "tomgang", når den anden beregningsenhed (32) udfører instruktioner, k e n d e t e g n e t ved, at den første beregningsenhed indeholder

a. organer (92), der er indrettet til at detektere begyndelsen af et tidsinterval, hvor den anden beregningsenhed vil udføre en instruktion, og hvor den første beregningsenhed vil være i "tomgang", og

b. organer (90, 94), der er indrettet til at vælge en diagnostisk operation til udførelse i den første beregningsenhed, mens den anden beregningsenhed udfører instruktionen, idet den diagnostiske funktion vælges blandt et antal tilgængelige diagnostiske operationer på en sådan måde, at den første beregningsenhed vil have afsluttet behandlingen af den diagnostiske operation, inden den anden beregningsenhed afslutter behandlingen af nævnte instruktion.

2. Databehandlingssystem ifølge krav 1, k e n d e t e g n e t ved, at det indeholder organer (62), der er indrettet til at tilføre den første og den anden beregningsenhed de samme operanddata, der kan anvendes af den første beregningsenhed til udøvelse af diagnostiske operationer.

3. Databehandlingssystem ifølge krav 2, k e n d e t e g n e t ved, at den første beregningsenhed er en flydende kommaberegningsenhed.

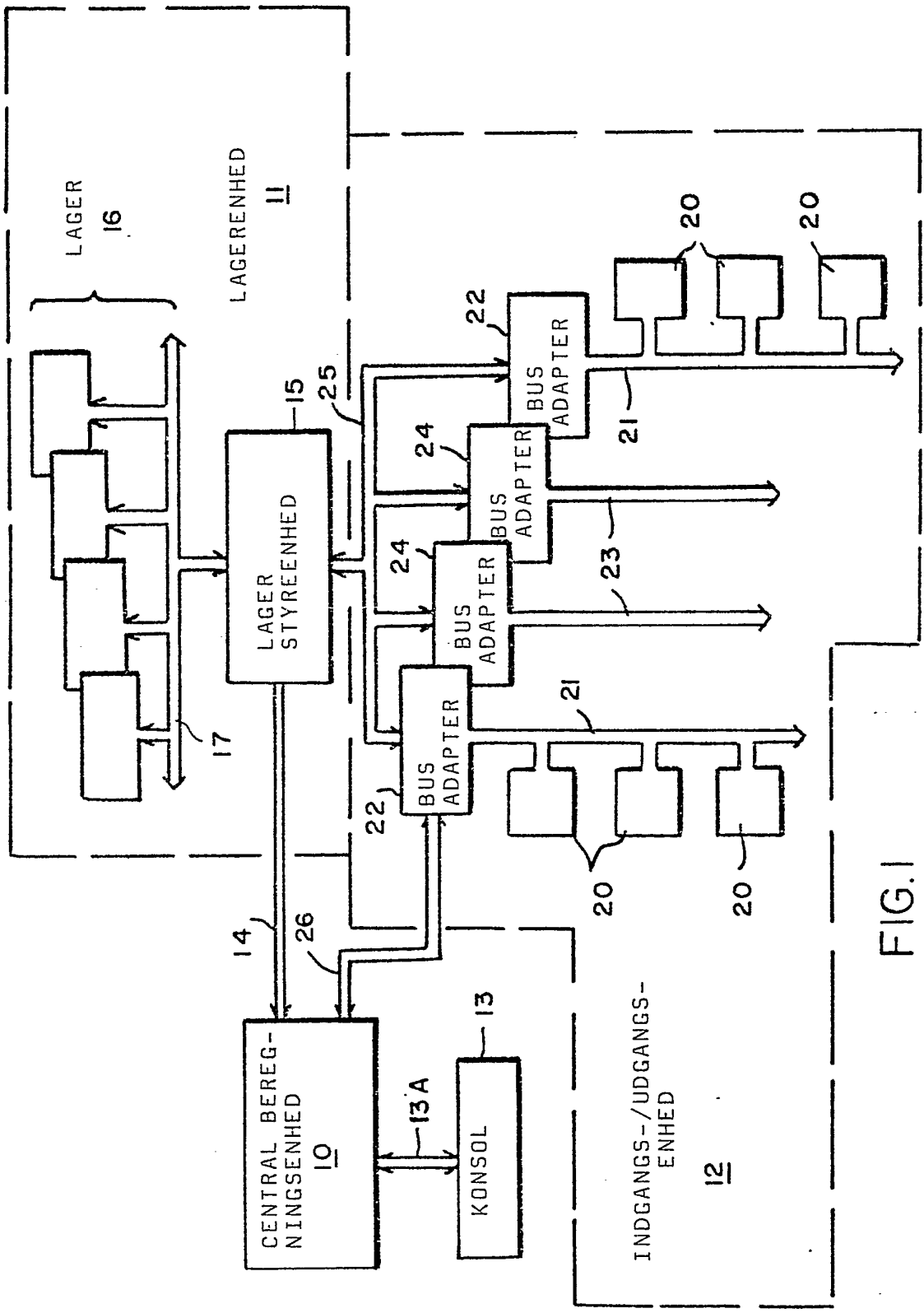
4. Databehandlingssystem ifølge krav 1, k e n d e t e g -
n e t ved, at det indeholder organer (62), der er ind-
rettet til at tilføre begge processorer de samme instruk-
tionskoder, og hvor de organer (92), der er indrettet
5 til at detektere begyndelsen af et tidsinterval, hvor
den anden beregningsenhed (32) udfører en instruktion,
og i hvilket den første beregningsenhed (34) vil være
i "tomgang", indbefatter organer, der er indrettet til
at dekode instruktionskoderne og identificere de instruk-
10 tioner, der skal udføres af den anden beregningsenhed.
5. Databehandlingssystem ifølge krav 4, k e n d e -
t e g n e t ved, at organerne (90, 94) til at vælge
en diagnoseoperation, der skal udføres i den første
beregningenhed, medens den anden beregningsenhed udfører
15 en instruktion, er indrettet til at vælge en sådan diag-
nostisk operation i afhængighed af instruktionskoden
for den instruktion, der skal udføres af den anden be-
regningenhed.
6. Databehandlingssystem ifølge ethvert af kravene 1 - 5,
20 k e n d e t e g n e t ved, at de diagnostiske operationer
er indrettet til at undersøge funktionen af den første
beregningenheds styrelogik og dataveje.
7. Databehandlingssystem ifølge krav 1, k e n d e t e g -
n e t ved, at den første beregningsenhed (34) er en
25 flydende komma talberegningenhed, der er indrettet
til at udføre ikke-flydende komma talberegninger, hvilket
databehandlingssystem endvidere omfatter en operandbus
(44), der er indrettet til samtidigt til begge beregnings-
enheder at overføre instruktioner omfattende instruk-
30 tionskoder og operander for operationer, der skal ud-
føres af disse beregningsenheder, hvor hver instruktions-
kode enten angiver en operation, der skal udføres af
den første beregningsenhed, eller en operation, der

