



(12) **UTLEGNINGSSKRIFT**

(19) **NO**

(11) **167946**

(13) **B**

(51) Int Cl⁵ **G 06 F 15/16, H 04 J 3/17**

Styret for det industrielle rettsvern

(21) Søknadsnr 823774
(22) Inng. dag 11.11.82
(24) Løpedag 11.11.82
(41) Alm. tilgj. 13.05.83
(44) Utlekningadag 16.09.91
(62)

(86) Int. inng. dag og søknadsnummer
(85) Videreføringsdag
(30) Prioritet 12.11.81, US, 320244

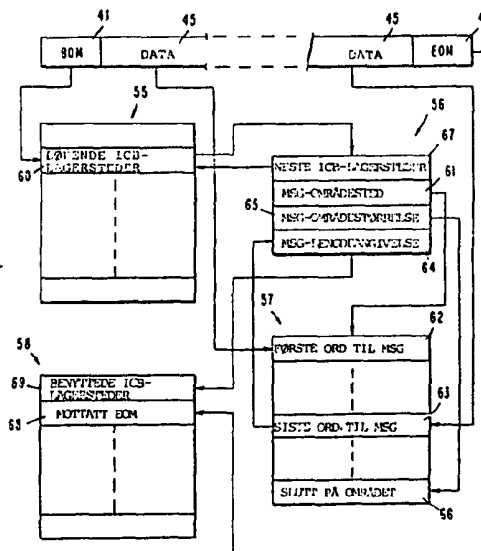
(71/73) Søker/Innehaver Hughes Aircraft Company, 7200 Hughes Terrace, Los Angeles, CA 90045-0066, US
(72) Oppfinner(e) Richard E. Swaney, Fullerton, CA, US
William D. Long, Fullerton, CA, US
(74) Fullmektig Arthur Øvrebø, Bryns Patentkontor AS, Oslo

(54) **Benevnelse Fremgangsmåte for bruk ved en multiplekset databuss**

(56) **Anførte publikasjoner Ingen**

(57) **Sammendrag**

Fremgangsmåte for bruk med en datamaskin som direkte overfører innkommende data i datamaskinens lager. En funksjonsadresse inneholdt i en mottatt melding blir benyttet for å indeksere en posisjonsangiver i en vektortabell. Den indeksede posisjonsangiveren adresserer en inngangsstyreblokk tilknyttet funksjonsadressen. Inngangsstyreblokken blir benyttet for å styre strømmen av data inn i datamaskinens lager. Inngangsstyreblokken styrer også tilgangen til datamaskinens lager for påfølgende meldinger som har samme funksjonsadresse.



Foreliggende oppfinnelse angår fremgangsmåte for bruk ved en multiplekset databuss som angitt i innledningen til krav 1.

Ved foreliggende grunnkommando, styrekommando, 5 kommunikasjons- og intelligensinformasjonssystem resulterer den tradisjonelle hierarkiske arkitekturen under styring av en sentral datamaskin i en ikke-fleksibel systemsammenstilling. En stor prosent av systemets utvikling, anskaffelses- og vedlikeholdskostnader til slike systemer utgjør grense- 10 snittkretsen, kabler og tilkoplinger o.l. Typiske systemspesifikasjoner krever en systemarkitektur som kan fortsette driften i løpet av feil og reparasjoner av enkelte enheter og i løpet av operasjonsmessige omforminger av systemet. Dette nødvendiggjør en sammenkopplingsstruktur som inneholder 15 ingen kritiske knutepunkter eller sentrale styreelementer. Disse kravene kan bli møtt ved å benytte et sammenkopplings-system som anvender en databuss som blir delt av alle enhetene og som innbefatter et standardbuss-grensesnitt for hver enhet.

20 Ved et system i stor målestokk krever tilkoplingen av datamaskiner, prosessorer, fremvisningsanlegg og perifere enheter en unik hierarkalsk anordning hvor funksjonene til en enhet blir bestemt ved dens stilling i sammenkoplingstopologien. Tilknyttede grensesnitt til noen av enhetstypene ville 25 dessuten endre seg avhengig av dets anbringelsessted i topologien. Det vanlige systemet inneholder en sentral datamaskin forbundet med minidatamaskiner gjennom grensesnittenheter, som igjen kommuniserer med perifere anordninger, fremvisningsanordninger og kommunikasjonsutstyr ved hjelp av styrere.

30 For å sikre kontinuerlig drift når feil oppstår ved enheten er det konvensjonelle systemet redundant ved at flere fullstendige systemer er anordnet for å kompensere for enhetsfeil. Når antallet enheter i systemet øker, øker antall punkt-til-punkt-forbindelser med en større hastighet og gjør 35 det således nødvendig med grensesnittenheter, kort og kabler. Sammenstillingen av sammenkoblede elementer i topologien kan variere fra system til system og følgelig er integrasjonen

av disse forskjellige systemene svært komplisert.

For å unngå noen av problemene i forbindelse med disse vanlige systemtypene kan en enkel multiplekset databuss bli anvendt for forbindelse mellom enhetene i systemet.

5 Sentraldatamaskinen kan bli eliminert ved å benytte fordelt behandling. Datamaskinenhetene er nødvendige for å grensesnittforbinde kun med databussen. Enhver datamaskin kan utføre enhver rolle i systemet siden hver datamaskin er forbundet direkte med alle andre datamaskiner, perifere anord-

10 ninger og fremvisningsanordninger ved hjelp av den multipleksede databussen. Fortsatt operasjon i løpet av feil ved en enhet krever kun en ytterligere reserveenhet. Resultatet er reduksjon av typer og antall enheter i systemet og antall grensesnittkort pr. enhet. Anskaffelses- og levetidskostnad-

15 er blir følgelig redusert på grunn av at inventar- og vedlikeholdskravene er redusert.

Når det benyttes en enkel multiplekset databuss, kan en datamaskins bussgrensesnitt godta meldinger adressert til den fra enhver enhet forbundet med bussen til enhver tid.

