



NORGE

(12) **UTLEGNINGSSKRIFT**

(19) NO

(11) 176119

(13) B

(51) Int Cl⁵ G 06 F 15/50

Styret for det industrielle rettsvern

(21) Søknadsnr	881557	(86) Int. inng. dag og søknadsnummer	
(22) Inng. dag	11.04.88	(85) Videreføringdag	
(24) Løpedag	11.04.88	(30) Prioritet	14.04.87, US, 38253
(41) Alm. tilgj.	17.10.88		
(44) Utlegningsdato	24.10.94		

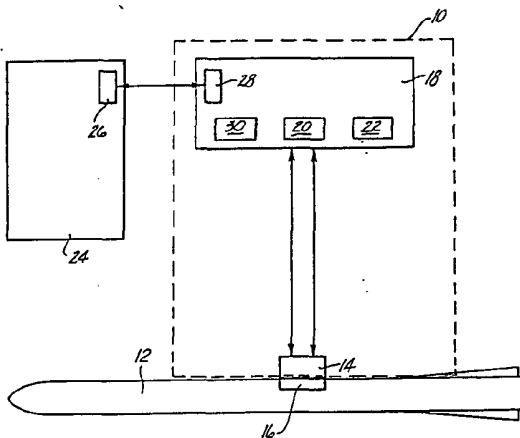
(71) Patentsøker	Hughes Aircraft Co, P.O. Box 80028, Los Angeles, CA 90080-0028, US
(72) Oppfinner	Jeffrey Lowenson, Chatsworth, CA, US Khosrow Hessamian, Chatsworth, CA, US
(74) Fullmektig	Jens F.C. Langfeldt, Bryns Patentkontor AS, Oslo

(54) Benevnelse **Anordning for testing av missil-systemer**

(56) Anførte publikasjoner **Mesures, Regulations, Automatisation, vol. 50, nr.1,jan. 1985, s.57,59,60, Paris, FR, "Sim. au sol de missiles, la rapidité avant tout".**

(57) Sammendrag

Anordning (10) for å teste missilprogramvare, bestående av en missildataprosessor grensesnittkrets (14) for å tilpasse dataprosessoren (16) for missilet med anordningen (10). Anordningen (10) omfatter dessuten en krets (18) for reell-tids innsamling av informasjon som fremkommer på minst en av hukommelsesbussene hos missil-dataprosessoren. Kretsen (18) for reell-tids innsamling av data er i elektrisk forbindelse med dataprosessorens grensesnittkrets (14) og en datamaskin (24).



Denne oppfinnelse vedrører missil-styrings og lednings-systemer, og nærmere bestemt en anordning for å teste missil-systemer, der nevnte anordning, som innbefatter en dataprosessor-grensesnittbuffer, står i elektrisk forbindelse med en operandhukommelse og en programhukommelse for en missil dataprosessor gjennom en flerhet av hukommelsesbusser, idet nevnte anordning dessuten står i elektrisk forbindelse med en datamaskin.

En anordning av denne type fremgår bl.a. av MESURES, REGULATIONS, AUTOMATISME, vol. 50, nr. 1, januar 1985, sidene 57, 59, 60; Paris, FR; "Simulation au sol de missiles: La rapidité avant tout".

Et område av særlig betydning innenfor missil-teknologien er den som vedrører missil-ledingssystemer. Disse systemer anvendes for å styre retningen av missiler under flukt. Visse ledingssystemer avføler kontrasten i utsendt eller reflektert stråling mellom målet og omgivelsene for å styre banen. Disse typer av ledingssystemer anvendes ofte når målene enten er fly eller skip. Andre typer av ledingssystemer anvender forutbestemt geografisk- eller himmelinformasjon til å lede missiler. I kommandoledingssystemer vil en bakkestasjon eller et fly følge både missilet og målet for derved å være i stand til å generere og sende de passende instruksjoner til å dirigere missilet mot målet. Andre typer av ledingssystemer, slik som strålebanestyrer-ledingssystemer, blir ofte anvendt.

I visse av de siste taktiske missiler, slik som AMRAAM og Phoenix, befinner samtlige midler for å følge et manøvrerende mål og å lede missilet mot målet seg innenfor selve missilet. Således vil en kampflyver typisk innfange et mål, avfyre et missil og så være fri til å konsentrere seg om andre mål. "Smarte" missiler, som beskrevet ovenfor, anvender en eller flere dataprosessorer (i det etterfølgende "ADP") plassert innenfor et missil til å behandle informasjon som er relatert

til missil og målbaner. Hver ADP utfører en lagret samling av programmer som setter nevnte ADP istand til å sende de passende styresignaler til forskjellige aktuatorer som styrer missilets flukt. Nevnte ADP'er blir vanligvis anvendt i forbindelse med en programhukommelse og en operandhukommelse. Programhukommelsen anvendes til å lagre instruksjonene som utføres av nevnte ADP, mens operandhukommelsen anvendes til å lagre dataene som instruksjonene i programhukommelsen utfører. Nevnte ADP kommuniserer med operandhukommelsen gjennom en operandhukommelse-databuss og en operandhukommelse-adressebuss. Likeledes står nevnte ADP i forbindelse med programhukommelsen gjennom en programhukommelse-adressebuss og en programhukommelse-databuss. På grunn av at missilledingsystemer ofte krever en relativt sofistikert analyse av data, er programvaren som anvendes i forbindelse med nevnte ADP ofte relativt komplisert og kan involvere en relativt lang fremkallingstid. En måte å redusere fremkallingstiden for nevnte ADP programvare er å anvende programvareemuleringssystemer. Slike programvareemuleringssystemer emulerer generelt aktiviteten for nevnte ADP slik at mellomliggende eller sluttresultater kan sammenlignes med de forventede resultater. Ved å sammenligne de faktiske resultater med de forventede resultater, er det mulig å bestemme hvorvidt programvaren fungerer på riktig måte.

Selvom slike programvare-emuleringssystemer reduserte utviklingstiden for ADP programvare, ga de ofte ikke flere egenskaper som ville ytterligere fremskynde ADP programvareutviklingen. Eksempelvis tillot disse systemer ikke reell-tidssamling innsamling av informasjon som fremkom i de forskjellige adresse og databussene. I tillegg tillot programvare-emuleringssystemer ikke generelt innholdet i programhukommelsen å bli dumpet på en ekstern hukommelse slik at innholdet i programhukommelsen kunne granskes. Dessuten, på grunn av at slike programvareemulerende systemer ikke generelt tillot in situ testing av ADP programvaren, var det

ikke ofte mulig å bestemme hvorvidt ADP 's program eller operandhukommelse ble skadet under installasjon. Til sist kunne slike systemer ikke anvendes for testing av opererbarheten hos programvaren som ble styrt av nevnte ADP.

5

Den innledningsvis nevnte anordning kjennetegnes, ifølge oppfinnelsen ved:

10

et eksternt direktelager som elektrisk står i forbindelse med nevnte programhukommelse,

15

en logisk grensesnittenheter for innsamling av data fra nevnte operandhukommelse og nevnte programhukommelse i reell-tid, idet nevnte logiske grensesnittenheter omfatter et direktelager som elektrisk står i forbindelse med nevnte dataprosessor grensesnittbuffer, og idet nevnte direktelager kan bevirkes til å lagre informasjon som fremkommer på en av nevnte hukommelsesbusser når dataprosessoren utfører en skrivekommando,

20

en dataprosessorsimulator som er i stand til å simulere utmatningen fra nevnte dataprosessor, idet nevnte dataprosessor simulator elektrisk står i forbindelse med nevnte logiske grensesnittenheter og nevnte eksterne direktelager, og

25

en datamaskin grensesnittmodul som er i stand til å tilpasse nevnte logiske grensesnittenheter og nevnte eksterne direktelager med nevnte datamaskin.

30

Ifølge ytterligere utførelsesformer av anordningen omfatter nevnte dataprosessor grensesnittbuffer en flershet av oversettere som kan bevirkes til å omdanne de logiske nivåer som er knyttet til nevnte dataprosessor til logiske nivåer som er tilknyttet nevnte eksterne direktelager og nevnte logiske grensesnittenheter.

35

Dessuten kan anordningen omfatte en første multiplekser som kan bevirkes til å multiplekse informasjon fra nevnte dataprosessor og nevnte dataprosessorsimulator, idet nevnte første multiplekser elektrisk står i forbindelse med nevnte

logiske grensesnittenhet og nevnte eksterne direktelager. I tillegg kan anordningen dessuten omfatte en andre multiplekser som kan bevirkes til å multiplekse informasjon fra nevnte dataprosessor og nevnte dataprosessorsimulator, idet 5 nevnte andre multiplekser elektrisk står i forbindelse med nevnte logiske grensesnittenhet. Den logiske grensesnittenheten er innrettet til å bevirke nevnte datamaskin til å ta pause når informasjon som fremkommer på en av nevnte hukommelsesbusser er av en forutbestemt verdi. Videre kan 10 den logiske grensesnittenheten bevirke telling av antallet av opptredener hvor den informasjon som fremkommer på en av nevnte hukommelsesbusser er av en forutbestemt verdi, låsing av informasjon som fremkommer på en av nevnte hukommelsesbusser, og sporing av informasjon som fremkommer på en av 15 nevnte hukommelsesbusser.

Ved ennu ytterligere utførelsesformer av anordningen kan nevnte logiske grensesnittenhet dessuten omfatte en første FIFU hukommelse som selektivt kan bevirkes til kontinuerlig å 20 motta informasjon som fremkommer på en av nevnte hukommelsesbusser. Særlig kan det være fordelaktig at den første FIFU hukommelsen selektivt kan bevirkes til å motta informasjon som fremkommer på en av nevnte hukommelsesbusser når informasjon som fremkommer på en av nevnte busser er av en 25 forutbestemt verdi.

I en annen utførelsesform av anordningen kan den omfatte en andre FIFU hukommelse som selektivt kan bevirkes til kontinuerlig å motta informasjon som fremkommer på en av nevnte 30 hukommelsesbusser. Særlig kan det være fordelaktig at den andre FIFU hukommelsen kan bevirkes til selektivt å motta informasjon som fremkommer på en av nevnte hukommelsesbusser når informasjon som fremkommer på en av nevnte hukommelsesbusser har en forutbestemt verdi.

35

Anordningen, ifølge oppfinnelsen, tillater dessuten at

innholdet i nevnte programhukommelse kan bli dumpet på nevnte eksterne direktelager.

5 Videre foreligger muligheten for at den logiske grensesnitt-enheten kan bevirkes til å levere informasjon til nevnte dataprosessor, slik at maskinvaren som drives av nevnte dataprosessor kan testes.

10 Forskjellige fordeler ved den foreliggende oppfinnelse vil fremtre for en fagmann ved lesing av den etterfølgende beskrivelse og med henvisning til de etterfølgende tegninger.

15 Fig. 1 er en skjematisk illustrasjon av anordningen for testing av missilsystemer som er gjort i henhold til læren ifølge den foretrukne utførelsesform av den foreliggende oppfinnelse vist i operativt forhold med et missil og en datamaskin.

20 Fig. 2 er en skjematisk illustrasjon av anordningen for testing av systemer vist i fig. 1.

25 Fig. 3-4 er skjematiske illustrasjoner av den logiske grensesnitt-enheten som er vist i fig. 2.

30 Fig. 5-8 er skjematiske illustrasjoner over fremgangsmåten for testing av missilsystemer i henhold til læren ifølge den foretrukne utførelsesform av foreliggende oppfinnelse og som viser bibliotekrutinene som kan anvendes til å skrive testprogrammer.

Idet der vises til fig. 1 er en anordning 10 for testing av missil-systemer vist. Anordningen 10 står elektrisk i forbindelse med et missil 12 gjennom grensesnittbufferen 14 som tilveiebringer middel for å tilpasse missilet til anordningen 10. Tilpasningsbufferen 14 står i forbindelse med en dataprosessor (det etterfølgende ADP") 16 hos missilet 12

in situ gjennom operandhukommelse-data (i det etterfølgende "OMD") bussen, operandhukommelse-adresse (i det etterfølgende "OMA" bussen, programhukommelse-data (i det etterfølgende "PMD") bussen og programhukommelse-adresse (i det etterfølgende "PMA" bussen for missilet 12. Grensesnittbufferen 14 anvendes til å omsette de logiske verdier som er knyttet til nevnte ADP 16 til de som kan anvendes av de andre komponentene for anordningen 10 som er beskrevet nedenfor. For å tilveiebringe middel for reelltidssinnsamling av data, omfatter anordningen 10 en logisk grensesnittet/esternt direktelager (i det etterfølgende "LIU/XRAM) 18. Nevnte LIU/XRAM 18 mottar reelltidssinformasjon fra missilet 12 gjennom grensesnittbufferen 14. Nevnte LIU/XRAM 18 omfatter et eksternt direktelager (heretter "XRAM") 20 som kan anvendes til å lagre et program som skal testes på ADP 16 eller på en ADP simulator som er beskrevet nedenfor. Nevnte XRAM 20 kan omfatte en 64K x 24 bit hukommelseoppstilling som anvender 96 1MS1400 - 55 hukommelsesanordninger, selvom det skal forstås at andre passende hukommelseslagringsanordninger kan anvendes.

Nevnte LIU/XRAM 18 omfatter også en ADP simulator 22. ADP simulatoren 22 anvendes til å simulere responsen hos nevnte ADP 16 slik at programmer som er lagret i XRAM 20 kan testes under anvendelse av ADP simulatoren 22 uten å behøve at nevnte ADP 16 er tilstede. ADP simulatoren 22 omfatter en 8 faseklokke (ikke vist) som genererer CØ0 - CØ3 og CT0 - CT3 sekvensmessige klokkesignaler som simulerer CØ0 - CØ3 og CT0 - CT3 klokkesignalene som genereres av ADP 16. I tillegg omfatter ADP simulatoren 22 en tidsstyrende logisk krets (ikke vist) som anvender utmatningen fra 8-fase klokken til å generere les/skriv signaler som simulerer operandhukommelse les/skriv signalene som genereres av ADP 16. Til sist omfatter ADP simulatoren 22 også en flerhet av tellere (ikke vist) som anvendes for å generere utmatninger som simulerer utmatningene fra OMA, OMD og PMA bussene.

