



(19)  
**Bundesrepublik Deutschland**  
**Deutsches Patent- und Markenamt**

(10) **DE 102 30 942 B4 2007.09.13**

(12)

## Patentschrift

(21) Aktenzeichen: **102 30 942.6**  
 (22) Anmeldetag: **09.07.2002**  
 (43) Offenlegungstag: **05.06.2003**  
 (45) Veröffentlichungstag  
 der Patenterteilung: **13.09.2007**

(51) Int Cl.<sup>8</sup>: **H04L 27/26 (2006.01)**  
**H03M 13/29 (2006.01)**

Innerhalb von drei Monaten nach Veröffentlichung der Patenterteilung kann nach § 59 Patentgesetz gegen das Patent Einspruch erhoben werden. Der Einspruch ist schriftlich zu erklären und zu begründen. Innerhalb der Einspruchsfrist ist eine Einspruchsgebühr in Höhe von 200 Euro zu entrichten (§ 6 Patentkostengesetz in Verbindung mit der Anlage zu § 2 Abs. 2 Patentkostengesetz).

(30) Unionspriorität:  
**2001/0044673 09.07.2001 KR**  
**2001/0051605 25.08.2001 KR**  
**2001/0052596 29.08.2001 KR**

(73) Patentinhaber:  
**Samsung Electronics Co., Ltd., Suwon, Kyonggi, KR**

(74) Vertreter:  
**Grünecker, Kinkeldey, Stockmair & Schwanhäusser, 80538 München**

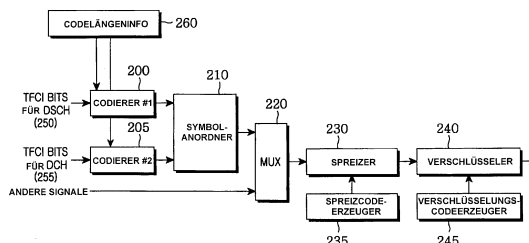
(72) Erfinder:  
**Hwang, Sung-Oh, Yongin, Kyonggi, KR; Lee, Kook-Heui, Songnam, Kyonggi, KR; Kim, Jae-Yoel, Kunpo, Kyonggi, KR; Park, Sang-Hwan, Suwon, Kyonggi, KR**

(56) Für die Beurteilung der Patentfähigkeit in Betracht gezogene Druckschriften:  
**EP 11 04 130 A2**  
**EP 10 09 174 A2**

(54) Bezeichnung: **Vorrichtung und Verfahren für die Symbolabbildung von TFCI-Bits für einen Hard Split-Modus in einem CDMA-Mobilkommunikationssystem**

(57) Hauptanspruch: Verfahren zum Abbilden von ersten kodierten TFCI (Transportformat-Kombinationsindikator)-Symbolen und zweiten kodierten TFCI-Symbolen auf einen Funkrahmen in einem Mobilkommunikationssystem zum Kodieren von  $k$  ersten TFCI-Bits und  $(10 - k)$  zweiten TFCI-Bits, wobei  $k$  eine variable Integerzahl mit einem Wert in einem Bereich von  $1 \leq k \leq 9$  ist, wobei das Verfahren die folgenden Schritte umfasst:

Multiplexen (1210) wenigstens einiger der ersten und zweiten kodierten TFCI-Symbole zum Erzeugen einer kombinierten Ausgabe mit einer durch die variable Integerzahl  $k$  beeinflussten Reihenfolge von Symbolen, und Abbilden aller oder weniger als aller der gemultiplexten, kodierten Symbole auf den Funkrahmen, um eine Anzahl der kodierten Symbole vorzusehen, die auf einen Funkrahmen abgebildet werden können.



**Beschreibung**

**[0001]** Die vorliegende Patentanmeldung beansprucht die Priorität gegenüber einer Patentanmeldung mit dem Titel „Apparatus and Method for Symbol Mapping TFCI Bits for a Hard Split Mode in a CDMA Mobile Communication System“, die am 9. Juli 2001 am koreanischen Patentamt eingereicht wurde und die Seriennummer 2001-44673 zugewiesen bekommen hat, gegenüber einer Patentanmeldung mit dem Titel „Apparatus and Method for Symbol Mapping TFCI Bits for a Hard Split Mode in a CDMA Mobile Communication System“, die am 25. August 2001 am koreanischen Patentamt eingereicht wurde und die Seriennummer 2001-51605 zugewiesen bekommen hat, und gegenüber einer Patentanmeldung mit dem Titel „Apparatus and Method for Symbol Mapping TFCI Bits for a Hard Split Mode in a CDMA Mobile Communication System“, die am 29. August 2001 am koreanischen Patentamt eingereicht wurde und die Seriennummer 2001-52596 zugewiesen bekommen hat, wobei der Inhalt aller dieser Patentanmeldungen hier unter Bezugnahme eingeschlossen ist.