20 Siden disse meldingene ikke er anmodet om av datamaskinen tilkopleet det adresserte busgrensesnittet, kan ikke ankomsttiden og kilden for meldingen ble bestemt før godtagelsen av meldingen foreligger fra bussen. Når to eller flere motkoplingsmeldinger blir adressert til samme datamaskin, gir

25 ikke mellommeldingsgapet nok tid for en datamaskin til lagring av motsatte meldinger ved bruk av vanlig lagringsteknikk. Slik vanlig teknikk innbefatter utføring av en avbruddsmotagerrutine fulgt av en inngangs-utgangs-driverrutine.

Den vanlige metoden å køføre innkommende meldinger

30 før overføringen til datamaskinens lager. Dette krever overkapasitet for høyhastighetsbuffere. Siden mellomankomst hastigheten og lengden på meldingene er ubestemt, øker sannsynligheten for bufferoverflømming når bithastighetskapasiteten til databussen øker. Vanlige systemer benytter også betrak-

35 telige behandlingsoverskudd ved bevegelse av mottatte meldinger fra ett sted i lageret (bufferområdet) til et annet sted (arbeidsområdet).

Flere teknikker blir nå benyttet ved eksisterende vanlige databusskonstruksjoner for å overføre meldinger til datamaskinlageret. En metode sørger for mottatte meldinger som skal til lageret på et fast sted i et datamaskinlager hvoretter software er nødvendig for å bevege de lagrede meldingene før mottagelse av ytterligere innkommende meldinger. De nylig innkomne meldingene ville ellers overskrive de tidligere mottatte meldingene og således avbryte datastrømmen. En annen teknikk sørger for buffring av én eller flere meldinger i et ytre lager. Når bufferen er full, går overflytmeldinger og påfølgende mottatte meldinger tapt. Meldingskilden blir informert om tapet av sendte meldinger og dirigert til å sende igjen etter en tidsforsinkelse. En tredje teknikk tilveiebringer overføring av et styreord som reserverer rom i lageret før sendingen av datameldingen. Dette sikrer at riktig mengde med lagerrom blir tildelt meldingen som skal bli sendt. Denne metoden krever imidlertid overføring av flere meldinger for å lagre data. Denne metoden krever også behandlingstid for å sette opp et lagerområde for data som senker operasjonshastigheten.

US-patent nr. 4 133 030 beskriver et system for overføring av data mellom et hovedlager ved databehandlings-system og kommunikasjonskanaler. Dette blir gjort under styring av kommunikasjonsstyreblokkene tilveiebrakt i et hjelpelager. Hovedlageret innbefatter datablokker. Hver styreblokk innbefatter en startadresse og et område og statusinformasjon som er anvendt ved overføring av data til blokken i hovedlageret. Sentralprosessen styrer lastingen av styreblokkene i hjelpelageret. Disse ble anvendt ved dataprosessen for å holde banen til dataoverføringsoperasjonen. Overføring av data blir utført ved styreblokkene som anvender så mange styreblokker som er nødvendig for hver kommunikasjonskanal. Startadressen er ikke en funksjonell adresse. Sentralprosessen retter lastingen av styreblokker inn i hjelpelageret.

US-patent nr. 4 155 119 beskriver en anordning ved hvilken en sentral behandlingsenhet og en inngangs-/utgangs-

enhet er koblet for toveis kommunikasjon, og er toveis-koblet med et lager via en virtuell adresseomsetter. Lageradressene til dataordene som skal bli overført til eller fra sentral-behandlingsenheten (digital datamaskin) er eksternt spesifisert ved anordninger forbundet med inngangs-/utgangsenheten. Ved behandling av eksterne spesifiserte adresser fra anordningen som virtuell adresse opererer sentralprosessor-datamaskinen innenfor datamaskinens virtuelle adresserom. Dette tillater inngangs-/utgangsbehandlingene å bruke adressekartsystemet til datamaskinen. Den anvendte metoden anvender her eksterne adresseringer av data i lageret for å adressere et spesifikt sted i lageret. Dette er en punkt-til-punkt-adressering.

En artikkel i "IBM Technical Disclosure Bulletin" av R.E. Birney m.fl.: "Suppress Exception Mechanism For Input/Output Devices", vol. 20, nr. 11b, april 1978 beskriver en anordning ved hvilken en inngangs-/utgangsstyreenhet kan rette overføringen av data mellom seg selv og hovedlagerenheten til prosessoren ved hjelp av en direkte tilgangslagermekanisme av periodisk-ta-typen. Styreanordningsblokkene i hovedlageret til prosessoren styrer den periodisk-ta-overføringen av data. Hver styreblokkanordning har sitt eget datalagerareal og reststatusblokk i hvilken statusinformasjonen blir satt ved fullførelse eller avslutning av dataoverføringsoperasjonen. Anordningsstyreblokkene kan bli kjedet sammen om nødvendig. Punkt-til-punkt ekstern adressering blir anvendt. Prosessoren tilfører styreenheten adressen til den første anordningsstyreblokken.

Ingen av disse publikasjonene beskriver et system som har følgende trekk:

1. Funksjonell adressering til forskjell fra punkt-til-punkt-adressering. Punkt-til-punkt-adresseringen vil ikke virke i sammenheng med foreliggende oppfinnelse.
2. Avføling av meldingen når den forplanter seg forbi en dataprosessor slik at den kan fortsette på bussen til

andre prosessorer. Ved punkt-til-punkt-adresseringen blir meldingen mottatt ved kun et sted.

3. Bestemmelsen av den funksjonelle adressen om eller ikke meldingen skal aksepteres.
4. Anvendelse av en akseptert funksjonell adresse for å indeksere en vektortabell for å peke mot en inngangsstyreblokk.
5. Bestemmelse av dataprosessoren ved hjelp av en inngangsstyreblokk som den har blitt pekt mot, lokalisering i lageret til den dataprosessoren ved hvilken dataen skal bli lagret.
6. Lagring av data i de lagerstedene som datameldingen på bussen er blitt avfølt.

US-patent nr. 4 084 228 og EP-patentpublikasjon nr. 36766 viser også eksempler på kjent teknikk.

