



NORGE

(12) **PATENT**

(19) NO

(51) Int Cl⁷

(11) **320115**

H 04 N 7/15

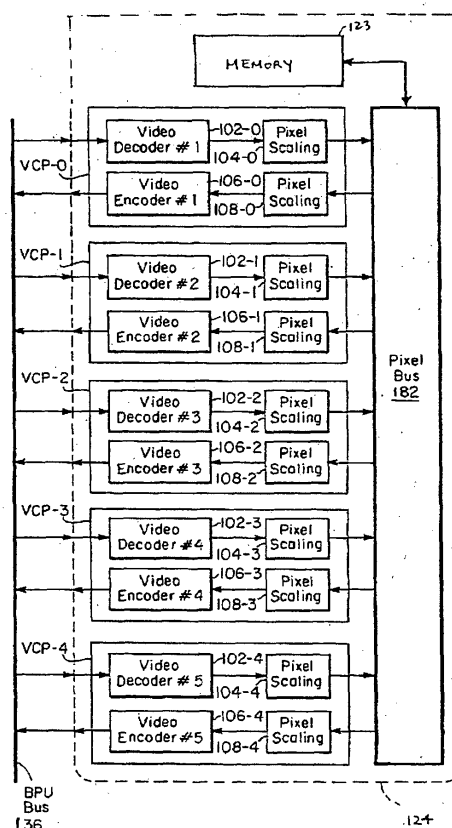
(13) **B1**

Patentstyret

(21)	Søknadsnr	20040661	(86)	Int.inng.dag og søknadsnr	
(22)	Inng.dag	2004.02.13	(85)	Videreføringsdag	
(24)	Løpedag	2004.02.13	(30)	Prioritet	Ingen
(41)	Alm.tilgj	2005.08.15			
(45)	Meddelt	2005.10.24			
(73)	Innehaver	Tandberg Telecom AS , Postboks 92, 1325 Lysaker, NO			
(72)	Oppfinner	Tom Erik Lia, Frosterudveien 39, 1832 ASKIM, NO			
		Tom-Ivar Johansen, Eckersbergs gate 35, 0266 OSLO, NO			
(74)	Fullmektig	Onsagers AS , Postboks 6963 St Olavs Plass, 0130 OSLO, NO			

(54)	Benevnelse	Anordning og fremgangsmåte for å generere CP-bilder.
(56)	Anførte publikasjoner	US 6288740 B1 US 6606112 B1

- (57) Sammendrag
- Den foreliggende oppfinnelsen vedrører generering av kontinuerlig-tilstedeværelsesbilder (CP) i en multipunktskontrollenhet (MCU) som styrer en multistedsvideokonferanse. Oppfinnelsen benytter bitstrukturen for ITU H.26*-standardene for å redusere prosesseringstiden og kravene i en MCU for å generere CP-betraktninger uten egenbetraktning. Ved å rearrangere makroblokker fra kodede CP-bilder, kan regioner i et CP-bilde fjernes for derved å tillate transittering av CP-bilder uten egenbetraktning til de deltagende steder. Rearrangeringsprosedyren reduserer signifikant behovet for koderessurser i en MCU.



Område for oppfinnelsen

Den foreliggende oppfinnelsen vedrører gjennomføring av videokonferanser, og spesielt generering av kontinuerlig-tilstedeværelsesbilder (eng: Continuous Presence (CP) images) i en multipunktskontrollenhet (eng: Multipoint Control Unit (MCU)).

Bakgrunn for oppfinnelsen

Transmisjon av bevegelige bilder i sanntid anvendes i mange applikasjoner slik som f.eks. gjennomføring av videokonferanser, nettmøter, TV-kringkasting og videotelefoni.

- 10 Det å representere bevegelige bilder krever imidlertid store mengder informasjon, idet digital video typisk beskrives ved å representere hver piksel i et bilde med 8 bits (1 byte). Slike ukomprimerte videodata fører til store bitvolumer, og kan ikke overføres over konvensjonelle kommunikasjonsnettverk og transmisjonslinjer i sanntid på grunn av begrenset båndbredde.
- 15 For å muliggjøre sanntids videotransmisjon kreves derfor høy grad av datakomprimering. Datakomprimering kan imidlertid gå på bekostning av bildekvalitet. Derfor har det blitt lagt stor innsats i å utvikle komprimeringsteknikker som tillater sanntids transmisjon av høykvalitetsvideo over båndbreddebegrensede dataforbindelser.
- 20 I videokomprimeringssystemer er hovedmålet å representere videoinformasjon med så liten kapasitet som mulig. Kapasitet defineres med bits, enten som en konstant verdi eller som bits/tidsenhet. I begge tilfeller er hovedmålet å redusere antallet bits. Den mest alminnelige videokodingsmetoden er beskrevet i standardene MPEG* og H.26*. Videodataene gjennomgår fire hovedprosesser før transmisjon, nemlig
- 25 prediksjon, transformasjon, kvantisering og entropikoding. Prediksjonsprosessen reduserer betraktelig mengden av bits som er nødvendig for at hvert bilde i en videosekvens skal overføres. Den drar fordel av at deler av sekvensen har likhet med andre deler av sekvensen. Siden prediktordelen er kjent både for koderen og dekodeeren, behøver bare forskjellen å overføres. Denne
- 30 forskjellen krever typisk mye mindre kapasitet for sin representasjon. Prediksjonen er hovedsakelig basert på bildeinnhold fra tidligere rekonstruerte bilder der beliggenheten av innholdet er definert ved bevegelsesvektorer. Prediksjonsprosessen utføres typisk på kvadratiske blokkstørrelser (f.eks. 16x16 piksler).
- 35 Videokonferansesystemer tillater også samtidig utveksling av audio-, video- og datainformasjon blant flere konferansesteder (eng.: conferencing sites). Systemer kjent som multipunktskontrollenheter (eng: multipoint control units (MCUs)) utfører svitsjefunksjoner for å tillate flere steder (eng.: sites) å kommunisere sammen i en konferanse. MCU'en linker stedene sammen ved mottak av rammer

med konferansesignaler fra stedene, prosesserer de mottatte signalene, og retransmitterer de prosesserte signalene til passende steder. Konferansesignalene innbefatter audio-, video-, data- og kontrollinformasjon. I en svitsjet konferanse blir videosignalet fra ett av konferansestedene, typisk det som har den mest høylydte

5 taler, kringkastet til hver av deltakerne. I en kontinuerlig-tilstedeværelseskonferanse blir videosignaler fra to eller flere steder romlig mikset for å danne et sammensatt videosignal for betraktning av konferansedeltakere. Kontinuerlig-deltakelsesbildet eller det sammensatte bildet er kombinert bilde som kan innbefatte levende videostrømmer, som stillbilder, menyer eller andre visuelle bilder fra deltakere på

10 konferansen.

I en typisk kontinuerlig-tilstedeværelseskonferanse blir videodisplayet inndelt i en sammensatt layout som har arealer eller områder (f.eks. kvadranter). Steder velges ved konferanseoppsettet fra stedene som er forbundet i konferansen for fremvisning i regionene. Felles sammensatte layouts innbefatter fire, ni eller seksten regioner.

15 Layouten velges og fastsettes deretter for konferansens varighet.

Noen konferansearrangementer tilveiebringer ulike sammensatte signaler eller en videomiks, slik at hvert sted kan betrakte en ulik blanding av steder. Andre arrangementer benytter stemmeaktivert kvadrantseleksjon for å assosiere steder med bestemte kvadranter. Et slikt arrangement muliggjør at konferansedeltakere kan

20 betrakte ikke bare faste videomikssteder, men også et sted valgt på basis av stemmeaktivitet. Imidlertid er layouten hva angår antall regioner eller kvadranter fastsatt for konferansen.