**[0002]** Die vorliegende Erfindung betrifft allgemein eine Sendevorrichtung und ein entsprechendes Verfahren für einen Hard split-Modus in einem CDMA-Mobilkommunikationssystem und insbesondere eine Abbildungsvorrichtung und ein entsprechendes Verfahren für das Übertragen von TFCI (Transport Format Combination Indicator = Transportformat-Kombinationsindikator)-Bits.

**[0003]** Allgemein wird ein gemeinsamer Abwärtsverbindungs-Kanal (DSCH) durch eine Vielzahl von Benutzern auf einer zeitgeteilten Basis gemeinsam genutzt. Der DSCH wird in Assoziation mit einem dedizierten Kanal (DCH) für jeden Benutzer hergestellt. Der DCH wird über einen dedizierten physikalischen Kanal (DPCH) übertragen, und der DPCH wird gebildet, indem ein dedizierter physikalischer Steuerkanal (DPCCH) und ein dedizierter physikalischer Datenkanal (DPDCH) auf einer zeitgeteilten Basis kombiniert werden.

**[0004]** Der DSCH wird über einen gemeinsamen physikalischen Abwärtsverbindungs-Kanal (PDSCH) übertragen, und die Kanalsteuerinformation für den PDSCH wird über den DPCCH in dem DPCH übertragen. Die über den DPCCH übertragene Steuerinformation umfasst Information zu (i) einem TPC (Transmission Power Control Command = Übertragungsleistungssteuerungs-Befehl) zum Steuern der Aufwärtsverbindungs-Übertragungsleistung von einem Benutzergerät, (ii) einem Pilotfeld, das für die Kanalvariationsschätzung, die Übertragungsleistungsmessung und die Schlitzsynchronisationserfassung von einem Knoten B zu einem Benutzergerät verwendet wird, und (iii) einem TFCI (Transport Format Combination Indicator = Transportformat-Kombinationsindikator). Von dieser Information werden der TPC und das Pilotfeld als physikalische Steuerinformation für den PDSCH und den DPCH verwendet, während der TFCI verwendet wird, um Informationseigenschaften (z.B. die Informationsübertragungsrate, und die Kombination verschiedener Informationen wie z.B. die Kombination von Sprachinformation und Paketinformation) der über den DSCH und den DPDCH übertragenen Daten anzugeben.

**[0005]** Der TFCI weist als Steuerinformation, die Informationseigenschaften der über die physikalischen Kanäle DSCH und DPDCH übertragenen Daten angibt, eine Länge von 10 Bits auf und ist in 32 Bits codiert. Das heißt, die Information zu einer Datenmenge wird durch 10 Bits ausgedrückt, wobei die 10-Bit-Information in 32 Bits codiert ist, um über den physikalischen Kanal übertragen zu werden.

**[0006]** Der TFCI wird über den physikalischen Kanal in dem folgenden Verfahren übertragen, das in der technischen Spezifikation 25.212 des 3GPP (3<sup>rd</sup> Generation Partnership Projekt) für das UMTS (Universal Mobile Telecommunication System) spezifiziert ist.

$a_k$  = k-tes Informationsbit der Transportkombinationsinformation ( $0 \leq k \leq 9$ )

$b_l$  = l-tes codiertes Bit der Transportkombinationsinformation ( $0 \leq l \leq 31$ )

$d_m$  = m-tes übertragene codiertes Bit der Transportkombinationsinformation

**[0007]**  $a_k$  ist eine 10-Bit-Information, die Rate, Typ und Kombination der über den DPDCH übertragenen Daten angibt,  $b_l$  umfasst 32 codierte Bits, die durch das Codieren von  $a_k$  erhalten werden, und  $d_m$  ist ein übertragene codiertes Bit, wobei  $b_l$  über den DPCCH übertragen wird. Dabei ist der Wert m eine Variable in Übereinstimmung mit bestimmten Bedingungen.

