



(12) **PATENT**

(19) NO

(11) **336023**

(13) **B1**

NORGE

(51) Int Cl.

H04B 7/26 (2006.01)
H04B 1/7083 (2011.01)
H04W 56/00 (2009.01)
H04J 11/00 (2006.01)
H04J 3/06 (2006.01)

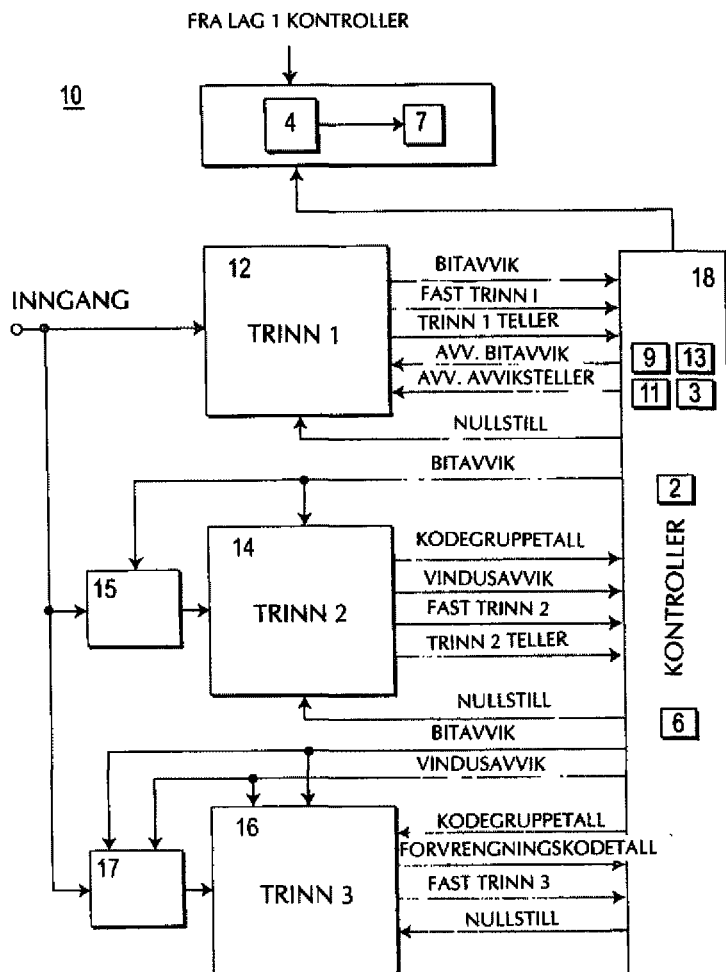
Patentstyret

(21)	Søknadsnr	20033770	(86)	Int.inng.dag og søknadsnr	2002.02.04 PCT/US2002/03217
(22)	Inng.dag	2003.08.25	(85)	Videreføringsdag	2003.08.25
(24)	Løpedag	2002.02.04	(30)	Prioritet	2001.02.27, US, 271642 2001.07.31, US, 918611
(41)	Alm.tilgj	2003.10.08			
(45)	Meddelt	2015.04.20			
(73)	Innehaver	Intel Corp, 2200 Mission College Boulevard, US-CA95054 SANTA CLARA, USA			
(72)	Oppfinner	Donald M Grieco, 18 Shore Road, US-NY11030 MANHASSETT, USA			
		Demir Alpasland, 1210 Towne House Village, US-NY11749 ISLANDIA, USA			
(74)	Fullmektig	Onsagers AS, Postboks 1813 Vika, 0123 OSLO, Norge			

(54)	Benevnelse	Innledende celledøkealgoritme
(56)	Anførte publikasjoner	WO 0044117 A2
(57)	Sammendrag	

Et system og fremgangsmåte for å etablere initiell synkronisering for linken mellom en UE og en basestasjon i et kommunikasjonsnettverk bruker vindusutelukkelseslogikk for å kunne unngå en fastlåsningsbetingelse ved en deteksjon av galt offentlig landmobilt nettverk (PLMN).

Kommunikasjonssignalet blir prosessert i en tretrinn beslutningsprosess. Det første beslutningstrinnet (12) bestemmer et bitavvik for den sterkeste veien som er detektert over en ramme med sampler. Som svar på den første beslutningen (12) vil det andre trinnet (14) generere et forvrengningskodegruppetaill og vindusavvik for å gjenvinne den andre synkroniseringskoden. Det tredje beslutningstrinnet (16) mottar en primær forvrengningskode som svar på kodegruppetaill for å synkronisere nevnte UE med basestasjonen.



Foreliggende oppfinnelse er generelt i området for brukerutstyr (UE) synkronisering med en basestasjon. Mer spesielt er den foreliggende oppfinnelsen i området for et cellesøkesystem som utnytter en forbedret initiell cellesøkealgoritme.

- 5 Initielle cellesøkealgoritmer blir brukt til å synkronisere UE med en basestasjon. UE oppnår denne prosedyren via en felles nedlinkskanal kalt den fysiske synkroniseringskanalen (PSCH). Med referanse til figur 2 har PSCH en struktur hvor den samme primære synkroniseringskoden (PSC) blir sendt ved begynnelsen av hvert vindu, mens en andre synkroniseringskode (SSC) blir sendt for hvert vindu, som
- 10 resulterer i femten (15) forskjellige SSC'er. Som en fagmann vet kan en ramme som har femten (15) vinduer sende femten (15) SSC'er.

- Sendingsrekkefølgen av SSC avhenger av det primære forvrengningskodegruppenummeret. Som et eksempel, i et femhundreogtolv (512)
- 15 cellesystem er det sekstifire (64) grupper. I hver gruppe vil mønstrene til SSC'ene og deres sykliske skift være forskjellige. Som et resultat er det femhundreogtolv (512) primære forvrengningskoder. Hver celle i det femhundreogtolv (512) cellesystemet er tildelt en kode slik at en kode blir brukt i mer enn en celle på et gitt mottaksområde.

- 20 Derfor vil cellesøkingssynkroniseringssystemet bestemme den primære forvrengningskoden til en celle som bruker en initiell cellesøkealgoritme. Felles initiell cellesøkealgoritme bruker tre (3) hovedalgoritmer: en trinn 1 algoritme bestemmer PSC og bestemmer et bitavvik, en trinn 2 algoritme bruker informasjonen gitt i trinn 1 og detekterer vindusavviket og kodegruppetallet, og en trinn 3 algoritme bruker
- 25 informasjonen gitt i trinn 2 algoritmen og detekterer den primære forvrengningskoden. Uheldigvis vil hver trinnsalgoritme ha en innbygd feil assosiert med seg. Feilen som er til stede i hvert trinn er forårsaket av UE deteksjonen av støyen assosiert med mottatt felles nedlinkskanal, som kan resultere i et høyt antall av falske deteksjoner.

- 30 Det er også slik at den felles initielle cellesøkealgoritmen ikke kan håndtere en avvisning fra det øvre laget i det gale offentlige landbaserte mobilnettverket (PLMN). Siden de fleste algoritmer detekterer den sterkeste cellen i den felles nedlinkskanalen, er det sannsynlig at hver gang algoritmen lokaliserer en celle med den samme PLMN vil den bli assosiert med cellen. Dette resulterer i den fastlåsing og til slutt gi en indikasjon
- 35 til UE at det ikke er noen tjeneste.

Følgelig eksisterer det et behov for et system og en fremgangsmåte som reduserer antallet av falske deteksjoner i den initielle celledeteksjonsalgoritmen og som er i stand til å overkomme fastlåsing assosiert med en avvisning på grunn av den gale PLMN.

5 Oppfinnelsen er angitt i patentkravene.

Det er beskrevet et system og en fremgangsmåte for å utføre initiell celledeteksjon og å etablere en kommunikasjonslink mellom en UE og en basestasjon i et kommunikasjonsnettverk. Kommunikasjonssignalet er først prosessert for å generere en indeksverdi og bitavvik assosiert med en primær synkroniseringskode, som produserer en første beslutning. En topprøve av kommunikasjonssignalet blir trukket ut og en andre prosessering blir utført på kommunikasjonssignalet. Den andre prosesseringen gjenvinner kodegruppetalet, vindusavviket og andre synkroniseringskode fra kommunikasjonssignalet, og produserer en andre beslutning. En tredje prosessering blir utført på kommunikasjonssignalet som svar på kodegruppetalet og vindusavviket, som gjenvinner den primære forvrengningskoden. Den primære forvrengningskoden blir så brukt for å synkronisere UE til cellen assosiert med den primære forvrengningskoden. Gjennom prosessen vil vindusutelukkelseslogikk bli anvendt for å øke screeningen av frekvensbåndene mens det utføres den initielle kodedeteksjonen. Gjennom et system av buffere og tellere, vil avviste bitavvik og avviste primære forvrengningskoder bli lagret med den hensikt for kontroll og sammenligning av vindusutelukkelseslogikken, som så kan omstarte beslutningsprosessen ved deteksjon av en gal PLMN, som derved utelukker en fastlåsningsbetingelse.

25 Figur 1 er en illustrasjon av det initielle celledeteksjonssystemet laget i henhold til den foretrukne utførelsen av den foreliggende oppfinnelsen.

Figur 2 er en illustrasjon av den fysiske synkroniseringskanalen (PSCH).

30 Figur 3 er et blokkdiagram av trinn 1 modulen i henhold til den foretrukne utførelsen av den foreliggende oppfinnelsen.

Figur 4 er et flytdiagram av trinn 1 modulen i henhold til den foretrukne utførelsen av den foreliggende oppfinnelsen.

35

Figur 5 er et blokkdiagram av trinn 2 modulen i henhold til den foretrukne utførelsen av den foreliggende oppfinnelsen.

Figur 6 er en grafisk illustrasjon av "Fast Hadamard Transform" (FHT) strukturen.

Figur 7 er en illustrasjon av innmatningsmatrisestrukturen i henhold til den foretrukne
5 utførelsen av den foreliggende oppfinnelsen.

Figur 8 er en illustrasjon av kodegruppematisestrukturen i henhold til den foretrukne
utførelsen av den foreliggende oppfinnelsen.

10 Figur 9 er en illustrasjon av korrelasjonsmatrisestrukturen i henhold til den foretrukne
utførelsen av den foreliggende oppfinnelsen.

Figurene 10 A og 10 B viser et flytdiagram av trinn 2 algoritmen i henhold til den
foretrukne utførelsen av den foreliggende oppfinnelsen.

15

Figur 11 er et blokkdiagram av trinn 3 modulen i henhold til den foretrukne utførelsen
av den foreliggende oppfinnelsen.

Figur 12 er et blokkdiagram av trinn 3 korrelatoren i henhold til den foretrukne
20 utførelsen av den foreliggende oppfinnelsen.

Figur 13A og 13B er et flytdiagram av trinn 3 algoritmen i henhold til den foretrukne
utførelsen av den foreliggende oppfinnelsen.

25 Figur 14A og 14B viser et flytdiagram av kontrollercellesøkebeslutningslogikken i
henhold til den foretrukne utførelsen av den foreliggende oppfinnelsen.

Figurene 15A og 15B viser et flytdiagram av kontrollervindusutelukkelseslogikken i
henhold til den foretrukne utførelsen av den foreliggende oppfinnelsen.

30

Cellesøkesynkroniseringssystemet 10 i henhold til den foretrukne utførelsen av den
foreliggende oppfinnelsen er illustrert i figur 1. Systemet 10 innbefatter en trinn 1
modul 12, en trinn 2 modul 14, en trinn 3 modul 16 og en kontroller 18 for å oppnå
synkroniseringen mellom et brukerstyr (UE) og en basestasjon. Med den hensikt å
35 oppnå denne synkroniseringen vil UE gjennom cellesøkesynkroniseringssystemet 10
bruke en initiell cellesøkealgoritme som her blir lagt frem.

Trinn 1 algoritmen til den initielle celledøkealgoritmen blir oppnådd ved å bruke trinn 1 modulen 12. Med referanse til figur 3 vil trinn 1 modulen 12 innbefatte to hierarkiske Golay korrelatorer (HGC) 21, 22, to absolutte verdimodifikatorer (AVM) 23, 24, en beslutningskrets 25, en normaliseringskrets 26, en oppslagstabell 27, en multiplikator
 5 28, en splitter 19 og en trinn 1 komparator 29. Det potensopphøyde cosinusfilteret (RRCFIR) 1 som er vist er ikke en del av trinn 1 modulen 12, men er illustrert her for å gi et komplett bilde.

Hensikten med trinn 1 modulen 12 er å finne den sterkeste veien over en rammeverdi av prøver UE har detektert og å bestemme bitavviket til den sterkeste veien. RRCFIR 1
 10 koblet til splitteren 19 er et pulsformet filter som samler nedlinkskommunikasjonssignalet fra basestasjonen med to ganger bitraten og sender videre det samplede signalet til splitteren 19. Splitteren 19 splitter det samplede signalet i dets like og odde sampler og gir dem videre til HGC 21, 22.

15 HGC 21, 22 er koblet til AVM 23, 24 og sampselvelgeren 34 i trinn 2 modulen 14 (illustrert i figur 5), som skal bli fremlagt heretter. HGC 21, 22 korrelerer PSC til innmatningssignalet. Som en fagmann vet, vil HGC 21, 22 sende ut den komplekse verdien til det like og odde sampelet til inngangssignalet respektivt. HGC 21, 22
 20 utsignaler blir sendt videre til AVM 23, 24 og sampselvelgeren 34.

AVM 23, 24 koblet til HGC 21, 22 og beslutningskretsen 25, bestemmer størrelsen til HGC 21, 22 ligningen for å generere størrelsen som blir bestemt i henhold til følgende ligning:

$$25 \quad \text{abs}(x) \sim \max(|x_{\text{real}}|, |x_{\text{imag}}|) + 0,5 * \min(|x_{\text{real}}|, |x_{\text{imag}}|) \quad \text{Ligning 1}$$

Bruken av den tilnærmede absolutte verdien i henhold til ligning 1 reduserer hardware påkrevd i denne implementeringen og forårsaker ikke noe signifikant
 30 ytelsesdegradering. Med en gang den tilnærmede absolutte verdien har blitt bestemt av AVM 23, 24 respektivt, vil det modifiserte like og odde samplet bli sendt ut til en beslutningskrets 25.