Det ville derfor være en forbedring ved databehandling å ha en metode for direkte lagring av mottatte meldinger i en datamaskins lager ved egnet behandlingsområde for således å unngå unødvendig bevegelse av meldinger i lager fra et innkommende bufferområde til et behandlingsområde.

Det ville også være en forbedring i forhold til det tidligere kjente å tilveiebringe en metode som unngår behovet for forsinket gjentakelse av meldinger mens datamaskinen prepareres og innstilles for mottakelse av meldingen.

For å overvinne problemene med tidligere kjente lagringstilgangsmetoder er det tilveiebrakt en fremgangsmåte av den innledningsvis nevnte art hvis karakteristiske trekk fremgår av krav 1. Ytterligere trekk ved oppfinnelsen fremgår av de øvrige uselvstendige kravene. Metoden kan bli anvendt ved en fordelt, styrt, multiplexet databuss som har flere databehandlingsenheter koplet dertil.

Metoden sørger for direkte adressering av lager til en databehandlingsenhet for å lagre en mottatt melding ved

et forutbestemt sted i det lageret. En funksjonsadresse inneholdt i den mottatte meldingen blir benyttet for å dirigere lagringen av meldingen i lageret. Hver databehandlingsenheter inneholder et bestemt sett med softwarefunksjoner
5 (programmer) som behandler data i en mottatt melding i løpet av deres utførelse. Funksjonsadressen identifiserer softwarefunksjonen som benytter data inneholdt i den mottatte meldingen. Hver softwarefunksjon blir påvirket av mottagelsen av en melding som inneholder funksjonsadressen korresponderende med den bestemte softwarefunksjon.
10

En softwarefunksjon blir skrevet på en måte som blir etablert ved inngangsstyreblokken og en vektortabell som blir benyttet for å lagre en mottatt melding i et forutbestemt område av lageret. Inngangsstyreblokken og
15 vektortabellen tilveiebringer de nødvendige styreord og posisjonsgivere som bevirker fortsatt funksjon av softwarefunksjonen så snart data har blitt mottatt.

Den mottatte meldingen er innbefattet av funksjonsadressen og dataordene. Funksjonsadressen identifiserer
20 softwarefunksjonen som skal benytte dataen. Denne funksjonsadressen blir benyttet av en direkte lagertilgangs- (DMA) -styrer for å adressere vektortabellen. Vektortabellen peker på inngangsstyreblokken betegnet med det adresserte ord. DMA-styreren benytter vektortabellen og inngangsstyreblokken for å bestemme ut fra funksjonsadressen hvor
25 dataordene i den mottatte meldingen skal lagres.

Vektortabellen er et lagret sett med forutbestemte lagersteder for inngangsstyreblokkene som er søkt tilgang ved hjelp av DMA-styreren. Inngangsstyreblokken er et lagret
30 sett med kommandoord som hjelper DMA-styreren med å bestemme hvor data skal lagres i lageret.

Metoden ifølge foreliggende oppfinnelse omfatter mottak av en melding som inneholder funksjonsadressen.

For å utføre foreliggende metode har databehandlingsenheten et bussgrensesnitt, en DMA-styrer, en behandlingsenhet og et lager. DMA-styreren benytter vektortabellen og en inngangsstyreblokk for å overføre en melding mottatt
35

fra bussgrensesnittet til lageret. Meldingen er innbefattet av en begynnelse av meldingsordet som inneholder funksjonsadressen fulgt av dataord og avsluttet med slutten av et meldingsord.

5 Når bussgrensesnittet godtar en melding fra databussen, overfører DMA-styreren meldingen til lageret som benytter funksjonsadressen inneholdt i begynnelsen av meldingsordet. Funksjonsadressen indekser en entring i vektortabellen. Entringen i vektortabellen identifiserer en inngangs-
10 styreblokk tilknyttet den bestemte funksjonsadressen. Inngangsstyreblokken inneholder kommandoord benyttet av DMA-styreren for å lagre data i lageret. Funksjonsadressen tilsvarende softwarefunksjonen som skal behandle den mottatte data.

15 Det første dataordet i meldingen blir lagret i et forutbestemt lagersted bestemt av et ord i inngangsstyreblokken. Påfølgende ord i meldingen blir lagret i lagersteder nærliggende det første lagerstedet inntil det siste ordet i meldingen er mottatt. Lagringen fortsetter alternativt inntil et maksimalt antall ord har blitt mottatt som
20 fyller det tildelte lagerområdet.

DMA-styreren begrenser ved hjelp av inngangsstyreblokken størrelsen på lagerområdet benyttet for å lagre en bestemt melding og lagrer en indikasjon på lengden av den
25 mottatte meldingen for bruk av softwarefunksjonen. Inngangsstyreblokken sørger for en inngang til vektortabellen for å oppdatere styreblokkindeksen tilveiebragt derved. Denne virkningen tillater DMA-styreren å identifisere den neste inngangsstyreblokken som skal bli nyttet for samme funksjons-
30 adresse. Slutten av meldingsordet blir lagret i en inngangsmelding avslutning kjø anordnet i lageret for å angi til softwarefunksjonen at data har blitt lagret.

Anvendelsen av den ovenfor beskrevne metode tillater direkte tilgang til et lager ved en databehandlings-
35 enhet uten avbrudd ved pågående softwareutføelse. Ovenfor nevnte metode tillater direkte lagertilgang som sikrer at behandlingseenheten kan motta meldinger så hurtig som de blir

mottatt over databussen. Denne metoden reduserer administrasjonsbehandlingen i behandlingsenheten og unngår unødvendig bevegelse av meldinger fra et innkommende bufferområde til et behandlingsområde. Denne metoden eliminerer også behovet
5 for forsinket mottagelse av en melding mens behandlingsenheten setter opp et område for mottagelse av innkomende data.

Oppfinnelsen skal nå beskrives nærmere med henvisning til medfølgende tegninger, hvor:

Fig. 1 viser fordelt styre- multiplekset databuss
10 som kan anvende metoden ifølge foreliggende oppfinnelse.

Fig. 2 viser prosessen med å frembringe en melding som blir overført over databussen.