For å kontrollere operasjonen av nevnte LIU/XRAM 18, er en datamaskin 24 tilveiebragt. Datamaskinen 24 innbefatter en LIU grensesnittmodul 26 som elektrisk står i forbindelse med en datamaskingrensesnittmodul 28 som er plassert innenfor

5 nevnte LIU/XRAM 18. Datamaskingrensesnittmodulen 28 tilveiebringer middel for å tilpasse nevnte LIU/XRAM 18 og datamaskinen 24 til å tillate nevnte LIU/XRAM 18 og datamaskinen til å kommunisere. Selvom datamaskinen 24 kan være en PDP 11/44 og anvende en bestemt programvare, vil det

10 forstås at andre passende datamaskiner og programvare kan anvendes. Datamaskinen 24 genererer forskjellige styresignaler som bestemmer hvilken informasjon nevnte LIU/XRAM 18 skal samle fra missilet 12. Disse styresignaler leveres av datamaskinen 24 til LIU/XRAM 18 og dekodes av en styrepulsdekodingskrets som er beskrevet nedenfor til forskjellige

15 register styrelogikk (heretter "RLS") styresignaler. Disse RLS styresignaler anvendes så av nevnte LIU/XRAM 18 til å identifisere hvilken informasjon som skal innsamles fra missilet 12. Disse RLS styresignaler og deres respektive

20 funksjoner er gitt i nedenstående tabell:

RLS STYRESIGNALER

<u>RLS</u>	<u>FUNKSJON</u>
0	INGEN OPERASJON
25 1	NULLSTILL LIU
2	INGEN OPERASJON
3	ENKELT TRINN
4	START ADP (LESTART)
5	STOPP ADP (LEPAUSE)
30 6	INGEN OPERASJON
7	LAST 271CR REGISTER
8	AVBRYT
9	LASTE 211CR REG
10	TILBAKESTILL TILPASNINGSTIDSUR & TELLER
35 11	LASTE TILPASNINGSTELLERREGISTER
12	LASTE DMA STARTADRESSE
13	LASTE DMA SLUTTADRESSE

	14	STROBE DMA TELLER
	15	LASTE 212CR REGISTER
	16	LASTE 213CR REGISTER
	17	TILBAKESTILL DAS SEKVENSDANNER
5	18	TILBAKESTILL TALLKRETSFLAGG
	19	INGEN OPERASJON
	20	LASTE PAUSEADRESSE
	21	LASTE HOLDEKRETSADRESSE
	22	LASTE TILPASNING/SPORE ADRESSE
10	23	LASTE MASKEREGISTER
	24	INGEN OPERASJON
	25	LASTE 233CR REGISTER (TRACE CTL)
	26	LASTE 231CR REGISTER (OMA/PMA, R/W)
	27	LASTE 232CR REGISTER (TRADE SEL)
15	28	TILBAKESTILL SPOREHUKOMMELSE OG GÅ
	29-31	IKKE ANVENDT
	32	LESEVALG FRA LIU/XRAM
	33	LASTE 351CR REGISTER (WRAPAROUND)
	34	NULLSTILL ADP OG XRAM (LEMR)
20	35	IKKE ANVENDT
	36	IKKE ANVENDT
	37	IKKE ANVENDT
	38	IKKE ANVENDT
	39	IKKE ANVENDT
25	40	GÅ INN I DATAMASKINLESEMODUS
	41	LASTE XRAM ADRESSE
	42	LASTE HØYE RAM DATA (PMD 16-23)
	43	LASTE LAVE XRAM DATA
	44	SKRIV TIL XRAM
30	45	DUMP ADP PROG HUKOMMELSE TIL XRAM
	46	ENKELTTRINN ADRESSETELLER OM DUMP
	47	FORLAT DATAMASKIN LESEMODUS
	48-55	IKKE ANVENDT
	56	LASTE FREKVENSKODE 1
35	57	LASTE FREKVENSKODE 2

Visse av RLS styresignalene anvendes for å laste styre-
registre på busser innenfor nevnte LIU/XRAM 18. Disse
registre, sammen med funksjonene som oppstår når visse biter
i et gitt register er av en viss verdi, er gitt i tabellen
5 som fremkommer ved slutten av beskrivelsen.

Nevnte LIU/XRAM 18 omfatter dessuten en logisk grensesnitt-
enhet (i det etterfølgende "LIU") 30 som tillater innsamling
av data fra enten ADP 16 eller ADP simulatoren 22 som, på
10 grunn av hastigheten som informasjon leveres i reell-tid fra
ADP 16 og ADP simulator 22, ikke kunne innsamles av data-
maskinen 24 uavhengig. Nevnte LIU 30 utøver styring over
både nevnte ADP 16 og nevnte ADP simulator 22 for derved å
tillate datamaskinen 24 å innsamle informasjon fra OMA, OMD
15 eller PMA bussene under anvendelse av hvilke som helst av
fire innsamlingsmodi. I pausemodusen genererer nevnte LIU 30
et styresignal som bevirker nevnte ADP 16 til å ta pause når
en bedtemt verdi fremkommer på OMA, OMD eller PMA bussene. I
tilpasningsmodusen teller nevnte LIU 30 antallet av
20 opptredener som en bestemt verdi fremkommer på enten OMA
eller PMA bussene. I holdemodusen, vil LIU 30 holde
informasjon som fremkommer på OMD bussen når en bestemt verdi
fremkommer på den ene eller andre av OMA eller PMA bussene.
Tilsist, når nevnte LIU 30 er i sporemodusen, innsamler LIU
25 30 det som fremkommer på enten PMA, OMA eller OMD bussene som
opptrer før og etter en viss hendelse, slik som når en
bestemt verdi fremkommer på OMA bussen. I tillegg til det
ovenstående tillater nevnte LIU 30 innsamlingen av data når
en kombinasjon av ovenstående modi anvendes. Eksempelvis kan
30 kan holdemodusen og pausemodusen anvendes til å bestemme den
medgåtte tid mellom to hendelser.

Som vist i fig. 2 mottar grensesnittbufferen 14 utmatninger
fra PMA, OMA og OMD bussene. Kun for denne presentasjon
35 formål vil programhukommelse-adressebussen og program-
hukommelse-databussen bli referert til som PMA og PMD busser
uansett hvorvidt signalene har passert gjennom oversetteren,

mottakeren etc. Likeledes vil operandhukommelse-adresse-bussen og operanddatabussen bli referert til som program OMA og OMD busser respektive uansett hvorvidt informasjonen leveres gjennom en oversetter, mottaker etc. Denne konvensjon vil også bli anvendt på databussene i LIU/XRAM 18, samt på utmatningen fra de forskjellige komponenter i LIU/XRAM 18. Grensesnittbufferen 18 mottar dessuten tidsstyringssignaler fra ADP 16. Disse signaler omfatter to av de åtte sekvensmessige klokkesignaler CØ0 og CT3, og les/skriv signalene CDBOMEN og COMWREN.

På grunn av at visse av komponentene i nevnte LIU/XRAM 18 anvender emitterstyrt logikk (heretter "ECL") som er naturlig hurtig, men frembringer mindre støy enn Schottky-innretninger, omfatter grensesnittbufferen 14 en flerhet av oversettere 32-44. Oversetterne 32-44 anvendes for å omforme de logiske nivåer som er knyttet til utmatningene fra nevnte ADP 16 til de som kan anvendes av nevnte LIU/XRAM 18. Oversetteren 32 mottar informasjon fra PMD bussen og leverer denne informasjon til nevnte XRAM 20 gjennom en oversetter 46 når en dumpeoperasjon opptrer. Utmatningen fra oversetteren 32 som leveres til nevnte XRAM 20 er et 24-biters ord som er representert som LPM-23. I tillegg mottar oversetteren 34 24-biters ord LPXM 0-23 fra nevnte XRAM 20 gjennom en oversetter 48 som så leveres til PMD gjennom ECL driveren 50. Driveren 50 anvendes for å styre når informasjonen som befinner seg i XRAM 20 skal leveres til programhukommelsen hos missilet 12 gjennom PMD bussen. Dersom LPMTEST innmatningen fra datamaskingrensesnittet 28 som leveres til ELLER-porten 55 gjennom oversetterne 36 og 76 er høy, genererer ELLER-porten 55 et klargjøringssignal til driveren 50 som tillater informasjon fra XRAM 20 å bli levert til PMD bussen. I tillegg, ved mottagelse av et CINT signal fra nevnte ADP 16, blir driveren 50 også klargjort til å bevirke informasjon fra XRAM 20 til å bli levert til ADP 16.

For å tillate LIU 30 å kunne tilpasses til enten ADP 16 eller ADP simulatoren 22, er en første ECL multiplekser 52 tilveiebragt. Multiplekseren 52 mottar 16-biters ord, representert som LDB 0-15, fra OMD bussen gjennom oversetteren 42 og 5 nevnte ECL differensial- til enkeltsidigmottakeren 54. I tillegg mottar multiplekseren 52 også visse av TIMING signalene (f.eks. CØ0 og CT3 klokkesignalene) fra nevnte ADP 16 gjennom oversetteren 44 og nevnte ECL differensial til enkeltsidigmottakeren 56. Multiplekseren 52 mottar dessuten 10 13-biters ord representert som LOMADD-12, fra OMA bussen gjennom driveren 58, oversetteren 40 og ECL differensial- til enkeltsidigmottakeren 60. På tilsvarende måte mottar multiplekseren 52 også operandhukommelse-adresse og operandhukommelse-datasignaler samt tidsstyringssignaler fra ADP 15 simulatoren 22 gjennom oversetterne 62-66. Utmatningen fra multiplekseren 52 som leveres til nevnte LIU 30 kan derfor representere operandhukommelse-adresse og operandhukommelse-data informasjonen fra nevnte ADP 16, eller operandhukommelse-adresse og operandhukommelse-datainformasjonen fra ADP 20 simulatoren 22.

For å tillate at programhukommelse-adresseinformasjonen genereres av nevnte ADP 16 eller ADP simulatoren 22 leveres til nevnte XRAM 20, er en andre ECL multiplekser 68 tilveiebragt. Den andre ECL multiplekseren 68 mottar informasjon fra PMA bussen hos ADP 16, representert som LPMADD 0-15, gjennom oversetteren 38 og mottakere 70 som virker som en ECL differensial til entakt-driver. I tillegg mottar multiplekseren 68 også programhukommelse-adresseinformasjon fra 30 ADP simulatoren 22 gjennom oversetteren 72. På grunn av at nevnte XRAM 20 står i forbindelse med en multiplekser 68 gjennom oversetteren 74, kan XRAM 20 motta programhukommelse-adresseinformasjon fra enten nevnte ADP 16 eller ADP simulatoren 22. I tillegg blir programhukommelse-adresseinformasjonen som genereres på utgangen for multiplekseren 68 35 også levert til LIU 30 og kan anvendes på den måte som er beskrevet nedenfor.

Datamaskinen 24 kan anvendes til sekvensmessig å granske innholdet av nevnte OMD eller nevnte PMD på en trinn-for-trinn måte. For å tillate datamaskinen 24 å utføre denne funksjon, kommuniserer XRAM 20 elektrisk med nevnte PMA og OMA busser gjennom oversetteren 78, driverne 80 og 82, samt telleren 84. Når brukeren velger sekvensmessig å granske nevnte OMD eller PMD busser, genererer datamaskinen 24 et LADDRCKL-signal som styrer PMA eller OMA bussene gjennom driverne 80 og 82. Ved mottakelse av LADDRCLK signalet, blir respektive datainnhold i programhukommelsen eller operandhukommelsen levert til OMA bussen eller PMA bussen. Når brukeren velger å motta informasjonen som forplanter seg i enten OMD eller PMD bussene, blir informasjonen på OMD eller PMD bussene så levert til datamaskinen 24 gjennom datamaskingrensesnittet 28. LADDRCLK signalet inkrementerer telleren 84 slik at innholdet i operand eller programhukommelsene sekvensmessig kan plasseres på henholdsvis PMD eller OMD bussene.

For å innsamle informasjon som fremkommer på OMA bussen, omfatter nevnte LIU 30 et direktelager 88 som vist i fig. 3 (B). Direktelageret 88 inneholder 2K ord av direktelagring, og mottar utmatningen fra OMD bussen gjennom en driver 80. For å identifisere startings og stoppingsadressene hvorfra innholdet i hukommelsen 88 skal leses, er det tilveiebragt et startadresse-register 92 og et sluttadresse-register 94. Startadresse-registeret 92 anvendes til å lagre startingsadressen på hvilken innholdet i hukommelsen 88 skal leses fra hukommelsen 88, som leveres til startadresseregisteret 92 fra datautgangsbussen D0 ved generering av et RLS12 styresignal. SAD utmatningen fra startadresseregisteret 92 leveres til en teller 96 som sekvensmessig inkrementerer adressen som initielt er lagret i startadresseregisteret 92 ved mottagelse av et RLS14 styresignal. På en lignende måte blir sluttadresseregisteret 94 anvendt til å lagre sluttadressen som skal leses fra hukommelsen 88. Sluttadressen lastes inn i sluttadresseregisteret 94 fra datautgangsbussen D0 ved

levering av et RLS13 styresignal. RAD utmatningen fra telleren 96 samt FAD utmatningen fra sluttadresseregisteret 94 leveres til en komparator 100. Ved mottagelse av et klargjørende COMPEN signal fra en direktelagertilgangs-
5 styrelogikkrets 98, bestemmer komparatoren 100 hvorvidt sluttadressen er blitt nådd ved å sammenligne RAD utmatningen fra telleren 96 med FAD utmatningen fra sluttadresse-
registeret 94. Dersom sluttadressen er blitt nådd, genererer komparatoren 100 et HOLD styresignal som instruerer direkte-
10 lagertilgang styrelogikkretsen 98 til å overføre innholdet i hukommelsen 88 til datamaskinen 24 på den måte som er beskrevet nedenfor. Utmatningen fra telleren 96 leveres også til en adressemultiplekser 102 som leverer 2KSEL og MA utmatningene til hukommelsen 88 som anvendes til å velge
15 adressestedet hvorfra informasjon i hukommelsen 88 skal leses.

