(12) **PATENT**

(19) NO (11) **337611** (13) **B1**

NORGE (51) Int Cl.

H04L 9/06 (2006.01)
H04L 9/08 (2006.01)
G09C 1/00 (2006.01)
H04K 1/00 (2006.01)
H04L 9/00 (2006.01)

Patentstyret

(21)	Søknadsnr	20045596	(86)	Int.inng.dag og søknadsnr	2003.03.07 PCT/JP2003/02689
(22)	Inng.dag	2004.12.22	(85)	Videreføringsdag	2004.12.22
(24)	Løpedag	2003.03.07	(30)	Prioritet	2002.05.23, JP, 2002-148786
(41)	Alm.tilgj	2005.02.18			
(45)	Meddelt	2016.05.09			

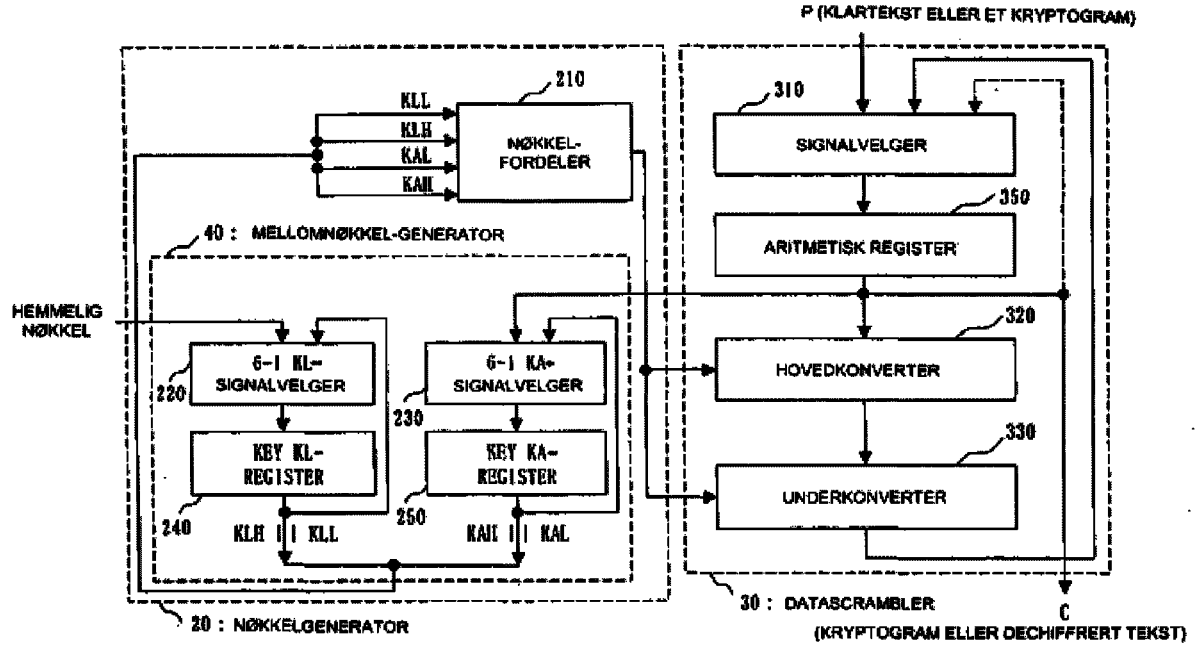
(73) Innehaver Mitsubishi Denki KK, 7-3, Marunouchi 2-chome, Chiyoda-ku, JP-100-8310 TOKYO, Japan
(72) Oppfinner Tomomi Kasuya, c/o Mitsubishi Denki KK, 2-3, Marunouchi 2-chome, Chiyoda-ku, JP-100-8310 TOKYO , Japan
Mitsuru Matsui, c/o Mitsubishi Denki KK, 2-3, Marunouchi 2-chome, Chiyoda-ku, JP-100-8310 TOKYO, Japan
Tetsuya Ichikawa, c/o Mitsubishi Electric Engineering Co Ltd, 2-6-2, Ohtemachi, JP-100-0004 TOKYO, Japan
(74) Fullmektig Bryn Aarflot AS, Postboks 449 Sentrum, 0104 OSLO, Norge

(54) Benevnelse **Fremgangsmåte og anordning for omforming av data**

(56) Anførte publikasjoner
AKASHI SATOH ET AL.:"A compact Rijndael hardware architecture with S-Box optimization", Springer, DE, 1 Jan. 2001, pages 239 - 254, ISBN: 978-3-540-24128-7.
AOKI ET AL.:"The 128-bit block cipher camellia", IEICE Transactions on fundamentals of electronics communications and computer sciences, Engineering sciences society, Tokyo, JP, vol. E85-A, No.1, 1 Jan. 2002, p. 11 - 24.

(57) Sammendrag

En underkonverter (330) tilveiebrakt i en dataomformingsanordning for kryptering/dekryptering av data har en dataomformingsfunksjon og en dataoverføringsfunksjon eller nøkkeloverføringsfunksjon, og omformer data og overfører data som er ikke-lineært omformet i en hovedkonverter (320) eller en nøkkel som er matet ut fra et key KL-register (240) ved å skifte mellom dataomformingsfunksjonen og data- eller nøkkeloverføringsfunksjonen



Foreliggende oppfinnelse vedrører en anordning og en fremgangsmåte for omforming av data i forbindelse med kryptering og/eller dekryptering.

En beskrivelse vil først bli gitt av beslektet teknikk.

Akashi Satoh m.fl.: "A compact Rijndael hardware architecture with S-Box optimization", Springer, DE, 1. januar 2001, sider 239-254, ISBN: 978-3-540-24128-7, beskriver kompakte og høyhastighets-maskinvarearkitekturer og logikkoptimaliseringsmetoder for Rijndael AES-algoritme, hvor krypterings- og dekrypteringsdatabaner kombineres, og alle aritmetiske komponenter gjenbrukes.

Aoki m.fl.: "The 128-bit block cipher camellia", IEICE Transactions on fundamentals of electronics communications and computer sciences, Engineering sciences society, Tokyo, JP, vol. E85-A, nr. 1, 1. januar 2002, sider 11-24, beskriver et 128-bits blokkchiffer (blokkryptogram) med navnet Camellia og som støtter 128-bits blokkstørrelse og 128-, 192- og 256-bits nøkler, dvs. de samme grensesnittspesifikasjoner som i den avanserte krypteringsstandarden (Advanced Encryption Standard / AES). Camellia er ytterligere beskrevet videre ned i beskrivelsen.

Figur 56 er et diagram som illustrerer oppbygningen og virkemåten til en beslektet dataomformingsanordning.

Som kan sees i figur 56, består dataomformingsanordningen for blokkryptering av en nøkkelgenerator 20 og en datascrambler 30.

Nøkkelgeneratoren 20 er en nøkkelgenereringsenhet som genererer en nøkkel for kryptering/dekryptering av data.

Datascrambleren 30 er en enhet som krypterer og dekrypterer innmatede data.

Nøkkelgeneratoren 20 omfatter en mellomnøkkel-generator 40 og en nøkkelfordeler 210. Mellomnøkkel-generatoren 40 mottar en hemmelig nøkkel og genererer en mellomnøkkel (key KL) og en utnøkkel (key KA) basert på den mottatte hemmelige nøkkelen. Nøkkelfordeleren 210 mottar mellomnøklene (key KL) og utnøklene (key KA) som genereres i mellomnøkkel-generatoren 40 (key KLL, key KLH, key KAL og key KAH), og skedulerer én av de innmatede nøklene for innmating til datascrambleren 30. I nøkkelgeneratoren 20 blir således nøkler generert og skedulert henholdsvis i mellomnøkkel-generatoren 40 og nøkkelfordeleren 210.

Datascrambleren 30, ved mottak av data P (klartekst), besørger omforming av dataene for kryptering og mater deretter ut omformede data som C (et kryptogram). Ved mottak av data P (et kryptogram) på den annen side, besørger dataskrambleren 30 omforming av dataene for dekryptering, og mater deretter ut data som C (dechiffrert tekst). Datascrambleren 30 utfører således prosesser for kryptering og dekryptering av data.

I datascrambleren 30 er en hovedkonverter 320 og en underkonverter 330 koplet i serie.

Hovedkonverteren 320 er en enhet som utfører ikke-lineær omforming. Mer spesifikt er hovedkonverteren 320 utstyrt med en F-funksjon som utfører ikke-lineær dataomforming for én runde eller flere runder, eller en del av F-funksjonen, og besørger ikke-lineær omforming av data med bruk av F-funksjonen eller delen av F-funksjonen. Figur 57 illustrerer en hovedkonverter 320 som er utstyrt med F-funksjonen for én eller flere runder.

Underkonverteren 330 omfatter minst én av en datakonverterenhet (FL) som utfører en lineær omforming av data, og en datainverterenhet (FL^{-1}) som utfører en omforming som er den inverse av den omformingen som utføres av datakonverterenheten (FL), og besørger lineær omforming av innmatede data med bruk av en innmatet nøkkel ved hjelp av datakonverterenheten (FL) eller datainverterenheten (FL^{-1}).

Signalvelgeren 310 velger ut ett av innsignalene omfattende det fra hovedkonverteren 320, det fra underkonverteren 330, P (klartekst eller et kryptogram) og en nøkkel. Signalvelgeren 310 illustrert i figur 56 omfatter en signalvelger som velger ut ett signal av fire innsignaler, som er ekvivalent med tre 2-1-signalvelgere som hver mater ut ett utsignal av to innsignaler.

Det aritmetiske registeret 350 er et minnelager som holder data som mates ut som hovedkonverteren 320, underkonverteren 330 og C (klartekst eller et kryptogram) for en forbestemt tidsperiode.

Datascrambleren 30 krypterer/dekrypterer således de innmatede dataene P (klartekst eller et kryptogram) gjennom flere repetisjoner av vekselvis ikke-lineær omforming i hovedkonverteren 330 og lineær omforming i underkonverteren 320, og mater deretter ut C (et kryptogram eller dechiffrert tekst).

En beskrivelse vil nå bli gitt av den innvendige oppbygningen til hovedkonverteren 320.

Figur 57 viser den innvendige oppbygningen til hovedkonverteren 320. Hovedkonverteren 320 i figur 57 omfatter seks F-funksjonsenheter. Antatt her at F-funksjonsenhetene er konstruert med en krets som er beregnet for en én-rundes F-funksjonsprosess, skal hovedkonverteren 320 i figur 57 da utføre den F-funksjonsbaserte, ikke-lineære dataomformingen for seks runder.

Når det gjelder kretsen for den seks-runders F-funksjonsprosessen, kan hovedkonverteren 320 være utstyrt med seks F-funksjonskretser eller eventuelt én enkelt F-funksjonskrets som gjentar F-funksjonsprosessen seks ganger for å gjennomføre nevnte seks-runders F-funksjonsbaserte databehandling.

I hovedkonverteren 320 blir først øvre data tatt fra innmatede data matet inn til en F-funksjonsenhet 321a. En nøkkel key 1 som er skedulert av nøkkel-fordeleren 210 blir også matet inn til denne. I F-funksjonsenheten 321a blir de øvre inndataene ikke-lineært omformet med bruk av nøkkelen som nevnt over. I en XOR-krets 322a blir de ikke-lineært omformede dataene XOR-behandlet med den nedre del av inndataene. Data som mates ut fra XOR-kretsen 322a blir matet inn til en F-funksjonsenhet 321b. F-funksjonsenheten 321b, i likhet med F-funksjonsenheten 321a, utfører ikke-lineær omforming, og de om-formede dataene blir deretter XOR-behandlet med øvre del av inndataene i en XOR-krets 322b. Data som mates ut fra XOR-kretsen 322b blir matet inn til en F-funksjonsenhet 321c. På denne måten blir samme prosess som den som ut-føres av F-funksjonsenheten 321a og XOR-kretsen 322a utført henholdsvis av F-funksjonsenheten 321b og XOR-kretsen 322b, av F-funksjonsenheten 321c og en XOR-krets 322c, av en F-funksjonsenhet 321d og en XOR-krets 322d, av en F-funksjonsenhet 321e og en XOR-krets 322e og av en F-funksjonsenhet 321f og en XOR-krets 322f. På denne måten utføres således den seks-runders F-funksjonsbaserte ikke-lineære omformingen (eller den én-rundes F-funksjons-baserte ikke-lineære omformingen blir gjentatt seks ganger), og deretter blir omformede data matet ut.

Oppbygningen av den ovenfor nevnte prosessen for ikke-lineær om-forming kalles en Feistel-struktur, som kjennetegnes ved at de øvre dataene og de nedre dataene byttes om og mates ut ved å motta én av de øvre dataene atskilt og de nedre dataene atskilt, ikke-lineært omforme de mottatte dataene, mate ut den ene av de øvre dataene og de nedre dataene omformet, XOR-behandle den ene av de øvre dataene og de nedre dataene som ble matet ut

med de andre av de øvre dataene og de nedre dataene, bytte om de XOR-behandlede dataene og de andre av de øvre dataene og de nedre dataene som ikke ble matet inn til F-funksjonsenheten, og mate ut de nedre dataene og de øvre dataene byttet om.

Typiske strukturer for randomisering av data er Feistel-strukturen og SPN-(Substitution Permutation Network)-strukturen. En hovedkonverter 320 med SPN-strukturen sies å være best egnet for parallellprosessering. En hovedkonverter 320 med Feistel-strukturen sies å være best egnet dersom man ønsker å redusere maskinvarens størrelse.

Merk at SPN-strukturen, i motsetning til Feistel-strukturen der de inn-matede dataene deles opp, er konstruert slik at en F-funksjon som dannes av et S-lag (ikke-lineært lag) og et P-lag (lineært lag) blir gjentatt.

En beskrivelse vil nå bli gitt av den innvendige oppbygningen til under-konverteren 330.

Figur 58 er et diagram som illustrerer kretser som danner under-konverteren 330.

Underkonverteren 330 i figur 58 omfatter en datakonverterenhet 50 og en datainverterenhet 70.

I datakonverterenheten 50 utføres en logisk AND-operasjon mellom de 32 øvre eller mest signifikante bit av 64-bits inndata og en nøkkel key 1 i en AND-krets 54, og resultatet av denne operasjonen blir rotasjonsskiftet én bit mot venstre. Deretter, i en XOR-krets 55, blir inndata XOR-behandlet med de 32 nedre eller minst signifikante bit av de innmatede dataene, og resultatet blir matet ut som de nedre 32 bit av et utsignal og også matet inn til en OR-krets 57. Deretter, i OR-kretsen 57, blir innmatingen logisk OR-behandlet med en nøkkel key 2, og resultatet av denne operasjonen blir XOR-behandlet med de øvre 32 bit av de innmatede dataene i en XOR-krets 56, fra hvilken resultatet blir matet ut som de øvre 32 bit av et utsignal. På denne måten blir de 64-bits innmatede dataene lineært omformet og deretter matet ut som et 64-bits ut-signal.

I datainverterenheten 70 utføres en logisk OR-operasjon mellom de nedre 32 bit av 64-bits inndata og en nøkkel key 3 i en OR-krets 74, og resultatet av denne operasjonen blir XOR-behandlet med de øvre 32 bit av de innmatede dataene i en XOR-krets 75, fra hvilken resultatet blir matet ut som de

øvre 32 bit av et utsignal og også matet inn til en AND-krets 77. I AND-kretsen 77 blir inndata logisk AND-behandlet med en nøkkel key 4, og resultatet blir deretter rotasjonsskiftet én bit mot venstre. Deretter, i en XOR-krets 76, blir en innmating XOR-behandlet med de nedre 32 bit av de innmatede dataene, og resultatet blir matet ut som de nedre 32 bit av et utsignal. På denne måten blir de 64-bits innmatede dataene lineært omformet av datakonverterenheten 50 og datainverterenheten 70, og deretter matet ut som et 64-bits utsignal. Merk at nøklene 1 til 4 er tilveiebragt av nøkkelfordeleren 210.

Figur 59 er et diagram som illustrerer en krets som deles av datakonverterenheten 50 og datainverterenheten 70 som et eksempel på underkonverteren 330.

I figur 59, ved innmating av et skiftesignal for å skifte mellom datakonverterenheten 50 og datainverterenheten 70, skiftes datakonverterenheten 50 og datainverterenheten 70. Mer spesifikt skifter i den delte kretsen i figur 59 en 2-1-signalvelger 99a, ved mottak av skiftesignalet, mellom et innsignal A og et innsignal E, og en 2-1-signalvelger 99b skifter mellom et innsignal C og et innsignal F.

Tilfellet der den delte kretsen fungerer som datakonverterenhet 50 vil bli beskrevet først.

2-1-signalvelgeren 99a velger innsignalet A fra innsignalene E og A, og mater ut signalet som et utsignal B. Deretter, i en AND-krets 101, blir en innmating logisk AND-behandlet med en nøkkel key 1, og resultatet blir rotasjonsskiftet én bit mot venstre. Deretter, i en XOR-krets 91, blir en innmating XOR-behandlet med de nedre 32 bit av de innmatede dataene, og resultatet blir matet ut som de nedre 32 bit av et utsignal og også matet inn 2-1-signalvelgeren 99b som innsignalet C. 2-1-signalvelgeren 99b velger innsignalet C fra innsignalene C og F, og mater ut signalet C som et utsignal D. Deretter, i en OR-krets 92, utføres en logisk OR-operasjon mellom utsignalet D og en nøkkel key 2, og resultatet blir deretter XOR-behandlet med de øvre 32 bit av de innmatede dataene i en XOR-krets 93, fra hvilken resultatet blir matet ut som de øvre 32 bit av et utsignal.

En beskrivelse vil nå bli gitt av tilfellet der den delte kretsen fungerer som datainverterenhet 70.

2-1-signalvelgeren 99a velger innsignalet F fra innsignalene C og F, og mater ut innsignalet F som et utsignal D. Deretter utfører OR-kretsen 92 en logisk OR-operasjon mellom utsignalet D og nøkkelen key 2, og resultatet av denne operasjonen blir XOR-behandlet med de øvre 32 bit av de innmatede dataene i XOR-kretsen 93, fra hvilken resultatet blir matet ut som de øvre 32 bit av et utsignal og også matet inn til en 2-1-signalvelgeren 99a som innsignalet E. 2-1-signalvelgeren 99b velger innsignalet E fra innsignalene A og E og mater ut innsignalet E som utsignalet B. Deretter, i AND-kretsen 101, utføres en logisk AND-operasjon mellom utsignalet B og nøkkelen key 1, og resultatet av denne operasjonen blir deretter rotasjonsskiftet én bit mot venstre. Dette rotasjonsskiftede resultatet blir da XOR-behandlet med de nedre 32 bit av de innmatede dataene i XOR-kretsen 91, og resultatet blir matet ut som de nedre 32 bit av et utsignal.

Figur 60 er et diagram som illustrerer en dataomformingsanordning der hovedkonverteren, i motsetning til dataomformingsanordningen i figur 56, er utstyrt med en $1/2^x$ ($x \geq 1$) F-funksjon som er konstruert for å prosessere F-funksjonen for mindre enn én runde.

I tilfellet der hovedkonverteren 320 er utstyrt med en 1/2 F-funksjon, for eksempel, kan det utføres en prosess over to sykluser ved hjelp av signalbanen fra hovedkonverteren 320 gjennom underkonverteren 330, signalvelgeren 310, det aritmetiske registeret 350 og tilbake til hovedkonverteren 320. Dette gjør det mulig å gjennomføre én runde av en F-funksjonsbasert ikke-lineær dataomformingsprosess. For å realisere en slik prosess er det i dataomformingsanordningen i figur 60, i motsetning til i omformeren i figur 56, tilveiebragt en signalbane fra det aritmetiske registeret 350 til signalvelgeren 310.

En beskrivelse vil nå bli gitt av virkemåten til hovedkonverteren 320 som anvender signalbanen fra det aritmetiske registeret 350 til signalvelgeren 310.

Figur 61 illustrerer den innvendige oppbygningen til hovedkonverteren 320.

Som kan sees i figur 61 omfatter hovedkonverteren 320 12 F-funksjonsenheter, som hver prosesserer F-funksjonen for mindre enn én runde, f.eks. halve F-funksjonen (en 1/2F-funksjon). Hovedkonverteren 320 i figur 61 omformer data med bruk av en F-funksjonsenhet 1321a, en F-funksjonsenhet 1321b, en XOR-krets 1322a og en XOR-krets 1322b, mens hovedkonverteren i

figur 57 utfører samme dataomforming med bruk av F-funksjonsenheten 321a og XOR-kretsen 322a.

Med henvisning til hovedkonverteren 320 i figur 61 vil den første runden bli forklart først. Øvre data skilt ut fra de øvre inndataene blir matet inn til F-funksjonsenheten 1321a. En nøkkel key 1H, som utgjøres av de øvre bit av nøkkelen key 1 skedulert av nøkkelfordeleren 210, blir også matet inn til F-funksjonsenheten 1321a. F-funksjonsenheten 1321a utfører ikke-lineær omforming av de øvre dataene med bruk av nøkkelen key 1H. Deretter blir de omformede dataene matet inn til XOR-kretsen 1322a og XOR-behandlet med de øvre dataene utskilt fra de nedre inndataene.

Data som mates ut fra XOR-kretsen 1322a holdes i det aritmetiske registeret 350 som mellomdata inntil en dataprosessering er ferdig i XOR-kretsen 1322b.

Nå vil prosessering for en andre runde bli beskrevet. Fra de øvre inndataene blir de nedre dataene atskilt matet inn til F-funksjonsenheten 1321b. En nøkkel key 1L, som utgjøres av de nedre bit av nøkkelen key 1 skedulert av nøkkelfordeleren 210, blir også matet inn til F-funksjonsenheten 1321b. F-funksjonsenheten 1321b besørger ikke-lineær omforming av de nedre dataene med bruk av nøkkelen key 1L. Deretter blir de omformede dataene matet inn til XOR-kretsen 1322b.

Nå skal mellomdataene, som er utdataene fra XOR-kretsen 1322a og som er lagt i det aritmetiske registeret 350, mates inn til XOR-kretsen 1322b. Etter dette er det behov for signalbanen fra det aritmetiske registeret 350 til signalvelgeren 310. Mer spesifikt gjør signalbanen fra det aritmetiske registeret 350 til signalvelgeren 310 det mulig å mate inn mellomdataene inneholdt i det aritmetiske registeret 350 til signalvelgeren 310. Signalvelgeren 310 velger de mottatte mellomdataene. Mellomdataene blir deretter matet inn til hovedkonverteren 320 via det aritmetiske registeret 350, og deretter XOR-behandlet med utdata fra F-funksjonsenheten 1321b i XOR-kretsen 1322b. Utdata fra XOR-kretsen 1322b blir matet inn til F-funksjonsenheten 1321c.

På denne måten blir samme prosess som den utført av F-funksjonsenheten 1321a, XOR-kretsen 1322a, F-funksjonsenheten 1321b og XOR-kretsen 1322b utført henholdsvis av en F-funksjonsenhet 1321c, en XOR-krets 1322c, en F-funksjonsenhet 1321d og en XOR-krets 1322d, av en F-funksjons-

enhet 1321e, en XOR-krets 1322e, en F-funksjonsenhet 1321f og en XOR-krets 1322f, av en F-funksjonsenhet 1321g, en XOR-krets 1322g, en F-funksjonsenhet 1321h og en XOR-krets 1322h, av en F-funksjonsenhet 1321i, en XOR-krets 1322i, en F-funksjonsenhet 1321j og en XOR-krets 1322j, og av en F-funksjonsenhet 1321k, en XOR-krets 1322k, en F-funksjonsenhet 1321l og en XOR-krets 1322l. Etter gjennomføring av den 12-runders, ikke-lineære dataomformingen i F-funksjonsenhetene (eller gjentagelse 12 ganger), blir de omformede dataene matet ut.

Problem 1.

Med henvisning til dataomformingsanordningene i figurene 56 og 60 anvender nøkkelgeneratoren 20 deler av hovedkonverteren 320 og deler av underkonverteren 330 for å generere en nøkkel for bruk til kryptering/dekryptering av data. Hensikten med å bruke deler av hovedkonverteren 320 og deler av underkonverteren 330 er å redusere dataomformingsanordningens totale størrelse.

Med denne nøkkelgenereringsoperasjonen beskrevet i detalj senere, er det for å generere en nøkkel med bruk av deler av hovedkonverteren 320 og deler av underkonverteren 330 nødvendig med en signalbane for å mate inn mellomnøkkelen (key KL) matet ut fra key KL-registeret 240 til signalvelgeren 310, som kan sees i figur 56. Denne utvidelsen av signalbanen fra key KL-registeret 240 til signalvelgeren 310 er et hinder for å redusere dataomformingsanordningens størrelse.

Antallet innsignaler til signalvelgeren 310 økes også som følge av signalbanen fra key KL-registeret 240 til signalvelgeren 310, slik at antallet signalvelgere i signalvelgeren 310 må økes. Dette er et annet hinder for å redusere dataomformingsanordningens størrelse.

Som tidligere nevnt ledsages den én-rundes F-funksjonsbaserte dataomformingen i to eller flere sykluser av et behov for å mate inn mellomdataene som lagres for en bestemt tidsperiode til hovedkonverteren 320. Denne utvidelsen av signalbanen for å overføre mellomdataene fra det aritmetiske registeret 350 til signalvelgeren 310 er nok et annet hinder for å redusere dataomformingsanordningens størrelse.

I tillegg gjør økningen av antallet innsignaler til signalvelgeren 310 som følge av signalbanen fra det aritmetiske registeret 350 til signalvelgeren 310 at antallet signalvelgere i signalvelgeren 310 må økes. Dette er nok et annet hinder for å redusere dataomformingsanordningens størrelse.

Problem 2.

I datascramblerne 30 i dataomformingsanordningene illustrert i figurene 56 og 60 er hovedkonverteren 320 og underkonverteren 330 seriekoplet. Dette bestemmer operasjonsfrekvensen entydig ved signalbanen fra hovedkonverteren 320 gjennom underkonverteren 330, signalvelgeren 310, det aritmetiske registeret 350 og deretter tilbake til hovedkonverteren 320, hvilket gjør det umulig øke operasjonsfrekvensen. Det har derfor vært et ønske om å øke operasjonsfrekvensen ved å gjøre den lengste signalbanen i datascrambleren 30 kortere, og med det øke gjennomløpshastigheten betraktelig. I tillegg er det ikke tilveiebragt noen signalbane som gjør at data som mates ut fra signalvelgeren 310 og deretter det aritmetiske registeret 350 kan sendes til underkonverteren 330 uten å gå gjennom hovedkonverteren 320. Som følge av dette muliggjøres ikke en fleksibel respons til en endring av den innvendige oppbygningen til dataomformingsanordningen, noe som gir liten fleksibilitet når det gjelder den generelle virkemåten.

Som nevnt tidligere, i tilfeller der den én-rundes F-funksjonsbaserte dataomformingen utføres i to eller flere sykluser, blir deler av innmatede data (halvparten av inndataene for en 1/2F-funksjon) omformet i én syklus. Dette krever at signalbanen i datascrambleren 30 overfører omformede data for den aktuelle delen av innmatede data til det aritmetiske registeret 350 for lagring der og deretter overfører de omformede dataene til underkonverteren 330 etter en viss tid. Ellers er overføringsbanen i hovedkonverteren 320 nødvendig for å overføre de omformede dataene til underkonverteren 330 via hovedkonverteren 320 etter en viss tid.

I tillegg, med kretsen delt av datakonverterenheten 50 og datainverterenheten 70 som illustrert i figur 59, svarer signalbanen $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E \rightarrow B \rightarrow C \cdots$ til en loopkrets. Dette krever at den delte kretsen er konstruert for ikke å bli en overføringsenhet i praksis når den påvirkes av "signal racing" forårsaket av forskjeller i forplantningsforsinkelsen til skiftesignaler, støy eller

annet. Et annet problem er at logisk syntese-verktøy ikke kan anvendes i en slik krets med en loopkrets (feedback-loop-krets), og derfor kan ikke logisk syntese utføres på en effektiv måte.

Det er et mål ved foreliggende oppfinnelse å forminske en data-omformingsanordning.

Det er et annet mål ved foreliggende oppfinnelse å øke operasjons-frekvensen til en dataomformingsanordning.

Hovedtrekkene ved oppfinnelsen fremgår av de selvstendige patentkrav. Ytterligere trekk ved oppfinnelsen er angitt i de uselvstendige krav.

En dataomformingsanordning ifølge foreliggende oppfinnelse mottar en nøkkel og data og omformer data for én av kryptering eller dekryptering av dataene med bruk av den mottatte nøkkelen.

Dataomformingsanordningen er kjennetegnet ved at den omfatter en datascrambler som utfører dataomforming, og en styringsenhet som kontrollerer et overføringssignal som angir at én av nøkkelen eller dataene skal overføres.

Videre er styringsenheten kjennetegnet ved at den mater ut overførings-signalet i tilfellet med overføring av den ene av nøkkelen eller dataene.

Videre er datascrambleren kjennetegnet ved at den omfatter en under-konverter som utfører dataomformingen for den ene av kryptering eller de-kryptering ved å omforme de mottatte dataene med bruk av den mottatte nøkkelen, og overfører minst én av den mottatte nøkkelen og de mottatte dataene uten å utføre dataomforming ved mottak av overføringssignalet matet ut fra styringsenheten.

Datascrambleren er kjennetegnet ved at den videre omfatter en hoved-konverter som mottar dataene og ikke-lineært omformer de mottatte dataene.

Videre er styringsenheten kjennetegnet ved at den mater ut et data-overføringssignal som overføringssignalet ved overføring av dataene.

Videre er underkonverteren kjennetegnet ved at den mottar data-overføringssignalet matet ut fra styringsenheten og dataene som er ikke-lineært omformet av hovedkonverteren, og overfører de mottatte dataene i overens-stemmelse med det mottatte dataoverføringssignalet.

Dataomformingsanordningen er kjennetegnet ved at den videre omfatter en nøkkelgenerator som genererer nøkkelen.

Videre er styringsenheten kjennetegnet ved at den mater ut et nøkkeloverføringssignal som overføringssignalet ved overføring av nøkkelen.

Videre er underkonverteren kjennetegnet ved at den mottar nøkkeloverføringssignalet matet ut fra styringsenheten og nøkkelen generert av nøkkelgeneratoren, og overfører den mottatte nøkkelen i overensstemmelse med det mottatte nøkkeloverføringssignalet.

Nøkkelgeneratoren er kjennetegnet ved at den videre omfatter en mellomnøkkel-generator som mottar en hemmelig nøkkel og genererer en mellomnøkkel basert på den mottatte hemmelige nøkkelen.

Videre er underkonverteren kjennetegnet ved at den, ved mottak av nøkkeloverføringssignalet matet ut fra styringsenheten, overfører mellomnøkkelen generert av mellomnøkkel-generatoren til hovedkonverteren i overensstemmelse med det mottatte nøkkeloverføringssignalet.

Videre er hovedkonverteren kjennetegnet ved at den gjentar omforming og utmating av mellomnøkkelen overført av underkonverteren minst én gang.

Videre er underkonverteren kjennetegnet ved at den gjentar omforming og utmating av mellomnøkkelen matet ut fra hovedkonverteren minst én gang.

Videre er minst én av hovedkonverteren og underkonverteren kjennetegnet ved at den/de gjentar omforming og utmating av mellomnøkkelen minst én gang.

Videre er hovedkonverteren kjennetegnet ved at den mater ut mellomnøkkelen matet ut fra minst én av hovedkonverteren og underkonverteren som en utnøkkel.

Videre er mellomnøkkel-generatoren kjennetegnet ved at den mottar utnøkkelen matet ut fra hovedkonverteren og genererer en utvidet nøkkel som omfatter mellomnøkkelen og utnøkkelen.

Mellomnøkkel-generatoren er kjennetegnet ved at den omfatter en 6-1 KL-signalvelger som velger ut én nøkkel av seks mottatte nøkler, og et key KL-register som inneholder den ene nøkkelen valgt av 6-1 KL-signalvelgeren som mellomnøkkelen.

Videre er 6-1 KL-signalvelgeren kjennetegnet ved at den mottar en hemmelig nøkkel, mottar seks nøkler omfattende den hemmelige nøkkelen, mellomnøkkelen inneholdt i key KL-registeret samt fire nøkler oppnådd gjennom rotasjonsskift av mellomnøkkelen inneholdt i key KL-registeret med fire forskjellige tall, og velger ut én nøkkel av de seks mottatte nøklene.

Videre er key KL-registeret kjennetegnet ved at det inneholder en nøkkel valgt av 6-1 KL-signalvelgeren.

Videre er underkonverteren kjennetegnet ved at den, ved mottak av nøkkeloverføringssignalet matet ut fra styringsenheten, mottar nøkkelen inneholdt i key KL-registeret som mellomnøkkelen og overfører den mottatte mellomnøkkelen.

Mellomnøkkel-generatoren er kjennetegnet ved at den omfatter en 4-1-signalvelger som velger ut én nøkkel av fire mottatte nøkler, en 3-1 KL-signalvelger som velger ut én nøkkel av tre mottatte nøkler, og et key KL-register som inneholder den ene nøkkelen valgt av 3-1 KL-signalvelgeren som mellomnøkkelen.

Videre er 4-1-signalvelgeren kjennetegnet ved at den mottar fire nøkler oppnådd gjennom rotasjonsskift av mellomnøkkelen inneholdt i key KL-registeret med fire forskjellige tall og velger ut én nøkkel av de fire mottatte nøklene.

Videre er 3-1 KL-signalvelgeren kjennetegnet ved at den mottar en hemmelig nøkkel, mottar tre nøkler omfattende den hemmelige nøkkelen, den ene nøkkelen valgt av 4-1-signalvelgeren og mellomnøkkelen inneholdt i key KL-registeret, og velger ut én nøkkel av de tre mottatte nøklene.

Videre er key KL-registeret kjennetegnet ved at det inneholder en nøkkel valgt av 3-1 KL-signalvelgeren.

Videre er underkonverteren kjennetegnet ved at den, ved mottak av nøkkeloverføringssignalet matet ut fra styringsenheten, mottar nøkkelen inneholdt i key KL-registeret som mellomnøkkelen og overfører den mottatte mellomnøkkelen.

Nøkkelgeneratoren er kjennetegnet ved at den videre omfatter en nøkkelfordeler som mottar den utvidede nøkkelen generert av mellomnøkkel-

generatoren og en forbestemt konstant, og skedulerer en nøkkel for utmating av én av den mottatte utvidede nøkkelen og den mottatte, forbestemte konstanten til minst én av hovedkonverteren og underkonverteren i overensstemmelse med en forbestemt betingelse.

Underkonverteren er kjennetegnet ved at den omfatter minst én av en datakonverterenhet (FL) som utfører lineær dataomforming, og en datainverterenhet (FL^{-1}) som utfører en dataomforming som er den inverse av den som utføres av datakonverterenheten (FL).

Videre er minst én av datakonverterenheten (FL) og datainverterenheten (FL^{-1}) kjennetegnet ved at den/de utfører dataomformingen og mottar overføringssignalet matet ut fra styringsenheten og overfører minst én av dataene og nøkkelen uten å utføre dataomforming i overensstemmelse med det mottatte overføringssignalet dersom styringsenheten mater ut overføringssignalet.

Styringsenheten er kjennetegnet ved at den mater ut et nøkkeloverføringssignal og et maskesignal som overføringssignalene for overføring av den mottatte nøkkelen.

Videre er minst én av datakonverterenheten (FL) og datainverterenheten (FL^{-1}) kjennetegnet ved at den/de overfører nøkkelen, ved mottak av nøkkeloverføringssignalet og maskesignalet matet ut fra styringsenheten, ved å undertrykke de mottatte dataene i overensstemmelse med det mottatte nøkkeloverføringssignalet og la den mottatte nøkkelen slippe gjennom uforandret i overensstemmelse med det mottatte maskesignalet.

Styringsenheten er kjennetegnet ved at den mater ut et DATA TRANSFER-signal som er et dataoverføringssignal som overføringssignalet for overføring av de mottatte dataene.

Videre er minst én av datakonverterenheten (FL) og datainverterenheten (FL^{-1}) kjennetegnet ved at den/de overfører dataene, ved mottak av DATA TRANSFER-signalet matet ut fra styringsenheten, ved å undertrykke den mottatte nøkkelen og la de mottatte dataene slippe gjennom uforandret i overensstemmelse med det mottatte DATA TRANSFER-signalet.

Underkonverteren er kjennetegnet ved at den omfatter en 1/2-underkonverterenhet som implementerer dataomforming for lineær dataomforming og datainvertering for en dataomforming som er den inverse av den lineære dataomformingen i en felles krets, og underkonverteren er kjennetegnet ved at den omformer dataene med bruk av 1/2-underkonverterenheten, mottar overføringssignalet matet ut fra styringsenheten dersom styringsenheten matet ut overføringssignalet, og overfører minst én av nøkkelen og dataene i overensstemmelse med det mottatte overføringssignalet.

Underkonverteren er kjennetegnet ved at den omfatter en datakonverterenhet (FL) som utfører lineær dataomforming, og en datainverterenhet (FL^{-1}) som utfører en dataomforming som er den inverse av den som utføres av datakonverterenheten (FL), der datakonverterenheten (FL) og datainverterenheten (FL^{-1}) er anordnet serielt.

Videre er én av datakonverterenheten (FL) og datainverterenheten (FL^{-1}) kjennetegnet ved at den mottar én av dataene omformet av den andre av datakonverterenheten (FL) og datainverterenheten (FL^{-1}), den overførte nøkkelen og de overførte dataene, og utfører én av omforming av data, overføring av en nøkkel og overføring av data med bruk av den ene av de omformede dataene, den overførte nøkkelen eller de overførte dataene som ble mottatt.

Dataomformingsanordningen er kjennetegnet ved at den mottar én av en 128-bits nøkkel, en 192-bits nøkkel eller en 256-bits nøkkel, og omformer de mottatte dataene med bruk av de mottatte nøklene.

En fremgangsmåte for omforming av data ifølge foreliggende oppfinnelse mottar en nøkkel og data, og utfører dataomforming for minst én av kryptering og dekryptering av de mottatte dataene med bruk av den mottatte nøkkelen.

Fremgangsmåten er kjennetegnet ved at den omfatter det å:

mate ut et overføringssignal som angir overføring av én av den mottatte nøkkelen og de mottatte dataene ved overføring av den ene av den mottatte nøkkelen og de mottatte dataene, og

utføre dataomformingen for den ene av kryptering og dekryptering ved å omforme de mottatte dataene med bruk av den mottatte nøkkelen, og overføre

minst én av den mottatte nøkkelen og de mottatte dataene uten å utføre dataomformingen ved mottak av det utmatede overføringssignalet.

Et program for omforming av data ifølge foreliggende oppfinnelse mottar en nøkkel og data, og utfører dataomforming for minst én av kryptering og dekryptering av de mottatte dataene med bruk av den mottatte nøkkelen.

Dataomformingsprogrammet er kjennetegnet ved at det instruerer en datamaskin til å:

mate ut et overføringssignal som angir overføring av én av den mottatte nøkkelen og de mottatte dataene ved overføring av den ene av den mottatte nøkkelen og de mottatte dataene, og

utføre dataomformingen for den ene av kryptering og dekryptering ved å omforme de mottatte dataene med bruk av den mottatte nøkkelen, og overføre minst én av den mottatte nøkkelen og de mottatte dataene uten å utføre dataomformingen ved mottak av det utmatede overføringssignalet.

Et datamaskin-lesbart lagringsmedium ifølge foreliggende oppfinnelse er et lagringsmedium som inneholder et dataomformingsprogram for å motta en nøkkel og data, og å utføre dataomforming for minst én av kryptering og dekryptering av de mottatte dataene med bruk av den mottatte nøkkelen.

Her er dataomformingsprogrammet kjennetegnet ved at det instruerer en datamaskin til å:

mate ut et overføringssignal som angir overføring av én av den mottatte nøkkelen og de mottatte dataene ved overføring av den ene av den mottatte nøkkelen og de mottatte dataene, og

utføre dataomformingen for den ene av kryptering og dekryptering ved å omforme de mottatte dataene med bruk av den mottatte nøkkelen, og overføre minst én av den mottatte nøkkelen og de mottatte dataene uten å utføre dataomformingen ved mottak av det utmatede overføringssignalet.

Dataomformingsanordningen er kjennetegnet ved at den videre omfatter en nøkkelgenerator for å generere en nøkkel.

Videre er nøkkelgeneratoren kjennetegnet ved at den videre omfatter en mellomnøkkel-generator som mottar en hemmelig nøkkel, genererer en mellom-

nøkkel basert på den mottatte hemmelige nøkkelen og genererer en utnøkkel basert på den genererte mellomnøkkelen med bruk av hovedkonverteren og underkonverteren.

Mellomnøkkel-generatoren er kjennetegnet ved at den omfatter en 6-1 KL-signalvelger som mottar seks nøkler og velger ut én nøkkel av de seks mottatte nøklene, et key KL-register som inneholder den ene nøkkelen valgt av 6-1 KL signalvelgeren som mellomnøkkelen, en 6-1 KA-signalvelger som velger ut én nøkkel av seks nøkler, og et key KA-register som inneholder den ene nøkkelen valgt av 6-1 KA-signalvelgeren som utnøkkelen.

Videre er 6-1 KL-signalvelgeren kjennetegnet ved at den mottar en hemmelig nøkkel, mottar seks nøkler omfattende den hemmelige nøkkelen, mellomnøkkelen inneholdt i key KL-registeret samt fire nøkler oppnådd gjennom rotasjonsskift av mellomnøkkelen inneholdt i key KL-registeret med fire forskjellige tall, og velger ut én nøkkel av de seks mottatte nøklene.

Videre er key KL-registeret kjennetegnet ved at det inneholder en nøkkel valgt av 6-1 KL-signalvelgeren som en mellomnøkkel.

Videre er 6-1 KA-signalvelgeren kjennetegnet ved at den mottar en utnøkkel generert med bruk av hovedkonverteren og underkonverteren, mottar seks nøkler omfattende den mottatte utnøkkelen, utnøkkelen inneholdt i key KA-registeret samt fire nøkler oppnådd gjennom rotasjonsskift av utnøkkelen inneholdt i key KA-registeret med fire forskjellige tall, og velger ut én nøkkel av de seks mottatte nøklene.

Videre er key KA-registeret kjennetegnet ved at det inneholder den ene nøkkelen valgt av 6-1 KA-signalvelgeren som en utnøkkel.

Mellomnøkkel-generatoren er kjennetegnet ved at den omfatter en 2-1-signalvelger som velger ut én nøkkel av to mottatte nøkler, en 4-1-signalvelger som velger ut én nøkkel av fire nøkler, en 3-1 KL-signalvelger som velger ut én nøkkel av tre nøkler, et key KL-register som inneholder den ene nøkkelen valgt av 3-1 KL-signalvelgeren som en mellomnøkkel, en 3-1 KA-signalvelger som velger ut én nøkkel av tre nøkler, og et key KA-register som inneholder den ene nøkkelen valgt av 3-1 KA-signalvelgeren som en utnøkkel.

Videre er 2-1-signalvelgeren kjennetegnet ved at den velger ut én nøkkel av mellomnøkkelen inneholdt i key KL-registeret og utnøkkelen inneholdt i key KA-registeret.

Videre er 4-1-signalvelgeren kjennetegnet ved at den mottar fire nøkler oppnådd gjennom rotasjonsskift av den ene nøkkelen valgt av 2-1-signalvelgeren med fire forskjellige tall, og velger ut én nøkkel av de fire mottatte nøklene.

Videre er 3-1 KL-signalvelgeren kjennetegnet ved at den mottar en hemmelig nøkkel, mottar tre nøkler omfattende den hemmelige nøkkelen, den ene nøkkelen valgt av 4-1-signalvelgeren og mellomnøkkelen inneholdt i key KL-registeret, og velger ut én nøkkel av de tre nøklene.

Videre er key KL-registeret kjennetegnet ved at den inneholder den ene nøkkelen valgt av 3-1 KL-signalvelgeren som en mellomnøkkel.

Videre er 3-1 KA-signalvelgeren kjennetegnet ved at den mottar en utnøkkel generert med bruk av hovedkonverteren og underkonverteren, mottar tre nøkler omfattende den mottatte utnøkkelen, den ene nøkkelen valgt av 4-1-signalvelgeren og utnøkkelen inneholdt i key KA-registeret, og velger ut én nøkkel av de tre nøklene.

Videre er key KA-registeret kjennetegnet ved at det inneholder den ene nøkkelen valgt av 3-1 KA-signalvelgeren som en utnøkkel.

Mellomnøkkel-generatoren er kjennetegnet ved at den omfatter en 2-1 KL-signalvelger som velger ut én nøkkel av to nøkler, et key KL-register som inneholder den ene nøkkelen valgt av 2-1 KL-signalvelgeren, en 2-1 KA-signalvelger som velger ut én nøkkel av to nøkler, et key KA-register som inneholder den ene nøkkelen valgt av 2-1 KA-signalvelgeren, en 2-1-signalvelger som velger ut én nøkkel av to nøkler, og en 8-1-signalvelger som velger ut én nøkkel av åtte nøkler.

Videre er 2-1 KL-signalvelgeren kjennetegnet ved at den mottar en hemmelig nøkkel og velger ut én nøkkel av den mottatte hemmelige nøkkelen og nøkkelen inneholdt i key KL-registeret.

Videre er 2-1 KA-signalvelgeren kjennetegnet ved at den mottar en utnøkkel generert med bruk av hovedkonverteren og underkonverteren, og velger

ut én nøkkel av den mottatte utnøkkelen og nøkkelen inneholdt i key KA-registeret.

Videre er 2-1-signalvelgeren kjennetegnet ved at den velger ut én nøkkel av to nøkler valgt av 2-1 KL-signalvelgeren og 2-1 KA-signalvelgeren.

Videre er 8-1 KL-signalvelgeren kjennetegnet ved at den mottar åtte nøkler oppnådd gjennom rotasjonsskift av den ene nøkkelen valgt av 2-1-signalvelgeren med åtte forskjellige tall, og velger ut én nøkkel av de åtte mottatte nøklene.

En anordning for å omforme data ifølge foreliggende oppfinnelse er utstyrt med en datascrambler for å omforme data.

Her er datascrambleren kjennetegnet ved at den omfatter en hoved-konverter som mottar data og utfører ikke-lineær omforming av de mottatte dataene, og en underkonverter som mottar data og utfører lineær omforming av de mottatte dataene, der hovedkonverteren og underkonverteren er anordnet serielt.

Hovedkonverteren, som mottar en nøkkel og data, er kjennetegnet ved at den utfører den ikke-lineære omformingen av de mottatte dataene med bruk av den mottatte nøkkelen basert på en F-funksjon, som er en funksjon som brukes til den ikke-lineære dataomformingen, og mater ut data frembragt gjennom den ikke-lineære dataomformingen.

Videre er underkonverteren, som mottar en nøkkel og data, kjennetegnet ved at den utfører den lineære omformingen av de mottatte dataene med bruk av den mottatte nøkkelen og mater ut data frembragt gjennom den lineære dataomformingen.

Videre er hovedkonverteren og underkonverteren kjennetegnet ved at de gjentar dataomformingen i hovedkonverteren og dataomformingen i under-konverteren, og omformer data for minst én av kryptering og dekryptering.

Hovedkonverteren er kjennetegnet ved at den omfatter en F-funksjons-enhet som gjentar den ikke-lineære omformingen av de mottatte dataene basert på F-funksjonen med bruk av den mottatte nøkkelen, og mater ut de om-formede dataene mer enn én gang.

F-funksjonsenheten er kjennetegnet ved at den gjentar den ikke-lineære dataomformingen basert på F-funksjonen mer enn én gang for å fullføre den ikke-lineære dataomformingen basert på F-funksjonen for én runde ved å gjenta 2^X ganger den ikke-lineære dataomformingen av de mottatte dataene basert på en $1/2^X$-F-funksjon ($X \geq 0$) med bruk av den mottatte nøkkelen og mate ut de omformede dataene, og for å gjenta utmating av dataene som har gjennomgått den ikke-lineære dataomformingen mer enn én gang.

F-funksjonsenheten er kjennetegnet ved at den mottar én av øvre data og nedre data atskilt, utfører den ikke-lineære omformingen av den mottatte ene av de øvre dataene og nedre dataene, mater ut den ene av de øvre dataene og nedre dataene omformet, XOR-behandler den utmatede ene av de øvre dataene og nedre dataene med en annen av de øvre dataene og de nedre dataene, bytter om de XOR-behandlede dataene og den andre av de øvre dataene og de nedre dataene som ikke ble mottatt av F-funksjonsenheten, og mater ut de ombyttede dataene.

Dataomformingsanordningen er kjennetegnet ved at den videre omfatter en nøkkelgenerator for å generere en nøkkel.

Her er nøkkelgeneratoren kjennetegnet ved at den omfatter en mellomnøkkel-generator som mottar en hemmelig nøkkel, genererer en mellomnøkkel basert på den mottatte hemmelige nøkkelen og genererer en utnøkkel basert på mellomnøkkelen med bruk av hovedkonverteren og underkonverteren.

Nøkkelgeneratoren er kjennetegnet ved at den videre omfatter en nøkkelfordeler som mottar mellomnøkkelen generert av mellomnøkkel-generatoren, utnøkkelen og en forbestemt konstant, og skedulerer en nøkkel som skal anvendes av hovedkonverteren og underkonverteren for data-omformingen, basert på den mottatte mellomnøkkelen, den mottatte utnøkkelen og den mottatte, forbestemte konstanten i overensstemmelse med en for-bestemt betingelse.

Videre er hovedkonverteren og underkonverteren begge kjennetegnet ved at de mottar nøkkelen skedulert av nøkkelfordeleren og omformer de respektive mottatte dataene basert på de respektive mottatte nøklene.

Mellomnøkkel-generatoren er kjennetegnet ved at den omfatter en 6-1 KL-signalvelger som mottar seks nøkler og velger ut én nøkkel av de seks mottatte nøklene, et key KL-register som inneholder den ene nøkkelen valgt av 6-1 KL signalvelgeren som mellomnøkkelen, en 6-1 KA-signalvelger som velger ut én nøkkel av seks nøkler, og et key KA-register som inneholder den ene nøkkelen valgt av 6-1 KA-signalvelgeren som utnøkkelen.

Videre er 6-1 KL-signalvelgeren kjennetegnet ved at den mottar en hemmelig nøkkel, mottar seks nøkler omfattende den hemmelige nøkkelen, mellomnøkkelen inneholdt i key KL-registeret og fire nøkler oppnådd gjennom rotasjonsskift av mellomnøkkelen inneholdt i key KL-registeret med fire forskjellige tall, og velger ut én nøkkel av de seks mottatte nøklene.

Videre er key KL-registeret kjennetegnet ved at det inneholder en nøkkel valgt av 6-1 KL-signalvelgeren som en mellomnøkkel.

Videre er 6-1 KA-signalvelgeren kjennetegnet ved at den mottar en utnøkkel generert med bruk av hovedkonverteren og underkonverteren, mottar seks nøkler omfattende den mottatte utnøkkelen, utnøkkelen inneholdt i key KA-registeret og fire nøkler oppnådd gjennom rotasjonsskift av utnøkkelen inneholdt i key KA-registeret med fire forskjellige tall, og velger ut én nøkkel av de seks mottatte nøklene.

Videre er key KA-registeret kjennetegnet ved at det inneholder den ene nøkkelen valgt av 6-1 KA-signalvelgeren som en utnøkkel.

Mellomnøkkel-generatoren er kjennetegnet ved at den omfatter en 2-1-signalvelger som velger ut én nøkkel av to nøkler, en 4-1-signalvelger som velger ut én nøkkel av fire nøkler, en 3-1 KL-signalvelger som velger ut én nøkkel av tre nøkler, et key KL-register som inneholder den ene nøkkelen valgt av 3-1 KL-signalvelgeren som en mellomnøkkel, en 3-1 KA-signalvelger som velger ut én nøkkel av tre nøkler, og et key KA-register som inneholder den ene nøkkelen valgt av 3-1 KA-signalvelgeren som en utnøkkel.

Videre er 2-1-signalvelgeren kjennetegnet ved at den velger ut én nøkkel av mellomnøkkelen inneholdt i key KL-registeret og utnøkkelen inneholdt i key KA-registeret.

Videre er 4-1-signalvelgeren kjennetegnet ved at den mottar fire nøkler oppnådd gjennom rotasjonsskift av den ene nøkkelen valgt av 2-1-signalvelgeren med fire forskjellige tall, og velger ut én nøkkel av de fire mottatte nøklene.

Videre er 3-1 KL-signalvelgeren kjennetegnet ved at den mottar en hemmelig nøkkel, mottar tre nøkler omfattende den hemmelige nøkkelen, den ene nøkkelen valgt av 4-1-signalvelgeren og mellomnøkkelen inneholdt i key KL-registeret, og velger ut én nøkkel av de tre nøklene.

Videre er key KL-registeret kjennetegnet ved at det inneholder den ene nøkkelen valgt av 3-1 KL-signalvelgeren som en mellomnøkkel.

Videre er 3-1 KA-signalvelgeren kjennetegnet ved at den mottar en utnøkkel generert med bruk av hovedkonverteren og underkonverteren, mottar tre nøkler omfattende den mottatte utnøkkelen, den ene nøkkelen valgt av 4-1-signalvelgeren og utnøkkelen inneholdt i key KA-registeret, og velger ut én nøkkel av de tre nøklene.

Videre er key KA-registeret kjennetegnet ved at det inneholder den ene nøkkelen valgt av 3-1 KA-signalvelgeren som en utnøkkel.

Mellomnøkkel-generatoren er kjennetegnet ved at den omfatter en 2-1 KL-signalvelger som velger ut én nøkkel av to nøkler, et key KL-register som inneholder den ene nøkkelen valgt av 2-1 KL-signalvelgeren, en 2-1 KA-signalvelger som velger ut én nøkkel av to nøkler, et key KA-register som inneholder den ene nøkkelen valgt av 2-1 KA-signalvelgeren, en 2-1-signalvelger som velger ut én nøkkel av to nøkler, samt en 8-1-signalvelger som velger ut én nøkkel av åtte nøkler.

Videre er 2-1 KL-signalvelgeren kjennetegnet ved at den mottar en hemmelig nøkkel og velger ut én nøkkel av den mottatte hemmelige nøkkelen og nøkkelen inneholdt i key KL-registeret.

Videre er 2-1 KA-signalvelgeren kjennetegnet ved at den mottar en utnøkkel generert med bruk av hovedkonverteren og underkonverteren og velger ut én nøkkel av den mottatte utnøkkelen og nøkkelen inneholdt i key KA-registeret.

Videre er 2-1-signalvelgeren kjennetegnet ved at den velger ut én nøkkel av de to nøklene valgt av 2-1 KL-signalvelgeren og 2-1 KA-signalvelgeren.

Videre er 8-1-signalvelgeren kjennetegnet ved at den mottar åtte nøkler oppnådd gjennom rotasjonsskift av den ene nøkkelen valgt av 2-1-signalvelgeren med åtte forskjellige tall, og velger ut én nøkkel av de åtte mottatte nøklene.

Underkonverteren er kjennetegnet ved at den omfatter minst én av en datakonverterenhet (FL) som utfører lineær dataomforming, og en datainverterenhet (FL^{-1}) som utfører en dataomforming som er den inverse av den som utføres av datakonverterenheten (FL), og utfører dataomformingen med bruk av minst én av datakonverterenheten (FL) og datainverterenheten (FL^{-1}).

Underkonverteren er kjennetegnet ved at den omfatter en 1/2-underkonverterenhet som implementerer dataomforming for lineær omforming og datainvertering for en dataomforming som er den inverse av den lineære dataomformingen i en felles krets, og omformer dataene med bruk av 1/2-underkonverterenheten.

Dataomformingsanordningen er kjennetegnet ved at den mottar én av en 128-bits nøkkel, en 192-bits nøkkel eller en 256-bits nøkkel, og utfører dataomforming for én av kryptering og dekryptering av de mottatte dataene med bruk av den mottatte nøkkelen.

En fremgangsmåte for omforming av data ifølge foreliggende oppfinnelse er kjennetegnet ved at den omfatter det å:

utføre en ikke-lineær omforming av mottatte data og mate ut data frembragt gjennom den ikke-lineære dataomformingen, med bruk av en hovedkonverter anordnet parallelt med en underkonverter som utfører en lineær dataomforming, som mottar en nøkkel og data og utfører den ikke-lineære dataomformingen med bruk av den mottatte nøkkelen basert på en F-funksjon, som er en funksjon som brukes til den ikke-lineære dataomformingen, og

omforme mottatte data, med bruk av en mottatt nøkkel, for minst én av kryptering og dekryptering ved gjenta det å motta en nøkkel og data, det å utføre en lineær omforming av de mottatte dataene med bruk av den mottatte nøkkelen med underkonverteren som er anordnet parallelt med hoved-

konverteren, og det å mate ut dataene frembragt gjennom den lineære dataomformingen.

Et program for omforming av data ifølge foreliggende oppfinnelse er kjennetegnet ved at den instruerer en datamaskin til å:

utføre en ikke-lineær omforming av mottatte data og mate ut data frembragt gjennom den ikke-lineære dataomformingen, med bruk av en hovedkonverter anordnet parallelt med en underkonverter som utfører en lineær dataomforming, som mottar en nøkkel og data og utfører den ikke-lineære dataomformingen med bruk av den mottatte nøkkelen basert på en F-funksjon, som er en funksjon som brukes til den ikke-lineære dataomformingen, og

omforme mottatte data, med bruk av en mottatt nøkkel, for minst én av kryptering og dekryptering ved gjenta det å motta en nøkkel og data, det å utføre en lineær omforming av de mottatte dataene med bruk av den mottatte nøkkelen med underkonverteren som er anordnet parallelt med hovedkonverteren, og det å mate ut dataene frembragt gjennom den lineære dataomformingen.

Et datamaskin-lesbart lagringsmedium ifølge foreliggende oppfinnelse er kjennetegnet ved at det inneholder et dataomformingsprogram som instruerer en datamaskin til å:

utføre en ikke-lineær omforming av mottatte data og mate ut data frembragt gjennom den ikke-lineære dataomformingen, med bruk av en hovedkonverter koplet i parallell med en underkonverter som utfører en lineær dataomforming, som mottar en nøkkel og data og utfører den ikke-lineære dataomformingen med bruk av den mottatte nøkkelen basert på en F-funksjon, som er en funksjon som brukes til den ikke-lineære dataomformingen, og

omforme mottatte data, med bruk av en mottatt nøkkel, for minst én av kryptering og dekryptering ved gjenta det å motta en nøkkel og data, det å utføre en lineær omforming av de mottatte dataene med bruk av den mottatte nøkkelen med underkonverteren som er anordnet parallelt med hovedkonverteren, og det å mate ut dataene frembragt gjennom den lineære dataomformingen.

Figur 1 er et diagram som illustrerer en utførelse av en dataomformingsanordning ifølge en første utførelsesform.

Figur 2 er et diagram som illustrerer en prosess der en mellomnøkkelgenerator 40 genererer en utnøkkel basert på en mellomnøkkel med en 128-bits nøkkel.

Figur 3 er et diagram som illustrerer en innvendig oppbygning av og en virkemåte for en nøkkelfordeler 210.

Figur 4 er et diagram som illustrerer en virkemåte for en datascrambler 30 for kryptering/dekryptering.

Figur 5 er et diagram som illustrerer en innvendig oppbygning av og en virkemåte for en F-funksjonsenhet 321.

Figur 6 er et diagram som illustrerer en utførelse av en dataomformingsanordning der en hovedkonverter 320 og en underkonverter 330 vist figur 1 er anordnet i omvendt rekkefølge.

Figur 7 er et diagram som illustrerer en utførelse av en dataomformingsanordning der hovedkonverteren 320 og underkonverteren 330 er koplet i parallell.

Figur 8 er et diagram som illustrerer en innvendig oppbygning av en 6-1 KL-signalvelger 220 og en 6-1 KA-signalvelger 230 i mellomnøkkel-generatoren 40.

Figur 9 er et diagram som illustrerer et annet eksempel på utførelse av mellomnøkkel-generatoren 40.

Figur 10 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 er utstyrt med en nøkkeloverføringsfunksjon.

Figur 11 er et diagram som illustrerer en utførelse av underkonverteren 330 der datainverterenheten 70 er utstyrt med nøkkeloverføringsfunksjonen.

Figur 12 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 og datainverterenheten 70 begge er utstyrt med nøkkeloverføringsfunksjonen.

Figur 13 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 er utstyrt med en dataoverføringsfunksjon ifølge en andre utførelsesform.

Figur 14 er et diagram som illustrerer en utførelse av underkonverteren 330 der datainverterenheten 70 er utstyrt med dataoverføringsfunksjonen.

Figur 15 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 og datainverterenheten 70 begge er utstyrt med dataoverføringsfunksjonen.

Figur 16 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 er utstyrt med nøkkeloverføringsfunksjonen og dataoverføringsfunksjonen, og datainverterenheten 70 er utstyrt med en nøkkeloverføringsfunksjon ifølge en tredje utførelsesform.

Figur 17 er et diagram som illustrerer en utførelse av underkonverteren 330 der datainverterenheten 70 er utstyrt med nøkkeloverføringsfunksjonen og dataoverføringsfunksjonen, og datakonverterenheten 50 er utstyrt med nøkkeloverføringsfunksjonen.