Beslutningskretsen 25, koblet til AVM 23, 24 og kontrolleren 18, bestemmer bitavviket.
 35 Det modifiserte like og odde samplet som er sendt ut fra AVM 23, 24 blir matet inn i en MUX 8 i beslutningskretsen 25, og blir kombinert til en enkel strøm. Denne strømmen er representasjonen av styrken til signalet sendt i et av samplene til hvert vindu i hver

ramme. Som illustrert i figur 2, er det totusenfemhundreogseksti (2560) biter i hvert vindu og femten (15) vinduer i hver ramme. Siden inngangssignalet blir samplet med to ganger bitraten, er det 5120 sampler i hvert vindu. Derfor vil beslutningskretsen 25 bestemme lokasjonen av PSC i signalet, bitavviket, ved å feie gjennom de 5120
 5 akkumulerte samplene i enden av hvert vindu.

Strømmen som er generert av MUX blir sendt videre til en akkumulator (ikke vist) i beslutningskretsen 25. Denne akkumulatoren har femtusenettthundreogtjue (5120) sampel langt register som lagrer den akkumulerte sampelverdien for hvert vindu for
 10 hver ramme, og opererer på vindusraten. Styrken til signalet for hvert sampel i et vindu blir lagt til styrken av signalet for hvert sampel i hvert etterfølgende vindu. Som et eksempel vil samplet i vindu 1 innbefatte følgende signalstyrkeverdier {1,5,3,7}, samplene til vindu 2 innbefatter følgende signalstyrkeverdier {2,4,8,3}. Initielt vil registrene i akkumulatoren ha verdiene {0,0,0,0}. Etter hvert som hver sampelverdi fra
 15 vindu 1 blir lagt til registrene i akkumulatoren, vil registerverdiene forandre seg følgende. For eksempel, når den første sampelverdien til vindu 1 blir lagt til den første registerverdien, vil akkumulatoren ha verdiene {1,0,0,0}, når den andre prøven til vinduet 1 blir lagt til den andre registerverdien, vil akkumulatoren ha verdiene {1,5,0,0} og så videre. Med en gang den siste sampelverdien til vindu 1 blir lagt til
 20 akkumulatoren vil den første sampelverdien til vinduet 2 bli lagt til det første registeret i akkumulatoren, som resulterer i at akkumulatoren har verdiene {3,5,3,7}, når den andre sampelverdien i vindu 2 blir lagt til den andre registerverdien vil akkumulatoren ha verdiene {3,9,3,7}. Den foretrukne utførelsen av den foreliggende oppfinnelsen vil flushe registrene i akkumulatoren etter at fem (5) rammer har blitt akkumulert, som er
 25 ekvivalent med syttifem (75) vinduer. Antallet av akkumulerte rammer blir talt med en trinn 1 teller (ikke vist) i beslutningskretsen 25.

En beslutning, bestemmelsen av bitavviket, i beslutningskretsen 25 blir generert i enden av hver ramme, femten (15) vinduer. Beslutningskretsen 25 bestemmer hvilket register
 30 i akkumulatoren som har maksimalt akkumulerte sampelverdier MAX og tildeler en indeks til den. Indeksen samsvarer med halve bitlokasjonen til PSC signalet for basestasjonen med det sterkeste signalet.

Bitavviktildelingen blir bestemt ved å bruke HGC avviksverdien til 511. For en
 35 fagmann vil utsignalet til HGC bli forsinket med 256 biter. Derfor, når beslutningskretsen 25 tildeler en indeks til toppverdien, vil HGC avviksverdien bli trukket fra. Siden PSC er 256 biter lang, 512 sampler lang, vil fratrekk av HGC avvike

6

fra indeksen være lik med å sette bitavviket til begynnelsen av vinduet. Dersom indeksen som er generert av beslutningskretsen 25 blir større enn HGC avviksverdien på 511, så vil bitavviket bli beregnet i henhold til ligning 2 nedenfor:

$$5 \quad \text{chip_offset} = \text{INDEX} - 511 \quad \text{Ligning 2}$$

Dersom indeksen er mindre enn HGC avviksverdien så vil bitavviket bli beregnet i henhold til ligning 3 nedenfor:

$$10 \quad \text{chip_offset} = 5120 + \text{INDEX} - 511 \quad \text{Ligning 3}$$

Som illustrert i figur 3 vil beslutningskretsen 25 også innbefatte en maskegenerator 5, som blir brukt til å ekskludere et vindu rundt et avvist bitavvik fra beslutningen i beslutningskretsen 25. Denne maskegeneratoren 5 forhindrer dermed
 15 beslutningskretsen 25 fra å utnytte en indeks assosiert med et avvist bitavvik. Detaljene til maskegeneratoren 5 vil bli lagt frem heretter.

Det beregnede bitavviket og rammetellertrinn 1 telleren er sendt ut til en kontroller 18 som blir fremlagt heretter. Beslutningskretsen 25 sender også ut maksimalt akkumulert
 20 bitverdi MAX og den akkumulerte bitverdien OUTPUT for alle registre.

Den akkumulerte bitverdien OUTPUT for alle registrene blir matet ut til en normaliseringskrets 26 hvor den blir samlet med 20% av bitraten (en ut av fem), summert, og så normalisert til 1024. Rammetellertrinnet 1 telleren blir matet ut til
 25 oppslagstabellen 27 for å bestemme den korrekte forsterkningsfaktoren basert på antallet av rammer som er akkumulert. Utgangen av normaliseringskretsen 26 og oppslagstabellen 27 blir så multiplisert i multiplikatoren 28. Utgangen av multiplikatoren 28 blir betraktet som støyterskelen blir sendt videre til en trinn 1 komparator krets 29 som blir sammenlignet med maksimalt akkumulert sampelverdi
 30 MAX. Dersom den maksimalt akkumulerte sampelverdien MAX er større enn støyterskelen, vil den differensielle forsterkeren 29 sende ut et fast høyt trinn 1 signal til kontrolleren, som indikerer en god beslutning for trinn 1, ellers vil et lavt signal bli sendt ut.

35 Som sagt tidligere vil bitavviket og andre utganger bli bestemt i enden av hver ramme. Derfor vil påliteligheten til den første beslutningen være mindre enn den for den andre fordi den andre beslutningen blir gjort over tretti vinduer istedenfor femten vinduer.

Påliteligheten øker etter hvert som antallet av vinduer som blir akkumulert øker. Det høyeste pålitelige utgangssignalet blir generert i den M1^{te} rammen, der M1 er et heltall større eller lik med en (1). Kontrolleren 18 nullstiller rammetellertrinn 1 telleren og akkumulatorregistrene i enden av hver M1^{te} ramme. Ytelsesresultatene under

5 forskjellige kanalsvekkelser viser at fem rammeintegrasjoner er tilstrekkelig for å detektere PSC. Imidlertid, kan denne integreringen bli forandret til flere eller færre rammer.

Et flytdiagram av trinn 1 modulen er illustrert i figur 4. UE detekterer mottaket av

10 kommunikasjonene over den felles nedlinkskanalen (trinn 401) og sampler signalet med to ganger bitraten og genererer like og odde sampler (trinn 402). Disse like odde sampler blir gitt videre til den hierarkiske Golay korrelatoren (HGC) 21, 22 (trinn 403). HGC 21, 22 sender så videre utgangene til AVM 23, 24 og sampelveelgeren 34 (trinn 404). AVM 23, 24 tilnærmer størrelsesorden til de like og odde utgangene mottatt fra

15 HGC 21, 22 (trinn 405) og sender dem videre til beslutningskretsen 25 (trinn 406). Ved mottak av utgangsstørrelsene vil beslutningskretsen 25 kombinere størrelsene (trinn 407), som representerer signalstyrken til signalet sendt i et av samplene for hvert vindu i hver ramme. Signalstyrken for hvert sampel blir akkumulert for alle vinduer innenfor hver ramme (trinn 408). Beslutningskretsen 25 bestemmer så hvilket sampel i rammen

20 som har maksimalt akkumulert sampelverdi (trinn 409) og tildeler en indeks til det (trinn 410). Basert på indeksen vil en bitverdi bli tildelt til indeksen (trinn 411), kjent som bitavviket, og sendes ut til kontrolleren 18 (trinn 412). En støytterskelverdi blir så generert ved å bruke den akkumulerte bitverdien for alle sampler og rammetelleren (trinn 413) og så sammenlignet med maksimalt akkumulert sampelverdi (414), som

25 indikerer en fast eller tentativ beslutning til kontrolleren 18 (trinn 415).

Med referanse tilbake til figur 1, vil utgangene av trinn 1 modul 12, bitavviket, det faste trinn 1 og trinn 1 telleren, bli sendt videre til kontrolleren 18. Kontrolleren 18 sender videre bitavvikene til trinn 2 modul 14. Som sagt ovenfor, vil trinn 2 modulen 14 bruke

30 en trinn 2 algoritme som tar bitavviket sendt ut fra trinn 1 og HGC 21, 22 sender ut og detekterer vindusavvikene og kodegruppetallet. Trinn 2 modulen 14 illustrert i figur 5, innbefatter en trinn 2 komparator 30, en forsinkelse 32, en sampelveelger 34, en konjugerer 36, en kompleks multiplikator 38, en "Fast Hadamard Transform" (FHT) 33, en omhylningsfjerner 31, en inngangsmatrisegenerator 35, en RS koder 37 og en trinn 2

35 beslutningskrets 39.

Hensikten med trinn 2 algoritmen er å gi trinn 3 algoritmen forvrengningskodegruppetallet og vindusavviket. Bitavviket fra trinn 1 modulen 12 blir sendt fra kontrolleren 18 til forsinkelsen 32 i trinn 2 modulen 14. Bitavviket blir
forsinket i en ramme gjennom forsinkelsen 32 for å tillate trinn 1 modulen å gjøre en
5 første beslutning. Det forsinkede bitavviket blir så sendt videre til sampelvelgeren 34 som er koblet til forsinkelsen 32, en konjugerer 36 og HGC 21, 22 i trinn 1 modulen 12. Ved å bruke indeksen bestemt av beslutningskretsen 25, vil sampelvelgeren 34 trekke ut topp HGC 21, 22 utgangene fra innmatningssignalet, som så blir konjugert i konjugererens 36 og sendt ut til den komplekse multiplikatoren 38.

10

Det samme kommunikasjonssignalet i trinn 1 modul 12 blir matet inn til en opplinjeringsskrets 15, som opplinjrer inngangssignalet slik at trinn 2 modul 14 begynner sin søking for forvrengningskodegruppetallet og vindusavviket i begynnelsen av vinduet. Med en gang signalet er opplinjert vil opplinjeringsskretsen 15 sende videre
15 dette til trinn 2 modulen 14. Selv om det er totusenfemhundreogseksti (2560) biter i hvert vindu, vil det være åpenbart fra figur 2 at PSC er lokalisert innenfor de første 256 bitene i hvert vindu. Siden bitavviket har blitt bestemt i trinn 1 modulen, vil trinn 2 modulen bestemme SSC ved å bruke plasseringen til den sterkeste PSC i de første 256 bitene i hvert vindu. Som en fagmann vet, når SSC kodene blir generert, vil en
20 omhylningssekvens bli påtrykket radene i en Hadamard matrise med den hensikt å ha noe ortogonalitet mellom PSC og SSC kodene. Denne omhylningen må bli fjernet før man går videre i den gjenværende delen av trinn 2 algoritmen. Denne omhylningsfjerningen blir oppnådd i omhylningsfjernerens 31.

25 Med en gang omhylningen har blitt fjernet fra inngangssignalet, blir signalet sendt ut fra omhylningsfjernerens 31 til FHT transformerens 33 koblet til omhylningsfjernerens 31 og multiplikatoren 38, som reduserer kompleksiteten til den rene Hadamard korrelasjonsoperasjonen. Figur 6 er en illustrasjon av FHT strukturen. Utgangen av FHT transformerens 33 blir multiplisert med den konjugerte til toppen av HGC 21, 22 i
30 kompleksmultiplikatoren 38 koblet til konjugererens 36 og FHT transformerens 33. Bruken av den konjugerte av toppen sendt ut fra HGC 21, 22 gir en fasekorreksjon til FHT utgangen og transformerer en inngang som samsvarer med den sendte SSC koden til den reelle aksens.

35 Med en gang FHT transformerens 33 sender ut og har blitt multiplisert i den komplekse multiplikatoren 38, vil realdelen til FHT bli sendt ut og sendt videre til inngangsmatrisegeneratoren 35 fra multiplikatoren 38, som putter FHT utgangene inn i

en realmatrise på 15 x 16, kalt inngangsmatrisen. I inngangsmatrisen er det femten (15) vinduer og i hvert vindu 16 elementer for en ramme. Inngangsmatrisen blir oppdatert pr. ramme. Inngangsmatrisen blir så sendt videre til beslutningskretsen 39 hvor en bestemmelse av vindusavviket og kodegruppenummeret blir gjort. Strukturen til
5 inngangsmatrisen er illustrert i figur 7.

En korrelasjonsmatrise blir generert i trinn 2 beslutningskretsen 39 ved å bruke inngangsmatrisen 35 og en kjent kodegruppematrikse, som resulterer i en 64 x 15 matrise. Korrelasjonsmatrisen blir nullstilt når rammetelleren for trinn 2 modulen når
10 M2, tilsvarende det som blir lagt frem i trinn 1 modulen. Med den hensikt å generere korrelasjonsmatrisen vil beslutningskretsen 39 gå gjennom hvert av elementene i kodegruppematriksen og elementene i inngangsmatrisen 35 i samsvar med ligning 4 nedenfor:

$$\begin{aligned} & \text{corrMatrix}[i][j] += \\ & \text{inputMatrix}[k][\text{code_group_matrix}[i][k]] \end{aligned} \quad \text{Ligning 4}$$

hvor j er en heltall som økes fra 0 til 14 med 1, som representerer sykliske skift utført i identitetsmatrisen med hensyn til kolonnene, i er et heltall som økes fra 0 til 63 med 1,
20 og k er et heltall som blir økt fra 0 til 14 med 1. Strukturen til kodegruppematriksen og den resulterende korrelasjonsmatrisen er illustrert i figurene 8 og 9 respektivt. Med en gang korrelasjonsmatrisen har blitt generert, vil maksimal innmating bli funnet av beslutningskretsen 39. Den tilsvarende raden til den maksimale innmatingen som blir funnet er kodegruppetalet og kolonnen er vindusavviket.