For å tilveiebringe adresserende informasjon til hukommelsen 88 når informasjon skal skrives inn i hukommelsen 88, er en
20 direkte-lagertilgang modus dekodingskrets 104 og en datainnhentingssystem-datadekoder 106 tilveiebragt. Direkte-lagertilgang modus dekodingskretsen 104 anvendes til å generere klargjørende LIUEN utmatning eller en blokkerende DMA utmatning for direkte-lagertilgang styrekretsen 98 avhengig
25 av hvorvidt grunntall 10 verdien av informasjonen for OMA bussen som mottas gjennom driveren 105 er henholdsvis enten 4071 eller 4070. Datainnhentingssystemets datadekodingskrets 106 anvendes for å generere klargjørende DATDECO, INC og DATAVAL signaler når et COMWREN skrivesignal mottas fra
30 nevnte ADP 16 eller simulatoren 22, og informasjonen som fremkommer i OMA bussen gjennom driveren 105 har en grunntall 10 verdi lik 4064 som angir at en skriveoperasjon skal finne sted.

35 For å levere adresseinformasjon til hukommelsen 88 under en skriveoperasjon, er en merketeller 108 og en tre-biters sideholdekrets 110 tilveiebragt. Merketelleren 108 mottar

åtte biter fra OMD bussen og er derfor i stand til å aksessere informasjon som befinner seg innenfor en bestemt 256 steds side i hukommelsen 88. Adresseinformasjonen på OMA bussen lagres til merketelleren 108 når et klargjørende LOAD signal mottas fra en merkedekodingskrets 112. Merkedekodingskretsen 112 anvendes for å klargjøre merketelleren 108 når nevnte ADP 16 avgir en COMWREN kommando og informasjonen som fremkommer på OMA bussen som mottas gjennom driveren 105 har en grunntall 10 verdi lik 3927. Adresseinformasjonen i merketelleren 108 inkrementeres av WCTCLK utmatningen fra ELLER porten 114 ved mottagelse av INC utmatningen fra datainnhentingsystemets datadekodingskrets 106 eller ved en utmatning fra merkedekodingskretsen 112 når informasjonen som fremkom på OMA bussen som en grunntall 10 verdi lik 3927. Tre-biters sideholdekretsen 110 mottar tre informasjonsbiter fra OMD bussen til å indikere hvilken side den bestemte informasjonen som fremkom på OMD bussen skal skrives. Følgelig mottar adressemultiplekseren 102 sidevekslende (paging) informasjon fra tre-biters sideholdekretsen 110 og informasjon vedrørende forflytning innenfor en bestemt side fra merketelleren 108. For å inkrementere siden på hvilken informasjon er lagret i hukommelsen 88, er det tilveiebragt en komparator 115 og en OG port 116. Komparatoren 115 bestemmer hvorvidt utmatningen fra merketelleren 108 er lik 255 som representerer den maksimale forflytning innenfor en bestemt side som en adresse kan ha. Dersom komparatoren 115 bestemmer at forflytningen har overskredet 255, generere komparatoren 115 en utmatning til OG porten 116, hvilket bevirker OG porten 116 til å levere en PGSTB utmatning til tre-bit sideholdekretsen 110. PGSTB utmatningen fra OG porten 116 instruerer tre-bit sideholdekretsen 110 til å inkrementere sideadressen som skal leveres til hukommelsen 88 gjennom adressemultiplekseren 102.

Direkte-lagertilgang styrelogikkretsen 98 kommuniserer med datamaskinen 24 gjennom oversettere 118-122 som anvendes til å omdanne ECL logikknivåene i direkte-lagertilgang styre-

logikkretsen 98 til TTL logikk nivåene i datamaskinen 24. Oversetteren 118 mottar en LREADY fra datamaskinen 24 som anvendes til å indikere at datamaskinen 24 er i stand til å motta den informasjon som er lagret i hukommelsen 88. 5 Oversetteren 120 leverer et IREADY fra direkte-lagertilgang styrelogikkretsen 98 til datamaskinen 24 som anvendes til å indikere at direkte-lagertilgang styrelogikkretsen 98 er klar til å levere den informasjon som er lagret i hukommelsen 88. Oversetteren 122 anvendes til å levere et IBUSY signal fra 10 datamaskinen 24 til direkte-lagertilgang styrelogikkretsen 98 som anvendes til å indikere at datamaskinen 24 er opptatt og ikke i stand til å motta innholdet i hukommelsen 88. Når direkte-lagertilgang styrelogikkretsen 98 og datamaskinen 24 er i stand til å overføre informasjon som er lagret i 15 hukommelsen 88, leverer direkte-lagertilgang styrelogikkretsen 98 et \overline{WR} til \overline{WE} inngangen på hukommelsen 88 som bevirker ID utmatning fra hukommelsen 88 til å bli levert til en datamultiplekser 124 gjennom et lesevalgregister 126. ID utmatningen fra hukommelsen 88 leveres så fra datamultipleks- 20 eren 124 til datamaskinen 24 gjennom oversetteren 128 ved mottakelse av et RLS32 styresignal ved lesevalgregisteret 126.

Som angitt ovenfor tillater nevnte LIU 30 datamaskinen 24 25 enten å utføre en pause, tilpasning, låsning eller sporeoperasjoner ved opptredenden av visse betingelser. For å velge hvilke av disse operasjoner som skal utføres, er 212CR 132 registeret og 213CR 134 registeret tilveiebragt. 212CR 132 og 213CR 134 registrene mottar data fra datamaskinen 24 30 gjennom en driver 136, og lastes ved leveringen av henholdsvis RLS15 og RLS16 styresignalene. Innholdet i nevnte 212CR 132 og nevnte 213CR 134 anvendes til å bestemme hvilke av en flerhet av signaler identifisert som MATCH1 til og med MATCH4, generert på den måten som beskrevet nedenfor, 35 bevirker visse hendelser til å opptre som så klargjør nevnte LIU/XRAM 18 til å utføre pause, tilpasnings, låsnings (holde), og sporeoperasjonene som er beskrevet ovenfor.

For å bevirke nevnte ADP 16 til å ta pause når en MATCH1 operasjon skal utføres, er en OG port 138 tilveiebragt. OG porten 138 mottar et MATCH1 signal gjennom driveren 139 og CR1 biten fra 213CR 134 registeret. Dersom MATCH1 signalet og CR1 biten fra 213CR 134 registeret begge er høye, er utmatningen fra OG porten 138 høy. Utmatningen fra OG porten 138 setter en pauseholdekrets 140, som kan tilbakestilles ved levering av et RLS10 styresignal, hvorved bevirkes at pauseholdekretsen 140 leverer et logisk høyt signal til ELLER porten 142. Utmatningen fra ELLER porten 142 leveres gjennom driveren 143 og grensesnittbufferen 14 til nevnte ADP 16 som bevirker nevnte ADP 16 til å ta pause. En pause kan altså genereres når nevnte CRO i 213CR 134 registeret er logisk høy, hvilket representerer en instruksjon fra datamaskinen 24 til å stoppe nevnte ADP 16. I tillegg leveres utmatningen fra ELLER porten 142 til en tidsmåler/teller styrelogikkrets 144. Tidsmåler/teller styrelogikkretsen 144 står elektrisk i forbindelse med en tidsmåler 148 som drives av en 5MHz oscillator 150. Når utmatningen fra ELLER porten 142 som mottas av tidsmåler/teller styrelogikkretsen 144 er logisk høy, instruerer tidsmåler/teller styrelogikkretsen 144 tidsmåleren 148 til å stoppe tidsmåling og generere en ITMMR utmatning som indikerer tidspunktet hvor pausen opptrådte. ITMR utmatningen fra tidsmåleren 148 leveres så til datamaskinen 24 gjennom datamultiplekseren 124.

For å tillate LIU/XRAM 18 til å la nevnte ADP 16 ta pause etter at et forutvalgt antall av MATCH1 signaler er blitt generert, er en komparator 154 tilveiebragt. Komparatoren 154 mottar utmatningen fra et tilpasningstelleregister 155 som lastes med et forutvalgt antall av tilfeller ved hvilke en MATCH1 operasjon skal opptre fra datautgangsbussen DO ved mottagelse av et RLS11 styresignal. En tidsmåler/teller 156 teller så antallet av opptredener som en MATCH1 operasjon har opptrådt og genererer en reagerende IMC utmatning som leveres til komparatoren 154. Dersom komparatoren 154 bestemmer at CDAT utmatningen fra tilpasningstelleregisteret 155 er lik

IMC utmatningen fra tidsmåleren/telleren 156, genererer komparatoren 154 en logisk høy CTMCH utmatning som leveres til OG porten 158. Dersom verdien av CR2 biten i 212CR 132 registeret også er høy, blir utmatningen fra OG porten 158
5 logisk høy, hvorved det leveres et pausesignal til data-maskinen 24 gjennom ELLER porten 142 og driveren 135.

For å tillate nevnte LIU/XRAM 18 å telle antallet av opptredener som MATCH3 og MATCH4 signalene er blitt generert,
10 er OG porten 162 og 164 tilveiebragt. OG porten 162 mottar MATCH3 signalet gjennom driveren 166 og en CR3 bit fra 213CR 134 registeret. På tilsvarende måte blir MATCH4 signalet mottatt av OG porten 164 gjennom driveren 166, samt utmatningen fra CR4 biten fra 213CR 134 registeret. Dersom
15 utmatningen fra den ene eller andre av OG portene 162 og 164 er logiske høye, leveres det høye signalet til teller/tidsmåler styrelogikkretsen 144 gjennom ELLER porten 172 som instruerer tidsmåleren/telleren 156 til å begynne å telle. Når utmatningen fra tidsmåleren/telleren 156 er lik verdien som er lagret i tilpasningstelleregisteret 155 slik som
20 bestemt av komparatoren 154, genererer komparatoren 154 en logisk høy CTMCH utmatning som leveres til OG porten 158. Dersom CR2 biten fra 213CR 134 registeret er logisk høy, blir utmatningen fra OG porten 158, og dermed utmatningen fra ELLER porten 142 logisk høye. På grunn av at utmatningen fra
25 ELLER porten 142 er høy, leveres et pausesignal til nevnte ADP 16 gjennom driveren 143 og grensesnittbufferen 14. I tillegg, dersom CR5 biten fra 212CR 132 registeret også er høy, blir utmatningen fra OG porten 174, og dermed ELLER porten 176, logisk høy, hvorved det genereres et TRACE 1 (spore 1) signal som anvendes til å initiere sporeoperasjonen på den måte som er beskrevet nedenfor. Følgelig kan nevnte LIU 30 anvendes til å pause operasjonen av nevnte ADP 16 eller initiere sporing etter at et forutbestemt antall av
30 MATCH3 og MATCH4 signaler er blitt generert.
35

For å bestemme tidspunktet ved hvilket et MLLATCH2 signal genereres, er en OG port 180 tilveiebragt. OG porten 180 mottar et MATCH2 signal samt CR5 biten fra 213CR 134 registert. Når MATCH2 signalet og CR5 biten er høye, blir logisk høy utmatning levert av OG porten 180 til tidsmåler/-teller styrelogikkretsen 144. Tidsmåler/teller styrelogikkretsen 144 bevirker så tidsmåleren 148 til å generere en ITMR utmatning som indikerer tidspunktet hvor MATCH2 signalet opptrådte. ITMR utmatningen fra tidsmåleren 148 leveres så til datamaskinen 24 gjennom datamultiplekseren 124.

For å låse dataene som fremkommer på OMD bussen når et MATCH2 eller MATCH4 signal genereres, er OG portene 184 og 186 tilveiebragt. OG porten 186 mottar en CR6 bit fra 213CR 134 registeret samt et MATCH4 signal gjennom driveren 168. På tilsvarende måte mottar OG porten 184 CR6 biten fra 213CR 134 registeret samt MATCH2 signalet fra driveren 188. Utmatningene fra OG porten 184 og 186 leveres til en ELLER port 190 som genererer en LATCH (holde) utmatning som så leveres til en holdekrete 192. Hvis den ene eller andre av utmatningene fra OG porten 184 og 186 er logisk høye, bevirker ELLER porten 190 holdekrete 192 til å lagre informasjonen som fremkommer på OMD bussen. Holdekrete 192 er derfor i stand til å generere en LFD utmatning som inneholder informasjonen som fremkommer på OMD bussen når LATCH signalet blir opptatt av holdekrete 192. I tillegg blir LATCH signalet fra ELLER porten 190 også levert til en vippe 196. Vippen 196 anvendes til å generere en LATCHF utmatning for å indikere at en låsningsoperasjon har opptrådt. Vippen 196 mottar også en RLS18 som anvendes til å tilbakestille vippen 196.