Figur 18 er et diagram som illustrerer en utførelse av underkonverteren der datakonverterenheten 50 og datainverterenheten 70 begge er utstyrt med nøkkeloverføringsfunksjonen og dataoverføringsfunksjonen.

Figur 19 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 er utstyrt med nøkkeloverføringsfunksjonen og dataoverføringsfunksjonen.

Figur 20 er et diagram som illustrerer en utførelse av underkonverteren 330 der datainverterenheten 70 er utstyrt med nøkkeloverføringsfunksjonen og dataoverføringsfunksjonen.

Figur 21 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 er utstyrt med både nøkkeloverføringsfunksjonen og dataoverføringsfunksjonen, og datainverterenheten 70 er utstyrt med dataoverføringsfunksjonen.

Figur 22 er et diagram som illustrerer en utførelse av underkonverteren 330 der datainverterenheten 70 er utstyrt med nøkkeloverføringsfunksjonen og dataoverføringsfunksjonen, og datakonverterenheten 50 er utstyrt med dataoverføringsfunksjonen.

Figur 23 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 er utstyrt med dataoverføringsfunksjonen og datainverterenheten 70 er utstyrt med nøkkeloverføringsfunksjonen.

Figur 24 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 er utstyrt med nøkkeloverføringsfunksjonen og datainverterenheten 70 er utstyrt med dataoverføringsfunksjonen.

Figur 25 er et diagram som illustrerer en utførelse av underkonverteren 330 der datainverterenheten 70 og datakonverterenheten 50 er koplet i serie, og der både datakonverterenheten 50 og datainverterenheten 70 er utstyrt med dataoverføringsfunksjonen.

Figur 26 illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 og datainverterenheten 70 i figur 25 har byttet plass.

Figur 27 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 og datainverterenheten 70 er koplet i serie, og der datakonverterenheten 50 er utstyrt med nøkkeloverføringsfunksjonen og dataoverføringsfunksjonen og datainverterenheten 70 er utstyrt med dataoverføringsfunksjonen.

Figur 28 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 og datainverterenheten 70 i figur 27 har byttet plass.

Figur 29 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 og datainverterenheten 70 er koplet i serie, og der datakonverterenheten 50 er utstyrt med dataoverføringsfunksjonen og datainverterenheten 70 er utstyrt med nøkkeloverføringsfunksjonen og dataoverføringsfunksjonen.

Figur 30 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 og datainverterenheten 70 i figur 29 har byttet plass.

Figur 31 er et diagram som illustrerer en utførelse av underkonverteren 330 der en 1/2-underkonverterenhet 90 er utstyrt med nøkkeloverføringsfunksjonen og dataoverføringsfunksjonen ifølge en fjerde utførelsesform.

Figur 32 er et diagram som illustrerer en utførelse av underkonverteren 330 der en 1/2-underkonverterenhet 90 er utstyrt med dataoverføringsfunksjonen.

Figur 33 er et diagram som illustrerer en utførelse av underkonverteren 330 der en 1/2-underkonverterenhet 90 er utstyrt med nøkkeloverføringsfunksjonen.

Figur 34 er et diagram som illustrerer en prosess for kryptering av data som utføres i en CAMELLIA-basert dataomformingsanordning som anvender en 128-bits nøkkel.

Figur 35 er et diagram som illustrerer en prosess for dekryptering av data som utføres i en CAMELLIA-basert dataomformingsanordning som anvender en 128-bits nøkkel.

Figur 36 er et diagram som illustrerer en innvendig oppbygning av F-funksjonen i en CAMELLIA-basert dataomformingsanordning.

Figur 37 er et diagram som illustrerer en komplett utførelse og virkemåte ifølge en femte utførelsesform.

Figur 38 er et diagram som illustrerer en komplett utførelse og virkemåte ifølge en sjette utførelsesform.

Figur 39 er et diagram som illustrerer en komplett utførelse og virkemåte ifølge en ellevte utførelsesform.

Figur 40 er et diagram som illustrerer en komplett utførelse og virkemåte ifølge en tolvte utførelsesform.

Figur 41 er et diagram som illustrerer en komplett utførelse og virkemåte ifølge en trettende utførelsesform.

Figur 42 er et diagram som illustrerer en komplett utførelse og virkemåte ifølge en fjortende utførelsesform.

Figur 43 er et diagram som illustrerer en komplett utførelse og virkemåte ifølge en femtende utførelsesform.

Figur 44 er et diagram som illustrerer en komplett utførelse og virkemåte ifølge en sekstende utførelsesform.

Figur 45 er et diagram som illustrerer en komplett utførelse og virkemåte ifølge en syttende utførelsesform.

Figur 46 er et diagram som illustrerer en komplett utførelse og virkemåte ifølge en attende utførelsesform.

Figur 47 er et diagram som illustrerer en komplett utførelse og virkemåte ifølge en syvende utførelsesform.

Figur 48 er et diagram som illustrerer en komplett utførelse og virkemåte ifølge en åttende utførelsesform.

Figur 49 er et diagram som illustrerer en komplett utførelse og virkemåte ifølge en niende utførelsesform.

Figur 50 er et diagram som illustrerer en komplett utførelse og virkemåte ifølge en tiende utførelsesform.

Figur 51 er et diagram som illustrerer en komplett utførelse og virkemåte ifølge en nittende utførelsesform.

Figur 52 er et diagram som illustrerer en komplett utførelse og virkemåte ifølge en tyvende utførelsesform.

Figur 53 er et diagram som illustrerer en operasjon der mellomnøkkel-generatoren 40 genererer utnøkkelen basert på mellomnøkkelen for en 192- eller 256-bits nøkkel.

Figur 54 er et diagram som illustrerer en prosess for kryptering av data som utføres i en CAMELLIA-basert dataomformingsanordning som anvender en 192- eller 256-bits nøkkel.

Figur 55 er et diagram som illustrerer en prosess for dekryptering av data som utføres i en CAMELLIA-basert dataomformingsanordning som anvender en 192- eller 256-bits nøkkel.

Figur 56 er et diagram som illustrerer en utførelse av og en virkemåte for en beslektet dataomformingsanordning.

Figur 57 illustrerer et eksempel på utførelse av hovedkonverteren 320.

Figur 58 er et diagram som illustrerer kretsen som danner under-konverteren 330.

Figur 59 er et diagram som illustrerer en krets som deles av data-konverterenheten 50 og datainverterenheten 70, og som danner under-konverteren 330.

Figur 60 illustrerer et annet eksempel på utførelse av og virkemåte for den beslektede dataomformingsanordningen.

Figur 61 illustrerer et annet eksempel på utførelse av hovedkonverteren 320.

Figur 62 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 og datainverterenheten 70 er koplet i serie, og der datakonverterenheten 50 og datainverterenheten 70 begge er utstyrt med nøkkeloverføringsfunksjonen og dataoverføringsfunksjonen.

Figur 63 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 og datainverterenheten 70 i figur 62 har byttet plass.

Figur 64 illustrerer et eksempel på utførelse av en CAMELLIA-basert hovedkonverter 320.

Utførelsesform 1

En beskrivelse vil nå bli gitt av dataomformingsanordningen ifølge denne utførelsesformen.

Dataomformingsanordning

Figur 1 er et diagram som illustrerer en utførelse av og en virkemåten for en dataomformingsanordning ifølge denne utførelsesformen.

Denne utførelsesformen omfatter ikke "signalbanen for å mate inn mellomnøkkelen (key KL) som mates ut fra key KL-registeret 240 til signalvelgeren 310", og omfatter heller ikke "signalbanen for å mate inn data som mates ut fra key KL-registeret 240 til signalvelgeren 310". Disse signalbanene er illustrert i figurene 56 og 60. Grunnen til dette er at en underkonverter 330 i denne utførelsesformen har en ekstra nøkkel-/dataoverføringsfunksjon i tillegg til sin primære og opprinnelige funksjon med å omforme data.

I det følgende vil en beskrivelse bli gitt av generering av nøkler og kryptering/dekryptering av data med bruk av nøkkel-/dataoverføringsfunksjonen i underkonverteren 330. Andre komponenter og virkemåter er de samme som de beskrevet i forbindelse med figurene 56 og 60, og vil derfor ikke bli beskrevet her.

I denne utførelsesformen blir ikke mellomnøkkelen (key KL) matet ut fra key KL-registeret 240 matet direkte inn til signalvelgeren 310, men blir i stedet matet inn til underkonverteren 330 via nøkkelfordeleren 210 ved hjelp av den tradisjonelle signalbanen fra key KL-registeret 240 til nøkkelfordeleren 210. Underkonverteren 330, som kan være i "dataomformingsmodus" eller i "nøkkel-/dataoverføringsmodus", skifter til "nøkkel-/dataoverføringsmodus" ved mottak av en nøkkel, og overfører den innmatede nøkkelen til signalvelgeren 310.

Videre blir i denne utførelsesformen data som er ikke-lineært omformet av hovedkonverteren 320 ikke matet direkte inn til signalvelgeren 310, men blir matet inn til underkonverteren 330 først. Underkonverteren 330, når den mottar dataene som er ikke-lineært omformet av hovedkonverteren 320, skifter til "nøkkel-/dataoverføringsmodus" og overfører de innmatede dataene til signalvelgeren 310.

Overføringsoperasjonene som på denne måten gjennomføres av underkonverteren 330 gjør det mulig å kvitte seg med de to signalbanene illustrert i figurene 56 og 60.

Merk at en stiplet linje i figur 1 angir en "signalbane for å overføre mellomdataene fra det aritmetiske registeret 350 til signalvelgeren 310" som er nødvendig for å mate inn mellomdataene, som holdes for en gitt tidsperiode, til hovedkonverteren 320 i tilfellet der hovedkonverteren 320 utfører en F-funksjonsbasert dataomforming for én runde over to eller flere sykluser, som nevnt tidligere. På den annen side er ikke "signalbanen fra det aritmetiske registeret 350 til signalvelgeren 310" angitt av den stiplede linjen nødvendig i tilfellet der hovedkonverteren 320 utfører en F-funksjonsbasert dataomforming for én runde innenfor én syklus. Det samme gjelder for en signalbane angitt av en stiplet linje i figur 6, som beskrives senere.

Fremgangsmåte for generering av nøkler

En beskrivelse vil nå bli gitt av en fremgangsmåte for generering av en mellomnøkkel og en utnøkkel i mellomnøkkel-generatoren 40.

Figur 2 er et diagram som illustrerer en operasjon der mellomnøkkel-generatoren 40 genererer en utnøkkel basert på en mellomnøkkel.

Først blir en hemmelig nøkkel matet inn til mellomnøkkel-generatoren 40 og lagt i key KL-registeret 240 som mellomnøkkelen (key KL) ved hjelp av en 6-1 KL-signalvelger 220. Den hemmelige nøkkelen inneholdt i key KL-registeret 240 blir matet inn til hovedkonverteren 320 som mellomnøkkelen (key KL) ved hjelp av nøkkelfordeleren 210. I den første F-funksjonsenheten 321a i hovedkonverteren 320 blir de øvre eller mest signifikante bit av den innmatede mellomnøkkelen (key KL) ikke-lineært omformet med bruk av en konstant $\Sigma 1$ som er matet ut fra nøkkelfordeleren 210, og deretter XOR-behandlet med de nedre eller minst signifikante bit av mellomnøkkelen (key KL) i XOR-kretsen 322a og matet inn til F-funksjonsenheten 321b. Tilsvarende blir i F-funksjonsenheten 321b en nøkkel matet ut fra XOR-kretsen 322a ikke-lineært omformet med bruk av en konstant $\Sigma 2$ som er matet ut fra nøkkelfordeleren 210, og deretter XOR-behandlet med de nedre bit av mellomnøkkelen (key KL) i XOR-kretsen 322b. Deretter blir den resulterende utnøkkelen som de øvre bit av

nøkkelen og nøkkelen matet ut fra XOR-kretsen 322a som de nedre bit av nøkkelen matet ut til underkonverteren 330.

Underkonverteren 330 mottar disse dataene og utfører en eksklusiv ELLER-(XOR)-operasjon mellom de øvre bit og de nedre bit av nøkkelen ved hjelp av to eksklusiv ELLER-operatorer (XOR-operatorer) innlemmet i datakonverterenheten 50 og to eksklusiv ELLER-operatorer (XOR-operatorer) innlemmet i datainverterenheten 70 i underkonverteren 330. Deretter blir de resulterende utdataene sendt tilbake til hovedkonverteren 320.

Hovedkonverteren 320 utfører den to-trinns omformingen, som involverer F-funksjonsenheten 321b, XOR-kretsen 322a, F-funksjonsenheten 321b og XOR-kretsen 322b i hovedkonverteren 320, på samme måte som i den ovenfor beskrevne prosessen med bruk av delen av hovedkonverteren 320, og bytter deretter om de øvre bit og de nedre bit av den omformede nøkkelen og mater ut de ombyttede dataene.

De utmatede dataene mates inn til 6-1 KA-signalvelgeren 230 i mellomnøkkel-generatoren 40 og blir lagt i key KA-registeret 250 som utnøkkelen (key KA). Mellomnøkkel-generatoren 40 genererer således utnøkkelen (key KA) basert på mellomnøkkelen (key KL) med bruk av deler av hovedkonverteren 320 og deler av underkonverteren 330 som komponenter for å utføre kryptering/dekryptering. Fire nøkler, omfattende nøkkelen key KLH bestående av de øvre bitene og nøkkelen key KLL bestående av de nedre bitene av den genererte mellomnøkkelen (key KL), og nøkkelen key KAH bestående av de øvre bitene og nøkkelen key KAL bestående av de nedre bitene av den genererte utnøkkelen (key KA), blir matet inn til nøkkelfordeleren 210 og benyttet som nøkkel for kryptering/dekryptering av data (kalt en utvidet nøkkel). Deretter blir den således genererte utnøkkelen (key KA) og mellomnøkkelen (key KL) brukt til å generere en annen mellomnøkkel og en annen utnøkkel i hver gitte periode gjennom den samme prosessen.

Skedulering av nøkler

En beskrivelse vil nå bli gitt av en utførelse av og en virkemåte for nøkkelfordeleren 210.

Figur 3 er et diagram som illustrerer en innvendig oppbygning av og en virkemåte for nøkkelfordeleren 210.

Mellomnøkkelen (key KL) matet ut fra mellomnøkkel-generatoren 40 blir delt inn i en nøkkel key KLH omfattende de øvre bitene og en nøkkel key KLL omfattende de nedre bitene, og matet inn til en 4-1-signalvelger 216 og en 4-1-signalvelger 217. Utnøkkelen (key KA) matet ut fra mellomnøkkel-generatoren 40 blir også delt inn i nøkler key KAY og key KAL og matet inn til 4-1-signalvelgeren 216 og 4-1-signalvelgeren 217 på samme måte. 4-1-signalvelgeren 216 og 4-1-signalvelgeren 217 velger ut én av de fire nøklene. Deretter blir et signal valgt av 4-1-signalvelgerne 216, 217 og et signal oppnådd gjennom et én-bits rotasjonsskift av det valgte signalet mot høyre henholdsvis matet inn til en 2-1-signalvelger 214, 215. Grunnen til at signalet blir rotasjonsskiftet én bit mot høyre er som følger. Som beskrevet tidligere bruker mellomnøkkel-generatoren 40 underkonverteren 330 til å generere utnøkkelen (key KA). Når dette skjer, blir signalet rotasjonsskiftet én bit mot venstre av en rotasjonsskifter i underkonverteren 330. Derfor, antatt at signalet vil bli rotasjonsskiftet én bit mot venstre, blir signalet på forhånd rotasjonsskiftet én bit mot høyre, slik at rotasjonsskiftene ikke får noen innvirkning på det endelige resultatet. Nøkkel-fordeleren 210 vil imidlertid ikke alltid utføre et én-bits rotasjonsskift mot høyre. Dette avhenger av antallet bit og retningen til et rotasjonsskift som rotasjons-skift-enheten i underkonverteren 320 vil utføre på et signal. Med andre ord vil nøkkelfordeleren 210 på forhånd utføre et rotasjonsskift på signalet med samme antall bit som og i motsatt retning av den til et rotasjonsskift som vil bli gjort på signalet av rotasjonsskifteren i underkonverteren 330. Derfor vil 2-1-signalvelgeren 214 og 2-1-signalvelgeren 215, som skal velge ut et signal knyttet til en nøkkel av disse to signalene, alltid velge en nøkkel som på forhånd er rotasjonsskiftet et forbestemt antall bit, og mate ut nøkkelen til under-konverteren 330 når den mater ut en nøkkel til underkonverteren 330 for å generere utnøkkelen (key KA).

Nøkler matet ut fra 2-1-signalvelgeren 214 og 2-1-signalvelgeren 215 blir matet inn til underkonverteren 330 i tilfellet der underkonverteren 330 brukes til å generere utnøkkelen (key KA), og blir matet inn til en 2-1-signalvelger 212 i tilfellet der hovedkonverteren 320 brukes til å generere utnøkkelen (key KA) og under kryptering/dekryptering av data. Da blir en nøkkel som har gjennomgått et én-byte rotasjonsskift mot venstre eller høyre matet inn til 2-1-signalvelgeren 212. Grunnen til at nøkkelen som er rotasjonsskiftet én byte mot venstre eller

høyre blir matet inn til 2-1-signalvelgeren 212, er at prosessen for kryptering/dekryptering av data trenger denne nøkkelen dersom F-funksjonsenheten utgjøres av flere deler som prosesserer F-funksjonen for mindre enn én runde, som for eksempel 1/2, 1/4 og 1/8, som vil bli beskrevet mer i detalj senere.

Signalvelgeren 212 velger ut én nøkkel av disse to nøklene og mater inn den valgte nøkkelen til en 2-1-signalvelger 211. En 8-1-signalvelger 213 mottar konstanter Σ1 til Σ4, hver delt inn i øvre data og nedre data, velger ut ett signal av disse åtte innmatede signalene og mater inn det valgte signalet til 2-1-signalvelgeren 211. 2-1-signalvelgeren 211 velger ett av de to innmatede signalene og mater ut det valgte signalet til hovedkonverteren 320 som en nøkkel.

Kryptering/dekryptering av data

En beskrivelse vil nå bli gitt av kryptering/dekryptering av data utført av datascrambleren 30.

Figur 4 er et diagram som illustrerer en virkemåte for datascrambleren 30 ved kryptering/dekryptering.

Først innmates P (klartekst eller et kryptogram). Det er her antatt at P (klartekst eller et kryptogram) er 128 bit lang. De innmatede dataene, P, blir matet inn til en XOR-krets 31a og XOR-behandlet med en nøkkel (128 bit) som er matet inn til og deretter blir matet ut fra nøkkelgeneratoren 20 via mellomnøkkel-generatoren 40 og nøkkelfordeleren 210. Merk at den hemmelige nøkkelen først blir matet inn til mellomnøkkel-generatoren 40, deretter valgt av 6-1 KL-signalvelgeren 220, deretter lagt i key KL-registeret 240 som mellomnøkkelen (key KL), og deretter matet inn til nøkkelfordeleren 210 som mellomnøkkelen (key KL).

Med CAMELLIA (camellia) innrettet for blokk-basert kryptering med fellesnøkler anvendes eksklusive ELLER-operatorer i underkonverteren 330 som XOR-kretsen 31a og XOR-kretsen 31b. Mer spesifikt, som kan sees i figur 58, deles innmatede data opp i øvre data bestående av de øvre bitene og nedre data bestående av de nedre bitene. Deretter blir hver del av de oppdelte dataene og en innmatet nøkkel XOR-behandlet i XOR-kretsen 55 og XOR-kretsen 56 i datakonverterenheten 50, eller i XOR-kretsen 75 og XOR-kretsen 76 i datainverterenheten 70, og så matet ut.

De utmatede dataene blir omformet av hovedkonverteren 320 og underkonverteren 330 med bruk av én av de utvidede nøklene matet ut fra nøkkelfordeleren 210. I figur 4 utføres omforming av data skiftesvis i rekkefølgen: hovedkonverter 320a, underkonverter 330a, hovedkonverter 320b, underkonverter 330b og hovedkonverter 320c.

De på denne måten omformede dataene blir XOR-behandlet med en nøkkel matet ut fra nøkkelfordeleren 210 i XOR-kretsen 31b i underkonverteren 330 og matet ut som C (et kryptogram eller dechiffrert tekst).

En detaljert beskrivelse av CAMELLIA-basert omforming av data utført av hovedkonverteren 320 og underkonverteren 330 i datascrambleren 30 vil nå bli gitt med henvisning til figurene 1 og 4.

Data matet ut fra XOR-kretsen 31a blir delt opp i øvre data og nedre data og matet inn til hovedkonverteren 320a. I hovedkonverteren 320a blir hver del av de innmatede dataene ikke-lineært omformet, og de øvre dataene og nedre dataene blir byttet om som illustrert i figur 4, slik at de omformede nedre dataene nå blir behandlet som øvre data og de omformede øvre dataene blir behandlet som nedre data, og deretter matet inn til underkonverteren 330a.

I underkonverteren 330a blir de innmatede dataene lineært omformet. Som kan sees i figur 1 blir de omformede dataene matet inn til signalvelgeren 310, deretter lagt i det aritmetiske registeret 350 og så matet inn til hovedkonverteren 320 (vist som hovedkonverter 320b i figur 4).

Hovedkonverteren 320b og underkonverteren 330b utfører de samme prosessene som henholdsvis ble utført av hovedkonverteren 320a og underkonverteren 330a. Den samme prosessen som den som ble utført av hovedkonverteren 320a blir gjentatt i hovedkonverteren 320c. Utdataene fra hovedkonverteren 320c blir XOR-behandlet med nøkkeldata matet ut fra nøkkelfordeleren 210 i XOR-kretsen 31b og deretter matet ut som C. Med CAMELLIA anvendes en eksklusiv logikkoperator tilveiebragt i underkonverteren 330 som 31b, i likhet med XOR-kretsen 30a. Med CAMELLIA utføres videre dataomforming med bruk av samme hovedkonverter 320 som de respektive hovedkonvertere 320a, 320b og 320c, og ved å gjenta samme prosess. Alternativt vil imidlertid hovedkonverterne 320a, 320b og 320c også kunne utgjøres av separate enheter med samme innvendige oppbygning. Det samme gjelder for underkonverterne 330a og 330b.

Merk her at i tilfellet der hovedkonverteren 320 er utstyrt med komponenter for å prosessere F-funksjonen for én runde og da utfører en seks-runders F-funksjonsbasert dataomforming som illustrert i figur 57, prosessen i hovedkonverteren 320 blir gjentatt seks ganger, og med det kompletteres den seks-runders F-funksjonsbaserte dataomformingen. Dette betyr, som kan sees i figur 1, at hovedkonverteren 320 fullfører den seks-runders F-funksjonsbaserte dataomformingen ved å gjenta passet gjennom loopkretsen fra hovedkonverteren 320 gjennom signalvelgeren 310, det aritmetiske registeret 350 og tilbake til hovedkonverteren 320 seks ganger. Som følge av dette krever den beslektede teknikken illustrert i figurene 56 og 60 "signalbanen for å mate inn data som er matet ut fra hovedkonverteren 320 til signalvelgeren 310".

I denne utførelsesformen har imidlertid underkonverteren 330 en overføringsfunksjon, som vil bli beskrevet senere, og derfor kan data matet ut fra hovedkonverteren 320 bli matet inn til signalvelgeren 310 ved at de overføres av underkonverteren 330. I dataomformingsanordningen ifølge denne utførelsesformen fjerner således bruken av "signalbanen for å mate inn data matet ut fra hovedkonverteren 320 til signalvelgeren 310 ved at de overføres av underkonverteren 330" behovet for "signalbanen for å mate inn data som er matet ut fra hovedkonverteren 320 til signalvelgeren 310".

Overordnet omforming - hovedkonverter 320.

Den innvendige oppbygningen av og virkemåten til hovedkonverteren 320 er beskrevet tidligere med henvisning til figurene 57 og 61.

Som nevnt tidligere kalles fremgangsmåten for ikke-lineær omforming i hovedkonverteren 320 beskrevet nedenfor FEISTEL. Spesifikt omfatter denne fremgangsmåten for ikke-lineær omforming det å dele opp innmatede data i øvre data bestående av de øvre bitene og nedre data bestående av de nedre bitene, ikke-lineært omforme én av de oppdelte øvre dataene og nedre dataene med bruk av F-funksjonen, generere data for innmating til F-funksjonen basert på den ene av de øvre dataene eller nedre dataene som er ikke-lineært omformet og den andre av de øvre dataene og de nedre dataene, dele opp de genererte dataene i øvre data og nedre data og omforme på nytt med bruk av F-funksjonen og ved å gjenta de ovenfornevnte prosessene.

Overordnet omforming - hovedkonverter 320 - F-funksjonsenhet 321.

En beskrivelse vil nå bli gitt av en innvendig oppbygning av og en virkemåte for F-funksjonsenheten 321 tilveiebragt i hovedkonverteren 320.

Figur 5 er et diagram som illustrerer en innvendig oppbygning av og en virkemåte for en F-funksjonsenhet 321.

Først blir innmatede data XOR-behandlet med en utvidet nøkkel i en XOR-krets 323, og deretter delt inn i åtte deler og matet inn til en S-funksjon 324. Den utvidede nøkkelen er definert som en kombinasjonsnøkkel av utnøkkelen (key KA) og mellomnøkkelen (key KL) som genereres av mellomnøkkel-generatoren 40 basert på den hemmelige nøkkelen. I CAMELLIA for en 128-bits nøkkel, er den utvidede nøkkelen 256 bit lang. Mellomnøkkelen (key KL) blir delt opp i en nøkkel key KLH bestående av de øvre bit og en nøkkel key KLL bestående av de nedre bit, og utnøkkelen (key KA) blir også delt opp i en nøkkel key KAH bestående av de øvre bit og en nøkkel key KAL bestående av de nedre bit. Deretter blir én av disse fire nøklene, skedulert av nøkkelfordeleren 210, matet inn til XOR-kretsen 323. S-funksjonen 324 er en sammensatt funksjon (S_1 til S_4) av en invers aritmetisk funksjon GAF (2^8) og en affin omforming, og utfører en byte-basert ikke-lineær omforming. De omformede og deretter utmatede dataene blir matet inn til en P-funksjon 325, skramblet av P-funksjonen 325 i en lineær omforming og deretter matet ut.

En beskrivelse vil nå bli gitt av en virkemåte i et tilfelle der en F-funksjonsenhet utgjøres av en del som prosesserer for halvparten av F-funksjonen (en 1/2F-funksjon).

Delen som prosesserer for halvparten av F-funksjonen i F-funksjonsenheten dannes av XOR-kretsen 323, fire S-bokser S_4 324e til S_1 324h, og omtrent halvparten av P-funksjonsenheten i figur 5. Med denne oppbygningen utføres først en halvpart av en F-funksjonsbasert dataomforming for én runde. Deretter gjentas den samme prosessen for å fullføre F-funksjonsprosessen for én runde. I den første prosessen anvendes en nøkkel og data som er rotasjonsskiftet én byte mot venstre eller høyre i nøkkelfordeleren 210 som beskrevet tidligere. Dette én-byte rotasjonsskiftet mot venstre eller høyre av nøkkelen og dataene kan gi samme resultat som å skifte en S-boks én byte uten å endre plasseringen av boksene S_1 til S_4. Mer spesifikt, som kan sees i figur 5, kan samme resultat som å plassere S-boksene i rekkefølgen S_2, S_3, S_4, S_1, som

representert ved S_1 324a til S_4 324d, oppnås ved å mate inn en nøkkel og data som er rotasjonsskiftet én byte uten å endre plasseringen av S-boksene S_1, S_2, S_3, S_4, som representert ved S_4 324e til S_1 324h.

Gjennom disse operasjonene kompletteres den én-rundes F-funksjons-prosessen over to sykluser.

Dataomformingsanordning med hovedkonverter 320 og underkonverter 330 anordnet i omvendt rekkefølge.

Figur 6 er et diagram som illustrerer en utførelse av og en virkemåte for en dataomformingsanordning der hovedkonverteren 320 og underkonverteren 330 i figur 1 er anordnet i omvendt rekkefølge.

Også i tilfellet der dataomformingsanordningen har hovedkonverteren 320 og underkonverteren 330 anordnet i omvendt rekkefølge anvender underkonverteren 330 overføringsfunksjonen for å overføre og å mate ut data til hovedkonverteren 320, akkurat som i dataomformingsanordningen illustrert i figur 1. Bruken av en slik signalbane gjør at hovedkonverteren 320 kan gjennomføre den seks-runders F-funksjonsbaserte dataomformingen. Følgelig er "signalbanen for å mate inn data matet ut fra underkonverteren 330 til signalvelgeren 310" overflødig.

Mellomnøkkelen (key KL) matet ut fra key KL-registeret 240 blir ikke matet direkte inn til signalvelgeren 310, men blir i stedet matet inn til underkonverteren 330 via nøkkelfordeleren 210 med bruk av signalbanen fra key KL-registeret 240 til nøkkelfordeleren 210. Underkonverteren 330, ved mottak av en nøkkel, overfører den innmatede nøkkelen til hovedkonverteren 320 ved hjelp av overføringsfunksjonen.

Overføringsoperasjonen som således utføres av underkonverteren 330 fjerner behovet for de to databanene assosiert med "signalbanen for å mate inn mellomnøkkelen (key KL) matet ut fra key KL-registeret 240 til signalvelgeren 310" og "signalbanen for å mate inn data matet ut fra hovedkonverteren 320 til signalvelgeren 310" eller "signalbanen for å mate inn data matet ut fra underkonverteren 330 til signalvelgeren 310" illustrert i figurene 56 og 60.

Dataomformingsanordning med hovedkonverter 320 og underkonverter 330 anordnet parallelt.

Figur 7 er et diagram som illustrerer en dataomformingsanordning som skiller seg fra de i figurene 1 og 6 ved at hovedkonverteren 320 og underkonverteren 330 er anordnet parallelt, og ved at det er tilveiebragt en signalvelger 340 som velger ut ett utsignal fra to innmatede signaler.

Andre elementer enn de angitt over er de samme som de i figurene 1 og 6.

Dataomformingsanordningen som er innrettet på denne måten, med hovedkonverteren 320 og underkonverteren 330 anordnet parallelt, må ha en signalvelger 340 for å velge ut ett av signalene som mates ut fra hovedkonverteren 320 og underkonverteren 330. Følgelig mottar hovedkonverteren 320 og underkonverteren 330 et signal som er valgt av signalvelgeren 310 mellom et signal som er valgt av signalvelgeren 340 og sendt gjennom det aritmetiske registeret 350, og P (klartekst eller et kryptogram).

Videre blir ikke mellomnøkkelen (key KL) matet ut fra key KL-registeret 240 matet direkte inn til signalvelgeren 310 på det tidspunktet utnøkkelen (key KA) blir generert. Mellomnøkkelen (key KL) blir matet inn til underkonverteren 330 via nøkkelfordeleren 210 ved hjelp av en signalbane fra key KL-registeret 240 til nøkkelfordeleren 210. Underkonverteren 330, når den mottar nøkkelen, overfører den mottatte nøkkelen til hovedkonverteren 320 ved hjelp av overføringsfunksjonen. På denne måten kan behovet for "signalbanen for å mate inn mellomnøkkelen matet ut fra key KL-registeret 240 til signalvelgeren 310" fjernes.

Videre kan også behovet for de to databanene assosiert med "signalbanen for å mate inn data matet ut fra hovedkonverteren 320 til signalvelgeren 310" eller "signalbanen for å mate inn data matet ut fra underkonverteren 330 til signalvelgeren 310" fjernes.

Innvendig oppbygning av mellomnøkkel-generatoren 40

En beskrivelse vil nå bli gitt av en utførelse av 6-1 KL-signalvelgeren 220 og 6-1 KA-signalvelgeren 230 i mellomnøkkel-generatoren 40.

Figur 8 er et diagram som illustrerer en utførelse av 6-1 KL-signalvelgeren 220 og 6-1 KA-signalvelgeren 230 i mellomnøkkel-generatoren 40.

Mellomnøkkelen (key KL) som holdes i key KL-registeret 240 i mellomnøkkel-generatoren 40, blir matet ut til nøkkelfordeleren 210, og blir også matet inn til 6-1 KL-signalvelgeren 220 igjen. 6-1 KL-signalvelgeren 220 omfatter en 6-1-signalvelger 221.

I 6-1 KL-signalvelgeren 220 blir den innmatede mellomnøkkelen (key KL) og også fire signaler frembragt gjennom rotasjonsskift av mellomnøkkelen (key KL) med fire forskjellige, vilkårlige tall matet inn til 6-1-signalvelgeren 221. De fire signalene som mates inn til denne kan for eksempel være frembragt ved å rotasjonsskifte mellomnøkkelen 17 og 15 bit henholdsvis mot venstre og høyre, som ikke er vist i figuren. De seks signalene omfattende mellomnøkkelen (key KL), de fire rotasjonsskiftede signalene og den hemmelige nøkkelen behandles som seks innsignaler, og 6-1-signalvelgeren 221 velger ett ut utsignal fra de seks innsignalene og legger det valgte utsignalet i key KL-registeret som en ny mellomnøkkel (key KL).

Fremgangsmåten for å generere en ny utnøkkel (key KA) basert på utnøkkelen (key KA) er den samme som for å generere en ny mellomnøkkel (key KL) basert på mellomnøkkelen (key KL).

Figur 9 er et diagram som illustrerer en annen utførelse av mellomnøkkel-generatoren 40.

Figur 9, til forskjell fra figur 8, viser deling av en signalvelger representert ved en 4-1-signalvelger 223. Spesifikt blir mellomnøkkelen (key KL) matet ut fra key KL-registeret 240 og utnøkkelen (key KA) matet ut fra key KA-registeret 250 matet inn til en 2-1-signalvelger 224. 2-1-signalvelgeren 224 velger én av de to nøklene, genererer deretter fire signaler ved å rotasjonsskifte den valgte nøkkelen med fire forskjellige tall, og mater så ut de fire signalene til 4-1-signalvelgeren 223. 4-1-signalvelgeren 223 velger ett av de fire signalene og mater deretter ut det valgte signalet til en 3-1 KL-signalvelger 222 eller en 3-1 KA-signalvelger 232.

3-1 KL-signalvelgeren 222 velger ut én nøkkel av nøkkelen valgt av 4-1-signalvelgeren 223, den hemmelige nøkkelen og mellomnøkkelen (key KL) som var inneholdt i key KL-registeret 240, og key KL-registeret 240 mottar den valgte nøkkelen som en ny mellomnøkkel.

Tilsvarende velger 3-1 KA-signalvelgeren 232 ut én nøkkel av nøkkelen valgt av 4-1-signalvelgeren 223, den genererte utnøkkelen (key KA) og ut-

nøkkelen (key KA) som var inneholdt i key KA-registeret 250, og key KA-registeret 250 mottar den valgte nøkkelen som en ny utnøkkel (key KA).

Til forskjell fra utførelsen vist i figur 8, der det er nødvendig med ti 2-1-signalvelgerenheter, krever utførelsen vist i figur 9 kun åtte 2-1-signalvelgerenheter. Sammenliknet med utførelsen av mellomnøkkel-generatoren 40 illustrert i figur 8 kan derfor utførelsen vist i figur 9 spare to 2-1-signalvelgerenheter. På denne måten kan kretsens størrelse reduseres.

Merk at den utførelsen av mellomnøkkel-generatoren 40 som er vist i figur 8 vil kunne anvendes i dataomformingsanordningen i enhver utførelsesform av foreliggende oppfinnelse. Likeledes vil den utførelsen av mellomnøkkel-generatoren 40 som er vist i figur 9 også kunne anvendes i dataomformingsanordningen i enhver utførelsesform av foreliggende oppfinnelse.

Videre kan den utførelsen av mellomnøkkel-generatoren 40 som er vist i figur 51 og som beskrives senere anvendes i dataomformingsanordningen i enhver utførelsesform av foreliggende oppfinnelse.

Underordnet omforming - Underkonverter 330.

En beskrivelse vil nå bli gitt av en utførelse av og en virkemåte for underkonverteren 330.

Det vil her bli gitt en beskrivelse av tilfellet der minst én av datakonverterenheten 50 og datainverterenheten 70 har nøkkeloverføringsfunksjonen ifølge denne utførelsesformen.

Underordnet omforming - Underkonverter 330 - Datakonverterenhet 50 med nøkkeloverføringsfunksjon.

Figur 10 er et diagram som illustrerer en utførelse av og en virkemåte for underkonverteren 330.

I forhold til utførelsen av datakonverterenheten 50 og datainverterenheten 70 fra kjent teknikk beskrevet i forbindelse med figur 58, omfatter disse enhetene i denne utførelsesformen i tillegg et overføringssignal for å overføre en nøkkel eller data samt kretser i tilknytning til overføringssignalet.

I figur 10 har datakonverterenheten 50 funksjonen å overføre en innmatet nøkkel.

Spesifikt blir et overføringssignal for å overføre en nøkkel matet inn til datakonverterenheten 50. Datakonverterenheten 50, når den mottar overføringssignalet, overfører den mottatte nøkkelen basert på overføringssignalet.

Mer spesifikt kontrolleres overføringssignalene av en styringsenhet 5. Ved overføring av en nøkkel mater styringsenheten ut et FL-nøkkeloverføringssignal og et FL-maskesignal. Datakonverterenheten 50 mottar FL-nøkkeloverføringssignalet og FL-maskesignalet matet ut fra styringsenheten 5.

Det vil nå bli gitt konkrete beskrivelser av en nøkkeloverføringsprosess som utføres av datakonverterenheten 50 med bruk av disse overføringssignalene.

Ved overføring av en nøkkel blir FL-nøkkeloverføringssignalet satt til 0 og matet inn til en AND-krets 51. AND-kretsen 51 mottar også mål-data som skal krypteres/dekrypteres.

Siden FL-nøkkeloverføringssignalet har verdien 0, blir inndataene undertrykket av AND-enheten i AND-kretsen 51 og dermed nullet ut. Med andre ord vil utdataene fra AND-kretsen 51 være 0 uavhengig av inndataenes verdi.

De øvre eller mest signifikante bitene av dataene tilordnet verdien 0 matet ut fra AND-kretsen 51 blir matet inn til en OR-krets 53, og de nedre eller minst signifikante bitene blir matet inn til XOR-kretsen 55.

I mellomtiden blir FL-maskesignalet matet inn til en NOT-krets 52. I tilfellet med overføring av en nøkkel setter styringsenheten 5 FL-maskesignalet til 0, slik at utsignalet fra NOT-kretsen 52 er 1. Som følge av dette får utsignalet fra OR-kretsen 53, som således mottar signaler med verdi 1 og 0, verdien 1.

AND-kretsen 54 mottar 1 matet ut fra OR-kretsen 53 samt informasjon fra nøkkelen key 1, slik at utdataene fra AND-kretsen 54 alltid er key 1.

Nøkkelen key 1 matet ut fra AND-kretsen 54 blir rotasjonsskiftet én bit mot venstre, og deretter matet inn til XOR-kretsen 55. Nøkkelen key 1 har på forhånd blitt rotasjonsskiftet én bit mot høyre i nøkkelfordeleren 210, som kan sees i figur 3. Som følge av dette kan nøkkelen key 1 matet ut fra AND-kretsen 54 tilbakeføres til sin opprinnelige verdi for overføring ved å bli rotasjonsskiftet én bit mot venstre.

XOR-kretsen 55 mottar de nedre bitene, tilordnet verdien 0, matet ut fra AND-kretsen 51, slik at den aritmetiske operasjonen i AND-kretsen 55 mater ut nøkkelen key 1 som den er. Dette blir de nedre bitene av utsignalet.

Datakonverterenheten 50 kan således mate ut nøkkelen key 1 som den er som utsignalet basert på FL-nøkkeloverføringssignalet og FL-maskesignalet.

Tilsvarende, med FL-nøkkeloverføringssignalet og FL-maskesignalet, blir nøkkelen key 2 overført som den er som utsignalet. Denne virkemåten vil bli beskrevet nedenfor.

FL-maskesignalet har verdien 0 som nevnt tidligere. Derfor mottar en AND-krets 58 0 og nøkkelen key 1 matet ut fra XOR-kretsen 55, og mater alltid ut 0.

OR-kretsen 57 mottar nøkkelen key 2 og 0, og dens utsignal vil således alltid være key 2.

Nøkkelen key 2 blir matet inn til XOR-kretsen 56, der den blir XOR-behandlet med 0 fra de øvre dataene matet ut fra AND-kretsen 51, slik at utmatingen fra XOR-kretsen 56 alltid vil være key 2. Denne blir de øvre bitene av utsignalet.

Følgelig blir FL-nøkkeloverføringssignalet og FL-maskesignalet matet inn og nøkkelen key 1 og nøkkelen key 2 kan bli overført som de er. Merk at selv om styringsenheten 5 for å kontrollere FL-nøkkeloverføringssignalet og FL-maskesignalet, som begge er overføringssignaler, ikke er vist i figurene 11, 12 og 14 til 33, overføringssignalene vil være kontrollert av styringsenheten 5 på samme måte som illustrert i figur 10.

Underordnet omforming - Underkonverter 330 - Datainverterenhet 70 med nøkkeloverføringsfunksjon.

En beskrivelse vil nå bli gitt av tilfellet der datainverterenheten 70 er utstyrt med funksjonen for å overføre en innmatet nøkkel.

Figur 11 er et diagram som illustrerer tilfellet der datainverterenheten 70 har nøkkeloverføringsfunksjonen.

En AND-krets 71 mottar et FL^{-1}-nøkkeloverføringssignal og data.

I likhet med det ovenfornevnte FL-nøkkeloverføringssignalet, har FL^{-1}-nøkkeloverføringssignalet verdien 0, og data som mates inn til AND-kretsen 71 blir derfor undertrykket og således nullet ut, slik at utdataene fra AND-kretsen 71 alltid vil ha verdien 0.

I likhet med det ovenfornevnte FL-nøkkeloverføringssignalet har FL^{-1}-nøkkeloverføringssignalet verdien 0, og derfor er begge signalene som mates

inn til en AND-krets 73 null, slik at utdataene fra AND-kretsen 73 alltid vil ha verdien 0.

OR-kretsen 74, som mottar utdataene fra AND-kretsen 73, dvs. verdien 0 og nøkkelen key 3, vil mate ut key 3.

XOR-kretsen 75 mottar de øvre bitene, dvs. verdien 0, av utdataene fra AND-kretsen 71, og mater derfor ut key 3. Denne blir de øvre bitene av utsignalet.

En OR-krets 78 mottar en verdi 1, som er verdien til FL^{-1}-maskesignalet (0) invertert i en NOT-krets 72, og nøkkelen key 3, og mater derfor ut 1. AND-kretsen 77 mottar utdataene (1) fra OR-kretsen 78 og nøkkelen key 4, og mater derfor ut key 4. Nøkkelen key 4 blir rotasjonsskiftet én bit mot venstre og deretter matet inn til XOR-kretsen 76. Som over har nøkkelen key 4 på forhånd blitt rotasjonsskiftet én bit mot høyre i nøkkelfordeleren 210, og deretter matet inn til datainverterenheten 70. Som følge av dette vil nøkkelen key 4 gjenopprette sin opprinnelige verdi ved å bli rotasjonsskiftet én bit mot venstre her.

XOR-kretsen 76 som mottar de nedre bitene, dvs. verdien 0, av utdataene fra AND-kretsen 71 og nøkkelen key 4, mater ut key 4. Denne blir de nedre bitene av utdataene.

Datainverterenheten 70 kan således mate ut en innmatet nøkkel (key 3, key 4) som den er ved mottak av overføringssignalene, nemlig FL^{-1}-nøkkeloverføringssignalet og FL^{-1}-maskesignalet.

Underordnet omforming - Underkonverter 330 - Datakonverterenhet 50 og datainverterenhet 70, begge med nøkkeloverføringsfunksjon.

En beskrivelse vil nå bli gitt av tilfellet der både datakonverterenheten 50 og datainverterenheten 70 er utstyrt med funksjonen for å overføre en innmatet nøkkel.

Figur 12 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 og datainverterenheten 70 begge er utstyrt med nøkkeloverføringsfunksjonen.

Oppbygningen av og virkemåten til datakonverterenheten 50 er tilsvarende som for datakonverterenheten 50 illustrert i figur 10, og oppbygningen av og virkemåten til datainverterenheten 70 er tilsvarende som for datainverterenheten 70 illustrert i figur 11. Disse vil derfor ikke bli beskrevet i detalj her.

Det at minst én av datakonverterenheten 50 og datainverterenheten 70 er utstyrt med nøkkeloverføringsfunksjonen fjerner behovet for signalbanen for å overføre en nøkkel fra key KL-registeret 240 til signalvelgeren 310, som illustrert i figurene 56 og 60, slik at nøkkelen kan mates inn til underkonverteren 330 fra key KL-registeret 240 via nøkkelfordeleren 210. I tillegg gjør overføringssignalet for å overføre en nøkkel som mates inn til underkonverteren 330 det mulig for underkonverteren 330 å overføre nøkkelen til signalvelgeren 330.

Ved å muliggjøre nøkkeloverføring med bruk av denne signalbanen er det mulig å redusere det totale antallet signalvelgere i dataomformingsanordningen. Mer spesifikt deles i dataomformingsanordningen ifølge denne utførelsesformen funksjonen ved å generere den utvidede nøkkelen i mellomnøkkel-generatoren 40 og ved å utføre dataomforming i hovedkonverteren 320 og underkonverteren 330 som illustrert i figur 2 for å realisere en kompakt dataomformingsanordning. Videre kan bruken av "signalbanen for å overføre mellomnøkkelen (key KL) fra key KL-registeret 240 til signalvelgeren 310 via nøkkelfordeleren 210 i underkonverteren 330 og deretter til hovedkonverteren 320 via det aritmetiske registeret 350" i denne utførelsesformen redusere økningen av antallet signalvelgere i forhold til bruken av "signalbanen for å overføre mellomnøkkelen (key KL) fra key KL-registeret 240 til signalvelgeren 310 og deretter til hovedkonverteren 320 via det aritmetiske registeret 350", som illustrert i figurene 56 og 60.

På denne måten å redusere økningen av antallet signalvelgere i datascrambleren i dataomformingsanordningen for blokk-basert kryptering, og dermed redusere det totale antallet logikkporter i kretsene, gjør det mulig å redusere kretsbrikkenes samlede størrelse og strømforbruket. Dataomformingsanordningen for blokk-basert kryptering ifølge denne utførelsesformen kan således anvendes også i mobile anordninger, så som mobiltelefoner, hvor det er et sterkt ønske om å redusere størrelsen og å oppnå et lavt strømforbruk.

Som kan sees i figurene 10 til 33 kan de innmatede nøklene enten være forskjellige eller de samme. FL-nøkkeloverføringssignalet og FL^{-1}-nøkkeloverføringssignalet kan også være det samme signalet. FL-maskesignalet og FL^{-1}-maskesignalet kan også være det samme signalet.

Utførelsesform 2

I denne utførelsesformen beskrives tilfellet der minst én av datakonverterenheten 50 og datainverterenheten 70 er utstyrt med dataoverføringsfunksjonen.

Underordnet omforming - Underkonverter 330 - Datakonverterenhet 50 med dataoverføringsfunksjon.

I denne utførelsesformen beskrives tilfellet der underkonverteren 330 er utstyrt med dataoverføringsfunksjonen.

Figur 13 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 er utstyrt med dataoverføringsfunksjonen.

Styringsenheten 5 mater inn FL-dataoverføringssignalet til datakonverterenheten 50 som et signal om å overføre data. FL-dataoverføringssignalet som mates inn til datakonverterenheten 50 har verdien 0. Dette signalet blir matet inn til en AND-krets 59 og en AND-krets 60.

En AND-krets 54 mottar de øvre bitene av innmatede data og nøkkelen key 1. Utdataene fra AND-kretsen 54 er ikke gitt på forhånd, men avhenger av verdien til de innmatede dataene. Utsignalet fra AND-kretsen 60 på den annen side, har alltid verdien 0, uavhengig av verdien til utsignalet fra AND-kretsen 54, også med inndata oppnådd gjennom et én-bits rotasjonsskift mot venstre av de utmatede dataene, siden det andre innsignalet, FL-dataoverføringssignalet, har verdien 0. Utdataene, dvs. verdien 0, fra AND-kretsen 60 blir matet inn til XOR-kretsen 55, hvor de innmatede dataene og de nedre bitene blir XOR-behandlet. Siden utmatingen fra AND-kretsen 60 er 0, blir de nedre bitene av de innmatede dataene matet ut fra XOR-kretsen 55 som den nedre delen av utsignalet.

I mellomtiden blir utdataene fra XOR-kretsen 55 og en nøkkel matet inn til OR-kretsen 57 som innsignaler. Merk her at utsignalet fra OR-kretsen 57 ikke er gitt på forhånd, men at FL-dataoverføringssignalet alltid har verdien 0, slik at utsignalet fra AND-kretsen 59 har verdien 0. I XOR-kretsen 56 blir de øvre bitene av innmatede data og utdataene, dvs. 0, fra AND-kretsen 59 XOR-behandlet. Følgelig blir de øvre bitene av de innmatede dataene matet ut som de øvre dataene av et utsignal.

Datakonverterenheten 50, når den mottar FL-dataoverføringssignalet som overføringssignalet, kan således mate ut de mottatte dataene som de er uavhengig av den innmatede nøkkelen.

Merk at oppbygningen til datainverterenheten 70 i figur 13 er tilsvarende den til datainverterenheten 70 illustrert i figur 57, og denne vil derfor ikke bli beskrevet i detalj her.

Underordnet omforming - Underkonverter 330 - Datainverterenhet 70 med dataoverføringsfunksjon.

Figur 14 er et diagram som illustrerer en utførelse av underkonverteren 330 der datainverterenheten 70 har dataoverføringsfunksjonen.

Datainverterenheten 70 mottar FL^{-1}-dataoverføringssignalet for å overføre data. Ved overføring av data har FL^{-1}-dataoverføringssignalet verdien 0, slik at en AND-krets 79 mater ut verdien 0 uavhengig av verdien til et utsignal fra OR-kretsen 74. XOR-kretsen 75 mater derfor ut de øvre bitene av inmatede data som de er som de øvre dataene av utsignalet.

FL^{-1}-dataoverføringssignalet blir matet inn til en AND-krets 80, slik at utsignalet fra AND-kretsen 80 har verdien 0 uavhengig av verdien til utgangssignalet fra AND-kretsen 77. De nedre bitene av inmatede data blir således matet ut som de er fra XOR-kretsen 76 som de nedre data av utsignalet.

Datainverterenheten 70 kan således overføre data som de er som utsignalet.

Underordnet omforming - Underkonverter 330 - Datakonverterenhet 50 og datainverterenhet 70 begge med dataoverføringsfunksjon.

Figur 15 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 og datainverterenheten 70 begge er utstyrt med dataoverføringsfunksjonen.

Datakonverterenheten 50 har samme oppbygning som datakonverterenheten 50 i figur 13, og datainverterenheten 70 har samme oppbygning som datainverterenheten 70 i figur 14. Datakonverterenheten 50 og datainverterenheten 70 i figur 15 er derfor i stand til å mate ut sine respektive inndata som de er som utdata.

Det at underkonverteren 330 er i stand til å overføre innmatede data som de er til signalvelgeren 310 kan fjerne behovet for signalbanen for å overføre utdata fra hovedkonverteren 320 til signalvelgeren 310 vist i figur 56.

Som kan sees i figur 60, under kryptering/dekryptering av data i tilfellet der hovedkonverteren 320 har F-funksjonen som prosesserer for mindre enn én runde, vil mellomdataene være inneholdt i det aritmetiske registeret 350 for en gitt tidsperiode for at hovedkonverteren 320 skal kunne gjennomføre den F-funksjonsbaserte, ikke-lineære omformingen for mindre enn én runde, som beskrevet tidligere. Dette betyr at hovedkonverteren 320 trenger en egen loopkrets svarende til loopkretsen i figur 60 for å mate ut mellomdataene matet ut fra hovedkonverteren 320 til det aritmetiske registeret 350 via signalvelgeren 310.

Bruken av dataoverføringsfunksjonen til underkonverteren 330 i denne utførelsesformen på den annen side, kan fjerne behovet for den ovenfornevnte loopkretsen. Mer spesifikt blir mellomdata matet ut fra hovedkonverteren 320 overført av underkonverteren 330 og matet inn til signalvelgeren 310. Signalvelgeren 310 velger de mottatte mellomdataene, og med det overføres mellomdataene til hovedkonverteren 320.

Bruken av denne databanen gjør det mulig å redusere antallet innsignaler til signalvelgeren 310 sammenliknet med antallet innsignaler til signalvelgeren 310 vist i figurene 56 og 60. Som følge av dette kan antallet signalvelgere reduseres.

Tilsvarende fjerner dataomformingsanordningene i figurene 6 og 7 behovet for signalbanen fra hovedkonverteren 320 til signalvelgeren 310, slik at anordningen kan gjøres mer kompakt. I tillegg gjør reduksjonen av antallet signalvelgere det mulig å oppnå et lavt strømforbruk.

Merk at FL-dataoverføringssignalet og FL^{-1}-dataoverføringssignalet kan være samme signal.

Utførelsesform 3

I denne utførelsesformen beskrives tilfellet der minst én av datakonverterenheten 50 og datainverterenheten 70 er utstyrt med nøkkeloverføringsfunksjonen og dataoverføringsfunksjonen.

Underordnet omforming - Underkonverter 330 - Datakonverterenhet 50 med nøkkeloverføringsfunksjon og dataoverføringsfunksjon og datainverterenhet 70 med nøkkeloverføringsfunksjon.

Figur 16 er et diagram som illustrerer en innvendig oppbygning av underkonverteren 330 der datakonverterenheten 50 er utstyrt med nøkkeloverføringsfunksjonen og dataoverføringsfunksjonen og datainverterenheten 70 er utstyrt med nøkkeloverføringsfunksjonen.

Oppbygningen av og virkemåten til datainverterenheten 70 er tilsvarende som for datainverterenheten 70 i figur 11 med nøkkeloverføringsfunksjonen, og vil derfor ikke bli beskrevet her.

Oppbygningen av og virkemåten til datakonverterenheten 50 tilsvarer kombinasjonen av de til datakonverterenheten 50 i figur 10 med nøkkeloverføringsfunksjonen og de til datakonverterenheten 50 i figur 13 med dataoverføringsfunksjonen, og vil derfor ikke bli beskrevet her.

I datakonverterenheten 50 tjener FL-nøkkeloverføringssignalet til å undertrykke og således nulle ut innmatede data, og FL-maskesignalet tjener til å slippe gjennom en innmatet nøkkel uforandret.

FL-dataoverføringssignalet tjener til å nulle ut en innmatet nøkkel for å slippe gjennom data uforandret.

Følgelig kan det ikke overføres data i tilfellet der både FL-nøkkeloverføringssignalet og FL-maskesignalet har verdien 0 som overføringssignal for å overføre en nøkkel, slik at FL-dataoverføringssignalet ikke kan ha 0 som et overføringssignal for å overføre data. Tilsvarende kan det ikke overføres en nøkkel i tilfellet der FL-dataoverføringssignalet har verdien 0 som overføringssignal for å overføre data, slik at FL-nøkkeloverføringssignalet og FL-maskesignalet ikke kan ha 0 som overføringssignal for å overføre en nøkkel.

Dersom ingen av FL-nøkkeloverføringssignalet, FL-maskesignalet, FL-dataoverføringssignalet, et FL^{-1}-nøkkeloverføringssignal eller et FL^{-1}-maskesignal har verdien 0 som overføringssignal, utfører da datakonverterenheten 50 og datainverterenheten 70 en lineær omforming av innmatede data, som er dens formål.

Det vil først bli gitt en beskrivelse av en virkemåte for datakonverterenheten 50 for å overføre en nøkkel.

Datakonverterenheten 50 mottar 0 som FL-nøkkeloverføringssignal og 0 som FL-maskesignal. Siden det ikke skal overføres data, forblir FL-data-overføringssignalet uforandret og har verdien 1.

Først vil AND-kretsen 51 undertrykke og således nulle ut data som følge av FL-nøkkeloverføringssignalet. Nøkkelen key 1 passerer uforandret gjennom AND-kretsen 54, og blir deretter rotasjonsskiftet én bit mot venstre og matet inn til AND-kretsen 60. Siden FL-dataoverføringssignalet har verdien 1 passerer nøkkelen key 1 uforandret gjennom AND-kretsen 60. I XOR-kretsen 55 blir nøkkelen key 1 XOR-behandlet med de nedre bitene av utdataene fra AND-kretsen 51, som har verdien 0. Nøkkelen key 1 blir matet ut som de nedre data av utsignalet.

Nøkkelen key 2 passerer uforandret gjennom OR-kretsen 57 sammen med verdien 0, som er matet ut fra AND-kretsen 58, passerer uforandret gjennom AND-kretsen 59 sammen med FL-dataoverføringssignalet, blir deretter XOR-behandlet med 0, som er de nedre bitene av utdataene matet ut fra AND-kretsen 51 i XOR-kretsen 56, og passerer dermed uforandret også gjennom XOR-kretsen 56, og blir de øvre dataene av utsignalet. Datakonverterenheten 50 er således i stand til å overføre en nøkkel (key 1, key 2) uforandret.

En beskrivelse vil nå bli gitt av en virkemåte for datakonverterenheten 50 for å overføre data.

FL-dataoverføringssignalet har verdien 0. FL-nøkkeloverføringssignalet og FL-maskesignalet har verdien 1.

AND-kretsen 51 slipper dataene gjennom uforandret, og de nedre bitene av disse dataene blir matet inn til XOR-kretsen 55. AND-kretsen 60 mottar 0 fra FL-dataoverføringssignalet, slik at AND-kretsen 60 mater ut 0. De nedre bitene av dataene som mates inn til XOR-kretsen 55 passerer gjennom XOR-kretsen 55, og blir deretter matet ut som de nedre dataene av utsignalet.

Tilsvarende mater AND-kretsen 59 ut 0 siden FL-dataoverføringssignalet har verdien 0. De nedre bitene av dataene som mates inn til XOR-kretsen 56 passerer gjennom XOR-kretsen 56, og blir deretter matet ut som de øvre dataene av utsignalet.

På denne måten kan datakonverterenheten 50 overføre data som de er.

Nøkkeloverføringssignalet, slik som FL-nøkkeloverføringssignalet og FL^{-1}-nøkkeloverføringssignalet, og maskesignalet, slik som FL-maskesignalet og

FL^{-1}-maskesignalet, overfører således en nøkkel, og FL-dataoverføringssignalet overfører data.

Underordnet omforming - Underkonverter 330 - Datakonverterenhet 50 med nøkkeloverføringsfunksjon og datainverterenhet 70 med nøkkeloverførings-funksjon og dataoverføringsfunksjon.

Figur 17 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 er utstyrt med nøkkeloverføringsfunksjonen og datainverterenheten 70 er utstyrt med nøkkeloverføringsfunksjonen og data-overføringsfunksjonen.

Oppbygningen av og virkemåten til datakonverterenheten 50 er tilsvarende som datakonverterenheten 50 i figur 10 med nøkkeloverførings-funksjonen, og vil derfor ikke bli beskrevet her.

Datainverterenheten 70 fungerer på samme måte som datakonverter-enheten 50 i figur 16. Virkemåten til datainverterenheten 70 er således beskrevet over, og vil derfor ikke bli gjentatt her i detalj.

Underordnet omforming - Underkonverter 330 - Datakonverterenhet 50 og data-inverterenhet 70 begge med nøkkeloverføringsfunksjon og dataoverførings-funksjon.

Figur 18 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 og datainverterenheten 70 begge er utstyrt med nøkkeloverføringsfunksjonen og dataoverføringsfunksjonen.

Overføringsoperasjonene som utføres av datakonverterenheten 50 og datainverterenheten 70 er beskrevet tidligere, og vil ikke bli gjentatt her. I denne utførelsesformen er både datakonverterenheten 50 og datainverterenheten 70 utstyrt med både nøkkeloverføringsfunksjonen og dataoverføringsfunksjonen, slik at dataomformingsanordningen kan utføre en avansert prosess for å over-føre en nøkkel og data.

Underordnet omforming - Underkonverter 330 - Datakonverterenhet 50 med nøkkeloverføringsfunksjon og dataoverføringsfunksjon.

Figur 19 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 er utstyrt med nøkkeloverføringsfunksjonen

og dataoverføringsfunksjonen, mens datainverterenheten 70 ikke har noen av disse overføringsfunksjonene.

Virkemåten til datakonverterenheten 50 og datainverterenheten 70 er beskrevet tidligere, og vil ikke bli gjentatt her.

Underordnet omforming - Underkonverter 330 - Datainverterenhet 70 med nøkkeloverføringsfunksjon og dataoverføringsfunksjon.

Figur 20 er et diagram som illustrerer en utførelse av underkonverteren 330 der datainverterenheten 70 er utstyrt med nøkkeloverføringsfunksjonen og dataoverføringsfunksjonen, mens datakonverterenheten 50 ikke har noen av disse overføringsfunksjonene.

Detaljene ved virkemåten til datakonverterenheten 50 og datainverterenheten 70 er beskrevet tidligere, og vil ikke bli gjentatt her.

Underordnet omforming - Underkonverter 330 - Datakonverterenhet 50 med nøkkeloverføringsfunksjon og dataoverføringsfunksjon og datainverterenhet 70 med dataoverføringsfunksjon.

Figur 21 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 er utstyrt med både nøkkeloverføringsfunksjonen og dataoverføringsfunksjonen, mens datainverterenheten 70 er utstyrt med dataoverføringsfunksjonen.