25 Tilsvarende med trinn 1 modulen 12, dersom maksimal korrelasjon MAX 2 er større enn terskelen, vil komparatorkretsen 30 sende ut et fast høyt trinn 2 signal til kontrolleren 18 som indikerer en fast beslutning, ellers blir et lavt signal sendt ut som indikerer en tentativ beslutning. Terskelverdien blir beregnet ved å bruke
30 gjennomsnittlig størrelsesverdi i korrelasjonsmatrisen:

$$Th = k \frac{1}{960} \left(\sum_{i=0}^{63} \sum_{j=0}^{14} \text{mag}(c_{ij}) \right)$$

Ligning 5

$$k = 5.12, P_{fa} = 10^{-4}$$

hvor P_{FA} er sannsynligheten for falsk alarm. Trinn 2 modul 14 sender ut til kontrolleren 18 kodegruppetalet, vindusavviket, det faste trinn 2 signalet, og trinn 2 telleren.

Flytdiagrammet for trinn 2 algoritmen er illustrert i figur 10. Trinn 2 modulen mottar 5 kommunikasjonssignalet fra basestasjonen over nedlinkskanalen (trinn 1001). En omhylningssekvens blir fjernet fra kommunikasjonssignalet (trinn 1002a) og sendt ut til en FHT transformerer 33 (trinn 1003a). På samme tid vil bitavviket fra trinn 1 modulen 12 bli matet inn til en forsinkelse 32 i trinn 2 modulen 14 (trinn 1002b) og sendt videre til en sampelvelger 34, som trekker ut toppen for like og odde utganger generert av 10 HGC 21, 22 i trinn 1 modulen 12 basert på bitavviket (trinn 1003b). Utgangen fra FHT transformerer 33 blir så multiplisert med den konjugerte til toppverdien for den like eller odde sampel sendt ut fra sampelvelgeren 34 (trinn 1004) og transformerer en inngang i FHT utgangen som samsvarer med SSC koden på den reelle aksen (trinn 1005). Realdelen til FHT sendes ut for hvert vindu i ramme som blir sendt videre til 15 inngangsmatrisegeneratoren 35 (trinn 1006). Inngangsmatrisegeneratoren 35 skaper så inngangsmatrisen (trinn 1007), inngangsmatrisen blir så sendt videre til beslutningskretsen 39 som bestemmer vindusavviket og kodegruppetalet (trinn 1008). Ved å bruke inngangsmatrisen og kjent kodegruppematrise, vil beslutningskretsen 39 generere en korrelasjonsmatrise (trinn 1009). Med en gang korrelasjonsmatrisen har 20 blitt generert, vil beslutningskretsen 39 lokalisere maksimal innmatning i korrelasjonsmatrisen (trinn 1010), for hvilket den tilsvarende raden i den maksimale innmatningen som blir funnet blir bestemt som kodegruppetalet og kolonnen blir vindusavviket. Kodegruppetalet og vindusavviket blir så sendt videre til kontrolleren 18 (trinn 1011). En terskelverdi blir så beregnet ved å bruke gjennomsnittlig 25 størrelsesverdi i korrelasjonsmatrisen (trinn 1012) og sammenlignet med maksimal korrelasjon (trinn 1013), og sende videre en indikasjon om en fast eller tentativ beslutning til kontrolleren 18 (trinn 1014).

Bitavviket sendt ut fra trinn 1 modulen 12 og vindusavviket og kodegruppetalet sendt 30 ut fra trinn 2 modulen blir sendt videre av kontrolleren 18 til trinn 3 modulen 16, som bruker en trinn 3 algoritme med den hensikt å bestemme hvilken av dem som har den primære forvrengningskoden som kommer med den minst sannsynlige falske alarmen (P_{FA}) når kodegruppetalet blir gitt. Det er åtte primære forvrengningskoder i hver kodegruppe.

35

Blokkdiagrammet i trinn 3 modulen 16 er illustrert i figur 11. Tilsvarende til trinn 2 modulen 14 blir kommunikasjonssignalet matet til en andre opplinjeringsskrets 17 som

opplinjerer inngangssignalet slik at trinn 3 modulen 16 begynner sin søking etter forvrengningskodetallet i begynnelsen av rammen. Med en gang inngangssignalet har blitt opplinjert vil opplinjeringsskretsen 17 sende videre inngangssignalet til trinn 3 modulen 16. Trinn 3 modulen innbefatter åtte (8) forvrengningskodegeneratorene 5 $40_1 \dots 40_8$, åtte (8) korrelatorkretser $41_1 \dots 41_8$, en støyestimeringskrets 42, en trinn 3 beslutningskrets 44, en beslutningsunderstøttelseskrets 45, en forsterkningskrets 46, og en komparatorkrets 47. Kodegruppetalet generert i trinn 2 modulen 14 blir matet til de åtte (8) forvrengningskodegeneratorene $40_1 \dots 40_8$ og forvrengningskodene blir generert fra disse. Utgangen fra forvrengningskodegeneratorene $40_1 \dots 40_8$ blir sendt videre til 10 forvrengningskodekorrelatorene $41_1 \dots 41_8$ respektivt.

Sammen med forvrengningskodene sendt ut fra forvrengningskodegeneratorene $40_1 \dots 40_8$ vil kommunikasjonssignalet etter prosessering i opplinjeringsskretsen 17 bruke bitavviket og vindusavviket sendt ut fra kontrolleren 18, som blir matet inn til 15 korrelatorene $41_1 \dots 41_8$. Korrelatorene $41_1 \dots 41_8$ bruker ikke-koherent integrasjon over et visst antall av vinduer. Integrasjonen kan være over multiple rammer. Korrelasjonen blir gjort koherent for hvert symbol som samsvarer med de 256-bit dataene. Den absolutte verdien til korrelasjonsresultatene blir akkumulert over $10 \cdot N$ symboler pr. ramme, der N er antallet av vinduer som blir akkumulert fra begynnelsen av en ramme. 20 I et enkelt vindu er det ti 256-bit lange datadelar, derfor vil ti 256-bit koherent bitkorrelasjoner og ti akkumulasjoner bli gjort pr. vindu. Figur 12 viser detaljene til en korrelator 41_1 .

Etter at korrelatorene $41_1 \dots 41_8$ genererer utgangene, vil maksimal utgang og dens 25 indeks måtte bli funnet. Trinn 3 beslutningskretsen 44 tar utgangene fra forvrengningskodekorrelatorene $41_1 \dots 41_8$, bestemmer korrelatoren $41_1 \dots 41_8$ med den maksimale utgangen, og genererer en indeks fra dette. Indeksen er forvrengningskodetallet. Forvrengningskodetallet blir så sendt videre til beslutningsunderstøttelseskretsen 45 og kontrolleren 18. 30 Beslutningsunderstøttelseskretsen 45 observerer den siste M3 beslutningen gjort av beslutningskretsen 44. Dersom en kode gjentar seg selv mer enn k repetisjoner ut av M3 innganger, vil koden som har blitt gjentatt være forvrengningskodetallet som er sendt ut fra beslutningsunderstøttelseskretsen 45 bare bli brukt når det ikke er noen fast beslutning over etterfølgende M3 rammer. Selv om beslutningsunderstøttelseskretsen 35 bare er illustrert i trinn 3 modulen 16, kan en beslutningsunderstøttelseskrets 45 som fremlagt i trinn 3 modulen 16 bli brukt for både trinn 1 og trinn 2 modulene 12, 14 fremlagt ovenfor.

En fast beslutning blir indikert når den bestemte maksimale korrelasjonsverdien er større enn den beregnede terskelverdien. Terskelverdien blir beregnet ved å bruke støyestimatorkretsen 42, som blir brukt for støymåling, og en forsterkningsfaktor.

- 5 Støyen blir bestemt ved å ta størrelsesorden til forskjellen mellom etterfølgende felles pilotsymboler. Denne fremgangsmåten for støyestimering eliminerer enhver bias i støyestimeringen på grunn av ortogonal signalinterferens. Resultatet fra støyestimatoren 42 blir multiplisert med forsterkningsfaktoren i multiplikatoren 46, som er bestemt til å være terskelen. Når den bestemte maksimale korrelasjonen er større enn
10 den beregnede terskelen, vil komparatoren 47 sende ut et høyt trinn 3 fast signal som indikerer en fast beslutning, ellers et lavt signal som indikerer en tentativ beslutning.

- Flytdiagrammet for trinn 3 algoritmen er illustrert i figur 13. Kodegruppetalet sendt ut fra trinn 2 modulen 14 blir matet til trinn 3 modulen 16, forvrengningskodegeneratorene
15 $40_1 \dots 40_8$ (trinn 1301), som så genererer forvrengningskoden fra disse (trinn 1302).

- Utgangene fra forvrengningskodegeneratorene blir så sendt videre til forvrengningskodekorrelatorene $41_1 \dots 41_8$ (trinn 1303). Sammen med forvrengningskodene sendt ut fra forvrengningskodegeneratorene $40_1 \dots 40_8$, vil kommunikasjonssignalet bli korrelert i forvrengningskodekorrelatorene $41_1 \dots 41_8$ (trinn
20 1304), som så genererer ti 256-bit koherent korrelasjoner og ti ikke-koherente akkumulasjoner pr. tidsvindu (trinn 1305). De akkumulerte resultatene blir sendt videre til trinn 3 beslutningskretsen 44 (trinn 1306). Beslutningskretsen 44 bestemmer korrelatoren som har maksimal utgang og genererer en indeks fra dette som er forvrengningskodetallet (trinn 1307). En terskelverdi blir så beregnet (trinn 1308) og
25 sammenlignet med maksimal korrelasjonsverdi (trinn 1309). Dersom maksimal korrelasjonsverdi er større enn den beregnede terskelen, vil trinn 3 modulen 16 sende ut et høyt trinn 3 fast signal (trinn 1310), som resulterer i at beslutningskretsen 44 sender ut forvrengningskodetallet til kontrolleren 18 (trinn 1311). Ellers vil et lavt signal bli sendt ut til kontrolleren 18 (trinn 1312) og forvrengningskodetallet blir sendt ut til
30 beslutningsunderstøttelseskretsen 45 (trinn 1313). Siden beslutningsunderstøttelseskretsen 45 observerer de siste M3 beslutningene gjort i beslutningskretsen 44, vil et forvrengningskodetall bli sendt ut til kontrolleren 18 når forvrengningskoden gjentar seg selv k ganger ut av M3 innmatninger (trinn 1311).

- 35 Med referanse tilbake til figur 1, innbefatter kontrolleren 18 en avvist bitavviksbuffer 9, en avvist bitavviksteller 11, en avvist primær forvrengningskodevektorbuffer 13, en avvist primær forvrengningskodeteller 3, en beslutningslogikkrets 2 og en

vindusutelukkelseslogikkrets 6. Kontrolleren 18 blir brukt til å lage bedre beslutninger under hele cellesøkealgoritmen i henhold til den foretrukne utførelsen av den foreliggende oppfinnelsen.

5 Flytdiagrammet til beslutningslogikken brukt av kontrolleren 18 for å bestemme den primære forvrengningskode for synkronisering med den sendende basestasjonen er illustrert i figur 14. Kontrolleren 18 mottar bitavviket, det faste trinn 1 signalet og trinn 1 tellersignalet fra trinn 1 modulen 12 (trinn 1401). Dersom det faste trinn 1 signalet er høyt, vil kontrolleren 18 sende videre det faste bitavviket til trinn 2 modulen 14
 10 (trinn 1402a), ellers vil et tentativt bitavvik bli sendt videre (trinn 1402b). Trinnet 2 modul 14 genererer kodegruppetallet, vindusavviksverdi, trinn 2 fast signal, og trinn 2 teller (trinn 1403). Dersom det faste trinn 2 signalet er høyt, vil kontrolleren sende videre den faste kodegruppen til trinn 3 modulen (trinn 1404a). Ellers vil kontrolleren 18 sende videre et tentativt kodegruppetall til trinn 3 modulen 16 (trinn 1404b) og
 15 dersom trinn 2 telleren er mindre enn M2, vil trinn 2 modulen 14 fortsette å generere kodegruppetallet (trinn 1403). Dersom trinn 2 telleren er lik med M2, så vil trinn 2 modulen 14 bli nullstilt (trinn 1407), som resulterer i at trinn 2 modulen genererer et kodetall og bitavvik (trinn 1403). Trinn 3 modulen 16 genererer så et forvrengningskodetall og et fast trinn 3 signal (trinn 1405) generert i trinn 1403, og tar
 20 i mot vindusavviket og kodegruppetallet. Dersom det faste trinn 3 signalet er høyt, så vil beslutningslogikkretsen 2 bestemme at forvrengningskodetallet er fast og slutter beslutningslogikkprosessen. Dersom det faste trinn 3 signalet er lavt og det faste trinn 1 signalet høyt eller trinn 2 telleren er mindre enn M2, vil trinn 2 modulen fortsette å generere et kodegruppetall (trinn 1403). Ellers vil trinn 2 modulen motta et
 25 nullstillingssignal fra kontrolleren 18 og nullstiller trinn 2 telleren til 0 (trinn 1407). Denne prosedyren fortsetter helt til beslutningen sendt ut i trinn 3 modulen 16 er fast.

På grunn av en mulig initiell frekvensfeil i VCO, kan overskytende tap av signalkorrelasjon finne sted. Derfor vil VCO frekvensen bli forandret med den hensikt
 30 å kontrollere maksimal mulig frekvensfeil mellom UE og cellen. Ved initialisering av UE vil kontrolleren 18 initialisere cellesøkefrekvensen ved å bruke frekvenssyntetisereren 20. Med referanse til figur 1 vil frekvenssyntetisereren 20 innbefatte en adaptiv frekvenskrets (AFC) og spenningskontrollert oscillator (VCO) 7 eller numerisk kontrollert oscillator (NCO). AFC 4 koblet til kontrolleren 18 og VCO
 35 7 innbefatter en frekvenstildelingstabell (FAT) og en frekvenstrinntabell (FST).