For å generere et avbruddssignal ved opptreden av enten et MATCH1, MATCH2, MATCH3 eller MATCH4 signal, er OG porter 198-204 tilveiebragt. OG porten 198 mottar en CRO bit fra 212CR 132 registeret, samt MATCH1 signalet fra driveren 139. I tillegg mottar OG1 porten 200 CR1 biten fra 212CR 132

registeret samt MATCH2 signalet fra driveren 188. Likeledes mottar OG porten 202 MATCH3 signalet samt CR2 biten fra 212CR 132 registeret, og OG porten 204 mottar MATCH4 signalet samt CR3 biten fra 212CR 132 registeret. Utmatningene fra OG porten 198-204 leveres gjennom en ELLER port 209 til en ELLER port som i sin tur er koblet til en 800 nanosekunders pulsgenerator 212. Pulsgeneratoren 212 anvendes for å generere en 800 nanosekunders puls når hvilke som helst av MATCH1 - MATCH4 signalene er logisk høye og den korresponderende biten i 212CR 132 registeret også er høy. Utmatningen fra pulsgeneratoren 212 leveres til en inngang til OG port 214, hvis andre inngang er koblet til 211CR 216 registeret. 211CR 216 registeret anvendes til å klargjøre eller blokkere OG porten 214, og lastes ved levering av et RLS 9 styresignal. Når utmatningen fra 211Cr 216 registeret og pulsgeneratoren 212 begge er høye, genererer OG porten 214 en logisk høy LSET15 utmatning som leveres til nevnte ADP 16 gjennom grensesnittbufferen 14. LSET15 utmatningen fra OG porten 214 fortolkes så av nevnte ADP 16 som et avbruddsignal som stopper operasjonen av nevnte ADP 16. ELLER porten 210 mottar også et RLS 8 styresignal som anvendes til å initiere et avbrudd fra datamaskinen 24 direkte. For å tilveiebringe direkte aksess til MATCLLH1 - MATCH4 signalene, leveres utmatningen fra driverne 166, 168, 188 og 206 til driverne 218-224 som er koblet til testpunkter som er koblet til testpunkter som er plassert på huset (ikke vist) for nevnte LIU/XRAM 18. Utmatningene fra driverne 218-224 kan så kobles til et oscilloskop slik at MATCH1-MATCH4 signalene kan observeres.

For å generere MATCH1 signalet, er en komparator 226 tilveiebragt. En komparator 226 mottar utmatningen fra en holdekrets 228 som elektrisk kommuniserer med datautgangsbussen D0 fra nevnte LIU 30. Når et RLS 20 styresignal genereres, vil holdekretsen 228 låse de data som fremkommer på datautgangsbussen D0. I tillegg mottar komparatoren 226 utmatningen fra en multiplexer/holdekrets 230 som mottar

operandhukommelse adresseringsinformasjon fra OMA bussen og programhukommelse adresseringsinformasjon fra holdekretsen 232. En CT3 klokkepuls er gitt til holdekretsen 230 gjennom holdekretsen 253 som elektrisk kommuniserer med ELLER porten 255, mens CØ0 klokkesignalet leveres til holdekretsen 232 gjennom driveren 257. Holdekretsen 232 kommuniserer elektrisk med PMA bussen og leverer en låst programhukommelse adresserings LPMA innmatning til multiplekseren/ holdekretsen 230. Multiplekseren/holdekretsen 230 mottar dessuten CR1 biten fra 231CR registeret som lastes inn i holdekretsen 234 ved levering av et RLS26 styresignal. Når komparatoren 226 bestemmer at M1R utmatningen fra holdekretsen 228 er lik utmatningen fra multiplekseren/holdekretsen 230, leveres en logisk høy utmatning til en inngang på en OG port 236. OG porten 236 mottar også utmatningen fra multiplekseren 237 som multiplekser CT3 og OMAVAL klokkesignalene genereres på den måte som er angitt nedenfor. Når begge innmatninger til OG porten 236 er logisk høye, blir et logisk høyt MATCH1 signal levert gjennom driveren 238.

For å generere MATCH2 signalet, er en komparator 240 tilveiebragt. Komparatoren 240 mottar M2R utmatningen fra holdekretsen 242 som elektrisk står i forbindelse med data utgangsbussen D0. Holdekretsen 242 mottar også RLS21 styresignalet som anvendes til å laste adresserende informasjon fra datautgangsbussen D0 inn i holdekretsen 242. Komparatoren 240 mottar også utmatningen fra en multiplekser/holderkrets 242. Multiplekseren/holdekretsen 242 mottar operandhukommelse-adresseringsinformasjon fra OMA bussen, og en programhukommelse-adresseringsinformasjon fra holdekretsen 232. Et CT3 klokkesignal leveres til holdekretsen 242 gjennom holdekretsen 253 som elektrisk står i forbindelse med ELLER porten 255. Dersom utmatningen fra multiplekseren/holdekretsen 242 er lik M2R utmatningen fra holderkretsen 242, blir det logiske høye signalet levert til en inngang på en OG port 244. Den andre innmatningen til OG porten 244 er koblet til en multiplekser 246 som mottar CT3

klokkesignalet samt OMAVAL klokkesignalet som genereres på den måte som er beskrevet nedenfor. Dersom begge innmatninger til OG porten 244 er logisk høye, blir utmatningen fra OG porten 244 som representerer nevnte MATCH2 logisk høy. Denne utmatning leveres gjennom driveren 248.

For å generere MATCH3 signalet, tilveiebringes en komparator 250. Komparatoren 250 mottar utmatningen fra en OG port 252 som har en inngang koblet til en multiplekser 254. Multiplekseren 254 mottar informasjon som fremkommer på OMA og OMD bussene samt programhukommelse-adresseringsinformasjon fra holdekretsen 232. Utmatningen fra multiplekseren 254 styres av verdien av CR1 og CR2 bitene fra 232CR. Den andre innmatningen til OG porten 252 er fra en holdekrets 256 som mottar maskeringsinformasjon fra datautgangsbussen D0 ved levering av et RLS23 styresignal. På grunn av at utmatningene fra både holdekretsen 256 og multiplekseren 254 leveres til OG porten 252, representerer utmatningen fra OG porten 252 kun de bitene som leveres fra multiplekseren 254 som ikke er maskert (dvs. ikke tilsvarer null biter) ved hjelp av informasjonen i holdekretsen 256. Komparatoren 250 mottar også en innmatning fra holdekretsen 258 som mottar tilpasnings/sporeadressen fra datautgangsbussen D0 ved mottagelse av et RLS22 styresignal. Dersom utmatningen fra OG porten 252 og M3R utmatningen fra holdekretsen 258 er like, genererer komparatoren 250 et logisk høyt signal som leveres til en inngang på OG porten 260. Den andre inngangen til OG porten 252 mottar utmatningen fra en multiplekser 262 som genereres av OMAVAL og CT3 klokkesignalene på den måte som er beskrevet nedenfor. Dersom begge innmatninger til OG porten 260 er logisk høye, blir en logisk høy utmatning levert til ECL driver 264 for å frembringe et logisk høyt MATCH3 signal.

For å generere MATCH4 signalet, er OG porten 266 tilveiebragt. OG porten 266 mottar en innmatning fra OG porten 244 samt fra komparatoren 250. Dersom begge innmatninger fra Og

porten 266 og komparatoren 250 er logisk høye, genererer OG porten 244 en logisk høy utmatning som leveres til driveren 268 for å frembringe et logisk høyt MATCH4 signal.

5 For å lagre informasjonen på nevnte PMA, OMA og OMD busser under en sporeoperasjon, er to FIFO hukommelser 270 og 272 tilveiebragt. FIFO hukommelsen 270 mottar informasjon som fremkommer på enten nevnte OMA, OMD eller PMA busser fra
10 multiplekseren 254 gjennom holdekretsen 274, mens FIFO hukommelsen 272 mottar informasjon som fremkommer på OMD bussen gjennom holdekretsen 276. FIFO hukommelsen 270 kan leses når verdien av CR14 biten fra 223 CR registeret er høy og RLS 25 styresignalet genereres. Når dette skjer, mottar holdekretsen 278 CR14 biten fra datautgangsbussen D0 og
15 leverer CR 14 biten til klargjøringsinngangen på FIFO hukommelsen 270 gjennom ELLER porten 280. På grunn av at verdien av CR 14 biten er høy, blir FIFO hukommelsen 270 klargjort for derved å generere en utmatning identifisert som TDATA. På tilsvarende måte kan FIFO hukommelsen 272 leses fra når
20 verdien av CR14 biten i 233 CR registeret er lav og RLS25 styresignalet genereres. Når dette skjer mottar holdekretsen 278 CR14 biten fra datautgangsbussen D0 og leverer CR14 biten til klargjøringsinngangen på FIFO hukommelsen 272 gjennom den inverterende inngangen på ELLER porten 282. På grunn av at
25 verdien av CR14 biten er lav, er utmatningen fra ELLER porten 282 høy, hvorved frembringes en logisk høy klargjøringsinngang til FIFO hukommelsen 272. TDATA utmatningene fra hukommelsene 270 og 272 leveres til datamaskinen 24 gjennom datamultiplekseren 284 ved levering av de passende styresignaler til holdekretsen 286.
30

Informasjonen som fremkommer på OMA, OMD og PMA bussene kan skrives inn i de respektive FIFO hukommelser 270 og 272 i den ene eller den andre av to modi: kontinuerlig sporing eller
35 triggersporing. I den kontinuerlige sporingsmodus, blir informasjonen kontinuerlig skrevet inn i FIFO hukommelsen 270 og 272. I trigger-sporingsmodusen skrives informasjonen inn

i FIFU hukommelsen 270 og 272 kun når et TRACE 1 signal er blitt generert på den måte som er beskrevet ovenfor. De kontinuerlige sporingsmodus velges når RLS25 styresignalet er blitt generert som bevirker CR13 biten fra 233CR registeret til å bli høy. Når dette skjer, mottar holdekretsen 278 CR13 biten fra datautgangsbussen D0 og leverer CR13 biten til klargjøringsinngangen på FIFU hukommelsen 270 gjennom ELLER porten 280 og 288. I tillegg leverer holdekretsen 278 CR13 biten til klargjøringsinngangen på FIFU hukommelsen 272 gjennom ELLER porten 288 og ELLER porten 282. På grunn av at CR13 biten er høy, blir FIFU hukommelsen 270 og 272 klargjort.

For å klargjøre FIFU hukommelsene 270 og 272 når trigger sporemodusen velges, tilveiebringes holdekretsen 290 og 292. Holdekretsen 292 mottar TRACE 1 signalet og genererer en reagerende TACE TRIG utmatning. TRACE TRIG utmatningen fra holdekretsen 292 leveres så til en inngang på OG porten 294. De andre to innmatningene til Og porten 294 er CR13 biten fra 233CR registeret og en utmatning fra en komparator 296 som anvendes til å avslutte sporingsoperasjonen på den måte som er beskrevet nedenfor. Selvom CR13 biten fra 233CR registeret er høy under sporing, er utmatningen fra OG porten 294 logisk lav, ettersom utmatningen fra komparatoren 296, som anvendes til å avslutte sporingsoperasjonen, er lav. Utmatningen fra OG porten 294 leveres til inngangen på holdekretsen 290 som kan tilbakestilles ved levering av et RLS28 styresignal. LAST utmatningen fra holdekretsen 290 leveres til klargjøringsinngangen på FIFU hukommelsen 270 gjennom ELLER porten 280 og 288, og til klargjøringsinngangen på FIFU hukommelsen 272 gjennom ELLER porten 282 og 288. På grunn av at LAST utmatningen fra holdekretsen 290 er logisk lav når logisk høye signaler mottas av OG porten 294 fra komparatoren 296, blir FIFU hukommelsene 270 og 272 satt ut av funksjon når et høyt signal genereres fra utgangen på en komparator 296.

For å styre adressen til hvilken data skal skrives inn i eller leses fra FIFU hukommelsene 270 og 272, er en multiplekser/holdekrets 298 tilveiebragt. Multiplekseren/holdekretsen 298 står elektrisk i forbindelse med de adresserende portene på FIFU hukommelsene 270 og 272, og mottar lese-adresse RAC utmatningen fra en holdekrets 300 som styrer adressen fra hvilken informasjon FIFO hukommelsene 270 og 272 skal leses. Holdekretsen 300 mottar de første åtte biter CR0-CR7 fra 233CR registeret når innholdet i 233CR registeret leveres til holdekretsen 300 når et RLS25 styresignal genereres.

Under en skriveoperasjon blir den adresserende informasjon som leveres av multiplekseren/holdekretsen 298 til FIFU hukommelsen 270 og 272 mottatt fra en skriveteller 302. Skrivetelleren 302 genererer en sekvensielt inkrementert AWC utmatning som så leveres til FIFU hukommelsene 270 og 272 gjennom multiplekseren/holdekretsen 298. For å styre skrivetelleren 302, tilveiebringes en posisjonsteller 304. Posisjonstelleren 304 klargjøres ved levering av et TRACE TRIG signal fra holdekretsen 292 og genererer en TPC utmatning som reaksjon. TPC utmatningen fra posisjonstelleren 304 leveres til komparatoren 296 som også mottar RAC utmatningen fra holdekretsen 300. Dersom TPC utmatningen og RAC signalet er like, og som indikerer at den brukerspesifiserte terminaladressen i skriveoperasjonen er blitt nådd, går utmatningen fra komparatoren 296 logisk høy, og leveres til OG porten 294. På grunn av at de to innmatningene til OG porten 294 også er høye når en TRACE (spore) modus initieres, går utmatningen fra OG porten 294 logisk høy, hvorved frembringes en logisk høy LAST utmatning fra holdekretsen 290. En logisk høy LAST utmatning til skrivetelleren 302 setter så skrivetelleren 302 ut av funksjon, slik at AWC signalet ikke sekvensmessig inkrementeres.

På grunn av at multiplekser/holdekretsen 298 settes av MUX utmatningen fra ELLER porten 288, vil multiplekseren/hold-

kretsen 298 bli satt hvis enten CR13 biten fra 233CR registeret er høy, som indikerer at en kontinuerlig spore skrivemodus er blitt valgt, eller LAST utmatningen fra holdekretsen 290 er lav, som indikerer at trigger spore skrivemodusen er blitt valgt. For å gi et klokkesignal for holdekretsen 290 og multiplekseren/holdekretsen 290, er en 5 42 ± 8 nanosekunders klokke 306 tilveiebragt. Utmatningen fra klokken 306 leveres til en ELLER port 309 som i sin tur er koblet til klokkeinngangen på multiplekseren/holdekretsen 10 298. Klokken 306 mottar utmatningen fra en multiplesker 308 som i sin tur mottar TRACE 1 innmatningen samt RAM1VAL utmatningen fra multiplekseren 262. Multiplekseren 262 genererer et RAM1VAL fra OMAVAL klokkesignalet samt CT3 klokkesignalene. Klokken 306 driver enn 50 nanosekunders 15 klokke 311 som gir klokkeimpulser til skrivetelleren 302. I tillegg leveres utmatningen fra klokken 311 også til en OG port 313 som mottar LAST utmatning fra holdekretsen 290. Utmatningen fra OG porten 313 leveres til FIFU hukommelsene 270 og 272 for derved å tilveiebringe klokkeimplser for FIFU 20 hukommelsene 270 og 272 under en skriveoperasjon.