Virkemåten til datakonverterenheten 50 og datainverterenheten 70 er beskrevet tidligere, og vil ikke bli gjentatt her.

Underordnet omforming - Underkonverter 330 - Datainverterenhet 70 med nøkkeloverføringsfunksjon og dataoverføringsfunksjon og datakonverterenhet 50 med dataoverføringsfunksjon.

Figur 22 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 er utstyrt med dataoverføringsfunksjonen og datainverterenheten 70 er utstyrt med både nøkkeloverføringsfunksjonen og dataoverføringsfunksjonen.

Virkemåten til datakonverterenheten 50 og datainverterenheten 70 er beskrevet tidligere, og vil ikke bli gjentatt her.

Underordnet omforming - Underkonverter 330 - Datakonverterenhet 50 med dataoverføringsfunksjon og datainverterenhet 70 med nøkkeloverførings-funksjon.

Figur 23 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 er utstyrt med dataoverføringsfunksjonen og datainverterenheten 70 er utstyrt med nøkkeloverføringsfunksjonen.

De respektive virkemåtene til disse er beskrevet tidligere, og vil ikke bli gjentatt her.

Underordnet omforming - Underkonverter 330 - Datakonverterenhet 50 med nøkkeloverføringsfunksjon og datainverterenhet 70 med dataoverførings-funksjonen.

Figur 24 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 er utstyrt med nøkkeloverføringsfunksjonen og datainverterenheten 70 er utstyrt med dataoverføringsfunksjonen.

De respektive virkemåtene til disse er beskrevet tidligere, og vil ikke bli gjentatt her.

Underordnet omforming - Underkonverter 330 - Datainverterenhet 70 og data-konverterenhet 50 koplet i serie - Datakonverterenhet 50 og datainverterenhet 70 begge med nøkkeloverføringsfunksjon.

Figur 25 er et diagram som illustrerer en utførelse der datainverter-enheten 70 og datakonverterenheten 50 er koplet i serie, og der både data-inverterenheten 70 og datakonverterenheten 50 er utstyrt med dataoverførings-funksjonen.

De respektive virkemåtene til disse er beskrevet tidligere, og vil ikke bli gjentatt her.

I den utførelsen som er illustrert i figur 25 blir data overført av data-inverterenheten 70 matet inn til datakonverterenheten 50, og deretter matet ut som utsignal som videresendes av datakonverterenheten 50.

Det at datainverterenheten 70 og datakonverterenheten 50 er anordnet og koplet serielt gjør at den lineære dataomformingen kan bli utført ikke bare av datakonverterenheten 50 og datainverterenheten 70 sammen, men også av datainverterenheten 70 alene eller av datakonverterenheten 50 alene. Mer

spesifikt kan data som er lineært omformet av datainverterenheten 70 bli matet inn til datakonverterenheten 50, hvorfra de mottatte dataene blir overført uten at det utføres noen lineær omforming. Det er også mulig at datainverterenheten 70 overfører mottatte data til datakonverterenheten 50, hvorpå datakonverterenheten 50 alene besørger lineær omforming.

Følgelig er dette den utførelsen som gjelder i tilfellet der data skal omformes av datakonverterenheten 50 alene eller av datainverterenheten 70 alene. Det samme resutatet kan oppnås med underkonverterne 330 illustrert i figurene 26 til 30, som beskrives senere.

Underordnet omforming - Underkonverter 330 - Datakonverterenhet 50 og datainverterenhet 70 koplet i serie - Datakonverterenhet 50 og datainverterenhet 70 begge med dataoverføringsfunksjon.

Figur 26 illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 og datainverterenheten 50 i figur 25 har byttet plass.

Virkemåten til og resultatet av disse er tilsvarende som for underkonverteren 330 i figur 26, og vil derfor ikke bli beskrevet her.

Underordnet omforming - Underkonverter 330 - Datakonverterenhet 50 og datainverterenhet 70 koplet i serie - Datakonverterenhet 50 med nøkkeloverføringsfunksjon og dataoverføringsfunksjon og datainverterenhet 70 med dataoverføringsfunksjon.

Figur 27 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 i figur 26 også er gitt nøkkeloverføringsfunksjonen.

Den innvendige oppbygningen av og virkemåten til datakonverterenheten 50 og datainverterenheten 70 er beskrevet tidligere, og vil ikke bli gjentatt her.

Underordnet omforming - Underkonverter 330 - Datainverterenhet 70 og datakonverterenhet 50 koplet i serie - Datakonverterenhet 50 med nøkkeloverføringsfunksjon og dataoverføringsfunksjon og datainverterenhet 70 med dataoverføringsfunksjon.

Figur 28 er et diagram som illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 og datainverterenheten 70 har byttet plass.

Oppbygningen og virkemåten til datakonverterenheten 50 og datainverterenheten 70 er beskrevet tidligere, og vil ikke bli gjentatt her.

Underordnet omforming - Underkonverter 330 - Datakonverterenhet 50 og datainverterenhet 70 koplet i serie - Datakonverterenhet 50 med dataoverføringsfunksjon og datainverterenhet 70 med nøkkeloverføringsfunksjon og dataoverføringsfunksjon.

Figur 29 er et diagram som illustrerer en utførelse der datainverterenheten 70 i figur 26 også er gitt nøkkeloverføringsfunksjonen.

Oppbygningen og virkemåten til datakonverterenheten 50 og datainverterenheten 70 er beskrevet tidligere, og vil ikke bli gjentatt her.

Underordnet omforming - Underkonverter 330 - Datainverterenhet 70 og datakonverterenhet 50 koplet i serie - Datakonverterenhet 50 med dataoverføringsfunksjon og datainverterenhet 70 med nøkkeloverføringsfunksjon og dataoverføringsfunksjon.

Figur 30 illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 og datainverterenheten 70 i figur 29 har byttet plass.

Deres innvendige oppbygning og virkemåte er beskrevet tidligere, og vil ikke bli gjentatt her.

Følgelig tjener FL-nøkkeloverføringssignalet og FL^{-1}-nøkkeloverføringssignalet det formål å undertrykke og således nulle ut innmatede data, mens FL-maskesignalet og FL^{-1}-maskesignalet tjener det formål å la en innmatet nøkkel passere gjennom uforandret.

Videre tjener FL-dataoverføringssignalet og FL^{-1}-dataoverføringssignalet det formål å nulle ut en innmatet nøkkel for å la data passere gjennom uforandret.

Alle de seks ovenfornevnte signalene er overføringssignaler. Når de ikke mottar disse overføringssignalene, utfører datakonverterenheten 50 og datainverterenheten 70 den lineære dataomformingen som er deres egentlige formål, som illustrert i beslektet teknikk.

Underordnet omforming - Underkonverter 330 - Datakonverterenhet 50 og datainverterenhet 70 koplet i serie - Datakonverterenhet 50 og datainverterenhet 70 begge med nøkkeloverføringsfunksjon og dataoverføringsfunksjon.

Figur 62 omfatter den utførelsen av datakonverterenheten 50 som er illustrert i figur 27 og den utførelsen av datainverterenheten 70 som er illustrert i figur 29. Mer spesifikt er datakonverterenheten 50 og datainverterenheten 70, som er koplet i serie, utstyrt med både nøkkeloverføringsfunksjonen og dataoverføringsfunksjonen.

Oppbygningen og virkemåten til datakonverterenheten 50 og datainverterenheten 70 er beskrevet tidligere, og vil ikke bli gjentatt her.

Underordnet omforming - Underkonverter 330 - Datainverterenhet 70 og datakonverterenhet 50 koplet i serie - Datakonverterenhet 50 og datainverterenhet 70 begge med nøkkeloverføringsfunksjon og dataoverføringsfunksjon.

Figur 63 illustrerer en utførelse av underkonverteren 330 der datakonverterenheten 50 og datainverterenheten 70 i figur 62 har byttet plass.

Oppbygningen og virkemåten til disse er beskrevet tidligere, og vil ikke bli gjentatt her.

Utførelsesform 4

I denne utførelsesformen vil det bli gitt en beskrivelse av en utførelse av og en virkemåte for en 1/2-underkonverterenhet 90 i hvilken datakonverterenheten 50 og datainverterenheten 70 er implementert i en delt krets, som støtter nøkkeloverføringsfunksjonen og dataoverføringsfunksjonen.

Underordnet omforming - Underkonverter 330 - 1/2-underkonverterenhet 90 med nøkkeloverføringsfunksjon og dataoverføringsfunksjon.

Figur 31 er et diagram som illustrerer en utførelse av underkonverteren 330 der 1/2-underkonverterenheten 90 er utstyrt med nøkkeloverføringsfunksjonen og dataoverføringsfunksjonen.

I forhold til figur 59, forklart i forbindelse med beslektet teknikk, er nøkkeloverføringssignalet, maskesignalet og dataoverføringssignalet lagt til. I forbindelse med innmating av disse overføringssignalene er det derfor tilveiebragt ytterligere kretser for overføring av en nøkkel og data.

Et skiftesignal er et signal for å skifte mellom datakonverterenheten 50 og datainverterenheten 70. I tilfellet der signalet A blir valgt av skiftesignalet mellom signalene A og E som mates inn til 2-1-signalvelgeren 99a, og deretter matet ut som utsignal B, og signalet C blir valgt av skiftesignalet mellom signalene C og F som mates inn til 2-1-signalvelgeren 99b, og deretter matet ut som utsignal D, utfører 1/2-underkonverterenheten 90 samme dataomforming som den som utføres av datakonverterenheten 50.

På den annen side, i tilfellet der 2-1-signalvelgeren 99a ved skiftesignalet velger signalet E som utsignal B, og 2-1-signalvelgeren 99b ved skiftesignalet velger signalet F som utsignal B, utfører 1/2-underkonverterenheten 90 samme dataomforming som den som utføres av datainverterenheten 70.

I tilfellet der 1/2-underkonverterenheten 90 ved skiftesignalet fungerer som datakonverterenheten 50, er virkemåten illustrert i figur 31 den samme som den til datakonverterenheten 50 vist i figur 18. Spesifikt svarer nøkkel-overføringssignalet til FL-nøkkeloverføringssignalet i figur 18, maskesignalet til FL-maskesignalet i figur 18 og dataoverføringssignalet til FL-dataoverførings-signalet i figur 18.

Konkret motsvarer hver krets som følger. En krets 98 svarer til AND-kretsen 51 (figur 18). En krets 91 svarer til XOR-kretsen 55 (figur 18). En krets 95 svarer til AND-kretsen 60 (figur 18). En krets 101 svarer til AND-kretsen 54 (figur 18). En krets 94 svarer til OR-kretsen 53 (figur 18). En krets 100 svarer til NOT-kretsen 52 (figur 18). En krets 96 svarer til AND-kretsen 58 (figur 18). En krets 92 svarer til OR-kretsen 57 (figur 18). En krets 97 svarer til AND-kretsen 59 (figur 18). En krets 93 svarer til XOR-kretsen 56 (figur 18).

1/2-underkonverterenheten 90, med denne overensstemmende krets-strukturen, kan utføre funksjonen til datakonverterenheten 50 i figur 18. Mer spesifikt er 1/2-underkonverterenheten 90 i stand til å omforme data og også til å overføre nøkkelen (key 1, key 2) ved mottak av nøkkeloverføringssignalet ved å mate ut en innmatet nøkkel som utsignal. Den ovenfornevnte virkemåten er den samme som den til datakonverterenheten 50 i figur 18, og vil derfor ikke bli beskrevet her.

I tilfellet der 1/2-underkonverterenheten 90 ved skiftesignalet fungerer som datainverterenheten 70, er virkemåten til 1/2-underkonverterenheten 90

illustrert i figur 31 den samme som den til datainverterenheten 70 i figur 18. Spesifikt svarer nøkkeloverføringssignalet til FL^{-1}-nøkkeloverføringssignalet i figur 18, maskesignalet til FL^{-1}-maskesignalet i figur 18 og dataoverførings-signalet til FL^{-1}-dataoverføringssignalet i figur 18.

Konkret motsvarer hver krets som følger. Kretsen 98 svarer til AND-kretsen 71 (figur 18), kretsen 91 svarer til XOR-kretsen 76 (figur 18), kretsen 95 svarer til AND-kretsen 80 (figur 18), kretsen 101 svarer til AND-kretsen 77 (figur 18), kretsen 94 svarer til OR-kretsen 78 (figur 18), kretsen 96 svarer til AND-kretsen 73 (figur 18), kretsen 92 svarer til OR-kretsen 74 (figur 18), kretsen 97 svarer til AND-kretsen 79 (figur 18) og kretsen 93 svarer til XOR-kretsen 75 (figur 18).

1/2-underkonverterenheten 90, med denne overensstemmende kretsstrukturen, kan utføre funksjonen til datainverterenheten 70 i figur 18. Mer spesifikt er 1/2-underkonverterenheten 90 i stand til å utføre en invers omforming og også til å overføre nøkkelen (key 3, key 4) ved mottak av nøkkeloverføringssignalet ved å mate ut en innmatet nøkkel som utsignal. Den ovenfornevnte virkemåten er den samme som den til datainverterenheten 70 i figur 18, og vil derfor ikke bli beskrevet her.

Underordnet omforming - Underkonverter 330 - 1/2-underkonverterenhet 90 med dataoverføringsfunksjon.

Figur 32 er et diagram som illustrerer en utførelse av underkonverteren der 1/2-underkonverterenheten 90 er tillagt dataoverføringsfunksjonen.

Som i figur 31 har 1/2-underkonverterenheten 90 samme virkemåte som datakonverterenheten 50 i figur 13 i tilfellet der 2-1-signalvelgeren 99a velger signal A og 2-1-signalvelgeren 99b velger signal C. I dette tilfellet svarer dataoverføringssignalet til FL-dataoverføringssignalet.

1/2-underkonverterenheten 90 har videre samme virkemåte som datainverterenheten 70 i figur 14 i tilfellet der 2-1-signalvelgeren 99a velger signal E og 2-1-signalvelgeren 99b ved skiftesignalet velger signal F. I dette tilfellet svarer dataoverføringssignalet til FL^{-1}-dataoverføringssignalet.

1/2-underkonverterenheten 90 innrettet på denne måten kan utføre en dataomforming, og også overføre data uten omforming ved å mate ut data ved mottak av overføringssignalet som angir overføring av data.

Disse virkemåtene er beskrevet tidligere, og vil derfor ikke bli gjentatt her.

Underordnet omforming - Underkonverter 330 - 1/2-underkonverterenhet 90 med nøkkeloverføringsfunksjon.

Figur 33 er et diagram som illustrerer en utførelse av underkonverteren 330 der 1/2-underkonverterenheten 90 er tillagt nøkkeloverføringsfunksjonen.

Som i figur 31 fungerer 1/2-underkonverterenheten 90 på samme måte som datakonverterenheten 50 i figur 10 i tilfellet der 2-1-signalvelgeren 99a velger signal A og 2-1-signalvelgeren 99b velger signal C. I dette tilfellet svarer nøkkeloverføringssignalet og maskesignalet henholdsvis til FL-nøkkel-overføringssignalet og FL-maskesignalet.

1/2-underkonverterenheten 90 fungerer videre på samme måte som datainverterenheten 70 i figur 11 i tilfellet der 2-1-signalvelgeren 99a velger signal E og 2-1-signalvelgeren 99b ved skiftesignalet velger signal F. I dette tilfellet svarer nøkkeloverføringssignalet og maskesignalet henholdsvis til FL^{-1}-nøkkeloverføringssignalet og FL^{-1}-maskesignalet.

1/2-underkonverterenheten 90 innrettet på denne måten kan utføre en dataomforming, men også overføre data uten omforming ved å mate ut data ved mottak av overføringssignalet som angir overføring av data.

Disse virkemåtene er beskrevet tidligere, og vil derfor ikke bli gjentatt her.

Som beskrevet i denne utførelsesformen er således 1/2-underkonverterenheten 90 tilveiebragt for å implementere datakonverter-enheten 50 og datainverterenheten 70 i den delte kretsen utstyrt med nøkkel-overføringsfunksjonen og dataoverføringsfunksjonen. Dette gjør det mulig å redusere dataomformingsanordningens totale størrelse ved å redusere størrelsen til underkonverteren 330 og ved å redusere økningen av antallet signalvelgere som følge av at behovet for signalbanene for nøkler og data fjernes, som beskrevet tidligere.

I den første til fjerde utførelsesformen ble dataomformingsanordningen for blokk-basert kryptering beskrevet med fokus på utførelser basert på CAMELLIA. Underkonverterne 330 utstyrt med overføringsfunksjonen beskrevet over vil imidlertid også kunne anvendes i en hvilken som helst dataomformings-

anordning som utfører blokk-basert kryptering, for eksempel CAMELLIA, MISTY og KASUMI.

Dataomformingsanordningene beskrevet i forbindelse med figur 1, figur 6 eller figur 7 kan omfatte én eller to 2-1-signalvelgere innlemmet i signalvelgeren 310.

Til sammenlikning krever dataomformingsanordningen fra beslektet teknikk beskrevet tidligere tre 2-1-signalvelgere for å velge ut ett utsignal av fire signaler, som kan sees i figur 56.

I tillegg, i tilfellet der dataomformingsanordningen anvender 1/2F-funksjonen vist i figur 60, kreves det fire 2-1-signalvelgere for å velge ut ett utsignal fra fem innmatede signaler.

Følgelig gjør dataomformingsanordningene illustrert i figurene 1, 6 eller 7 det mulig å redusere antallet signalvelgere i datascrambleren 30 i forhold til dataomformingsanordningene fra beslektet teknikk.

Videre, med henvisning til dataomformingsanordningen illustrert i figur 7, er hovedkonverteren 320 og underkonverteren 330 koplet i parallell, og derfor er det nødvendig med en signalvelger 340. Signalvelgeren 340 utgjøres av én enkelt 2-1-signalvelger som mottar to utsignaler, henholdsvis matet ut fra hovedkonverteren 320 og underkonverteren 330, og velger ut ett av de to mottatte signalene. Det er således nødvendig med to 2-1-signalvelgere for signalvelgerne 310 og 340 i figur 7.

Følgelig gjør også dataomformingsanordningen illustrert i figur 7 det mulig å redusere antallet signalvelgere i datascrambleren 30 i forhold til dataomformingsanordningene fra beslektet teknikk.

Utførelsesform 5

I denne utførelsesformen vil det bli gitt en beskrivelse av en CAMELLIA-basert blokkrypterer der hovedkonverteren 320 og underkonverteren 330 er anordnet parallelt.

CAMELLIA støtter en blokklengde på 128 bit, og nøkkellengder på 128, 192 eller 256 bit kan brukes.

Algoritmestrukturen er FEISTEL-strukturen beskrevet tidligere. Krypteringsprosessen og dekrypteringsprosessen kan i hovedtrekk implementeres i samme maskinvare eller programvare.

F-funksjonen avhenger av nøkkellengden, nærmere bestemt 18 runder for en 128-bits nøkkel (6 runder × 3 i hovedkonverteren 320 i figur 34) og 24 runder for en 192- eller 256-bits nøkkel som illustrert i figurene 54 og 55. Figurene 54 og 55 vil bli beskrevet senere.

Figur 34 er et diagram som illustrerer en CAMELLIA-basert krypteringsprosess for en 128-bits nøkkel. Spesifikt illustrerer figur 34 tilfellet der P (klartekst) blir omformet (kryptert) med bruk av hovedkonverteren 320 og underkonverteren 330, hvorpå C (et kryptogram) blir matet ut. I figur 34 er FL (en dataomformingsfunksjon) og FL^{-1} (en invers dataomformingsfunksjon) tilveiebragt mellom hver seks-runders F-funksjon.

På venstre side illustrerer figur 34 den samme prosessen som den som utføres av datascrambleren 30 i figur 4. Spesifikt svarer XOR-kretsen 31a og XOR-kretsen 31b i figur 4 henholdsvis til en XOR-krets 600 og en XOR-krets 601 i figur 34, og i praksis styrer XOR-enheten i underkonverteren 330 prosessen. Det forutsettes at alle de innmatede nøklene vist i figur 34 er skedulert av og matet ut fra nøkkelfordeleren som vist i figur 4.

På høyre side illustrerer figur 34 et diagram som tilsvarer det i figur 57.

Figur 35 er et diagram som illustrerer en CAMELLIA-basert dekrypteringsprosess for en 128-bits nøkkel.

Figur 35 illustrerer tilfellet der C (et kryptogram) blir omformet med bruk av hovedkonverteren 320 og underkonverteren 330, hvorpå P (dechiffrert tekst) blir matet ut.

Virkemåtene illustrert i figurene 34 og 35 er beskrevet tidligere, og vil ikke bli gjentatt her.

En detaljert beskrivelse vil nå bli gitt av oppbygningen til F-funksjonen for CAMELLIA.

Figur 36 er et diagram som illustrerer en utførelse av F-funksjonen for CAMELLIA.

F-funksjonen for CAMELLIA anvender SPN-struktur internt, og i hovedtrekk blir data prosessert i enheter bestående av åtte bit representert ved inndata (1) til inndata (8). F-funksjonen for CAMELLIA omfatter en S-funksjon 324 som dannes av S-bokser og lineær omforming i eksklusive ELLER-(XOR)-kretser som kalles en P-funksjon 325.

I F-funksjonen 321 blir først inndata (1) til inndata (8), som alle er åtte bit lange, matet inn, og deretter blir disse 64 bitene med innmatede data XOR-behandlet med 64 bit henholdsvis fra åtte bit lange nøkler (1) til (8) og matet ut. Utdataene blir matet inn til S-funksjonen 324 og deretter ikke-lineært omformet byte for byte av S-funksjonen 324, som syntetiserer den inverse aritmetiske operasjonen GF (2^8) og affin omforming.

Dataene gjennomgår deretter en eksklusiv ELLER-basert lineær omforming i P-funksjonen 325. Gjennom disse operasjonene blir data scramblet og deretter matet ut som utdata (1) til utdata (8).

F-funksjonen for CAMELLIA støtter en databredde på 64 bit. Figur 36 viser to sett av S-bokser S_1 til S_4 tilveiebragt i S-funksjonen 324 (ett sett omfattende S_1, S_2, S_3 og S_4 nederst i figur 36 og et annet sett omfattende S_2, S_3, S_4 og S_1 over dette).

Som følge av dette, som kan sees i figur 36, kan prosessen også gjennomføres ved først å omforme inndata (1) til inndata (4) og deretter omforme resten av de innmatede dataene omfattende inndata (5) til inndata (8). I dette tilfellet, i den andre runden av dataomforming, for at kretsen skal kunne brukes som den er med S-boksene anordnet i rekkefølgen S_1 til S_4, må dataene på forhånd rotasjonsskiftes én byte, og deretter kan de rotasjonsskiftede inndata (5) til inndata (8) mates inn. Dette gjør at dataene svarer til S-boksene S_1 til S_4 uten at oppbygningen til S-funksjonen 324 endres.

F-funksjonen implementerer således en ikke-lineær dataomforming gjennom eksklusiv ELLER-(XOR)-operasjoner mellom nøkler og innmatede data, to runder med prosessering i de fire typene S-funksjoner (S_1 til S_4) og prosessering i P-funksjonen 325.

Den typiske CAMELLIA-operasjonen, som innbefatter S-funksjonen 324 i S-boksene (invers aritmetisk krets på GF (2^8) + affin omforming) S_1 til S_4, P-funksjonen 325, dataomformingsfunksjonen (FL) og den inverse dataomformingsfunksjonen (FL^{-1}), kan implementeres med en enkel kombinasjon av boolske algebraer.

Det vil nå bli gitt en detaljert beskrivelse av en komplett utførelse og en CAMELLIA-prosess.

Figur 37 er et diagram som illustrerer en komplett utførelse og en CAMELLIA-prosess.

I CAMELLIA, dersom den hemmelige nøkkelen som mates inn er en 128-bits nøkkel, blir nøkkelen utvidet internt til en 256-bits nøkkel, og nøkkelen utvidet som den utvidede nøkkelen brukes til kryptering/dekryptering av data.

Dersom den hemmelige nøkkelen som mates inn er en 192- eller 256-bits nøkkel, blir nøkkelen utvidet internt til en 512-bits nøkkel for bruk til kryptering/dekryptering av data. Tilfellet med en 192- eller 256-bits nøkkel vil bli beskrevet senere.

Det vil nå først bli gitt en beskrivelse av et oppbygningsmessig trekk ved CAMELLIA.

Hele CAMELLIA-algoritmen utføres gjennom gjentatt prosessering i den samme F-funksjonen i hovedkonverteren 320. F-funksjonen er bygget opp som vist i figur 36.

I datascrambleren 30, som illustrert ved XOR-kretsene 31a og 31b i figur 4, utføres en eksklusiv ELLER-operasjon mellom inn- eller utdataene og en nøkkel. Dette kalles "whitening".

Videre, i datascrambleren 30, er underkonverteren 330, som omfatter dataomforming (FL) og invers dataomforming (FL^{-1}), plassert mellom hovedkonverterne 320, som omfatter den seks-runders F-funksjonen. Dette er illustrert i figurene 34 og 35.

Som nevnt tidligere blir den utvidede nøkkelen (en mellomnøkkel + en utnøkkel) generert som illustrert i figur 2.

Dette viser at dataomformingsanordningen for å implementere CAMELLIA-algoritmen kan være utstyrt med underkonverteren 330, som omfatter dataomforming (FL) og invers dataomforming (FL^{-1}), P-funksjonen 325 og de fire typene S-bokser.

P-funksjonen 325 kan gjøres mindre ved å skrive den basert på fremgangsmåten vist i "Specification of Camellia - a 128 bit Block Cipher".

Spesifikt, ifølge den ovenfornevnte spesifikasjonen, kan P-funksjonen skrives som følger.

$z1' = z1 + z3 + z4 + z6 + z7 + z8$

$z2' = z1 + z2 + z4 + z5 + z7 + z8$

$z3' = z1 + z2 + z3 + z5 + z6 + z8$

$z4' = z2 + z3 + z4 + z5 + z6 + z7$

$z5' = z1 + z2 \quad + z6 + z7 + z8$

z 6' = z 2 + z 3 + z 5 + z 7 + z 8

z 7' = z 3 + z 4 + z 5 + z 6 + z 8

z 8' = z 1 + z 4 + z 5 + z 6 + z 7

"+"-tegnet i likningene over for å beregne z1' til z8' angir en eksklusiv ELLER-operasjon.

Verdiene z1 til z8 er utmatinger henholdsvis fra S1, S2, S3, S4, S5 (=S2), S6 (=S3), S7 (=S4) og S8 (=S1). Dersom nå z5 til z8 henholdsvis transformeres til zz2, zz3, zz4 og zz1, kan likningene skrives som følger.

z 1' = z 1 + z 3 + z 4 + z z 1 + z z 3 + z z 4

z 2' = z 1 + z 2 + z 4 + z z 1 + z z 2 + z z 4

z 3' = z 1 + z 2 + z 3 + z z 1 + z z 2 + z z 3

z 4' = z 2 + z 3 + z 4 + z z 2 + z z 3 + z z 4

z 5' = z 1 + z 2 + z z 1 + z z 3 + z z 4

z 6' = z 2 + z 3 + z z 1 + z z 2 + z z 4

z 7' = z 3 + z 4 + z z 1 + z z 3 + z z 3

z 8' = z 1 + z 4 + z z 2 + z z 3 + z z 4

Basert på dette vil det å prosessere over to klokker, for eksempel å prosessere S1 til S4 for å mate ut Z1 til Z4 og prosessere S1 til S4 for å mate ut zz1 til zz4, etc. gjøre det mulig å forminske kretsen i P-funksjonen til omtrent halve størrelsen.

Med henvisning til figur 37 vil det nå bli gitt en beskrivelse av en CAMELLIA-basert dataomformingsanordning.

Dataomformingsanordningen for CAMELLIA illustrert i figur 37 omfatter hovedkonverteren 320 og underkonverteren 330 anordnet i parallell.

Underkonverteren 330 omfatter datakonverterenheten 50 og data-inverterenheten 70.

Hovedkonverteren 320 har en F-funksjonsenhet som omfatter en 1/2F-funksjon. I tilfellet der hovedkonverteren 320 er utstyrt med en F-funksjonsenhet for mindre enn én F-funksjon, dvs. en $1/2^x$ ($x \geq 1$) F-funksjon, som eksemplifisert av 1/2F-funksjonen i figur 61, vil et utmatet resultat fra XOR-kretsen 1322a, som er et mellomresultat i prosessen som utføres av F-funksjonsenheten 1321a og prosessen som utføres av F-funksjonsenheten 1321b, bli lagret.

Generelt, dersom man reduserer antallet enheter i den installerte F-funksjonen og anvender en fremgangsmåte for å utføre en én-rundes F-funksjons-

basert dataomforming i en arkitektur med flere loop-runder, vil størrelsen til kretsen for F-funksjonen reduseres. På den annen side øker antallet styrekretser for å kontrollere loopen og antallet kretser så som signalvelgere for å mate inn nøkler til hver F-funksjon. Det må således foretas en avveining mellom kretsstørrelsen for F-funksjonen og kretsstørrelsen for loop-styring.

I forbindelse med reduksjon av størrelsen til en CAMELLIA-basert dataomformingsanordning er det derfor nødvendig å vurdere hvilken F-funksjon som skal installeres og antallet repetisjoner. Mer spesifikt bør det vurderes nøye hvorvidt å implementere en dataomformingsanordning for CAMELLIA med bruk av én enkelt F-funksjon tilveiebragt i hovedkonverteren 320, hvorvidt å redusere antallet S-bokser i F-funksjonen og oppnå en én-rundes F-funksjonsbasert dataomforming gjennom en prosess bestående av flere sykluser, etc. Dette er vurderinger knyttet til avveiningen mellom reduksjonen av kretsens størrelse som oppnås ved å anvende en F-funksjonsenhet med mindre enn én F-funksjon og økningen av kretsens størrelse ved å anvende loopen ledsaget av økningen av antallet signalvelgere, etc.

Videre anvendes med CAMELLIA, som nevnt tidligere, deler av hovedkonverteren 320 i datascrambleren 30 som funksjon for å generere utnøkkelen (key KA). På grunn av dette er det også nødvendig å nøye vurdere effekten av å øke antallet signalvelgere og annet som legges til for å kunne anvende F-funksjonen i datascrambleren 30.

Som beskrevet i forbindelse med figur 36, med F-funksjonen for CAMELLIA, anvendes de fire typene S-bokser (S_1, S_2, S_3 og S_4) for 8-bits innmating/utmating to ganger hver. Det er derfor også nødvendig her å vurdere hvorvidt å installere åtte enheter av S-boksene eller å installere fire enheter med repetisjon to ganger, eller liknende.

Ifølge "On Hardware Implementation of 128-bit Block Ciphers (III)", beskrevet i "Proceedings of the 2001 Symposium on Cryptograph and Information Security", omfatter kretsen for én enkelt S-boks omtrent 200 logiske porter, og dersom antallet S-bokser reduseres med fire fra 8 til 4 kan derfor antallet logiske porter reduseres med omtrent 800.

På den annen side er minst 32 enheter av 2-1-signalvelgere (omtrent 100 logiske porter med NAND-kretser) nødvendig for repetisjoner, noe som må tas med i avveiningen.

Basert på dette faktum forventes det at kretsen kan gjøres mindre ved å installere fire S-bokser med to gangers repetisjon heller enn å installere åtte S-bokser.

Med F-funksjonsenheten 321 i dataomformingsanordningen for CAMELLIA kan således dataomforming bli utført i én runde dersom det er tilveiebragt åtte S-bokser, og alternativt kan dataomforming bli utført i to runder med fire S-bokser. Begge alternativer kan anvendes. Med tanke på kretsens størrelse er imidlertid alternativet med to gjentatte dataomforminger det beste.

Ved anvendelse av CAMELLIA-algoritmen illustrert i figur 37 kan kryptering/dekryptering av data utføres av mellomnøkkel-generatoren 40 i henhold til følgende trinnsekvens.

Prosesseringstrinn i syklusen til mellomnøkkel-generatoren 40 vil nå bli beskrevet.

Først, i trinn 1, utføres whitening med bruk av underkonverteren 330.

Deretter utføres i trinn 2 en operasjon for halvparten av én runde av F-funksjonen (1/2F-funksjon) med bruk av hovedkonverteren 320.

Tilsvarende utføres i trinn 3 en operasjon for den andre halvdelen av én runde av F-funksjonen (1/2F-funksjon) med bruk av hovedkonverteren 320.

I trinnene 4 til 13 gjentas trinnene 2 og 3 fem ganger.

I trinn 14 utføres en operasjon for dataomformings-(FL)-funksjonen og den inverse dataomformings-(FL^{-1})-funksjonen i underkonverteren 330 for dataomforming.

Deretter, i trinnene 15 til 27, gjentas trinnene 2 til 14.

Deretter, i trinnene 28 til 39, gjentas trinnene 2 til 13.

Til slutt, i trinn 40, utføres samme whitening som i trinn 1.

Merk her at trinn 1 representerer operasjonen som utføres av XOR-kretsen 31a i figur 4 og trinn 40 representerer operasjonen som utføres av XOR-kretsen 31b i figur 4. Med andre ord anvender XOR-kretsene 31a og 31b henholdvis XOR-enhetene i datakonverterenheten 50 og datainverterenheten 70 i underkonverteren 330.

Oppbygningen og virkemåten til nøkkelgeneratoren 20 er tilsvarende som beskrevet tidligere, og vil derfor ikke bli gjentatt her.

I trinnene 2 og 3 beskrevet ovenfor utføres én enkelt F-funksjonsbasert dataomforming i to sykluser av hovedkonverteren 320. Denne dataprosessen vil nå bli beskrevet i detalj med henvisning til figur 37 og figur 64.

Figur 64 skiller seg fra figur 61 ved at en nedre nøkkel blir matet inn først som en inn-nøkkel og at en øvre nøkkel deretter blir matet inn for å implementere prosessen.

Først vil trinn 1 bli beskrevet i detalj. Innmatede data P (klartekst eller dechiffrert tekst) velges av en 2-1-signalvelger 311 og deles inn i øvre data og nedre data. De øvre dataene gjennomgår whitening i datakonverterenheten 50 i underkonverteren 330, og de nedre dataene mates inn til datainverterenheten 70 i underkonverteren 330 og gjennomgår en tilsvarende whitening. De whitening-behandlede øvre dataene og nedre dataene blir henholdsvis matet inn til en 2-1-signalvelger H 341 og en 2-1-signalvelger L 342 i en 2-1-signalvelger 340. Hver del av de innmatede dataene blir henholdsvis valgt av 2-1-signalvelgeren H 341 eller 2-1-signalvelgeren L 342 og deretter henholdsvis lagt i et aritmetisk register H 351 og et aritmetisk register L 352.

Nå vil en prosess i trinn 2 bli beskrevet.

De øvre bitene av de øvre dataene inneholdt i det aritmetiske registeret H 351 blir matet inn til en 2-1-signalvelger 312, og de nedre bitene av de øvre dataene blir rotasjonsskiftet med én byte og deretter matet inn til 2-1-signalvelgeren 312. 2-1-signalvelgeren 312 velger de nedre, rotasjonsskiftede bitene av de to innmatede dataene, og mater ut de valgte bitene til hovedkonverteren 320. Rotasjonsskiftet av de valgte nedre bitene med én byte muliggjør optimal bruk og innmating av inndata (5) til inndata (8) for S-boksene, som kan sees i figur 36. I hovedkonverteren 320 utføres den øvre halvparten av den første runden av dataomformingen illustrert i figur 64 av F-funksjonsenheten 321, som har 1/2F-funksjonen. Merk her at F-funksjonsenheten 321 i figur 37 og F-funksjonsenhetene 1321a til 1321l i figur 64 med 1/2F-funksjonen er bygget opp på samme måte. Med henvisning til en dataomforming som utføres av F-funksjonsenheten 1321a i figur 64, blir den nedre halvparten av bitene av inn-matede øvre data omformet med bruk av en nøkkel key 1L, og de omformede dataene blir deretter matet ut til XOR-kretsen 1322a. XOR-kretsen 1322a mottar de omformede dataene matet ut fra F-funksjonsenheten 1321a og XOR-behandler de mottatte dataene med innmatede nedre data. Med andre ord blir

dataene (mellomdata) som mates ut fra hovedkonverteren 320 matet inn til en 3-1-signalvelger L 342 og deretter lagt i et aritmetisk register L 352. Samtidig passerer de øvre dataene av P inneholdt i det aritmetiske registeret H 351 uforandret gjennom 2-1-signalvelgeren 311, og blir deretter overført, for eksempel ved hjelp av dataoverføringsfunksjonen til datakonverterenheten 50 i underkonverteren 330, og lagt i det aritmetiske registeret H 351 igjen via 2-1-signalvelgeren H 341 fra det aritmetiske registeret H 351.

Nå vil en prosess i trinn 3 bli beskrevet.

Prosesseringen til F-funksjonsenheten 1321b i figur 64 utføres i den andre prosesseringssyklusen i hovedkonverteren 320 i figur 37. Spesifikt blir, uten å gjennomgå det én-bytes rotasjonsskiftet, de øvre bitene av de øvre dataene matet inn til 2-1-signalvelgeren 312 valgt av 2-1-signalvelgeren 312 og deretter matet ut til hovedkonverteren 320. Gjennom denne operasjonen blir dataene som danner den øvre halvdelen av bitene av de øvre dataene ikke-lineært omformet av F-funksjonsenheten 1321b og deretter matet ut XOR-kretsen 1322b. XOR-kretsen 1322b mater inn hovedkonverteren 320 mellom-dataene, som mates ut fra hovedkonverteren 320 og legges i det aritmetiske registeret L 352 i den første syklusen, som det andre innsignalet, og dermed mates mellomdataene inn til XOR-kretsen 1322b. De XOR-behandlede ut-dataene fra XOR-kretsen 1322b velges av 2-1-signalvelgeren H 341, og blir deretter lagt i det aritmetiske registeret H 351. Samtidig legges de øvre dataene av P i det aritmetiske registeret L 352 av 3-1-signalvelgeren 342. Dette betyr at de øvre dataene og de nedre dataene som skal brukes til dataomforming i den andre runden i hovedkonverteren 320 henholdsvis er inneholdt i det aritmetiske registeret H 351 og det aritmetiske registeret L 352.

I trinnene 4 til 13 gjentas trinnene 2 og 3 fem ganger.

Mer spesifikt utføres dataomformingen for den andre runden av F-funksjonsenheten 1321c og XOR-kretsen 1322c i én syklus og av F-funksjons-enheten 1321d og XOR-kretsen 1322d i en annen syklus, og disse to syklusene svarer sammen til trinnene 4 og 5. Prosessene i den tredje til sjette runden utføres på samme måte, og svarer til trinnene 6 til 13.

Merk som tidligere angitt at virkemåten til F-funksjonsenhetene 1321a til 1321l i figur 64 er den samme som virkemåten til F-funksjonsenheten 321 i figur 37.

Nå vil en prosess i trinn 14 bli beskrevet.

Denne prosessen representerer prosessen som utføres av underkonverteren 330 i figur 37.

Først blir de øvre dataene og de nedre dataene, som ble prosessert i trinn 13 og deretter henholdsvis lagt i det aritmetiske registeret H 351 og i det aritmetiske registeret L 352, matet inn til 2-1-signalvelgeren 311 og deretter valgt og henholdsvis matet inn til datakonverterenheten 50 og datainverterenheten 70.

I datakonverterenheten 50 og datainverterenheten 70 blir de innmatede dataene lineært omformet. Deretter blir dataene omformet av datakonverterenheten 50 matet inn til 2-1-signalvelgeren H 341, mens dataene omformet av datainverterenheten 70 blir matet inn til 3-1-signalvelgeren L 342. Deretter blir de valgt og henholdsvis lagt i det aritmetiske registeret H 351 og i det aritmetiske registeret L 352.

Prosessene i trinnene 15 til 27 svarer til prosessene i hovedkonverteren 320 og underkonverteren 330 i figur 37.

Prosessene i trinnene 28 til 39 svarer til prosessene i hovedkonverteren 320 i figur 37.

I trinn 40, som i trinn 1, utføres whitening med bruk av en XOR-enhet i underkonverteren 330.

Gjennom disse trinnene i mellomnøkkel-generatoren 40 muliggjøres utmating av et kryptogram C etter kryptering hvis de innmatede dataene, P, er klartekst, og utmating av en dechiffrert tekst, C, etter dekryptering i den samme kretsen som brukes til kryptering hvis de innmatede dataene P er et kryptogram.

I den CAMELLIA-baserte dataomformingsanordningen i figur 37 gjør den parallelle anordningen av hovedkonverteren 320 og underkonverteren 330 det mulig å redusere prosesseringstiden i hver syklus og å øke operasjonsfrekvensen i forhold til om de var serielt anordnet.

Videre, med den parallelle anordningen av hovedkonverteren 320 og underkonverteren 330, tilveiebringes signalbanen for å mate inn et signal til underkonverteren 330 uten at signalet må passere gjennom hovedkonverteren 320 og signalbanen for å mate inn et signal til hovedkonverteren 320 uten at signalet må passere gjennom underkonverteren 330. Dette muliggjør en

fleksibel tilpasning til endringer i anordningens oppbygning og virkemåte, for eksempel tillegging eller fjerning av komponenter eller liknende i fremtidige utførelser.

I den CAMELLIA-baserte dataomformingsanordningen der hovedkonverteren 320 og underkonverteren 330 er serielt anordnet på den annen side, i tilfellet der den én-rundes dataomformingen utføres av F-funksjonsprosessen over to eller flere sykluser, siden data som skal omformes i én syklus er en del av inndata, er signalbanen i datascrambleren 30 for å holde data omformet basert på den aktuelle delen av inndataene i det aritmetiske registeret 350 og overføre de omformede dataene til underkonverteren 330 etter en forbestemt tid nødvendig. Alternativt er overføringsbanen i hovedkonverteren 320 for å overføre dataene til underkonverteren 330 via hovedkonverteren 320 etter den forbestemte tiden nødvendig.

I denne utførelsesformen gjøres imidlertid både den ekstra signalbanen og den ekstra overføringsfunksjonen i hovedkonverteren 320 overflødig siden hovedkonverteren 320 og underkonverteren 330 er parallelt anordnet. Dette er med på å hindre at størrelsen til kretsene i anordningen øker.

I tillegg, ved bruk av den delte kretsen illustrert i figur 59 som implementerer datakonverterenheten 50 og datainverterenheten 70, blir signalbanen $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E \rightarrow B \rightarrow C \cdots$en loopkrets. Derfor bør loopkretsen være konskruert på en slik måte at den ikke blir en overføringskrets når den påvirkes av "signal racing", støy, etc. forårsaket av forskjeller i skiftesignalenes forplantningsforsinkelse i en praktisk LSE-basert implementasjon av kretsen. Et annet problem er at logisk synteseverktøy ikke er i stand til å håndtere en slik krets med loopkretsen (en "FEED-LOOP"-krets), slik at det ikke er mulig å oppnå en effektiv logisk syntese.

For å løse dette problemet er datakonverterenheten 50 og datainverterenheten 70 i underkonverteren 330 tilveiebragt som separate enheter i figur 37. Dette gjør at dataomformingsanordningen unngår problemer knyttet til "racing" og annet.

I tillegg, som nevnt tidligere, kan underkonverteren 330 i figur 37, som omfatter nøkkel-/dataoverføringsfunksjonen, fjerne behovet for nøkkeloverføringsbanen fra key KL-registeret 240 og dataoverføringsbanen fra hovedkonverteren 320. Dette kan bidra til en ytterligere forminskning av data-

omformingsanordningen for CAMELLIA-basert blokkryptering og til å oppnå et lavt strømforbruk.

Utførelsesform 6

En sjette utførelsesform vil nå bli beskrevet.

Figur 38 illustrerer en CAMELLIA-basert dataomformingsanordning ifølge en sjette utførelsesform. Figur 38 skiller seg fra 37 ved at underkonverteren 330 omfatter 1/2-underkonverterenheten 9, som implementerer datakonverterenheten 50 og datainverterenheten 70 i den delte kretsen. Som følge av dette fjernes behovet for 2-1-signalvelgeren 215 og 4-1-signalvelgeren 217 i figur 37.

Dataomformingsanordningen ifølge denne utførelsesformen trenger således ikke noen av de fire signalvelgerne som kreves for 2-1-signalvelgeren 215 og 4-1-signalvelgeren 217, eller signalbanen for å mate inn en nøkkel matet ut fra 2-1-signalvelgeren 215 til underkonverteren 330. Dette gjør det mulig å forenkle nøkkelfordeleren 210, og dermed redusere dataomformingsanordningens størrelse ytterligere.

Utførelsesform 7

Figur 47 er et blokkdiagram som illustrerer en CAMELLIA-basert dataomformingsanordning ifølge en syvende utførelsesform.

Denne utførelsesformen skiller seg fra den illustrert i figur 37 ved at F-funksjonsenheten 321 i hovedkonverteren 320 er utstyrt med en 1/8F-funksjon. Med andre ord utfører hovedkonverteren 320 ifølge denne utførelsesformen en én-rundes F-funksjonsbasert dataomforming over åtte sykluser. Følgelig, i forhold til figur 37, er 2-1-signalvelgeren 312 i figur 37 byttet ut med en 8-1-signalvelger 315. De andre komponentene er de samme som de i figur 37.

Utførelsesform 8

Figur 48 er et blokkdiagram som illustrerer en CAMELLIA-basert dataomformingsanordning ifølge en åttende utførelsesform.

Denne utførelsesformen skiller seg fra den illustrert i figur 47 ved at underkonverteren 330 er utstyrt med 1/2-underkonverterenheten 90. Som følge av dette fjernes behovet for 2-1-signalvelgeren 215 og 4-1-signalvelgeren 217 i figur 47.

Utførelsesform 9

En annen utførelsesform er illustrert i figur 49.

Figur 49 er et blokkdiagram som illustrerer en CAMELLIA-basert data-omformingsanordning ifølge en niende utførelsesform.

Denne utførelsesformen skiller seg fra den illustrert i figur 37 ved at F-funksjonsenheten 321 i hovedkonverteren 330 er utstyrt med en 1/4F-funksjon. Følgelig er 2-1-signalvelgeren 312 i figur 37 byttet ut med en 4-1-signalvelger 316 i figur 49. Hovedkonverteren 320 omformer data i fire sykluser for å gjennomføre en én-rundes F-funksjonsbasert dataomforming med F-funksjonsenheten 321 med bruk av 16-bits inndata valgt av 4-1-signalvelgeren 316. De andre komponentene er de samme som de i figur 37.

Utførelsesform 10

Figur 50 er et blokkdiagram som illustrerer en CAMELLIA-basert data-omformingsanordning ifølge en tiende utførelsesform.

Denne utførelsesformen skiller seg fra utførelsesformen i figur 49 ved at underkonverteren 330 er utstyrt med 1/2-underkonverterenheten 90. I forhold til figur 49 er derfor 2-1-signalvelgeren 215 og 4-1-signalvelgeren overflødige. Andre komponenter er de samme som de i figur 49.

Utførelsesform 11

En ellevte utførelsesform vil nå bli beskrevet.

Figur 39 er et blokkdiagram som illustrerer en CAMELLIA-basert data-omformingsanordning ifølge den ellevte utførelsesformen.

Figur 39 skiller seg fra figur 37 ved at hovedkonverteren 320 omfatter en F-funksjonsenhet 321 med én enkelt F-funksjon. Som følge av dette kan hoved-konverteren 320 utføre F-funksjonsprosessen for én runde i én syklus, hvilket fjerner behovet for 2-1-signalvelgeren 312 i figur 37. 2-1-signalvelgeren 212 i figur 37 er også overflødig, og 8-1-signalvelgeren 213 er byttet ut med en 4-1-signalvelger 218 som velger ut én av fire konstanter.

Utførelsesform 12

En tolvte utførelsesform vil nå bli beskrevet.

Figur 40 er et blokkdiagram som illustrerer en CAMELLIA-basert data-omformingsanordning ifølge den tolvte utførelsesformen.

I figur 40 er det lagt til en 2-1-signalvelger 313. Siden underkonverteren 330 er utstyrt med 1/2-underkonverterenheten 90, velges én av de øvre dataene og de nedre dataene av dataene valgt av 2-1-signalvelgeren 311. I denne utførelsesformen utføres prosessen i hovedkonverteren 320 i én syklus, slik at 2-1-signalvelgeren 312 akkurat som i figur 39 er overflødig. Også 2-1-signalvelgeren 215 eller 4-1-signalvelgeren 217 i figur 39 er overflødig.

Utførelsesform 13

En trettende utførelsesform vil nå bli beskrevet.

Figur 41 er et blokkdiagram som illustrerer en CAMELLIA-basert data-omformingsanordning ifølge den trettende utførelsesformen.

Figur 41 skiller seg fra figur 39 ved at hovedkonverteren 320 ikke gjentar prosessen i F-funksjonsenheten 321 seks ganger, men har seks runder av F-funksjonsenheten 321 som er seriekoplet og omformer data. Som følge av dette er det i denne utførelsesformen et ekstra utsignal fra hovedkonverteren 320. Grunnen til dette er at utdata fra den andre F-funksjonsrunden i hoved-konverteren 320 henholdsvis skal mates inn til en 3-1-signalvelger H 343 og en 4-1-signalvelger L 344 og deretter henholdsvis legges i det aritmetiske registeret H 351 og i det aritmetiske registeret L 352. Følgelig mottar 3-1-signal-velgeren H 343 tre signaler, mens 4-1-signalvelgeren L 344 mottar fire signaler.

I tillegg er det tilveiebragt fire sett av 4-1-signalvelgere 500 og 4-1-signal-velgere 501, og det er tilveiebragt ekstra signalvelgere for å mate inn fire nøkler som velges av disse signalvelgerne til hovedkonverteren 320. Videre mottar underkonverteren 330 og hovedkonverteren 320 nøkler også fra andre signal-velgere, nemlig en 4-1-signalvelger 502 og en 4-1-signalvelger 503 i nøkkel-fordeleren 210.

Utførelsesform 14

En fjortende utførelsesform vil nå bli beskrevet.

Figur 42 er et blokkdiagram som illustrerer en CAMELLIA-basert data-omformingsanordning ifølge den fjortende utførelsesformen.

Figur 42, i likhet med figur 41, skiller seg fra figur 40 ved at F-funksjonsenheten i hovedkonverteren 320 er utstyrt med den seks-runders F-funksjonen koplet i serie. Som følge av dette, i likhet med tilfellet i figur 41, er antallet innsignaler til 3-1-signalvelgeren H 343 og 4-1-signalvelgeren L 344 økt med én sammenliknet med figur 40, og det er nødvendig med fire sett av 4-1-signalvelgere 500 og 4-1-signalvelgere 501. Underkonverteren 330 og hovedkonverteren 320 mottar nøkler også fra andre signalvelgere i nøkkelfordeleren 210, nemlig 4-1-signalvelgeren 502 og 3-1-signalvelgeren 504. 3-1-signalvelgeren 504 mottar tre innsignaler.

Utførelsesform 15

En annen utførelsesform er illustrert i figur 43.

Figur 43 er et blokkdiagram som illustrerer en CAMELLIA-basert dataomformingsanordning ifølge en femtende utførelsesform.

Denne utførelsesformen skiller seg fra den illustrert i figur 41 ved at hovedkonverteren 320 er utstyrt med en F-funksjonsenhet 321 med den to-runders F-funksjonen. Som følge av dette, sammenliknet med figur 41, er 3-1-signalvelgeren H 343 og 4-1-signalvelgeren L 344 henholdsvis byttet ut med en 2-1-signalvelger H 341 og en 3-1-signalvelger L 342, og de fire settene av signalvelgere omfattende 4-1-signalvelgerne 500 og 4-1-signalvelgerne 501 er overflødige.

Utførelsesform 16

En annen utførelsesform er illustrert i figur 44.

Figur 44 er et blokkdiagram som illustrerer en CAMELLIA-basert dataomformingsanordning ifølge en sekstende utførelsesform.

Denne utførelsesformen skiller seg fra utførelsesformen i figur 42 ved at F-funksjonsenheten 321 i hovedkonverteren 320 er den to-runders F-funksjonen. Som følge av dette er 3-1-signalvelgeren H 343 og 4-1-signalvelgeren L 344 i figur 42 henholdsvis byttet ut med en 2-1-signalvelger H 341 og en 3-1-signalvelger L 342, og de fire settene av signalvelgere omfattende 4-1-signalvelgeren 500 og 4-1-signalvelgeren 501 er overflødige.

Utførelsesform 17

Figur 45 er et blokkdiagram som illustrerer en CAMELLIA-basert data-omformingsanordning ifølge en syttende utførelsesform.

I denne utførelsesformen omfatter F-funksjonsenheten 321 i hoved-konverteren 320 en tre-runders F-funksjon. Som følge av dette, i forhold til figur 41, er de fire settene av signalvelgere omfattende 4-1-signalvelgeren 500 og 4-1-signalvelgeren 501 overflødige, og en 4-1-signalvelger 505 er lagt til i stedet. Signalet som velges av 4-1-signalvelgeren 505 blir matet inn til hoved-konverteren 320.

Utførelsesform 18

Figur 46 er et blokkdiagram som illustrerer en CAMELLIA-basert data-omformingsanordning ifølge en attende utførelsesform.

I likhet med utførelsesformen vist i figur 45 omfatter F-funksjonsenheten 321 i hovedkonverteren 320 en tre-runders F-funksjon. Denne utførelsesformen skiller seg fra den i figur 45 ved at underkonverteren 330 omfatter 1/2-under-konverterenheten 90. Andre komponenter er de samme som de i figuren.

Utførelsesform 19

Figur 51 er et blokkdiagram som illustrerer en CAMELLIA-basert data-omformingsanordning ifølge en nittende utførelsesform.

For det første er mellomnøkkel-generatoren 40 i denne utførelsesformen forskjellig fra den i figur 37. Oppbygningen til mellomnøkkel-generatoren 40 i denne utførelsesformen er ekvivalent med den til mellomnøkkel-generatoren 40 i figur 37, blant annet. Mellomnøkkel-generatoren 40 i figur 37 kan derfor byttes ut med mellomnøkkel-generatoren 40 i figur 51.

En beskrivelse vil nå bli gitt av mellomnøkkel-generatoren 40 i figur 51.

Først mottar en 2-1 KL-signalvelger 291 en innmatet hemmelig nøkkel og en mellomnøkkel (key KL) inneholdt i key KL-registeret 240, velger ut ett av disse to innmatede signalene og legger en nøkkel i key KL-registeret 240. En 2-1 KA-signalvelger 292 mottar en utnøkkel generert av mellomnøkkel-generatoren 40 og en utnøkkel (key KA) inneholdt i key KA-registeret 250. 2-1 KA-signalvelgeren 292 velger ut ett av disse to innmatede signalene og holder det valgte signalet i key KA-registeret 250.

En 2-1-signalvelger 227 velger ut én nøkkel av mellomnøkkelen (key KL) og utnøkkelen (key KA), henholdsvis inneholdt i og matet ut fra key KL-registeret 240 og key KA-registeret 250, og mater ut en nøkkel til en 8-1-signalvelger 228. I 8-1-signalvelgeren 228 blir en nøkkel valgt av 2-1-signalvelgeren 227 rotasjonsskiftet med åtte forskjellige antall bit, nemlig 0, 15, 30, 45, 60, 77, 94 og 111, mot venstre eller høyre som kan sees i figur 51. Dersom antallet bit for rotasjonsskiftet er 0, blir ikke dataene skift-behandlet. Dersom antallet bit for rotasjonsskiftet er 15, blir dataene rotasjonsskiftet 15 bit mot høyre eller venstre. Det samme gjelder for de andre verdiene. Gjennom rotasjonsskift av data på denne måten frembringes åtte signaler. Deretter velger 8-1-signalvelgeren 228 ett av de åtte signalene, og mater ut det valgte signalet.

Denne virkemåten gjør at mellomnøkkel-generatoren 40 i denne utførelsesformen kan fungere på samme måte som mellomnøkkel-generatoren 40 i figur 37. På denne måten blir øvre halvdel av databitene matet ut fra mellomnøkkel-generatoren 40 KLH, mens nedre halvdel av databitene blir KLL, og disse blir henholdsvis matet inn til en 2-1-signalvelger 510 og en 2-1-signalvelger 511 i nøkkelfordeleren 210. Følgelig kan 4-1-signalvelgeren 216 og 4-1-signalvelgeren 217 i figur 37 henholdsvis byttes ut med 2-1-signalvelgeren 510 og 2-1-signalvelgeren 511 i denne utførelsesformen.

Mellomnøkkel-generatoren 40 illustrert i figur 52, i likhet med mellomnøkkel-generatoren 40 illustrert i figur 37, krever derfor ti 2-1-signalvelgere. Videre krever 2-1-signalvelgeren 510 og 2-1-signalvelgeren 511 kun to 2-1-signalvelgere. Følgelig er det totale antallet 2-1-signalvelgere nødvendig for mellomnøkkel-generatoren 40, 2-1-signalvelgeren 510 og 2-1-signalvelgeren 511, lik 12.

Mellomnøkkel-generatoren 40 illustrert i figur 37 krever ti 2-1-signalvelgere, mens 4-1-signalvelgeren 216 og 4-1-signalvelgeren 217 krever seks 2-1-signalvelgere. Følgelig er det totale antallet 2-1-signalvelgere nødvendig for mellomnøkkel-generatoren 40, 4-1-signalvelgeren 216 og 4-1-signalvelgeren 217, lik 16.

Dataomformingsanordningen ifølge denne utførelsesformen kan således redusere antallet 2-1-signalvelgere med fire sammenliknet med dataomformingsanordningen illustrert i figur 37.

Denne utførelsesformen gjør det derfor mulig å redusere størrelsen som følge av reduksjonen av antallet signalvelgere, og også å oppnå et lavt strømforbruk som følge av reduksjonen av antallet logikkporter ledsaget av reduksjonen av antallet signalvelgere.

Merk at den utførelsen av mellomnøkkel-generatoren 40 som er beskrevet i denne utførelsesformen også vil kunne anvendes som mellomnøkkel-generator i alle andre utførelsesformer av oppfinnelsen.

Utførelsesform 20

Figur 52 er et blokkdiagram som illustrerer en CAMELLIA-basert dataomformingsanordning ifølge en tyvende utførelsesform.

Denne utførelsesformen skiller seg fra den illustrert i figur 51 ved at underkonverteren 330 omfatter 1/2-underkonverterenheten 90. Følgelig er ikke 2-1-signalvelgeren 215 og 2-1-signalvelgeren 511 i figur 51 nødvendige i denne utførelsesformen. Andre komponenter er de samme som de i figur 51.

Merk at det med "rotasjonsskift-bitantallet" referert til i figur 51 og figur 52 menes antallet bit for rotasjonsskiftet.

Utførelsesform 21

Figurene 34 og 35, som beskrevet i forbindelse med den fjerde utførelsesformen, illustrerer CAMELLIA-prosessen for kryptering/dekryptering med en 128-bits nøkkel.

Alle de utførelsene av dataomformingsanordninger som er beskrevet i forbindelse med de gitte utførelsesformene av foreliggende oppfinnelse, kan imidlertid anvendes for en hvilken som helst dataomformingsanordning som implementerer CAMELLIA-algoritmen for kryptering/dekryptering, ikke bare for en 128-bits nøkkel, men også for en 192- eller 256-bits nøkkel.

Figur 53 er et diagram som illustrerer en prosess for å generere en 192-bits nøkkel.

Som angitt tidligere blir det med en 128-bits nøkkel generert en 256-bits nøkkel som den utvidede nøkkelen. Ved innmating av en 192- eller 256-bits hemmelig nøkkel er lengden til den utvidede nøkkelen 512 bit.

I figur 53 er en nøkkel key KL og en nøkkel key KR mellomnøkler, og en nøkkel key KA og en nøkkel key KB er utnøkler. Her er alle nøklene KL, KR, KA

og KB 128 bit lange, og sammensetting av nøklene gir derfor en 512-bits utvidet nøkkel.

Ved innmating av en 256-bits hemmelig nøkkel tilordnes nøkkelen key KL 128 bit, nærmere bestemt den øvre halvdelen av bitene i den innmatede hemmelige nøkkelen, og nøkkelen key KR tilordnes de nedre 128 bit.

Nøklene key KL og key KR blir henholdsvis XOR-behandlet og deretter matet inn til en del av hovedkonverteren 320, som kan sees i figur 53.

Figur 53 svarer til høyre side av figur 2, som viser prosessen for å generere en utvidet nøkkel i tilfellet der den hemmelige nøkkelen er 128 bit lang.

Prosessen for å generere utnøkkelen key KA basert på en innmatet nøkkel som illustrert til venstre i figur 53, er den samme som prosessen for å generere utnøkkelen key KA illustrert i figur 2, bortsett fra at den innmatede nøkkelen er basert på resultatet av en XOR-behandling med nøkkelen key KL eller key KR. Figur 2 viser ikke prosessen med å generere utnøkkelen key KB basert på key KR vist til høyre i figur 53. Prosessen med å generere utnøkkelen (key KB) vil derfor bli beskrevet nå.