Fig. 3 viser sammensetningen av en typisk melding som blir sendt over databussen.

15 Fig. 4 viser et diagram som kan anvendes for å vise fremgangsmåten ifølge foreliggende oppfinnelse.

Fig. 5 og 6 viser flytdiagram som viser fremgangsmåten ifølge oppfinnelsen.

På fig. 1 er vist en multiplekset databuss hvor
20 foreliggende oppfinnelse kan bli anvendt. Systemet innbefatter en seriedatabuss 21 som er koplet til flere enheter, slik som datamaskiner, fremvisningsanordninger og perifere anordninger. Flere datamaskiner 22-1, ... 22-N, flere fremviser-anordninger 23-1, ... 23-N, og flere perifere anordninger 24-1,
25 ... 24-N kan f.eks. være forbundet med databussen 21. Databussen 21 kan være en vanlig koaksial eller triaksial kabel med en midtleder omgitt av én eller to jordede skjjermer og egnede transformatorkoplinger anordnet for å forbinde datamaskiner, fremvisningsanordninger og perifere anordninger
30 med denne. Datamaskinene 22 forbundet til databussen 21 kan være av samme type eller være av forskjellige typer. Fremvisningsanordningene 23 kan være tekstanordninger, grafiske anordninger eller store skjjermer eller lignende. De perifere anordninger 24 innbefatter slikt utstyr som terminaler
35 (CRT/tastaturer), modemer, skrivere, diskenheter og magnetiske båndenheter eller lignende.

Hver av enhetene forbundet med databussen 21 over-

fører informasjon langs bussen 21 til hvilke som helst av de andre enhetene. Den første datamaskinen 22-1 kan f.eks. overføre informasjon til en av de andre datamaskinene koplet til bussen 21 eller overføre informasjon til enhver av de 5 perifere anordningene 24 eller noen av fremvisningsanordningene 23. Ved en stor systemsammenstilling er det mulig for flere enheter å overføre data langs databussen 21 til én enhet ved hovedsakelig samme tidspunkt. Et riktig konstruert system tillater mottagelse av en rekke meldinger som har 10 lite eller overhode ikke noe gap derimellom uten tap av noen form for data.

Fig. 2 viser behandlingen hvor en melding blir frembragt ved en vanlig sendeenhet. En datamaskin, slik som den første datamaskinen 22-1, er innbefattet av et lager 31-1, 15 en behandlingsenhet 32-1, en direkte lagertilgangs(DMA)-styrer 33-1 og et bussgrensesnitt 34-1. Behandlingsenheten 32-1 utfører et program 35 og en inngangs/utgangsinstruksjon gir DMA-styreren 33-1 stedet for utgangsstyreblokken 38 i lageret 31-1. Når bussgrensesnittet 34-1 overfører en melding 20 39 på databussen 21, starter meldingen 39 med en synkropuls (S) 40 fulgt av begynnelsen av et meldingsord (BOM) 41, dataord 45, en synkropuls (S) 40' og slutten av et meldings- (EOM) ord 44.

DMA-styreren 33-1 gjenfinner begynnelsen av meldingsordet 41 fra utgangsstyreblokken 38 for å danne den 25 første delen av meldingen 39. DMA-styreren 33-1 gjenfinner så startadressen 42 fra utgangsstyreblokken 38. Startadressen 42 angir hvor dataordene 45 er lagret i lageret 31-1 og hvor mange dataord (ordtelleverdier 43) skulle bli overført. Etter begynnelsen av meldingsordet 41 gjenfinner data- 30 bussgrensesnittet 34-1 dataordene 45 fra lageret 31-1. DMA-styreren 33-1 gjenviner så til slutt slutten av meldingsordet 44 fra utgangsstyreblokken 38 for å danne siste del av meldingen 39.

35 Meldingen 39 blir mottatt av en bestemt databasehandlingsenhet, identifisert som datamaskinen 22-N. Bussgrensesnittet 34-N til mottagerdatamaskinen 22-N mottar den

sendte meldingen 39 og bestemmer om denne bestemte enheten skal motta den sendte data heri. Fig. 3 viser også strukturen til den mottatte melding 39. Meldingen 39 omfatter spesielt den første synkropulsen 40 fulgt av begynnelsen av meldingsordet 41 som inneholder en type kode 50 og en funksjonsadresse 49. Dataordene 45 følger begynnelsen av ordet 41. Den andre synkropulsen 40' følger dataordene 45, hvorefter slutten av meldingsordet 44 avslutter meldingen 39.

Fig. 4 viser et diagram nyttig for å vise fremgangsmåten ifølge foreliggende oppfinnelse. Fremgangsmåten ifølge foreliggende oppfinnelse tillater en direkte adressering av et datamaskinlager for å lagre meldinger deri. Som forklart ovenfor inneholder meldingen 39 en funksjonsadresse 49 nær dens begynnelse. Så snart et bestemt bussgrensesnitt 34 godtar meldingen 39, blir funksjonsadressen 49 benyttet av DMA-styreren 33 som en indeks i en vektor-(posisjonsangiver) tabell 55. Adressen i vektortabellen 55 peker på en inngangsstyreblokk (ICB) 56 som korresponderer med den bestemte funksjonsadressen 49. Vektortabellen 55 er et lagret sett med forutbestemte lagersteder til inngangsstyreblokkene 56 som blir gitt tilgang av DMA-styreren 33. Inngangsstyreblokken 56 er et lagret sett med kommandoord som hjelper DMA-styreren 33 ved å bestemme hvor mottatte data skal lagres. Informasjonen lagret i inngangsstyreblokken 56 angir hvor data 45 skal lagres i lageret 31, om eller ikke behandlingsenheten 32 skal avbrytes etter at data 45 er lagret og hvorledes den skal settes opp for neste innkommende melding 39 med samme funksjonsadresse 49.