skal udføres af den anden beregningsenhed (32), og hvori detekteringsorganerne (92) omfatter organer, der er indrettet til at dekode de instruktionskoder, der fremkommer over operandbussen, og organer, der i afhængighed af dekodningsorganerne ved detektion af en instruktionskode for en ikke-flydende komma tidsoperation initierer den valgte diagnoseoperation, der skal udføres i den første beregningsenhed.

8. Selvaufprøvningsapparat ifølge krav 7, k e n d e - t e g n e t ved, at udførelsestiden for den valgte diagnostiske operation, der initieres i afhængighed af detektionen af den nævnte instruktionskode, er kortere end den anden beregningsenheds (32) udførelsestid for udførelsen af den til instruktionskoden svarende instruktion.

9. Selvaufprøvningsapparat ifølge krav 7, k e n d e - t e g n e t ved, at detektionen af en fejl i den første beregningsenhed (34), efter udførelsen af en selvaufprøvningsoperation, ikke meddeles til den anden beregningsenhed (32), før den næstfølgende flydende kommatales instruktion udføres af den første beregningsenhed.

10. Selvaufprøvningsapparat ifølge krav 7 eller krav 9, k e n d e t e g n e t ved, at den første beregningsenhed fortolker ikke-flydende kommataloperander, der overføres til den anden beregningsenhed over den nævnte operandbus, som flydende kommataloperander, og hvor sådanne operander anvendes som afprøvningsvariabler i de nævnte diagnostiske operationer.