OMAVAL klokkesignalet anvendes som et klokkesignal når informasjon overføres fra OMA bussen under en OMA lese- eller skriveoperasjon. For å generere OMAVAL signalet, tilveiebringes OG porten 310, 312 og ELLER porten 314. OG porten 25 312 mottar missilets lesepulser CDBOMEN gjennom mottakeren 316. Fra holdekretsen 318 mottar OG porten 312 også CR3 biten fra 233CR registeret som tilsvarer en operandhukommelseadresse leseoperasjon, eller en operandhukommelseadresse lese/skriveoperasjon. I tillegg mottar OG porten 310 missilets skrive- 30 signal COMWREN fra mottakeren 317, samt CR2 biten i 231CR registeret fra holdekretsen 318 som angir en operandhukommelseadresse skriveoperasjon eller en operandhukommelse lese/skriveoperasjon. Dersom den ene eller andre av 35 utmatningene fra OG portene 310 og 312 er logiske høy, blir utmatningene fra ELLER porten 314 logisk høy. Utmatningen fra ELLER porten 314 leveres til setteterminalen på holde-

kretsen 320, hvor tilbakestillingsterminalen på denne er koblet til ELLER porten 322. ELLER porten 322 mottar RLS28 styresignalet samt utmatningen fra en 50 ± 8 nanosekunders klokke 324 som, i sin tur, mottar CT3 klokkesignalene fra ADP bufferen 14 gjennom driveren 325. Utmatningen fra holde-
5 kretsen 320 leveres til en inngang på en OG port 326, hvis andre inngang mottar CT3 klokkesignalet. Utmatningen fra OG porten 326 representerer OMAVAL signalet som i sin tur leveres til multiplekseren 262.

10 RLS styresignalene genereres ved hjelp av en flerhet av styrestrobe dekodingskretser 327 og 328 som elektrisk kommuniserer med datamaskinens grensesnittmodul 28. Styrepuls (strob) dekodingskretsen 327 mottar forskjellige
15 styresignaler fra datamaskin-grensesnittmodulen 28 og genererer RLS8-RLS19 styresignalene. I tillegg mottar styrepuls dekodingskretsen 328 forskjellige styresignaler fra datamaskinen grensesnittmodulen 28 og genererer RLS20-RLS31 styresignalene. LIU 30 omfatter også en driver 330 som
20 mottar LPM utmatningen fra datamaskin-grensesnittmodulen 14 som leveres gjennom holdekretsen 332, slik at programhukommelse-datainformasjonen kan leveres til datamaskinen 24 under en rundhopp (wraparound) operasjon under hvilken data som leveres til nevnte ADP16 av datamaskinen 24 kan kontrolleres.
25 I tillegg blir data fra driveren 330 også levert til XRAM 20 gjennom oversetteren 334 under en dupingsoperasjon. LIU 30 omfatter også en holdekrets 335 som anvendes for å låse innholdet i 232CR registeret ved mottagelse av RLS27 styresignalet.

30 Datamaskinen 24 kan kjøre det programmet som er illustrert i flytskjemaene som fremgår av figurene 5-8. Ved trinn 336 blir missil utviklingsprogramvaren (i det etterfølgende "MDS") lastet og programmet får brukerkommandoen ved trinn
35 338. Ved trinn 340 bestemmer programmet hvorvidt brukeren har valgt hovedtilbakestillingskommandoen MR. Dersom brukerkommandoen er den hovedtilbakestillingskommando MR som

er blitt valgt, blir hovedtilbakestillingskommandoen MR utført ved trinn 342 som tilbakestiller missilenes data-prosessor, nevnte LIU 30, og samtlige registre. I tillegg settes programtelleren i ADP 16 lik null. Behandling leveres så til trinn 344 av returtrinnet 346. Ved trinn 344 bestemmer programmet hvorvidt ENDRE kommandoen er blitt valgt. Dersom ENDRE kommandoen er blitt valgt, utføres ENDRE kommandoen i trinn 347, som tillater brukeren å lappe programhukommelsen i ADP16 i sekvensmessige eller tilfeldige steder. Styring bringes så tilbake til trinn 348 ved hjelp av returtrinn 350. Ved trinn 348, bestemmer programmet hvorvidt SLETTE kommandoen er blitt valgt. Dersom så er tilfelle utføres trinn 352 som tilbakestiller nevnte ADP 16 samt dens programteller til null. Styring leveres så til trinn 354 ved hjelp av returtrinnet 356. Ved trinn 354 bestemmer programmet hvorvidt DUMPE kommandoen er blitt utført. Hvis så er tilfelle, utføres trinnet 358 som bevirker innholdet XRAM 20 til å bli fremvist på en terminal linjeskriver som er lagret på en bestemt platefil i data-maskinen 24. Styring leveres så til trinnet 360 via returtrinnet 362. Dersom XINT kommandoen er blitt valgt ved trinn 360 utføres trinnet 364 som bevirker at et avbrudd leveres til nevnte ADP 16. Styring leveres så til trinnet 366 ved hjelp av trinnet 368.

Programmet bestemmer så hvorvidt STOPPE kommandoen er blitt valgt ved trinn 366. Hvis STOPPE kommandoen er blitt valgt, utføres STOPPE kommandoen ved trinn 370, hvilket stopper utføringen av det program som er lagret i programhukommelsen i nevnte ADP 16. Styring leveres så til trinn 372 ved hjelp av returtrinnet 374. Ved trinn 372 bestemmer programmet hvorvidt LÅSE (LATCH) kommandoen er blitt spesifisert. Dersom LÅSE kommandoen er blitt spesifisert, utføres trinn 376 som setter brukeren i stand til å innfange data som fremkommer på OMA, OMD eller PMA bussene. Styring leveres så til trinn 378 ved hjelp av returtrinnet 380. Trinnet 380 bestemmer så hvorvidt PROM funksjonen er blitt utpekt. Hvis

så er tilfelle, utføres trinnet 382 som endrer programhukommelsen for ADP 16 fra XRAM 20 til nevnte PROM i missilet 12. Styring leveres så til trinn 384 ved hjelp av returtrinnet 386. Programmet bestemmer så hvorvidt FIFU kommandoen er
5 blitt valgt ved trinn 384. Hvis så er tilfelle utføres nevnte FIFU ved trinn 386 som bevirker de data som innsamles under sporingsmodusen til å bli fremvist. Styring bringes så tilbake til trinnet 388 ved hjelp av trinnet 390.

10 Ved trinn 388, bestemmer programmet hvorvidt KJØRE kommandoen er blitt valgt. Dersom KJØRE kommandoen er blitt valgt, utføres KJØRE kommandoen på trinn 392 som tillater brukeren å observere pause, tilpasnings, spore eller låseinformasjonen som fremviser dataen på OMA, OMD og PMA bussene. Etterat
15 kjørekommandoen er blitt utført, utført programmet trinnet 394 via trinnet 396. På trinn 394, bestemmer programmet hvorvidt TRINN funksjonen er blitt valgt. Hvis så er tilfelle, utføres trinnkommandoen på trinn 396, som tillater brukeren å skritte gjennom programmet i nevnte ADP en
20 instruksjon av gangen. Trinnet 398 utføres så ved hjelp av returtrinnet 400. På trinn 398 bestemmer programmet hvorvidt TILPASNING (MATCH) kommandoen er blitt valgt. Dersom TILPASNING kommandoen er blitt valgt, utføres TILPASNING kommandoen på trinn 402 som setter brukeren i stand til å
25 telle antallet av opptredener som et bestemt datamønster eller datakilde opptrådte, stoppe nevnte ADP 16 etter et bestemt antall av opptredener av et bestemt datamønster eller kilde, eller å trigge SPORE (TRACE) funksjonen ved et tilpasningskriterium. Etter at returtrinnet 404 er blitt
30 utført, bestemmer programmet hvorvidt LASTE (LOAD) kommandoen er blitt valgt på trinn 406. Dersom LASTE kommandoen er blitt valgt, utføres LASTE kommandoen på trinn 408 som laster et program inn i SRAM 20 fra en bruker spesifisert platefil i datamaskinen 24. Etter at returtrinnet 410 er blitt utført,
35 bestemmer programmet hvorvidt en NEI - - kommando er blitt valgt for trinn 412. Dersom en NEI - - kommando velges,

kansellerer programmet den utpekte funksjon og styring leveres til trinnet 416 ved hjelp av returtrinnet 418.

På trinn 416 bestemmer programmet hvorvidt PAUSE kommandoen er blitt valgt. Dersom PAUSE funksjonen er blitt valgt, utføres pausefunksjonen på trinn 424, hvilket stopper neste ADP 16 ved opptredenen av enten en bestemt programhukommelse-adresse, en operandadresse, et program eller en absolutt adresse. Etterat PAUSE kommandoen er blitt utført utføres trinnet 422 av returtrinnet 424. På trinn 422 bestemmer programmet hvorvidt FIL kommandoen er blitt valgt. Hvis så er tilfelle, utfører trinnet 426 FIL kommandoen som fremviser til brukeren navnet på den siste FILEN som ble anvendt for LASTE eller VERIFISERINGS funksjonene. Styring bringes så tilbake til trinnet 428 ved hjelp av returtrinnet 430. På trinn 428 bestemmer programmet hvorvidt SPORE kommandoen er blitt valgt. Dersom SPORE kommandoen er blitt valgt, utføres SPORE kommandoen på trinn 432. Dette tillater brukeren å innsamle data i hukommelsene 270 eller 272 enten kontinuerlig eller trigget på en bestemt verdi. På trinn 434 leveres styringen av programmet til trinnet 436. Dersom VERIFISER funksjonen er blitt valgt som bestemt av trinn 436, utføres trinn 438. Trinnet 438 utfører VERIFISERING kommandoen som så verifiserer programmet i nevnte XRAM 20 mot en brukers spesifiserte platefil i datamaskinen 24, eller skaper en lappefil. Styring bringes så tilbake til trinnet 440 ved hjelp av returtrinnet 442. På trinn 440, bestemmer programmet hvorvidt MATVAL kommandoen er blitt valgt. Hvis så er tilfellet, utføres MATVAL kommandoen på trinn 444 som teller antallet av opptredener av et bestemt datamønster når verdien av informasjon i nevnte OMD er lik envalgt verdi. Styring leveres så til trinnet 446 via returtrinnene 448. På trinn 446, bestemmer programmet hvorvidt UTGANGS kommandoen er blitt tatt i bruk av brukeren. Hvis så er tilfelle, går programmet ut fra et MDS program på trinn 450. Dersom UTGANGS kommandoen ikke er blitt utført, leveres styringen til trinn 338.

På grunn av at grensesnittbufferen 14 er koblet direkte til OMLA, OMB, PMA og PMD bussene i nevnte ADP 16, kan innholdet i program og operandhukommelsen granskes mens hukommelsen er in situ (dvs. i missilet 12) for derved å bestemme hvorvidt noen skade har opptrådt under installasjon. I tillegg kan programvaren initielt lastes fra datamaskinen 24 til nevnte XRAM 20 slik at den kan testes under anvendelse av nevnte XRAM 20. Dessuten kan informasjonen som er lagret i programhukommelsen i missilet 12 dumpes på nevnte XRAM 20 slik at innholdet i programhukommelsen kan testes ved å kjøre programmene på ADP simulator 22. Til sist, på grunn av at datamaskinen 24 kan anvendes til å styre nevnte ADP gjennom nevnte LIU 30, kan missilets maskinvare som styres av nevnte ADP 16 bli testet.

Det bør forstås at oppfinnelsen ble beskrevet i forbindelse med et bestemt eksempel derav. Andre modifikasjoner vil fremtre for fagfolk etter et studium av beskrivelsen, tegnningene og de etterfølgende patentkrav.