På fig. 4 er meldingen 39 vist avtegnende dens forskjellige deler som er av spesiell viktighet for fremgangsmåten ifølge foreliggende oppfinnelse. En software-funksjon (program) som blir utført ved databehandlingsenheten er skrevet på en måte som etablerer inngangsstyreblokken 56 og vektortabellen 55 benyttet for å lagre en mottatt melding 39 i lageret. Funksjonsadressen 49 inneholdt i meldingen 39 blir benyttet av DMA-styreren 33 for å indeksere vektortabellen 55. Funksjonsadressen 49 blir tilført vektor-

tabellen 55 som identifiserer lagerstedet 60 til den bestemte inngangsstyreblokken slik som inngangsstyreblokken 56. Inngangsstyreblokken 56 er en som er utpekt for en bestemt funksjonsadresse 49.

- 5 Så snart identifisert, benytter DMA-styreren 33 inngangsstyreblokken 56 for å styre strømmen av data 45 fra den mottatte meldingen 39 inn i inngangsmeldingsområdet 57 til lageret 31. DMA-styreren 33 lagrer det første dataordet til meldingen 39 ved et første lagersted 62 i inngangsmeldingsområdet som blir identifisert av et styreord merket meldings(MSG) -områdelagersted 61 i inngangsstyreblokken 56. DMA-styreren 33 fortsetter å overføre data fra meldingen 39 inn i inngangsmeldingsområdet 57 på nærliggende lagersteder inntil det siste ordet 63 til meldingen 39 har blitt mottatt.
- 10 Slutten av meldingsordet 44 blir detektert, som bevirker DMA-styreren 33 for å lagre slutten av meldingsordet 44, i et lagersted 68 til en inngangsmeldingsavslutningskø 58 og avslutter overføringen av dataord til inngangsmeldingsområdet 57. DMA-styreren 33 lagrer også informasjon angående lengden på mottatt melding i et lagersted 64 til inngangsstyreblokken 56 slik at softwarefunksjonen som benytter data 45 har en indikasjon av hvor meldingsendene er.

- Antall dataord 45 i mottatt melding 39 er normalt lik eller mindre enn størrelsen på inngangsmeldingsområdet 57. Dersom antall ord for dataene 45 inneholdt i mottatt melding 39 er større enn antall ord inneholdt i inngangsmeldingsområdet 57, fortsetter DMA-styreren 33 å lagre data i inngangsmeldingsområdet 57 kun inntil området 57 er fullt. DMA-styreren 33 benytter informasjonen lagret i lagerstedet 65 for å bestemme slutten av området 57. DMA-styreren 33 stopper lagringen av data når den når ordet i lagerstedet 66. DMA-styreren 33 avbryter lagringen av mottatte dataord dersom meldingen overskrider størrelsen på inngangsmeldingsområdet 57. Som nevnt tidligere blir informasjon angående lengden på den mottatte melding 39 lagret i inngangsstyreblokken 56 av DMA-styreren 33 og slutten av meldingsordet 44 blir lagret i inngangsmeldingsavslutningskøen 58. I begge

tilfeller blir stedet for den benyttede inngangsstyreblokk 56 lagret i inngangsmeldingsavslutningskøen 58 av DMA-styreren 33. Det hjelper softwarefunksjonen i å bestemme om meldingen 39 har blitt mottatt og hvilken styreblokk som
5 ble anvendt for å lagre data 45.

DMA-styreren 33 lagrer ord i inngangsmeldingsavslutningskøen 58 som utpeker lagerstedet 69 som en angivelse av den benyttede inngangsstyreblokken 56 og lagerstedet 68 som slutten av meldings- (EOM) ordet 44 for hver mottatt
10 melding 39. Inngangsmeldingsavslutningskøen 58 blir kontinuerlig oppdatert hvor den eldste entringen blir ødelagt ved ankomsten av den nyeste entringen. Inngangsmeldingsavslutningskøen 58 holder orden på tidligere mottatte meldinger med hensyn til den bestemte inngangsstyreblokken benyttet
15 for å lagre meldingen og slutten av meldingsordet.

Ordet i inngangsstyreblokken 56, som blir identifisert som neste inngangsstyreblokk-(ICB) lagersted 67 ble lagret i vektortabellen 55 ved hjelp av DMA-styreren 33. Dette sikrer at neste melding med samme funksjonsadresse
20 har et gyldig indeks (posisjonsangivelse) for styringen av lagringen av data inneholdt deri. Dette tillater multippel-inngangsstyreblokker å bli kjedet sammen eller at samme samme inngangsstyreblokk blir kontinuerlig benyttet.

Utførelsen beskrevet med henvisning til fig. 4
25 unngår den unødvendige bevegelsen av meldinger i lageret fra et innkommende bufferområde til et behandlingsområde. Behovet for forsinket mottagelse av meldingen mens behandlingsenheten blir innstilt for å motta blir også unngått. Direkte lagring av meldingen ved bruk av funksjonsadressen eliminerer behovet for at behandlingsenheten skal sortere ut
30 meldinger. Dette systemet tillater også at innkommende meldinger direkte oppdaterer en databussfil på periodisk basis uten å skape behandlingsadministrasjon eller systemavbrudd.

Fig. 5 viser en generell metode i flytdiagramform.
35 Metoden innbefatter trinnet med å motta en melding som inneholder en funksjonsadresse overført langs en databuss, som angitt i blokk 75. Funksjonsadressen inneholdt i den mot-

tatte meldingen benyttes for å indeksere en vektortabell, som angitt i blokk 76. Vektortabellen adressert på denne måten sørger for adressen til en inngangsstyreblokk som korresponderer med funksjonsadressen, som angitt i blokk

5 77. Inngangsstyreblokken angir lagerstedet for lagring av data inneholdt i den mottatte meldingen, som angitt i blokken 78. Data inneholdt i meldingen blir så til slutt lagret i lagerstedet, som angitt i blokken 79.