Med henvisning til fig. 1 er det vist en utførelsesform av en MCU 10 av den typen som er beskrevet i US patent 5 600 646. MCU 10 innbefatter også H-323-

25 funksjonalitet slik det er beskrevet i US patent 6 404 745. I tillegg har videoprosesseringen i MCU'en blitt forbedret, slik det vil beskrives videre her. Trekkene som er beskrevet her for MCU 10 kan utføres i en Tandberg MCU.

MCU-en 10 innbefatter i det minste én nettverksgrensesnittenhet (eng.: Network Interface Unit (NIU)) 120, minst én broprosesseringsenhet (eng.: Bridge Processing Unit (BPU)) 122, en videoprosesseringsenhet (eng.: Video Processing Unit (VPU)) 124, en dataprosesseringsenhet (eng.: Data Processing Unit) (DPU) 126, og en vertsprosesseringsenhet (eng.: Host Processing Unit (HPU)) 130. I tillegg til en industristandardarkitektur (ISA) vertskontrollbuss (eng.: a host Industry Standard Architecture control bus) 132, omfatter MCU-en 10 en nettverksbuss 134, en BPU-buss 136 og en X-buss 138. Nettverksbussen 134 overholder Multi-Vendor Integration Protocol (MVIP), mens BPU-bussen 136 og X-bussen er avledninger av MVIP-spesifikasjonen. HPU 130 tilveiebringer et styringsgrensesnitt (eng.: a management interface) for MCU-operasjoner. Hvert av de ovenstående MCU-elementer er videre beskrevet i de ovenfor angitte US patentene 5 600 646 og

30

35

40

6 404 745.

H.323-funksjonaliteten er tilveiebrakt ved tilføyelse av en gateway-prosesseringsenhet (eng.: Gateway Processing Unit (GPU)) 128 og en modifisert BPU omtalt som BPU-G 122A. GPU-en 128 kjører H.323-protokoller for oppkallssignaler og dannelsen og kontroll av audio-, video- og datastrømmer gjennom et Ethernet- eller annet LAN-grensesnitt 140 til endepunktterminaler. BPU-G 122A er en BPU 122 som er programmert til å prosessere audio-, video- og datapakker mottatt fra GPU 128.

MCU-operasjonen beskrives nå på høyt nivå, initielt for linjesvitsjet konferansegjennomføring og deretter for pakkesvitsjet H.323 konferansegjennomføring. I linjesvitsjet konferansegjennomføring gjøres digitale datarammer fra H.320-linjesvitsjede endepunktterminaler tilgjengelig på nettverksbussen 134 gjennom et nettverksgrensesnitt 142 til en NIU 120. BPU'ene 122 prosesserer datarammene fra nettverksbussen 134 for å produsere datarammer som gjøres tilgjengelig til andre BPU'er 122 på BPU-bussen 136. BPU'ene 122 ekstraherer også audioinformasjon fra datarammene.

BPU'ene 122 kombinerer komprimert videoinformasjon og mikset kodet audioinformasjon til rammer som plasseres på nettverksbussen 134 for transmisjon til respektive H.320-terminaler.

I tilfeller der de audiovisuelle terminaler opererer ved ulike transmisjonsrater, eller med ulike kompresjonsalgoritmer, eller skal mikses til et sammensatt bilde, sendes multiple videoinnganger til BPU 124 der videoinngangene dekomprimeres, mikses og rekomprimeres til en enkelt videostream. Denne enkeltvideostreamen sendes deretter tilbake gjennom BPU 122 som svitsjer videostreamen til de passende endepunktterminaler.

For pakkebasert H.323-konferansegjennomføring gjør GPU 128 audio-, video- og datapakker tilgjengelige på nettverksbussen 134. Datapakkene prosesseres gjennom DPU 126.

BPU-G 122A prosesserer audio- og videopakker fra nettverksbussen 134 for å produsere audio- og videokringkastingsmikser som plasseres på nettverksbussen 134 for transmisjon til respektive endepunktterminaler gjennom GPU 128. I tillegg prosesserer GPU-G 122A audio- og videopakker for å produsere datarammer som gjøres tilgjengelig for DPU'ene 122 på BPU-bussen 136. På denne måten tjener MCU 14 en gateway-funksjon hvorved regulære BPU'er 122 og BPU-G 122A kan utveksle audio og video mellom H.320- og H.323-terminaler på transparent måte.

Etter denne beskrivelsen av komponentene for MCU 10 som muliggjør de grunnleggende konferansebrodannelsesfunksjonene (eng.: the basic conference bridging functions), beskrives nå en høynivåbeskrivelse av fleksibiliteten som tilveiebringes av BPU 124 med henvisning til det funksjonelle blokkdiagrammet i fig. 2. I MCU 10 blir komprimert videoinformasjon fra opp til fem

audiovisuelle terminaler som er i samme konferanse, rutet til en bestemt VPU 124 over BPU-bussen 136. VPU-en 124 omfatter fem videokomprimeringsprosessorer (VCP0-VCP4), som hver har et par av videodekoder/-koder 102-i, 106-i, og pikskaleringsblokker 104-i, 108-i.

- 5 Et par av videodekoder/-koder 102-i, 106-i er tilordnet den komprimerte videoinformasjonsstrømmen som er assosiert med hvert bestemt sted (eng.: site) i konferansen. Hver videodekoder 102-i dekode den komprimerte videoinformasjonen ved bruk av algoritmen som passer kodingsalgoritmen for sitt assosierte sted. Innbefattet som del av videodekoderen 102-i kan være prosessering
- 10 for å bestemme rammingen, pakkene og sjekksummene som kan være del av transmisjonsprotokollen. Det skal bemerkes at en prosessorkodet videostream kan være tilordnet flere steder (f.eks. en kontinuerlig tilstedeværelsesapplikasjon som har mer enn fem steder i konferansen). I tillegg kan et dekode-/koderpar 102-i, 106-i svitsje mellom stedene innenfor en konferanse.
- 15 Den dekodete videoinformasjonen (f.eks. pikslene) skaleres opp eller ned, om nødvendig, ved en pikskaleringsblokk 104-i for å passe pikseloppløsningskravene for andre steder i konferansen som vil kode de skalerte pikslene. F.eks. kan et arbeidsbordsystem (eng.: a desktop system) kode ved en oppløsning på 256x240 piksler, mens en H-320-terminal kan kreve en pikseloppløsning på 358x288 piksler
- 20 for et Common Intermediate Format (CIF)-bilde. Andre alminnelige formater innbefatter Quarter Common Intermediate Format (QCIF) (176x144 piksler), 4CIF (704x576), SIF (352x240), 4SIF (704x480), VGA (640x480), SVGA (800x600) og XGA (1024x768).
- 25 VPU-en 124 innbefatter en pikselbuss 182 og minne 123. Systemet beskrevet i US patent 5 600 646 benytter en tidsdelt multipleksbuss. Spesielt leverer hver dekode 102-j piksler til pikselbuss 182 til minnet 123. Hver koder 106-j kan gjenvinne et hvilket som helst av bildene fra minnet 123 på pikselbussen for gjenkoding og/eller romlig miksing eller sammensetning. En annen pikskaleringsblokk 108-j er
- 30 forbundet mellom pikselbussen 182 og koderen 106-j for innstilling av pikseloppløsning for det samlede bildet etter behov.
- En kontinuerlig tilstedeværelsesapplikasjon blir nå beskrevet med henvisning til fig. 3 og 4. For enkelhets skyld er endepunktsterminalene vist som H-320-terminaler. I fig. 3 ankommer data fra steder 38 over et kommunikasjonsnettverk til respektive NIU'er 120. Fem steder 38 (A, B, C, D, E) er forbundet i konferansen. Stedene A og
- 35 B er vist forbundet til en bestemt NIU 120 som støtter multiple kodek-forbindelser (f.eks. et T1-grensesnitt). De andre stedene C, D og E er forbundet til NIU'er 120 som støtter bare én enkelt kodek-forbindelse (f.eks. et ISDN-grensesnitt). Hvert sted 38 plasserer én eller flere oktetter med digital data fra nettverksbussen 134 som usynkronisert H.221-rammede data. BPU'ene 122 bestemmer så H.221-rammingen
- 40 og oktettinnrettingen (eng.: octet alignment). Disse innrettede (eng.: aligned) data

gjøres tilgjengelige for alle andre enheter på BPU-bussen 136. DPU'ene 122 ekstraherer også audioinformasjon fra H.221-rammer og dekode audioen til 16 bits PCM-data. De dekodete audiodata gjøres tilgjengelig på BPU-bussen 136 for miksing med audiodata fra andre konferansesteder.