TABELL

233CR REGISTER (LASTET MED RLS25 STYRESIGNAL)

BIT POSISJON

15 14 13 12-8 7-0

FUNKSJON

0 X 0 X X

TRIG/FRITTLØPSMODUS SPORING
ARMERT

0 X 1 X X

KONTINUERLIG MODUS SPORING ARMERT

1 1 X X ADDR

LES FIFU HUKOMMELSE 270

1 0 X X ADDR

LES FIFU HUKOMMELSE 272

271 CR REGISTER (LASTET MED RLS7 STYRESIGNAL)

5		<u>BIT POSISJON</u>				
	3	2	1	0	<u>FUNKSJON</u>	
			X	0	VELG ADP SOM PROSESSOR	
10			0	1	VELG ADP SIMULATOR SOM PROSESSOR	
			0	X	XRAM ER PROGRAMHUKOMMELSE	
			1	0	PROM ER PROGRAMHUKOMMELSE	
15		0			OMA/PMA ADRESSE FRA ADP (TCDBOMEN)	
		1			OMA/PMA ADRESSE FRA TELLER (DUMP)	
20	0				DØDMANN SATT UT AV FUNKSJON	
	1				DØDMANN KLARGJØRING	

25

231CR REGISTER (LASTET MED RLS26 STYRESIGNAL)

30		<u>BIT POSISJON</u>				
	3	2	1	0	<u>FUNKSJON</u>	
	X	X	X	0	MATCH1 PÅ PMA	
30	X	X	X	1	MATCH1 PÅ OMA	
	X	X	0	X	MATCH2 PÅ PMA	
	X	X	1	X	MATCH2 PÅ OMA	
	0	1	X	X	OMA SKRIV KUN	
	1	0	X	X	OMA LES KUN	
35	1	1	X	X	OMA LES/SKRIV	
	0	0	X	X	INGEN OPERASJON	

232CR REGISTER (LASTET MED RLS27 STYRESIGNAL)

<u>BITPOSISJON</u>		<u>F U N K S J O N</u>	
1	0		
0	0	PMA, LÅST	(FIFU HUKOMMELSE 270)
0	1	OMA	(FIFU HUKOMMELSE 270)
1	0	OMD	(FIFU HUKOMMELSE 270)
1	1	INGEN OPERASJON	

212CR REGISTER (LASTET MED RLS15 STYRESIGNAL)

<u>BIT POSISJON</u>										<u>F U N K S J O N</u>	
9	8	7	6	5	4	3	2	1	0		
									1		BETINGELSESLØST AVBRUDD
									1		AVBRUDD PÅ MATCH1
								1			AVBRUDD PÅ MATCH2
						1					AVBRUDD PÅ MATCH3
					1						AVBRUDD PÅ MATCH4
				1							SPORE PÅ MATCH2
			1								SPORE PÅ TELLETILPASNING
		1									IKKE ANVENDT
	1										KLARGJØR TELLEMODUS
		0									KLARGJØR TIDSMÅLERMODUS
			0	0							SPORE FRITTKJØRING

351 CR-REGISTER (LASTET VED HJELP AV RLS33 STYRESIGNAL)

<u>BITPOSISJON</u>				<u>F U N K S J O N</u>	
3	2	1	0		
			0		NORMAL PDC I/O
			1		PDP RUNDHOPP (WRAPAROUND)
		1			IKKE ANVENDT
	1				IKKE ANVENDT
0					IKKE ANVENDT
1					IKKE ANVENDT

211CR REGISTER (LASTET MED RLS9 STYRESIGNAL)

<u>BIT POSISJON</u>		<u>F U N K S J O N</u>	
0			
0		INT 15 SETTE UT AV FUNKSJON	
1		INT 15 KLARGJØRE	

213CR REGISTER (LASTET MED RLS16 STYRESIGNAL)

<u>BIT POSISJON</u>		<u>F U N K S J O N</u>
7	6 5 4 3 2 1 0	
	1	STOPP ADP
5	1	PAUSE PÅ MATCH1
	1	PAUSE PÅ TELLETILPASNING
	1	TELLING PÅ MATCH3
	1	TELLING PÅ MATCH4
	1	TID PÅ MATCH2
10	1	LÅSE PÅ MATCH2
	1	LÅSE PÅ MATCH4

15

20

25

30

35

P a t e n t k r a v

1.

5 Anordning (10) for å teste missilsystemer, der nevnte anordning, som innbefatter en dataprosessor-grensesnittbuffer (14) står i elektrisk forbindelse med en operandhukommelse og en programhukommelse for en missil dataprosessor (16) gjennom en flerhet av hukommelsesbusser, idet nevnte anordning (10) dessuten står i elektrisk forbindelse med en
10 datamaskin (24), k a r a k t e r i s e r t v e d et eksternt direktelager (20) som elektrisk står i forbindelse med nevnte programhukommelse, en logisk grensesnittenhets (18) for innsamling av data fra nevnte operandhukommelse og nevnte programhukommelse i reell-
15 tid, idet nevnte logiske grensesnittenhets (18) omfatter et direktelager som elektrisk står i forbindelse med nevnte dataprosessors grensesnittbuffer (14), og idet nevnte direktelager kan bevirkes til å lagre informasjon som fremkommer på en av nevnte hukommelsesbusser når data-
20 prosessoren (16) utfører en skrivekommando, en dataprosessorsimulator (22) som er i stand til å simulere utmatningen fra nevnte dataprosessor (16), idet nevnte dataprosessor-simulator (22) elektrisk står i forbindelse med nevnte logiske grensesnittenhets (18) og nevnte eksterne
25 direktelager (20), og en datamaskin grensesnittmodul (26) som er i stand til å tilpasse nevnte logiske grensesnittenhets (18) og nevnte eksterne direktelager med nevnte datamaskin (24).

30 2.

Anordning som angitt i krav 1, k a r a k t e r i s e r t v e d at nevnte dataprosessors grensesnittbuffer (14) omfatter en flerhet av oversettere (32-34) som kan bevirkes til å omdanne de logiske nivåer som er tilhørende nevnte
35 dataprosessor (16) til logiske nivåer som er tilhørende nevnte eksterne direktelager (20) og nevnte logiske grensesnittenhets (14).

3.

Anordning som angitt i krav 2, k a r a k t e r i s e r t
v e d dessuten å omfatte en første multiplekser (52) som
kan bevirkes til å multiplekse informasjon fra nevnte
5 dataprosessor(16) og nevnte dataprosessorsimulator (22), idet
nevnte første multiplekser (52) elektrisk står i forbindelse
med nevnte logiske grensesnittedenhet (18) og nevnte eksterne
direktelager.

10 4.

Anordning som angitt i krav 3, k a r a k t e r i s e r t
v e d at nevnte anordning dessuten omfatter en andre
multiplekser (68) som kan bevirkes til å multiplekse
informasjon fra nevnte dataprosessor (16) og nevnte datapro-
15 sessor simulator (22), idet nevnte andre multiplekser (68)
elektrisk står i forbindelse med nevnte logiske grensesnitt-
enhet (18).

5. 20

Anordning som angitt i krav 4, k a r a k t e r i s e r t
v e d at nevnte logiske grensesnittedenhet (18) er innrettet
til å bevirke nevnte datamaskin (24) til å ta pause når
informasjon som fremkommer på en av nevnte hukommelsesbusser
er av en forutbestemt verdi.

25 6.

Anordning som angitt i krav 5, k a r a k t e r i s e r t
v e d at nevnte logiske grensesnittedenhet (18) kan bevirkes
til å telle antallet av opptredener hvor den informasjon som
30 fremkommer på en av nevnte hukommelsesbusser er av en
forutbestemt verdi.

7.

Anordning som angitt i krav 6, k a r a k t e r i s e r t
35 v e d at den nevnte logiske grenseenhet (18) kan bevirkes
til å låse informasjon som fremkommer på en av nevnte
hukommelsesbusser.

8.

Anordning som angitt i krav 7, k a r a k t e r i s e r t
v e d at nevnte logiske grensesnittenhet (18) kan bevirkes
til å spore informasjon som fremkommer på en av nevnte
5 hukommelsesbusser.

9.

Anordning som angitt i krav 1, k a r a k t e r i s e r t
v e d at nevnte logiske grensesnittenhet (18) dessuten
10 omfatter en første FIFU hukommelse (270), idet nevnte første
FIFU hukommelse (270) kan selektivt bevirkes til kontinuerlig
å motta informasjon som fremkommer på en avnevnte hukommel-
sesbusser.

10.

Anordning som angitt i krav 9, k a r a k t e r i s e r t
v e d at nevnte første FIFU hukommelse dessuten selektivt
kan bevirkes til å motta informasjon som fremkommer på en av
nevnte hukommelsesbusser når informasjon som fremkommer på en
15 av nevnte busser er av en forutbestemt verdi.
20

11.

Anordning som angitt i krav 10, k a r a k t e r i s e r t
v e d at nevnte anordning dessuten omfatter en andre FIFU
25 hukommelse (272), idet nevnte andre FIFU hukommelse (272)
selektivt kan bevirkes til kontinuerlig å motta informasjon
som fremkommer på en av nevnte hukommelsesbusser.

12.

Anordning som angitt i krav 11, k a r a k t e r i s e r t
30 v e d at nevnte andre FIFU hukommelse dessuten kan bevirkes
til selektivt å motta informasjon som fremkommer på en av
nevnte hukommelsesbusser når informasjon som fremkommer på en
av nevnte hukommelsesbusser har en forutbestemt verdi.

13.

Anordning som angitt i krav 1, k a r a k t e r i s e r t
v e d at nevnte anordning tillater innholdet i nevnte
programhukommelse å bli dumpet på nevnte eksterne direkte-
lager.

5

14.

Anordning som angitt i krav 1, k a r a k t e r i s e r t
v e d at nevnte logiske grensesnittenhet (18) dessuten kan
bevirkes til å levere informasjon til nevnte dataprosessor
(16) slik at maskinvaren som drives av nevnte dataprosessor
(16) kan testes.