Oppfinnelsen skal forklares nærmere med henvisning til fig. 6. Detaljerte trinn ved fremgangsmåten ifølge oppfinnelsen innbefatter mottagelse av meldingen fra bussgrensesnittet, som innbefatter godtagelse av meldingen og føring av funksjonsadressen til DMA-styreren, som angitt i blokk 75. DMA-styreren benytter funksjonsadressen for å indeksere vektortabellen, som angitt i blokk 76. Vektortabellen adresserer (peker mot) en inngangsstyreblokk som inneholder informasjoner angående hvor data skal lagres, som angitt i blokk 77. DMA-styreren lagrer det første ordet for data inneholdt i meldingen ved det forutbestemte lagerstedet angitt med et ord i inngangsstyreblokken, som angitt i blokken 80. Påfølgende dataord i meldingen blir lagret i lagerstedene nærliggende det første ordet inntil lagerområdet er fullt eller hele meldingen har blitt lagret, som angitt i blokk 81.

25 Også en angivelse av lengden på den mottatte meldingen blir lagret i inngangsstyreblokken for å sørge for at softwarefunksjonen som benytter datene får angitt hvor dataene slutter, som angitt i blokken 82. Slutten av meldingsordet og stedet for inngangsstyreblokken benyttet for lagret data blir lagret i en meldingsavslutningskø for å angi for softwarefunksjonen at data har blitt lagret, som angitt i blokk 83. Stedet for den neste inngangsstyreblokken som skal bli benyttet når neste melding med samme funksjonsadresse er mottatt, blir også lagret i vektortabellen, som angitt i blokk 84. Det siste trinnet tillater at flere inngangsstyreblokker kan bli kjedet sammen eller samme inngangsstyreblokk kan bli kontinuerlig benyttet.

167946

14

Det har blitt beskrevet en ny fremgangsmåte for å benytte direkte lagertilgang for å lagre meldinger mottatt fra databehandlingsenheter forbundet med en multiplekset databuss. Foreliggende oppfinnelse unngår unødvendig be-
5 vegelse av meldinger i lageret fra et innkommende bufferområde. Foreliggende metode unngår også behovet for å forsinke mottagelsen av innkommende meldinger mens en datamaskin prepareres for mottagelse og lagring av meldingene.

10

15

20

25

30

35

P a t e n t k r a v

1.

5 Fremgangsmåte for bruk ved en multiplekset databuss (21) som har flere tilkoblede databehandlingsenheter som sørger for direkte adressering av respektive datamaskinlagre tilknyttet hver av databehandlingsenhetene for å lagre deri mottatte datameldinger (39), idet databehandlingsenheten innbefatter en vektortabell (55) og en inngangsstyreblokk (56) som er
10 etablert av software-funksjoner utført av databehandlingsenhetene, idet datameldingen (39) innbefatter en funksjonsadresse (49) som identifiserer databehandlingsfunksjonen som skal bli utført på data som følger funksjonsadressen i datameldingen og data som skal bli overført til lageret,
15 k a r a k t e r i s e r t v e d avføling (34-N, fig. 2, fig. 4) av datameldingen (39) som inneholder funksjonsadressen (49) ettersom datameldingen forplanter seg på bussen (21),
bestemmelse (54, fig. 4) om data (45) skal bli akseptert når
20 funksjonsadressen (49) blir avfølt, anvendelse (DMA styrer 33-N, fig. 2) av funksjonsadressen (49) som en indeks i vektortabellen (55, fig. 6), idet funksjonsadressen (49) peker på en inngangsstyreblokk (60) korresponderende med den bestemte funksjonsadressen (49),
25 adressering (60 ved 55 til 56) av inngangsstyreblokken (56) designert av funksjonsadressen (49), anvendelse (DMA styrer 33-N, fig. 2) av inngangsstyreblokken (56) for å designere et datamaskinlagersted (57) hvor data (45) ved en mottatt datamelding (39) skal bli lagret (62-63),
30 og lagring (DMA styrer 33-N, fig. 2) av dataen (45) inneholdt i datameldingen (39) i datamaskinlagerstedet (57 til lager 31-N, fig. 2) designert av inngangsstyreblokken (56) når datameldingen på bussen (21) avføles.

2.

35 Fremgangsmåte ifølge krav 1, k a r a k t e r i s e r t v e d oppdatering (DMA styrer 33-N, fig. 2) av vektortabellen (55, fig. 6) med informasjon (67) utledet fra inngangs-

167946

16

styreblokken (56) som er indikativ for lokaliseringen av en inngangsstyreblokk som skal bli benyttet ved neste melding (39) som inneholder funksjonsadressen (49, fig. 3).

5 3.

Fremgangsmåte ifølge krav 1 og 2, k a r a k t e r i - s e r t v e d lagring (DMA styrer 33-N, fig. 2) av informasjonen (63 ved 64) som er indikativ for lengden på meldingen (39) for bruk av software-funksjonen.

10 4.

Fremgangsmåte ifølge krav 1, 2 eller 3, k a r a k t e r i - s e r t v e d lagring (DMA styrer 33-N, fig. 2) av informasjon (69) som er indikativ for hvilken inngangsstyreblokk (56) som ble benyttet for å lagre den mottatte meldingen (39).

5.

20 Fremgangsmåte ifølge krav 1, 2, 3 eller 4, k a r a k - t e r i s e r t v e d lagring (DMA styrer 33-N, fig. 2) av informasjon (68) som er indikativ for om meldingen har blitt mottatt.

6.

25 Fremgangsmåte ifølge krav 1, k a r a k t e r i s e r t v e d at lagringen av data innbefatter tilveiebringelse av dataord (45) i datameldingen (39), tilveiebringelse av kommandoord (61, 64, 65, 67) i inngangsstyreblokken (56),
30 lagring (DMA styrer 34-3, fig. 5) av første dataord (45) i datameldingen (39) i lagersteder (62-63) angitt sammenhengende (62) til det første dataordet (45) inntil meldingsarealet (57) er fullt eller inntil siste dataord (45) til datameldingen (39) er lagret i lageret (31-N, fig. 2).

35 7.

Fremgangsmåte ifølge krav 2, ved hvilken oppdateringen av vektortabellen er k a r a k t e r i s e r t v e d:

lagring (DMA styrer 34-3, fig. 5) av et kommandoord (67)
inneholdt i inngangsstyreblokken (57) i vektortabellen (55)
for således å oppdatere vektortabellen (55) med en indikasjon
av stedet i lageret (31-N, fig. 2) for neste inngangsstyre-
5 blokk (56) som skal bli anvendt.