- 5 Innrettede H.221-rammer mottas av VPU 124 for prosessering av koder-/dekoderelementer betegnet videokomprimeringsprosessorer (VCP'er). VPU 124 har fem VCP'er (fig. 2) som i dette eksemplet er respektivt tilordnet steder A, B, C, D, E. En VCP på VPU 124 som er tilordnet sted E er funksjonelt illustrert i fig. 4. Komprimert videoinformasjon (H.261) ekstraheres fra H.221-rammene og
10 dekodes av VCP som bilde X. Dekodervideobildet X plasseres på pikselbussen 182 gjennom en skaleringsblokk. Fig. 4 viser pikselbussen 182 med dekodete videorammer fra hvert sted A, B, C, D, E suksessivt gjenvunnet fra minnet 123 identifisert ved deres respektive RAM-adresser. VCP'en anordnet sted E mottar de dekodete videorammer fra stedene A, B, C og D som deretter blir anbrakt i
15 rutemønster (eng: tiled) (romlig mikset) til et enkelt sammensatt bilde I. Det rutemønsteranbrakte (eng.: the tiled) bildet I blir deretter kodet som H.261-video innen H.221-ramming og plassert på BPU-bussen 136 (fig. 3) for BPU-prosessering som beskrevet ovenfor.

- Slik det kan ses fra beskrivelsen ovenfor, krever transkoding betraktelige
20 prosesseringsressurser, idet råpikseldata å mikses og kodes for å danne en mikset betraktning eller en kontinuerlig tilstedeværelsesbetraktning. For å unngå egenbetraktning, dvs. å unngå at CP-betraktningene inneholder et bilde av de respektive deltakere som de transmitteres til, må MCU'en innbefatte minst en koder for hvert bilde i en CP-betraktning. For å muliggjøre CP 16 må MCU-en da
25 innbefatte minst 16 kodere.

Sammenfatning av oppfinnelsen

Det er en hensikt ved den foreliggende oppfinnelsen å tilveiebringe en fremgangsmåte og en anordning for å unngå egenbetraktning, som reduserer det nødvendige antall kodere og prosesseringstiden.

- 30 I samsvar med et første aspekt av den foreliggende oppfinnelsen, oppnås den ovenstående hensikt og andre fordeler ved en fremgangsmåte for å danne et kodet mål-kontinuerlig tilstedeværelsesbilde (CP), som fremsatt i det vedføyde selvstendige krav 1.

- I samsvar med et andre aspekt ved den foreliggende oppfinnelsen, oppnås den
35 ovenstående hensikt og andre fordeler ved en anordning i en Multipoint Control Unit (MCU) for å danne et kodet mål-CP-bilde, som fremsatt i det vedføyde, selvstendige krav 8.

Ytterligere hensikter og fordeler oppnås ved de trekk som er fremsatt i de uselvstendige krav.

Kort beskrivelse av tegningene

For å gjøre oppfinnelsen enklere å forstå, vil den følgende drøfting vise til de vedføyde tegninger, der:

Fig. 1 er et blokkdiagram av en MCU-konfigurasjon,

5 Fig. 2 er et skjematisk blokkdiagram av en utførelsesform av en VPU,

Fig. 3 er et blokkdiagram av en MCU-konfigurasjon som illustrerer dataflyt for kontinuerlig tilstedeværelseskonferansegjennomføring,

Fig. 4 er et blokkdiagram som illustrerer anbringelse i rutemønster i en kontinuerlig tilstedeværelseskonferanse,

10 Fig. 5 er et blokkdiagram for anordningene av blokkgruppe i et CIF-bilde,

Fig. 6 illustrerer blokkgruppelaget i samsvar med H-263,

Fig. 7 illustrerer makroblokklaget i samsvar med H.263,

Fig. 8 er blokkdiagrammet som illustrerer tre ulike kontinuerlige tilstedeværelsesbilder som benyttes i én utførelsesform av den foreliggende oppfinnelsen,

15

Fig. 9 er et skjematisk blokkdiagram for en utførelsesform av den foreliggende oppfinnelsen, og

Fig. 10 er et skjematisk flytskjema som illustrerer en utførelsesform av fremgangsmåten i samsvar med oppfinnelsen.

20 Beste utførelsesmåte for oppfinnelsen

Den foreliggende oppfinnelsen benytter bitstrukturen for ITU H.26*standarden for å redusere prosesseringstiden og behovene i en MCU for å generere CP-betraktninger uten egenbetraktninger. For å forstå bitstrukturkarakteristikkene som benyttes, blir bildeblokkstrukturen i samsvar med H.263 beskrevet i det følgende.

25 I samsvar med H.263 inndeles hvert bilde i blokker som representerer 8x8 piksler. Blokkene anordnes i makroblokker, noe som for luminansdelen av pikslene betyr 16 (8x8) blokker og for krominansdelen av pikslene betyr 4 (2x2) blokker. En gruppe av blokker (GOB) representerer normalt 22 makroblokker, og antallet GOB'er pr. bilde er 6 for sub-QCIF, 9 for QCIF, og 18 for CIF, 4CIF og 16CIF. GOB-

30 nummereringen gjøres ved bruk av vertikal skann av GOB'ene, ved å starte den øvre GOB (nr. 0) og å slutte med den nederste GOB. Et eksempel på arrangementet av GOB'er i et bilde er gitt for CIF-bildeformatet i fig. 5. Data for hver GOB består av en GOB-header fulgt av data for makroblokker. Data for GOB'er transmitteres pr. GOB i økende GOB-tall. Starten av en GOB identifiseres ved en Group of Block Start Code (GBSC). Strukturen av GOB-laget er vist i fig. 6.

35

Data for hver makroblokk består av en makroblokkheader fulgt av data for blokkene. Strukturen er vist i fig. 7. COD er bare til stede i bilder som ikke er av

type "INTRA" for hver makroblokk i disse bildene. En bit som når den settes til "0" signaliserer at makroblokken er kodet. Dersom den settes til "1", transmitteres ikke ytterligere informasjon for denne makroblokken, i dette tilfellet skal dekoderen behandle makroblokken som en inter-makroblokk med bevegelsesvektor for hele blokken lik null og uten koeffisientdata.

Dersom COD er satt til "0", innbefatter datadelen av makroblokken informasjon om de respektive blokker i makroblokken, og denne informasjonen er representert ved bevegelsesvektorer som angir posisjonen i tidligere bilder hvortil de innbefattede piksler er like.

10 Konvensjonelt krever unngåelse av egenbetraktning i et CP-bilde spesielt koding for hver deltakerne, som impliserer én koder for hver utgående datastrøm i MCU'en, som angitt i fig. 2. Den foreliggende oppfinnelsen benytter makroblokkstrukturen for allerede kodede videodata for å oppnå skreddersydd miks av et CP-bilde avhengig av mottakeren.