10

15

20

25

30

35

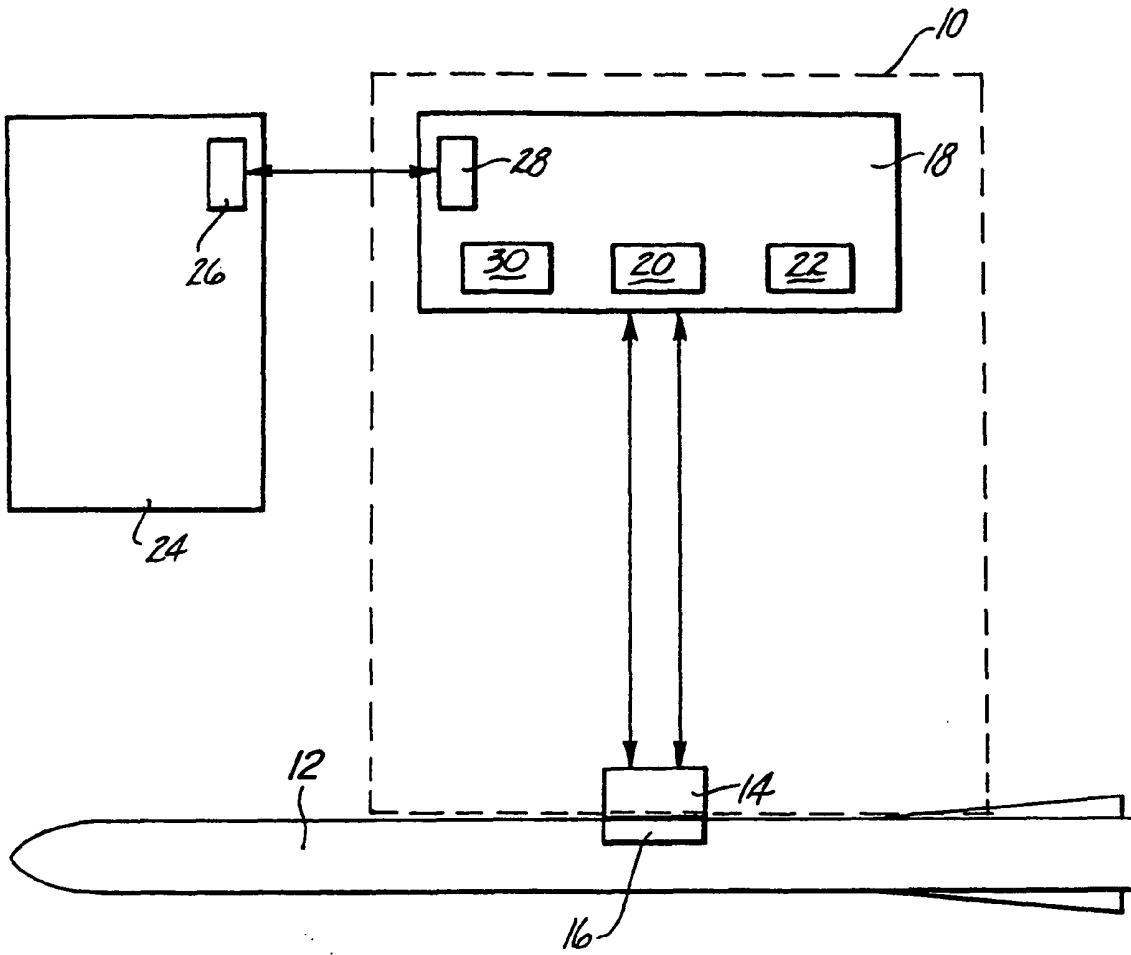


Fig-1

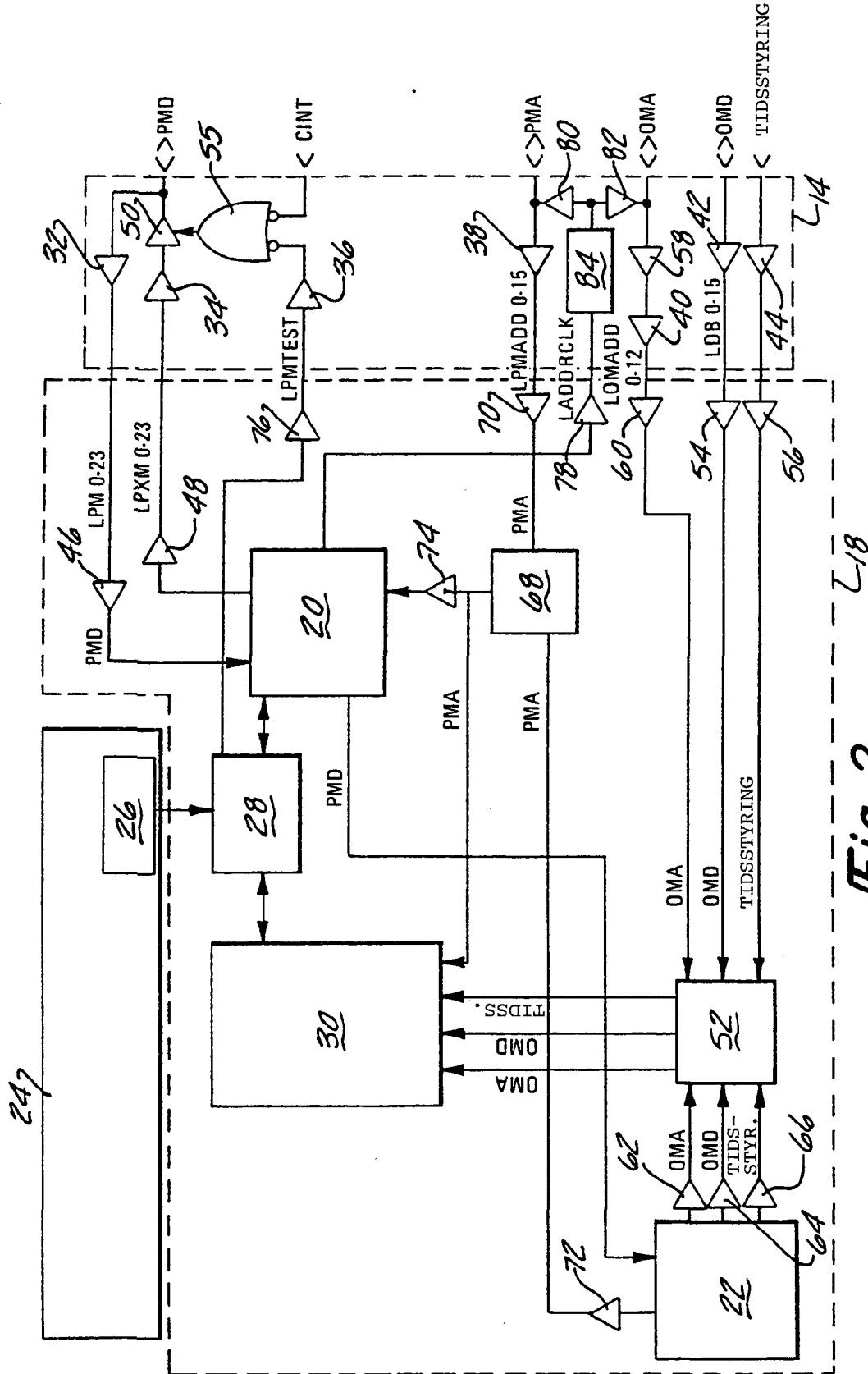


Fig-2

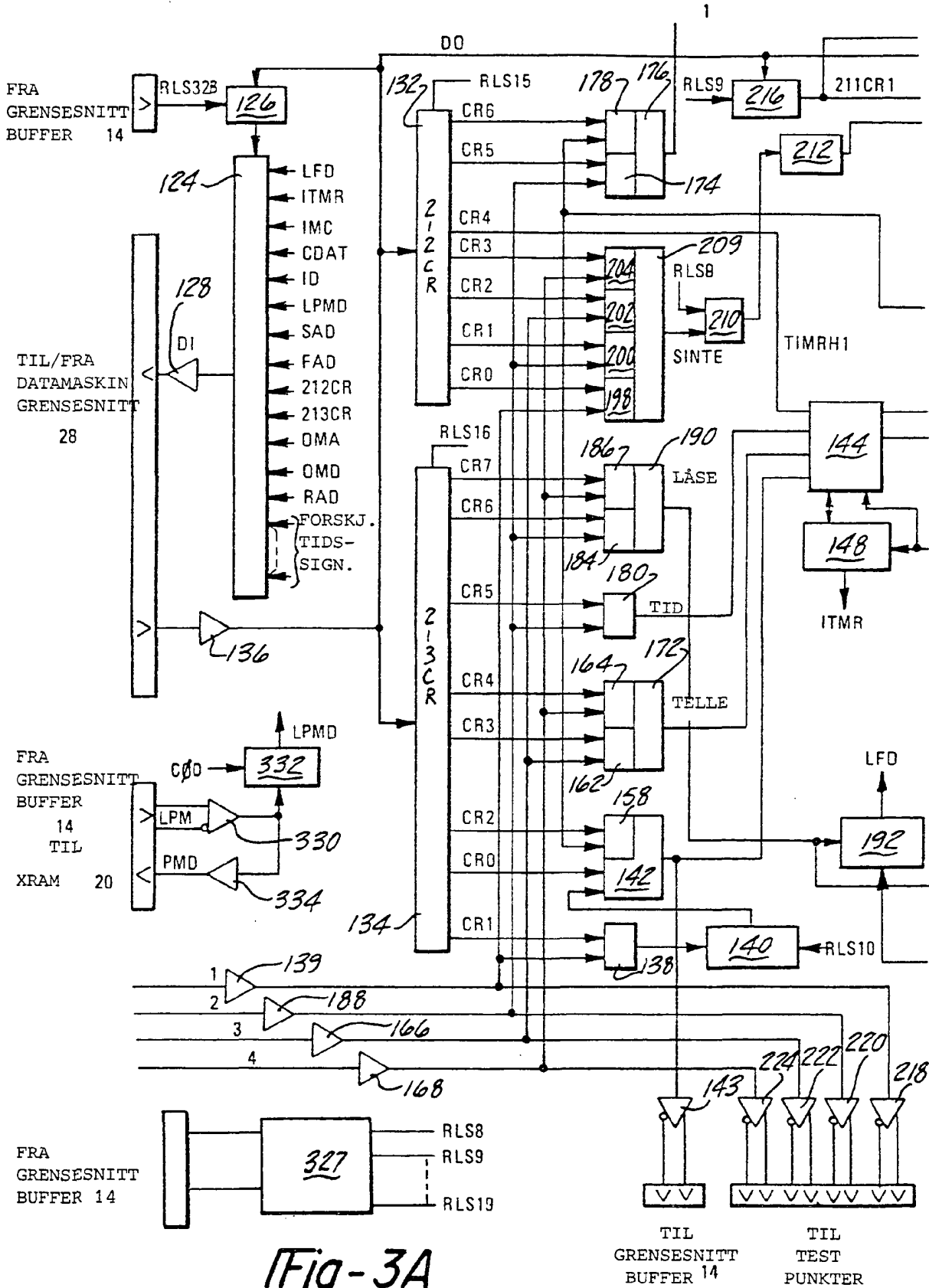


Fig-3A

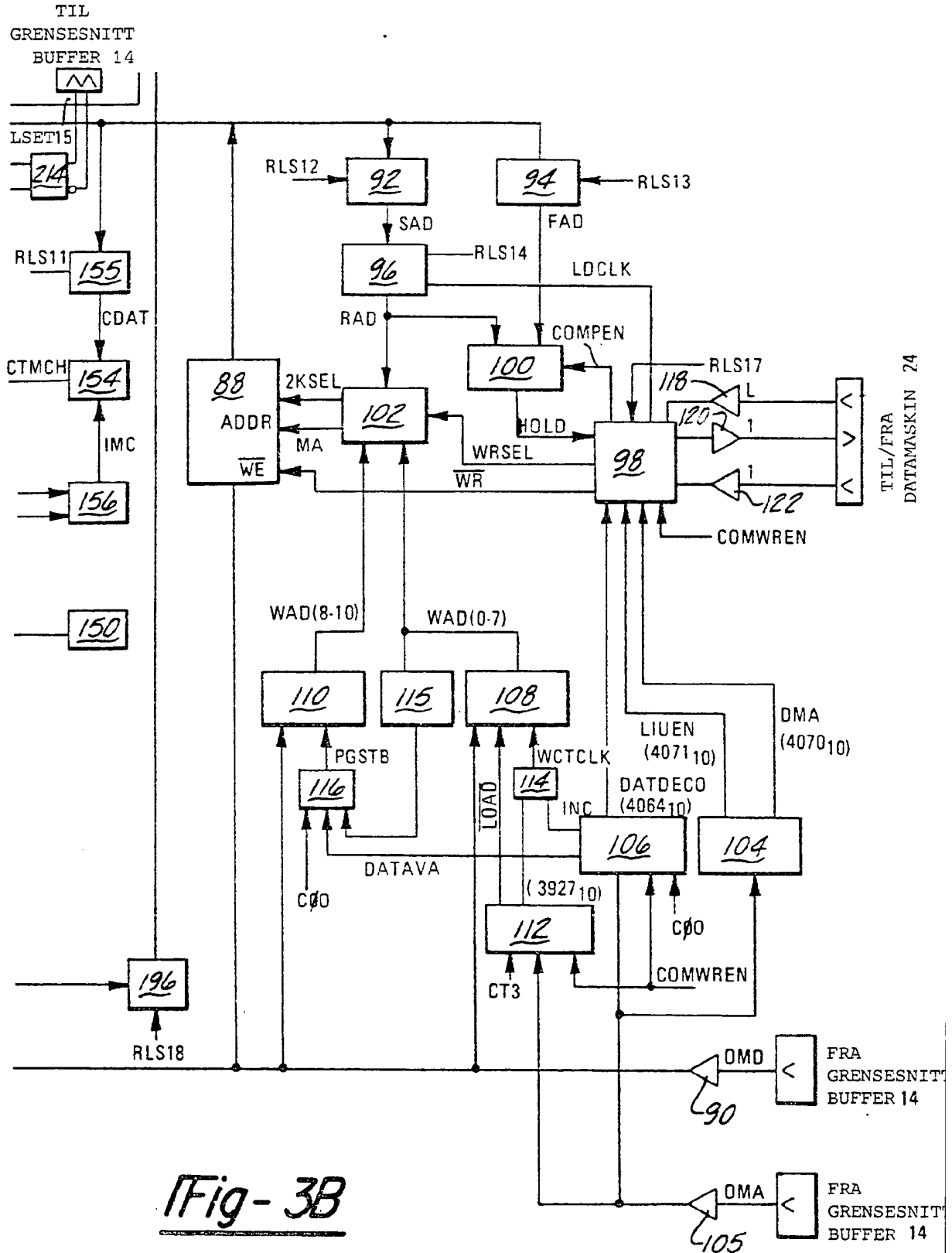


Fig-3B

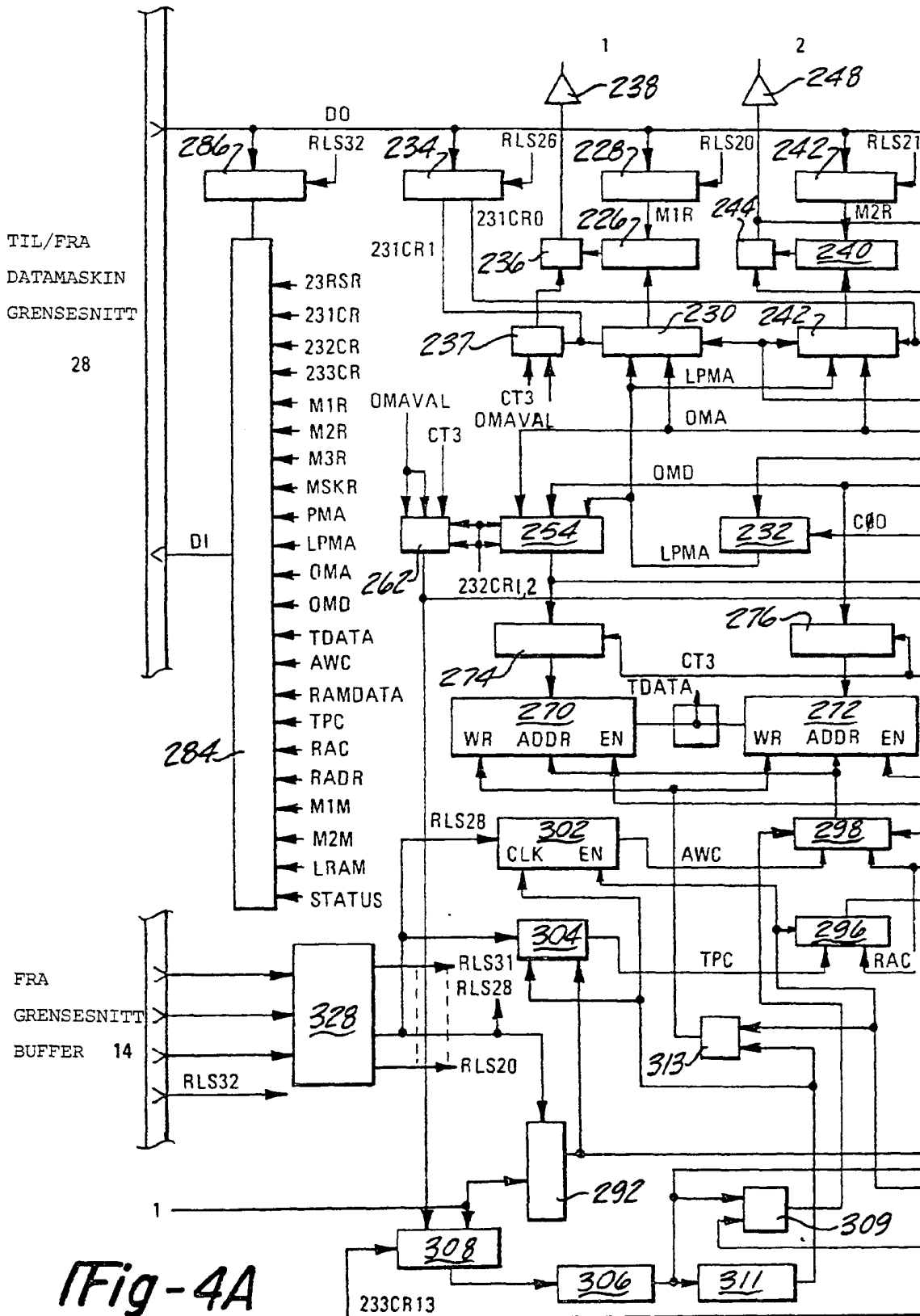
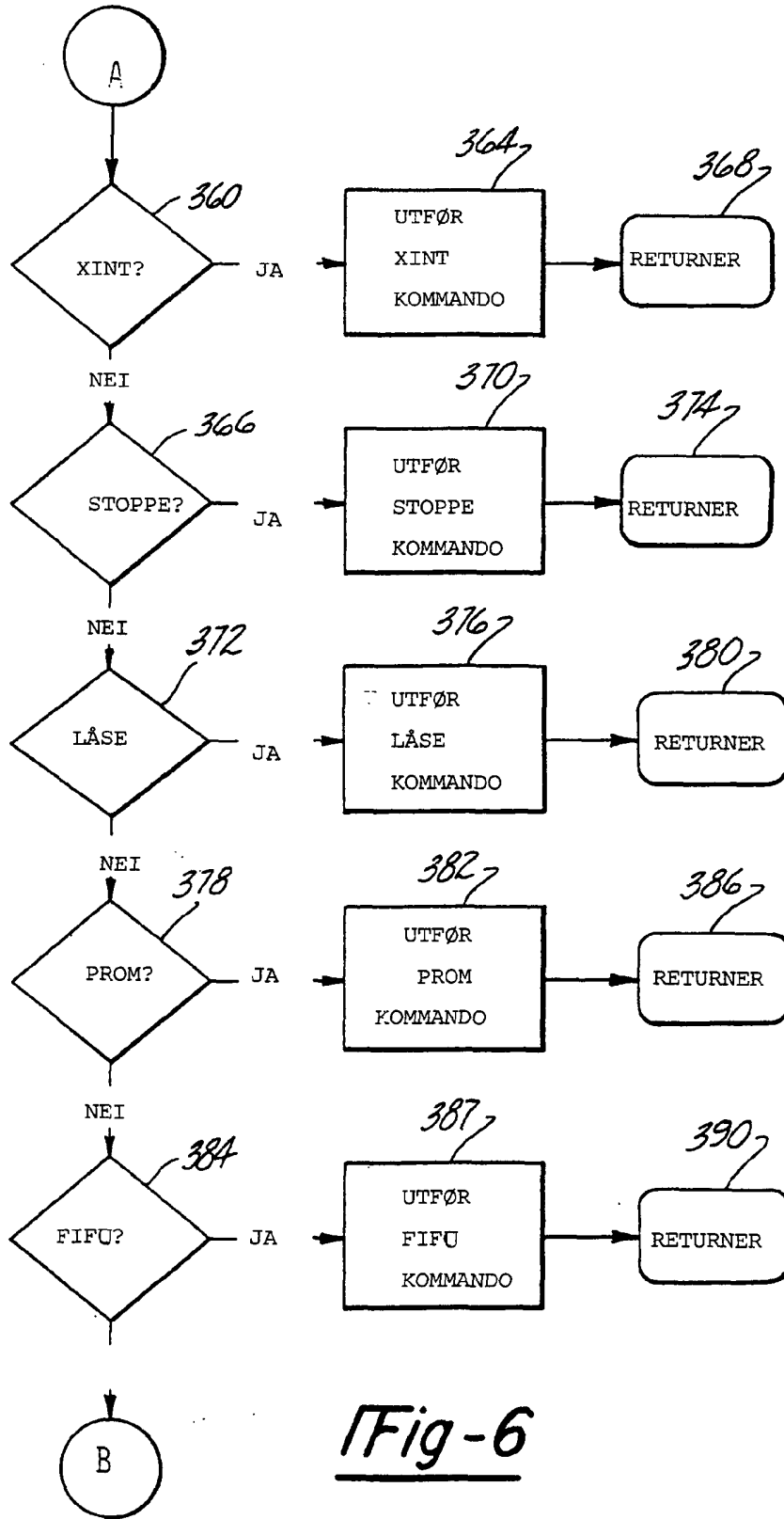