Ved innmating av en 256-bits nøkkel, anvendes de nedre 128 bit som inn-nøkkelen (key KR), som mates inn til hovedkonverteren 320. De øvre bitene av de nedre 128 bitene blir ikke-lineært omformet med en konstant $\Sigma 5$ med bruk av F-funksjonsenheten 321a i det første trinnet i hovedkonverteren 320, og deretter matet ut. Utdataene blir XOR-behandlet med de nedre bitene av innøkkelen (key KR) i XOR-kretsen 322a, og deretter matet inn til F-funksjonsenheten 321b. I F-funksjonsenheten 321b blir dataene ikke-lineært omformet på nytt med en annen konstant $\Sigma 6$, og de omformede dataene blir deretter XOR-behandlet med de øvre bitene av inn-nøkkelen (key KR) i XOR-kretsen 322b. Dataene omformet gjennom en operasjon i XOR-kretsen 322a blir matet ut som de øvre 64 bitene av utnøkkelen (key KB), og resultatdata fra en behandling i XOR-kretsen 322a blir matet ut som de nedre 64 bitene av utnøkkelen (key KB).

De på denne måten genererte utnøklene (key KA og key KB) og inn-nøklene (key KL og key KR) blir overført som en 512-bits utvidet nøkkel fra mellomnøkkel-generatoren 40 til nøkkelfordeleren 210, og deretter skedulert av nøkkelfordeleren 210 og brukt til kryptering/dekryptering av data.

Ved innmating av en 192-bits hemmelig nøkkel, anvendes de øvre 128 bit av den innmatede hemmelige nøkkelen som nøkkelen key KL. Videre anvendes de nedre 64 bit av den innmatede hemmelige nøkkelen som de øvre 64 bit av nøkkelen key KR. De nedre 64 bitene av nøkkelen key KR er de inverse av de øvre 64 bitene av nøkkelen key KR, som er de nedre 64 bitene av den innmatede hemmelige nøkkelen. Andre fremgangsmåter for å generere nøkler er de samme som fremgangsmåten for å generere den 256-bits hemmelige nøkkelen, og vil derfor ikke bli beskrevet her.

Figur 54 er et diagram som illustrerer CAMELLIA-kryptering med en 192- eller 256-bits nøkkel.

Sammenliknet med figur 34, som illustrerer CAMELLIA-kryptering med en 128-bits nøkkel, er antallet hovedkonvertere 320 økt fra tre til fire, og antallet underkonvertere 330 er økt fra to til tre. Krypteringsalgoritmen for en 192- eller 256-bits nøkkel anvender derfor en 24-runders F-funksjon for kryptering. Andre komponenter er de samme som de for en 128-bits nøkkel illustrert i figur 34, og vil derfor ikke bli beskrevet her.

Figur 55 er et diagram som illustrerer CAMELLIA-dekryptering med en 192- eller 256-bits nøkkel.

CAMELLIA-dekryptering med en 128-bits nøkkel er beskrevet tidligere i forbindelse med figur 35. I forhold til figur 35 er antallet hovedkonvertere 320 økt til fire og antallet underkonvertere 330 er økt til tre, og en 24-runders F-funksjon er tilveiebragt som i krypteringsprosessen. Andre komponenter er de samme som de for CAMELLIA-dekryptering med en 128-bits nøkkel, og vil derfor ikke bli beskrevet her.

Detaljene ved CAMELLIA-algoritmen for blokkryptering med 128-, 192- og 256-bits nøkler er beskrevet i "128-bit Block Cipher Camellia Algorithm Specification".

Alle utførelsesformene beskrevet over er anvendelige for en hvilken som helst dataomformingsanordning for en 128-, 192- eller 256-bits nøkkel.

Nøkkel- og dataoverføringsfunksjonen tilveiebragt i underkonverteren 320 kan anvendes i alle utførelsesformer av foreliggende oppfinnelse.

I alle de foregående utførelsesformene er virkemåtene til de respektive komponentene relatert til hverandre, og operasjonene som utføres av de respektive komponentene vil derfor kunne erstattes med en sekvens av

handlinger basert på relasjonen mellom operasjonene beskrevet over. Med dette kan utførelsesformene av oppfinnelsen omgjøres til fremgangsmåter.

Videre, dersom prosesseringen i de respektive komponentene byttes ut med deres virkemåter, kan de ovenfor beskrevne utførelsesformene omgjøres til utførelsesformer som viser programmer.

Enda videre, dersom programmene lagres i datamaskin-lesbare lagringsanordninger som lagrer programmer, kan utførelsesformene omgjøres til utførelsesformer som viser datamaskin-lesbare lagringsanordninger.

Alle utførelsesformer som viser programmer eller utførelsesformer som viser datamaskin-lesbare lagringsanordninger kan realiseres i form av datamaskin-eksekverbare programmer.

Prosesseringen i de respektive utførelsesformene som viser programmer og de i utførelsesformer som viser datamaskin-lesbare lagringsanordninger kan utføres av programmer som er lagret i et minne. Programmene leses fra minnet av en sentralprosesseringsenhet (CPU) og eksekveres av sentralprosesseringsenheten for å implementere prosesskjemaer. Merk at minnet og sentralprosesseringsenheten ikke er vist i figurene.

Merk også at programvaren eller programmet i hver utførelsesform kan være implementert i fastvarekode lagt i et ROM (Read Only Memory). Alternativt kan hver funksjon i de foregående programmene være implementert i en kombinasjon av programvare, fastvarekode og maskinvare.

Industriell anvendelighet

Begrensningen av økningen av signalvelgere og reduksjonen av antallet signalvelgere gjør det mulig å redusere anordningens størrelse.

Videre gjør reduksjonen av det totale antallet logikkporter i kretser det mulig å oppnå et lavt strømforbruk.

Enda videre er det mulig å øke operasjonsfrekvensen.

Underkonverteren kan overføre innmatede data eller en innmatet nøkkel.

En fleksibel tilpasning til en endring av anordningens oppbygning er mulig.

Det er mulig at én av datakonverterenheten 50 og datainverterenheten 70 besørger dataomforming og at den andre av datakonverterenheten 50 og datainverterenheten 70 overfører innmatede data eller en innmatet nøkkel.

Behovet for signalbanen fra hovedkonverteren til signalvelgeren er fjernet, slik at anordningen kan gjøres mer kompakt, mens reduksjonen av antallet signalvelgere gjør det mulig å oppnå et lavt strømforbruk.

PATENTKRAV

1. Dataomformingsanordning som mottar en nøkkel og data, og utfører dataomforming for én av kryptering eller dekryptering av de mottatte dataene med bruk av den mottatte nøkkelen, idet dataomformingsanordningen omfatter:

en datascrambler (30) som er egnet til å besørge eller utføre dataomforming, og

en styringsenhet (5) som er egnet til å kontrollere et overføringssignal som angir at én av nøkkelen og dataene skal overføres,

hvor styringsenheten (5) er egnet til å mate ut overføringssignalet når den ene av nøkkelen og dataene skal overføres,

hvor datascrambleren (30) omfatter en underkonverter (330) som er egnet til å utføre dataomformingen for den ene av kryptering og dekryptering ved å omforme de mottatte dataene med bruk av den mottatte nøkkelen, og som er egnet til å overføre minst én av den mottatte nøkkelen og de mottatte dataene uten å utføre dataomforming ved mottak av overføringssignalet matet ut fra styringsenheten (5), og

hvor underkonverteren (330) omfatter minst én av:

en datakonverterenhet (50, FL) som er egnet til å utføre lineær dataomforming, og

en datainverterenhet (70, FL^{-1}) som er egnet til å utføre en dataomforming som er den inverse av den som utføres av datakonverterenheten (50, FL),

karakterisert ved at:

minst én av datakonverterenheten (50, FL) og datainverterenheten (70, FL^{-1}) er egnet til å utføre dataomformingen og til å motta overføringssignalet matet ut fra styringsenheten (5) og til å overføre minst én av dataene og nøkkelen uten dataomformingen i henhold til det mottatte overføringssignalet når styringsenheten matet ut overføringssignalet,

hvor:

a) styringsenheten (5) er egnet til å mate ut et nøkkeloverføringssignal og et maskesignal som overføringssignalene for overføring av den mottatte nøkkelen, og minst én av datakonverterenheten (50, FL) og datainverterenheten (70, FL^{-1}) er egnet til å overføre nøkkelen ved mottak av nøkkeloverføringssignalet og

maskesignalet matet ut fra styringsenheten (5), ved å nulle ut de mottatte dataene i overensstemmelse med det mottatte nøkkeloverføringssignalet og la den mottatte nøkkelen slippe eller passere gjennom i overensstemmelse med det mottatte maskesignalet,

eller

b) styringsenheten (5) er egnet til å mate ut et DATA TRANSFER-signal som er et dataoverføringssignal, som overføringssignalet for overføring av de mottatte dataene, og minst én av datakonverterenheten (50, FL) og data-inverterenheten (70, FL^{-1}) er egnet til å overføre dataene ved mottak av DATA TRANSFER-signalet matet ut fra styringsenheten (5), ved å nulle ut den mottatte nøkkelen og la de mottatte dataene slippe eller passere gjennom i overensstemmelse med det mottatte DATA TRANSFER-signalet.

2. Dataomformingsanordning ifølge krav 1, hvor datascrambleren (30) videre omfatter:

en hovedkonverter (320) som er egnet for mottak av dataene og for ikke-lineær omforming av de mottatte dataene,

hvor styringsenheten (5) er egnet til å mate ut dataoverføringssignalet som overføringssignalet når dataene skal overføres, og

hvor underkonverteren (330) er egnet til å motta dataoverføringssignalet matet ut fra styringsenheten (5) og dataene som er ikke-lineært omformet av hovedkonverteren (320), og til å overføre de mottatte dataene i overens-stemmelse med det mottatte dataoverføringssignalet.

3. Dataomformingsanordning ifølge krav 1, videre omfattende:

en nøkkelgenerator (20) som er egnet til å generere nøkkelen,

hvor styringsenheten (5) er egnet til å mate ut et nøkkeloverføringssignal som overføringssignalet dersom nøkkelen skal overføres, og

hvor underkonverteren (330) er egnet til å motta nøkkeloverførings-signalet matet ut fra styringsenheten (5) og nøkkelen generert av nøkkel-generatoren og til å overføre den mottatte nøkkelen i overensstemmelse med det mottatte nøkkeloverføringssignalet.

4. Dataomformingsanordning ifølge krav 3, hvor nøkkelgeneratoren (20) videre omfatter:

en mellomnøkkel-generator (40) som er egnet til å motta en hemmelig nøkkel og til å generere en mellomnøkkel basert på den mottatte hemmelige nøkkelen,

hvor underkonverteren (330), ved mottak av nøkkeloverføringssignalet matet ut fra styringsenheten (5), er egnet til å overføre mellomnøkkelen, generert av mellomnøkkel-generatoren (40), til hovedkonverteren (320) i overensstemmelse med det mottatte nøkkeloverføringssignalet,

hvor hovedkonverteren (320) er egnet til å gjenta omforming og utmating av mellomnøkkelen overført av underkonverteren (330) minst én gang,

hvor underkonverteren (330) er egnet til å gjenta omforming og utmating av mellomnøkkelen matet ut fra hovedkonverteren (320) minst én gang,

hvor minst én av hovedkonverteren (320) og underkonverteren (330) er egnet til å gjenta omforming og utmating av mellomnøkkelen minst én gang,

hvor hovedkonverteren (320) mater ut mellomnøkkelen matet ut fra minst én av hovedkonverteren (320) og underkonverteren (330) som en utnøkkel, og

hvor mellomnøkkel-generatoren (40) er egnet til å motta utnøkkelen matet ut fra hovedkonverteren (320) og med dette genererer en utvidet nøkkel som omfatter mellomnøkkelen og utnøkkelen.

5. Dataomformingsanordning ifølge krav 4, hvor mellomnøkkel-generatoren (40) omfatter:

en 6-1 KL-signalvelger (220) som er egnet til å velge ut én nøkkel av seks mottatte nøkler, og

et key KL-register (240) som inneholder den ene nøkkelen valgt av 6-1 KL-signalvelgeren (220) som mellomnøkkelen,

hvor 6-1 KL-signalvelgeren (220) er egnet til å motta en hemmelig nøkkel, å motta seks nøkler omfattende den hemmelige nøkkelen, mellom-nøkkelen inneholdt i key KL-registeret (240) og fire nøkler oppnådd gjennom rotasjonsskift av mellomnøkkelen inneholdt i key KL-registeret (240) med fire forskjellige tall, og å velge ut én nøkkel av de seks mottatte nøklene,

hvor key KL-registeret (240) inneholder en nøkkel valgt av 6-1 KL-signalvelgeren (220), og

hvor underkonverteren (339), ved mottak av nøkkeloverføringssignalet matet ut fra styringsenheten (5), mottar nøkkelen inneholdt i key KL-registeret (240) som mellomnøkkelen og overfører den mottatte mellomnøkkelen.

6. Dataomformingsanordning ifølge krav 4, hvor mellomnøkkel-generatoren (40) omfatter:

en 4-1-signalvelger (223) som er egnet til å velge ut én nøkkel av fire mottatte nøkler,

en 3-1 KL-signalvelger (222) som er egnet til å velge ut én nøkkel av tre mottatte nøkler, og

et key KL-register (240) som inneholder den ene nøkkelen valgt av 3-1 KL-signalvelgeren (222) som mellomnøkkelen,

hvor 4-1-signalvelgeren (223) er egnet til å motta fire nøkler oppnådd gjennom rotasjonsskift av mellomnøkkelen inneholdt i key KL-registeret (240) med fire forskjellige tall, og å velge ut én nøkkel av de fire mottatte nøklene,

hvor 3-1 KL-signalvelgeren (222) er egnet til å motta en hemmelig nøkkel, å motta tre nøkler omfattende den hemmelige nøkkelen, den ene nøkkelen valgt av 4-1-signalvelgeren (223) og mellomnøkkelen inneholdt i key KL-registeret (240), og å velge ut én nøkkel av de tre mottatte nøklene,

hvor key KL-registeret (240) inneholder en nøkkel valgt av 3-1 KL-signalvelgeren (222), og

hvor underkonverteren (330), ved mottak av nøkkeloverføringssignalet matet ut fra styringsenheten (5), mottar nøkkelen inneholdt i key KL-registeret (240) som mellomnøkkelen og overfører den mottatte mellomnøkkelen.

7. Dataomformingsanordning ifølge krav 4, hvor nøkkelgeneratoren (20) videre omfatter en nøkkelfordeler (210) som er egnet til å motta den utvidede nøkkelen generert av mellomnøkkel-generatoren (40) og en forhåndsbestemt konstant, og å skedulere en nøkkel for utmating av én av den mottatte utvidede nøkkelen og den mottatte forhåndsbestemte konstanten til minst én av hovedkonverteren (320) og underkonverteren (330) i overensstemmelse med en forhåndsbestemt betingelse.

8. Dataomformingsanordning ifølge krav 1, hvor underkonverteren (330) omfatter:

en 1/2-underkonverterenhet (90) som er egnet til å implementere dataomforming for lineær dataomforming samt datainvertering for en dataomforming som er den inverse av den lineære dataomformingen i en felles krets, og

hvor underkonverteren (330) er egnet til å omforme dataene ved bruk av 1/2-underkonverterenheten (90), til å motta overføringssignalet matet ut fra styringsenheten (5) dersom styringsenheten (5) matet ut overføringssignalet, og til å overføre minst én av nøkkelen og dataene i overensstemmelse med det mottatte overføringssignalet.

9. Dataomformingsanordning ifølge krav 1, hvor datakonverterenheten (50, FL) og datainverterenheten (70, FL^{-1}) er anordnet serielt, og

hvor én av datakonverterenheten (50, FL) og datainverterenheten (70, FL^{-1}) er egnet til å motta én av dataene omformet av den andre av datakonverterenheten (50, FL) og datainverterenheten (70, FL^{-1}), den overførte nøkkelen og de overførte dataene, og til å utføre én av dataomforming, nøkkeloverføring og dataoverføring ved bruk av den ene av de omformede dataene, den overførte nøkkelen og de overførte dataene som er mottatt.

10. Dataomformingsanordning ifølge krav 1, hvor dataomformingsanordningen er egnet til å motta én av en 128-bits nøkkel, en 192-bits nøkkel og en 256-bits nøkkel, og til å omforme de mottatte dataene ved bruk av de mottatte nøklene.

11. Dataomformingsanordning ifølge krav 1, videre omfattende:

en nøkkelgenerator (20) egnet for å generere en nøkkel,

hvor nøkkelgeneratoren (20) videre omfatter en mellomnøkkel-generator (40) som er egnet til å motta en hemmelig nøkkel, å generere en mellomnøkkel basert på den mottatte hemmelige nøkkelen, og å generere en utnøkkel basert på den genererte mellomnøkkelen ved bruk av hovedkonverteren (320) og underkonverteren (330).

12. Dataomformingsanordning ifølge krav 11, hvor mellomnøkkel-generatoren (40) omfatter:

en 6-1 KL-signalvelger (220) som er egnet til å motta seks nøkler og til å velge ut én nøkkel av de seks mottatte nøklene,

et key KL-register (240) som inneholder den ene nøkkelen valgt av 6-1 KL-signalvelgeren (220) som mellomnøkkelen,

en 6-1 KA-signalvelger (230) som er egnet til å velge ut én nøkkel av seks nøkler, og

et key KA-register (250) som inneholder den ene nøkkelen valgt av 6-1 KA-signalvelgeren (230) som utnøkkelen,

hvor 6-1 KL-signalvelgeren (220) er egnet til å motta en hemmelig nøkkel, å motta seks nøkler omfattende den hemmelige nøkkelen, mellom-nøkkelen inneholdt i key KL-registeret (240) samt fire nøkler oppnådd gjennom rotasjonsskift av mellomnøkkelen inneholdt i key KL-registeret (240) med fire forskjellige tall, og å velge ut én nøkkel av de seks mottatte nøklene,

hvor key KL-registeret (240) inneholder en nøkkel valgt av 6-1 KL-signal-velgeren (220) som en mellomnøkkel,

hvor 6-1 KA-signalvelgeren (230) er egnet til å motta en utnøkkel generert ved bruk av hovedkonverteren (320) og underkonverteren (330), til å motta seks nøkler omfattende den mottatte utnøkkelen, utnøkkelen inneholdt i key KA-registeret (240) samt fire nøkler oppnådd gjennom rotasjonsskift av ut-nøkkelen inneholdt i key KA-registeret (250) med fire forskjellige tall, og til å velge ut én nøkkel av de seks mottatte nøklene, og

hvor key KA-registeret (250) inneholder den ene nøkkelen valgt av 6-1 KA-signalvelgeren (230) som en utnøkkel.

13. Dataomformingsanordning ifølge krav 11, hvor mellomnøkkel-generatoren (40) omfatter:

en 2-1-signalvelger (224) som er egnet til å velge ut én nøkkel av to nøkler,

en 4-1-signalvelger (223) som er egnet til å velge ut én nøkkel av fire nøkler,

en 3-1 KL-signalvelger (222) som er egnet til å velge ut én nøkkel av tre nøkler,

et key KL-register (240) som inneholder den ene nøkkelen valgt av 3-1 KL-signalvelgeren (222) som en mellomnøkkel,

en 3-1 KA-signalvelger (232) som er egnet til å velge ut én nøkkel av tre nøkler, og

et key KA-register (250) som inneholder den ene nøkkelen valgt av 3-1 KA-signalvelgeren (232) som en utnøkkel,

hvor 2-1-signalvelgeren (224) er egnet til å velge ut én nøkkel av mellomnøkkelen inneholdt i key KL-registeret (240) og utnøkkelen inneholdt i key KA-registeret (250),

hvor 4-1-signalvelgeren (223) er egnet til å motta fire nøkler oppnådd gjennom rotasjonsskift av den ene nøkkelen valgt av 2-1-signalvelgeren (224) med fire forskjellige tall, og å velge ut én nøkkel av de fire mottatte nøklene,

hvor 3-1 KL-signalvelgeren (222) er egnet til å motta en hemmelig nøkkel, å motta tre nøkler omfattende den hemmelige nøkkelen, den ene nøkkelen valgt av 4-1-signalvelgeren (223) og mellomnøkkelen inneholdt i key KL-registeret (240), og å velge ut én nøkkel av de tre nøklene,

hvor key KL-registeret (240) inneholder den ene nøkkelen valgt av 3-1 KL-signalvelgeren (222) som en mellomnøkkel,

hvor 3-1 KA-signalvelgeren (232) er egnet til å motta en utnøkkel generert ved bruk av hovedkonverteren (320) og underkonverteren (330), til å motta tre nøkler omfattende den mottatte utnøkkelen, den ene nøkkelen valgt av 4-1-signalvelgeren (223) og utnøkkelen inneholdt i key KA-registeret (250), og til å velge ut én nøkkel av de tre nøklene, og

hvor key KA-registeret (250) inneholder den ene nøkkelen valgt av 3-1 KA-signalvelgeren (232) som en utnøkkel.

14. Dataomformingsanordning ifølge krav 11, hvor mellomnøkkelgeneratoren (40) omfatter:

en 2-1 KL-signalvelger (291) som er egnet til å velge ut én nøkkel av to nøkler,

et key KL-register (240) som inneholder den ene nøkkelen valgt av 2-1 KL-signalvelgeren (291),

en 2-1 KA-signalvelger (292) som er egnet til å velge ut én nøkkel av to nøkler,

et key KA-register (250) som inneholder den ene nøkkelen valgt av 2-1 KA-signalvelgeren (292),

en 2-1-signalvelger (227) som er egnet til å velge ut én nøkkel av to nøkler, og

en 8-1-signalvelger (228) som er egnet til å velge ut én nøkkel av åtte nøkler,

hvor 2-1 KL-signalvelgeren (291) er egnet til å motta en hemmelig nøkkel og å velge ut én nøkkel av den mottatte hemmelige nøkkelen og nøkkelen inneholdt i key KL-registeret (240),

hvor 2-1 KA-signalvelgeren (292) er egnet til å motta en utnøkkel generert ved bruk av hovedkonverteren (320) og underkonverteren (330), og til å velge ut én nøkkel av den mottatte utnøkkelen og nøkkelen inneholdt i key KA-registeret (250),

hvor 2-1-signalvelgeren (227) er egnet til å velge ut én nøkkel av to nøkler valgt av 2-1 KL-signalvelgeren (291) og 2-1 KA-signalvelgeren (292), og

hvor 8-1 KL-signalvelgeren (228) er egnet til å motta åtte nøkler oppnådd gjennom rotasjonsskift av den ene nøkkelen valgt av 2-1-signalvelgeren (227) med åtte forskjellige tall, og til å velge ut én nøkkel av de åtte mottatte nøklene.

15. Fremgangsmåte for dataomforming, egnet til å utføres av en anordning i henhold til enhver av krav 1-14, og egnet for:

å motta en nøkkel og data, samt å utføre dataomforming for minst én av kryptering og dekryptering av de mottatte dataene med bruk av den mottatte nøkkelen, idet fremgangsmåten videre omfatter trinn med å:

mate ut et overføringssignal som angir én av den mottatte nøkkelen og de mottatte dataene som skal overføres, i tilfellet med overføring av den ene av den mottatte nøkkelen og de mottatte dataene,

utføre dataomformingen for den ene av datakryptering og data-dekryptering ved å omforme de mottatte dataene med bruk av den mottatte nøkkelen, idet dataomformingen er en lineær dataomforming og en omforming som er invers med den lineære dataomformingen, og

overføre minst én av den mottatte nøkkelen og de mottatte dataene uten å utføre dataomformingen ved mottak av det utmatede overføringssignalet, k a r a k t e r i s e r t v e d at:

a) et nøkkeloverføringssignal og et maskesignal mates ut som overføringssignalene for overføring av den mottatte nøkkelen, og nøkkelen overføres ved mottak av nøkkeloverføringssignalet og maskesignalet matet ved å nulle ut de mottatte dataene i overensstemmelse med det mottatte nøkkeloverføringssignalet og å la den mottatte nøkkelen slippe eller passere gjennom i overensstemmelse med det mottatte maskesignalet,

eller

b) et DATA TRANSFER-signal som er et dataoverføringssignal blir utmattet som overføringssignalet for overføring av de mottatte dataene, og dataene overføres ved mottak av DATA TRANSFER-signalet ved å nulle ut den mottatte nøkkelen og å la de mottatte dataene slippe eller passere gjennom i overensstemmelse med det mottatte DATA TRANSFER-signalet.

Fig. 1

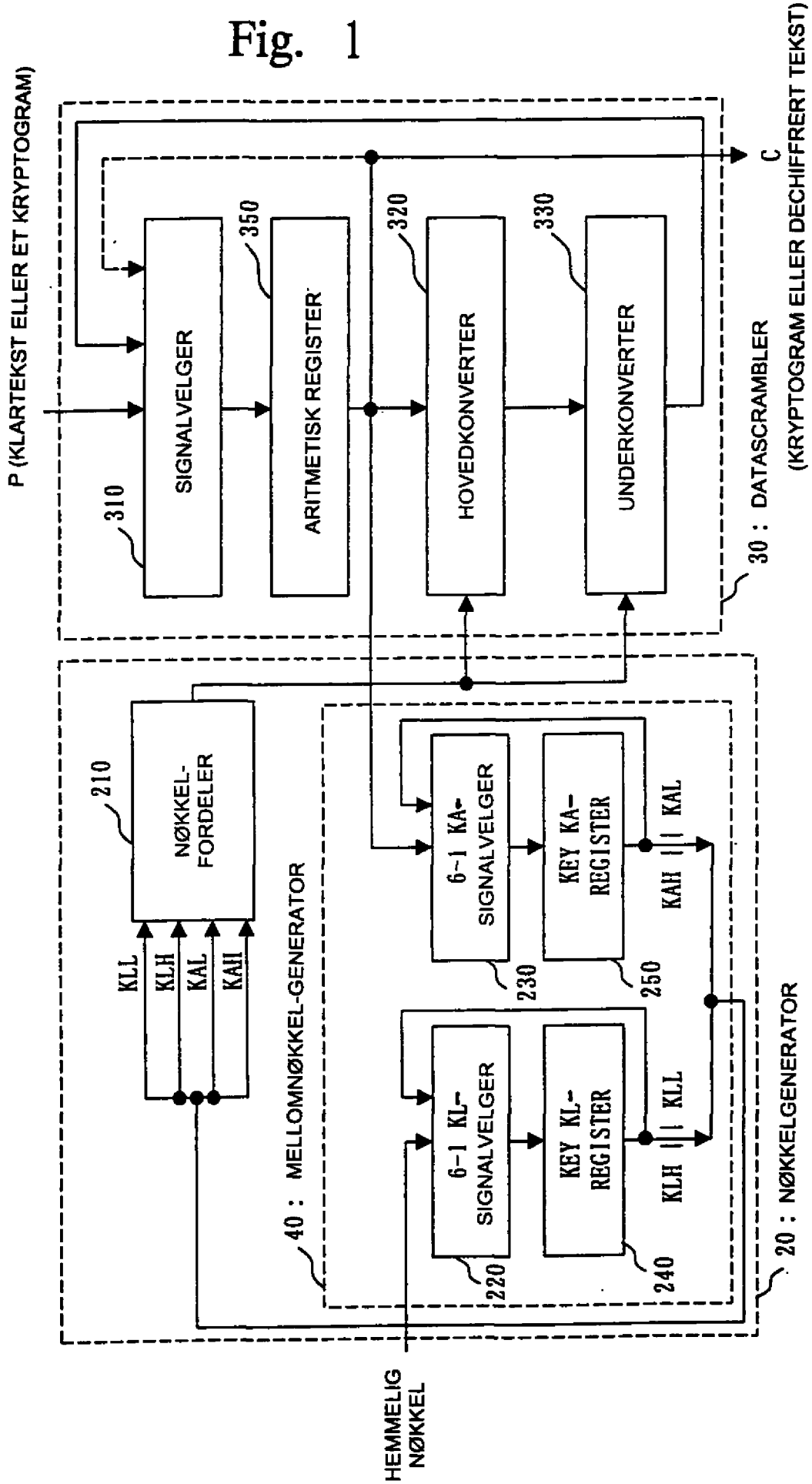

P (KLARTEKST ELLER ET KRYPTOGRAM)
C
(KRYPTOGRAM ELLER DECHIFFRERT TEKST)
30 : DATASCRAMBLER
310
SIGNALVELGER
350
ARITMETISK REGISTER
320
HOVEDKONVERTER
330
UNDERKONVERTER
20 : NØKKELGENERATOR
40 : MELLOMNØKKEL-GENERATOR
210
NØKKEL-FORDELER
KLL
KLH
KAL
KAH
6-1 KA-SIGNALVELGER
230
KEY KA-REGISTER
250
KAH
KAL
6-1 KL-SIGNALVELGER
220
KEY KL-REGISTER
240
KLH
KLL
HEMMELIG NØKKEL

Fig. 2

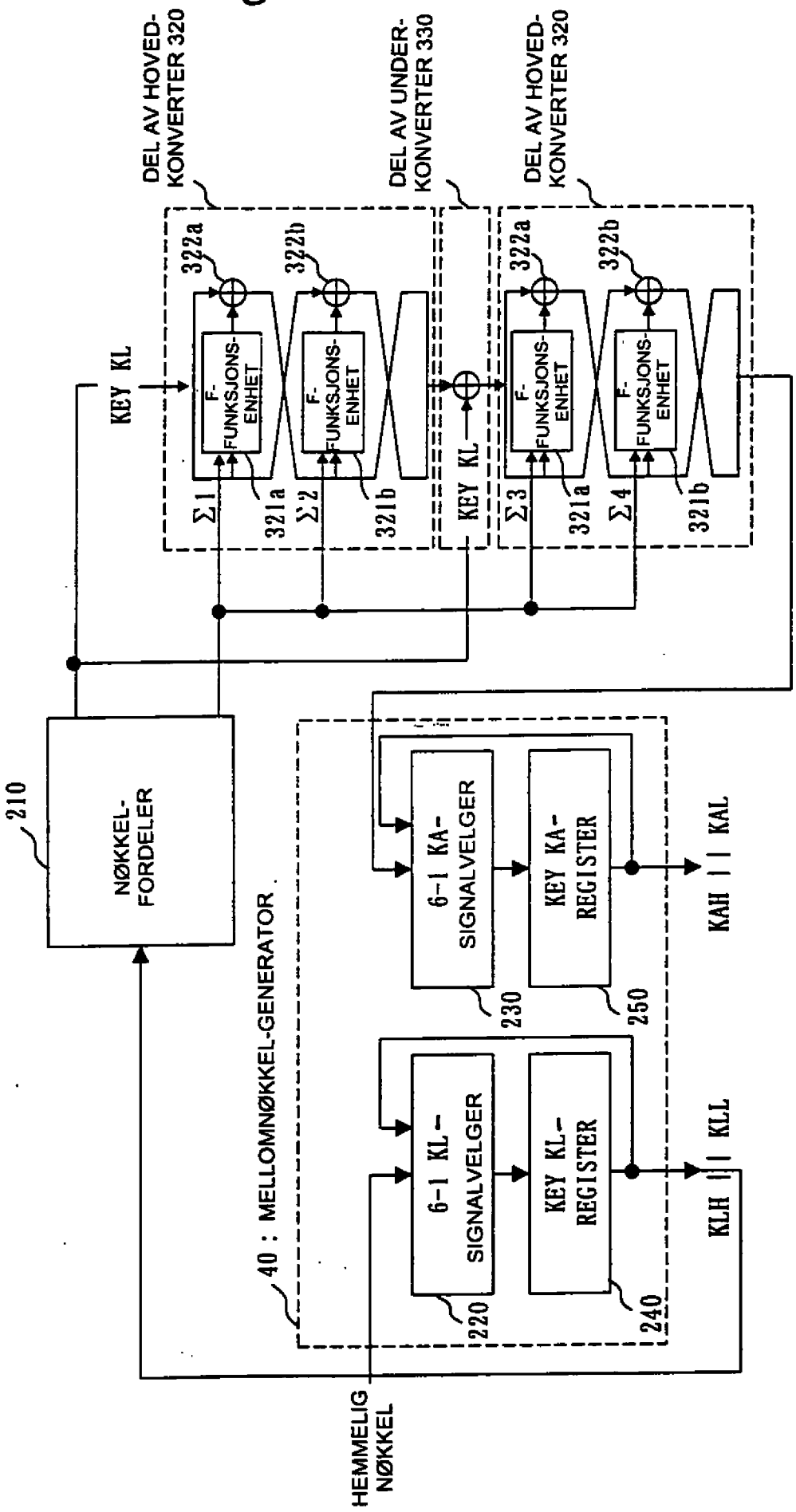

DEL AV HOVED-KONVERTER 320
DEL AV UNDER-KONVERTER 330
DEL AV HOVED-KONVERTER 320
322a
322b
F-FUNKSJONS-ENHET
KEY KL
Σ1
Σ2
Σ3
Σ4
321a
321b
210
NØKKEL-FORDELER
40 : MELLOMNØKKEL-GENERATOR
6-1 KA-SIGNALVELGER
KEY KA-REGISTER
KAH || KAL
230
250
6-1 KL-SIGNALVELGER
KEY KL-REGISTER
KLH || KLL
220
240
HEMMELIG NØKKEL

Fig. 3

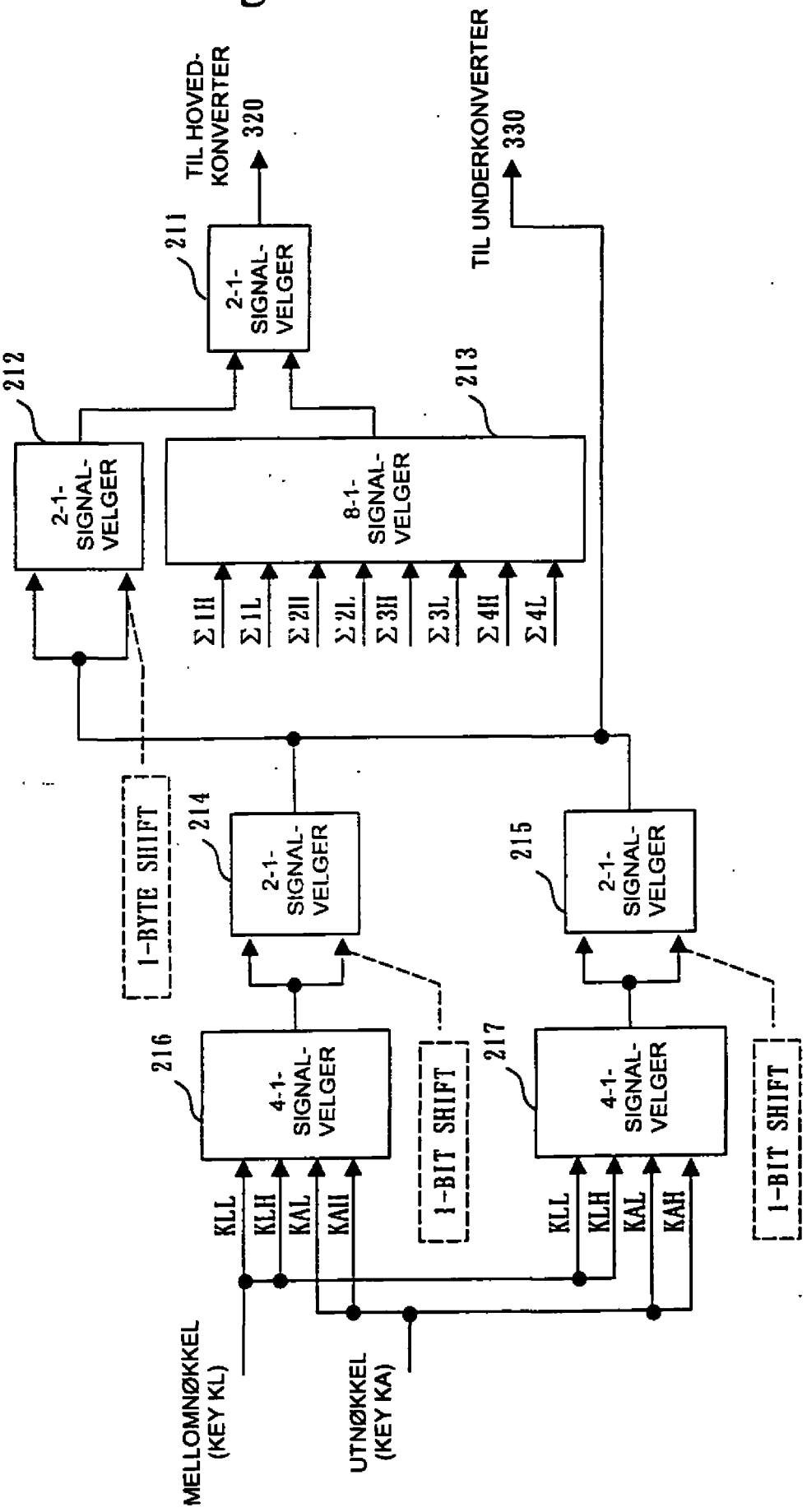

TIL HOVED-KONVERTER 320
TIL UNDERKONVERTER 330
211
2-1-SIGNAL-VELGER
212
2-1-SIGNAL-VELGER
213
8-1-SIGNAL-VELGER
Σ 1H
Σ 1L
Σ 2H
Σ 2L
Σ 3H
Σ 3L
Σ 4H
Σ 4L
1-BYTE SHIFT
214
2-1-SIGNAL-VELGER
215
2-1-SIGNAL-VELGER
216
4-1-SIGNAL-VELGER
1-BIT SHIFT
217
4-1-SIGNAL-VELGER
1-BIT SHIFT
KLL
KLH
KAL
KAH
KLL
KLH
KAL
KAH
MELLOMNØKKEL (KEY KL)
UTNØKKEL (KEY KA)

Fig. 4

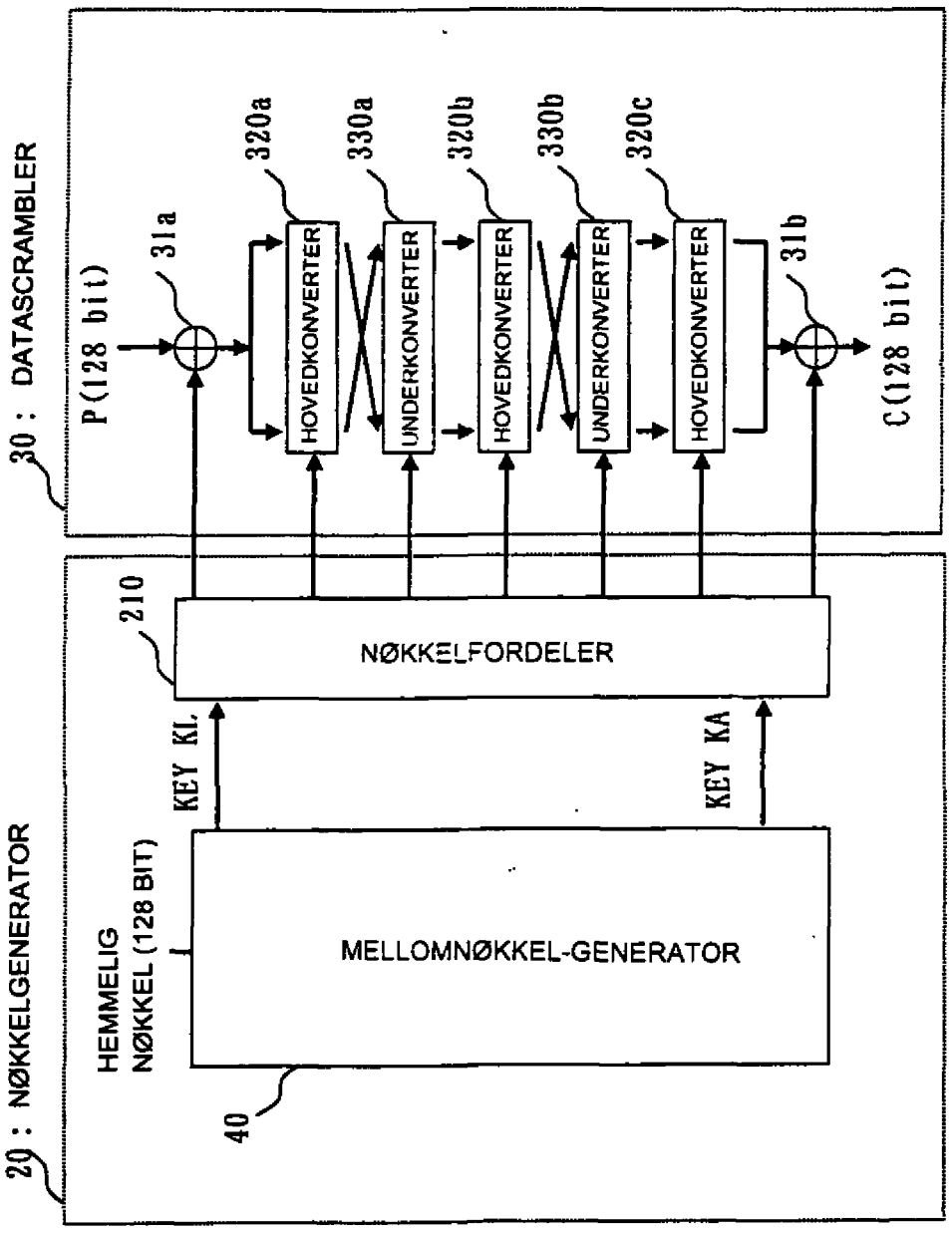

30 : DATASCRAMBLER
P(128 bit)
31a
320a
330a
320b
330b
320c
HOVEDKONVERTER
UNDERKONVERTER
HOVEDKONVERTER
UNDERKONVERTER
HOVEDKONVERTER
31b
C(128 bit)
210
NØKKELFORDELER
KEY KL
KEY KA
20 : NØKKELGENERATOR
HEMMELIG NØKKEL (128 BIT)
MELLOMNØKKEL-GENERATOR
40

Fig. 5

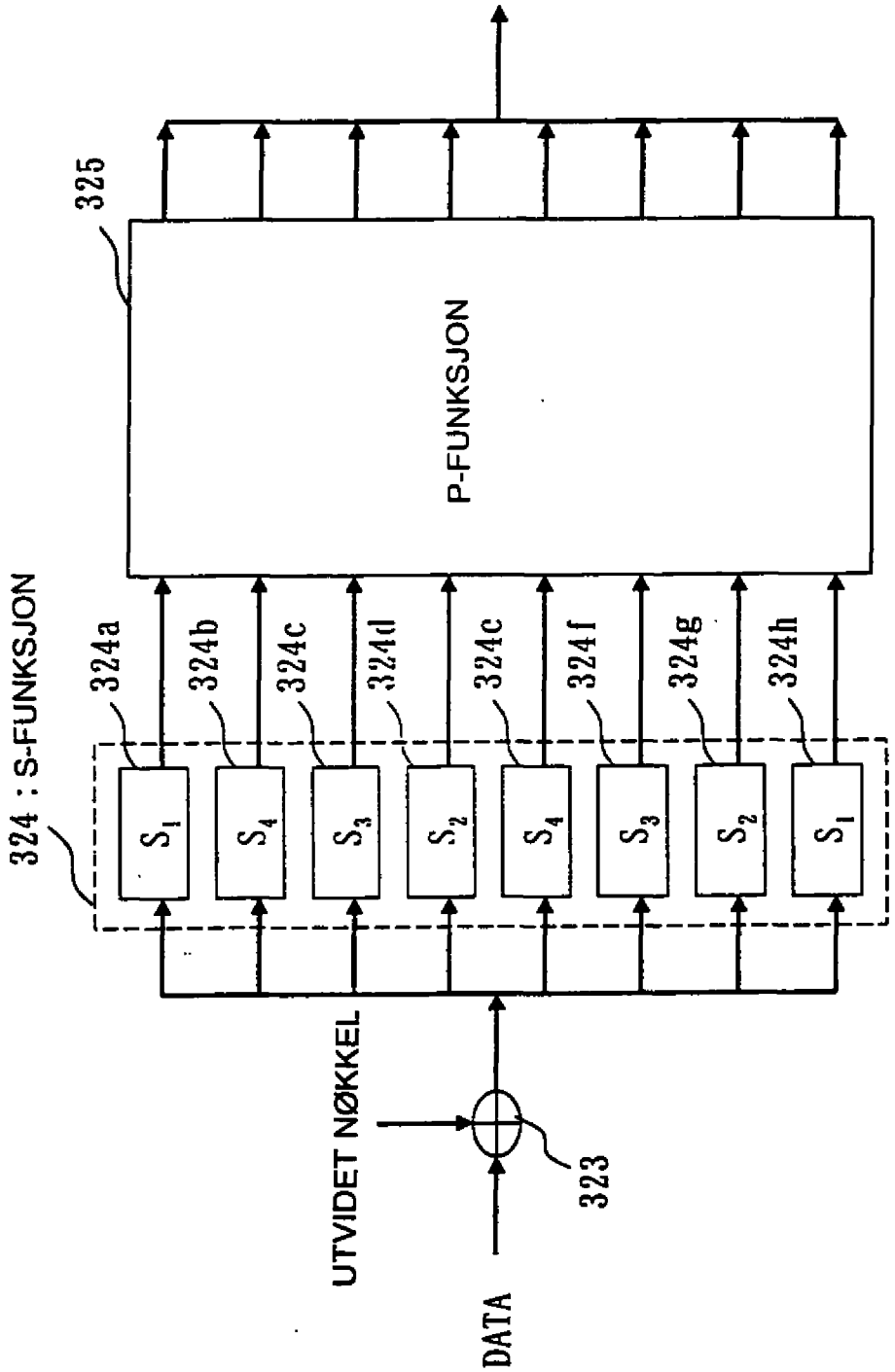

325
P-FUNKSJON
324 : S-FUNKSJON
324a
324b
324c
324d
324e
324f
324g
324h
S1
S4
S3
S2
S4
S3
S2
S1
UTVIDET NØKKEL
323
DATA

Fig. 6

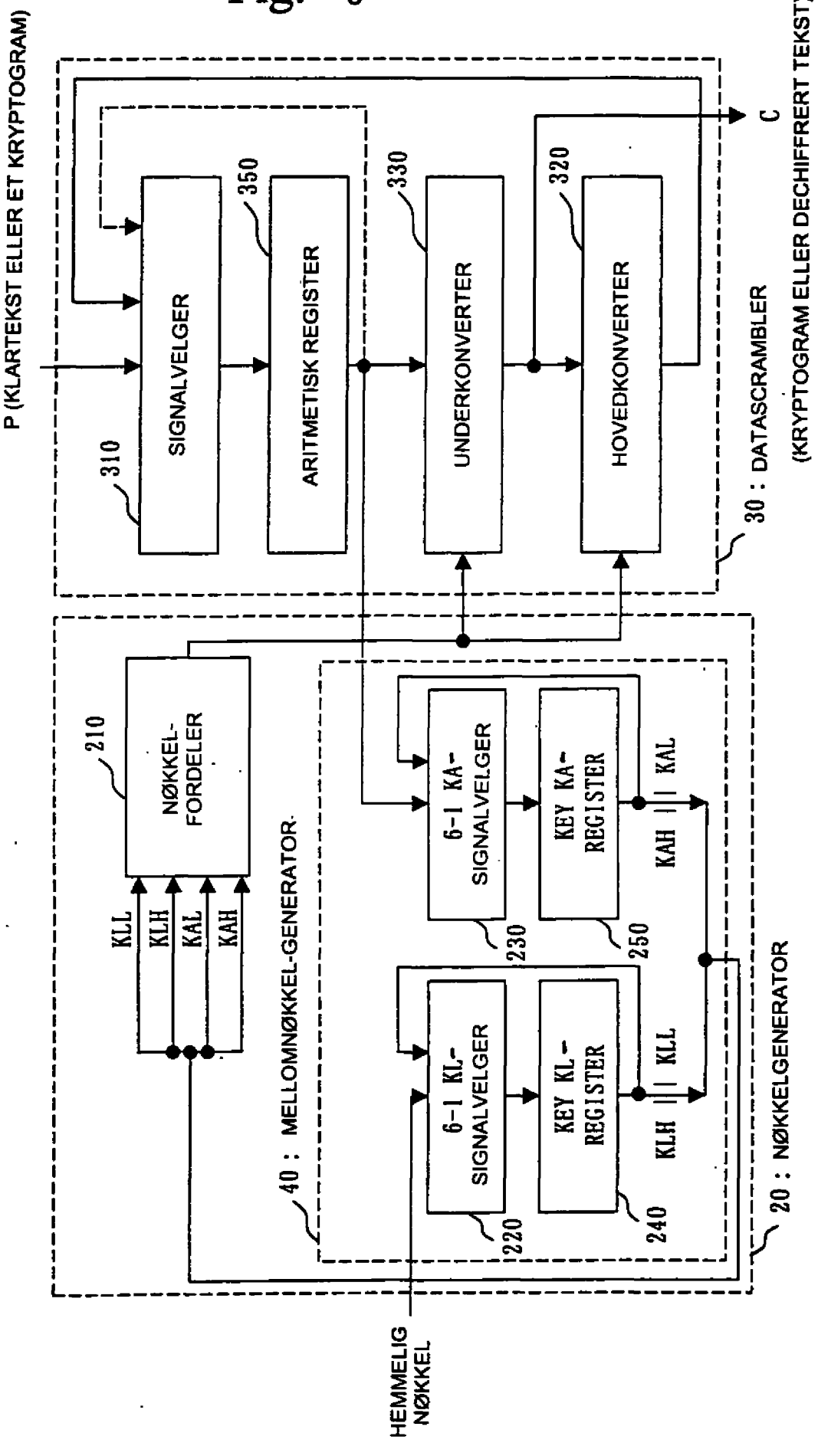

P (KLARTEKST ELLER ET KRYPTOGRAM)
SIGNALVELGER
ARITMETISK REGISTER
UNDERKONVERTER
HOVEDKONVERTER
310
350
330
320
C
30 : DATASCRAMBLER
(KRYPTOGRAM ELLER DECHIFFRERT TEKST)
NØKKEL-FORDELER
210
KLL
KLH
KAL
KAH
6-1 KA-SIGNALVELGER
KEY KA-REGISTER
KAH || KAL
230
250
6-1 KL-SIGNALVELGER
KEY KL-REGISTER
KLH || KLL
220
240
40 : MELLOMNØKKEL-GENERATOR
20 : NØKKELGENERATOR
HEMMELIG NØKKEL

Fig. 7

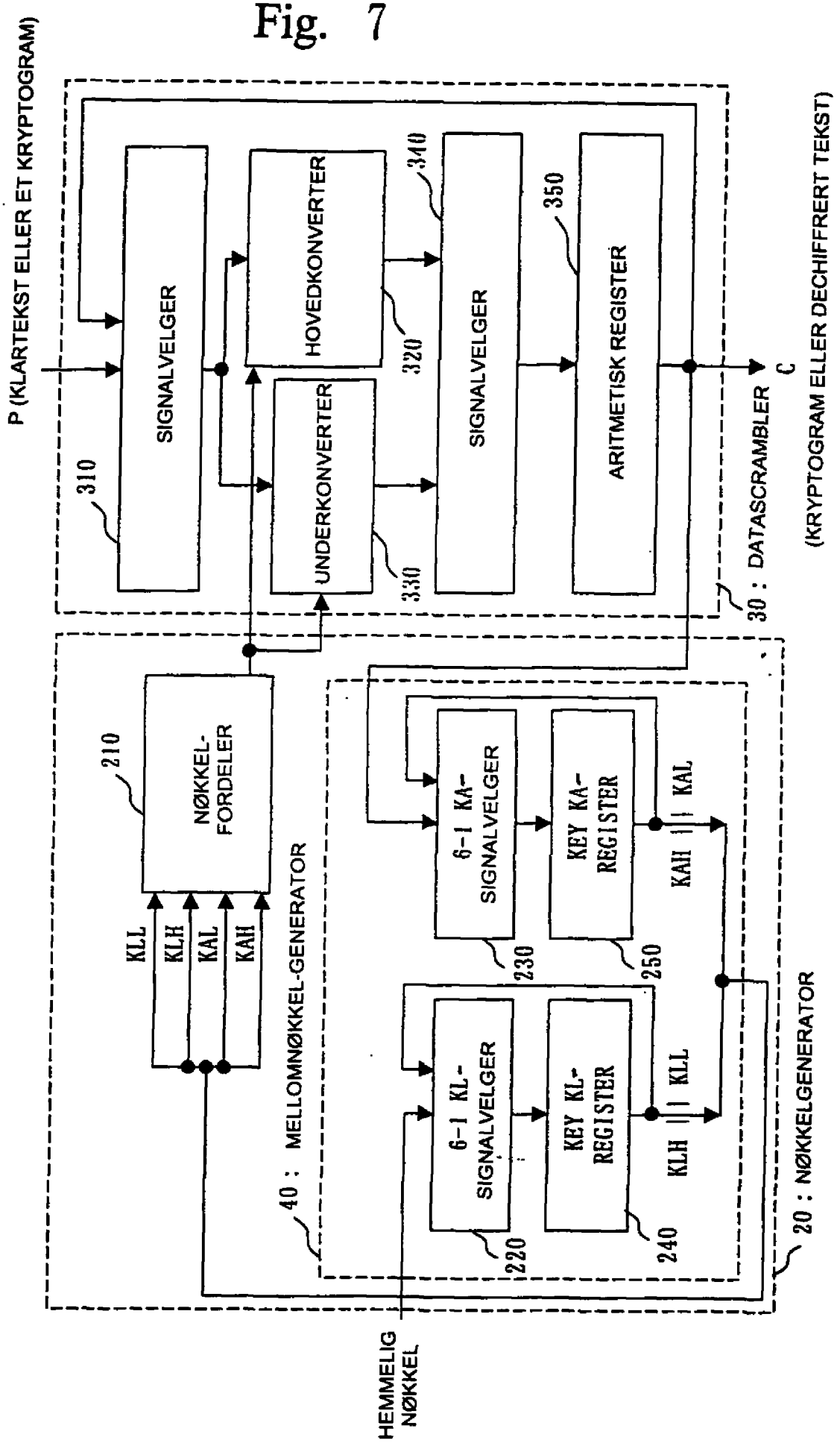

P (KLARTEKST ELLER ET KRYPTOGRAM)
SIGNALVELGER
310
HOVEDKONVERTER
320
UNDERKONVERTER
330
340
SIGNALVELGER
350
ARITMETISK REGISTER
C
(KRYPTOGRAM ELLER DECHIFFRERT TEKST)
30 : DATASCRAMBLER
NØKKEL-FORDELER
210
KLL
KLH
KAL
KAH
40 : MELLOMNØKKEL-GENERATOR
6-1 KA-SIGNALVELGER
230
KEY KA-REGISTER
250
KAH || KAL
6-1 KL-SIGNALVELGER
220
KEY KL-REGISTER
240
KLH || KLL
20 : NØKKELGENERATOR
HEMMELIG NØKKEL

Fig. 8

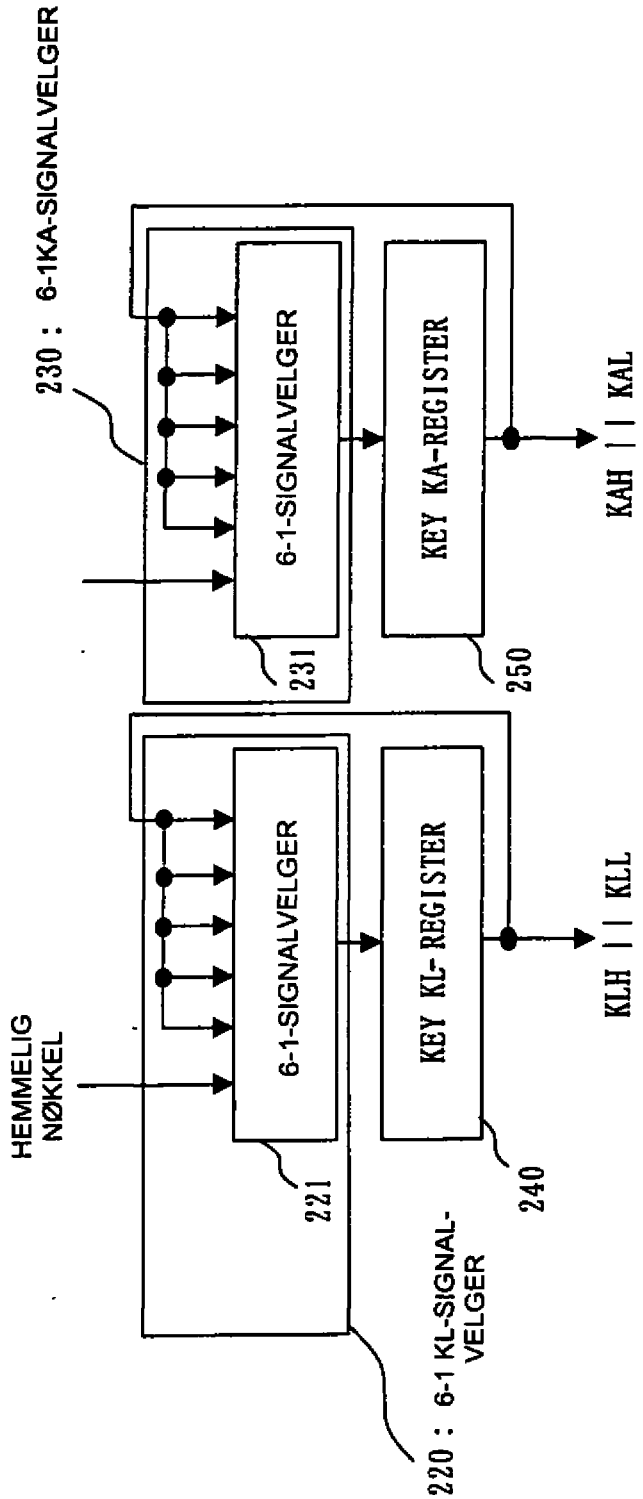

230 : 6-1KA-SIGNALVELGER
6-1-SIGNALVELGER
231
KEY KA-REGISTER
250
KAH || KAL
HEMMELIG NØKKEL
6-1-SIGNALVELGER
221
KEY KL- REGISTER
240
KLH || KLL
220 : 6-1 KL-SIGNAL-VELGER

Fig. 9

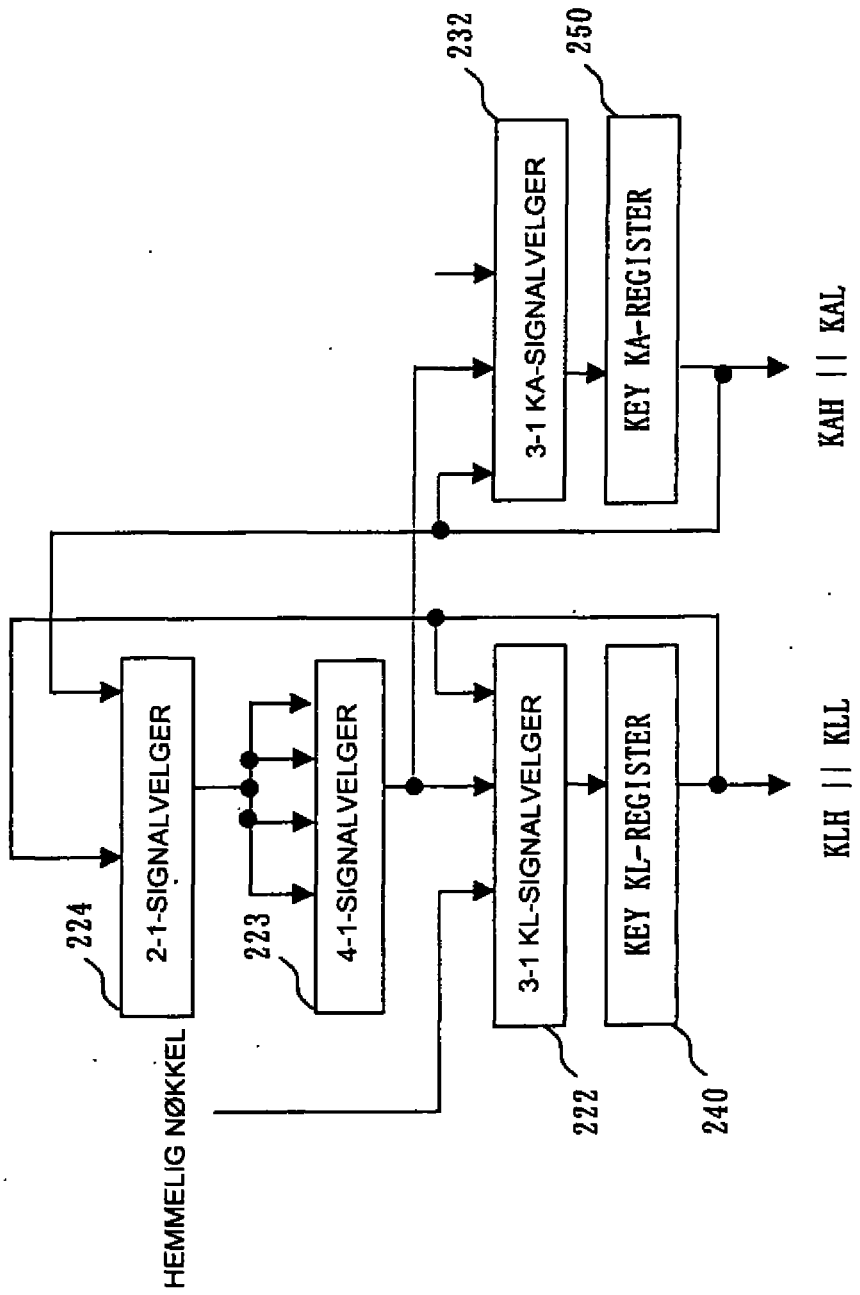

232
250
3-1 KA-SIGNALVELGER
KEY KA-REGISTER
KAH || KAL
2-1-SIGNALVELGER
4-1-SIGNALVELGER
3-1 KL-SIGNALVELGER
KEY KL-REGISTER
KLH || KLL
224
223
222
240
HEMMELIG NØKKEL