15 I den følgende eksempelutførelsesformen av den foreliggende oppfinnelsen, betrakt en konferanse med fem endepunktssteder som innhenter videobilder av CIP-format og koder bildene i samsvar med H.263-standarden. I MCU'en blir datastrømmene fra de respektive deltakerne dekodet ved dekodere som er assosiert med respektive MCU-innganger i samsvar med H.263. Etter dekoding vil råpikseldata fra de
20 respektive deltakerne være tilgjengelige på en intern bus i MCU'en, klar for miksing og transkoding. I tilfelle av fem deltakere ville det være nærliggende å velge et CP 4-format for det miksedde bildet for å overføres tilbake til de respektive steder. Det miksedde formatet kan velges av MCU i samsvar med prinsippene "Best Impression" beskrevet i US patentsøknad 10/601,095.

25 I samsvar med denne eksempelutførelsesformen av oppfinnelsen blir to ulike CP4-bilder, CP bilde 1 og CP bilde 2, kodet ved hver respektive koder som illustrert i fig. 8. CP bilde 1 innbefatter bildene mottatt fra stedene 1, 2, 3 og 4, mens CP bildene 2 innbefatter de mottatte bilder fra stedet 5 i én kvadrant, som etterlater de gjenværende kvadranter tomme. Ved koding av CP-bildene og anordning av de
30 kodede data i blokksystemet beskrevet ovenfor, sammenfaller kvadrantgrensene med makroblokkgrensene i GOB'ene. Slik som for CP bilde 1, innbefatter de første elleve makroblokker i den første GOB data fra bildet for sted 1, mens de elleve siste makroblokker i den første GOB innbefatter data fra bildet for stedet 2.

I samsvar med den foreliggende oppfinnelsen rearrangerer MCU'en makroblokkene
35 i hvert CP-bilde i samsvar med mottakeren. Som et eksempel, i CP-bildet transmittert til sted 4, erstattes de elleve siste makroblokker i hver av de ni siste GOB'er i CP-bildet 1 med de henholdsvis elleve første makroblokker i hver av de ni første GOB'er for CP-bilde 2. Dette resulterer i et nytt dekodet CP-bilde, som innbefatter bildet mottatt fra sted 5 i stedet for bildet mottatt fra sted 4. Dette CP-

bildet transmitteres tilbake til sted 4, og følgelig unngås egenbetragtning ved dette stedet.

Korresponderende erstatninger eller rearrangementer utføres for de fire andre CP-bildene respektivt assosiert med de andre steder.

- 5 Fig. 9 illustrerer et eksempel på den indre arkitektur for en MCU i samsvar med den foreliggende oppfinnelsen. Denne arkitekturen samsvarer med VPU for tidligere kjent teknikk som illustrert i fig. 2. Pikselbussen, minnet og
 10 pikselskaleringseenhetene er for enkelhets skyld erstattet med en mikse- og skaleringsenhet. Bemerk at bus 1 og bus 2 i fig. 9 alternativt kunne være sammensatt av en felles databus. Bemerk også at den virkelige implementasjonen kan være annerledes, og at bare enhetene som er relevant for den foreliggende oppfinnelsen er vist.

- 15 Inngangsdatastrømmene er dekodet med en separat dekode for hvert sted i samsvar med kodingsstandarden som benyttes, i dette tilfellet H.263. De dekodete data er i form av PCM-data, og er gjort tilgjengelige for en mikse- og skaleringsenhet (eng.: Mixing and Scaling Unit (MSU)) på databus 1.

- MSU mikser romlig PCM-dataene fra stedene 1, 2, 3 og 4, og danner CP-bilde 1. Et CP-bilde 2 blir også dannet ved å plassere PCM-dataene fra sted 5 i den første kvadrant, hvilket etterlater de gjenværende kvadranter tomme eller fylt med
 20 dummydata. PCM-dataene for de to romlig miksedte bildene gjøres deretter tilgjengelige for de to følgende koderne på databussen 2. Koderne henter PCM-data for CP-bildene generert av MSU'en fra databus 2 og koder hvert respektive bilde. Resultatet av kodingsprosessen er et antall makroblokker sammensatt i GOB'er som beskrevet ovenfor. I tilfelle av et CIF-format i samsvar med H.263, inneholder en
 25 GOB 22 makroblokker, og hvert bilde består av 18 GOB'er. Etter koding av bildene blir GOB'er påfølgende innsatt i en assosiert buffer. Størrelsen av buffere skal være tilstrekkelig stort til å romme GOB'ene for i det minste ett bilde.

- Imidlertid er antallet bits som representerer et kodet bilde ikke konstant, men det kan variere vesentlig i samsvar med variasjonen av bildeinnhold og bevegelser fra
 30 ett bilde til et annet. Antallet bits er også avhengig av hvorvidt bildet er intra-kodet eller inter-kodet, dvs. prediksjon fra nabomakroblokker i det samme bildet, eller prediksjon fra tidligere bilde(r).

- Når kodete data for komplett synkrone bilder innsettes i de respektive bufferne, er repakkerne klare til å rearrangere rekkefølgen for makroblokkene for å danne CP-
 35 bildene som er nødvendige for de assosierte utganger. Repakkerne vil være i stand til å identifisere og isolere makroblokkene ved hjelp av GOB'en og makroblokkheadere. Starten av hver GOB angis ved en unik startkode betegnet GBSC (Group of Block Start Code), fulgt av GN (Group Number) som angir GOB-tallet. I headerne for makroblokkene angir COD hvorvidt makroblokken er kodet

eller ikke. Dersom COD er "1", er ingen ytterligere informasjon tilstede for denne makroblokken, med bevegelsesvektor for hele blokken lik null og uten noen koeffisientdata. Dersom COD er "0", vil ytterligere data for makroblokken følge. Noen av de ytterligere data kan være av variabel lengde, men de ulike koder er

5 definert på en slik måte at lengden av hver kode er veldefinert.

Siden makroblokkene er identifiserbare og midlertidig lagret i en buffer, kan repakkerne lese ut makroblokkene i en hvilken som helst rekkefølge, og danne en hvilken som helst variant av et CP4-bilde fra bildet av de fem stedene. Som eksempel betrakt repakkeren som danner CP-bildet for sted 4. De første buffere

10 inneholder de kodede data for CP-bilde 1, mens den andre bufferen inneholder de kodede data for CP-bilde 2. Repakkeren henter GOB 1-9 for CP-bilde 1 i samme rekkefølge som det opptrer i den første bufferen for å danne det nye CP-bildet. Ved dannelse av GOB 10 vil repakkeren imidlertid identifisere og hente de elleve første makroblokker for GOB 10 i den første bufferen fulgt av de elleve første

15 makroblokker for GOB 1 i den andre bufferen. Videre dannes GOB 11 ved å hente de elleve første makroblokker i GOB 11 i den første bufferen fulgt av de elleve første makroblokker for GOB 2 i den andre buffer. De sju gjenværende GOB'er dannes tilsvarende, avsluttende med GOB 18, som dannes ved å hente de elleve første makroblokker for GOB 18 i den første bufferen fulgt av de elleve første

20 makroblokker for GOB 9 i den andre buffer.

Repakkerne kan forhåndsprogrammeres for å hente makroblokkene fra bufferne i en konstant rekkefølge, eller rekkefølgen av makroblokkene kan kontrolleres av en kontrollenhet (ikke vist) som tillater en repakke å danne ulike CP-bilder.

Fig. 10 er et skjematisk blokkdiagram som illustrerer en utførelsesform av

25 fremgangsmåten i samsvar med oppfinnelsen.

Fremgangsmåtettrinnene illustrert i fig. 10 er innbefattet i en fremgangsmåte for å danne et kodet mål-kontinuerlig tilstedeværelsesbilde (CP) i samsvar med en videokodingsstandard fra et antall kodede videosignaler som innbefatter definerte rekkefølger av makroblokker. Hver makroblokk omfatter kodede videosignaler som

30 samsvarer med et respektivt endepunktsvideobilde, mottatt fra endepunkter som deltar i en multipunkts videokonferanse.

Fremgangsmåten starter ved trinn 202.

Først utføres dekodingstrinnet 204, hvor videosignaler dekodes til de korresponderende endepunktsvideobilder.