- Når kontrolleren 18 blir initialisert vil AFC 4 sette frekvensen ved å bruke den første frekvensen i FAT og avviksverdien fra FST. Denne initielle frekvensen er frekvensen som er brukt i kontrolleren 18 for å utføre celledøkingen. FST er en tabell med trinnfrekvenser, eller avviksfrekvenser, for eksempel $\{9m2, -2,4, -4, 6, -6...N, -N\}$ som
- 5 blir brukt til å avvikle frekvensen brukt i kontrolleren 18. FAT inkluderer et mangfold av forhåndsbestemte frekvenser for hvilken kontrolleren 18, eller en nivå 1 kontroller (ikke vist) bruker for å lokalisere og synkronisere UE til basestasjonen. På grunn av denne fremstillingen vil mangfoldet av frekvenser som er opplistet være definert som $F_0, F_1, F_2...F_N$ i FAT og avviksfrekvensene i FST er definert som
- 10 $SF_0, SF_1, -SF_1, SF_2, -SF_2...SF_N, -SF_N$. Følgelig, når kontrolleren er initialisert, vil avviksfrekvensen være SF_0 og frekvensen $> F_0$. AFC4 kombinerer de to verdiene F_0+SF_0 og sender videre den resulterende frekvensverdien til VCO eller NCO7 som bibeholder UE frekvensen ved denne videresendte frekvensen.
- 15 Kontrolleren 18 utfører beslutningslogikken fremlagt ovenfor. Dersom det etter X antall av rammer er slik at det faste utgangstrinn 3 signalet ikke går høyt, vil kontrolleren signalisere AFC4 til trinn 2 til neste avvik i FST, for eksempel FS_1 . AFC4 vil så kombinere den nye avviksfrekvensen med frekvensen i FAT, F_0+SF_1 , og sender ut den resulterende frekvensen til VCO eller NCO7 for å bibeholde UE på denne
- 20 frekvensen.
- Kontrolleren 18 fortsetter å gå gjennom avviksfrekvensene i FST helt til et høyt signal blir detektert fra trinn 3 modulen 16 som indikerer en fast deteksjon eller helt til alle avviksfrekvensene har blitt forsøkt i kontrolleren 18. Med en gang alle
- 25 avviksfrekvensene har blitt forsøkt, vil AFC4 nullstille FST avviksfrekvensen til SF_0 , gå videre til neste frekvens i FAT, F_1 og kombinere disse to verdiene, F_1+SF_0 , for utsending til VCO eller NCO7. VCO eller NCO7 regulerer så UE frekvensen til denne nye resulterende frekvensen og kontrolleren 18 vil så utføre beslutningslogikken helt til et høyt signal blir detektert fra trinn 3 modulen 16. Denne prosessen med å gå igjennom
- 30 FST og så gå videre til neste FAT frekvens blir fortsatt helt til et høyt signal blir sendt ut fra trinn 3 modulen 16. Med en gang denne hendelsen finner sted vil deteksjonen av en forvrengningskode, låser AFC4 FST avviksverdien på dens nåværende posisjon, som ikke blir justert om igjen helt til kontrolleren 18 blir initialisert.
- 35 For en fagmann vil de fleste tjenestetilbydere i et kommunikasjonssystem ha forskjellige offentlige mobilnettverk (PLMN). UE utnytter den detekterte PLMN for å bestemme om eller ikke tjenestetilbyderen gir tjenester på stedet til UE. Kontrolleren

18 bruker en vindusutelukkelseslogikk innfor vindusutelukkelseslogikkretsen 6 for å bøte på en avvisning på grunn av gal PLMN. Side detekteringen av HGC 21, 22 sender ut en toppverdi som alltid gir den samme PLMN, bruker kontrolleren 18 vindusutelukkelseslogikken for å bøte på denne fastlåsing.

- 5 Vindusutelukkelseslogikkretsen blir koblet til beslutningslogikkretsen 2, avvist bitavvikvektorbuffer 9, en avvist bitavviksteller 11, en avvist primær forvrengningskodevektorbuffer 13, og en avvist primær forvrengningskodeteller 3. Vindusutelukkelseslogikkretsen 6 kontrollerer den primære forvrengningskoden som er sendt ut fra trinn 3 modulen mot den avviste primære forvrengningskoden som er lagret
- 10 i den avvist primærforvrengningskodevektorbufferen 13. Dersom den primære forvrengningskoden som er sendt ut fra trinn 3 modulen er funnet i bufferen 13, eller den gale PLMN er detektert, vil vindusutelukkelseslogikkretsen 6 avvise koden og initialiserer beslutningslogikkretsen igjen. Hver gang en primær forvrengningskode blir avvist, vil bitavviket som ble generert i trinn 1 modulen bli lagret i den avviste
- 15 bitavviksvektorbufferen 9 og brukt av maskegeneratoren 5. Maskegeneratoren 5 til beslutningskretsen 25 innenfor trinn 1 modulen 12 bruker verdien lagret i den avviste bitavviksvektorbufferen 9 og den avviste bitavvikstellersen 11 fra kontrolleren 18 for å bestemme hvilke biter i hvert vindu som skal ekskluderes i vinduet. Utelukkelsen av den detekterte primære forvrengningskoden og bitavvikene er gjort bare innenfor et
- 20 enkelt frekvensbånd. Buffere og tellere blir nullstilt når det er en kvittering fra basestasjonen eller et nytt frekvensbånd blir brukt i nivå 1 kontrolleren.

- For å justere frekvensbåndet som er brukt i kontrolleren 18 under vindusutelukkelseslogikkprosessen, signalerer lag 1 kontrollersignalene til AFC4 om å
- 25 gå videre til neste frekvens i FAT. Siden avviksfrekvensen i FST blir satt, vil AFC kombinere den nye frekvensen med den satte avviksfrekvensen. VCO eller NCO7 blir så justert for å bibeholde denne kombinerte frekvensen.

- Et flytdiagram av vindusutelukkelseslogikken som er brukt i kontrolleren er illustrert i
- 30 figur 15. Kontrolleren 18 kjører celledøkebeslutningslogikken og finner en primær forvrengningskode (trinn 1501). Den primære forvrengningskode blir gitt videre til de øvre lag (trinn 1502) som lagrer frekvensen og den primære forvrengningskodeindeksen (trinn 1503). Dersom PLMN er korrekt for den spesielle tjenestetilbyderen, vil UE bli synkronisert til basestasjonen, og prosessen blir terminert (trinn 1504). Dersom PLMN
- 35 ikke er korrekt og det er en frekvens gjenværende i FAT i AGC4, vil AGC4 gå videre til den neste frekvensen i FAT og kontrolleren 18 forandrer frekvensen, lagrer den primære forvrengningskoden i vektorbufferen 13, og nullstiller celledøkealgoritmen (trinn 1505).

Det bør legges merke til at feilbetingelser monitoreres enten ved tellerbufferen 3, 11 eller en tidtager for å bestemme om en feilbetingelse har funnet sted. En feilet betingelse indikerer at synkroniseringen ikke vil finne sted under de nåværende betingelsene (dvs. frekvens). Dersom det ikke er noen frekvenser tilbake i FAT,

5 begynner kontrolleren 18 å sveipe frekvensene med den lagrede primære forvrengningskoden (trinn 1506). Kontrolleren 18 setter så den første frekvensen og gir videre den avviste primære forvrengningskoden til den initielle celledøkingen med vindusutelukkelsesmetoden (trinn 1507). Kontrolleren 18 nullstiller den initielle celledøkingen med vindusutelukkelsesmetoden og nullstiller også feilbetingelsen (trinn

10 1508). Den avviste primære forvrengningskoden blir puttet inn i den avviste primærforvrengningskodevektorbufferen 13, og den avviste primære forvrengningskodetelleren blir inkrementert (trinn 1509). Cellesøkebeslutningslogikken blir kjørt og en primær forvrengningskode og bitavvik blir funnet (trinn 1510). Dersom den primære forvrengningskoden blir lagret i den avviste

15 primærforvrengningskodevektorbufferen 13, så vil bitavviket bli puttet videre inn i den avviste bitavviksbufferen 9 og den avvist bitavvikstelleren 11 blir inkrementert (trinn 1511). Cellesøkebeslutningslogikken blir igjen kjørt som utelukker et vindu rundt det avviste bitavviket (trinn 1512). Dersom den primære forvrengningskoden generert i denne celledøkebeslutningslogikken igjen blir lagret i den avviste primære

20 forvrengningskodevektorbufferen, så vil det detekterte bitavviket bli puttet inn i den avviste bitavviksvektorbufferen og den avviste bitavvikstelleren blir økt (trinn 1511) og celledøkebeslutningslogikken utelukker et vindu med verdi lik det avviste bitavviket som blir kjørt igjen (trinn 1512). Trinnene 1511 og 1512 fortsetter helt til den detekterte primære koden ikke er i listen ved hvilket punkt den primære forvrengningskoden blir

25 gitt videre til de øvre lag for å avvente en kvittering fra basestasjonen (trinn 1513). Dersom det er en feilbetingelse og det ikke er noen frekvenser igjen, indikerer kontrolleren 18 at ingen tjeneste er tilgjengelig (trinn 1517) og prosessen blir terminert. Dersom det var en feil og det var en frekvens som var igjen i båndbredden, blir kontrolleren 18 satt til en ny frekvens og gir videre den avvist primære

30 forvrengningskoden for denne frekvensen (trinn 1516). Kontrolleren 18 nullstiller så den initielle celledøkingen med vindusutelukkelsesmetoden og feilbetingelsesmonitoreringen (1508). Kontrolleren 18 vil så fortsette den initielle celledøkingen med vindusutelukkelsesmetoden som ble lagt frem ovenfor. Dersom det er ingen feilbetingelse og PLM er korrekt, vil kontrolleren 18 indikere at UE er

35 synkronisert til basestasjonen ved mottak av kvitteringen (trinn 1518), og prosessen blir terminert. Dersom PLMN er ukorrekt, vil den avvist primær forvrengningskoden bli puttet inn i den avviste primære forvrengningskodevektorbufferen 13 og den avviste

primære forvrengningskodetelleren 3 blir økt (trinn 1515).

Cellesøkebeslutningslogikken blir kjørt igjen som ekskluderer et vindu rundt den tidligere avviste bitavviksverdien (trinn 1512). Denne prosedyren fortsetter helt til kontrolleren indikerer at ingen tjeneste er tilgjengelig eller en kvittering fra

5 basestasjonen blir mottatt.

P a t e n t k r a v

1. Brukerutstyr, UE, for å etablere en kommunikasjonslink, omfattende:
 - 5 - en første modul (12) for å prosessere et mottatt kommunikasjonssignal og å generere en indeksverdi assosiert med en primær synkroniseringskode innenfor kommunikasjonssignalet, idet nevnte første modul omfatter:
 - en splitter (19) for å splitte et samplet kommunikasjonssignal til like og odde sampler;
 - 10 - en hierarkisk Golay-korrelator (21, 22), HGC, for å korrelere de like og odde sampler med en primær synkroniseringskode for kommunikasjonssignalet, og for å generere komplekse verdier av de like og odde sampler;
 - en absoluttverdimodifikator (23, 24), AVM, for å approksimere signalstyrkeverdier for hver av de like og odde sampler;
 - 15 - en beslutningskrets (25) for å akkumulere de approksimerte signalstyrkeverdier for samplene i en ramme for kommunikasjonssignalet, som tilordner en indeksverdi til den høyeste akkumulerte signalstyrkeverdi, og for bestemmelse av en chip-offsetverdi basert på nevnte indeksverdi;
 - en normaliseringskrets (26) for å normalisere den akkumulerte

20 signalstyrkeverdien;
 - en oppslagstabell (27) for å bestemme en forsterkningsfaktor basert på antall akkumulerte rammer;
 - en multiplikator (28) for å generere en støyterskel basert på den normaliserte signalstyrkeverdien og på forsterkningsfaktoren; og
 - 25 - en komparator (29) for å sammenlikne støyterskelen med den høyeste akkumulerte signalstyrkeverdien for å gjøre en første beslutning;
 - en andre modul (14) for å generere et forvrengningskodegruppetall, et vindusavvik, og andre synkroniseringskode basert på utgang tilveiebrakt av den første modul;
 - en tredje modul (16) for å innhente en primær forvrengningskode basert på

30 forvrengningskodegruppetallet og vindusavviket, og
 - en kontroller (18) koblet til den første modul (12), andre modul (14) og tredje modul (16) for å kontrollere en søkefrekvens for UE for å etablere kommunikasjonslinken.
2. UE i samsvar med krav 1, hvor den andre modul (14) omfatter:
 - 35 - en forsinkelsesmodul (32) for å forsinke chip-offseten bestemt i den første modulen inntil den første modulen tar den første beslutningen;
 - en sampelvelger (34) for å ekstrahere en spiss-HGC-utgang;

- en konjugerer (36) for å konjugere den ekstraherte spiss-HGC-utgang;
 - en omhylningsfjerner (31) for å fjerne en omhylningssekvens fra det mottatte kommunikasjonssignalet;
 - en Fast Hadamard Transformer (33), FHT, for transformering av en Hadamard-
5 matrise for kommunikasjonssignalet;
 - en kompleks multiplikator (38) for fasekorrigering av den transformerte Hadamard-matrisen ved bruk av den konjugerte spiss-HGC-utgangen;
 - en inngangsmatrisegenerator (35) for generering av en inngangsmatrise som omfatter realdeler av Hadamard-matriseutganger, av multiplikatoren;
 - 10 - en beslutningskrets (44) for å generere en korrelasjonsmatrise som omfatter forvrengningskode-gruppenummer og luke-offset;
 - en komparator (30) for sammenlikning av et maksimumsoppslag i korrelasjonsmatrisen med en terskel for avgivelse av en beslutning.
- 15 3. UE i samsvar med krav 1, hvor den tredje modul (16) omfatter:
- et flertall forvrengningskodegeneratorer ($40_1, \dots, 40_8$), hver for innhenting av kodegruppenummeret generert av den andre modulen og for generering av en forvrengningskode fra dette;
 - et flertall av korrelatorkretser ($41_1 \dots 41_8$) for å generere og akkumulere
20 korrelasjoner av datasymboler for kommunikasjonssignalet ved bruk av utgang fra forvrengningskodegeneratorene ($40_1, \dots, 40_8$);
 - en støyestimator (42);
 - en beslutningskrets (44) for å generere den primære forvrengningskode basert på utgang fra korrelatorkretrene ($41_1, \dots, 41_8$);
 - 25 - en beslutningsstøttekrets (45) for å avgi den primære forvrengningskode til kontrolleren (18);
 - en multiplikator (48) for å beregne en terskelverdi ved å multiplisere støyestimatutganger fra støyestimatoren (42) med en forsterkningsverdi; og
 - en komparator for å sammenlikne den beregnede terskelverdi til en maksimal
30 korrelasjonsverdi og for å generere en beslutning fra dette.
4. Fremgangsmåte for å generere en indeksverdi assosiert med en primærsynkroniseringskode i et kommunikasjonssignal, idet fremgangsmåten omfatter:
- å splitte (402) et samplet kommunikasjonssignal til like og odde sampler;
 - 35 - å korrelere (403) de like og odde sampler med en primær synkroniseringskode for kommunikasjonssignalet ved bruk av en hierarkisk Golay-korrelator, HGC,
 - å generere komplekse verdier av de like og odde sampler;

- å approksimere (406) signalstyrkeverdier for hver av de like og odde sampler;
- å akkumulere (407, 408) de approksimerte signalstyrkeverdier for samplene i en ramme for kommunikasjonssignalet;
- å tilordne (410) en indeksverdi til den høyeste akkumulerte signalstyrkeverdi;
- 5 - å normalisere den akkumulerte signalstyrkeverdien;
- å bestemme en forsterkningsfaktor basert på antall akkumulerte rammer;
- å generere en støyterskel ved å multiplisere den normaliserte signalstyrkeverdien og forsterkningsfaktoren;
- å sammenlikne (414) støyterskelen med den høyeste akkumulerte signalstyrkeverdien
- 10 for å gjøre en første beslutning.