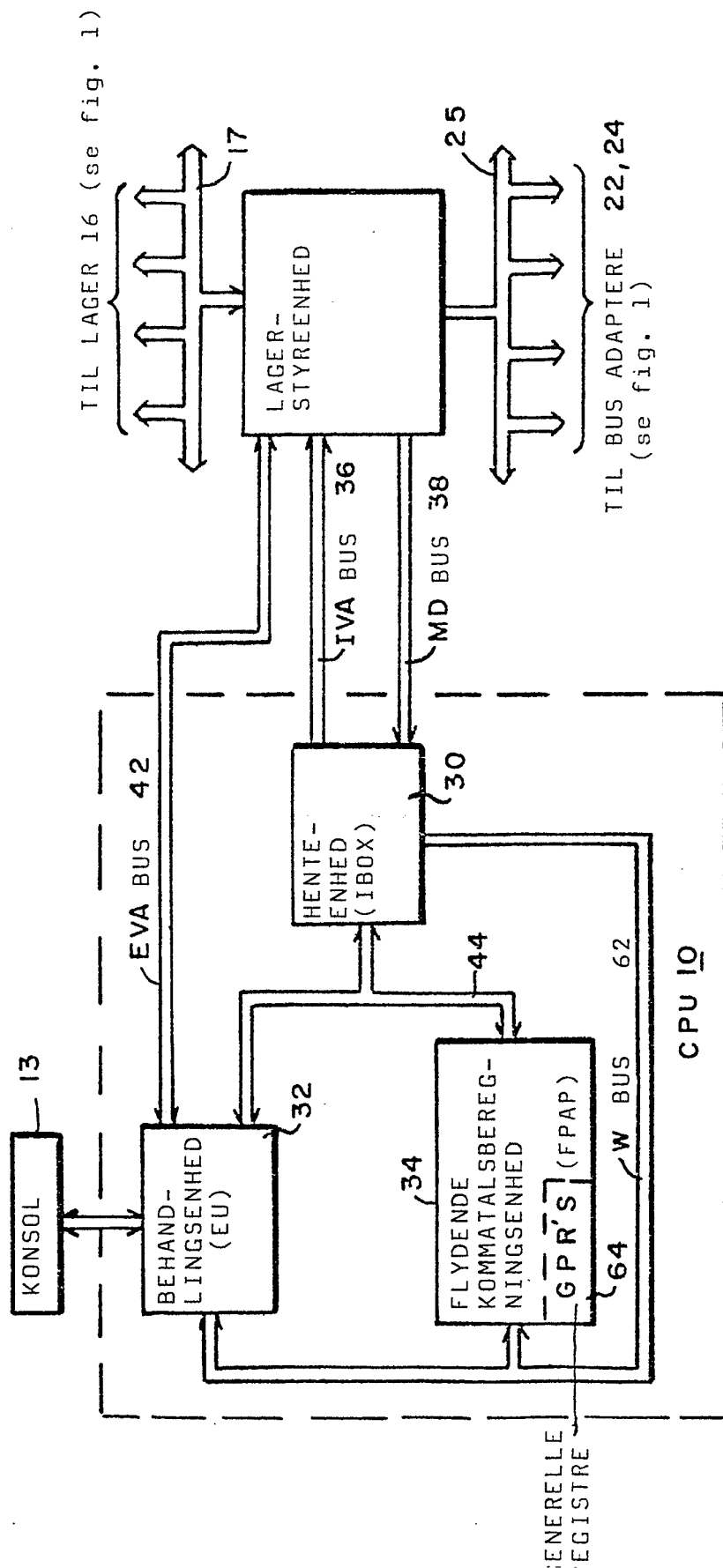


FIG. 2

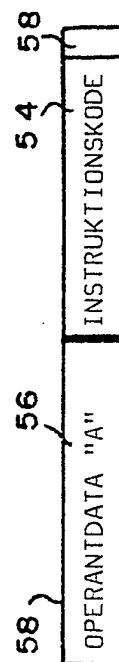


FIG. 3