**[0008]** Die Bedingungen für die Bestimmung der Anzahl von  $d_m$  Bits werden auf der Basis eines Übertragungsmodus des DPCCH und einer Datenrate des DPCH bestimmt. Der Übertragungsmodus des DPCCH umfasst einen normalen Übertragungsmodus und einen komprimierten Übertragungsmodus. Der komprimierte Übertragungsmodus wird verwendet, wenn ein Benutzergerät mit einem Hochfrequenz-Sendeempfänger in einem anderen Frequenzband zu messen versucht. Eine Operation in dem komprimierten Übertragungsmodus setzt die Übertragung in dem aktuellen Frequenzband vorübergehend aus, damit das Benutzergerät in ei-

nem anderen Frequenzband messen kann. Die in der ausgesetzten Periode zu übertragenden Daten werden unmittelbar vor und nach der ausgesetzten Periode komprimiert.

**[0009]** Die „Datenrate des DPCH“, die eine der Bedingungen für die Bestimmung der Anzahl von  $d_m$  Bits ist, bezieht sich auf eine physikalische Datenrate des DPCH und wird in Übereinstimmung mit einem Spreizfaktor (SF) der Daten bestimmt. Der SF liegt zwischen 4 und 512 und die Datenrate liegt zwischen 15 KBit/s und 1920 KBit/s. Wenn der SF höher ist, ist die Datenrate niedriger. Der Grund dafür, dass die Anzahl von  $d_m$  Bits in Übereinstimmung mit der Datenrate des DPCH bestimmt wird, liegt darin, dass die Größe (oder Länge) des TFCI-Felds zum Übertragen der TFCI-Bits des DPCH in Übereinstimmung mit der Datenrate des DPCH variiert.

**[0010]** Die Anzahl der für jede der Bedingungen übertragenen  $d_m$  Bits wird zur Bestimmung von  $d_m$  wie folgt berechnet:

A1. Normaler Übertragungsmodus, Datenrate des DPCH ist niedriger als 60 KBit/s

**[0011]** In einer Bedingung A1 zum Bestimmen der Anzahl von  $d_m$  Bits, ist die Anzahl der  $d_m$  Bits gleich 30. In dem 3GPP-Standard ist die Grundübertragungseinheit des physikalischen Kanals ein Funkrahmen. Der Funkrahmen weist eine Länge von 10 ms auf und umfasst 15 Zeitschlitze. Jeder Zeitschlitz weist Felder zum Übertragen des TFCI auf. Unter der Bedingung A1 weist jeder Zeitschlitz 2 TFCI-Übertragungsfelder auf, so dass die Anzahl der TFCI-Übertragungscodebits  $d_m$ , die für einen Funkrahmen übertragen werden können, gleich 30 ist. Während also die Anzahl der codierten Bits  $b_i$  auf der Basis des Informationsbits  $a_k$  gleich 32 ist, werden die letzten zwei Transportkombinations-Informationsbits  $b_{30}$  und  $b_{31}$  aufgrund einer Beschränkung der tatsächlich übertragenen Anzahl von TFCI-Feldern nicht übertragen.

A2. Normaler Übertragungsmodus, Datenrate des DPCH ist höher als 60 KBit/s

**[0012]** Unter einer Bedingung A2 zum Bestimmen der Anzahl von  $d_m$  Bits, ist die Länge des TFCI-Felds in dem Zeitschlitz gleich 8 Bits, während die Gesamtanzahl der  $d_m$ , die über den DPCH für einen Funkrahmen übertragen werden kann, gleich 120 ist. Wenn die Gesamtanzahl der  $d_m$  gleich 120 ist, wird  $b_i$  wie folgt wiederholt übertragen.

$d_0(b_0), d_{31}(b_{31}), d_{32}(b_0), \dots, d_{63}(b_{31}), \dots, d_{96}(b_0) \dots, d_{119}(b_{23})$

**[0013]** Für die Übertragung werden unter der Bedingung A2 die 0-ten bis 23-ten  $b_i$  Bits vier Mal wiederholt und werden die 24-ten bis 31-ten  $b_i$  Bits drei Mal wiederholt.