1/16

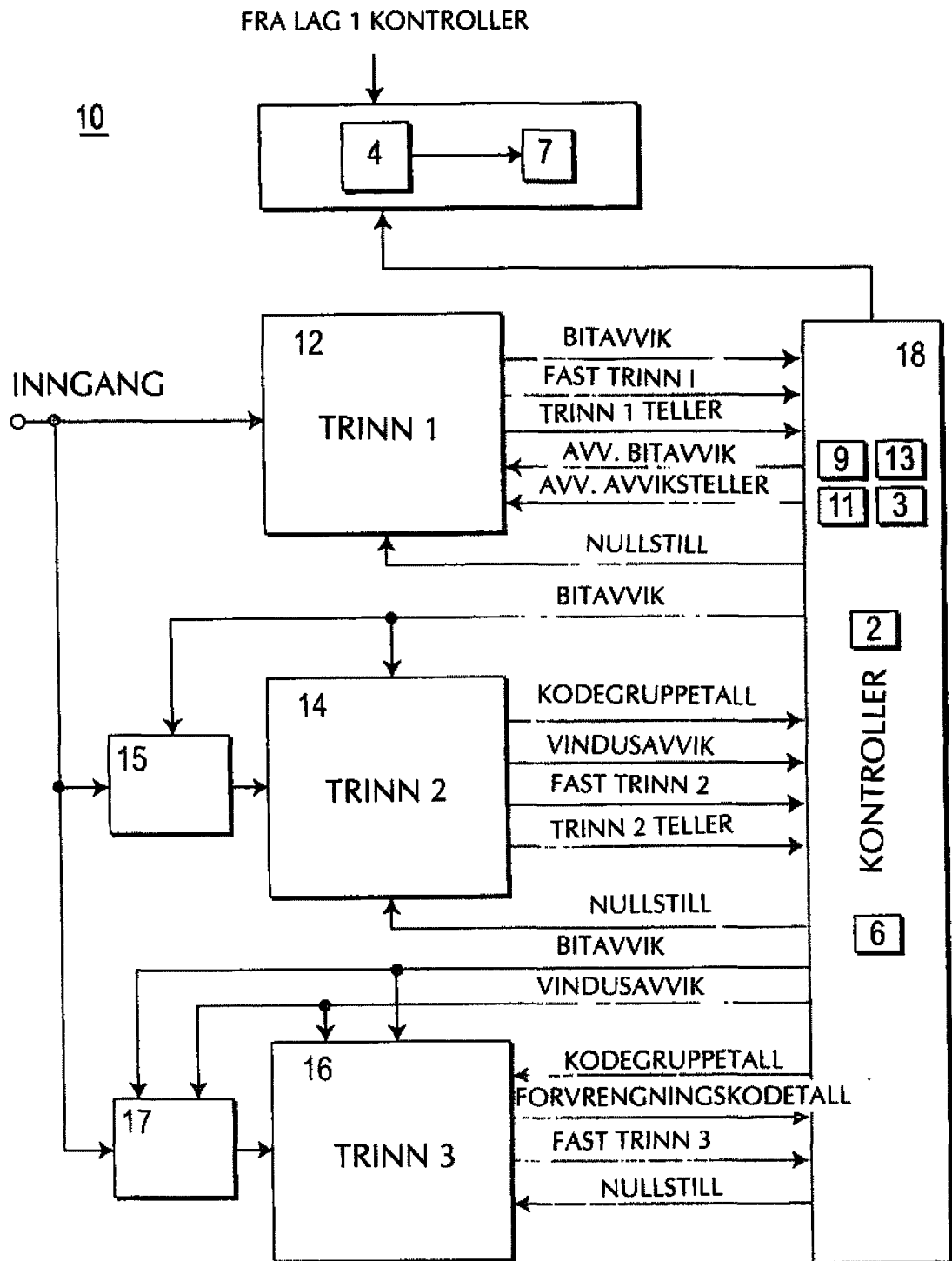
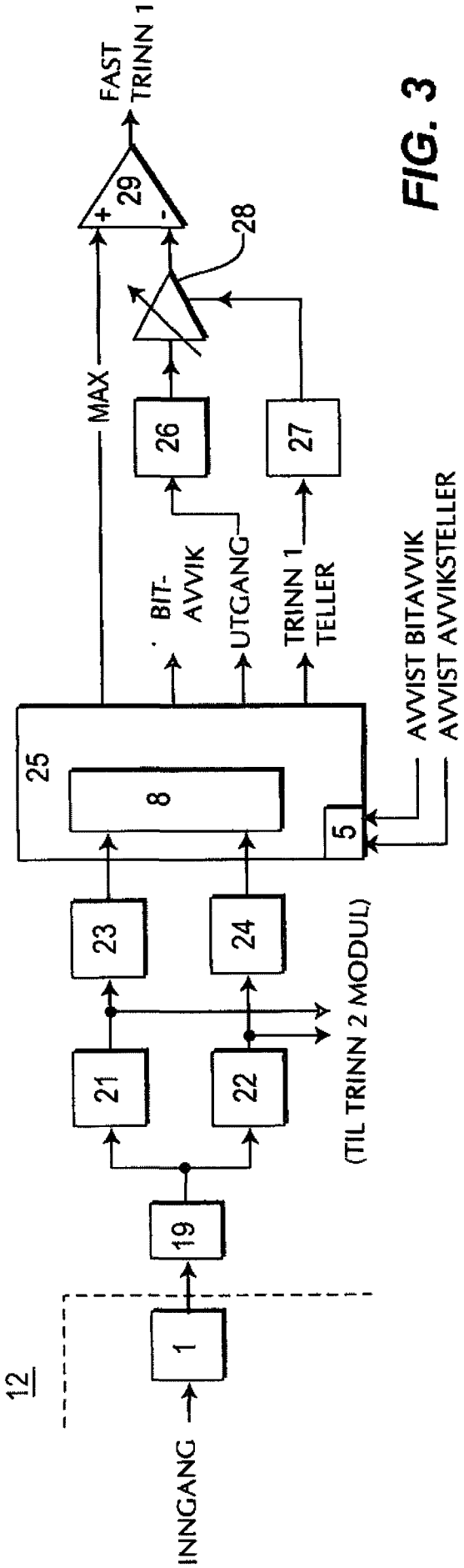
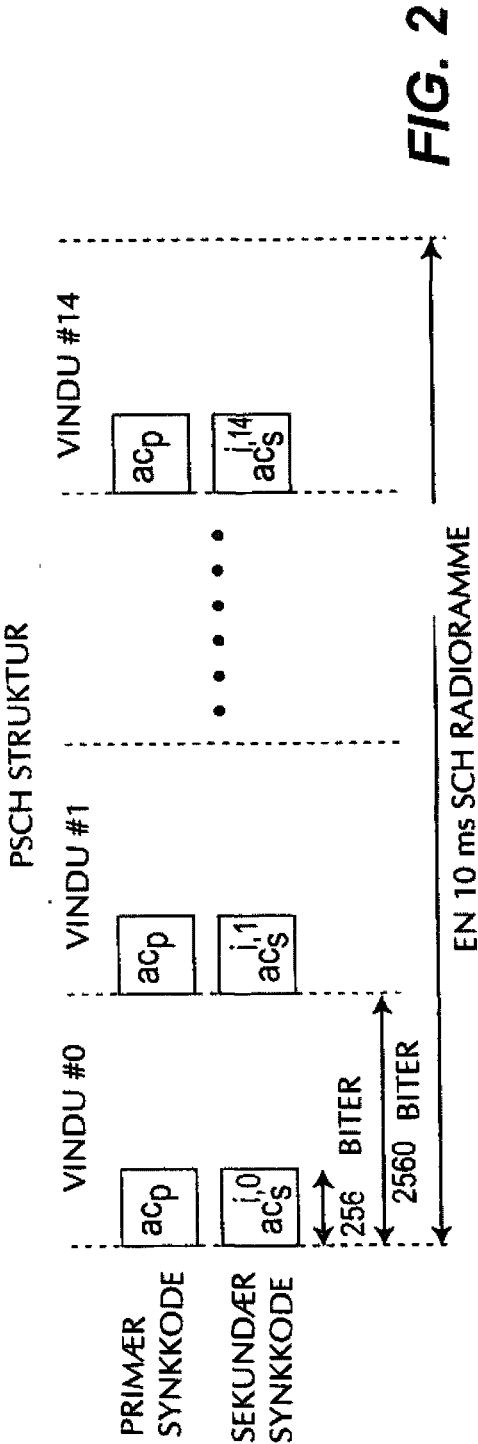