Fig. 10

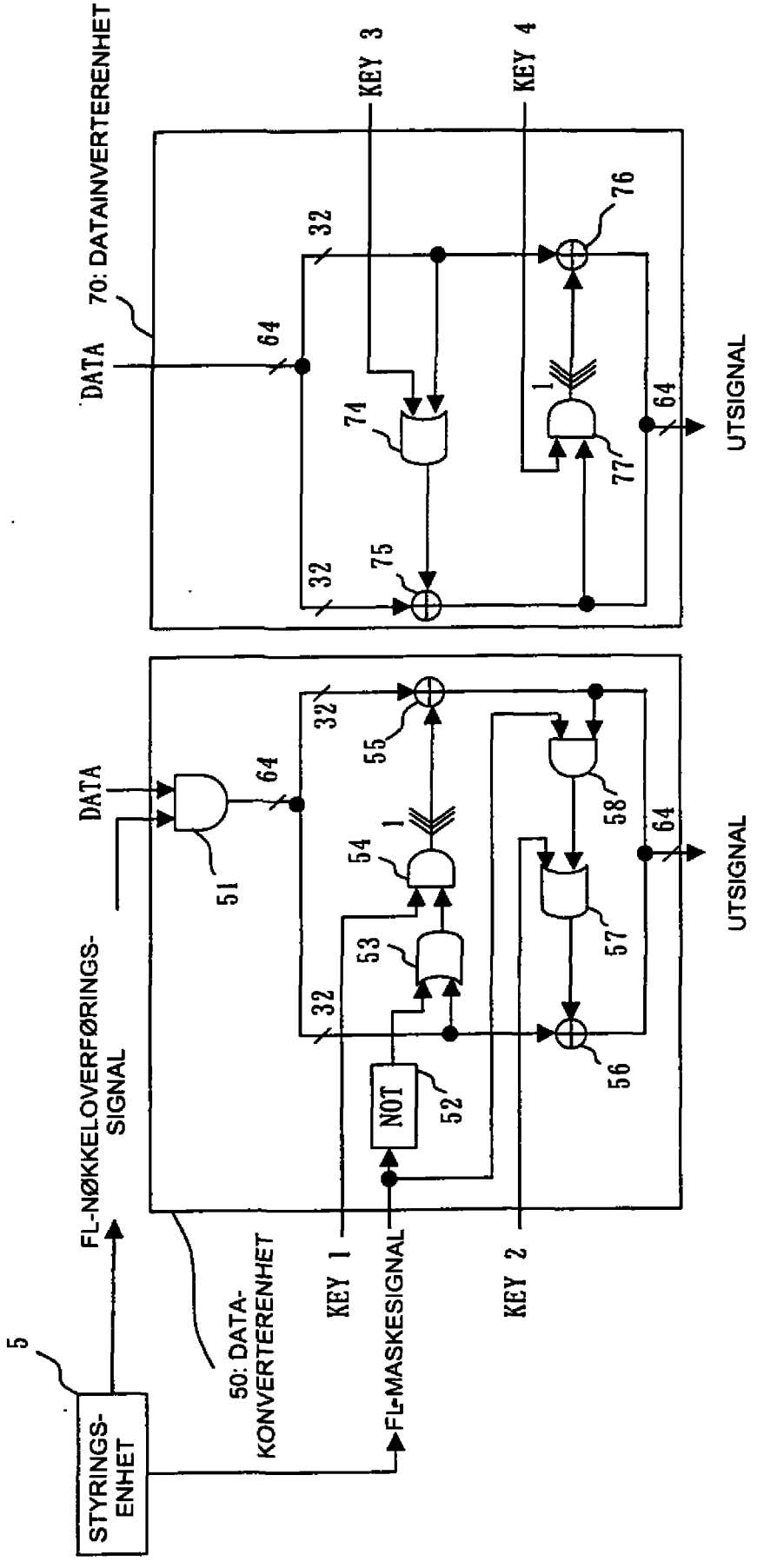

70: DATAINVERTERENHET
KEY 3
KEY 4
DATA
64
32
76
74
77
75
UTSIGNAL
DATA
51
55
54
53
58
57
56
52
NOT
FL-NØKKELOVERFØRINGS-SIGNAL
50: DATA-KONVERTERENHET
KEY 1
KEY 2
FL-MASKESIGNAL
5
STYRINGS-ENHET

Fig. 11

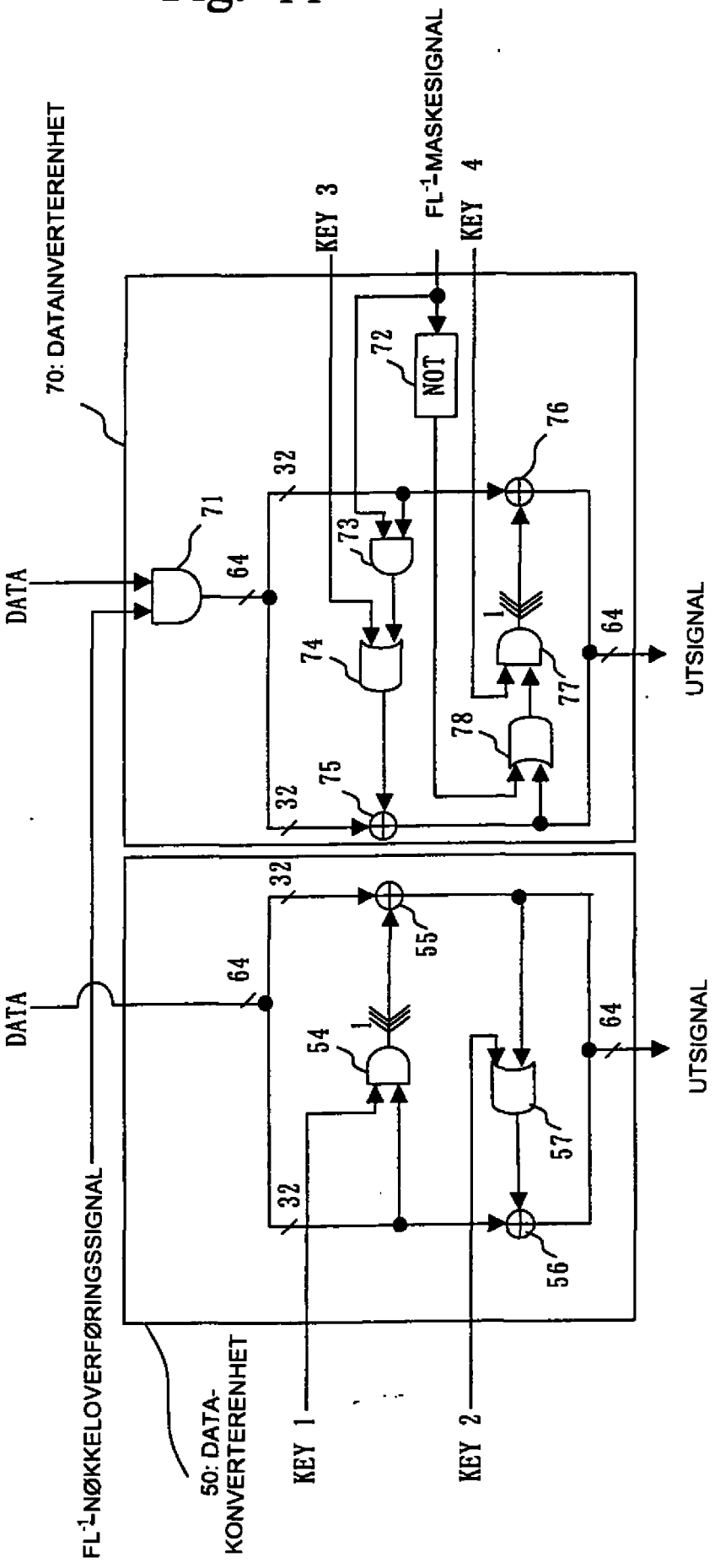

70: DATAINVERTERENHET
FL-1-MASKESIGNAL
KEY 3
KEY 4
NOT
72
76
32
71
73
64
DATA
74
75
77
78
64
UTSIGNAL
32
55
64
DATA
54
57
64
UTSIGNAL
32
56
FL-1-NØKKELOVERFØRINGSSIGNAL
50: DATA-KONVERTERENHET
KEY 1
KEY 2

Fig. 12

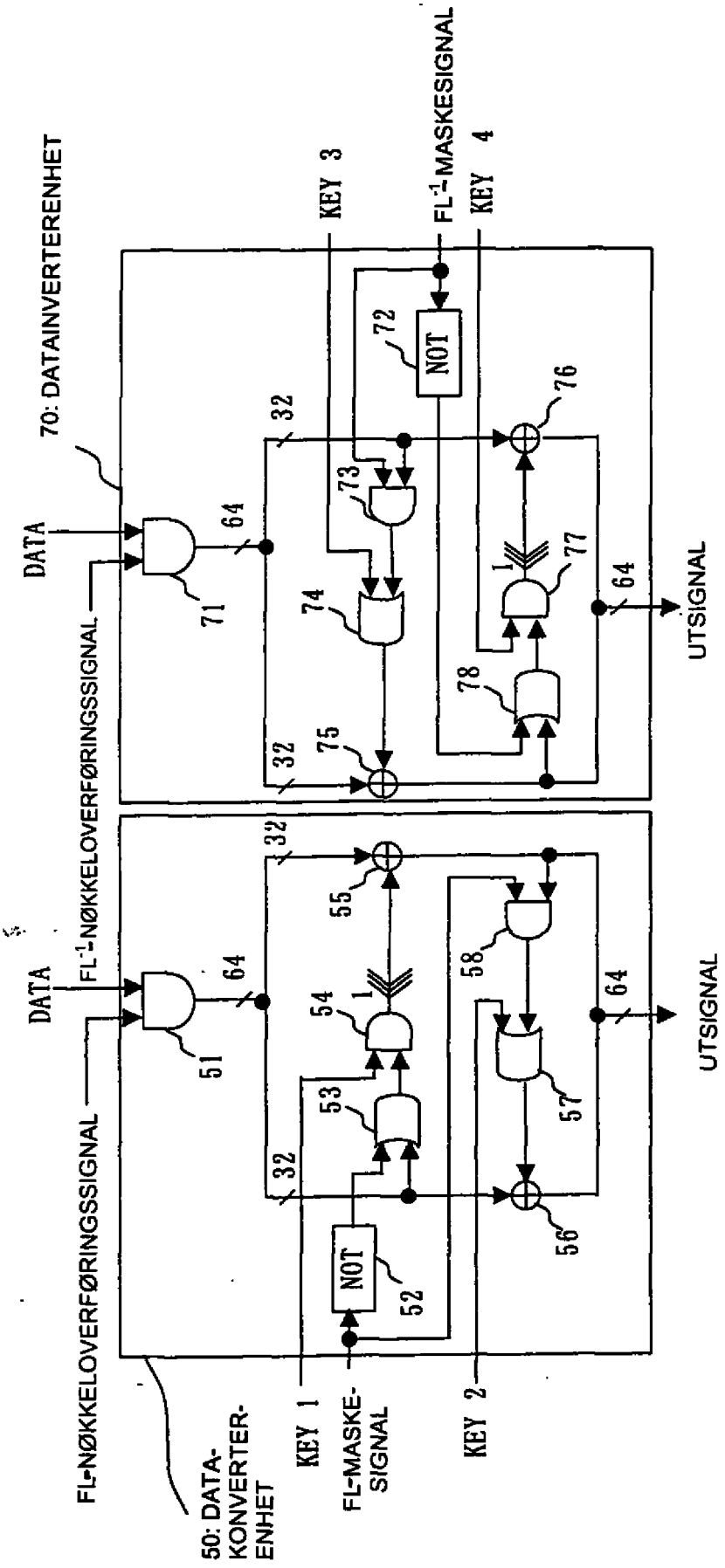

70: DATAINVERTERENHET
KEY 3
FL-1 MASKESIGNAL
KEY 4
DATA
64
71
32
73
72
NOT
76
74
77
1
64
UTSIGNAL
78
32
75
FL-1 NØKKELOVERFØRINGSSIGNAL
DATA
64
51
32
55
54
1
58
53
57
64
UTSIGNAL
32
56
NOT
52
FL-NØKKELOVERFØRINGSSIGNAL
50: DATA-KONVERTER-ENHET
KEY 1
FL-MASKE-SIGNAL
KEY 2

Fig. 13

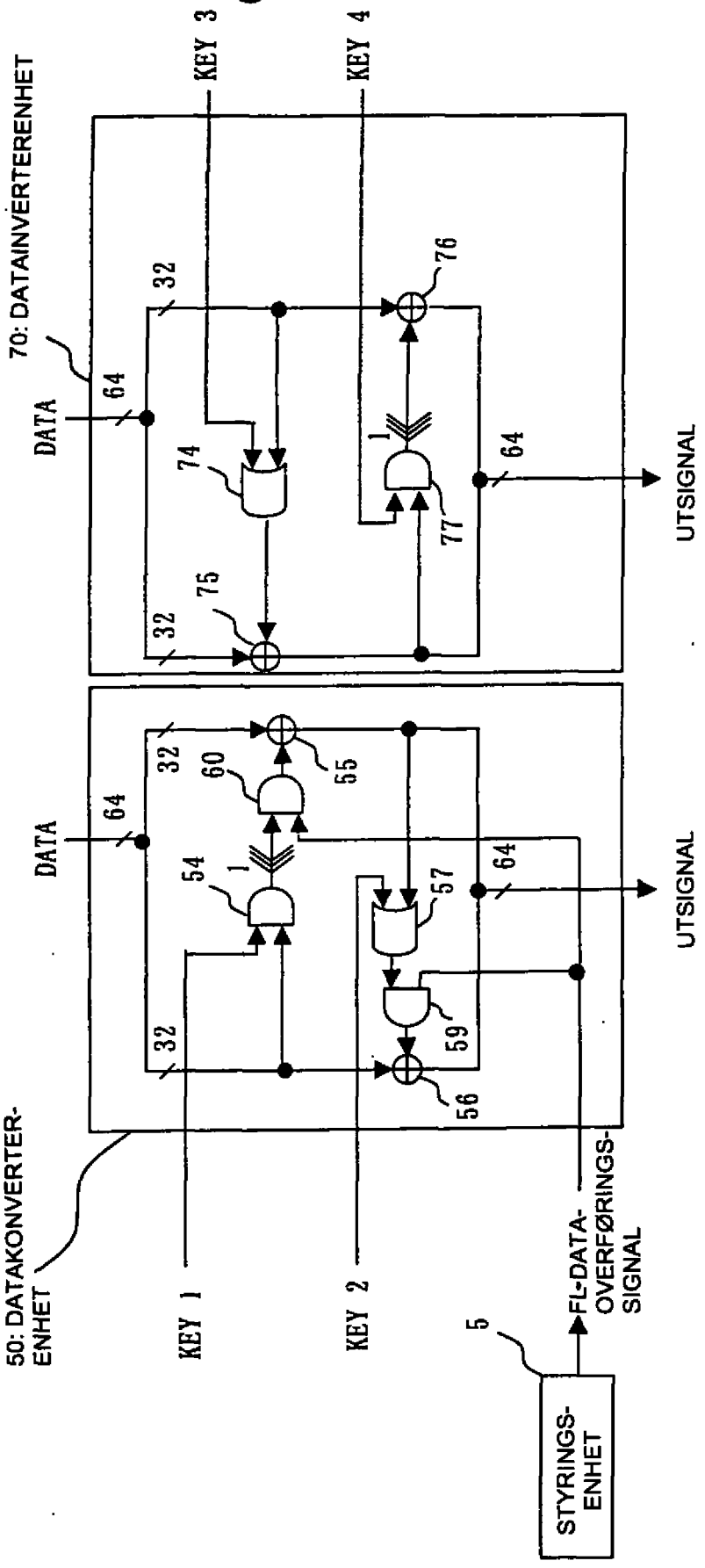

70: DATAINVERTERENHET
KEY 3
KEY 4
DATA
64
32
74
75
76
77
64
UTSIGNAL
DATA
64
32
60
65
54
57
64
UTSIGNAL
32
59
56
50: DATAKONVERTER-ENHET
KEY 1
KEY 2
5
FL-DATA-OVERFØRINGS-SIGNAL
STYRINGS-ENHET

Fig. 14

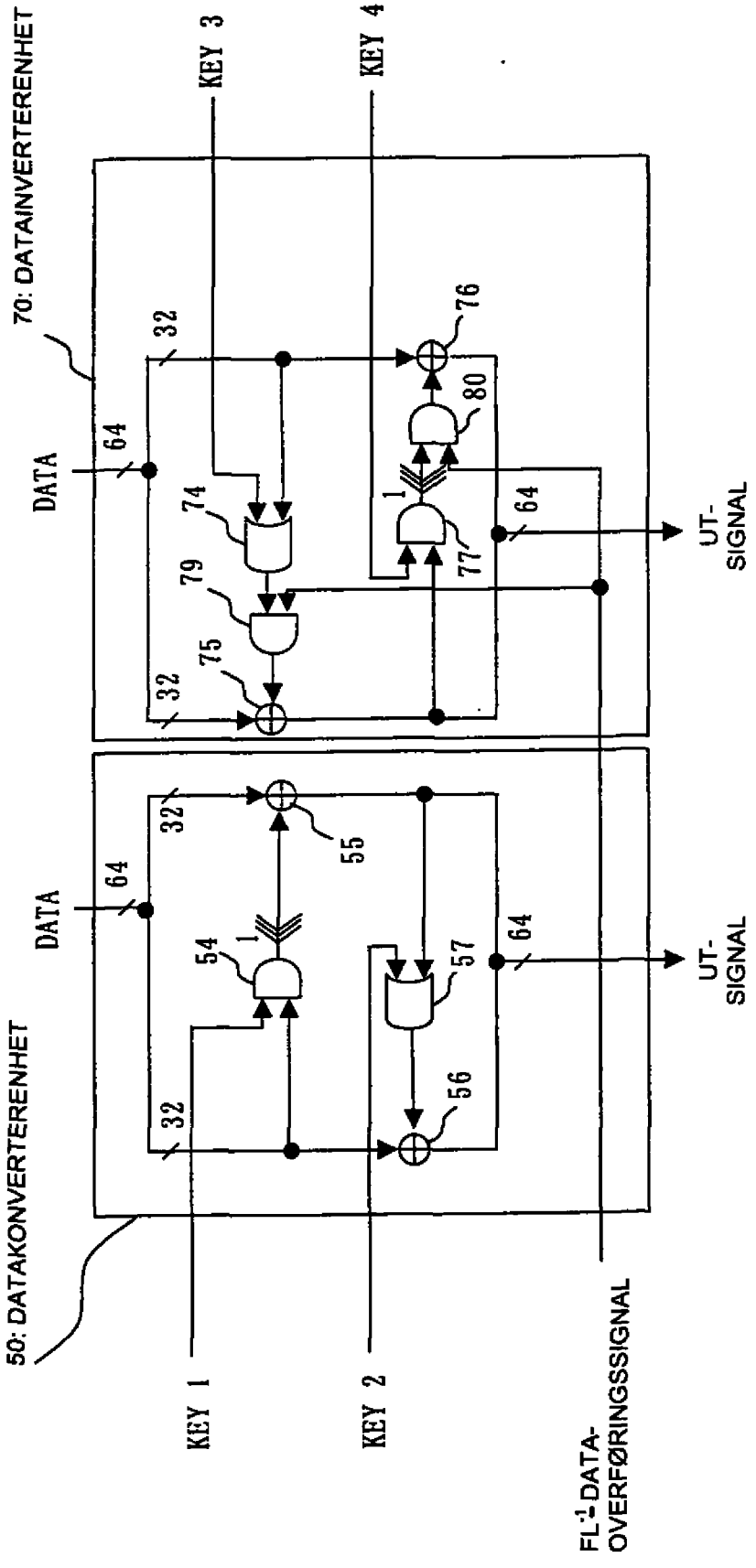

70: DATAINVERTERENHET
KEY 3
KEY 4
DATA
64
32
74
79
75
76
80
77
UT-SIGNAL
50: DATAKONVERTERENHET
KEY 1
KEY 2
54
55
56
57
FL-1 DATA-OVERFØRINGSSIGNAL

Fig. 15

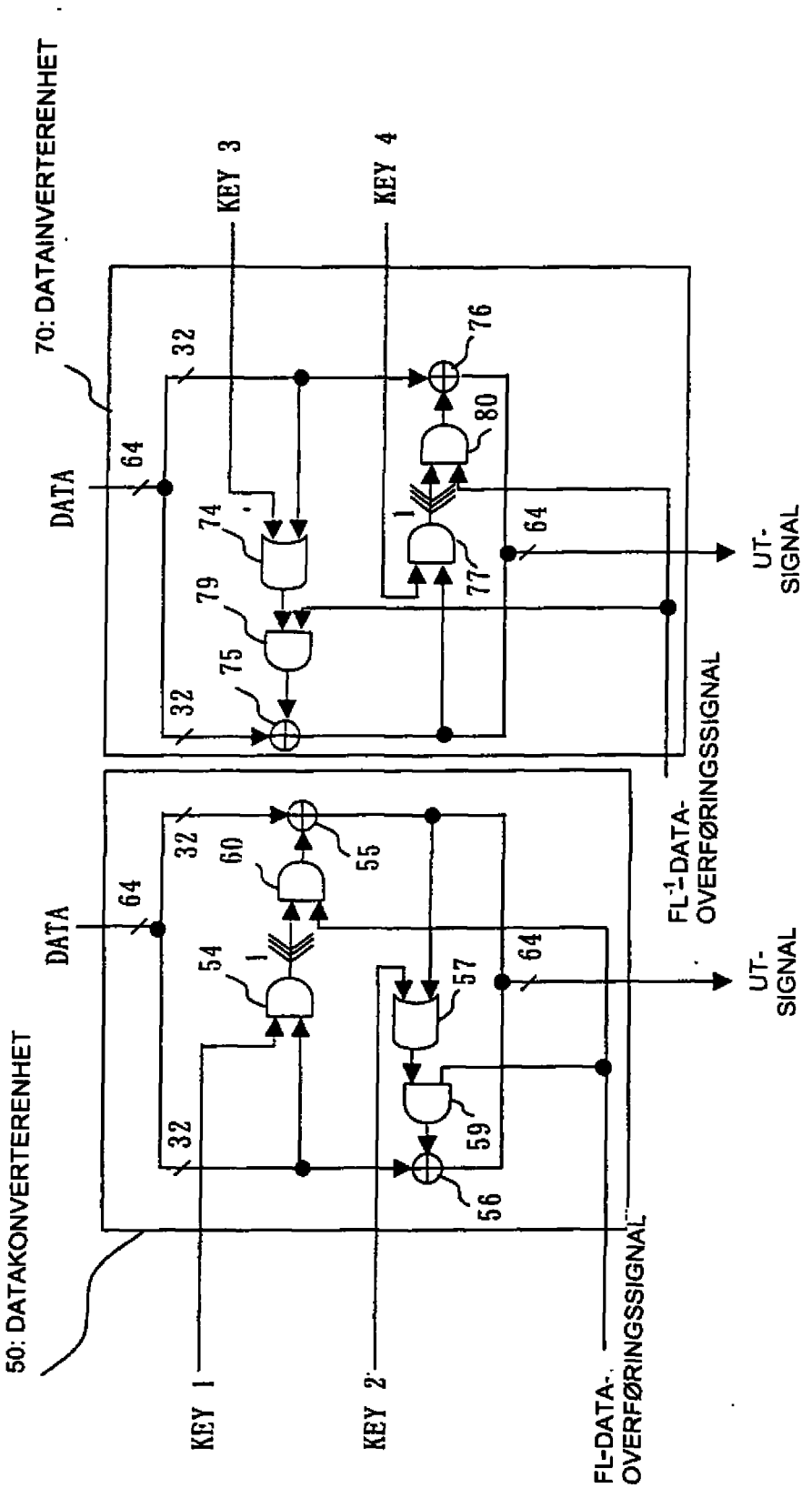

70: DATAINVERTERENHET
KEY 3
KEY 4
DATA
64
32
76
80
74
79
75
77
UT-SIGNAL
FL-1DATA-OVERFØRINGSSIGNAL
50: DATAKONVERTERENHET
60
55
54
57
59
56
KEY 1
KEY 2
FL-DATA-OVERFØRINGSSIGNAL

Fig. 16

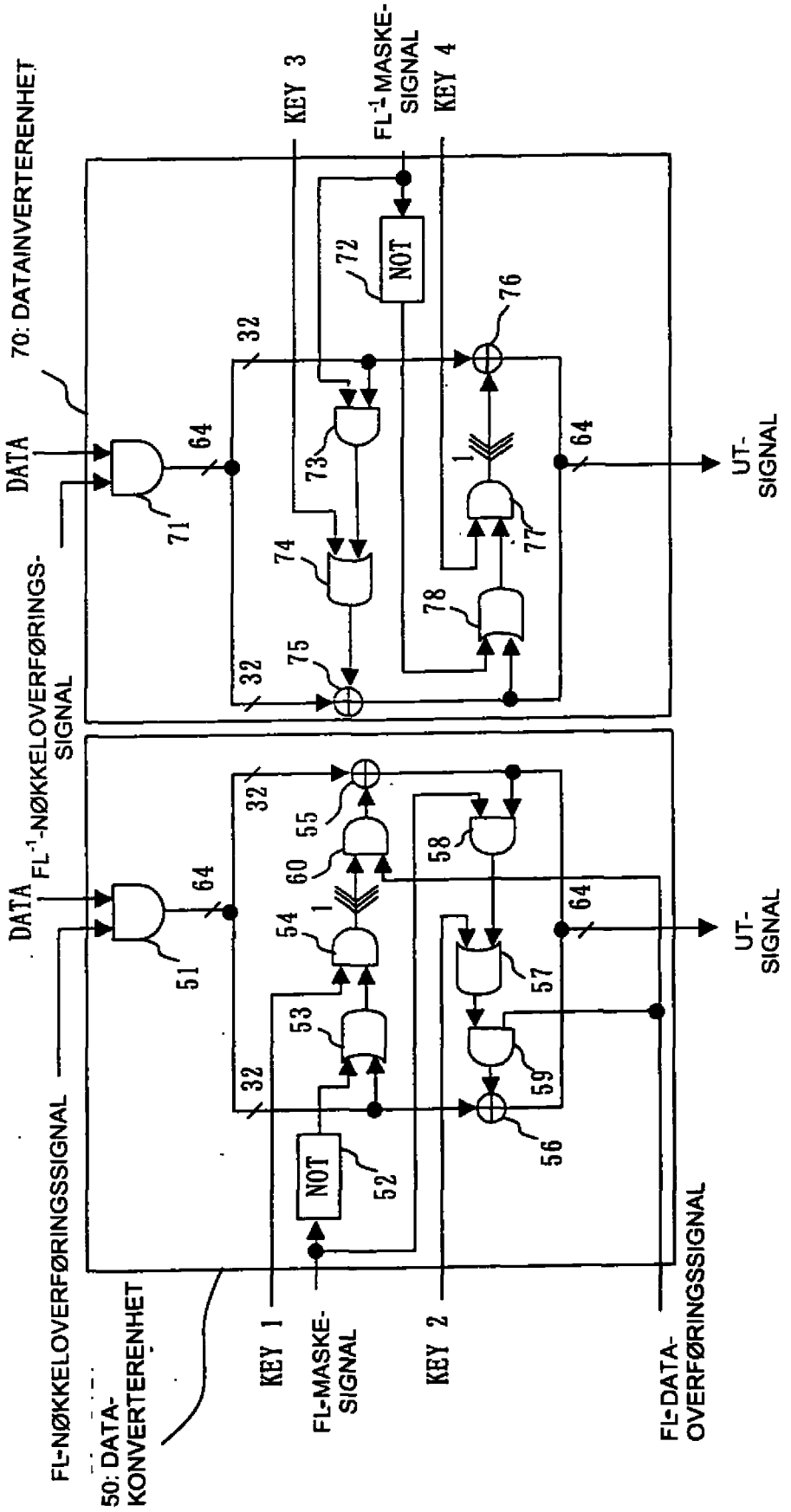

70: DATAINVERTERENHET
DATA
FL-1-NØKKELOVERFØRINGS-SIGNAL
KEY 3
FL-1 MASKE-SIGNAL
KEY 4
NOT
72
76
32
64
73
74
75
78
77
71
UT-SIGNAL
DATA
FL-1-NØKKELOVERFØRINGSSIGNAL
51
55
60
58
54
53
57
59
52
56
NOT
FL-NØKKELOVERFØRINGSSIGNAL
50: DATA-KONVERTERENHET
KEY 1
FL-MASKE-SIGNAL
KEY 2
FL-DATA-OVERFØRINGSSIGNAL

Fig. 17

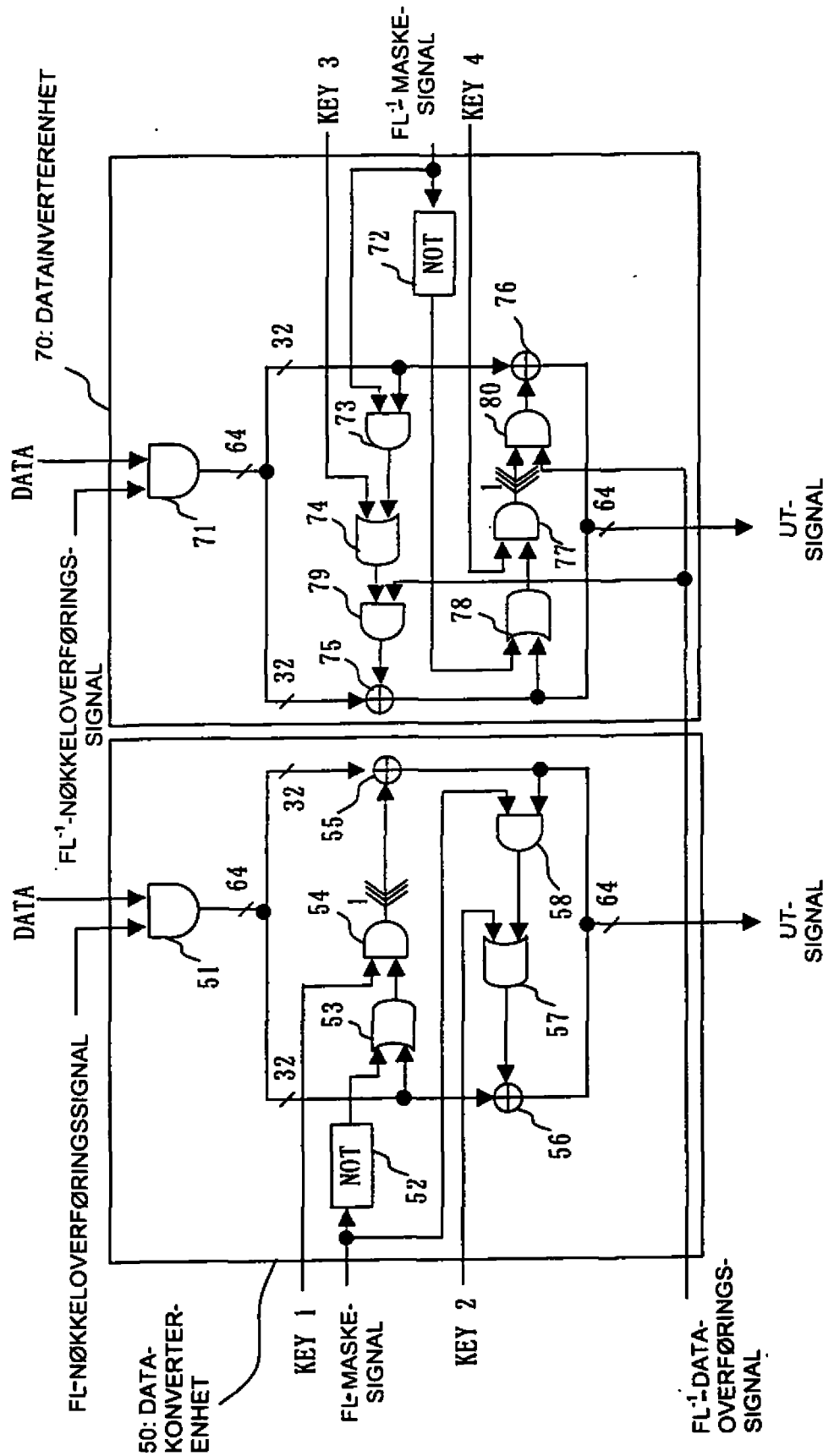

70: DATAINVERTERENHET
DATA
FL-1-NØKKELOVERFØRINGS-SIGNAL
KEY 3
FL-1 MASKE-SIGNAL
KEY 4
NOT
72
76
32
73
80
64
74
71
77
UT-SIGNAL
79
78
75
FL-NØKKELOVERFØRINGSSIGNAL
FL-1-NØKKELOVERFØRINGS-SIGNAL
55
58
54
57
53
56
NOT
52
50: DATA-KONVERTER-ENHET
KEY 1
FL-MASKE-SIGNAL
KEY 2
FL-1-DATA-OVERFØRINGS-SIGNAL

Fig. 18

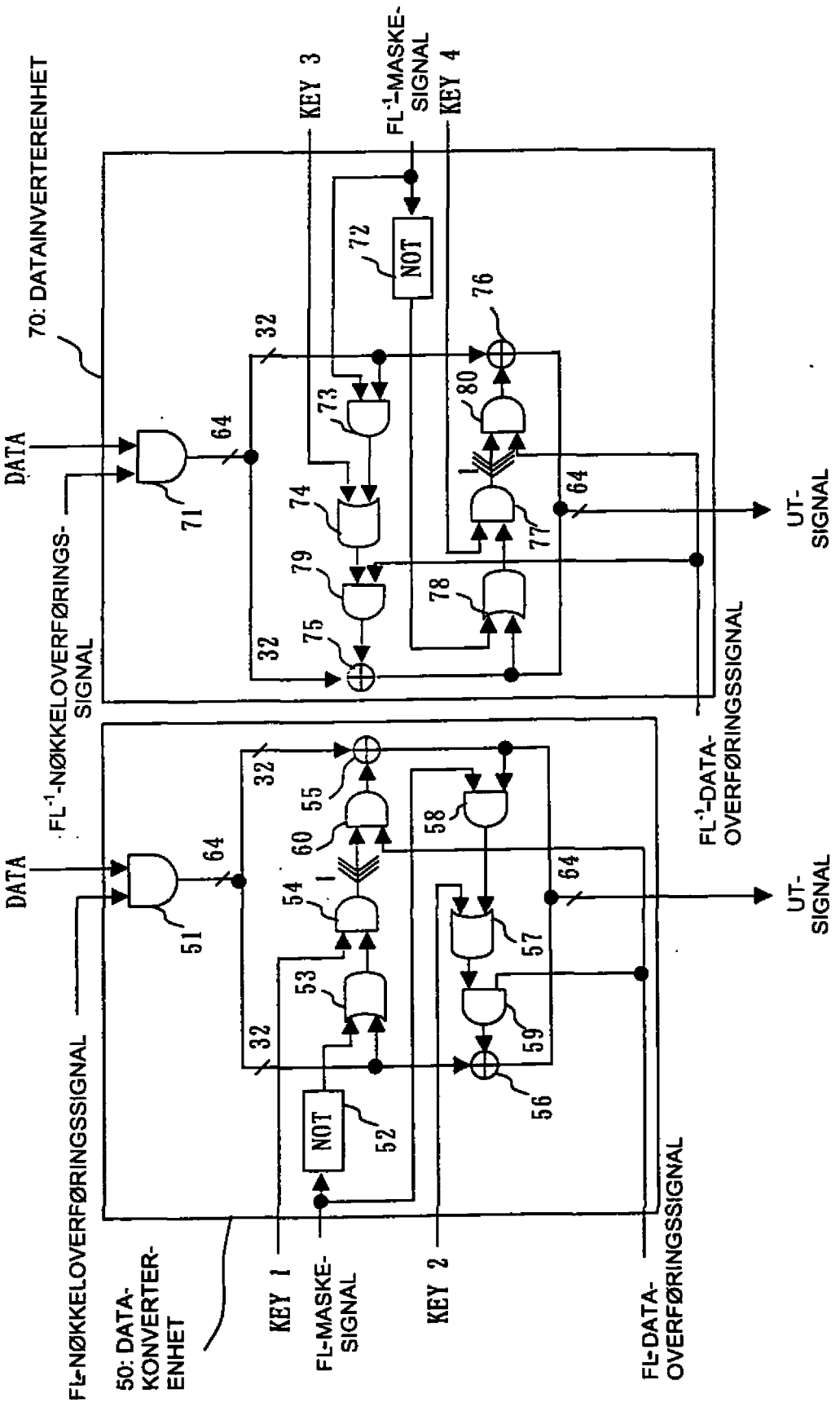

70: DATAINVERTERENHET
KEY 3
FL-1-MASKE-SIGNAL
KEY 4
DATA
FL-1-NØKKELOVERFØRINGS-SIGNAL
UT-SIGNAL
FL-1-DATA-OVERFØRINGSSIGNAL
NOT
50: DATA-KONVERTER-ENHET
KEY 1
FL-MASKE-SIGNAL
KEY 2
FL-NØKKELOVERFØRINGSSIGNAL
FL-DATA-OVERFØRINGSSIGNAL

Fig. 19

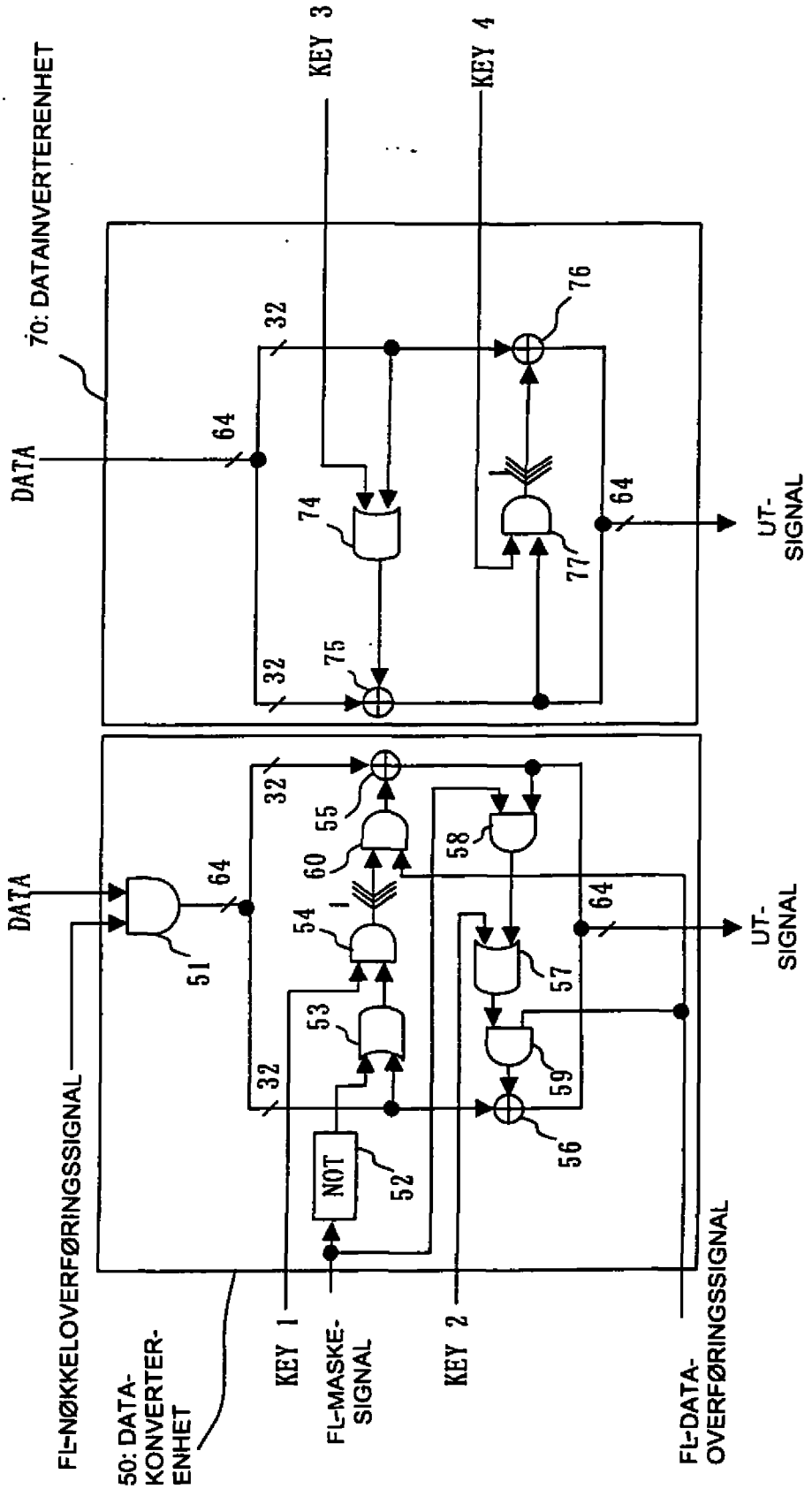

70: DATAINVERTERENHET
KEY 3
KEY 4
DATA
64
32
74
75
76
77
UT-SIGNAL
DATA
51
54
53
52
NOT
55
60
58
57
59
56
FL-NØKKELOVERFØRINGSSIGNAL
50: DATA-KONVERTER-ENHET
KEY 1
FL-MASKE-SIGNAL
KEY 2
FL-DATA-OVERFØRINGSSIGNAL

Fig. 20

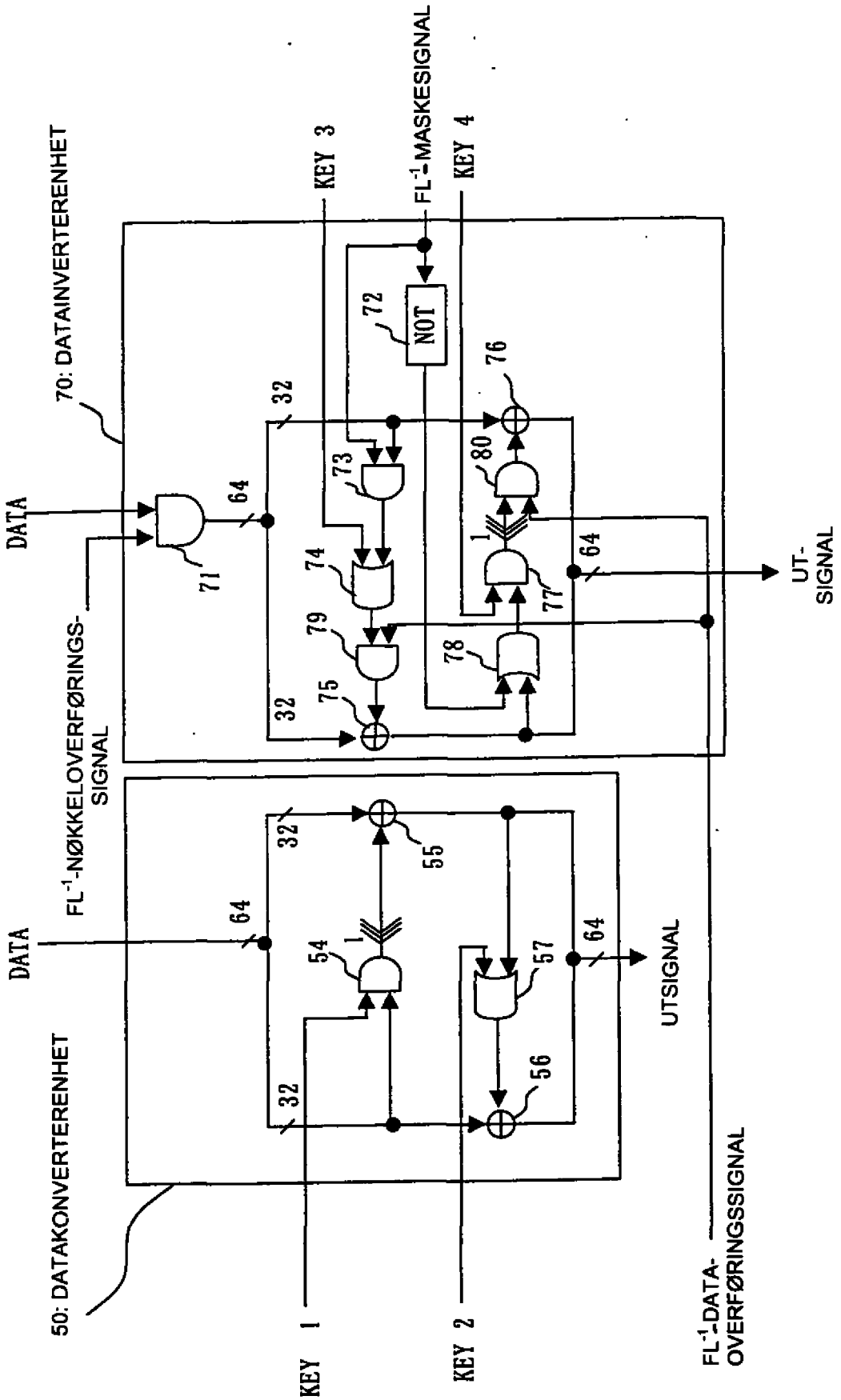

70: DATAINVERTERENHET
KEY 3
FL-1-MASKESIGNAL
KEY 4
72
NOT
76
32
80
73
DATA
64
71
74
1
64
UT-
SIGNAL
79
77
78
FL-1-NØKKELOVERFØRINGS-
SIGNAL
75
32
32
55
DATA
64
54
1
57
64
UTSIGNAL
56
32
50: DATAKONVERTERENHET
KEY 1
KEY 2
FL-1-DATA-
OVERFØRINGSSIGNAL

Fig. 21

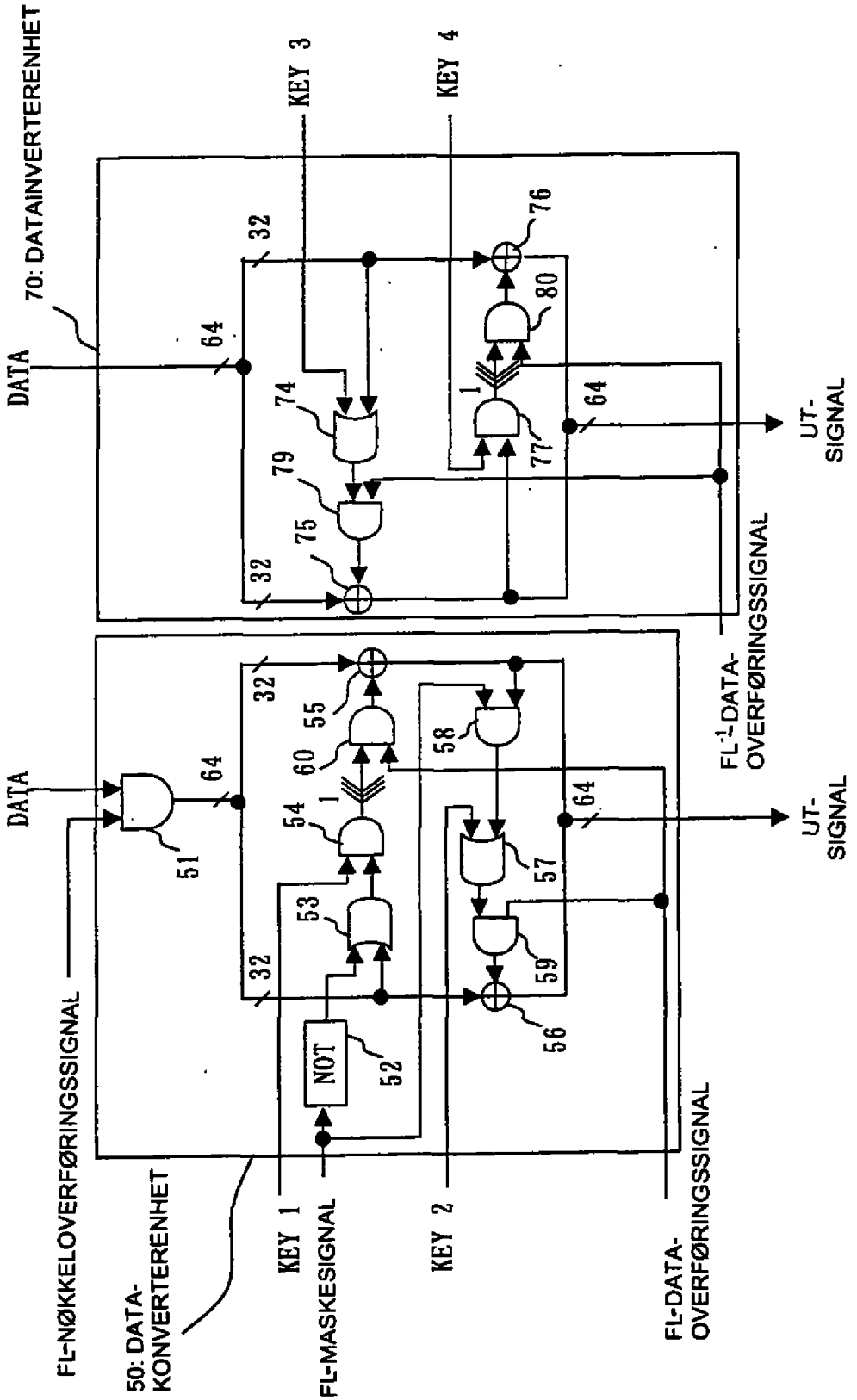

70: DATAINVERTERENHET
KEY 3
KEY 4
DATA
64
32
74
79
75
76
80
77
1
64
UT-SIGNAL
FL-1-DATA-OVERFØRINGSSIGNAL
DATA
64
51
32
55
60
54
1
53
58
57
59
56
64
UT-SIGNAL
NOT
52
FL-NØKKELOVERFØRINGSSIGNAL
50: DATA-KONVERTERENHET
KEY 1
FL-MASKESIGNAL
KEY 2
FL-DATA-OVERFØRINGSSIGNAL

Fig. 22

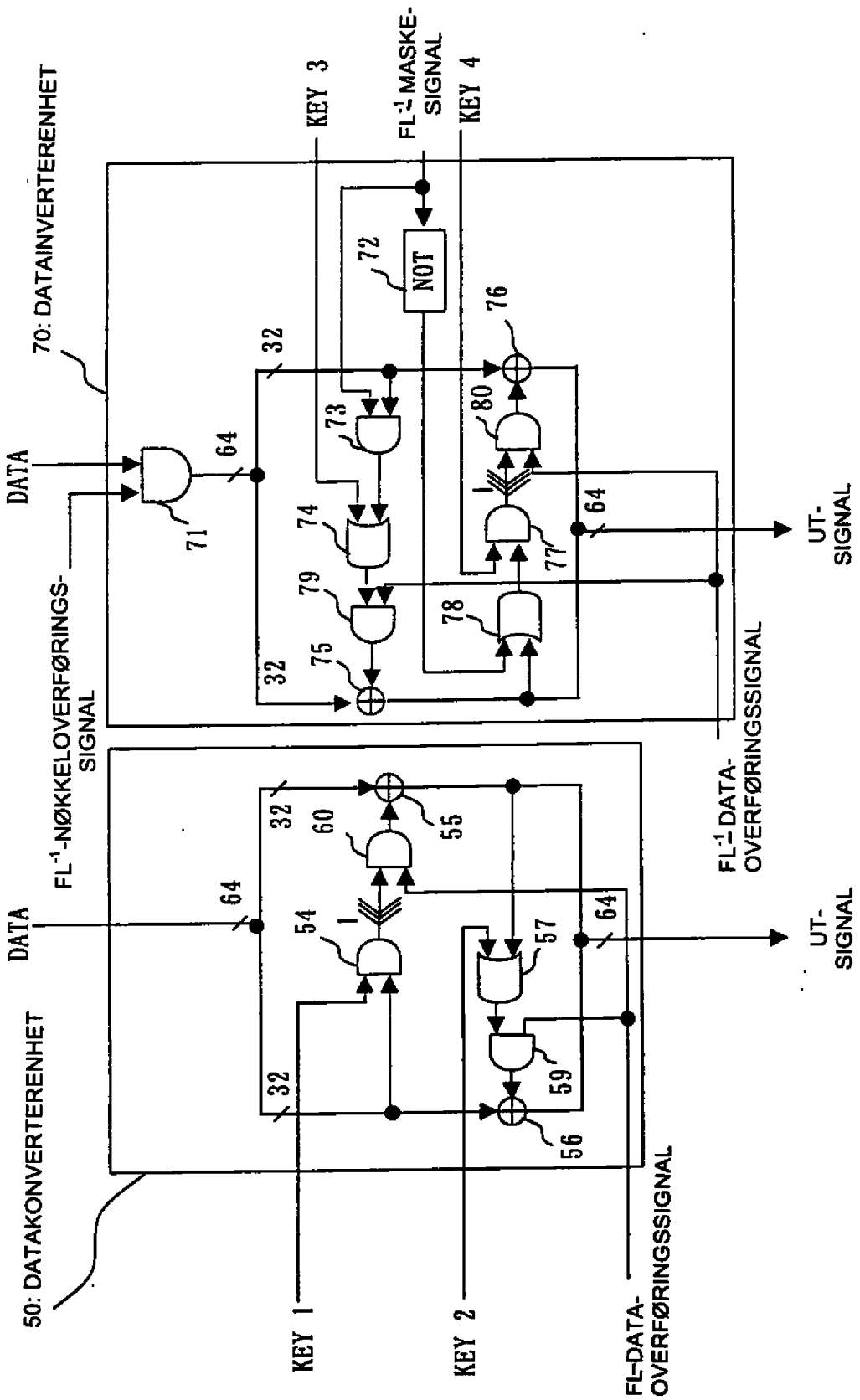

70: DATAINVERTERENHET
KEY 3
FL-1 MASKE-SIGNAL
KEY 4
NOT
72
76
32
73
80
DATA
64
71
74
64
UT-SIGNAL
79
77
78
FL-1-NØKKELOVERFØRINGS-SIGNAL
75
32
FL-1 DATA-OVERFØRINGSSIGNAL
32
60
55
64
DATA
54
57
64
UT-SIGNAL
59
32
56
50: DATAKONVERTERENHET
KEY 1
KEY 2
FL-DATA-OVERFØRINGSSIGNAL

Fig. 23

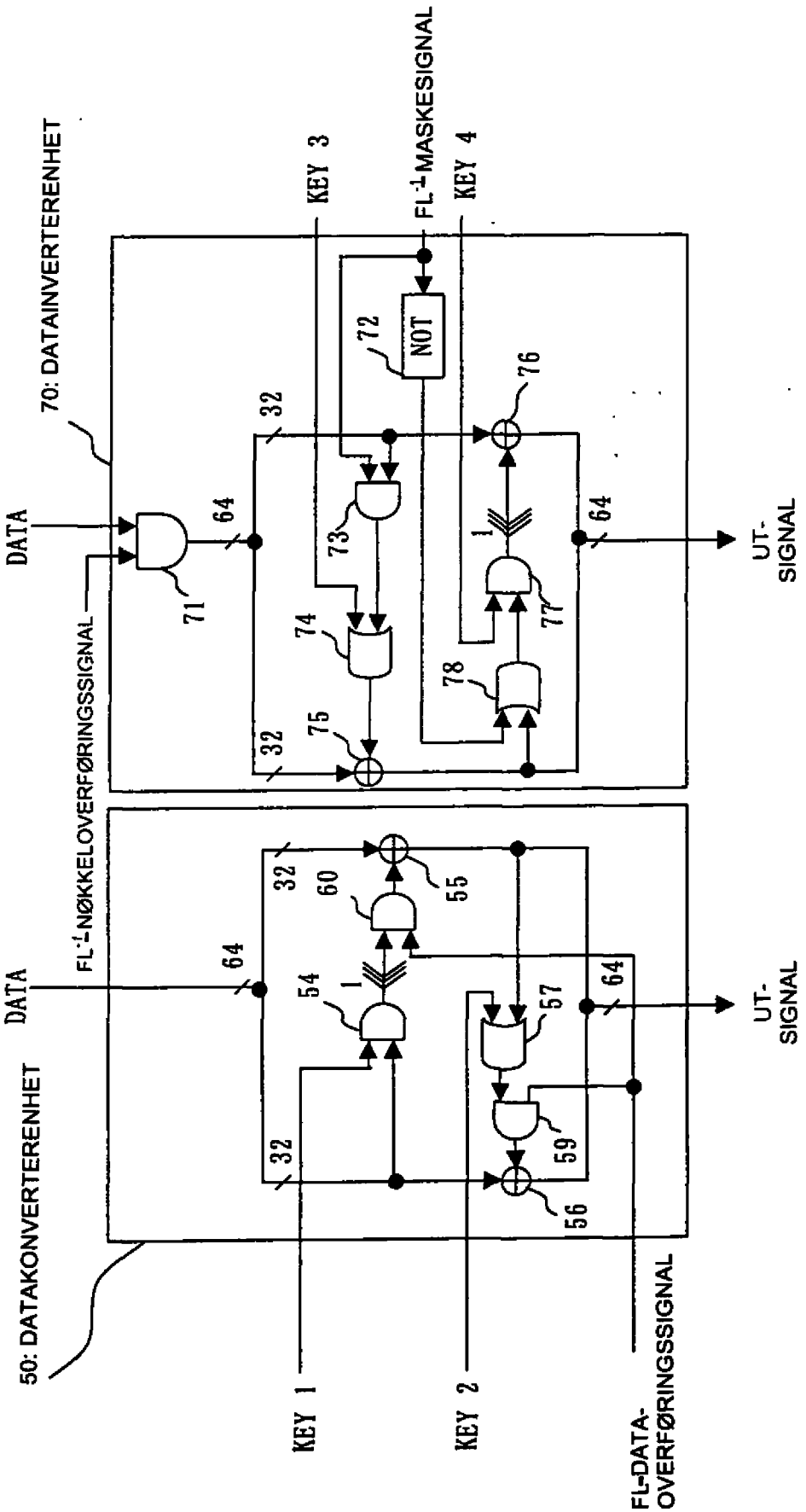

70: DATAINVERTERENHET
KEY 3
FL-1-MASKESIGNAL
KEY 4
DATA
64
71
32
72
NOT
73
74
75
76
77
78
UT-SIGNAL
FL-1-NØKKELOVERFØRINGSSIGNAL
50: DATAKONVERTERENHET
54
55
56
57
59
60
KEY 1
KEY 2
FL-DATA-OVERFØRINGSSIGNAL

Fig. 24

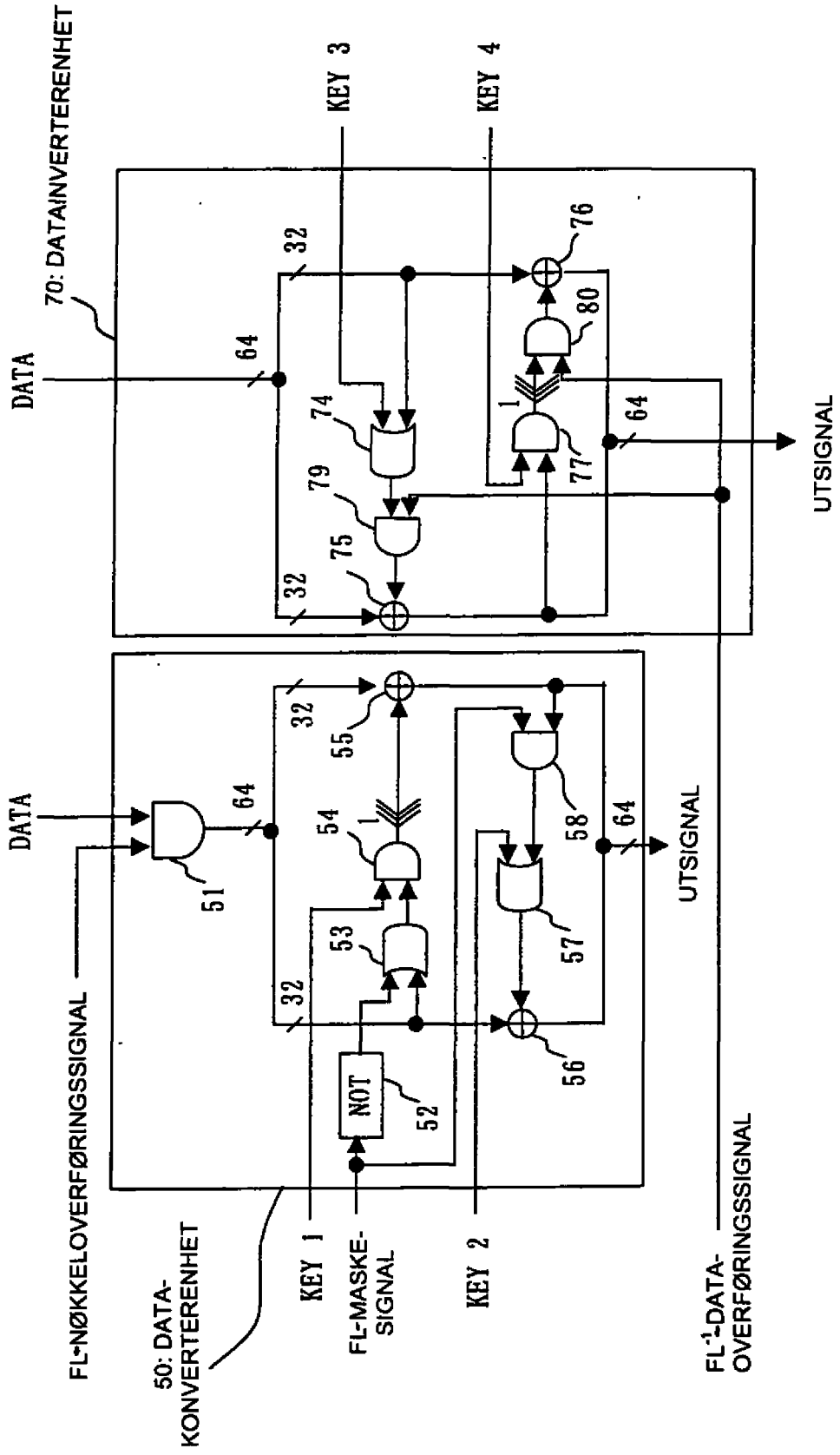

70: DATAINVERTERENHET
DATA
KEY 3
KEY 4
UTSIGNAL
FL-NØKKELOVERFØRINGSSIGNAL
50: DATA-KONVERTERENHET
KEY 1
FL-MASKE-SIGNAL
KEY 2
FL-1-DATA-OVERFØRINGSSIGNAL
NOT