35 Videre, i miksetrinnet 206, blir endepunktsvideobildene romlig mikset til et antall CP-bilder sammensatt av regioner respektivt assosiert med hvert av endepunktsvideobildene.

Videre, i kodingstrinnet 208, blir CP-bildene respektivt kodet til et antall kodede CP-bilder. Dette trinnet etablerer de definerte rekkefølger av makroblokker som

korresponderer med videokodingsstandarden og en sammensetning av regionsgrenser og makroblokkgrenser.

Etterfølgende de tre forberedende trinn 204, 206 og 208 ovenfor, utføres dannelsesstrinnet (eng.: the creating step) 210. I dannelsesstrinnet 210 dannes det
5 kodete mål-CP-bildet for å reorganisere nevnte rekkefølge av makroblokker på en forhåndsbestemt eller kontrollert måte.

Fordelaktig omfatter dannelsesstrinnet 210 å erstatte et første antall m makroblokker som representerer en regionsbredde i m antall påfølgende GOB'er som representerer en regionshøyde i en første av antallet av kodete CP-bilder med m antall
10 makroblokker i n antall påfølgende GOB'er fra en andre av antallet kodete CP-bilder.

Fremgangsmåten slutter ved trinn 212.

Som et alternativ kan de tre forberedende trinn for prosessen beskrevet med henvisning til fig. 10, dvs. dekodingsstrinnet 204, miksetrinnet 206 og kodingsstrinnet
15 210, erstattes med et enkelt forespørselstrinn (ikke illustrert). I dette forespørselstrinnet blir endepunktene forespurt om å kode de respektive endepunktsvideobildene i samsvar med videostandarden og en bestemt oppløsning, bitrate og skalering.

I sin mest grunnleggende form innbefatter fremgangsmåten i samsvar med
20 oppfinnelsen bare dannelsesstrinnet 210.

Utførelsesformene beskrevet så langt har blitt begrenset til dannelsen av CP4-bilder av CIF-format i samsvar med H.263-standarden. Imidlertid vil fagfolk innse at de grunnleggende prinsipper ved den foreliggende oppfinnelsen også er anvendelige på andre CP-bilder av andre formater. Som et eksempel kan den foreliggende
25 oppfinnelsen også benyttes ved dannelsen av CP-9-bilder. Da bestemmes bildegrensene innenfor hver GOB (i tilfelle av CIF, H.263) etter den sjuende og fjortende makroblokken (alternativt etter den åttende og femtende eller sjuende og femtende).

I ethvert tilfelle skal det være en repakker (eller i det minste en repakkerprosedyre) for hver MCU-utgang. Antallet kodere avhenger av forholdet mellom antall regioner i CP-bildene og antallet steder som skal fylle regionene for CP-bildene. Det må
30 være nok kodere til å generere én region for hvert sted. I tilfelle av CP 4 og åtte steder, vil to kodere være tilstrekkelig for å plassere én region for hvert sted i hver av de totalt åtte kvadrantene. Imidlertid vil økning av antallet steder til 9 kreve ytterligere én koder for å danne et tredje CP-4-bilde hvori den niende region kunne
35 ta opphold.

I en mer generell utførelsesform av oppfinnelsen utføres ingen dekoding og skalering/miksing i MCU'en. I stedet forespørres endepunktene om å transmittre et kodet bilde i samsvar med en viss standard, oppløsning, bitrate og skalering.

- Makroblokkene for de innkommende datastrømmer lastes da direkte inn i bufferne (fortrinnsvis én for hver datastrøm), og repakkerne rearrangerer makroblokkene i samsvar med forhåndsprogrammerte eller kontrollerte prosedyrer, hvilket danner CP-bilder uten egenbetragtning som kan transmitteres til de respektive
- 5 konferansesteder. Som et eksempel, betrakt fem steder som deltar i en konferanse. En CP-4-betraktning krever da at endepunktene plasserer deres respektive bilder i én av kvadrantene i et helt bilde før det kodes. Dette forespørres til endepunktene sammen med kodingsinformasjon slik som standard, oppløsning, bitrate og skalering. Repakkerne kan da enkelt rearrangere makroblokkene for de
- 10 innkommende datastrømmer når de er tilstede i de respektive bufferne slik det er beskrevet tidligere.

PATENTKRAV

1. Fremgangsmåte for å danne et andre antall kodet mål-kontinuerlig-tilstedeværelsesbilder (CP-bilder) i samsvar med en videokodingsstandard fra et antall kodede videosignaler innbefattende definerte rekkefølger av makroblokker,
 - 5 hver omfattende kodede videosignaler korresponderende med et respektivt endepunktsvideobilde, mottatt fra endepunkter som deltar i en multipunktsvideokonferanse,

karakterisert ved

å mikse råpikseldata dekodet fra videosignalene til et første antall ukodede CP-bilder,

10 å kode nevnte første antall ukodede CP-bilder til et første antall kodede CP-bilder i samsvar med de definerte rekkefølger av makroblokker,

å danne det andre antall kodet mål-CP-bilder ved å rearrangere nevnte rekkefølge av makroblokker på en forhåndsdefinert eller kontrollert måte.
 - 15 2. Fremgangsmåte i samsvar med krav 1,

karakterisert ved det følgende tilleggstrinn, som utføres før trinnet med å danne det kodede mål-CP-bildet:

å forespørre endepunktene om å kode de respektive endepunktsvideobildene i samsvar med videostandarden og en bestemt oppløsning, bitrate og skalering.
 - 20 3. Fremgangsmåte i samsvar med krav 1,

karakterisert ved de følgende tilleggstrinn, før trinnet med å danne det kodede mål-CP-bildet:

å dekode videosignalene til de korresponderende endepunktsvideobildene,

romlig å mikse endepunktsvideobildene til et antall CP-bilder sammensatt av regioner respektivt assosiert med hver av endepunktsvideobildene,

25 å kode nevnte antall CP-bilder til et antall kodede CP-bilder respektivt resulterende i de definerte rekkefølger av makroblokker korresponderende til videokodingsstandarden og en sammensetning av regionsgrenser og makroblokkgrenser.
 - 30 4. Fremgangsmåte i samsvar med krav 3,

karakterisert ved at de kodede CP-bilder og det kodede mål-CP-bildet hvert er av et CIF-format med 18 blokkgrupper (GOB'er), hver innbefattende 22 makroblokker, anordnet i en stackformasjon slik at de første ni GOB'er representerer øvre regioner og de siste ni GOB'er representerer nedre regioner.

5. Fremgangsmåte i samsvar med krav 4,
karakterisert ved at trinnet med å danne det kodede mål-CP-bildet
innbefatter:
 - 5 å erstatte m antall makroblokker som representerer en regionsbredde i n antall
påfølgende GOB'er som representerer en regionshøyde i et første av antallet kodede
CP-bilder med m antall makroblokker i n antall påfølgende GOB'er fra et andre av
antallet kodede CP-bilder.
 6. Fremgangsmåte i samsvar med krav 5,
karakterisert ved at $m=11$, $n=9$, og at nevnte regioner hver representerer
10 én kvadrant i et CP-bilde.
 7. Fremgangsmåte i samsvar med krav 5,
karakterisert ved at $m=7$ eller $m=8$, $n=6$ og at nevnte regioner hver
representerer en åttendedel av et CP-bilde.
 8. Anordning i en multipunktskontrollenhet (MCU) for å danne et kodet mål-
15 CP-bilde i samsvar med en videokodingsstandard fra et antall kodede
videoinngangssignaler, hvert korresponderende med et respektivt
endepunktsvideobilde, mottatt fra endepunkter som deltar i en
multipunktvideokonferanse,
karakterisert ved
 - 20 en dekode for hvert kodet videoinngangssignal, konfigurert til å dekode
videosignalene til de korresponderende endepunktsvideobilder,
en mikse- og skaleringsenhet, konfigurert til romlig å mikse endepunktsidebildene
til et antall CP-bilder sammensatt av regioner respektivt assosiert med hver av
endepunktsvideobildene,
 - 25 et antall kodere, konfigurert til å kode nevnte antall CP-bilder til et antall kodede
CP-bilder,
én databuffer for hver av nevnte kodere, som CP-bilder respektivt innsettes i, med
en makroblokk-rekkefølge korresponderende med videokodingsstandarden og en
sammensetning av regionsgrenser og makroblokkgrenser,
 - 30 én repakker for hver MCU-utgang konfigurert til å danne det kodede mål-CP-bildet
ved å hente nevnte makroblokker i en forhåndsbestemt eller kontrollert rekkefølge.
 9. Anordning i samsvar med krav 8,
karakterisert ved at de kodede CP-bilder og det kodede mål-CP-bildet
hvert er av et CIF-format med 18 blokkgrupper (GOB'er), hver innbefattende 22
35 makroblokker, anordnet i en stackformasjon slik at de første 9 GOB'er representerer
øvre regioner, og de siste 9 GOB'er representerer nedre regioner.
 10. Anordning i samsvar med krav 9,
karakterisert ved at repakkeren videre er konfigurert til å erstatte m