FIG. 1



3/16

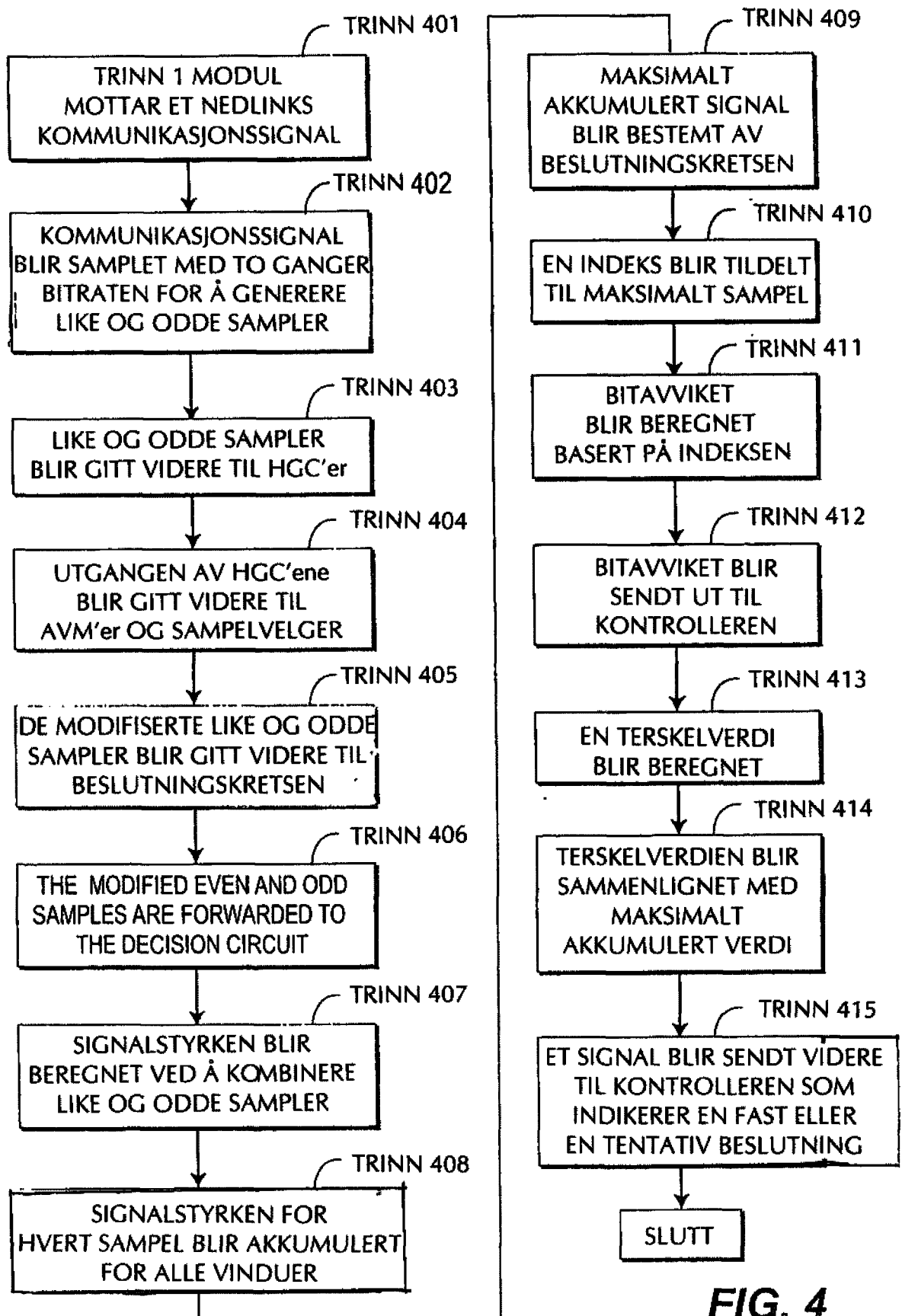


FIG. 4

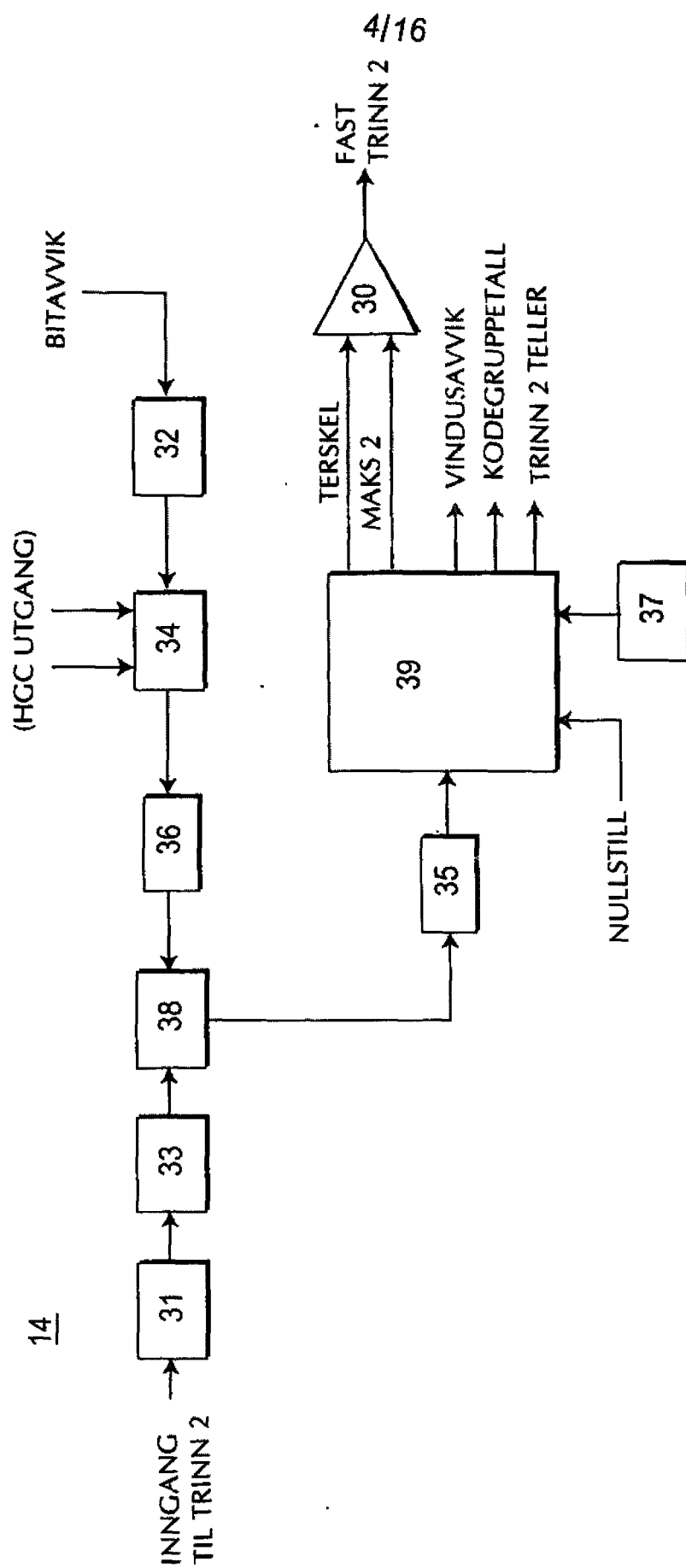
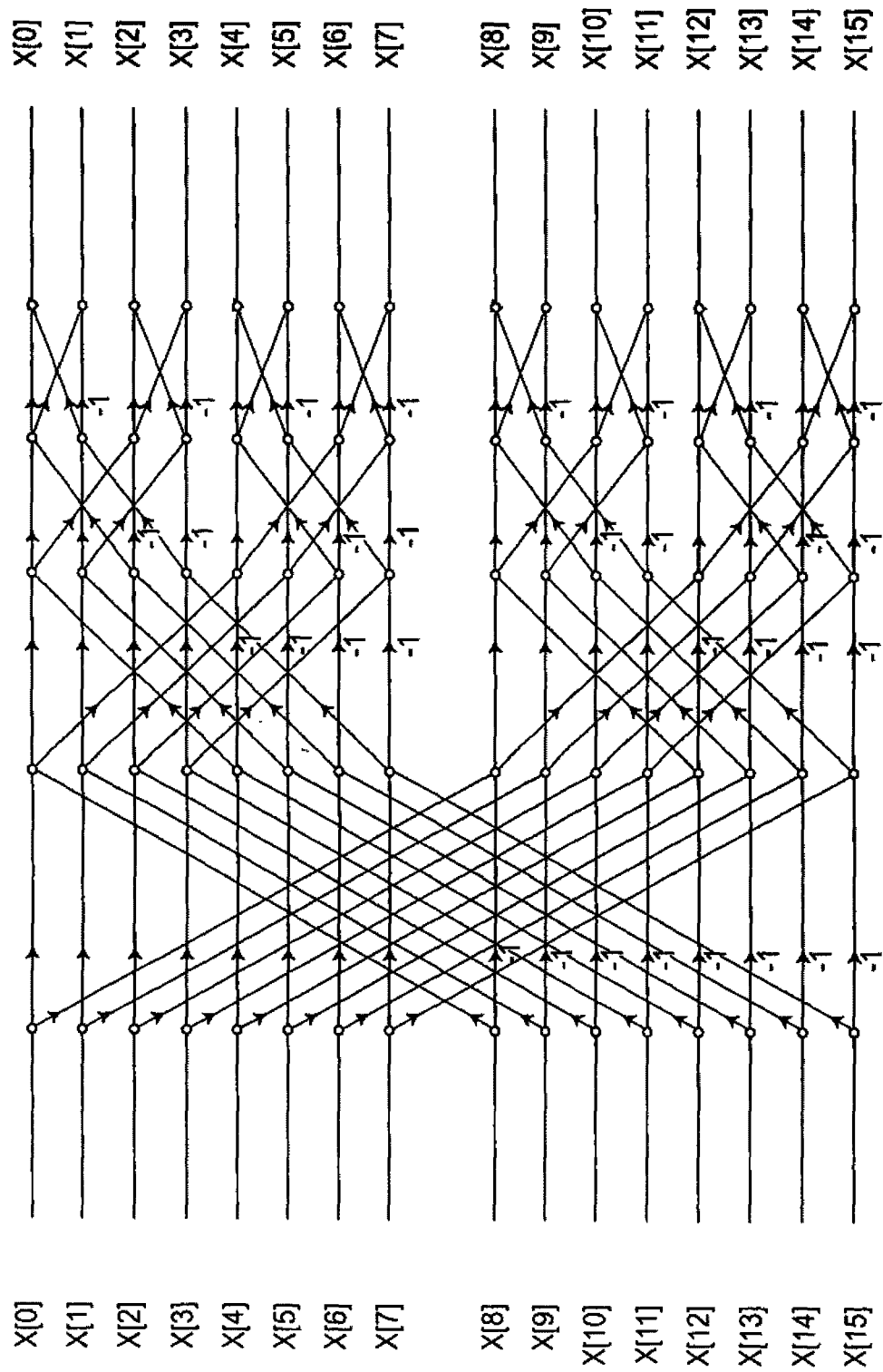


FIG. 5

5/16



HURTIG HADAMARD TRANSFORMERINGSSTRUKTUR

FIG. 6

6/16

$$\text{INNGANGSMATRISSE} = \begin{matrix} & \text{ELEMENTER I FHT UTGANG} \\ \begin{bmatrix} x_{1,1} & x_{1,2} & \cdot & \cdot & \cdot & x_{1,16} \\ x_{2,1} & x_{2,2} & \cdot & \cdot & \cdot & x_{2,16} \\ \cdot & \cdot & x_{i,j} & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ x_{15,1} & x_{15,2} & \cdot & \cdot & \cdot & x_{15,16} \end{bmatrix} \end{matrix} \quad 15 \times 16$$

$i = 1, 2, 3, \dots, 15$ VINDUER

$j = 1, 2, 3, \dots, 16$ ELEMENTER PR. VINDU FRA HGC UTGANGEN

FIG. 7

$$\text{KODE-GRUPPEMATRISSE} = \begin{matrix} \text{(KJENT MATRISSE)} \\ \begin{bmatrix} 1 & 1 & 2 & 8 & \cdot & \cdot & \cdot & 15 & 7 & 16 \\ 1 & 1 & 5 & 16 & \cdot & \cdot & \cdot & 14 & 12 & 10 \\ \cdot & \cdot & \cdot & \cdot & x_{i,j} & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 9 & 12 & 10 & 15 & \cdot & \cdot & \cdot & 12 & 16 & 10 \end{bmatrix} \end{matrix} \quad 64 \times 15$$

$i = 1, 2, 3, \dots, 64$ GRUPPER

$j = 1, 2, 3, \dots, 15$ VINDUER

$x_{i,j}$ = S-SCH KODER. EN UT AV 16 PR. VINDU

FIG. 8

$$\text{KORRELASJONSMATRISSE} = \begin{matrix} \begin{bmatrix} c_{0,0} & c_{0,1} & \cdot & \cdot & c_{0,14} \\ c_{1,0} & c_{1,1} & \cdot & \cdot & c_{1,14} \\ \cdot & \cdot & c_{i,j} & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ c_{63,0} & c_{63,1} & \cdot & \cdot & c_{63,14} \end{bmatrix} \end{matrix} \quad \text{PR. RAMME} \quad 64 \times 15$$

$c_{i,j} = \text{INNGANGSMATRISSE} [k] \text{ KODEGRUPPEMATRISSE} [i] [j] [k]$
K

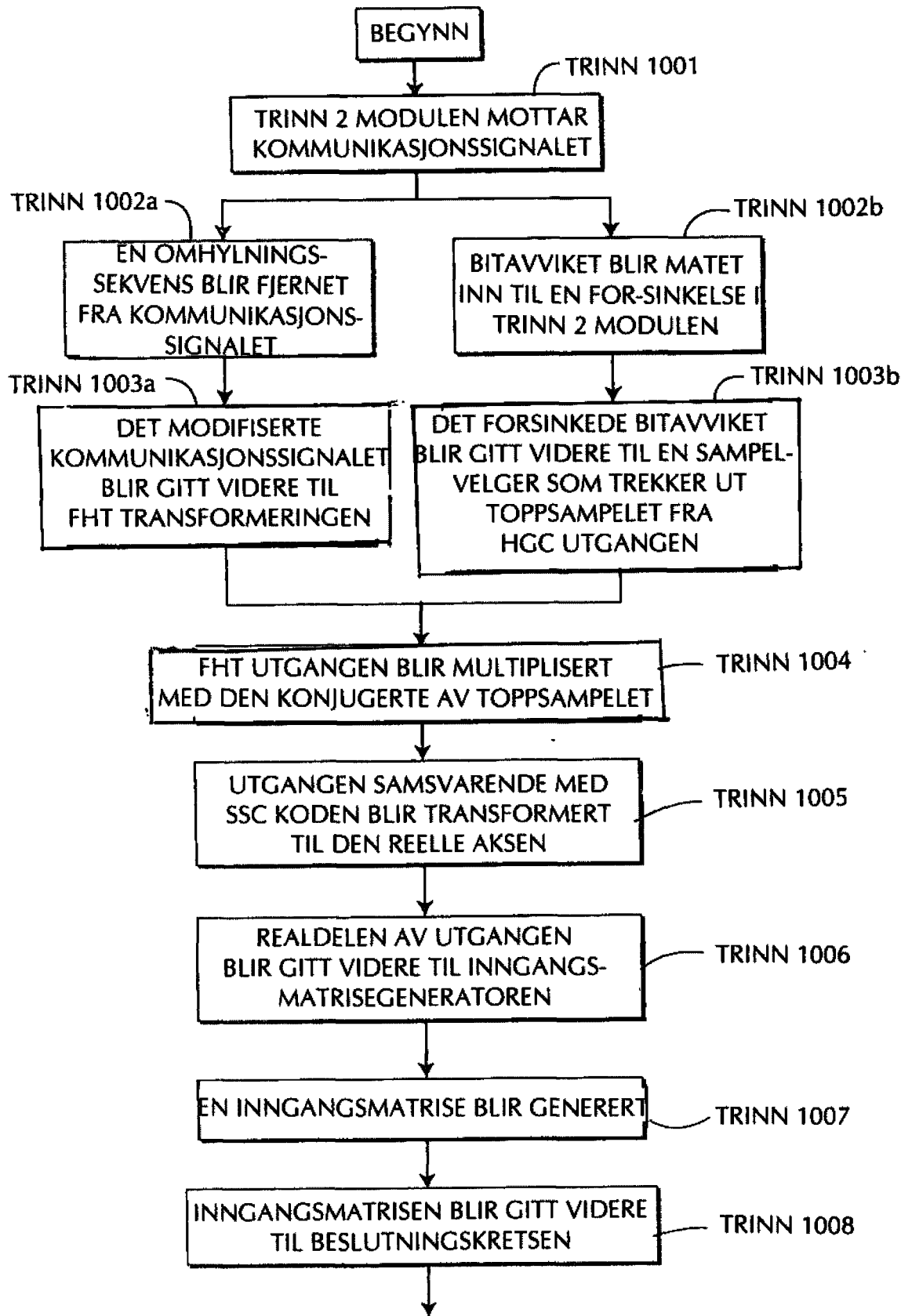
$i = 0, 1, 2, \dots, 63$ GRUPPER

$j = 0, 1, 2, \dots, 14$ SYK LISKE SKIFT MED HENSYN TIL KOLONNER (VINDUSAVVIK)