Fig. 25

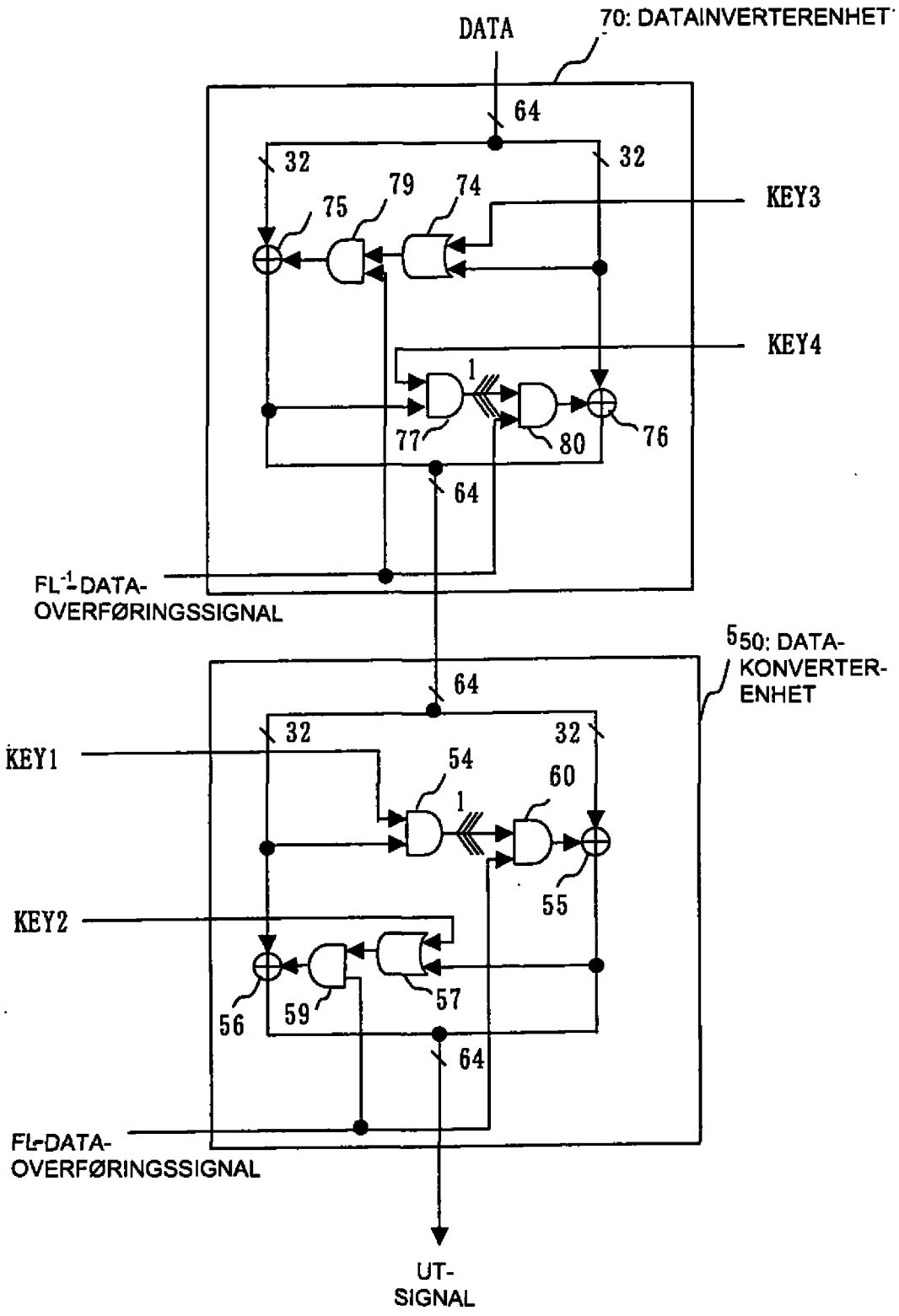

DATA
70: DATAINVERTERENHET
64
32
32
79
74
75
KEY3
KEY4
1
77
80
76
64
FL-1-DATA-OVERFØRINGSSIGNAL
550: DATA-KONVERTER-ENHET
64
32
32
KEY1
54
60
1
55
KEY2
56
59
57
64
FL-DATA-OVERFØRINGSSIGNAL
UT-SIGNAL

Fig. 26

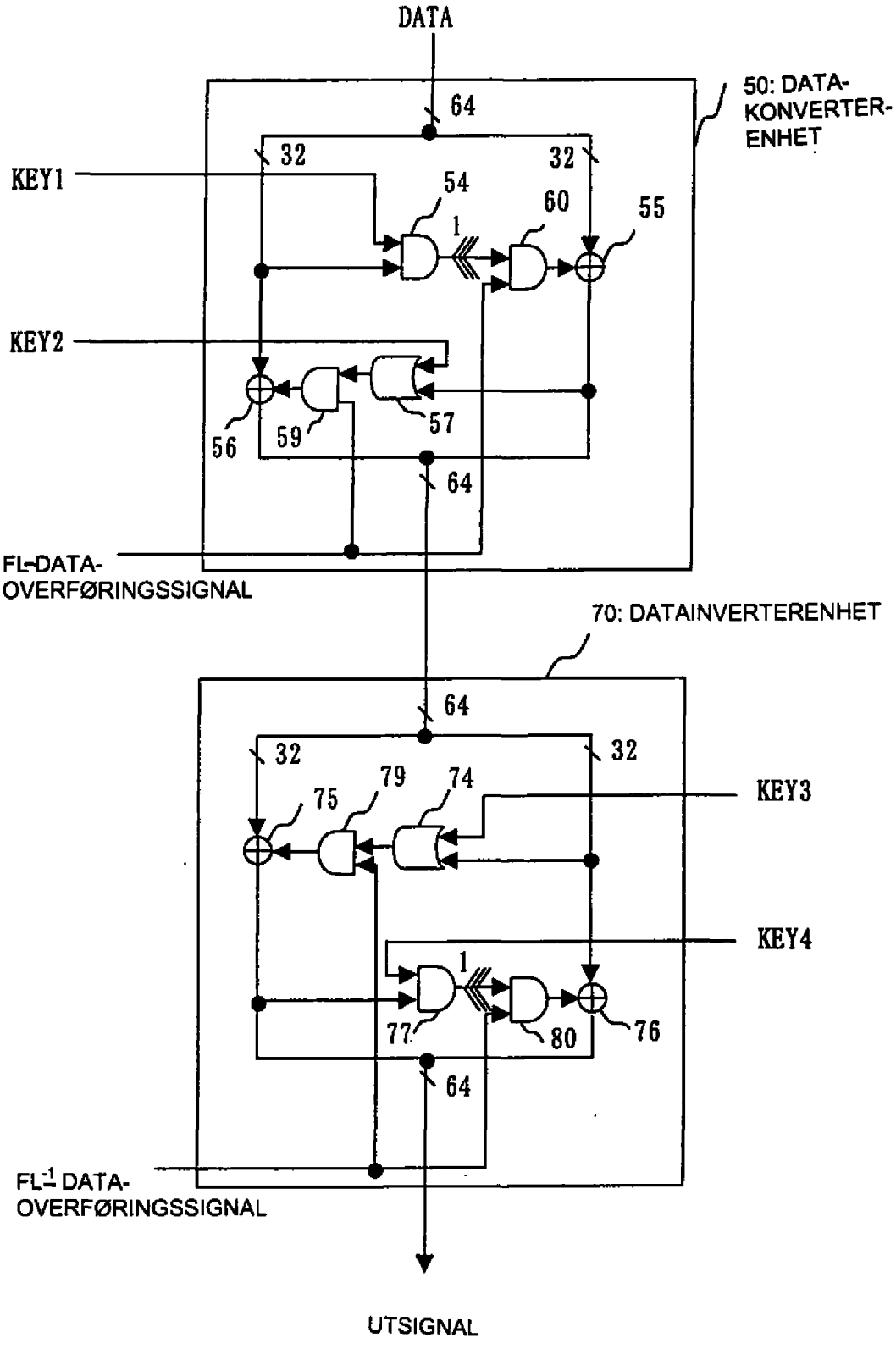

DATA
64
50: DATA-KONVERTER-ENHET
32
32
KEY1
54
60
55
1
KEY2
56
59
57
64
FL-DATA-OVERFØRINGSSIGNAL
70: DATAINVERTERENHET
64
32
32
79
74
75
KEY3
KEY4
1
77
80
76
64
FL-1 DATA-OVERFØRINGSSIGNAL
UTSIGNAL

Fig. 27

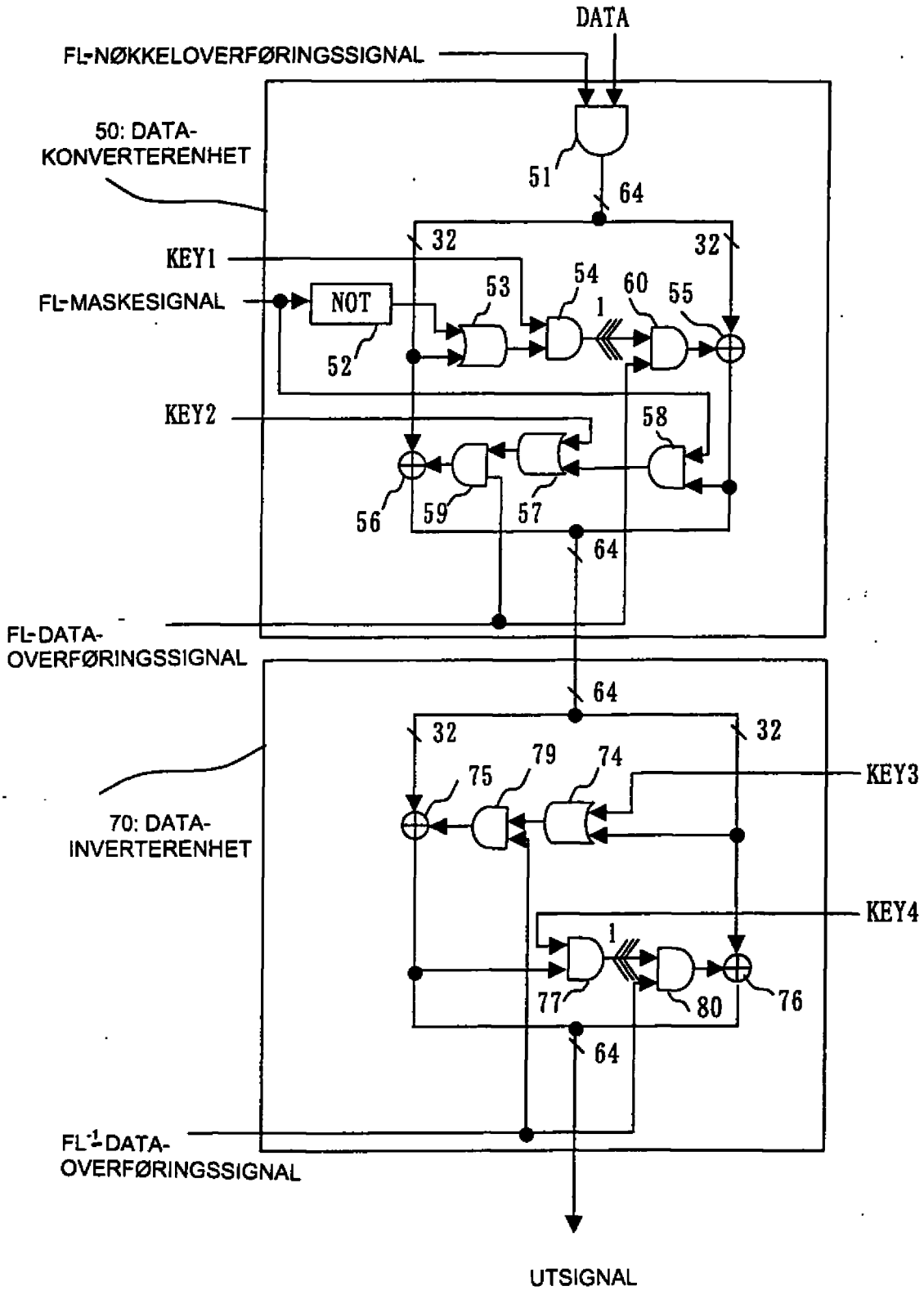

DATA
FL-NØKKELOVERFØRINGSSIGNAL
50: DATA-KONVERTERENHET
51
64
32
32
KEY1
FL-MASKESIGNAL
NOT
53
54
60
55
1
52
KEY2
58
56
59
57
64
FL-DATA-OVERFØRINGSSIGNAL
64
32
32
75
79
74
KEY3
70: DATA-INVERTERENHET
KEY4
1
77
80
76
64
FL-1-DATA-OVERFØRINGSSIGNAL
UTSIGNAL

Fig. 28

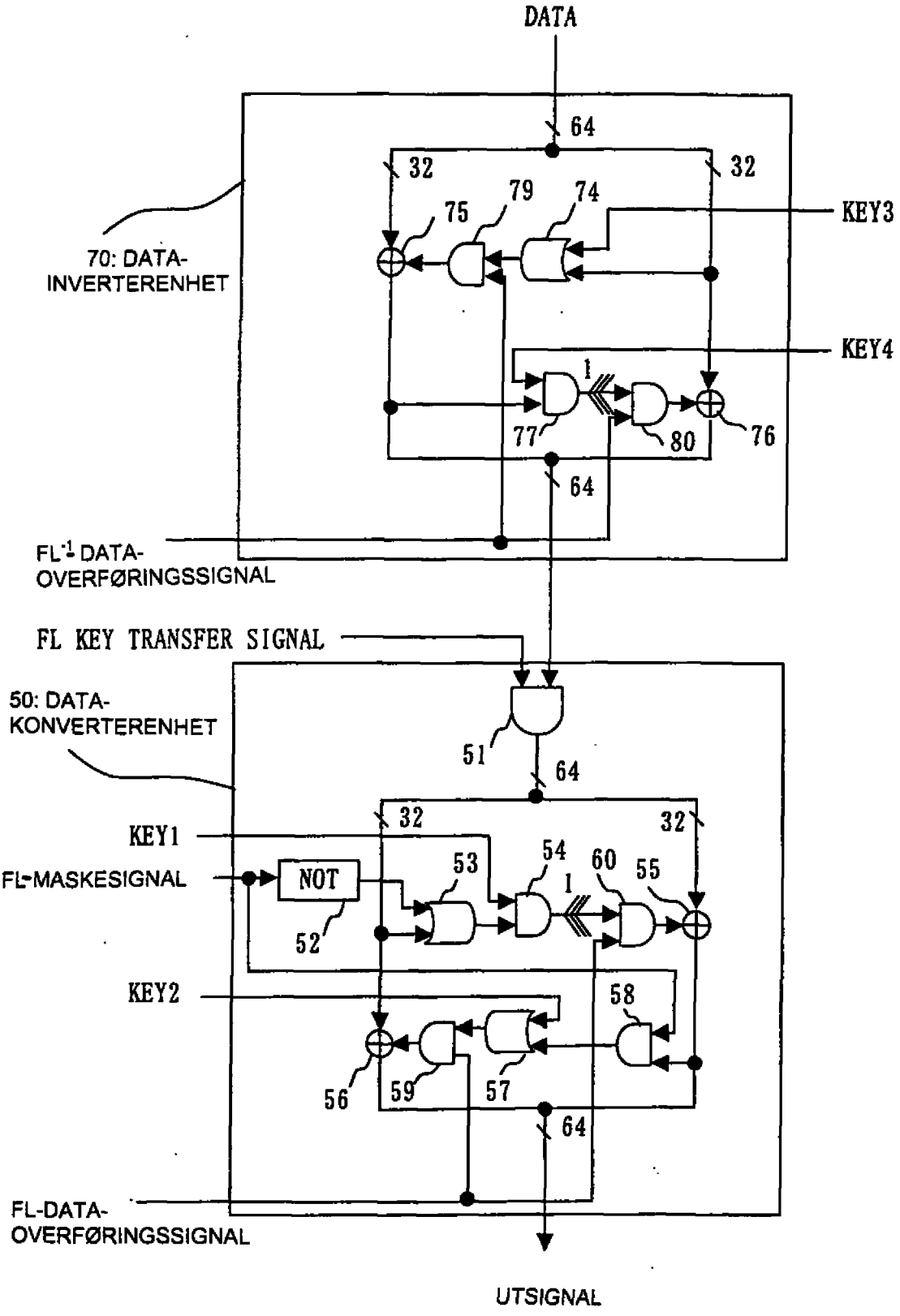

DATA
64
32
32
75
79
74
KEY3
70: DATA-
INVERTERENHET
KEY4
1
77
80
76
64
FL-1 DATA-
OVERFØRINGSSIGNAL
FL KEY TRANSFER SIGNAL
50: DATA-
KONVERTERENHET
51
64
32
32
KEY1
FL-MASKESIGNAL
NOT
53
54
1
60
55
52
KEY2
58
56
59
57
64
FL-DATA-
OVERFØRINGSSIGNAL
UTSIGNAL

Fig. 29

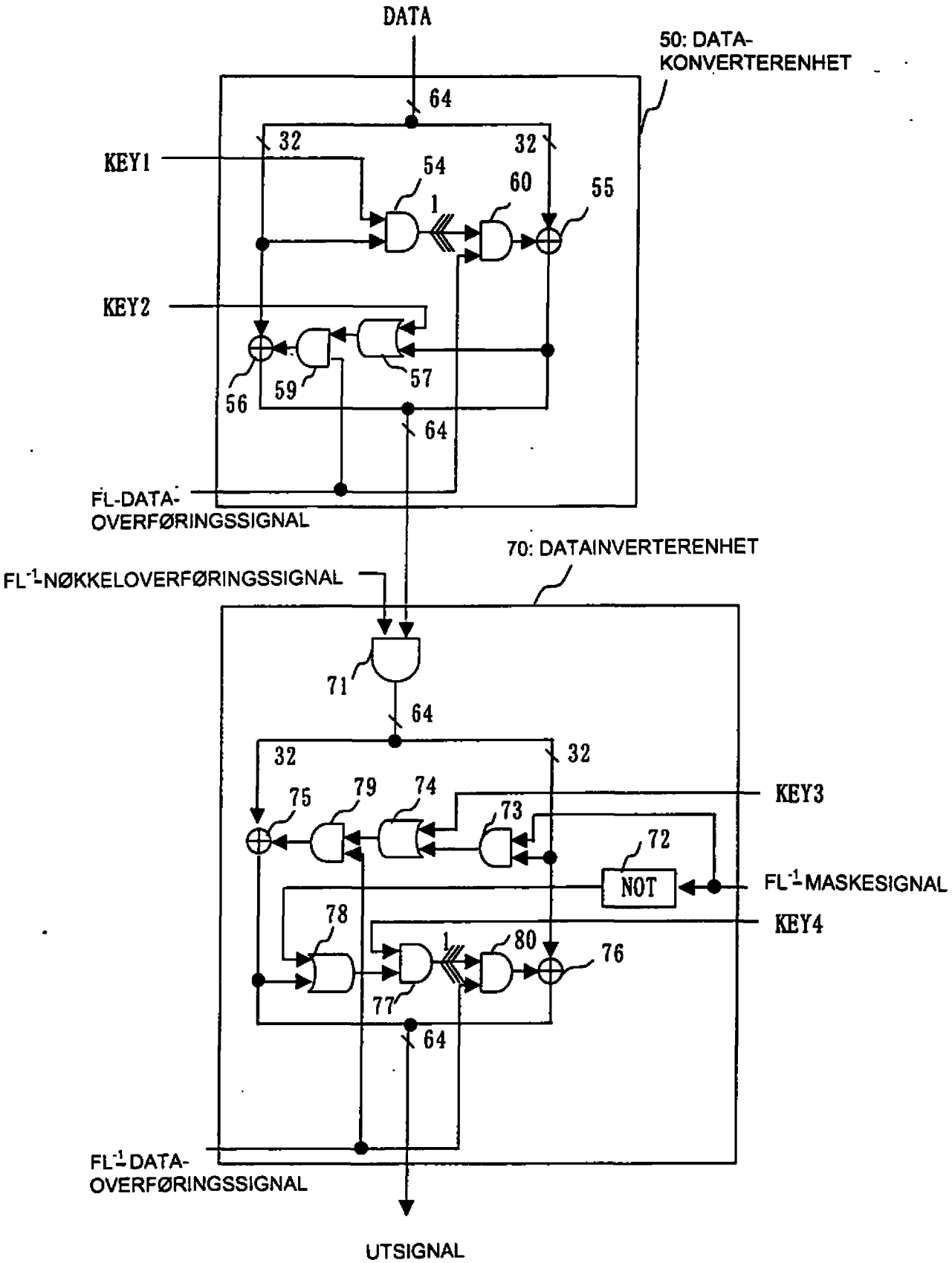

DATA
50: DATA-KONVERTERENHET
64
32
32
KEY1
54
1
60
55
KEY2
56
59
57
64
FL-DATA-OVERFØRINGSSIGNAL
70: DATAINVERTERENHET
FL-1-NØKKELOVERFØRINGSSIGNAL
71
64
32
32
75
79
74
73
KEY3
72
NOT
FL-1-MASKESIGNAL
78
KEY4
1
80
76
77
64
FL-1-DATA-OVERFØRINGSSIGNAL
UTSIGNAL

Fig. 30

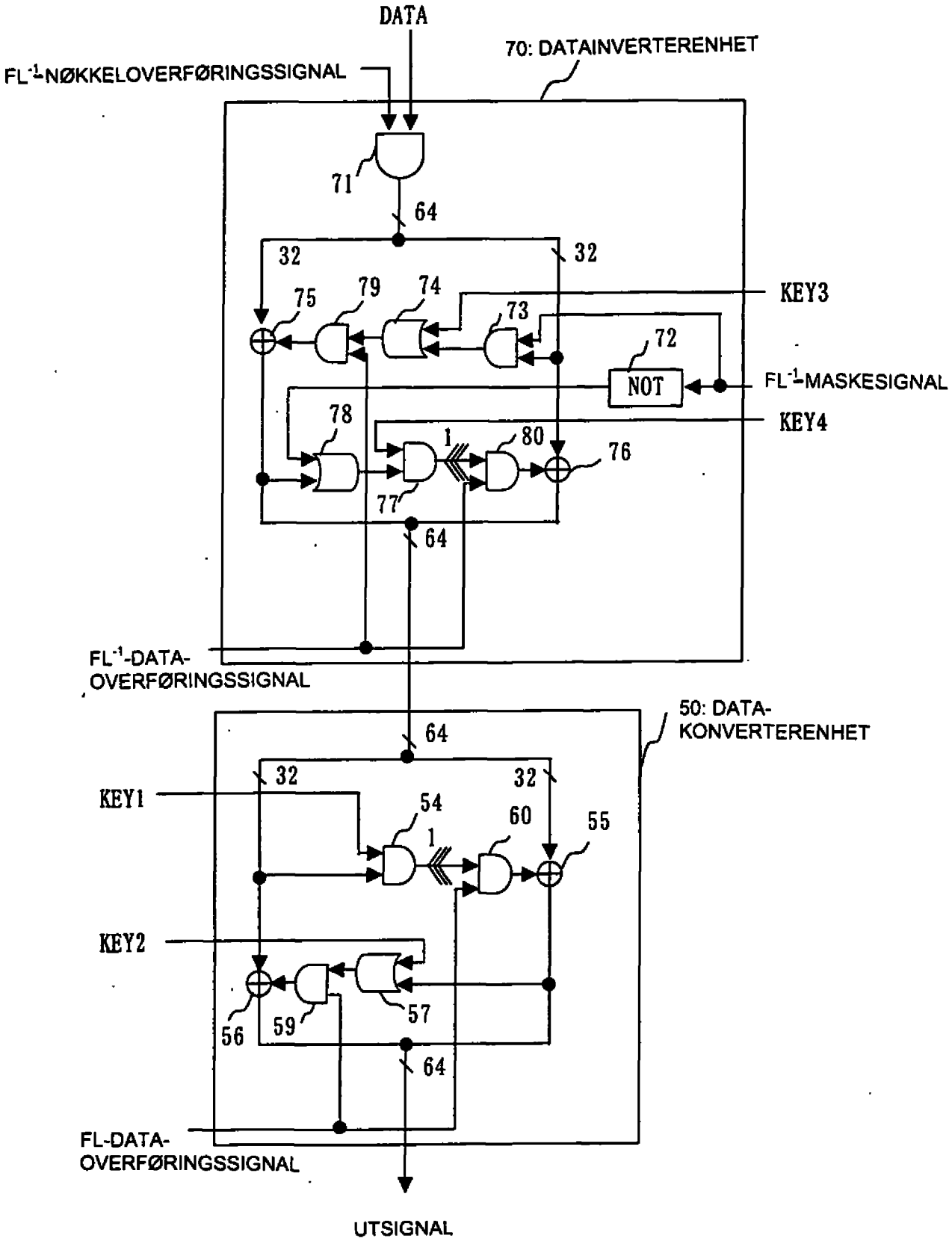

DATA
70: DATAINVERTERENHET
FL-1-NØKKELOVERFØRINGSSIGNAL
71
64
32
32
75
79
74
73
KEY3
72
NOT
FL-1-MASKESIGNAL
78
KEY4
1
80
76
77
64
FL-1-DATA-OVERFØRINGSSIGNAL
50: DATA-KONVERTERENHET
64
32
32
KEY1
54
60
55
1
KEY2
56
59
57
64
FL-DATA-OVERFØRINGSSIGNAL
UTSIGNAL

Fig. 31

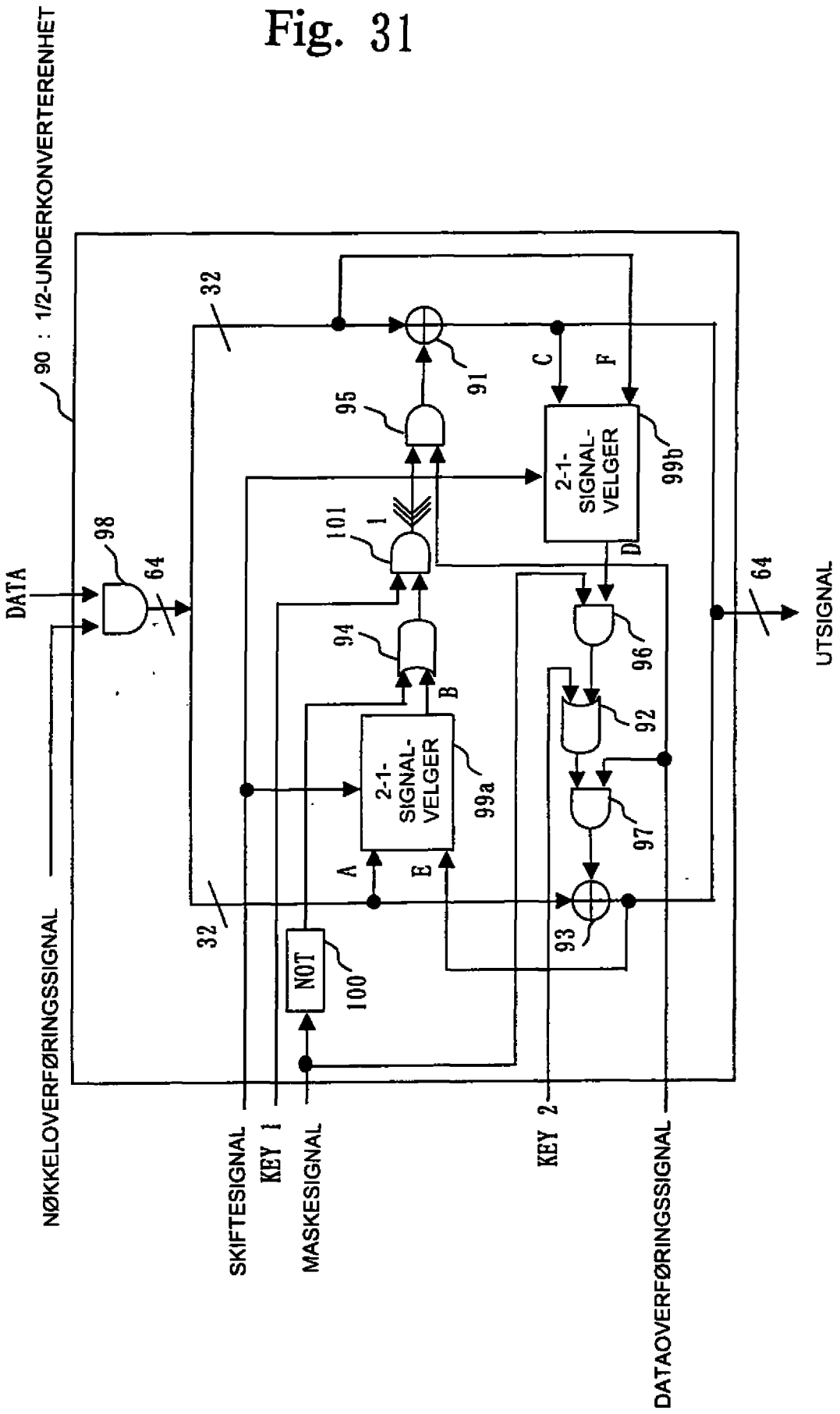

90 : 1/2-UNDERKONVERTERENHET
NØKKELOVERFØRINGSSIGNAL
DATA
98
64
32
91
95
101
94
2-1-SIGNAL-VELGER
99a
2-1-SIGNAL-VELGER
99b
NOT
100
A
B
C
D
E
F
96
92
97
93
64
UTSIGNAL
SKIFTESIGNAL
KEY 1
MASKESIGNAL
KEY 2
DATAOVERFØRINGSSIGNAL

Fig. 32

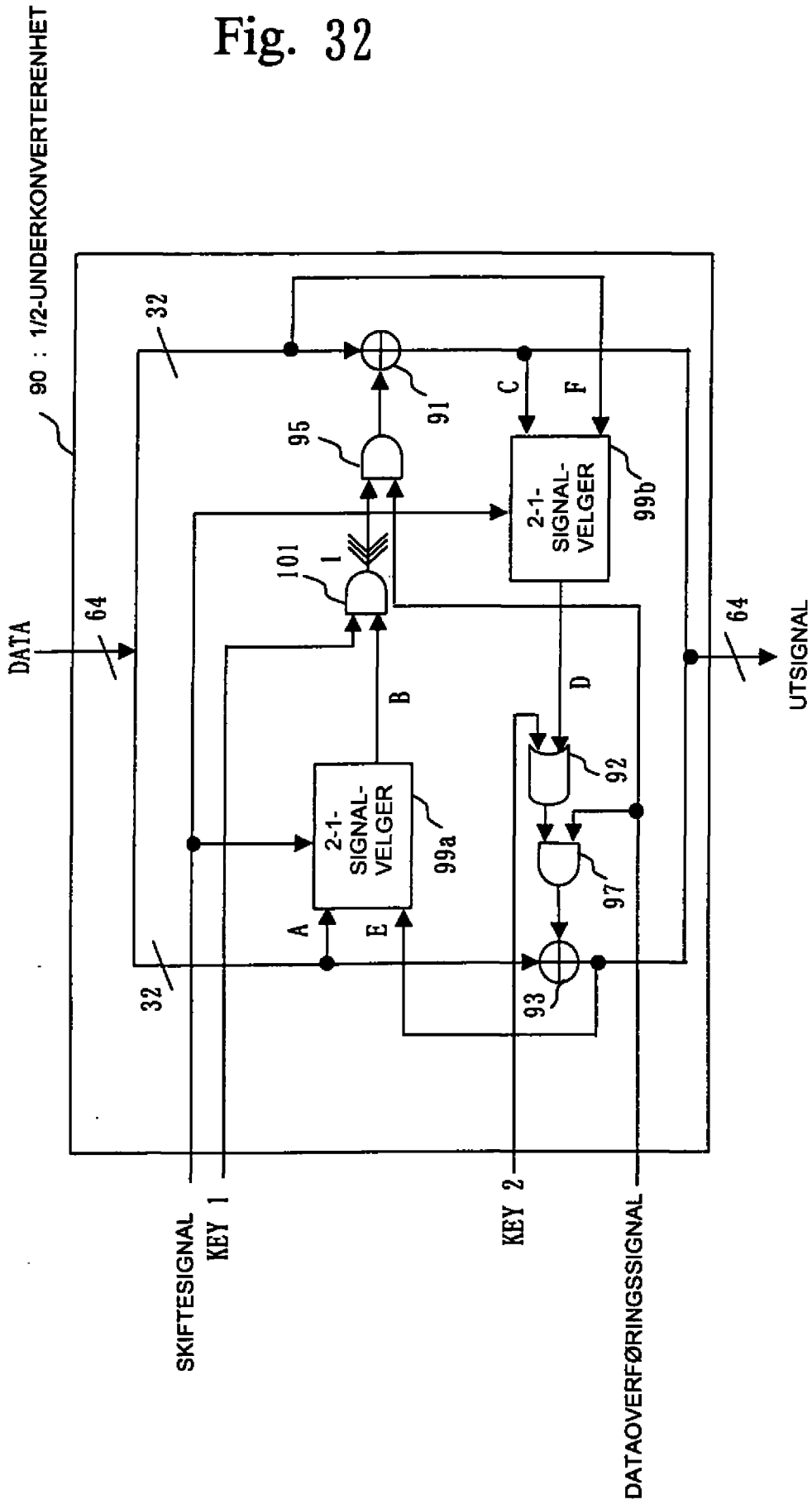

90 : 1/2-UNDERKONVERTERENHET
DATA
64
32
91
95
101
I
2-1-SIGNAL-VELGER
99b
C
F
B
D
92
97
93
2-1-SIGNAL-VELGER
99a
A
E
32
64
UTSIGNAL
SKIFTESIGNAL
KEY 1
KEY 2
DATAOVERFØRINGSSIGNAL

Fig. 33

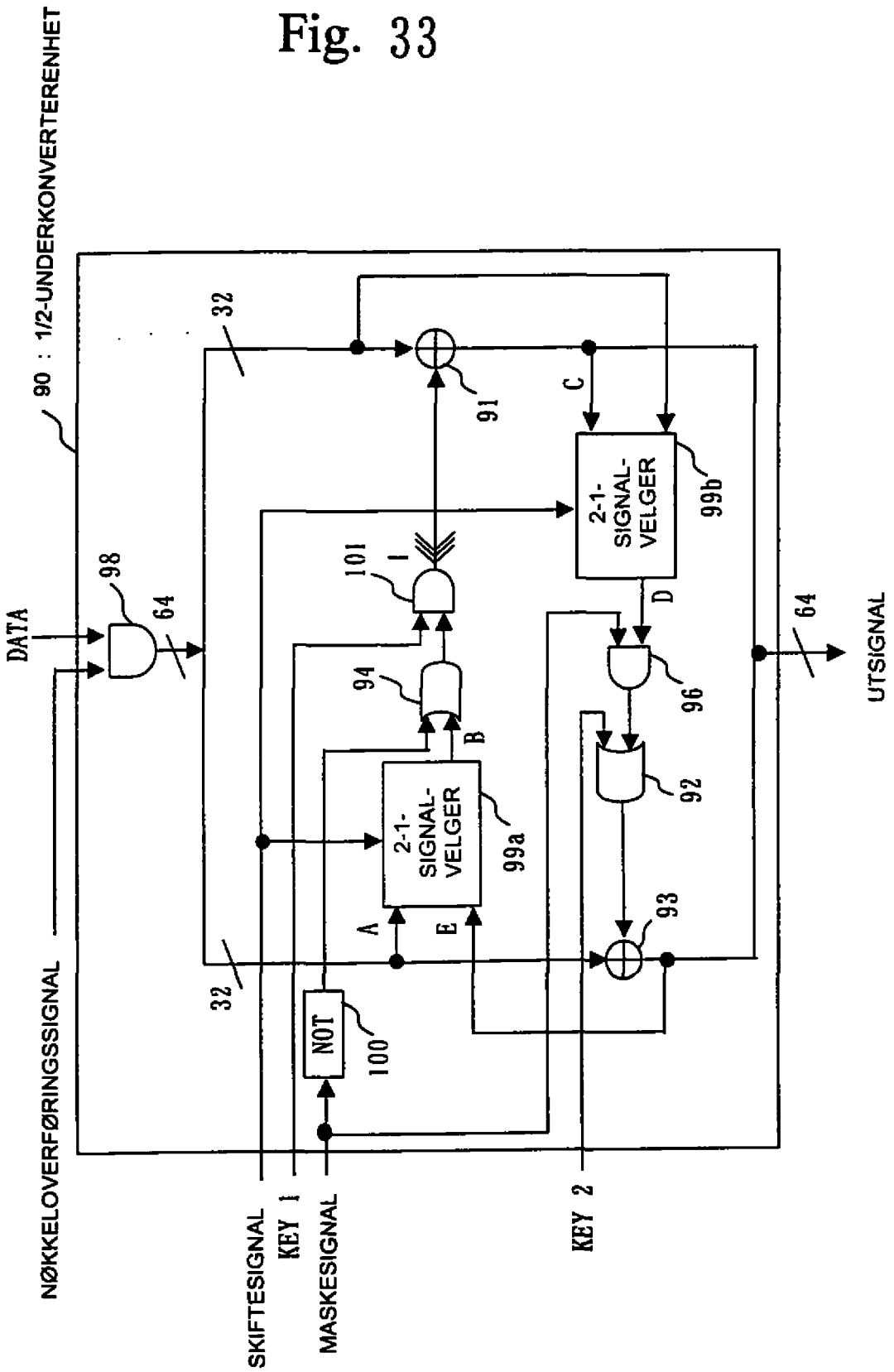

90 : 1/2-UNDERKONVERTERENHET
NØKKELOVERFØRINGSSIGNAL
DATA
98
64
32
91
C
2-1-
SIGNAL-
VELGER
99b
101
I
D
64
UTSIGNAL
94
96
B
92
2-1-
SIGNAL-
VELGER
99a
A
E
93
32
NOT
100
SKIFTESIGNAL
KEY 1
MASKESIGNAL
KEY 2

Fig. 34

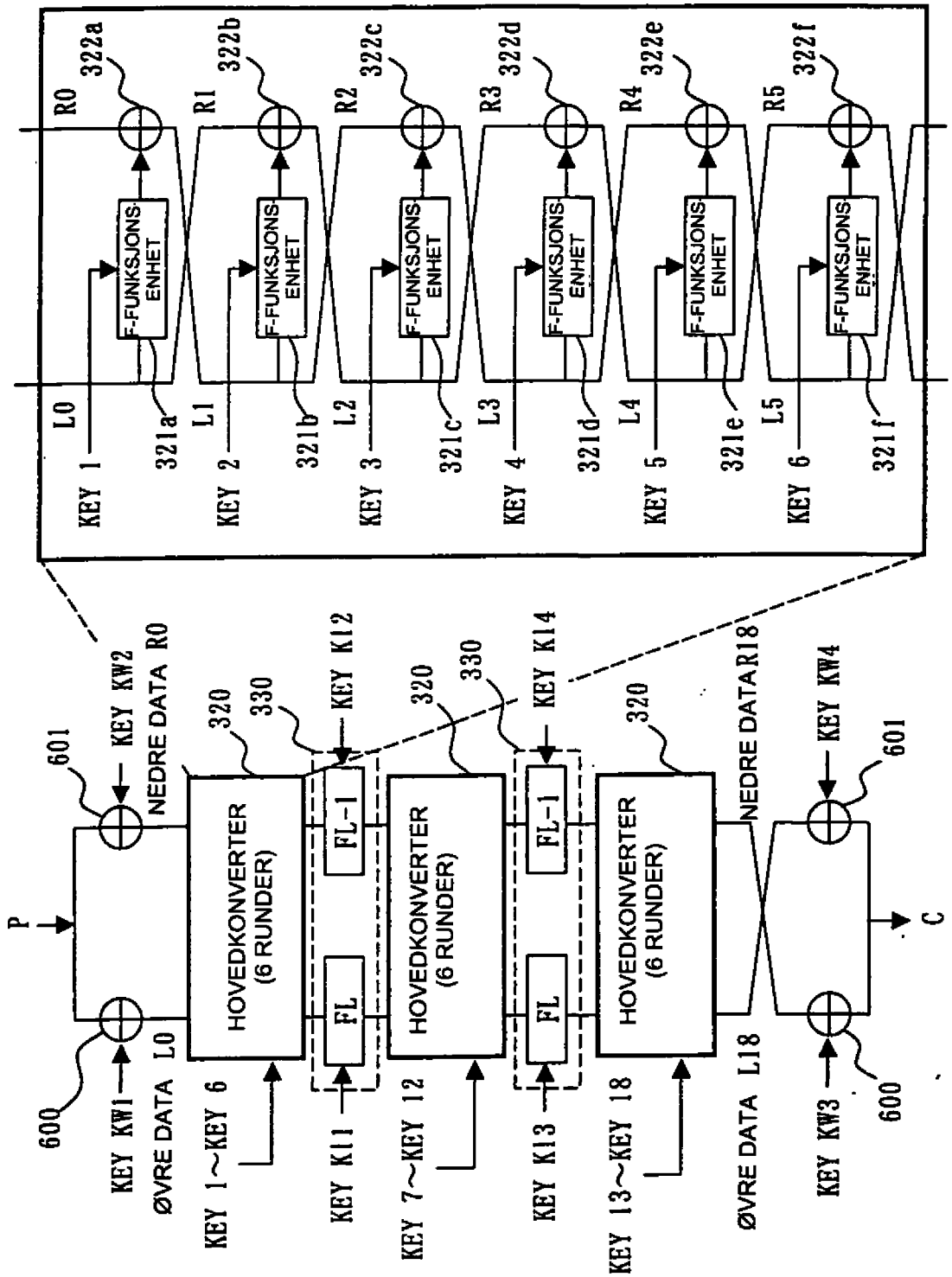

R0
322a
R1
322b
R2
322c
R3
322d
R4
322e
R5
322f
F-FUNKSJONS ENHET
L0
KEY 1
321a
L1
KEY 2
321b
L2
KEY 3
321c
L3
KEY 4
321d
L4
KEY 5
321e
L5
KEY 6
321f
P
601
KEY KW2
NEDRE DATA R0
320
330
KEY K12
KEY K14
HOVEDKONVERTER (6 RUNDER)
FL-1
FL
NEDRE DATA R18
KEY KW4
C
600
KEY KW1
ØVRE DATA L0
KEY 1~KEY 6
KEY K11
KEY 7~KEY 12
KEY K13
KEY 13~KEY 18
ØVRE DATA L18
KEY KW3

Fig. 35

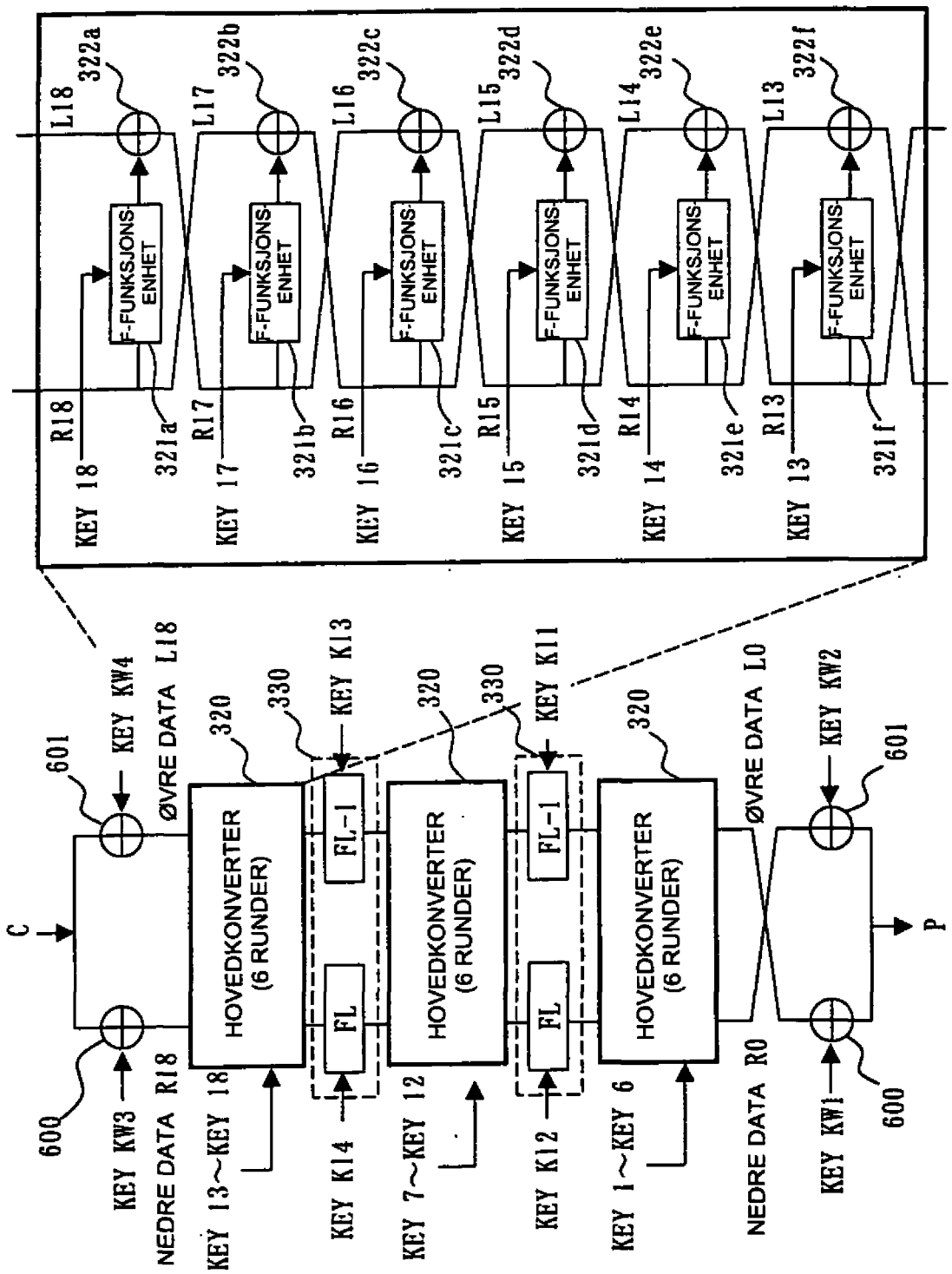

L18
322a
L17
322b
L16
322c
L15
322d
L14
322e
L13
322f
F-FUNKSJONS ENHET
R18
R17
R16
R15
R14
R13
KEY 18
KEY 17
KEY 16
KEY 15
KEY 14
KEY 13
321a
321b
321c
321d
321e
321f
C
601
KEY KW4
ØVRE DATA L18
320
330
KEY K13
KEY K11
HOVEDKONVERTER (6 RUNDER)
FL-1
FL
600
KEY KW3
NEDRE DATA R18
KEY 13~KEY 18
KEY K14
KEY 7~KEY 12
KEY K12
KEY 1~KEY 6
ØVRE DATA L0
KEY KW2
NEDRE DATA R0
KEY KW1
P

Fig. 36

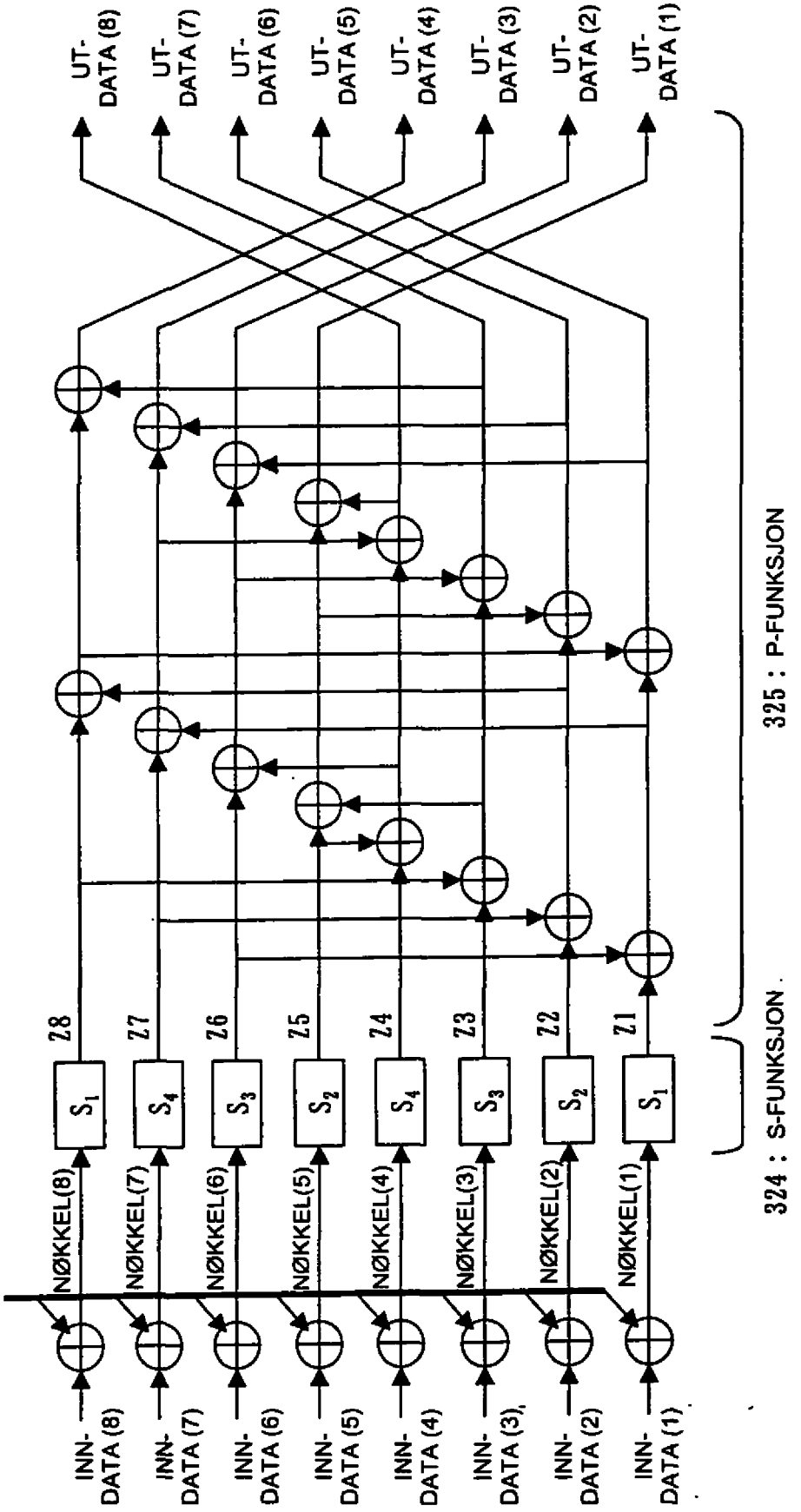

INN-DATA (8)
INN-DATA (7)
INN-DATA (6)
INN-DATA (5)
INN-DATA (4)
INN-DATA (3)
INN-DATA (2)
INN-DATA (1)
NØKKEL(8)
NØKKEL(7)
NØKKEL(6)
NØKKEL(5)
NØKKEL(4)
NØKKEL(3)
NØKKEL(2)
NØKKEL(1)
S1
S4
S3
S2
S4
S3
S2
S1
Z8
Z7
Z6
Z5
Z4
Z3
Z2
Z1
UT-DATA (8)
UT-DATA (7)
UT-DATA (6)
UT-DATA (5)
UT-DATA (4)
UT-DATA (3)
UT-DATA (2)
UT-DATA (1)
324 : S-FUNKSJON
325 : P-FUNKSJON

Fig. 37

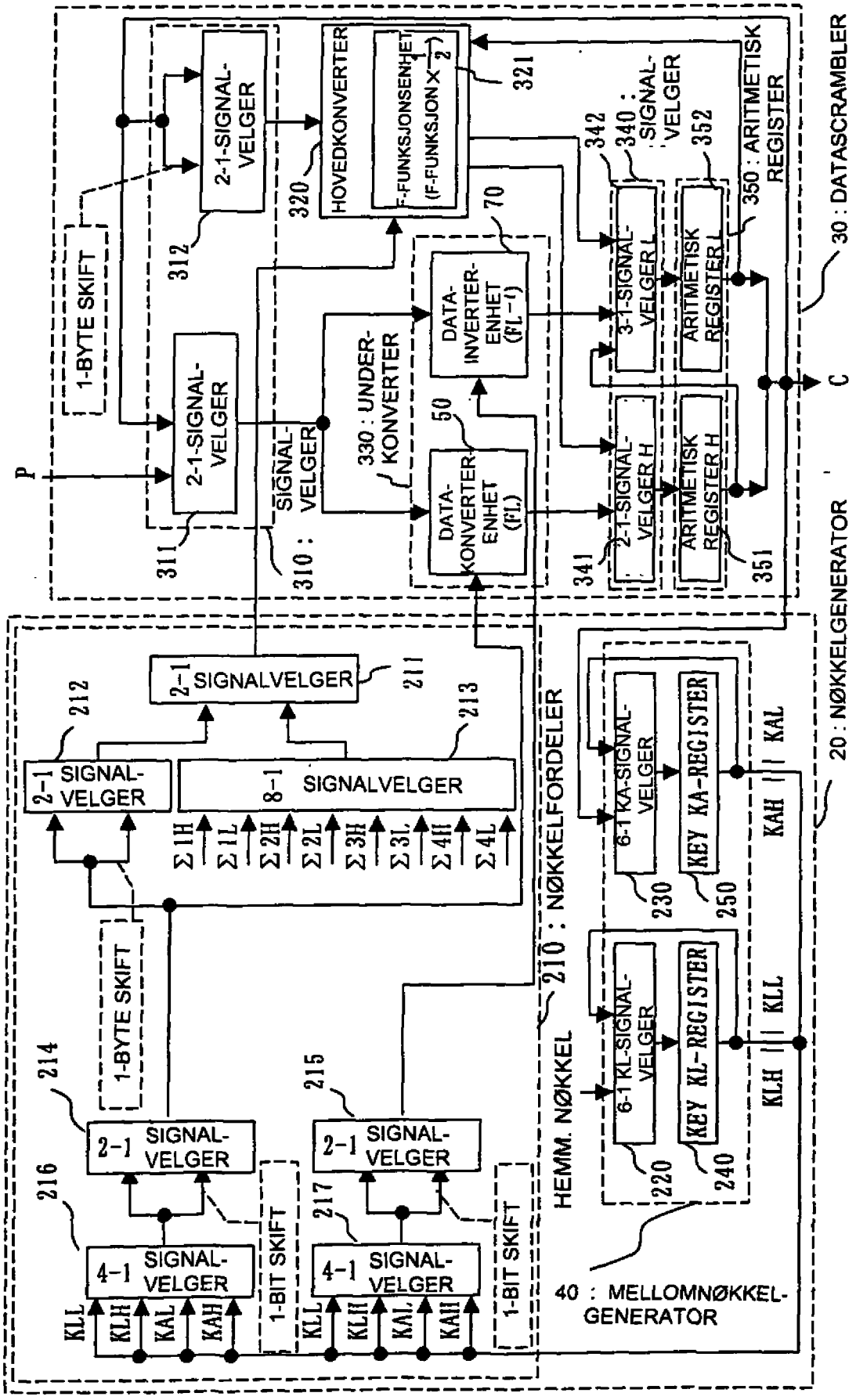

30 : DATASCRAMBLER
20 : NØKKELGENERATOR
210 : NØKKELFORDELER
HEMM. NØKKEL
40 : MELLOMNØKKEL-GENERATOR
1-BYTE SKIFT
2-1-SIGNAL-VELGER
HOVEDKONVERTER
F-FUNKSJONSENHET
(F-FUNKSJON $\times\frac{1}{2}$)
320
321
312
311
310 : SIGNAL-VELGER
330 : UNDER-KONVERTER
DATA-INVERTER-ENHET (FL^{-1})
DATA-KONVERTER-ENHET (FL)
70
50
340 : SIGNAL-VELGER
342
341
3-1-SIGNAL-VELGER L
2-1-SIGNAL-VELGER H
ARITMETISK REGISTER L
ARITMETISK REGISTER H
350 : ARITMETISK REGISTER
352
351
P
C
2-1 SIGNALVELGER
211
212
213
2-1 SIGNAL-VELGER
8-1 SIGNALVELGER
Σ1H Σ1L Σ2H Σ2L Σ3H Σ3L Σ4H Σ4L
214
215
216
217
4-1 SIGNAL-VELGER
1-BIT SKIFT
KLL KLH KAL KAH
6-1 KA-SIGNAL-VELGER
6-1 KL-SIGNAL-VELGER
KEY KA-REGISTER
KEY KL-REGISTER
230
250
1220
240
KAL
KAH
KLL
KLH

Fig. 38

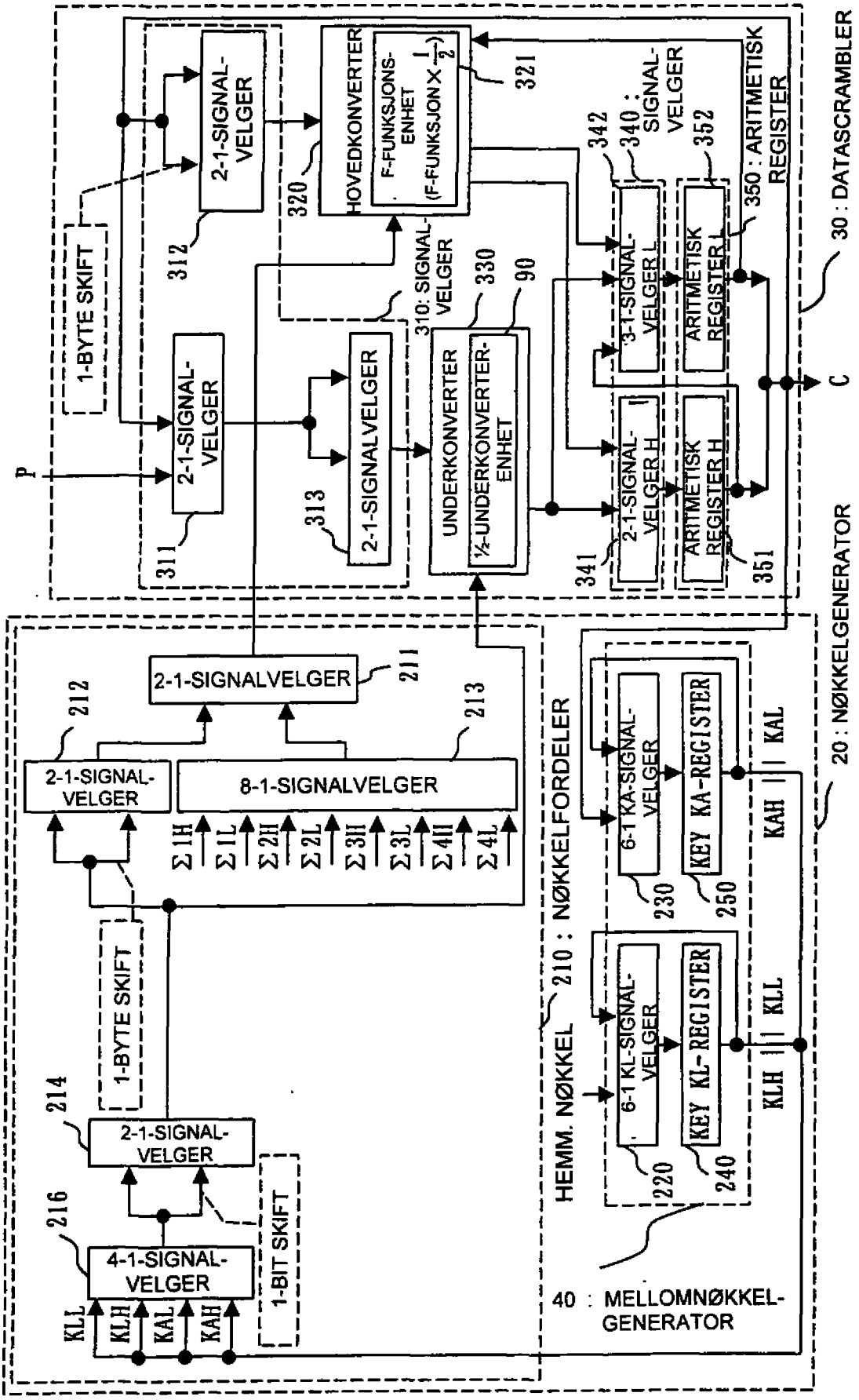

30 : DATASCRAMBLER
20 : NØKKELGENERATOR
210 : NØKKELFORDELER
HEMM. NØKKEL
40 : MELLOMNØKKEL-GENERATOR
1-BYTE SKIFT
2-1-SIGNAL-VELGER
HOVEDKONVERTER
F-FUNKSJONS-ENHET
(F-FUNKSJON X 1/2)
310: SIGNAL-VELGER
UNDERKONVERTER
½-UNDERKONVERTER-ENHET
2-1-SIGNALVELGER
3-1-SIGNAL-VELGER L
2-1-SIGNAL-VELGER H
340 : SIGNAL-VELGER
350 : ARITMETISK REGISTER
ARITMETISK REGISTER L
ARITMETISK REGISTER H
8-1-SIGNALVELGER
4-1-SIGNAL-VELGER
1-BIT SKIFT
6-1 KA-SIGNAL-VELGER
KEY KA-REGISTER
6-1 KL-SIGNAL-VELGER
KEY KL-REGISTER
KLL
KLH
KAL
KAH
P
C
311
312
313
320
321
330
90
341
342
351
352
211
212
213
214
216
220
230
240
250