antall makroblokker som representerer en regionsbredde i n antall påfølgende GOB'er som representerer en regionshøyde i et første av antallet kodede CP-bilder med m antall makroblokker i n antall påfølgende GOB'er fra et andre av antallet kodede CP-bilder.

- 5 11. Anordning i samsvar med krav 10,
 k a r a k t e r i s e r t v e d at $m=11$, $n=9$, og nevnte regioner hver representerer én
 kvadrant av et CP-bilde.
12. Anordning i samsvar med krav 10,
 k a r a k t e r i s e r t v e d at $m=7$ eller $m=8$, $n=6$ og at nevnte regioner hver
- 10 representerer en åttendedel av et CP-bilde.

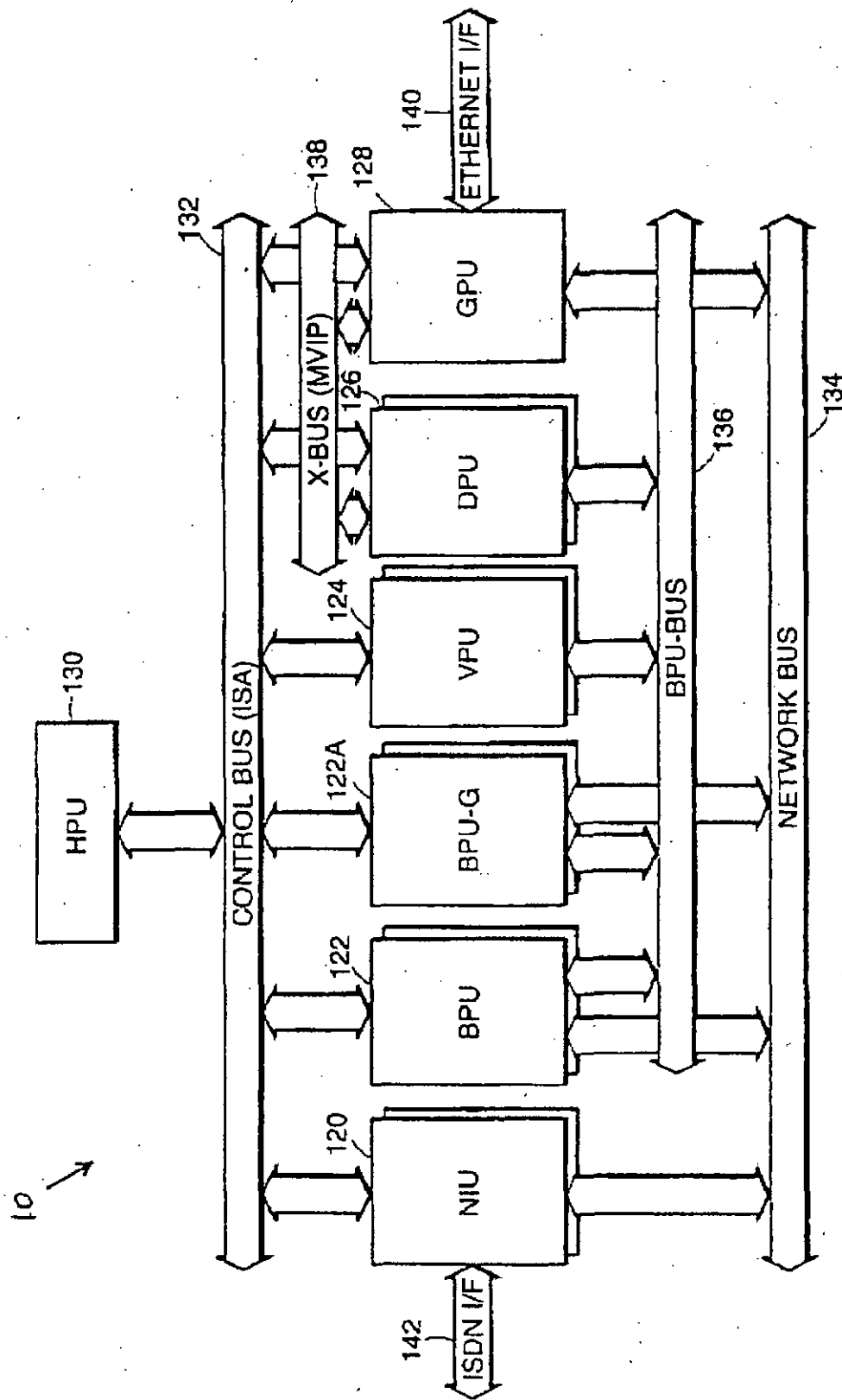


Figure 1

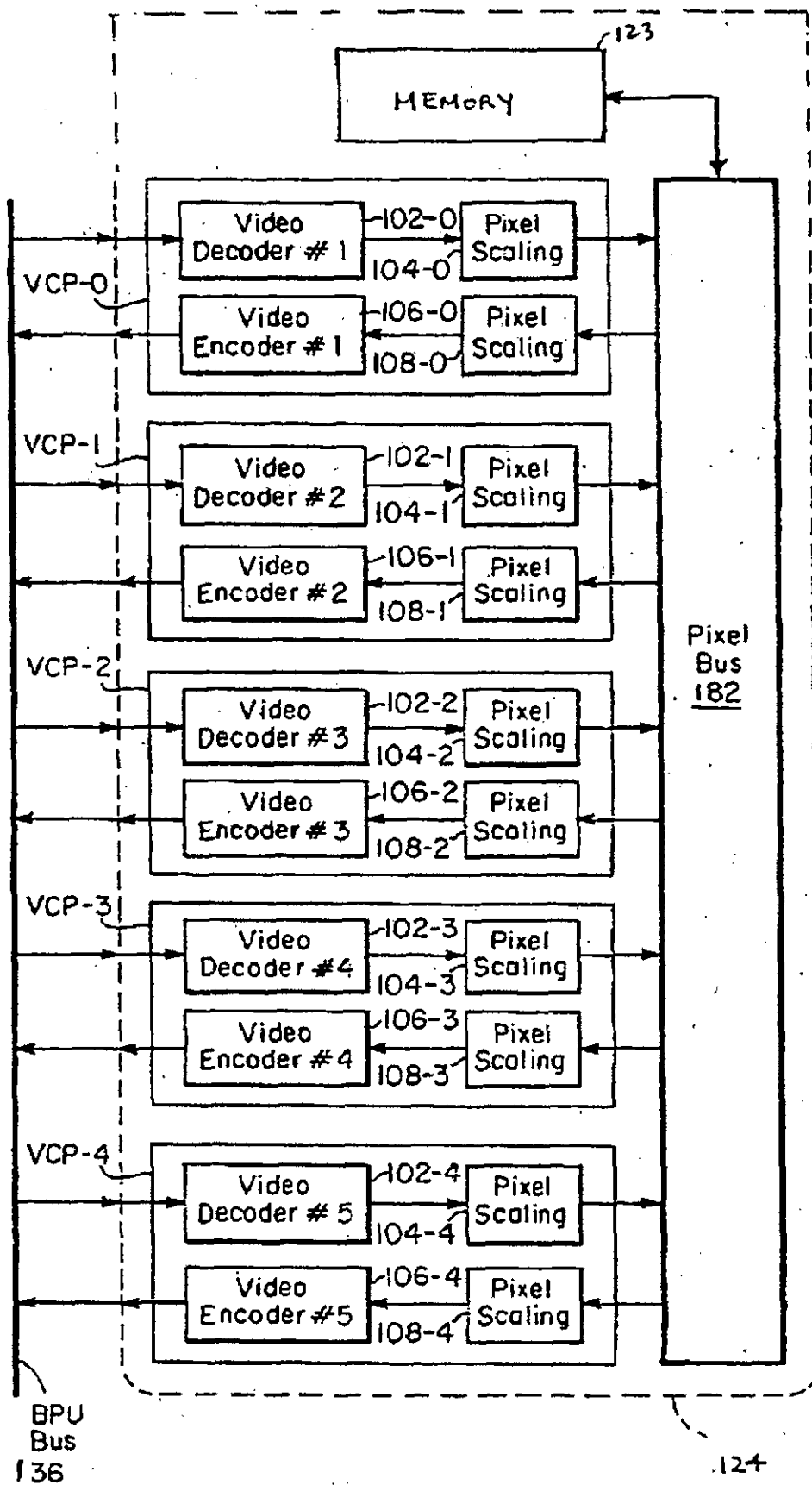


Figure 2

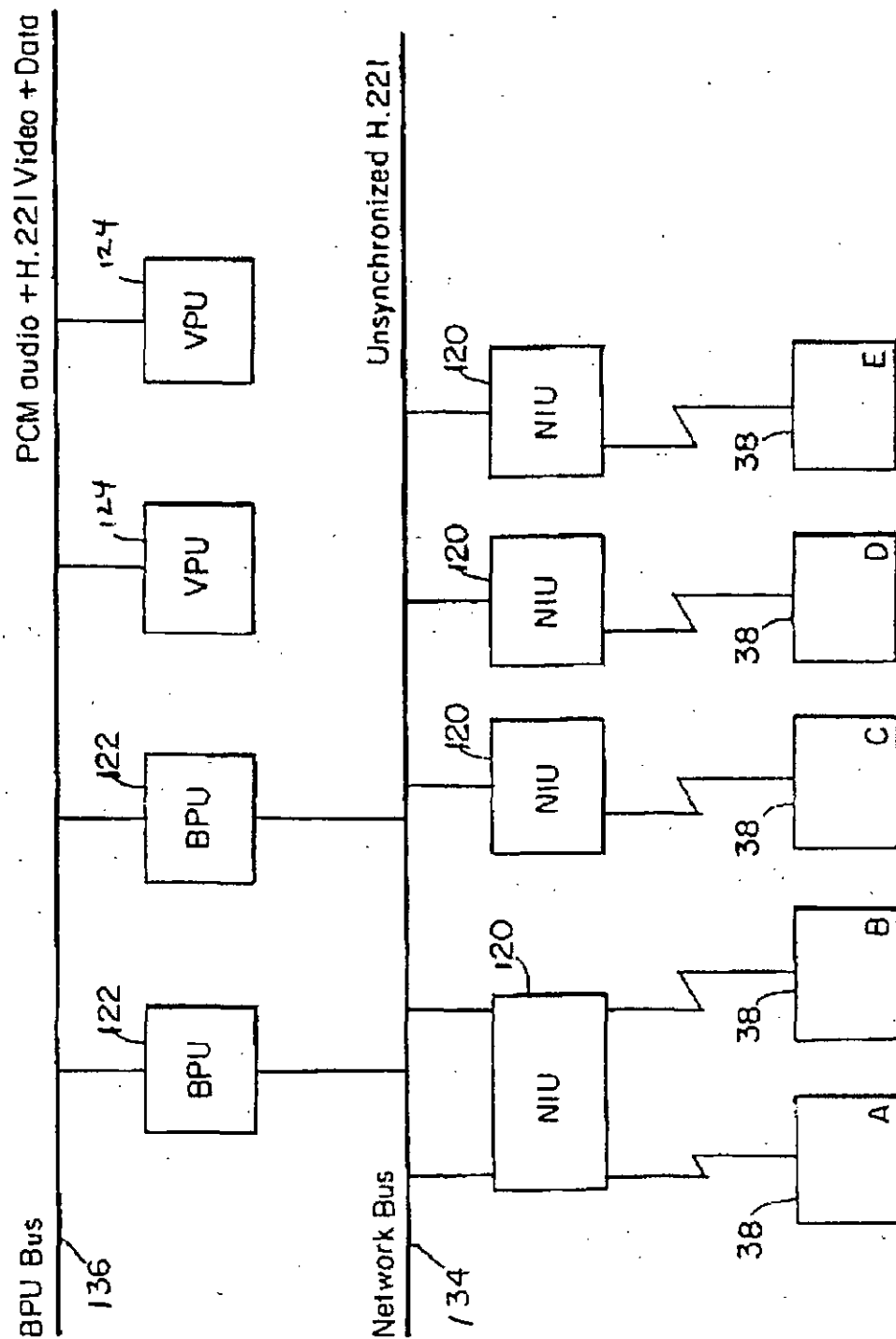


Figure 3

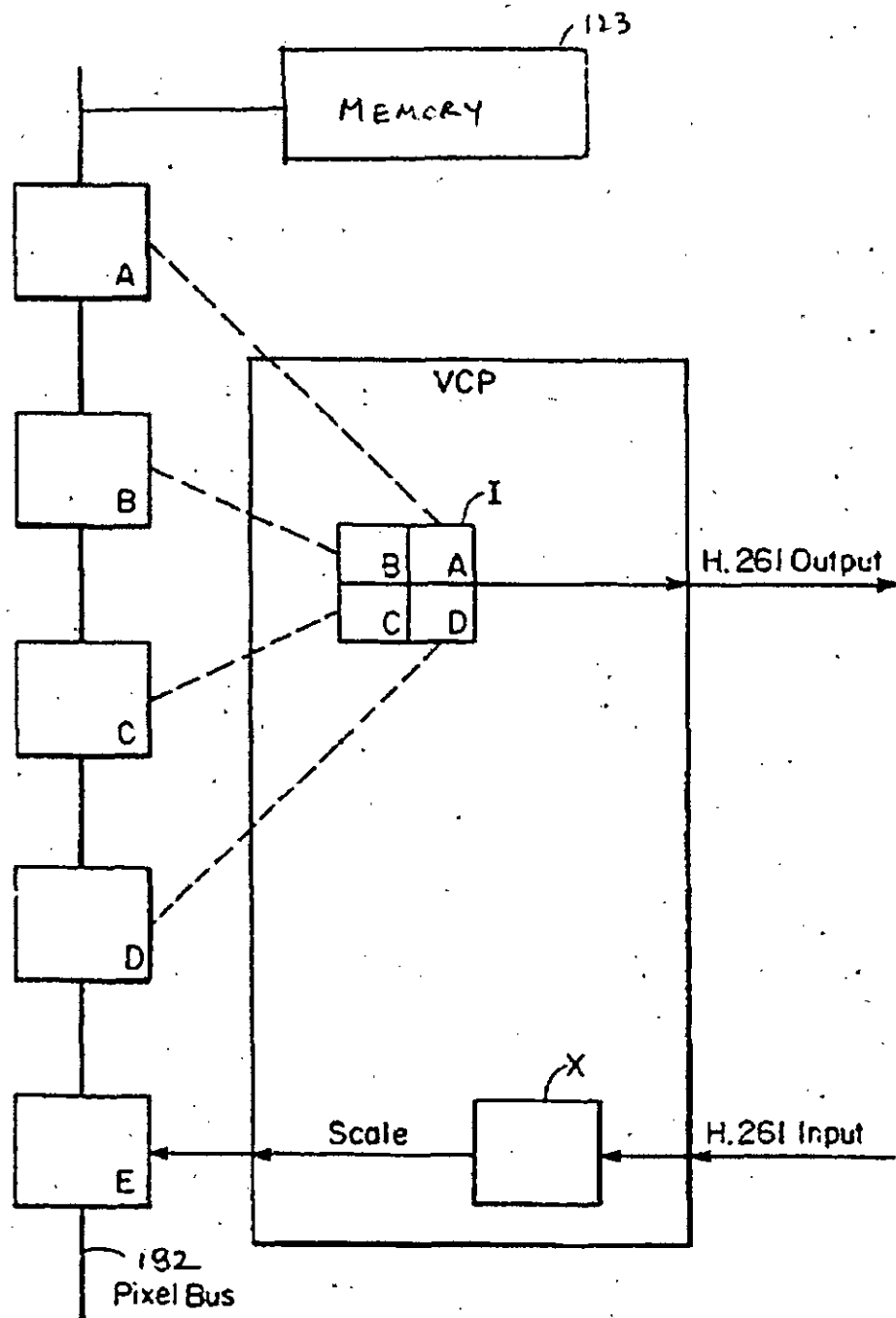


Figure 4

0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17