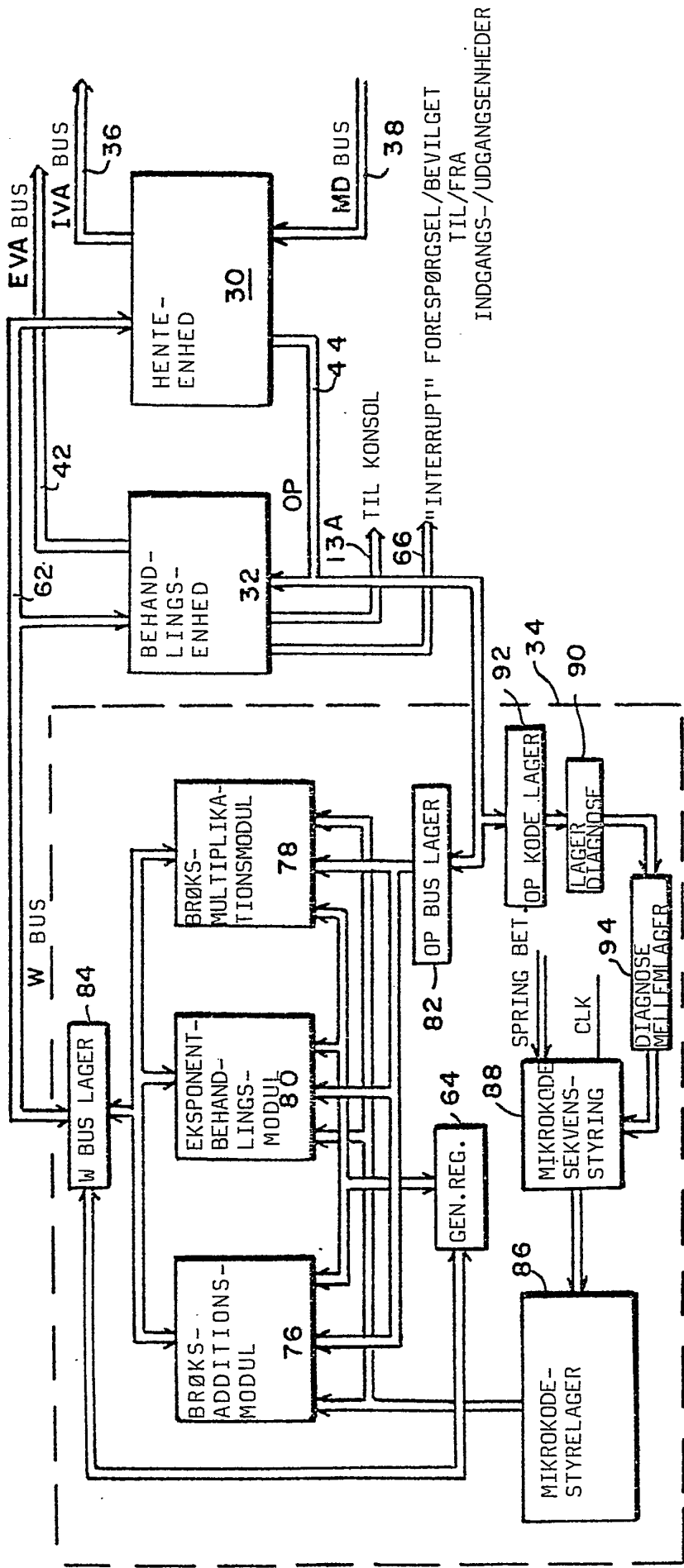


FIG.4

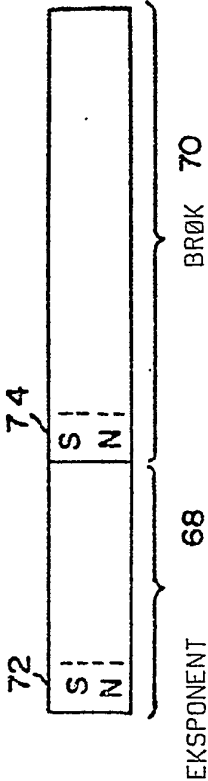


FIG.5