Fig. 39

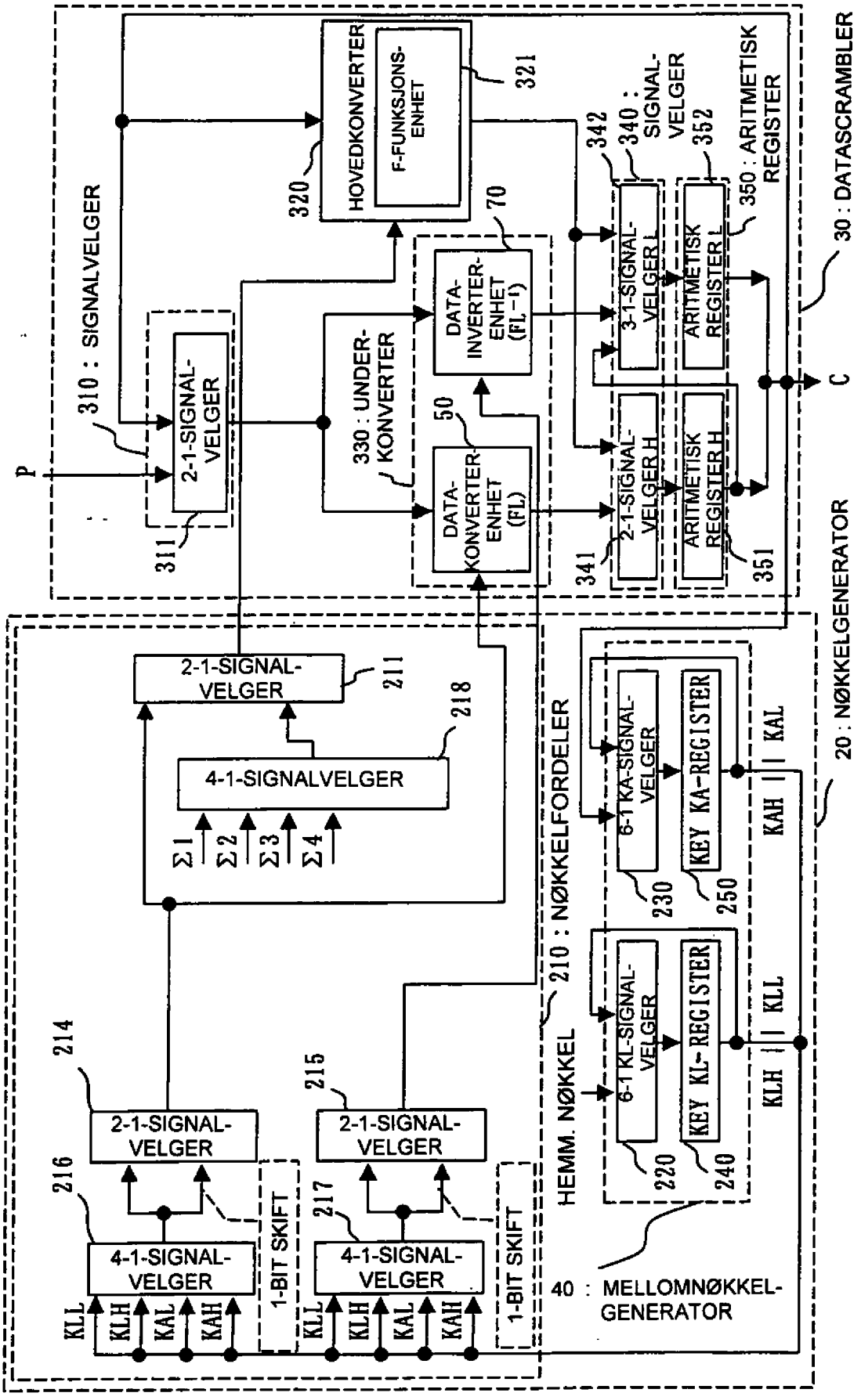

30 : DATASCRAMBLER
20 : NØKKELGENERATOR
310 : SIGNALVELGER
2-1-SIGNAL-VELGER
311
P
HOVEDKONVERTER
F-FUNKSJONS-ENHET
320
321
330 : UNDER-KONVERTER
DATA-INVERTER-ENHET (FL⁻¹)
70
DATA-KONVERTER-ENHET (FL)
50
340 : SIGNAL-VELGER
342
3-1-SIGNAL-VELGER L
2-1-SIGNAL-VELGER H
341
350 : ARITMETISK REGISTER
352
ARITMETISK REGISTER L
ARITMETISK REGISTER H
351
C
2-1-SIGNAL-VELGER
211
4-1-SIGNALVELGER
218
Σ1 Σ2 Σ3 Σ4
210 : NØKKELFORDELER
HEMM. NØKKEL
6-1 KA-SIGNAL-VELGER
230
KEY KA-REGISTER
250
KAL
KAH
6-1 KL-SIGNAL-VELGER
220
KEY KL-REGISTER
240
KLL
KLH
214
215
2-1-SIGNAL-VELGER
2-1-SIGNAL-VELGER
216
217
4-1-SIGNAL-VELGER
4-1-SIGNAL-VELGER
1-BIT SKIFT
1-BIT SKIFT
KLL KLH KAL KAH
KLL KLH KAL KAH
40 : MELLOMNØKKEL-GENERATOR

Fig. 40

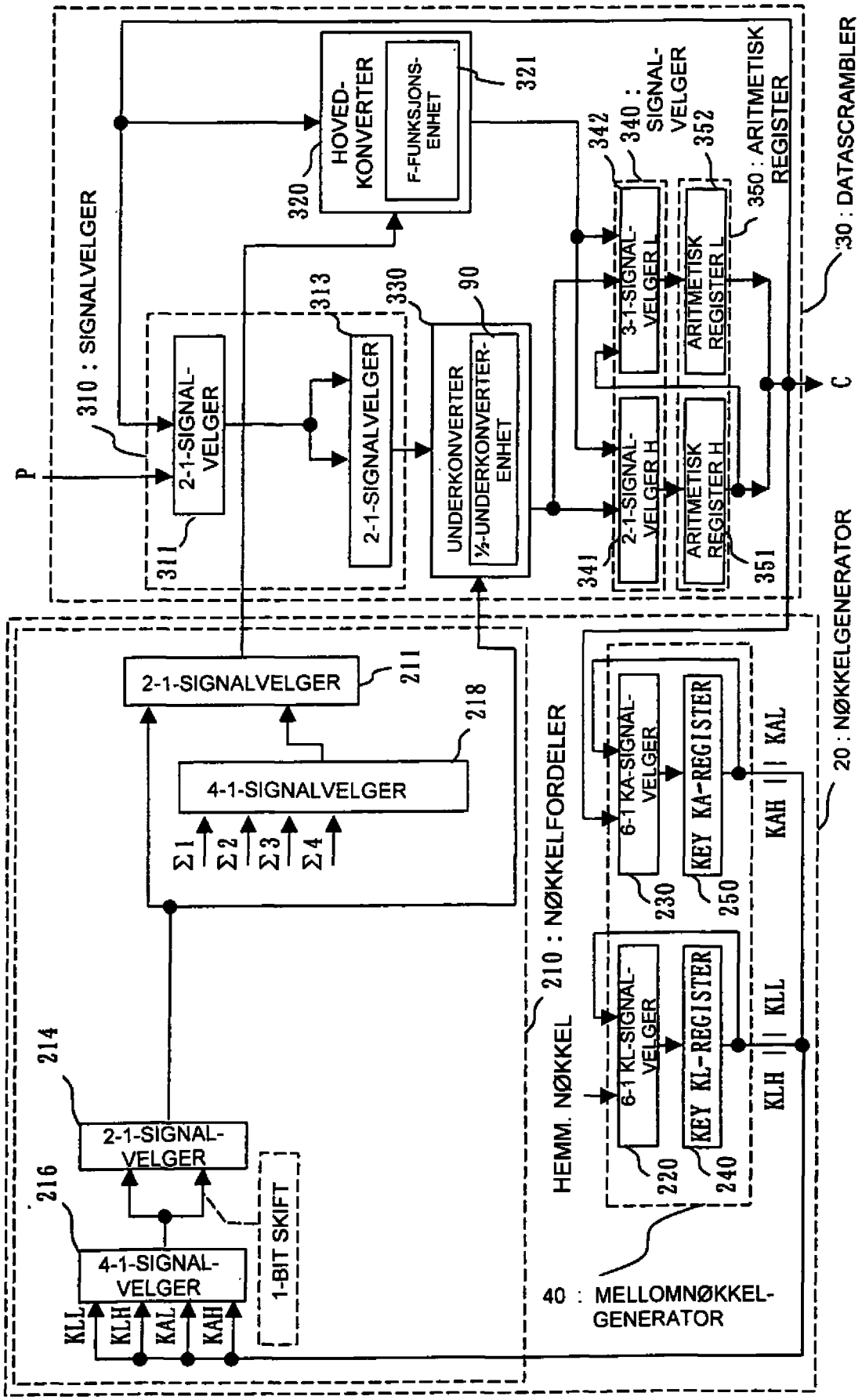

310 : SIGNALVELGER
2-1-SIGNAL-VELGER
311
313
2-1-SIGNALVELGER
330
90
UNDERKONVERTER
½-UNDERKONVERTER-ENHET
HOVED-KONVERTER
320
F-FUNKSJONS-ENHET
321
342
340 : SIGNAL-VELGER
352
350 : ARITMETISK REGISTER
3-1-SIGNAL-VELGER L
ARITMETISK REGISTER L
30 : DATASCRAMBLER
2-1-SIGNAL-VELGER H
ARITMETISK REGISTER H
341
351
P
C
2-1-SIGNALVELGER
211
218
4-1-SIGNALVELGER
Σ1
Σ2
Σ3
Σ4
214
2-1-SIGNAL-VELGER
216
4-1-SIGNAL-VELGER
1-BIT SKIFT
KLL
KLH
KAL
KAH
210 : NØKKELFORDELER
HEMM. NØKKEL
6-1 KA-SIGNAL-VELGER
KEY KA-REGISTER
230
250
6-1 KL-SIGNAL-VELGER
KEY KL-REGISTER
220
240
KAL
KAH
KLL
KLH
20 : NØKKELGENERATOR
40 : MELLOMNØKKEL-GENERATOR

Fig. 41

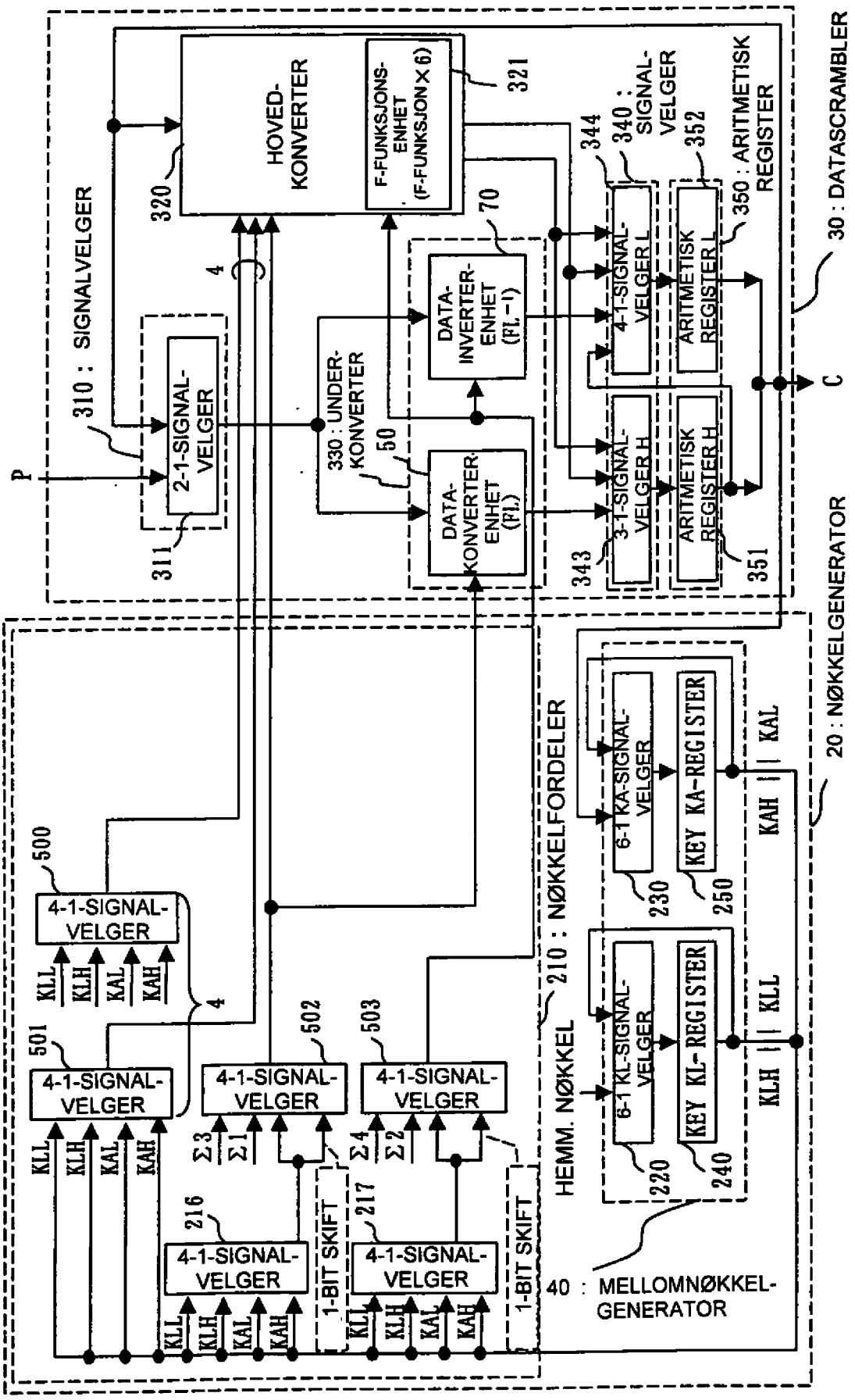

30 : DATASCRAMBLER
20 : NØKKELGENERATOR
310 : SIGNALVELGER
2-1-SIGNAL-VELGER
311
320
HOVED-KONVERTER
F-FUNKSJONS-ENHET (F-FUNKSJON X 6)
321
330 : UNDER-KONVERTER
DATA-INVERTER-ENHET (FL⁻¹)
70
DATA-KONVERTER-ENHET (FL)
50
340 : SIGNAL-VELGER
344
4-1-SIGNAL-VELGER L
3-1-SIGNAL-VELGER H
343
350 : ARITMETISK REGISTER
352
ARITMETISK REGISTER L
ARITMETISK REGISTER H
351
P
C
4
210 : NØKKELFORDELER
HEMM. NØKKEL
6-1 KA-SIGNAL-VELGER
230
KEY KA-REGISTER
250
6-1 KL-SIGNAL-VELGER
220
KEY KL-REGISTER
240
KAL
KAH
KLL
KLH
40 : MELLOMNØKKEL-GENERATOR
500
501
502
503
4-1-SIGNAL-VELGER
KLL
KLH
KAL
KAH
Σ3
Σ1
Σ4
Σ2
1-BIT SKIFT
216
217

Fig. 42

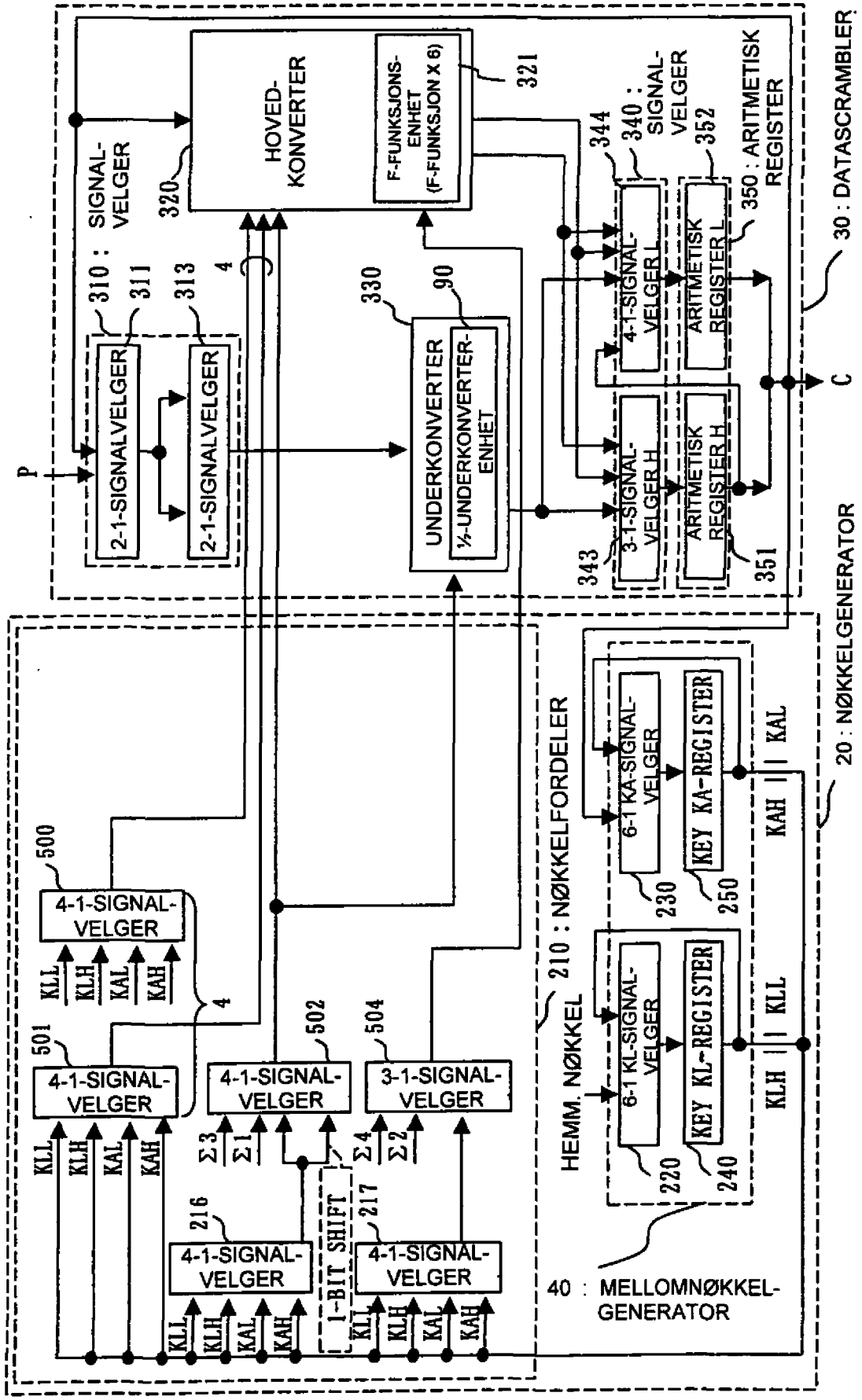

30 : DATASCRAMBLER
20 : NØKKELGENERATOR
210 : NØKKELFORDELER
HEMM. NØKKEL
40 : MELLOMNØKKEL-GENERATOR
310 : SIGNAL-VELGER
320
HOVED-KONVERTER
F-FUNKSJONS-ENHET (F-FUNKSJON X 6)
321
340 : SIGNAL-VELGER
344
352
350 : ARITMETISK REGISTER
330
90
UNDERKONVERTER
½-UNDERKONVERTER-ENHET
2-1-SIGNALVELGER
311
313
4
P
C
4-1-SIGNAL-VELGER L
3-1-SIGNAL-VELGER H
ARITMETISK REGISTER L
ARITMETISK REGISTER H
343
351
500
501
502
504
216
217
1-BIT SHIFT
4-1-SIGNAL-VELGER
3-1-SIGNAL-VELGER
KLL
KLH
KAL
KAH
Σ3
Σ1
Σ4
Σ2
6-1 KA-SIGNAL-VELGER
6-1 KL-SIGNAL-VELGER
KEY KA-REGISTER
KEY KL-REGISTER
230
250
220
240

Fig. 43

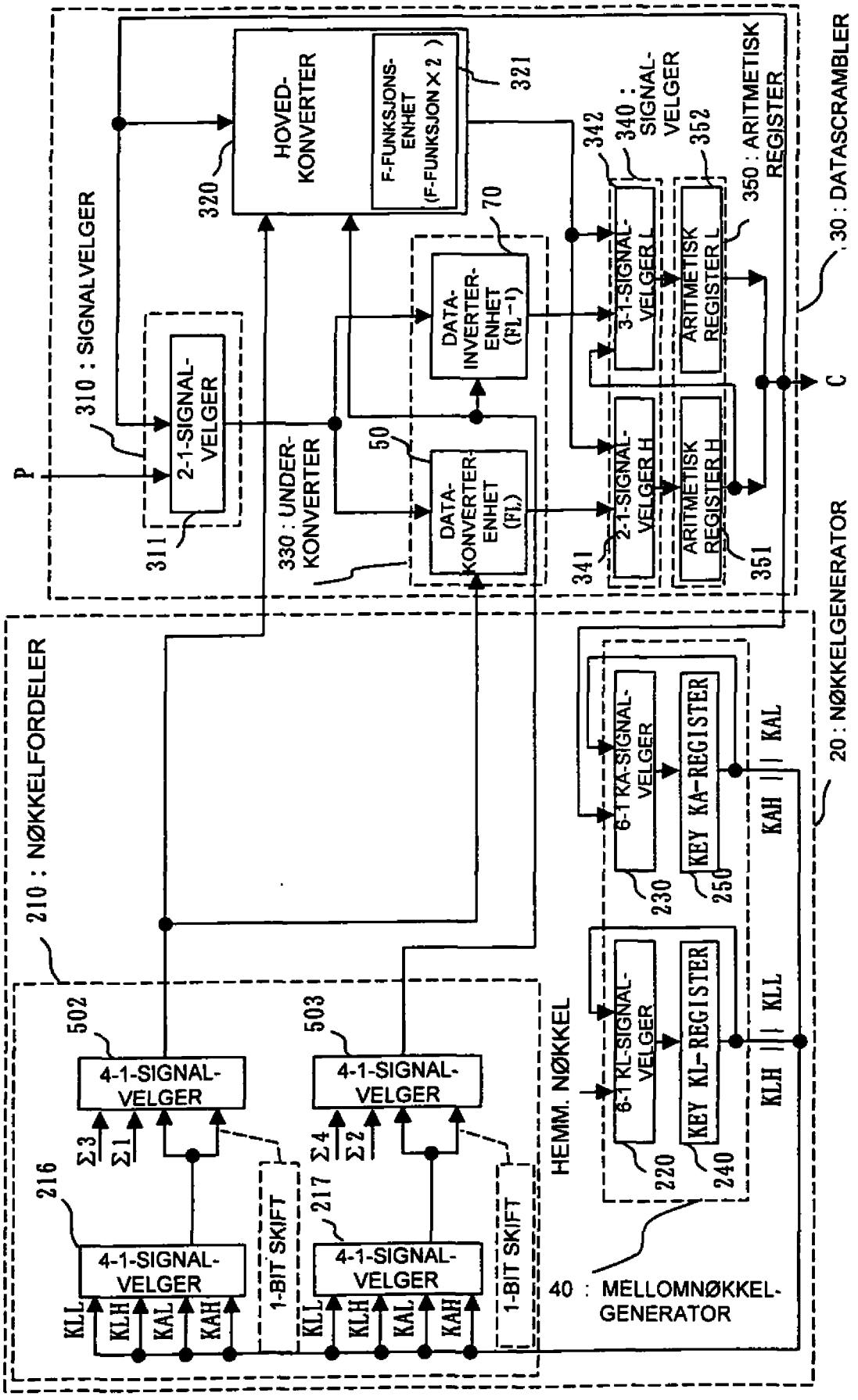

30 : DATASCRAMBLER
20 : NØKKELGENERATOR
210 : NØKKELFORDELER
310 : SIGNALVELGER
330 : UNDER-KONVERTER
HOVED-KONVERTER
F-FUNKSJONS-ENHET (F-FUNKSJON × 2)
2-1-SIGNAL-VELGER
DATA-KONVERTER-ENHET (FL)
DATA-INVERTER-ENHET (FL⁻¹)
340 : SIGNAL-VELGER
3-1-SIGNAL-VELGER L
2-1-SIGNAL-VELGER H
350 : ARITMETISK REGISTER
ARITMETISK REGISTER L
ARITMETISK REGISTER H
6-1 KA-SIGNAL-VELGER
KEY KA-REGISTER
6-1 KL-SIGNAL-VELGER
KEY KL-REGISTER
HEMM. NØKKEL
4-1-SIGNAL-VELGER
1-BIT SKIFT
40 : MELLOMNØKKEL-GENERATOR
P
C
KAL
KAH
KLH
KLL

Fig. 44

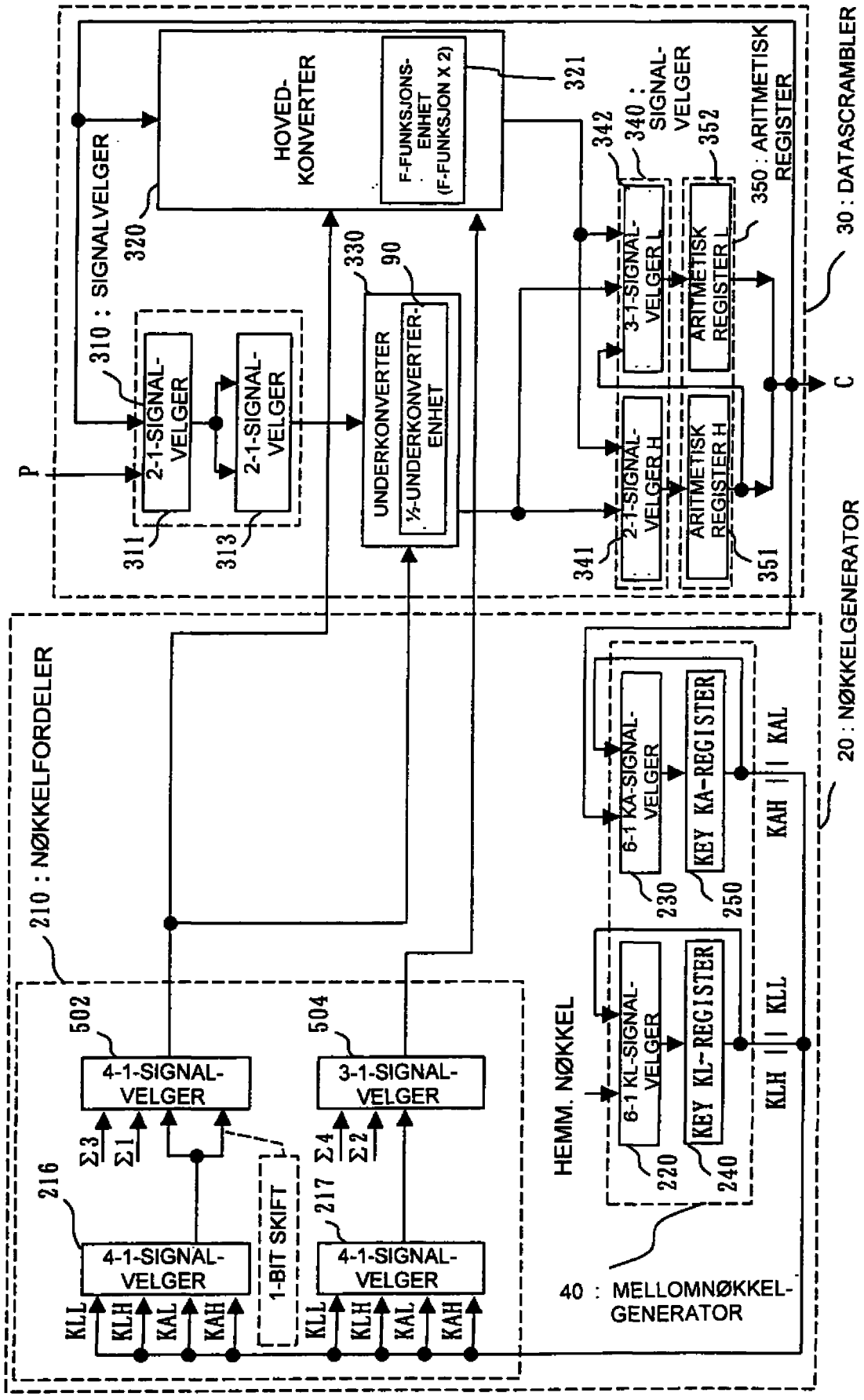

30 : DATASCRAMBLER
310 : SIGNALVELGER
320
HOVED-KONVERTER
F-FUNKSJONS-ENHET (F-FUNKSJON X 2)
321
342
340 : SIGNAL-VELGER
352
350 : ARITMETISK REGISTER
330
90
UNDERKONVERTER
½-UNDERKONVERTER-ENHET
3-1-SIGNAL-VELGER L
ARITMETISK REGISTER L
P
2-1-SIGNAL-VELGER
2-1-SIGNAL-VELGER
311
313
2-1-SIGNAL-VELGER H
ARITMETISK REGISTER H
341
351
C
20 : NØKKELGENERATOR
210 : NØKKELFORDELER
6-1 KA-SIGNAL-VELGER
KEY KA-REGISTER
230
250
KAL
KAH
502
504
4-1-SIGNAL-VELGER
3-1-SIGNAL-VELGER
Σ3
Σ1
Σ4
Σ2
HEMM. NØKKEL
6-1 KL-SIGNAL-VELGER
KEY KL-REGISTER
220
240
KLL
KLH
216
217
1-BIT SKIFT
4-1-SIGNAL-VELGER
4-1-SIGNAL-VELGER
KLL
KLH
KAL
KAH
40 : MELLOMNØKKEL-GENERATOR

Fig. 45

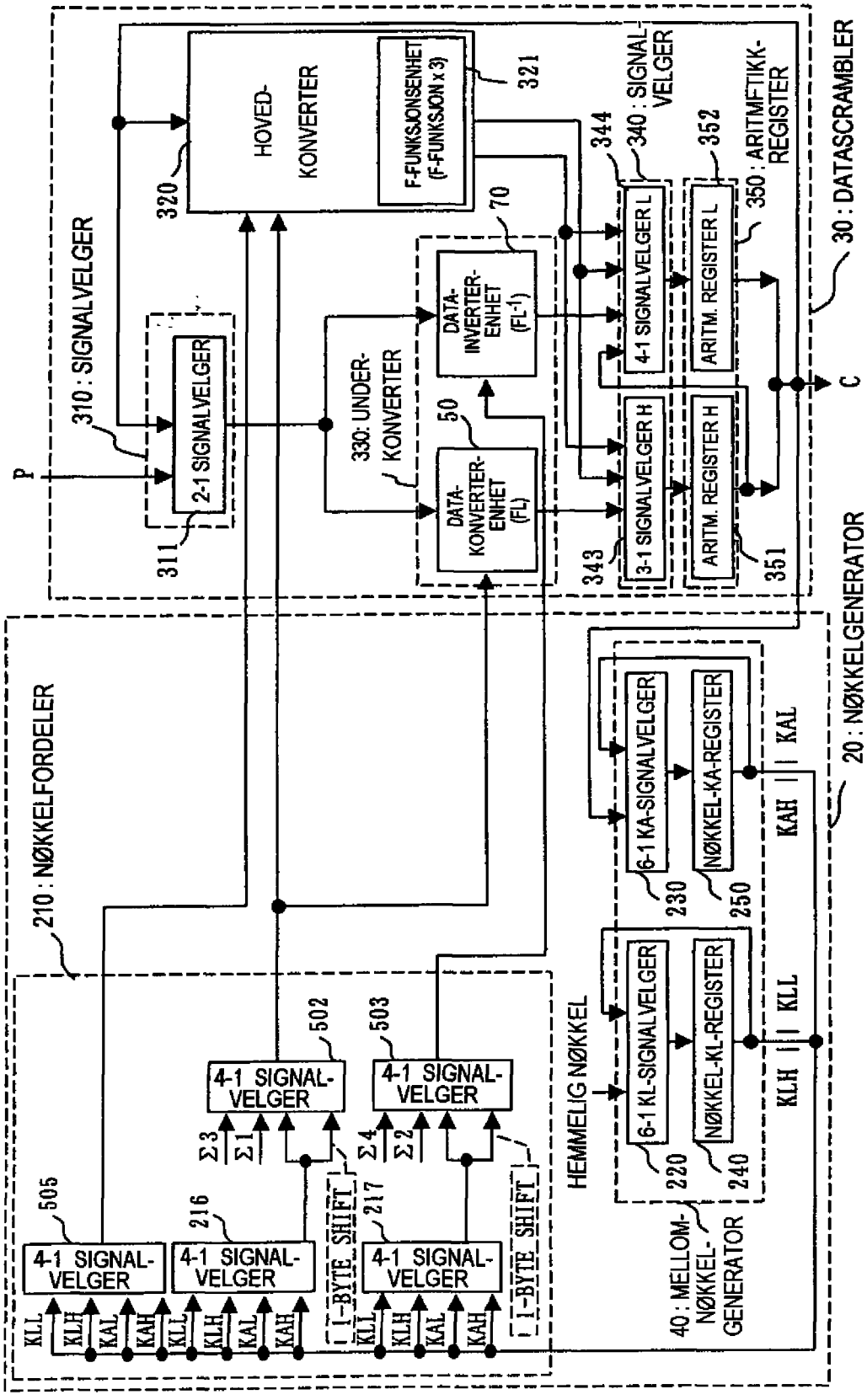

30 : DATASCRAMBLER
310 : SIGNALVELGER
320
HOVED-KONVERTER
F-FUNKSJONSENHET (F-FUNKSJON x 3)
321
344
340 : SIGNAL-VELGER
352
350 : ARITMETIKK-REGISTER
70
DATA-INVERTER-ENHET (FL-1)
4-1 SIGNALVELGER L
ARITM. REGISTER L
P
2-1 SIGNALVELGER
311
330: UNDER-KONVERTER
50
DATA-KONVERTER-ENHET (FL)
3-1 SIGNALVELGER H
ARITM. REGISTER H
343
351
C
20 : NØKKELGENERATOR
210 : NØKKELFORDELER
6-1 KA-SIGNALVELGER
230
NØKKEL-KA-REGISTER
250
KAL
KAH
HEMMELIG NØKKEL
6-1 KL-SIGNALVELGER
220
NØKKEL-KL-REGISTER
240
KLL
KLH
40 : MELLOM-NØKKEL-GENERATOR
502
503
4-1 SIGNAL-VELGER
4-1 SIGNAL-VELGER
Σ3
Σ1
Σ4
Σ2
505
216
217
4-1 SIGNAL-VELGER
4-1 SIGNAL-VELGER
4-1 SIGNAL-VELGER
1-BYTE SHIFT
1-BYTE SHIFT
KLL
KLH
KAL
KAH
KLL
KLH
KAL
KAH
KLL
KLH
KAL
KAH

Fig. 46

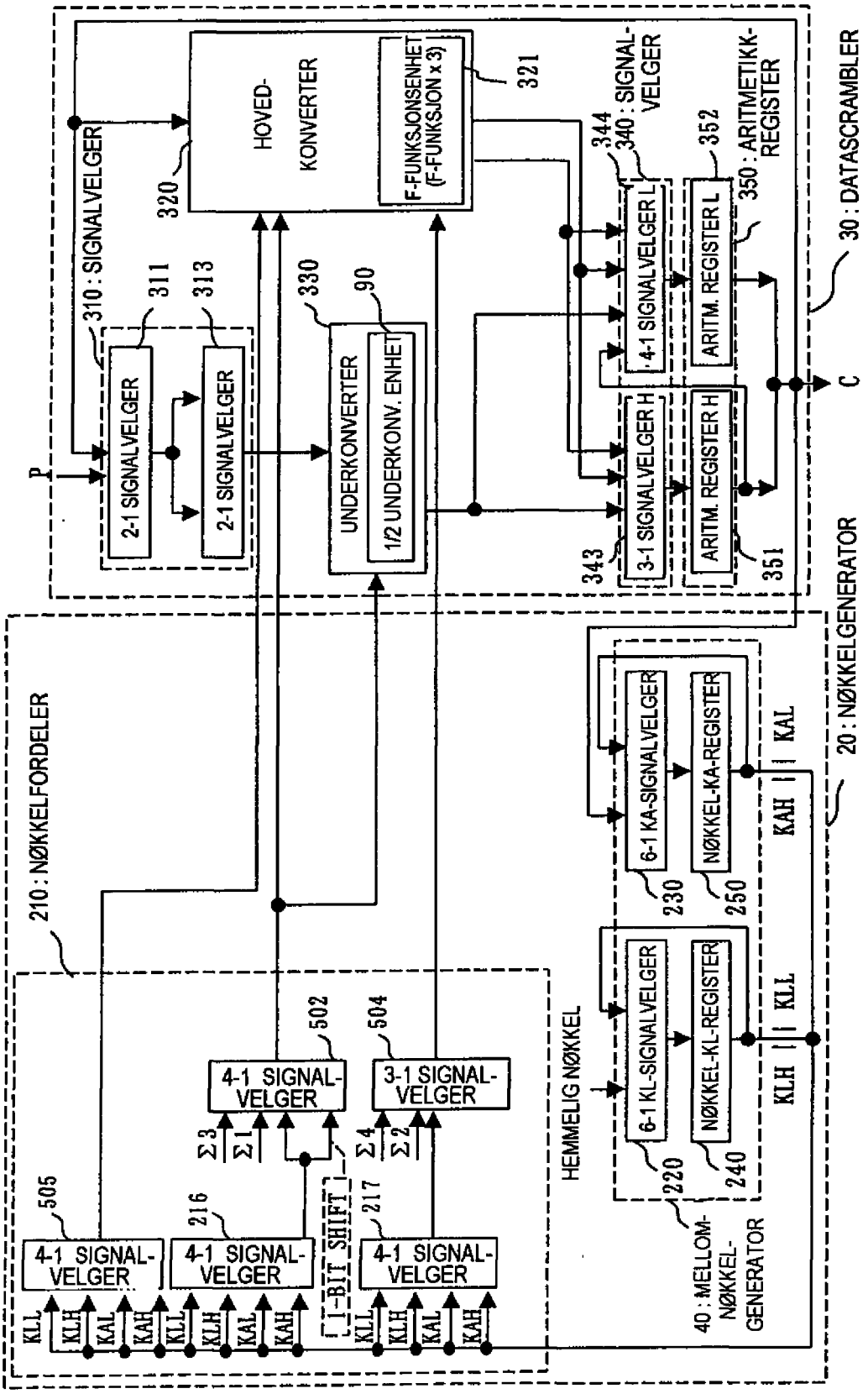

30 : DATASCRAMBLER
20 : NØKKELGENERATOR
210 : NØKKELFORDELER
310 : SIGNALVELGER
311
313
2-1 SIGNALVELGER
2-1 SIGNALVELGER
320
HOVED-
KONVERTER
F-FUNKSJONSENHET
(F-FUNKSJON x 3)
321
330
90
UNDERKONVERTER
1/2 UNDERKONV. ENHET
340 : SIGNAL-
VELGER
344
4-1 SIGNALVELGER L
3-1 SIGNALVELGER H
343
350 : ARITMETIKK-
REGISTER
352
ARITM. REGISTER L
ARITM. REGISTER H
351
P
C
4-1 SIGNAL-
VELGER
502
3-1 SIGNAL-
VELGER
504
Σ3
Σ1
Σ4
Σ2
1-BIT SHIFT
505
216
217
4-1 SIGNAL-
VELGER
4-1 SIGNAL-
VELGER
4-1 SIGNAL-
VELGER
KLL
KLH
KAL
KAH
HEMMELIG NØKKEL
6-1 KA-SIGNALVELGER
NØKKEL-KA-REGISTER
230
250
KAL
KAH
6-1 KL-SIGNALVELGER
NØKKEL-KL-REGISTER
220
240
KLH
KLL
40 : MELLOM-
NØKKEL-
GENERATOR

Fig. 47

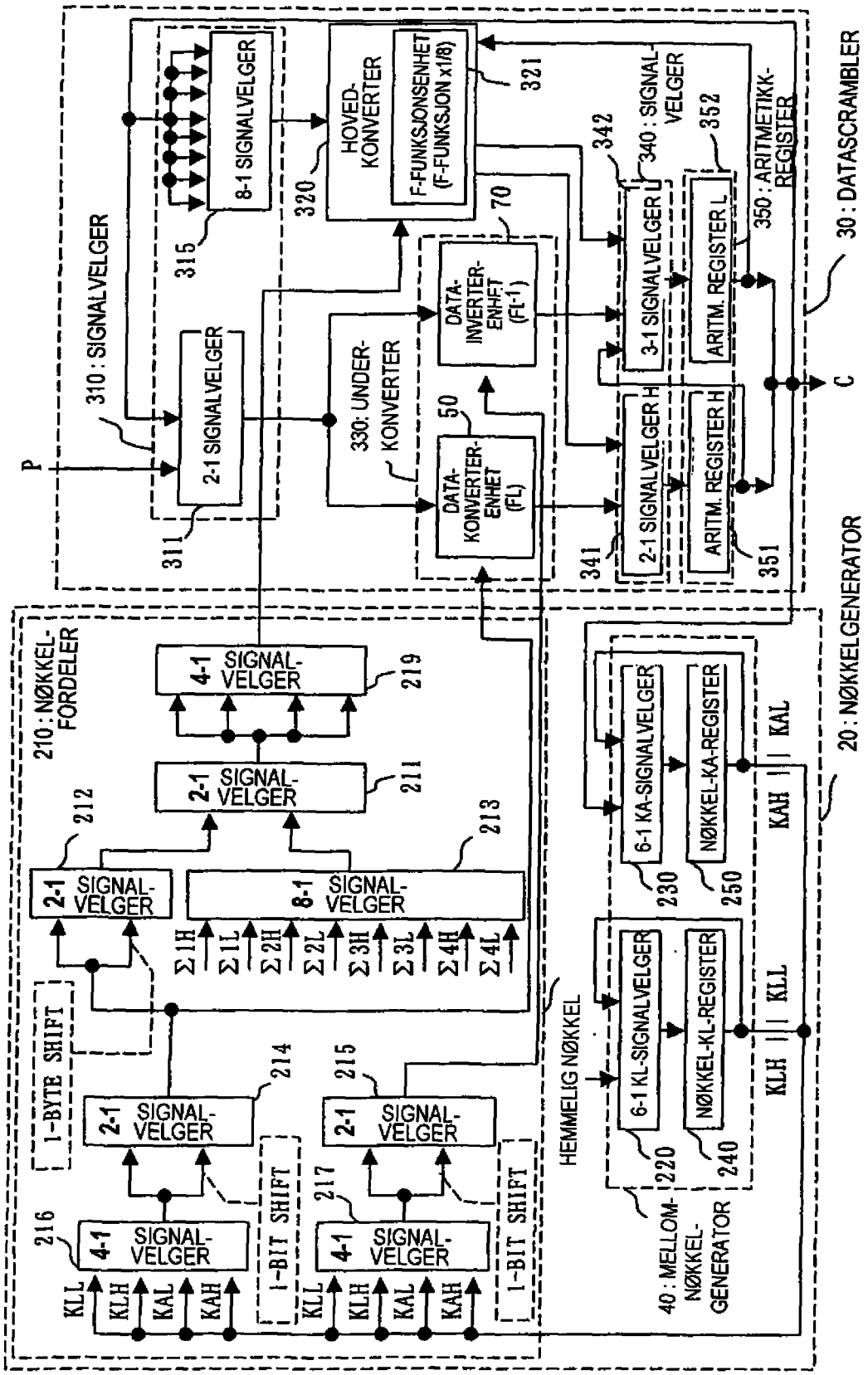

30 : DATASCRAMBLER
20 : NØKKELGENERATOR
310 : SIGNALVELGER
8-1 SIGNALVELGER
2-1 SIGNALVELGER
315
311
320
HOVED-KONVERTER
F-FUNKSJONSENHET (F-FUNKSJON x1/8)
321
330: UNDER-KONVERTER
70
DATA-INVERTER-ENHET (FI-1)
50
DATA-KONVERTER-ENHET (FL)
340 : SIGNAL-VELGER
342
3-1 SIGNALVELGER L
2-1 SIGNALVELGER H
341
350 : ARITMETIKK-REGISTER
352
ARITM. REGISTER L
ARITM. REGISTER H
351
P
C
210 : NØKKEL-FORDELER
4-1 SIGNAL-VELGER
219
2-1 SIGNAL-VELGER
211
212
213
2-1 SIGNAL-VELGER
8-1 SIGNAL-VELGER
Σ1H
Σ1L
Σ2H
Σ2L
Σ3H
Σ3L
Σ4H
Σ4L
1-BYTE SHIFT
214
215
2-1 SIGNAL-VELGER
2-1 SIGNAL-VELGER
HEMMELIG NØKKEL
216
217
4-1 SIGNAL-VELGER
4-1 SIGNAL-VELGER
1-BIT SHIFT
1-BIT SHIFT
KLL
KLH
KAL
KAH
KLL
KLH
KAL
KAH
6-1 KA-SIGNALVELGER
NØKKEL-KA-REGISTER
230
250
KAL
KAH
6-1 KL-SIGNALVELGER
NØKKEL-KL-REGISTER
220
240
KLL
KLH
40 : MELLOM-NØKKEL-GENERATOR

Fig. 48

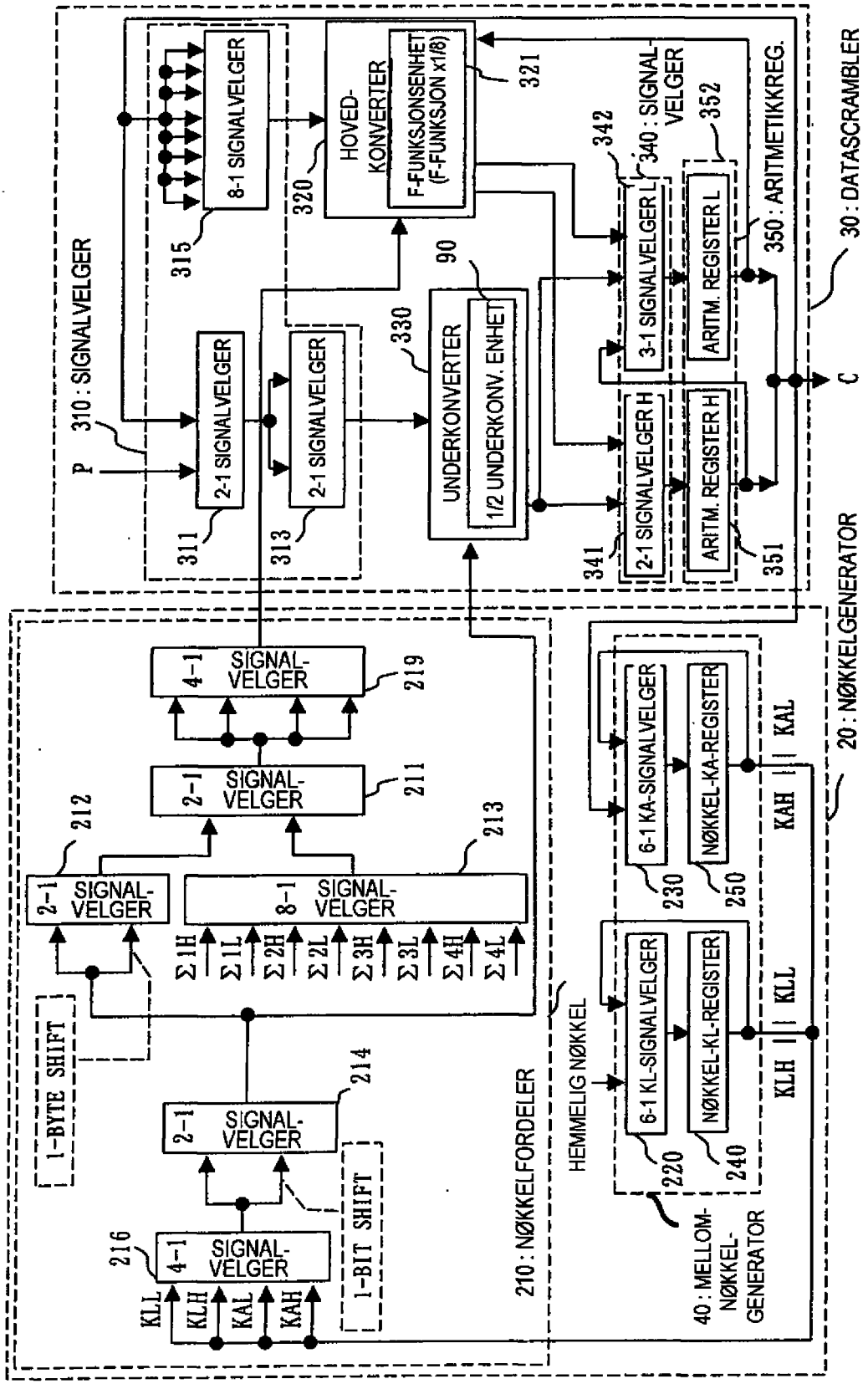

310 : SIGNALVELGER
P
8-1 SIGNALVELGER
315
320
HOVED-KONVERTER
F-FUNKSJONSENHET (F-FUNKSJON x1/8)
321
342
340 : SIGNAL-VELGER
352
2-1 SIGNALVELGER
311
2-1 SIGNALVELGER
313
330
UNDERKONVERTER
90
1/2 UNDERKONV. ENHET
3-1 SIGNALVELGER L
ARITM. REGISTER L
350 : ARITMETIKKREG.
30 : DATASCRAMBLER
C
2-1 SIGNALVELGER H
ARITM. REGISTER H
341
351
20 : NØKKELGENERATOR
4-1 SIGNAL-VELGER
219
2-1 SIGNAL-VELGER
211
212
213
2-1 SIGNAL-VELGER
8-1 SIGNAL-VELGER
Σ 1H
Σ 1L
Σ 2H
Σ 2L
Σ 3H
Σ 3L
Σ 4H
Σ 4L
1-BYTE SHIFT
214
2-1 SIGNAL-VELGER
1-BIT SHIFT
216
4-1 SIGNAL-VELGER
KLL
KLH
KAL
KAH
210 : NØKKELFORDELER
HEMMELIG NØKKEL
6-1 KA-SIGNALVELGER
NØKKEL-KA-REGISTER
230
250
KAL
KAH
6-1 KL-SIGNALVELGER
NØKKEL-KL-REGISTER
220
240
KLL
KLH
40 : MELLOM-NØKKEL-GENERATOR

Fig. 49

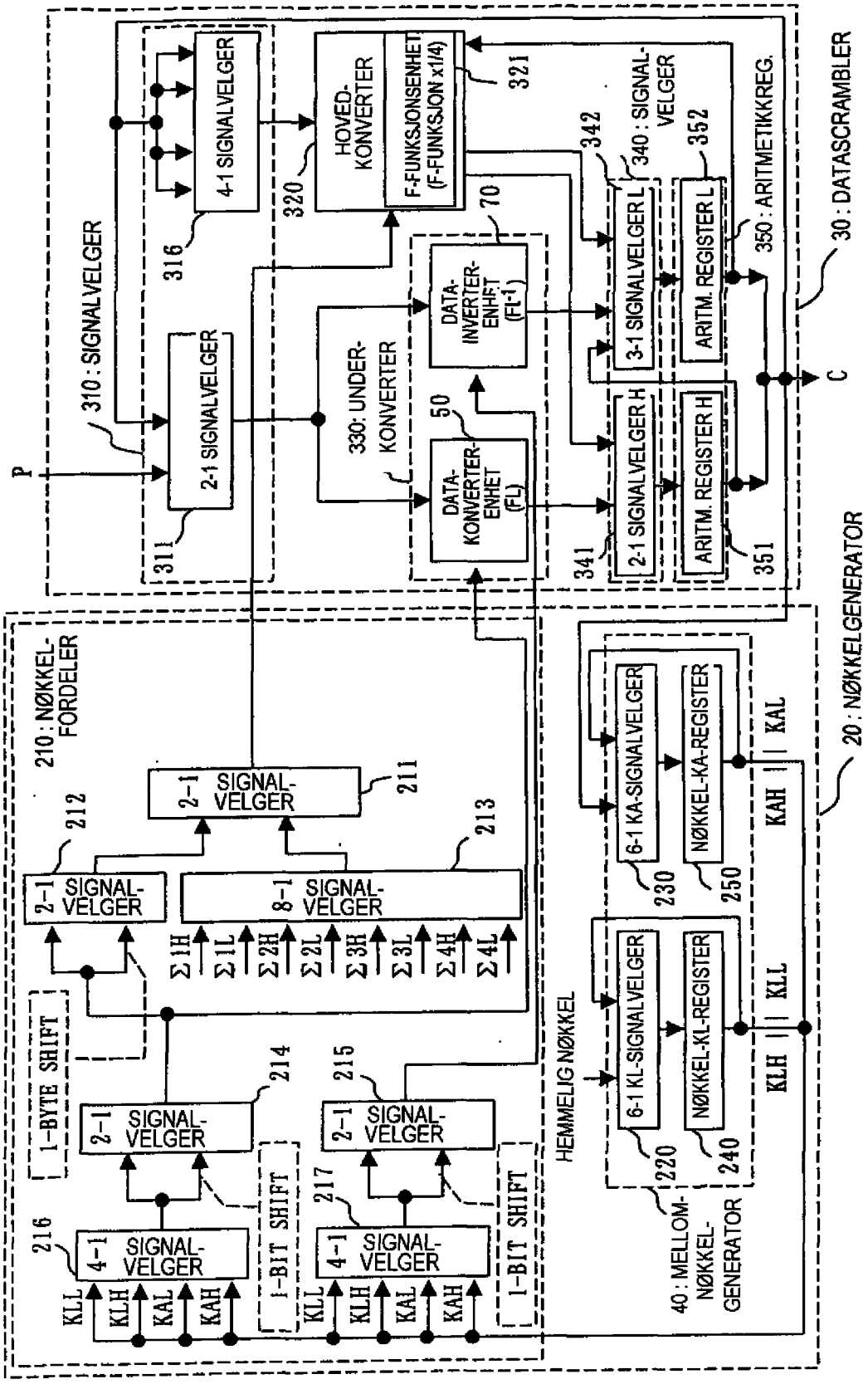

20 : NØKKELGENERATOR
30 : DATASCRAMBLER
310 : SIGNALVELGER
2-1 SIGNALVELGER
311
4-1 SIGNALVELGER
316
HOVED-KONVERTER
320
F-FUNKSJONSENHET (F-FUNKSJON x1/4)
321
330: UNDER-KONVERTER
DATA-INVERTER-ENHET (FI-1)
70
DATA-KONVERTER-ENHET (FL)
50
340 : SIGNAL-VELGER
342
3-1 SIGNALVELGER L
2-1 SIGNALVELGER H
341
350 : ARITMETIKKREG.
352
ARITM. REGISTER L
ARITM. REGISTER H
351
P
C
210 : NØKKEL-FORDELER
2-1 SIGNAL-VELGER
211
212
213
8-1 SIGNAL-VELGER
Σ1H Σ1L Σ2H Σ2L Σ3H Σ3L Σ4H Σ4L
1-BYTE SHIFT
214
215
1-BIT SHIFT
216
217
4-1 SIGNAL-VELGER
KLL KLH KAL KAH
HEMMELIG NØKKEL
6-1 KA-SIGNALVELGER
NØKKEL-KA-REGISTER
230
250
KAL KAH
6-1 KL-SIGNALVELGER
NØKKEL-KL-REGISTER
220
240
KLL KLH
40 : MELLOM-NØKKEL-GENERATOR

Fig. 50

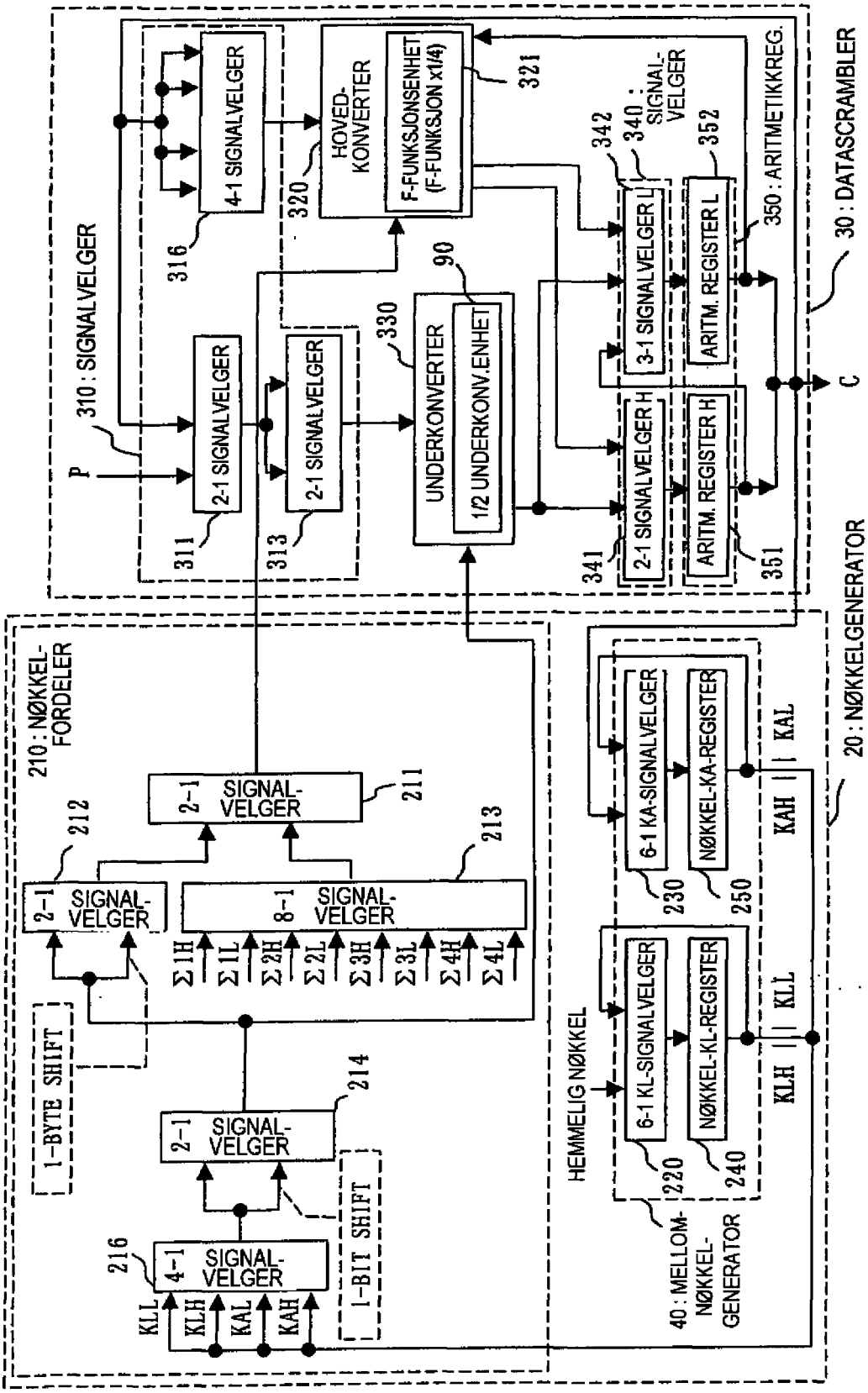

310 : SIGNALVELGER
P
311
2-1 SIGNALVELGER
313
2-1 SIGNALVELGER
316
4-1 SIGNALVELGER
320
HOVED-KONVERTER
321
F-FUNKSJONSENHET (F-FUNKSJON x1/4)
330
UNDERKONVERTER
90
1/2 UNDERKONV.ENHET
341
2-1 SIGNALVELGER H
342
340 : SIGNAL-VELGER
3-1 SIGNALVELGER L
350 : ARITMETIKKREG.
ARITM. REGISTER H
351
ARITM. REGISTER L
352
30 : DATASCRAMBLER
C
20 : NØKKELGENERATOR
210 : NØKKEL-FORDELER
212
2-1 SIGNAL-VELGER
211
2-1 SIGNAL-VELGER
213
8-1 SIGNAL-VELGER
Σ1H Σ1L Σ2H Σ2L Σ3H Σ3L Σ4H Σ4L
1-BYTE SHIFT
214
2-1 SIGNAL-VELGER
1-BIT SHIFT
216
4-1 SIGNAL-VELGER
KLL KLH KAL KAH
HEMMELIG NØKKEL
6-1 KA-SIGNALVELGER
230
NØKKEL-KA-REGISTER
250
KAL KAH
6-1 KL-SIGNALVELGER
220
NØKKEL-KL-REGISTER
240
KLL KLH
40 : MELLOM-NØKKEL-GENERATOR