Figure 5: Arrangement of GOBs in a CIF picture



Figure 6: Structure of GOB layer

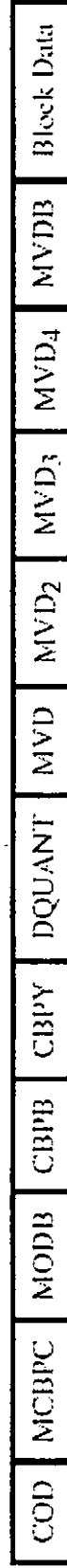


Figure 7: Structure of macroblock layer

Site 1	Site 2
Site 3	Site 4

CP image 1

Site 5	

CP image 2

Site 1	Site 2
Site 3	Site 5

Rearranged CP image transmitted to site 4

Figure 8

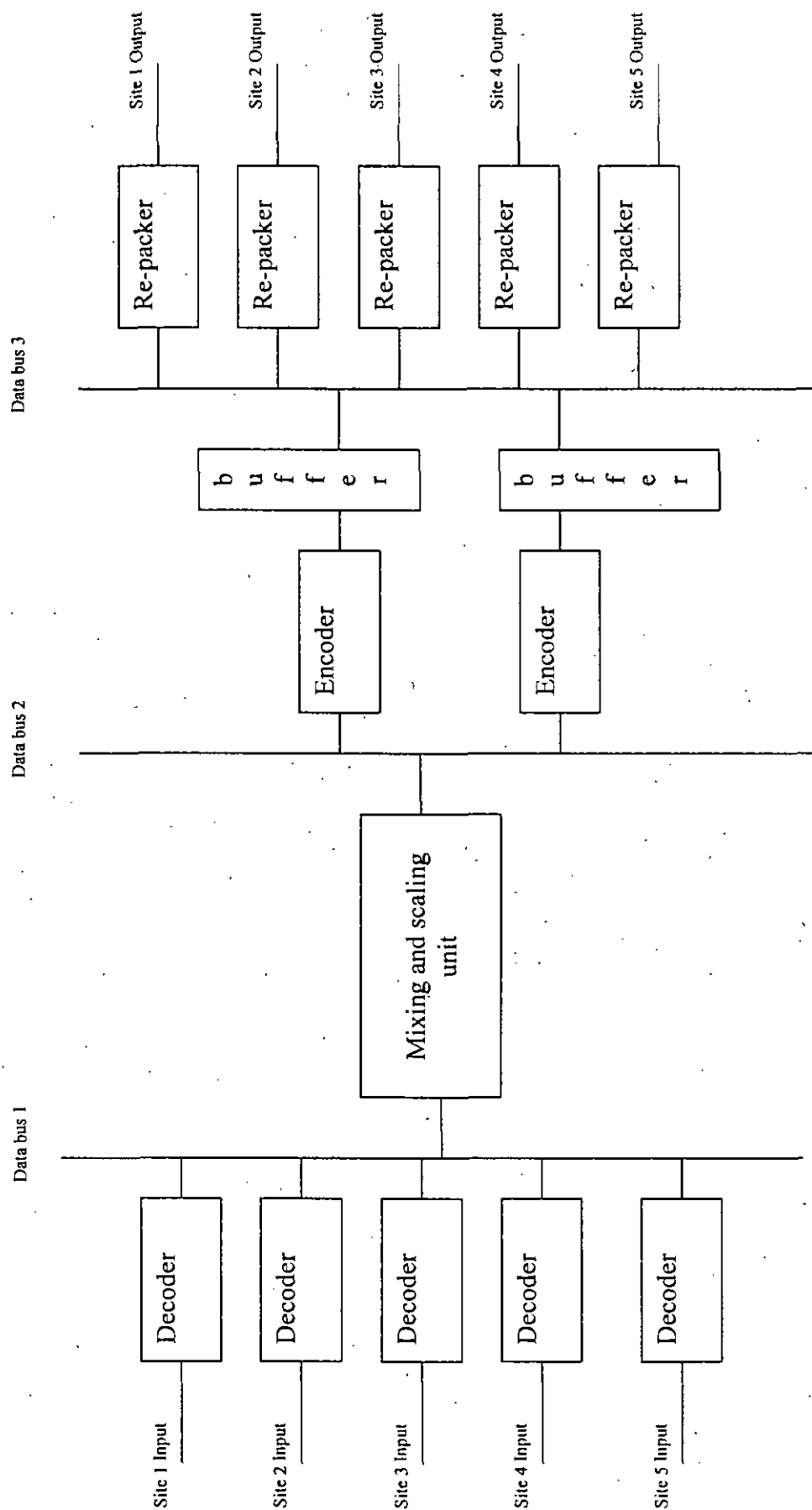


Figure 9

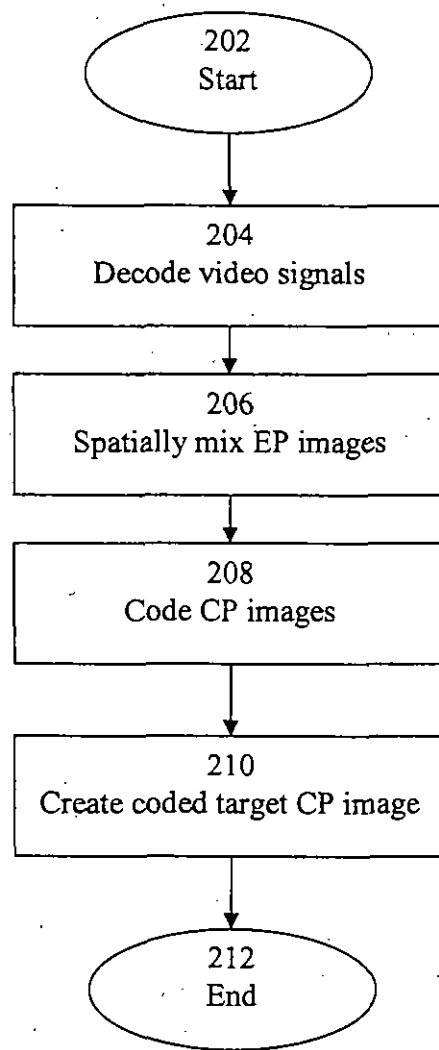


Figure 10