$k = 0, 1, 2, \dots, 14$ VINDUER PR. RAMME

FIG. 9

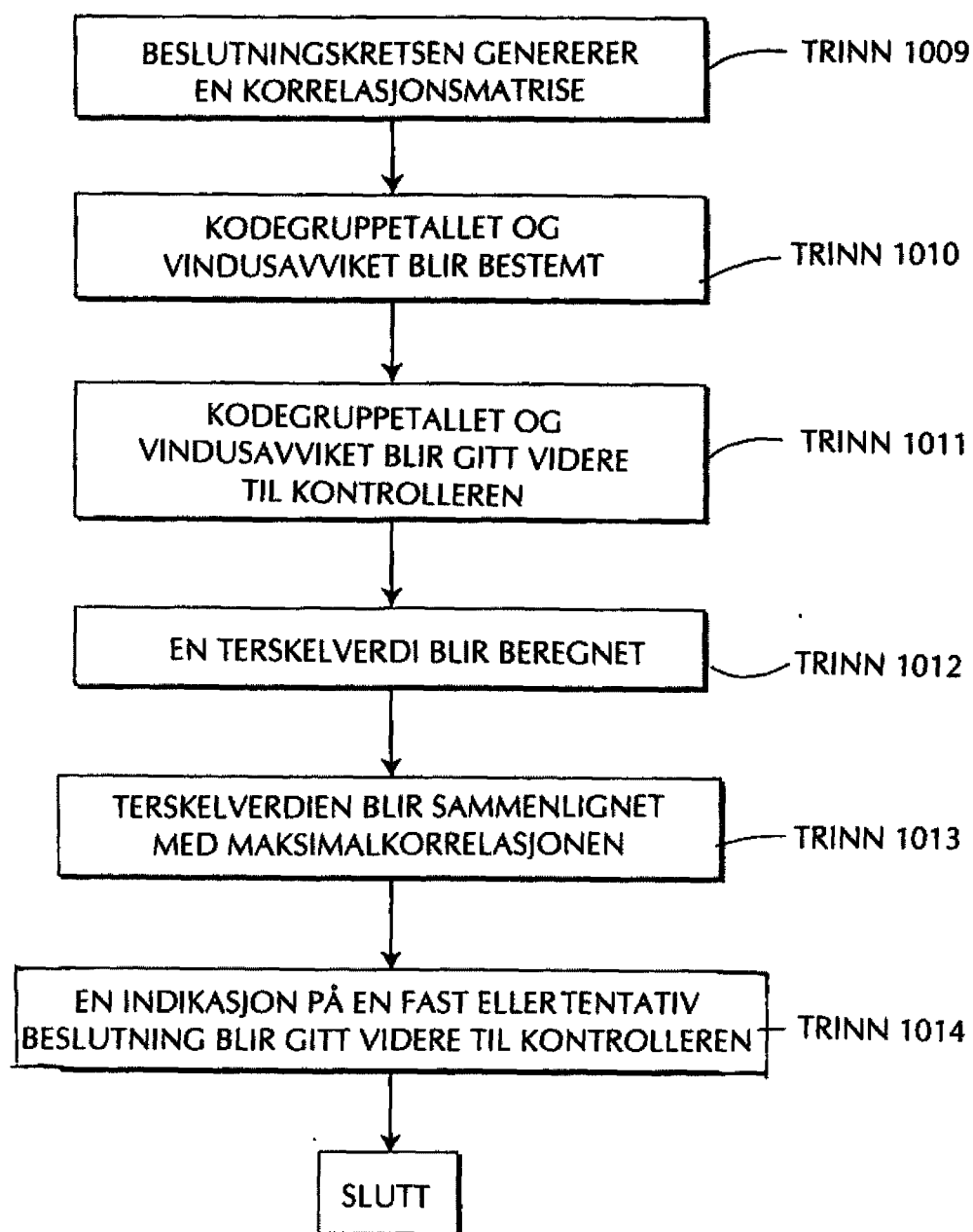
7/16



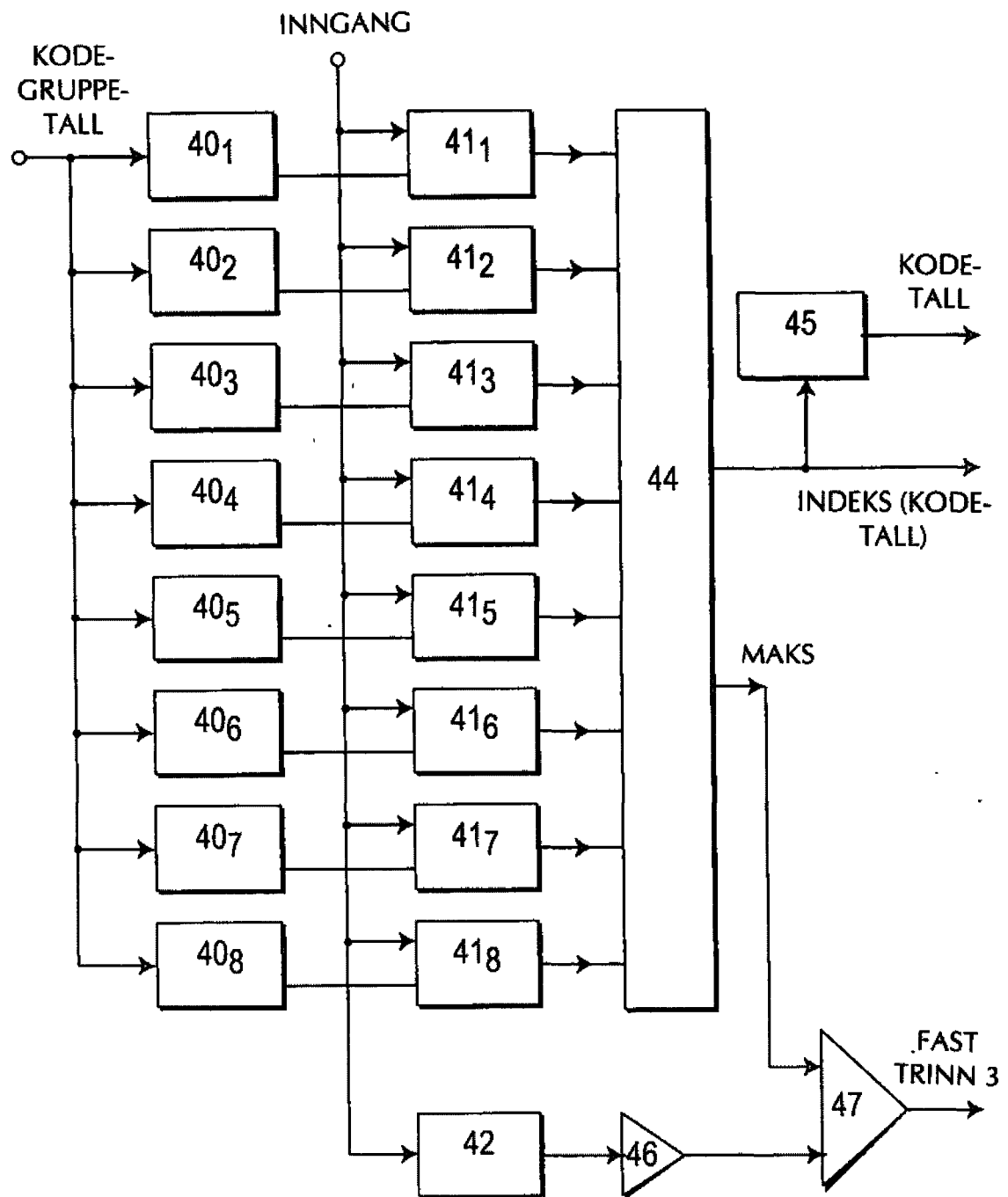
(FORTSETTER I FIGUR 10B)