A3. Komprimierter Übertragungsmodus, Datenrate des DPCH ist niedriger als 60 KBit/s oder gleich 120 KBit/s

**[0014]** Unter einer Bedingung A3 zum Bestimmen der Anzahl von  $d_m$  Bits, ist die Länge des TFCI-Felds in dem Zeitschlitz gleich 4 Bit, während die Anzahl der TFCIs, die für einen Funkrahmen übertragen werden kann, in Übereinstimmung mit der Anzahl der Zeitschlitze variiert, die in dem komprimierten Übertragungsmodus verwendet werden. In dem komprimierten Übertragungsmodus reicht die Anzahl der ausgesetzten Zeitschlitze von mindestens 1 bis zu höchstens 7, wobei die Anzahl der  $d_m$  Bits zwischen 32 und 56 liegt. Die Anzahl der übertragenen codierten Bits  $d_m$  ist auf höchstens 32 begrenzt, um alle 0-ten bis 31-ten  $b_i$  Bits zu mit der geänderten  $d_m$  zu übertragen und nicht die  $b_i$  Bits mit der anderen  $d_m$  zu übertragen.

A4 Komprimierter Übertragungsmodus, Datenrate des DPCH ist höher als 120 KBit/s oder gleich 60 KBit/s

**[0015]** Unter einer Bedingung A4 zum Bestimmen der Anzahl von  $d_m$  Bits, ist die Länge des TFCI-Felds in dem Zeitschlitz gleich 16 Bit, wobei die Anzahl der TFCIs, die für einen Funkrahmen übertragen werden können, in Übereinstimmung mit der Anzahl von Zeitschlitzen variiert, die in dem komprimierten Übertragungsmodus verwendet werden. In dem komprimierten Übertragungsmodus, reicht die Anzahl der ausgesetzten Zeitschlitze von mindestens 1 bis höchstens 7, während die Anzahl der  $d_m$  Bits zwischen 128 und 244 liegt. Die Anzahl der übertragenen codierten Bits  $d_m$  ist auf höchstens 128 begrenzt, um wiederholt die 0-ten bis 31-ten  $b_i$  Bits vier Mal bei der geänderten  $d_m$  zu übertragen und nicht die  $b_i$  Bits bei der anderen  $d_m$  zu übertragen.

**[0016]** In dem komprimierten Übertragungsmodus der Bedingungen A3 und A4 sind die  $d_m$  Bits in einer Periode soweit entfernt wie möglich von der ausgesetzten Periode angeordnet, um die Zuverlässigkeit bei der Übertragung der  $d_m$  Bits zu maximieren.

**[0017]** Bei Bedingungen A1, A2, A3 und A4 werden verwendet, wenn der TFCI die Transportkombination und den Typ des DPCH angibt. Ein Verfahren zum Unterteilen des TFCI in einen TFCI für den DSCH und einen TFCI für den DPCH während der Übertragung kann in zwei separate Verfahren unterteilt werden.

**[0018]** Ein erstes Verfahren ist für einen Hard Split-Modus (HSM), und ein -zweites Verfahren ist für einen Logical Split-Modus (LSM).

**[0019]** Der TFCI für den DCH wird als TFCI(Feld 1) oder als erster TFCI bezeichnet, und der TFCI für den DSCH wird als TFCI(Feld 2) oder als zweiter TFCI bezeichnet.

**[0020]** In dem LSM-Verfahren sind der TFCI(Feld 1) und der TFCI(Feld 2) als ein TFCI mit einem (32,10)-Teilcode des Reed-Muller-Codes zweiter Ordnung codiert. Der TFCI(Feld 1) und der TFCI(Feld 2) drücken 10-Bit-TFCI-Information in verschiedenen Verhältnissen aus, und die 10 Informationsbits werden von der Übertragung mit einem Blockcode, d.h. einem (32,10)-Teilcode des Reed-Muller-Codes zweiter Ordnung in Übereinstimmung mit den Bedingungen A1, A2, A3 und A4 codiert. Die Verhältnisse von TFCI(Feld 1) zu TFCI(Feld 2) sind 1:9, 2:8, 3:7, 4:6, 5:5, 6:4, 7:3, 8:2 und 9:1. Die Summe der ersten TFCI-Informationsbits und der zweiten TFCI-Informationsbits kann kleiner als 10 sein. Wenn in dem LSM die Summe der ersten TFCI-Informationsbits und der zweiten TFCI-Informationsbits kleiner als 10 ist, wird die fehlende Anzahl von Bits durch Nullen aufgefüllt. Auf diese Weise können die ersten TFCI-Informationsbits und die zweiten TFCI-Informationsbits mit einem (32,10)-Reed-Muller-Code codiert werden, bevor sie übertragen werden.