8.

Fremgangsmåte ifølge krav 7, k a r a k t e r i s e r t
v e d lagring (DMA styrer 34-3, fig. 5) av lengden på
10 meldingsindikasjonen (68 ved 58).

15

20

25

30

35

Fig. 1.

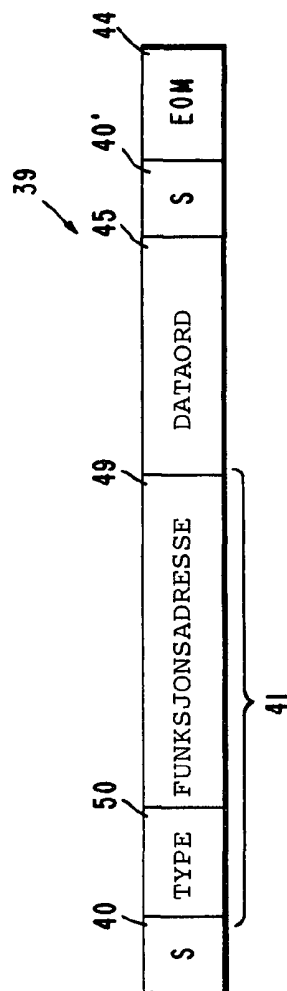
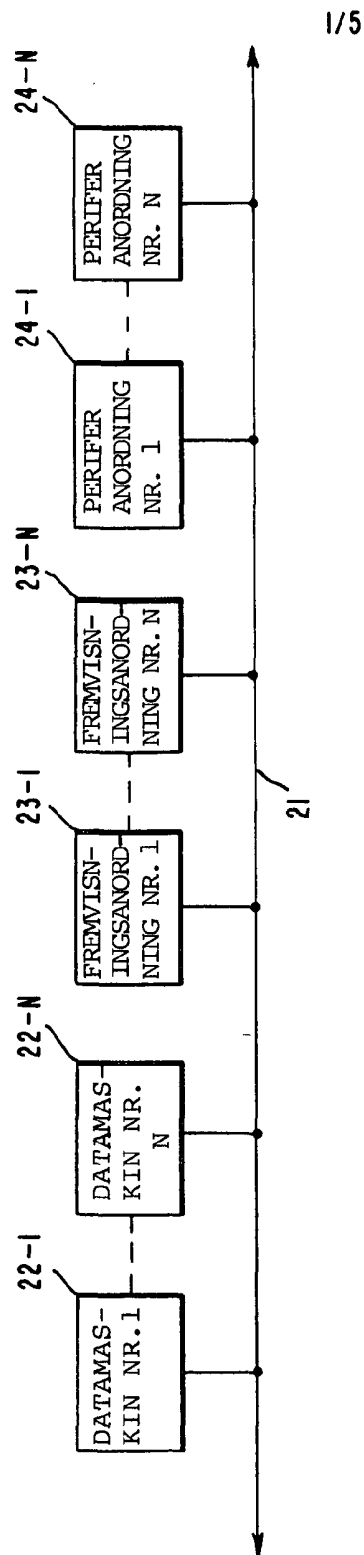


Fig. 3.

167946

Fig. 2.