Fig-4A



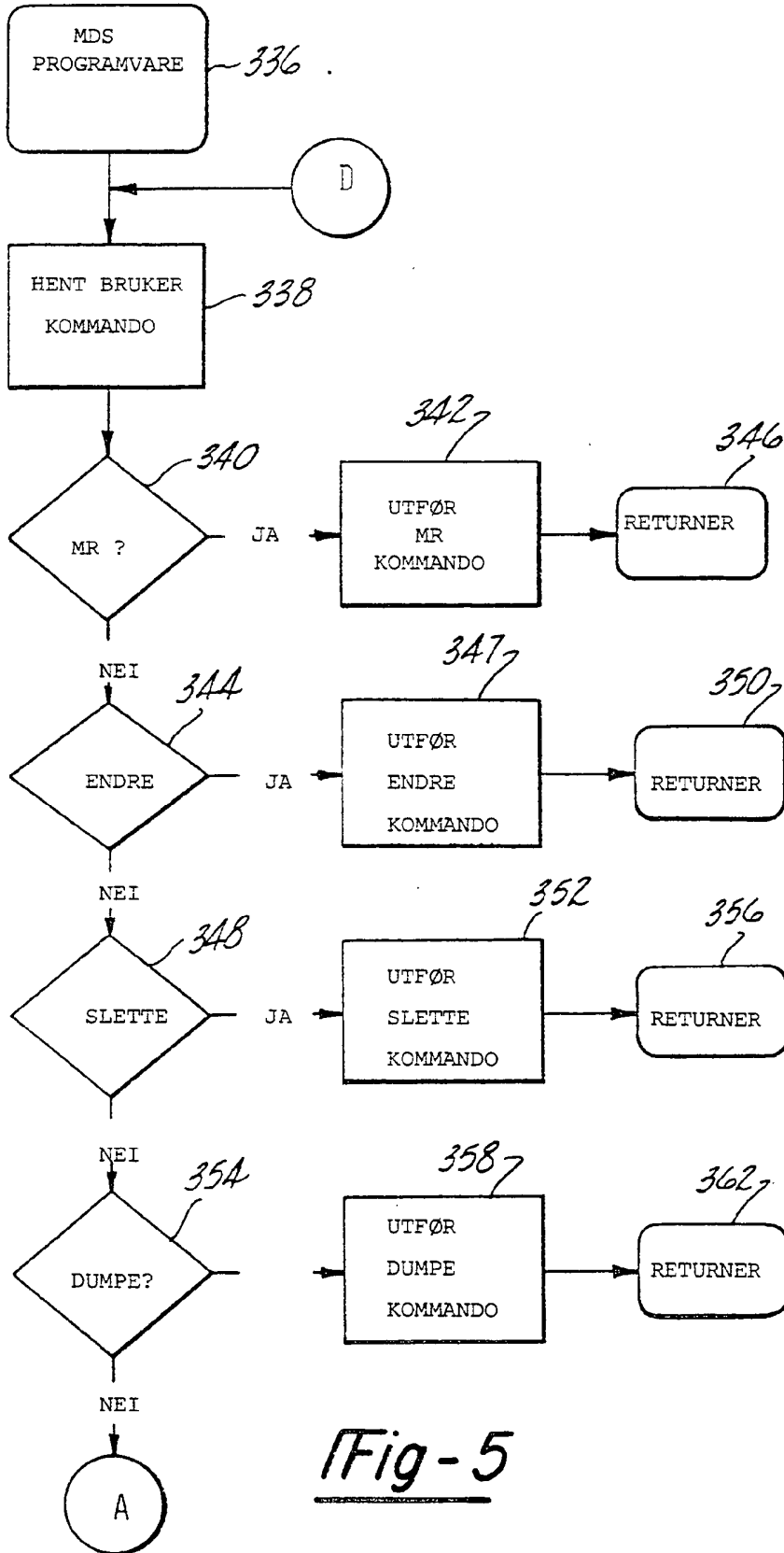


Fig-5

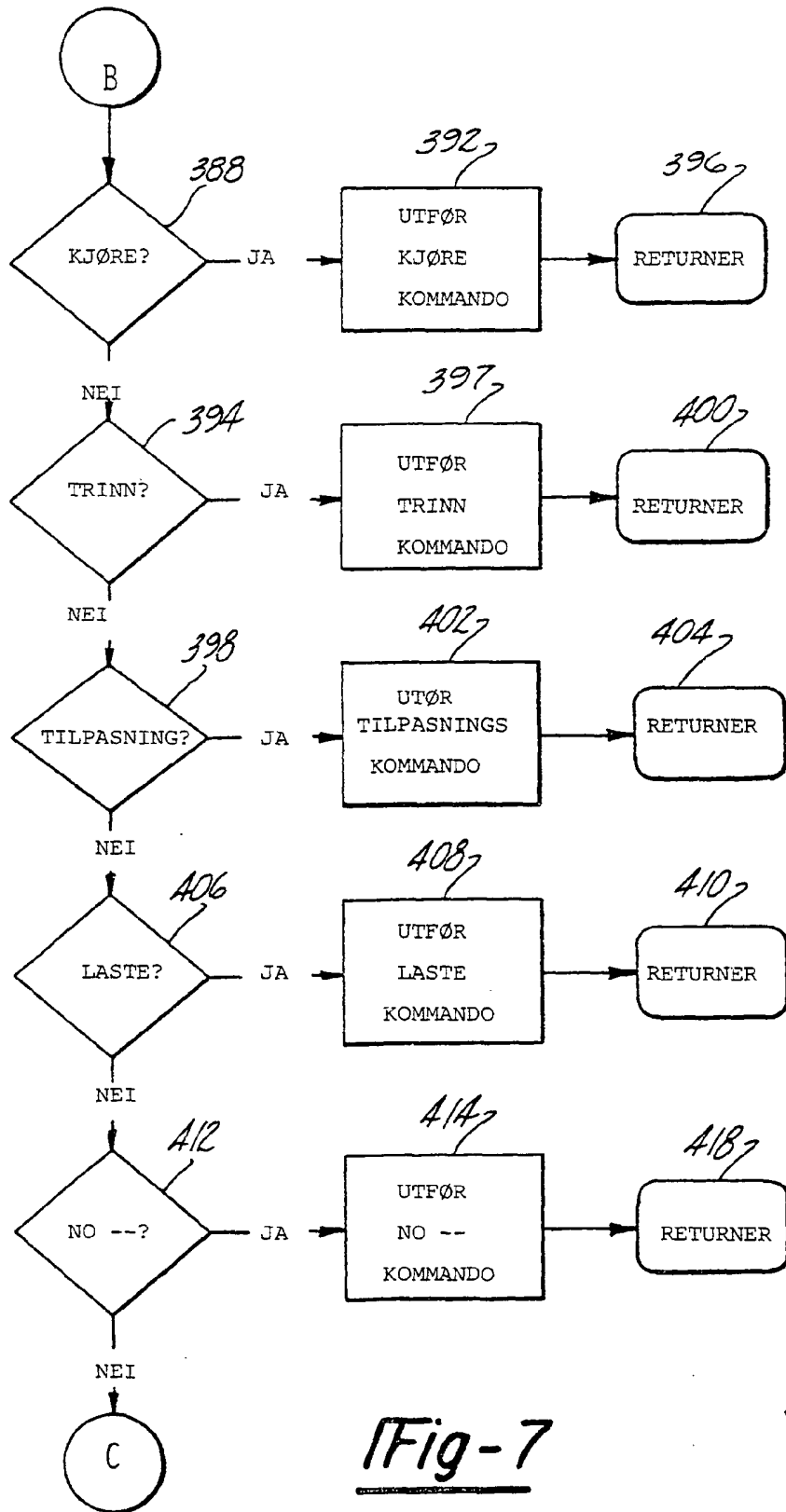


Fig-7

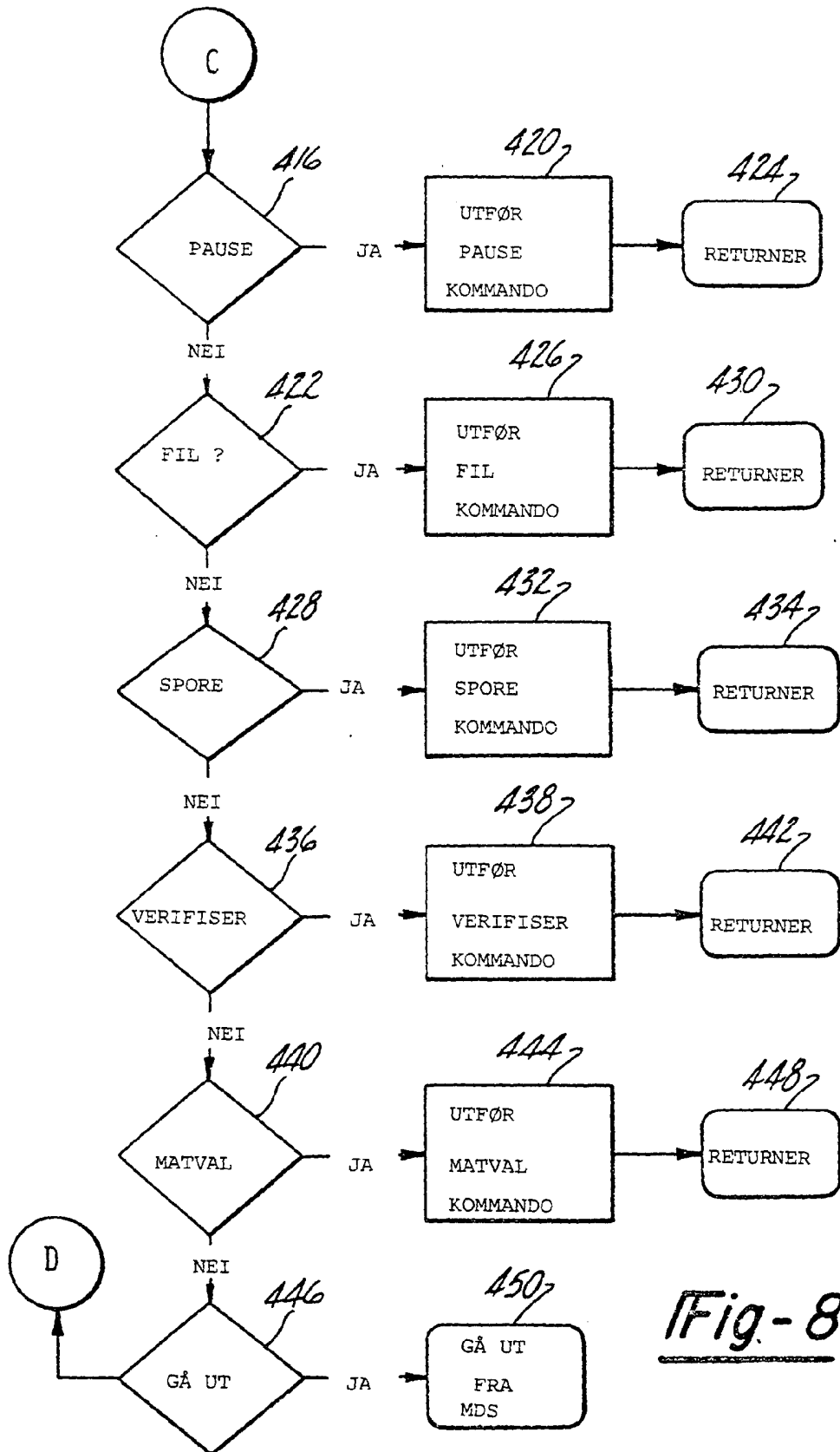


Fig-8