**[0021]** In dem HSM-Verfahren werden der TFCI(Feld 1) und der TFCI(Feld 2) jeweils fest mit 5 Bits ausgedrückt, wobei jede Information unter Verwendung eines (16,5)-Biorthogonalcodes ausgegeben wird, wobei dann die 16 Bits für den TFCI(Feld 1) und den TFCI(Feld 2) alternierend in Übereinstimmung mit den Bedingungen A1, A2, A3 und A4 übertragen werden. Wenn die maximale Anzahl der ersten TFCI-Informationsbits und die maximale Anzahl der zweiten TFCI-Informationsbits beide auf 5 begrenzt sind und wenn die Anzahl der ersten TFCI-Informationsbits oder der zweiten TFCI-Informationsbits 5 überschreitet, kann das HSM-Verfahren nicht verwendet werden. Wenn also die Anzahl der ersten TFCI-Informationsbits oder der zweiten TFCI-Informationsbits kleiner als 5 ist, werden die leeren Bits durch Nullen aufgefüllt, bevor die Codierung unter Verwendung eines (16,5)-Biorthogonalcodes durchgeführt wird.

**[0022]** [Fig. 1](#) zeigt einen Aufbau eines Senders auf der Basis der herkömmlichen HSM-Verfahrens. Wie in [Fig. 1](#) gezeigt codiert ein (16,5)-Biorthogonal-Codierer **100** einen 5-Bit-TFCI(Feld 1) für den DCH zu 16 codierten Symbolen und gibt die 16 codierten Symbole an einen Multiplexer **110** aus. Gleichzeitig codiert ein (16,5)-Biorthogonal-Codierer **105** einen 5-Bit-TFCI(Feld 2) für den DSCH zu 16 codierten Symbolen und gibt die 16 codierte Symbole an den Multiplexer **110** aus. Der Multiplexer **110** zeitmultiplext dann die 16 codierten Symbole aus dem Codierer **100** und die 16 codierten Symbole aus dem Codierer **105** und gibt die 32 Symbole nach der Anordnung aus. Ein Multiplexer **120** zeitmultiplext die 32 aus dem Multiplexer **110** ausgegebenen Symbole mit anderen Signalen und gibt seine Ausgabe an einen Spreizer **130** aus. Der Spreizer **130** spreizt das Ausgabesignal des Multiplexers **120** mit einem Spreizcode aus einem Spreizcodeerzeuger **135**. Ein Verschlüsseler **140** verschlüsselt das gespreizte Signal mit einem Verschlüsselungscode aus einem Verschlüsselungscodeerzeuger **145**.

**[0023]** Wenn ein Benutzergerät in einem wichen Übergabebereich lokalisiert ist, unterliegt das LSM-Verfahren aus den folgenden Gründen vielen Beschränkungen. Der einfacheren Darstellung halber wird im Folgenden eine kurze Beschreibung eines 3GPP-Funkübertragungsnetzes gegeben. Ein RAN (Radio Access Network = Funkzugriffsnetz) umfasst einen RNC (Radio Network Controller = Funknetz-Controller), einen durch den RNC kontrollierten Knoten B und ein Benutzergerät. Der RNC steuert den Knoten B, der als Basisstation dient, während das Benutzergerät als Endgerät dient. Der RNC kann in einen SRNC (Serving Radio Network Controller = Dienstfunknetzwerk-Controller) und einen CRNC (Control Radio Network Controller = Steuerfunknetzwerk-Controller) in Übereinstimmung mit den Beziehungen zu dem Benutzergerät unterteilt sein. Der SRNC ist ein RNC, bei dem das Benutzergerät registriert ist, wobei er Daten für das Senden zu und das Empfangen von dem Benutzergerät verarbeitet und das Benutzergerät steuert. Der CRNC ist ein RNC, mit dem das Benutzergerät derzeit verbunden ist, wobei er das Benutzergerät mit dem SRNC verbindet.