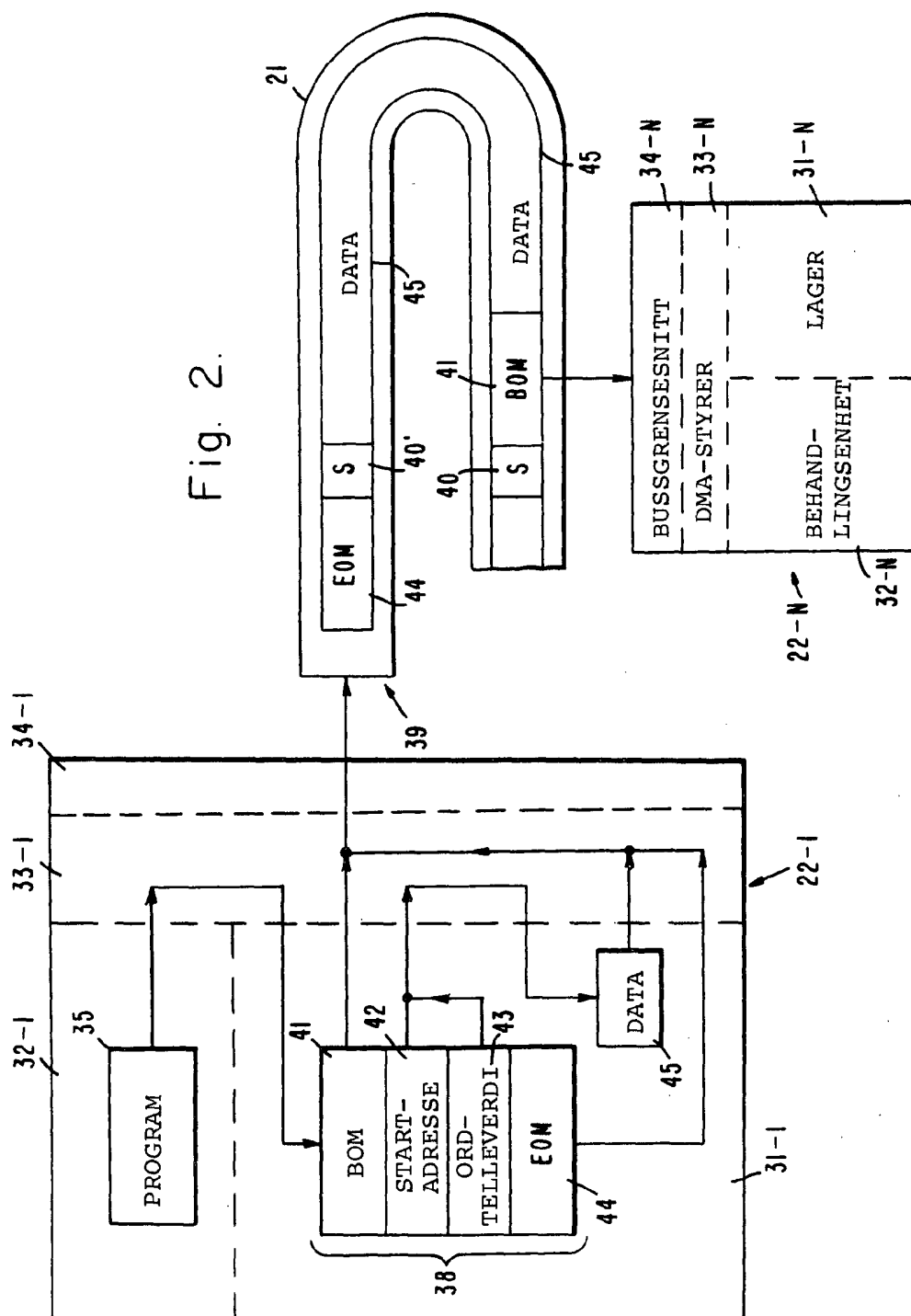


Fig. 4

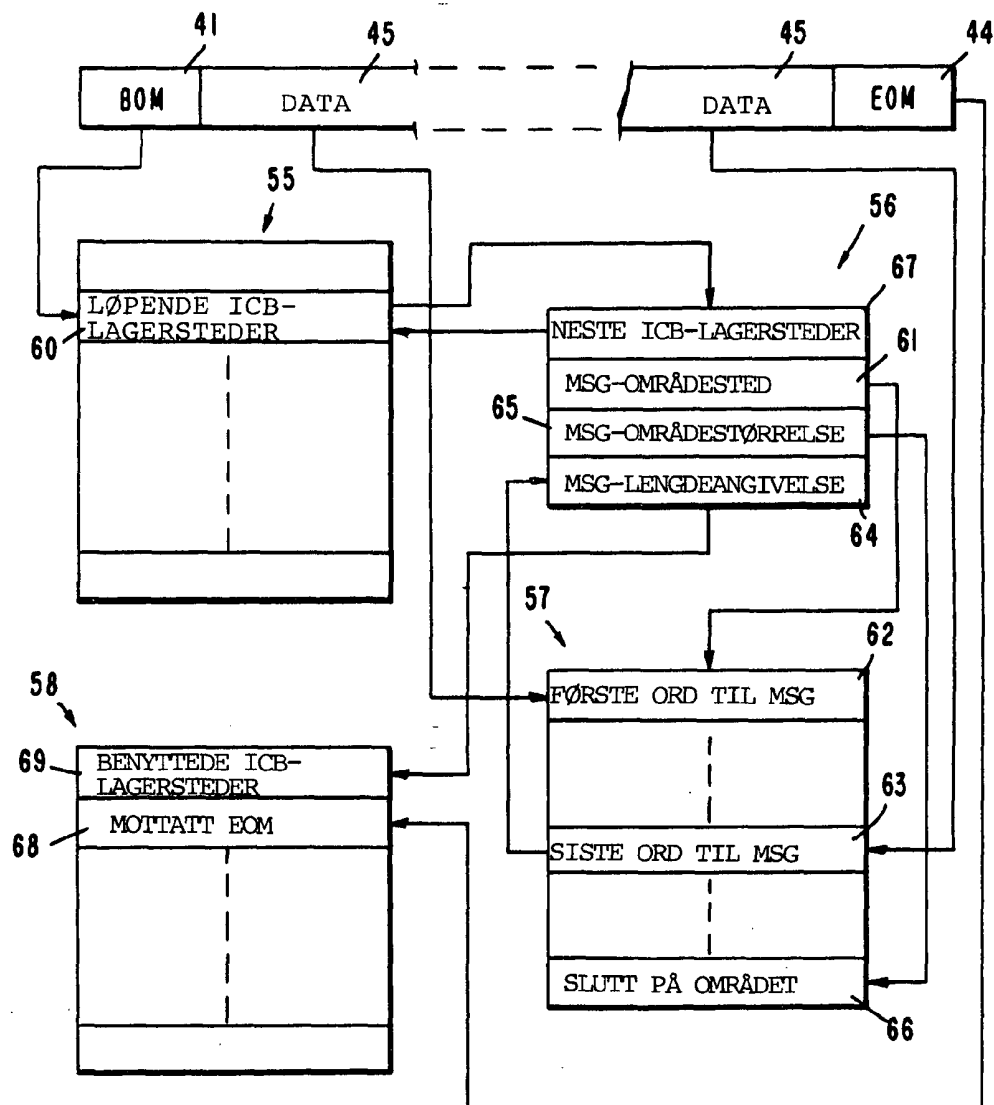


Fig. 5.

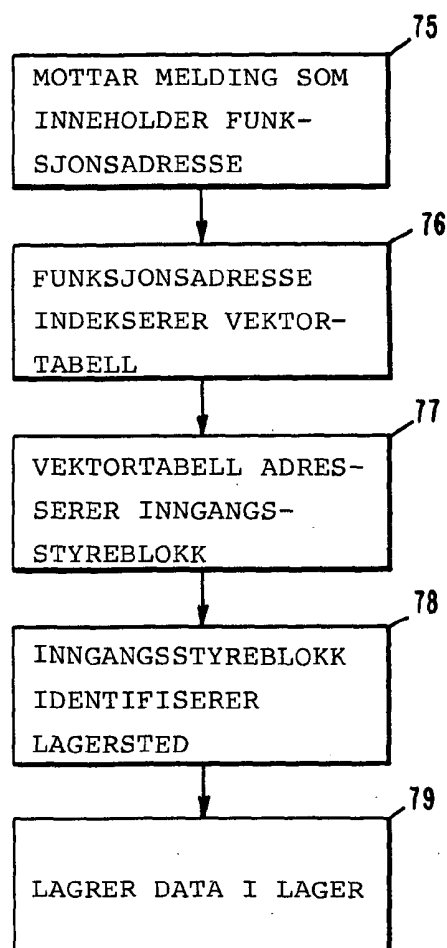


Fig. 6.