FIG. 10A

8/16

**FIG. 10B**

9/16

**FIG. 11**

10/16

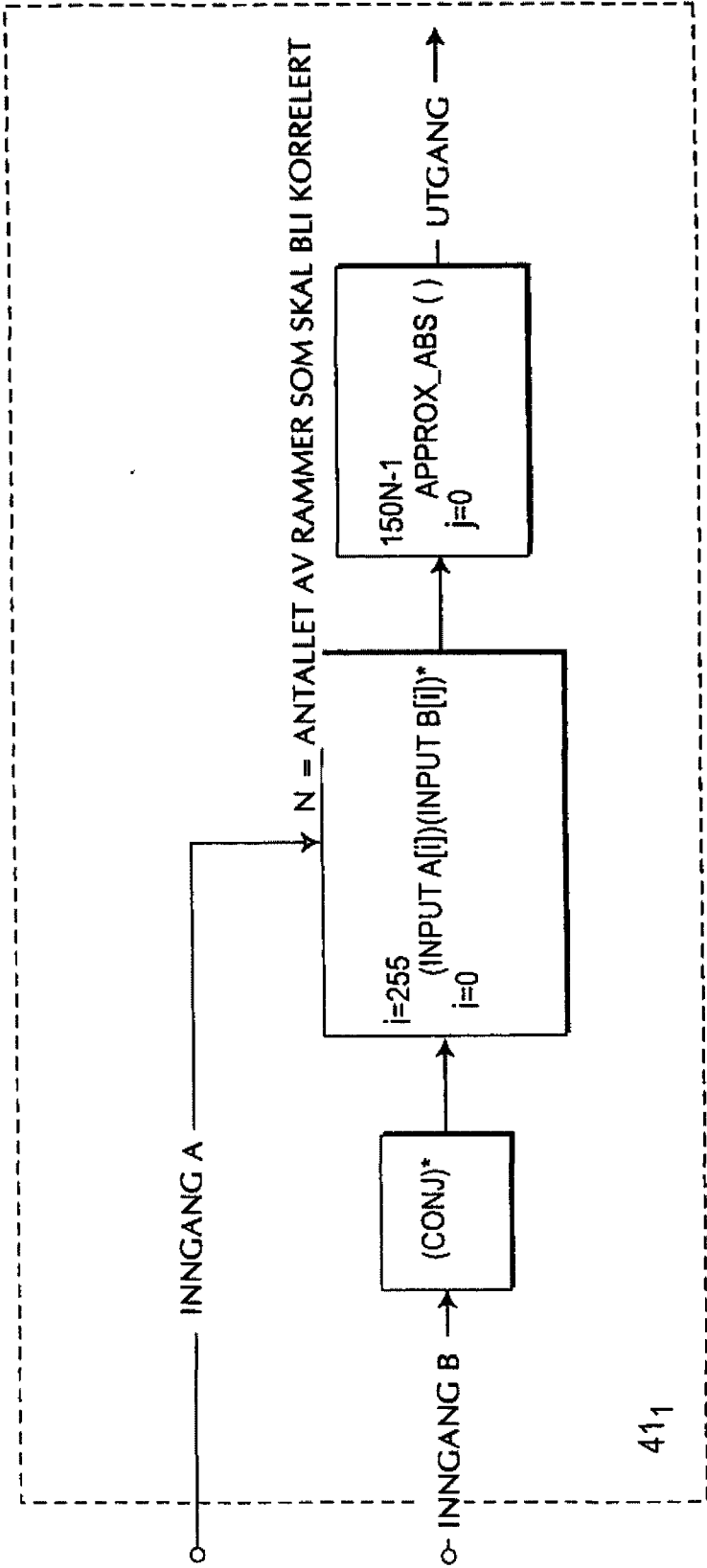


FIG. 12

11/16

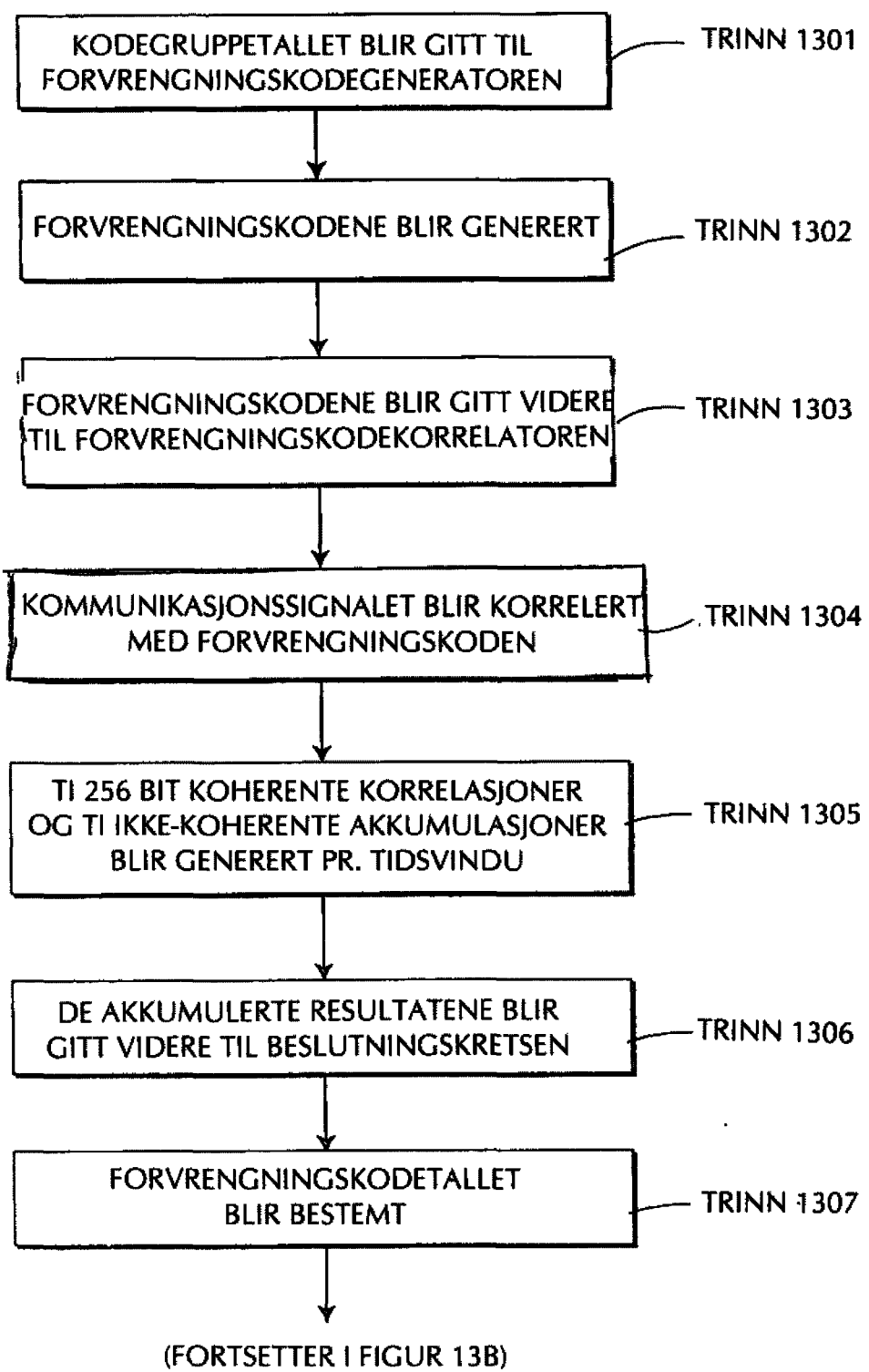


FIG. 13A

12/16

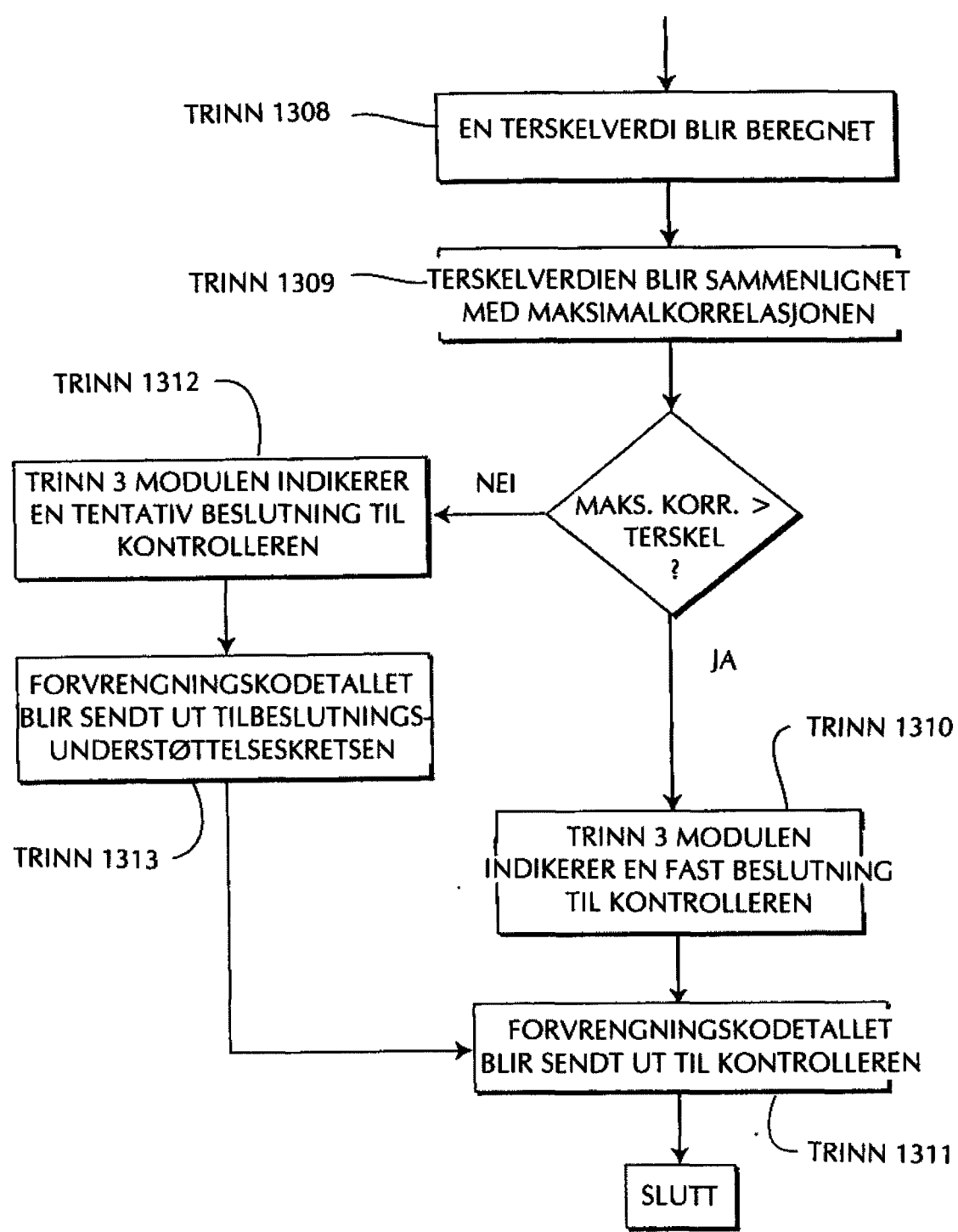


FIG. 13B

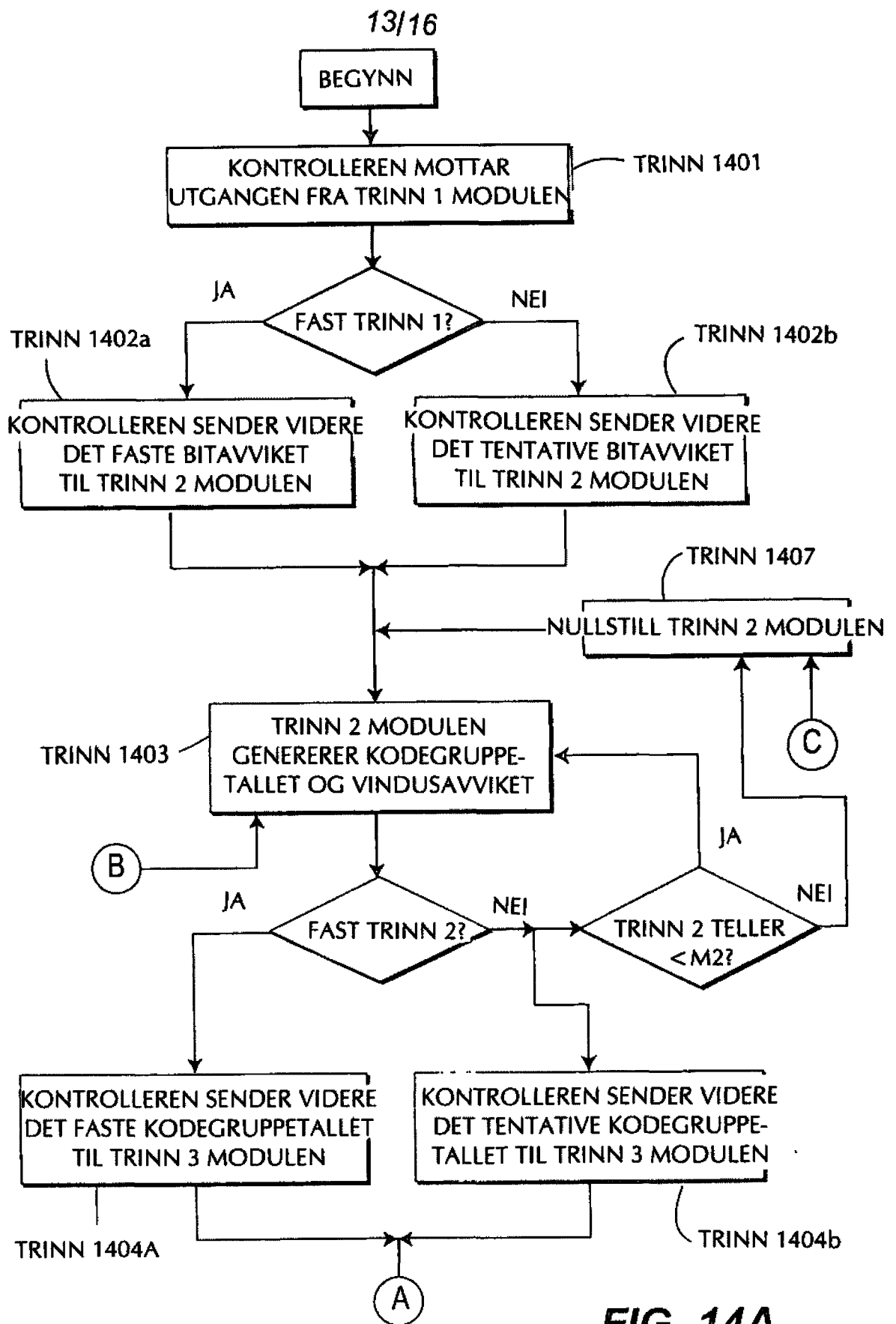
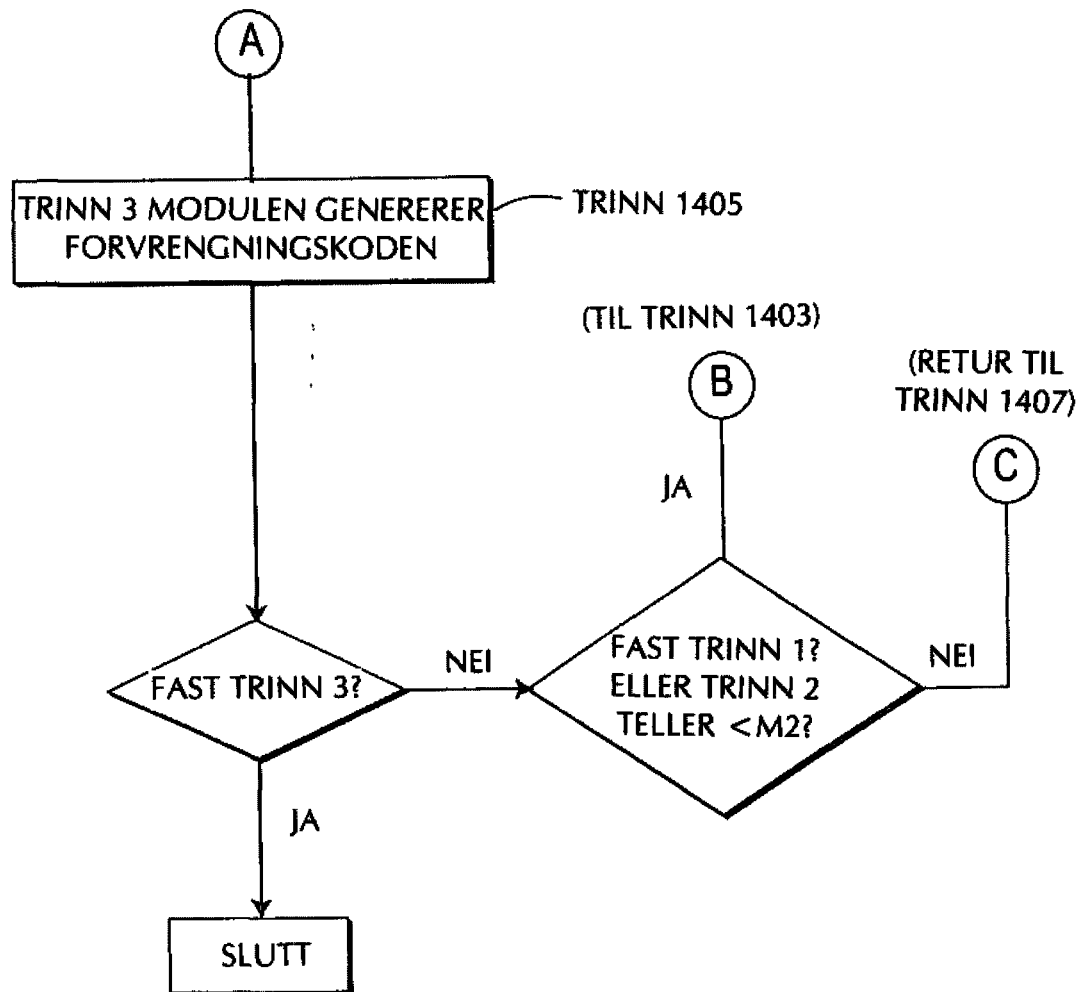


FIG. 14A

14/16

**FIG. 14B**

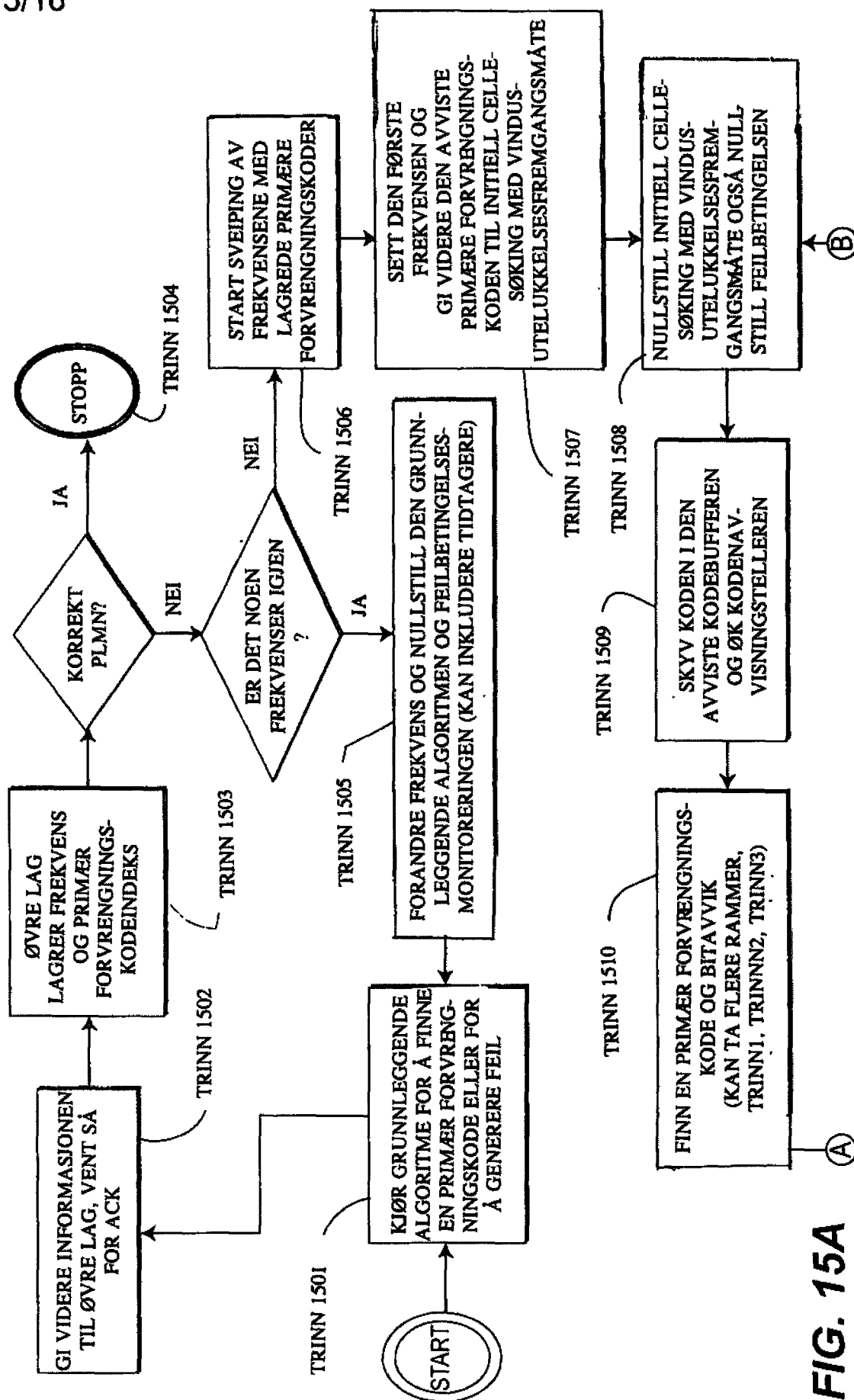


FIG. 15A