Fig. 51

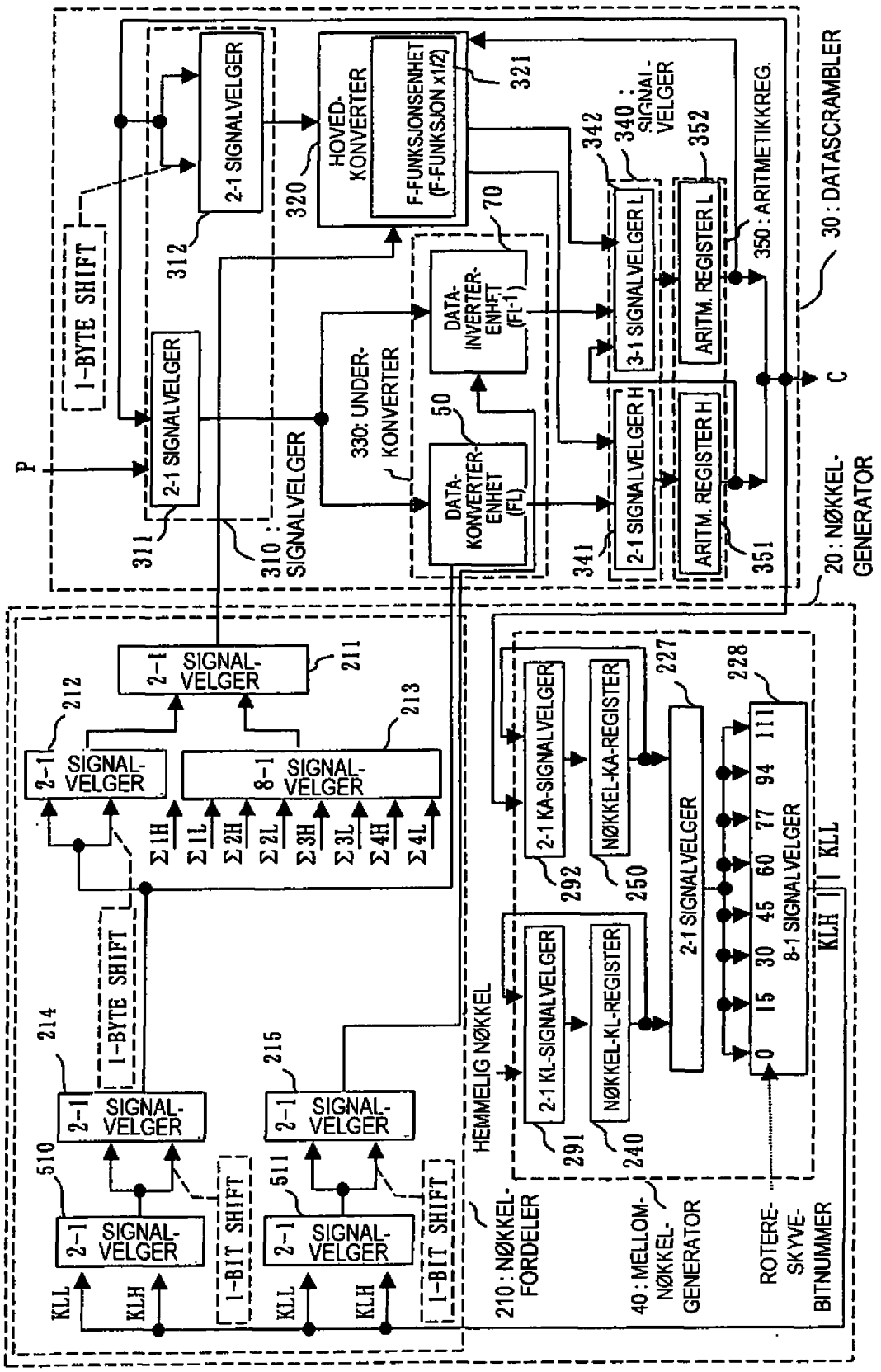

20 : NØKKEL-GENERATOR
30 : DATASCRAMBLER
1-BYTE SHIFT
2-1 SIGNALVELGER
310 : SIGNALVELGER
311
312
HOVED-KONVERTER
320
F-FUNKSJONSENHET (F-FUNKSJON x1/2)
321
330: UNDER-KONVERTER
DATA-KONVERTER-ENHET (FL)
50
DATA-INVERTER-ENHET (FL-1)
70
340 : SIGNAL-VELGER
341
342
3-1 SIGNALVELGER L
2-1 SIGNALVELGER H
ARITM. REGISTER L
ARITM. REGISTER H
350 : ARITMETIKKREG.
351
352
P
C
2-1 SIGNAL-VELGER
211
212
213
8-1 SIGNAL-VELGER
Σ1H
Σ1L
Σ2H
Σ2L
Σ3H
Σ3L
Σ4H
Σ4L
1-BYTE SHIFT
214
215
510
511
1-BIT SHIFT
KLL
KLH
HEMMELIG NØKKEL
210 : NØKKEL-FORDELER
2-1 KA-SIGNALVELGER
2-1 KL-SIGNALVELGER
NØKKEL-KA-REGISTER
NØKKEL-KL-REGISTER
292
291
250
240
227
228
2-1 SIGNALVELGER
8-1 SIGNALVELGER
111
0 15 30 45 60 77 94
40 : MELLOM-NØKKEL-GENERATOR
ROTERE-SKYVE-BITNUMMER

Fig. 52

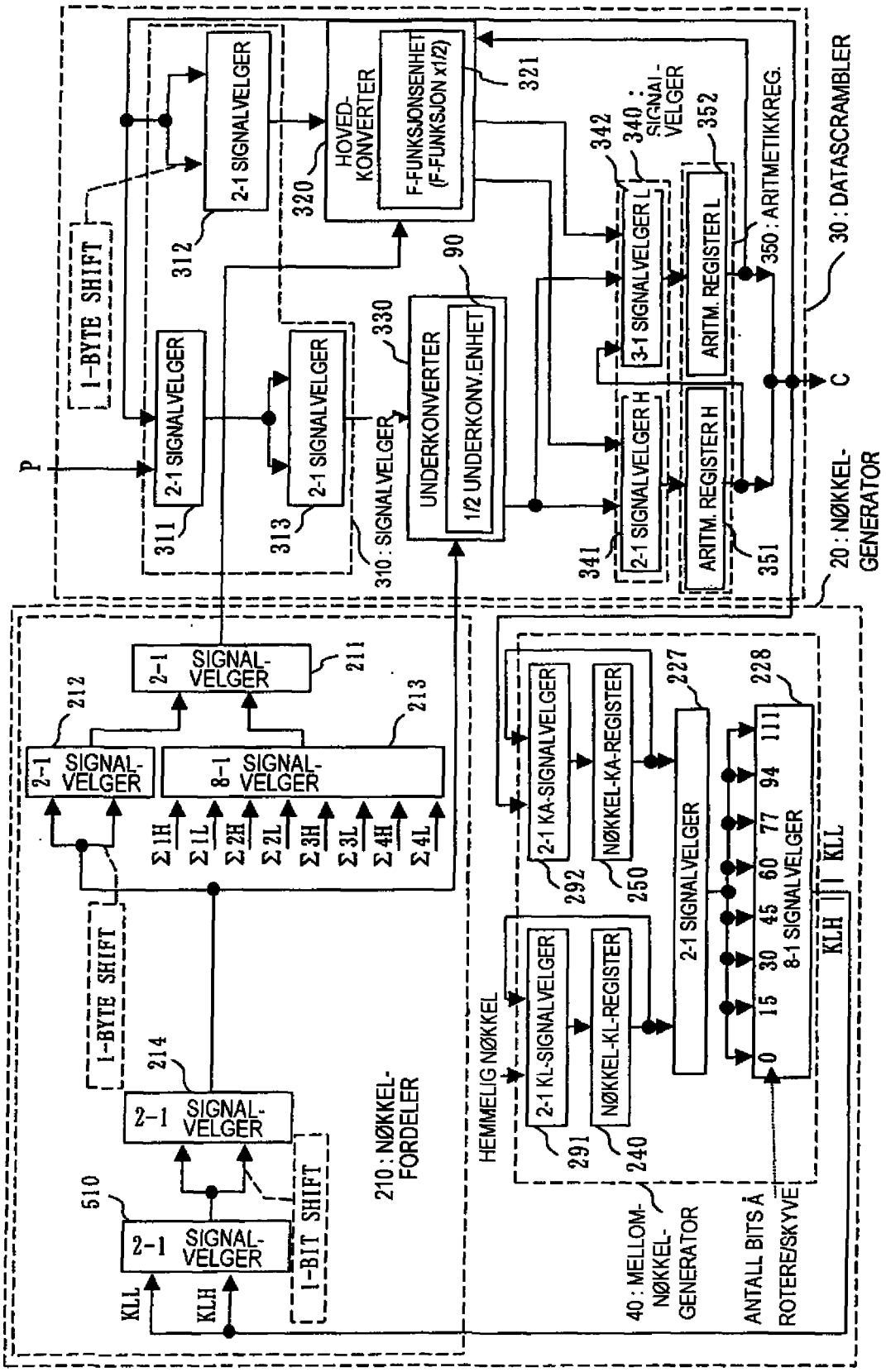

30 : DATASCRAMBLER
20 : NØKKEL-GENERATOR
1-BYTE SHIFT
2-1 SIGNALVELGER
HOVED-KONVERTER
F-FUNKSJONSENHET (F-FUNKSJON x1/2)
321
320
312
311
313
310 : SIGNALVELGER
330
UNDERKONVERTER
1/2 UNDERKONV.ENHET
90
342
340 : SIGNAL-VELGER
352
3-1 SIGNALVELGER
ARITM. REGISTER L
ARITM. REGISTER H
350 : ARITMETIKKREG.
341
351
P
C
2-1 SIGNAL-VELGER
8-1 SIGNAL-VELGER
211
212
213
Σ 1H
Σ 1L
Σ 2H
Σ 2L
Σ 3H
Σ 3L
Σ 4H
Σ 4L
1-BIT SHIFT
214
510
KLL
KLH
210 : NØKKEL-FORDELER
HEMMELIG NØKKEL
2-1 KA-SIGNALVELGER
NØKKEL-KA-REGISTER
227
228
292
250
2-1 KL-SIGNALVELGER
NØKKEL-KL-REGISTER
291
240
0 15 30 45 60 77 94 111
40 : MELLOM-NØKKEL-GENERATOR
ANTALL BITS Å ROTERE/SKYVE

Fig. 53

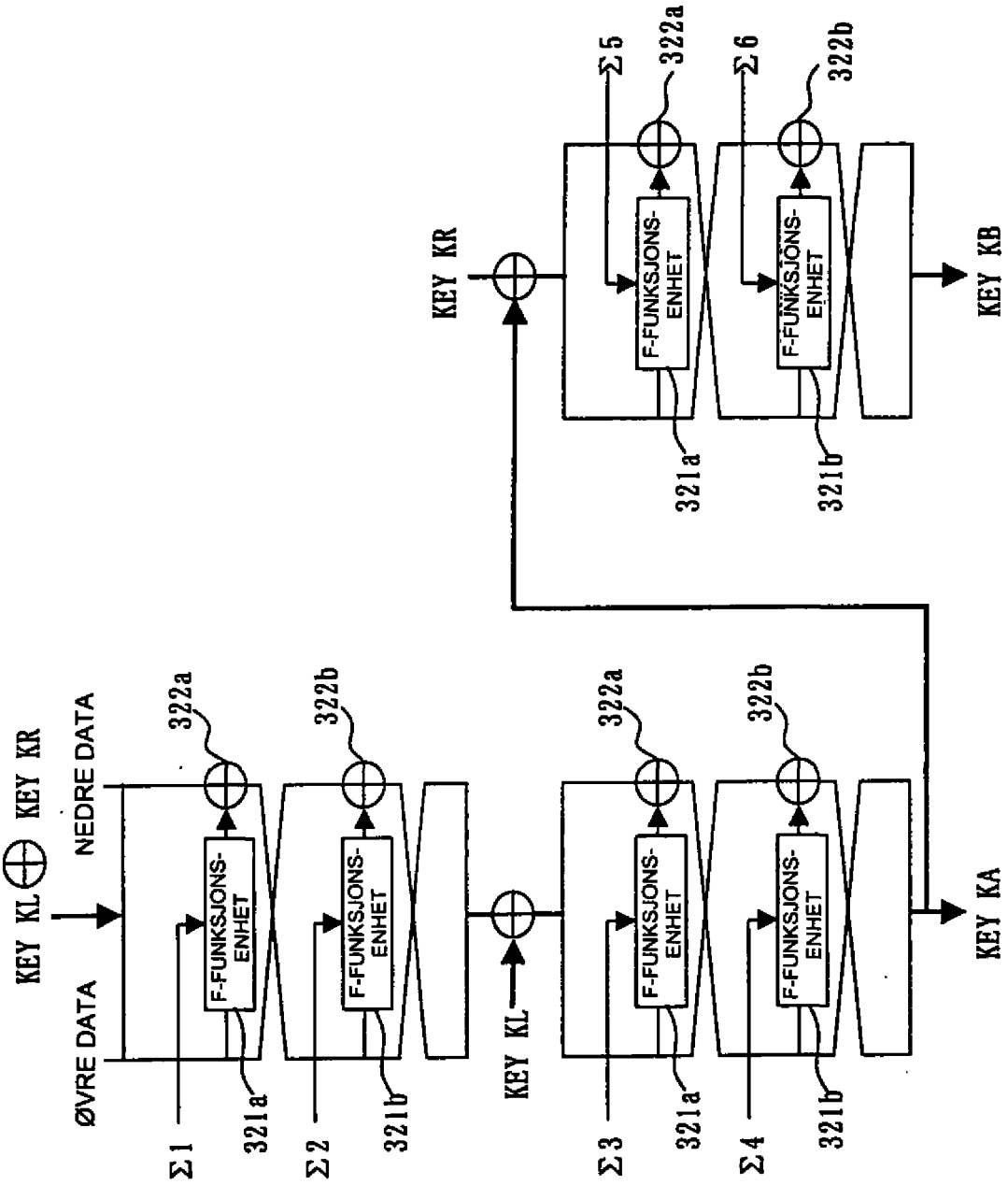

KEY KL ⊕ KEY KR
NEDRE DATA
ØVRE DATA
F-FUNKSJONS-ENHET
322a
322b
321a
321b
Σ1
Σ2
Σ3
Σ4
Σ5
Σ6
KEY KL
KEY KR
KEY KA
KEY KB

Fig. 54

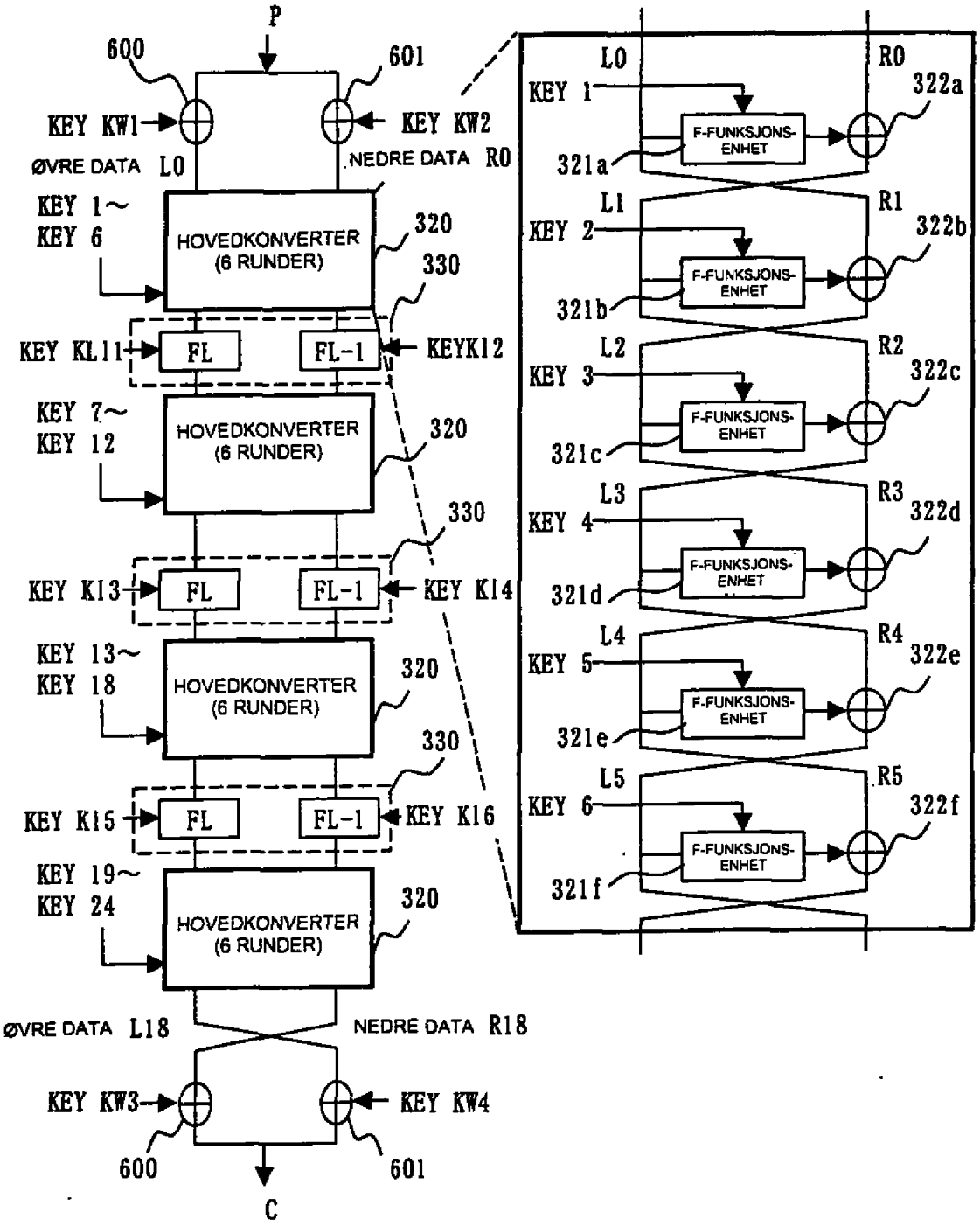

P
600
601
KEY KW1
KEY KW2
ØVRE DATA L0
NEDRE DATA R0
KEY 1~
KEY 6
HOVEDKONVERTER
(6 RUNDER)
320
330
KEY KL11
FL
FL-1
KEYK12
KEY 7~
KEY 12
HOVEDKONVERTER
(6 RUNDER)
320
330
KEY K13
FL
FL-1
KEY K14
KEY 13~
KEY 18
HOVEDKONVERTER
(6 RUNDER)
320
330
KEY K15
FL
FL-1
KEY K16
KEY 19~
KEY 24
HOVEDKONVERTER
(6 RUNDER)
320
ØVRE DATA L18
NEDRE DATA R18
KEY KW3
KEY KW4
600
601
C
L0
R0
KEY 1
322a
F-FUNKSJONS-
ENHET
321a
L1
R1
KEY 2
322b
F-FUNKSJONS-
ENHET
321b
L2
R2
KEY 3
322c
F-FUNKSJONS-
ENHET
321c
L3
R3
KEY 4
322d
F-FUNKSJONS-
ENHET
321d
L4
R4
KEY 5
322e
F-FUNKSJONS-
ENHET
321e
L5
R5
KEY 6
322f
F-FUNKSJONS-
ENHET
321f

Fig. 55

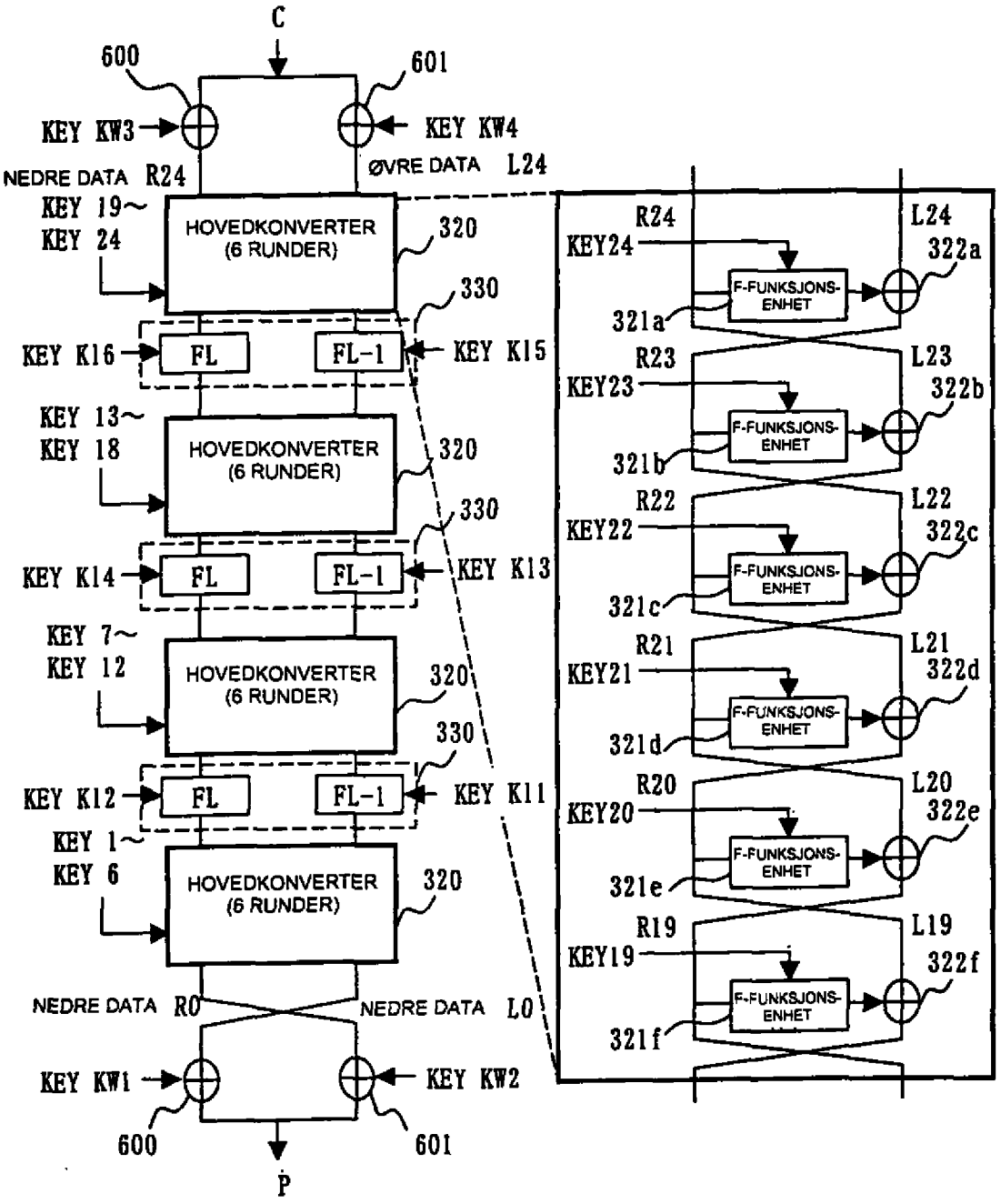

C
600
601
KEY KW3
KEY KW4
NEDRE DATA R24
ØVRE DATA L24
KEY 19~
KEY 24
HOVEDKONVERTER
(6 RUNDER)
320
330
KEY K16
FL
FL-1
KEY K15
KEY 13~
KEY 18
KEY K14
KEY K13
KEY 7~
KEY 12
KEY K12
KEY K11
KEY 1~
KEY 6
NEDRE DATA R0
NEDRE DATA L0
KEY KW1
KEY KW2
P
R24
L24
KEY24
322a
F-FUNKSJONS-
ENHET
321a
R23
L23
KEY23
322b
321b
R22
L22
KEY22
322c
321c
R21
L21
KEY21
322d
321d
R20
L20
KEY20
322e
321e
R19
L19
KEY19
322f
321f

Fig. 56

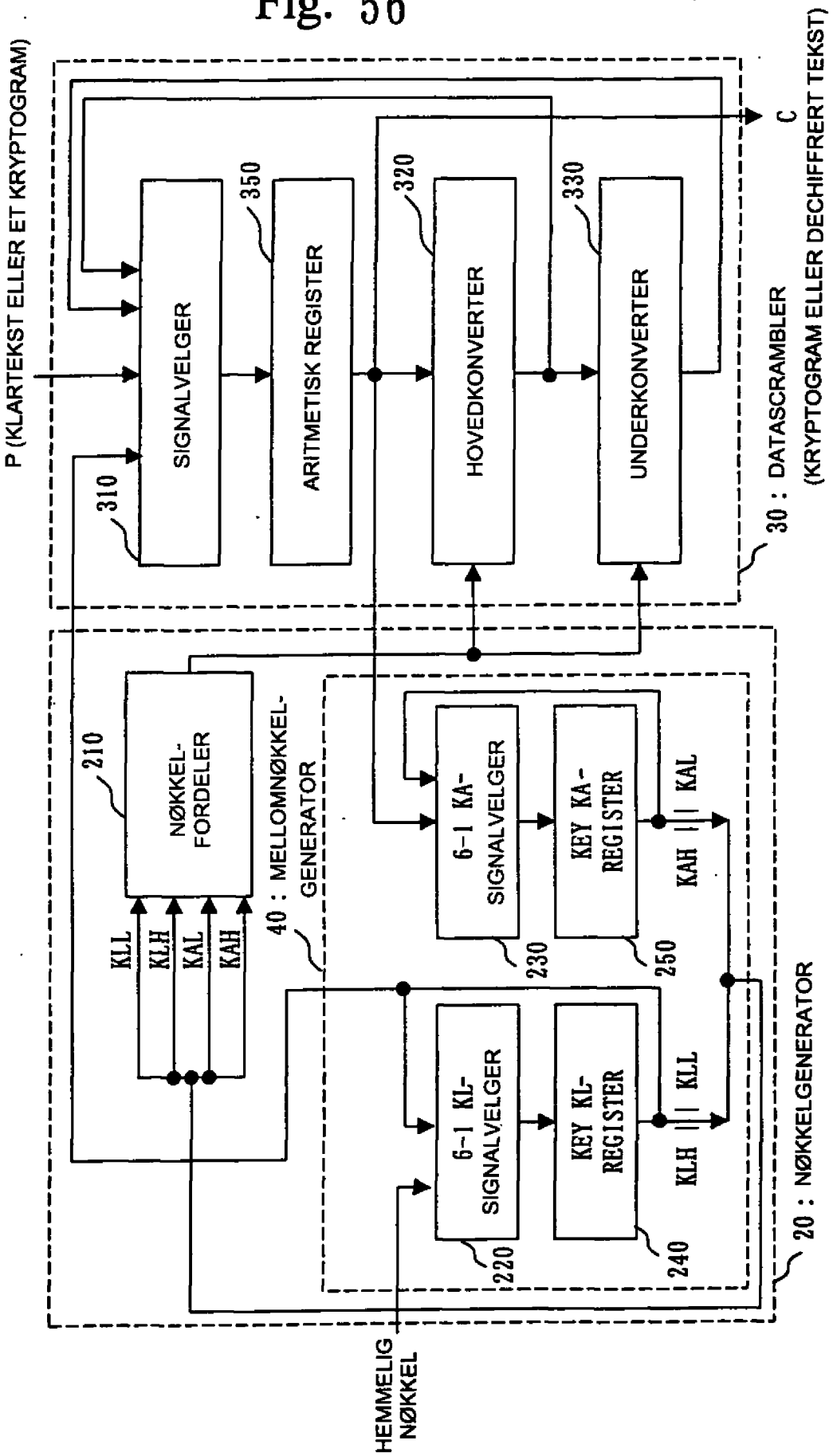

P (KLARTEKST ELLER ET KRYPTOGRAM)
SIGNALVELGER
ARITMETISK REGISTER
HOVEDKONVERTER
UNDERKONVERTER
C
(KRYPTOGRAM ELLER DECHIFFRERT TEKST)
30 : DATASCRAMBLER
310
350
320
330
NØKKEL-FORDELER
210
KLL
KLH
KAL
KAH
40 : MELLOMNØKKEL-GENERATOR
6-1 KA-SIGNALVELGER
KEY KA-REGISTER
230
250
6-1 KL-SIGNALVELGER
KEY KL-REGISTER
220
240
20 : NØKKELGENERATOR
HEMMELIG NØKKEL

Fig. 57

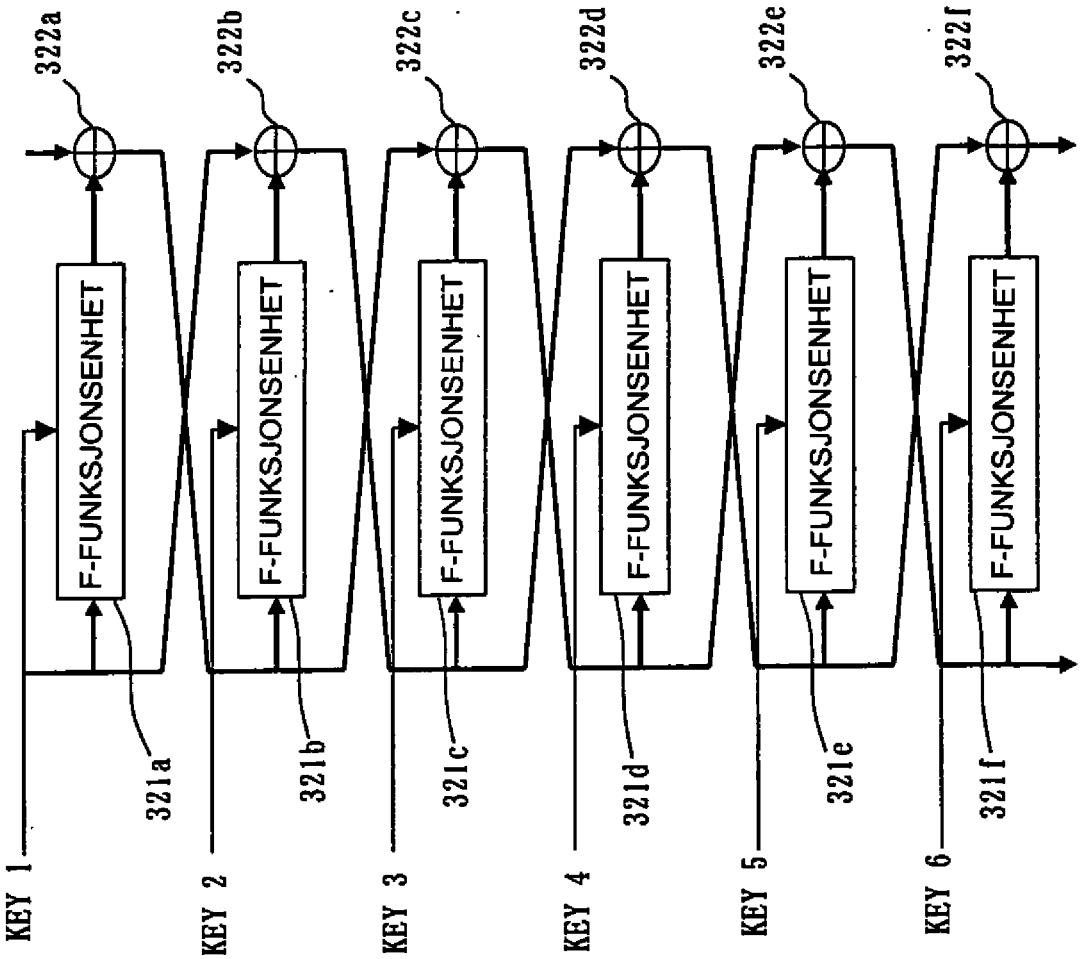

322a
322b
322c
322d
322e
322f
F-FUNKSJONSENHET
F-FUNKSJONSENHET
F-FUNKSJONSENHET
F-FUNKSJONSENHET
F-FUNKSJONSENHET
F-FUNKSJONSENHET
321a
321b
321c
321d
321e
321f
KEY 1
KEY 2
KEY 3
KEY 4
KEY 5
KEY 6

Fig. 58

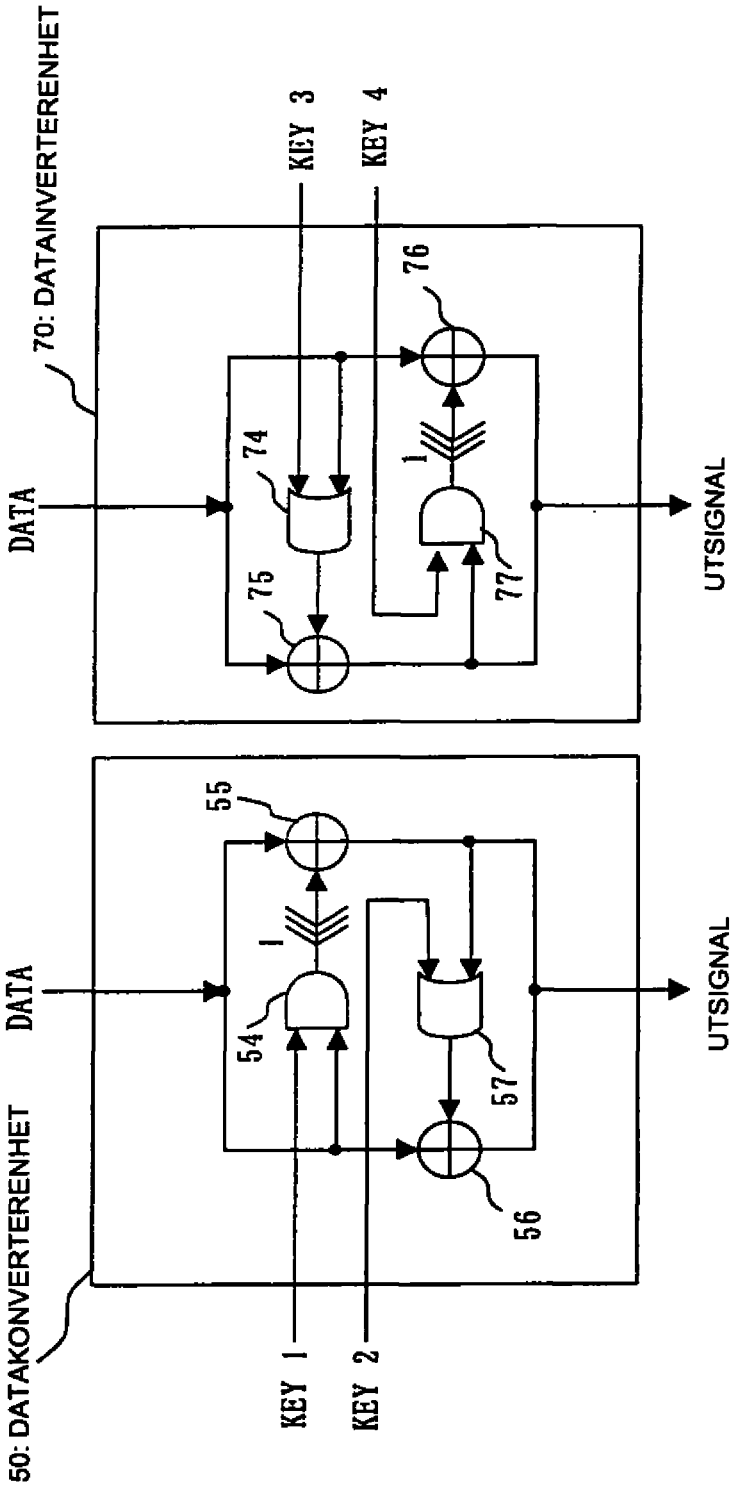

70: DATAINVERTERENHET
KEY 3
KEY 4
76
74
DATA
75
77
UTSIGNAL
55
54
57
56
50: DATAKONVERTERENHET
KEY 1
KEY 2

Fig. 59

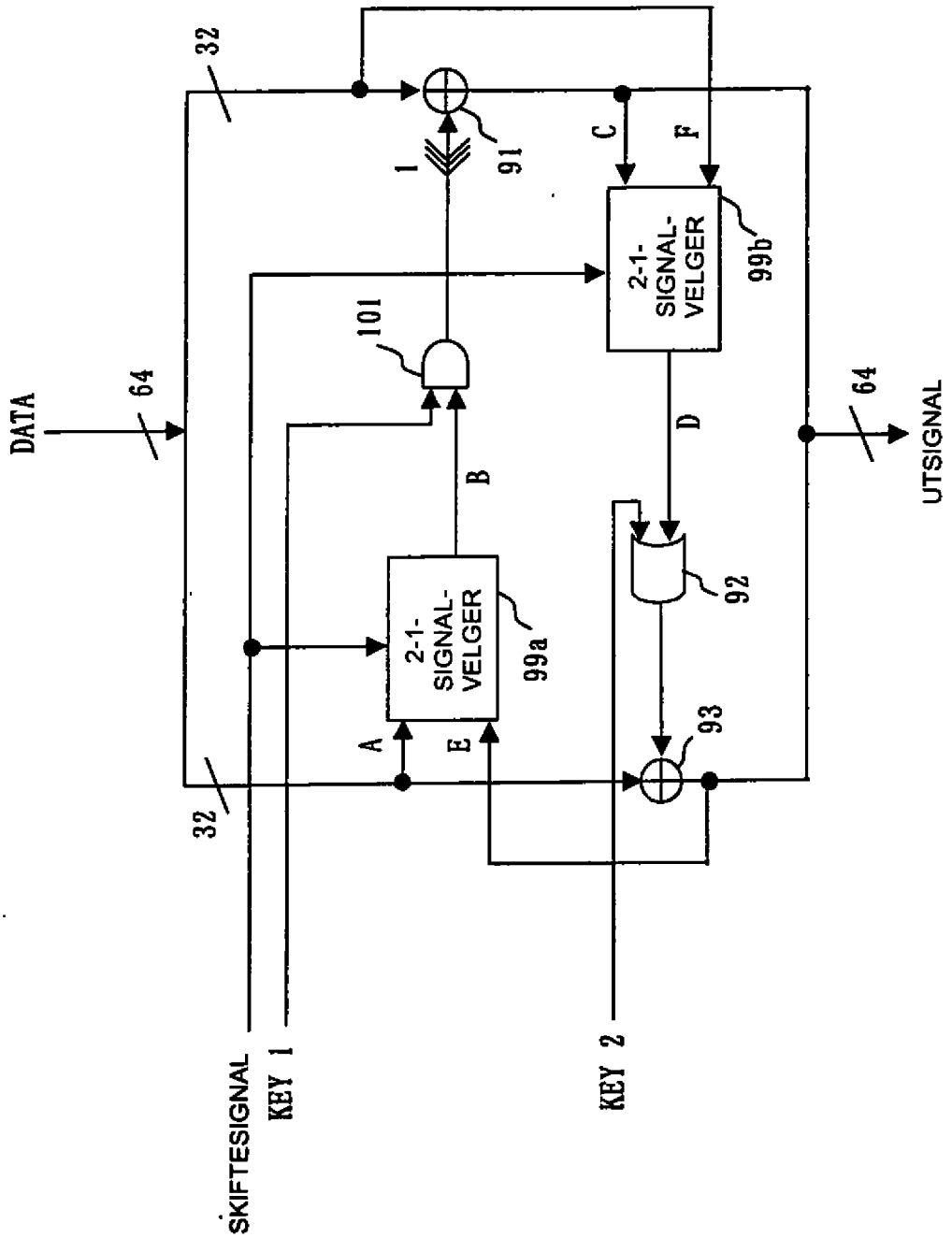

32
91
1
C
F
2-1-
SIGNAL-
VELGER
99b
101
DATA
64
B
D
64
UTSIGNAL
92
2-1-
SIGNAL-
VELGER
99a
A
E
93
32
SKIFTESIGNAL
KEY 1
KEY 2

Fig. 60

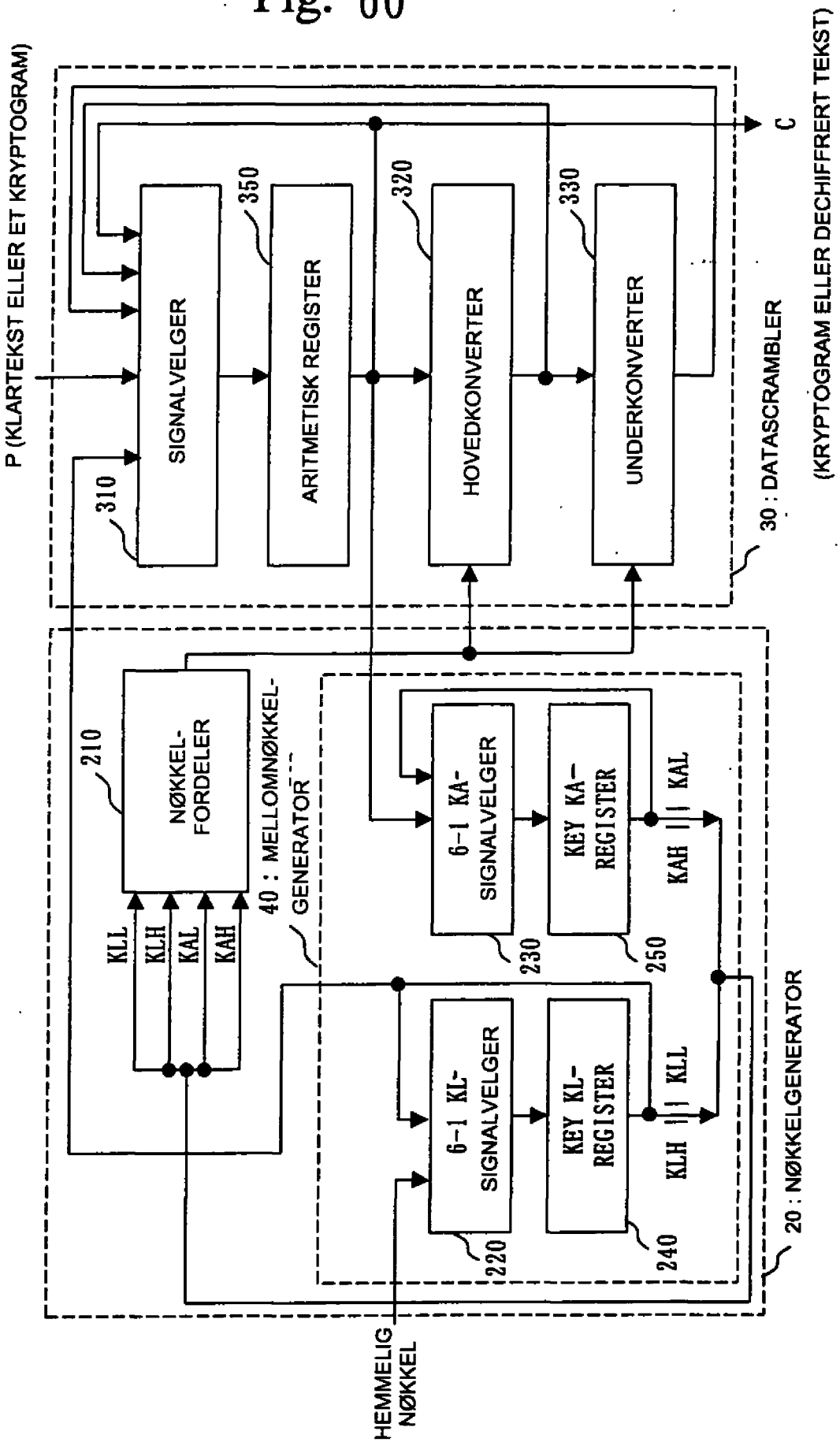

P (KLARTEKST ELLER ET KRYPTOGRAM)
SIGNALVELGER
ARITMETISK REGISTER
HOVEDKONVERTER
UNDERKONVERTER
C
(KRYPTOGRAM ELLER DECHIFFRERT TEKST)
310
350
320
330
30 : DATASCRAMBLER
NØKKEL-FORDELER
210
40 : MELLOMNØKKEL-GENERATOR
KLL
KLH
KAL
KAH
6-1 KA-SIGNALVELGER
KEY KA-REGISTER
230
250
6-1 KL-SIGNALVELGER
KEY KL-REGISTER
220
240
20 : NØKKELGENERATOR
HEMMELIG NØKKEL

Fig. 61

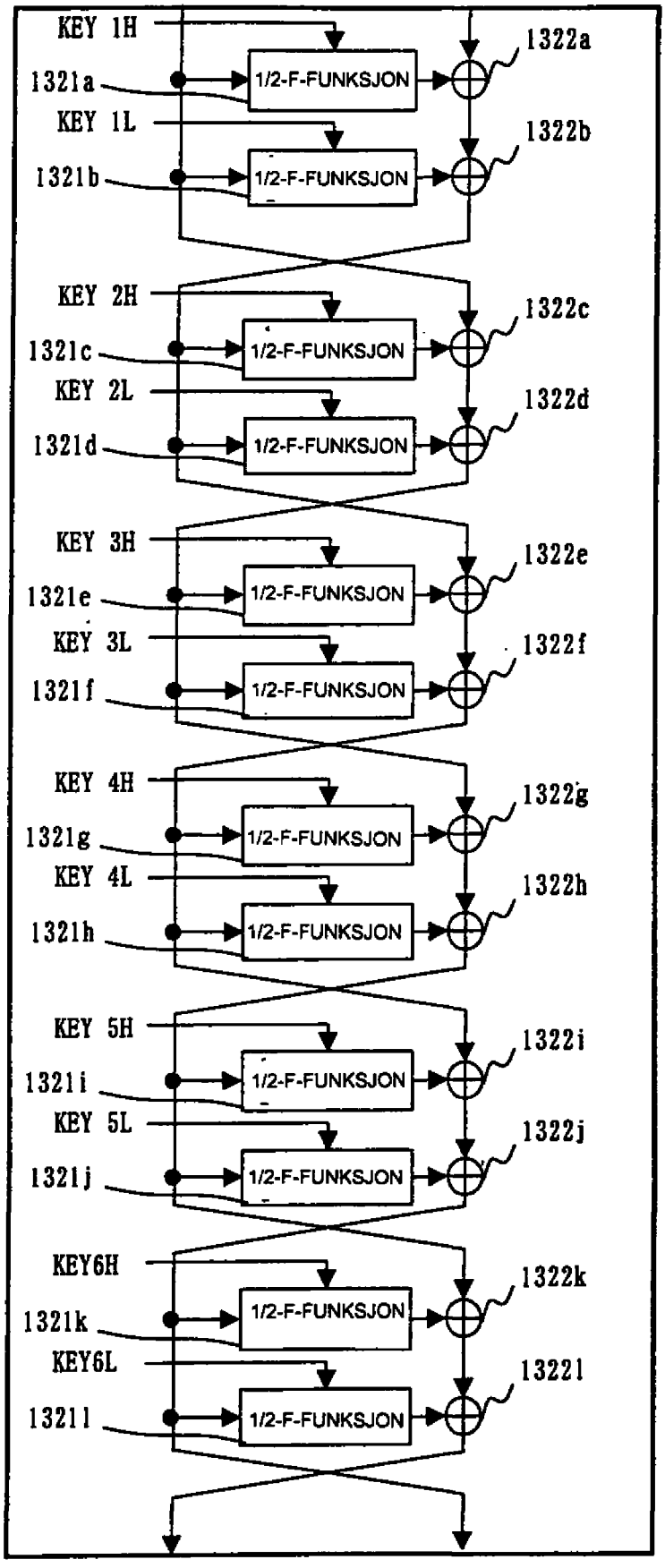

KEY 1H
1321a
1/2-F-FUNKSJON
1322a
KEY 1L
1321b
1/2-F-FUNKSJON
1322b
KEY 2H
1321c
1/2-F-FUNKSJON
1322c
KEY 2L
1321d
1/2-F-FUNKSJON
1322d
KEY 3H
1321e
1/2-F-FUNKSJON
1322e
KEY 3L
1321f
1/2-F-FUNKSJON
1322f
KEY 4H
1321g
1/2-F-FUNKSJON
1322g
KEY 4L
1321h
1/2-F-FUNKSJON
1322h
KEY 5H
1321i
1/2-F-FUNKSJON
1322i
KEY 5L
1321j
1/2-F-FUNKSJON
1322j
KEY6H
1321k
1/2-F-FUNKSJON
1322k
KEY6L
13211
1/2-F-FUNKSJON
13221

Fig. 62

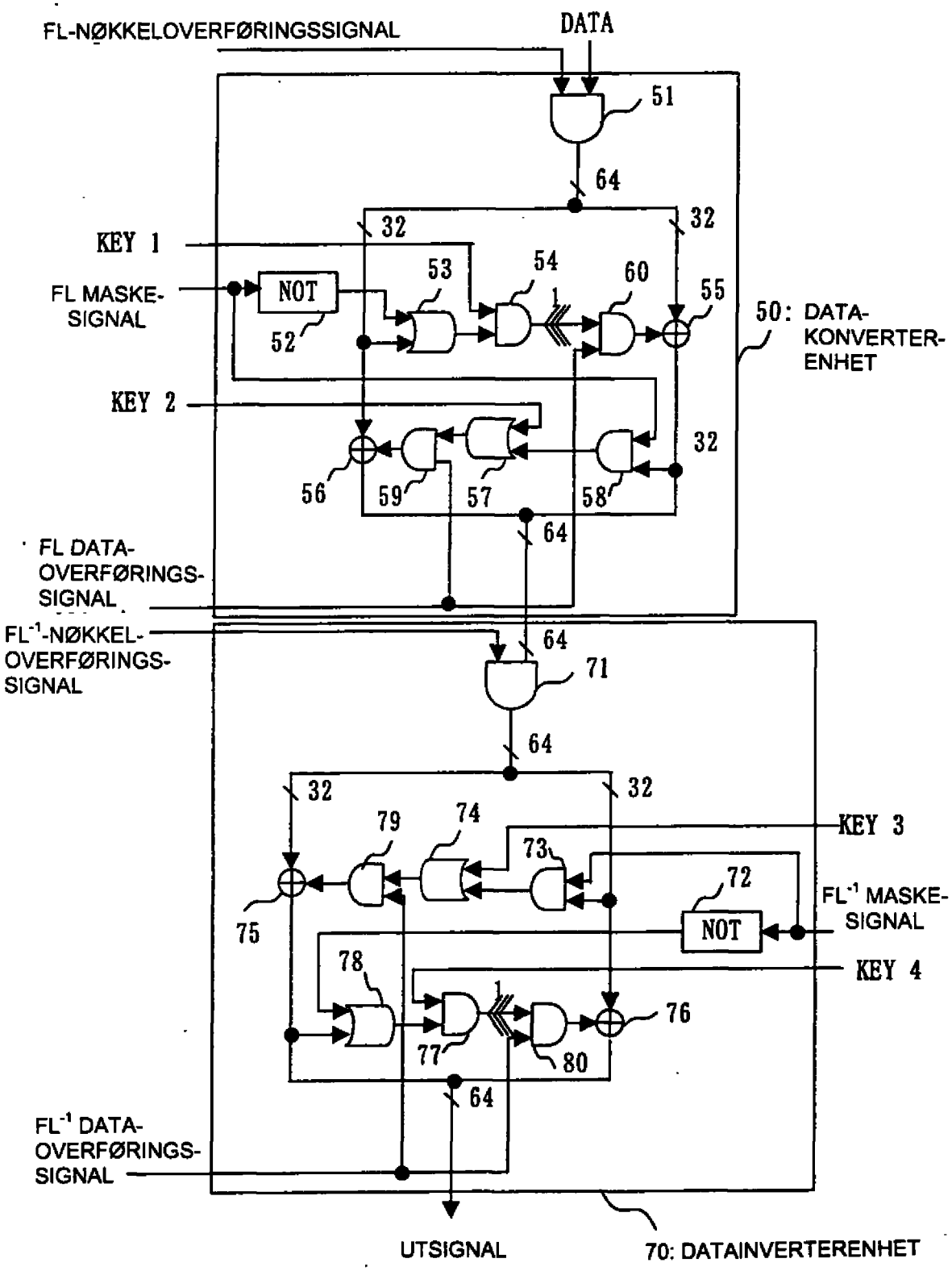

FL-NØKKELOVERFØRINGSSIGNAL
DATA
51
64
32
32
KEY 1
FL MASKE-SIGNAL
NOT
52
53
54
60
55
50: DATA-KONVERTER-ENHET
KEY 2
56
59
57
58
32
64
FL DATA-OVERFØRINGS-SIGNAL
FL-1-NØKKEL-OVERFØRINGS-SIGNAL
64
71
64
32
32
KEY 3
79
74
73
72
FL-1 MASKE-SIGNAL
75
NOT
78
KEY 4
76
77
80
64
FL-1 DATA-OVERFØRINGS-SIGNAL
UTSIGNAL
70: DATAINVERTERENHET

Fig. 63

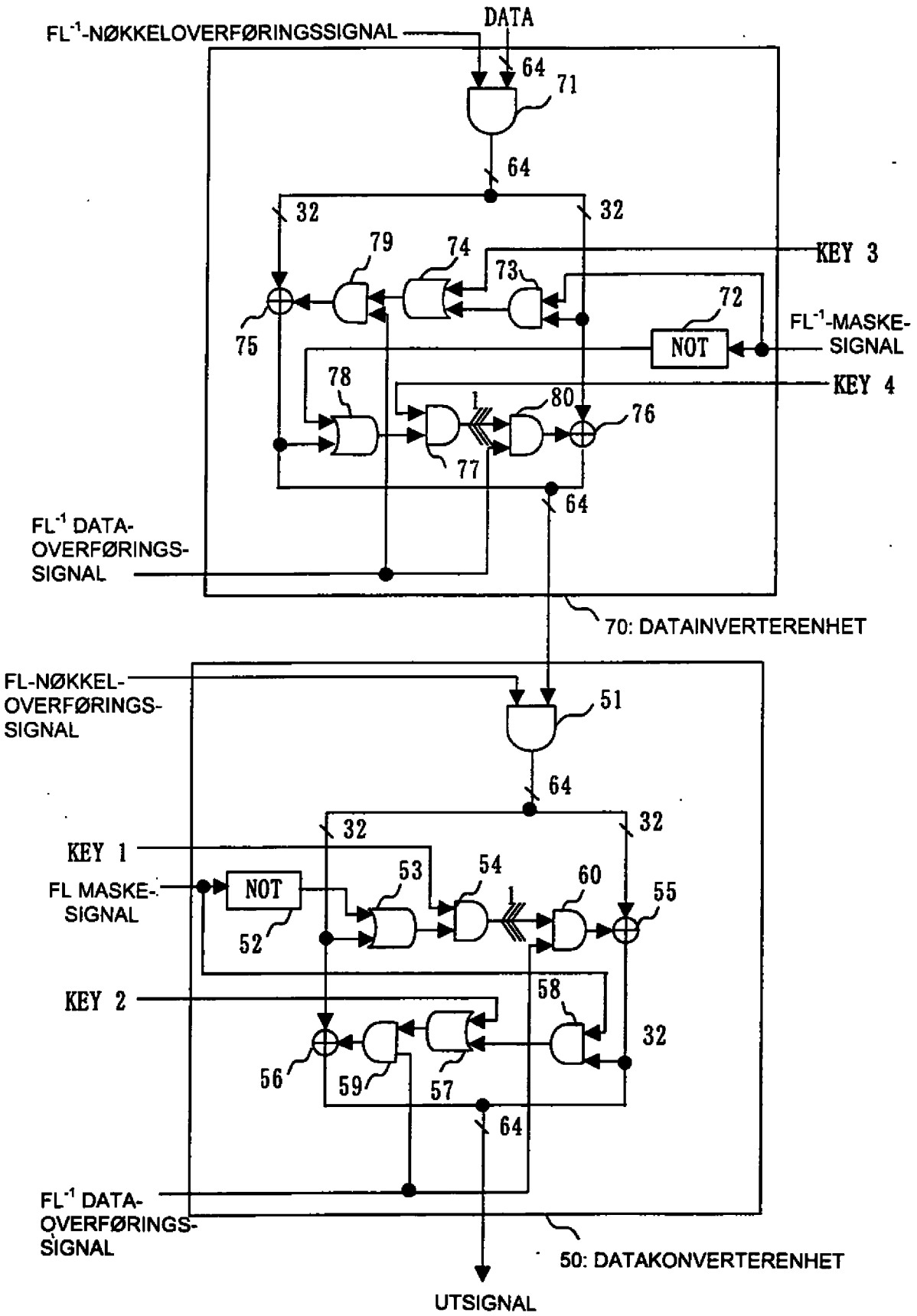
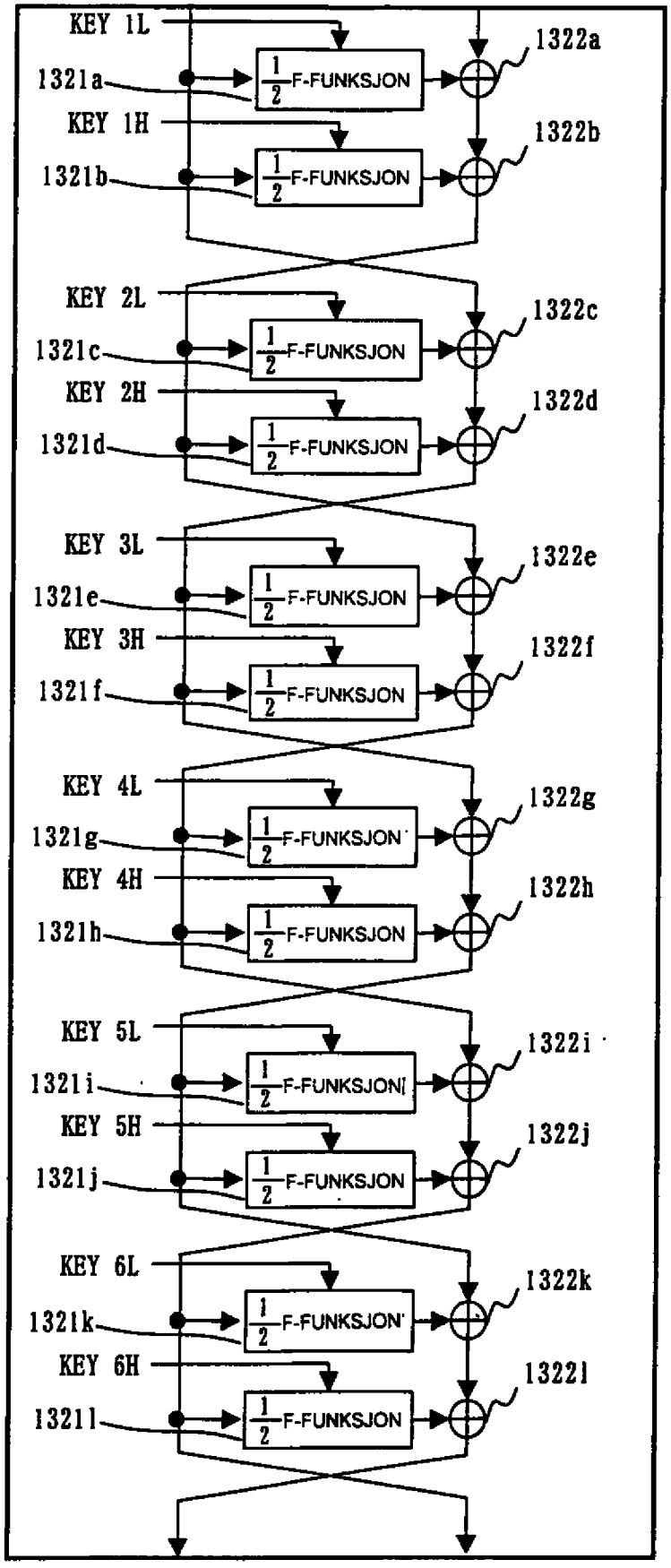

FL-1-NØKKELOVERFØRINGSSIGNAL
DATA
64
71
64
32
32
79
74
73
KEY 3
72
75
NOT
FL-1-MASKE-SIGNAL
78
80
KEY 4
76
77
64
FL-1 DATA-OVERFØRINGS-SIGNAL
70: DATAINVERTERENHET
FL-NØKKEL-OVERFØRINGS-SIGNAL
51
64
32
32
KEY 1
FL MASKE-SIGNAL
NOT
53
54
60
55
52
KEY 2
58
32
56
59
57
64
FL-1 DATA-OVERFØRINGS-SIGNAL
50: DATAKONVERTERENHET
UTSIGNAL

Fig. 64


KEY 1L
1321a
KEY 1H
1321b
1/2 F-FUNKSJON
1322a
1322b
KEY 2L
1321c
KEY 2H
1321d
1322c
1322d
KEY 3L
1321e
KEY 3H
1321f
1322e
1322f
KEY 4L
1321g
KEY 4H
1321h
1322g
1322h
KEY 5L
1321i
KEY 5H
1321j
1322i
1322j
KEY 6L
1321k
KEY 6H
1321l
1322k
1322l