**[0024]** Wenn die Knoten B, die mit dem Benutzergerät kommunizieren, zu verschiedenen RNCs gehören, können die Knoten B, die keinen DSCH übertragen, den Wert der codierten TFCI-Bits für den DSCH nicht erkennen, so dass die codierten TFCI-Bits nicht korrekt an das Benutzergerät übertragen werden können.

**[0025]** In dem oben genannten HSM werden die TFCI-Informationsbits für den DSCH und die TFCI-Inforna-

tionsbits für den DCH unabhängig voneinander codiert, so dass das Benutzergerät keine Schwierigkeiten hat, die empfangenen TFCI-Bits zu decodieren. In dem gegenwärtigen 3GPP-HSM sind jedoch die Anzahl der TFCI-Bits für den DCH und die Anzahl der TFCI-Bits für den DSCH beide auf 5 Bits festgelegt, um 32 Informationsbits auszudrücken. Wenn also mehr TFCI-Bits für den DCH oder den DSCH benötigt werden, kann das HSM nicht verwendet werden.

**[0026]** Die EP 1 009 174 A2 bezieht sich auf eine Einrichtung und Verfahren zur Realisierung von TFCI-Mehrdiensten in einem mobilen Kommunikationssystem. Um ein mobiles Kommunikationssystem mit mehreren unterschiedlichen Diensten auszustatten, wird ein Transportformatkombinationsindikator (TFCI) mit einem Transportformatindikator (TFI) verwendet. Der TFCI wird auf einen zugeordneten physikalischen Datenkanal (DPDCH) geladen und übertragen. Die Einrichtung und das Verfahren zur Realisierung des TFCI Mehrdienstes in einem mobilen Kommunikationssystem fordert den Mehrdienst durch Laden des TFI, der eine äußere Kodierung, äußere Verschachtelung, innere Kodierung, innere Verschachtelung und ein Ratenangleichsystem, welches für einen Mehrdienst notwendig ist, sind in dem Transportformatindikator (TFI) gespeichert. Dieser TFI wird über einen zugeordneten Kanal mit den folgenden Schritten übertragen: Multiplexen des Transportformatindikatoren (TFI) basierend auf dem Mehrdienst und Durchführen einer Ratenanpassung; Verschachteln und Umwandeln des Ratenangleichsignals in einen kodierten zusammengesetzten Transportkanal und Demultiplexen/Aufspalten des umgewandelten Signals und Laden des Signals auf einen oder eine Vielzahl von zugeordneten physikalischen Datenkanälen entsprechend einer Rate des umgewandelten Signals und Übertragen des TFCI, wobei dadurch mehr unterschiedliche Dienste durch Realisierung eines TFCI bereitgestellt werden, wobei der TFCI einen Transportformatindikator (TFI) aufweist, der auf den zugeordneten physikalischen Steuerkanal geladen und übertragen wird und von dem Netzwerk erkannt wird.

**[0027]** Die EP 1 104 130 A2 bezieht sich auf ein Verfahren zum Kodieren und Übertragen eines Transportformatkombinationsindicators. Ein Verfahren und Matrizen zum Übertragen eines Transportformatkombinationsindicators wird bereitgestellt. Ein Verfahren zum Übertragen eines TFCI in einem mobilen Kommunikationssystem umfasst einen Schritt des Erkennens einer Anzahl von eingegebenen TFCI Bits, Kodieren mit einer konventionellen Matrix, und Übertragen der kodierten Bits über den Sendekanal, wobei die konventionelle Matrix 5 Spaltenvektoren von 32 Elementen von binärem Code, der von einem OVSF-Code hergeleitet ist. Der OVSF-Code wird mit den niederwertigeren Bits des TFCI multipliziert, ein Spaltenvektor der 32 Elemente weisen den Wert 1 auf, und 4 Spaltenvektoren der 32 Elemente des binären Codes werden von einem Maskencode hergeleitet, wobei der Maskencode mit den höherwertigeren Bits des TFCI multipliziert werden soll.

**[0028]** Es ist die Aufgabe der vorliegenden Erfindung, eine verbesserte Vorrichtung und ein verbessertes Verfahren zum Senden/Empfangen von TFCI-Bits in einem CDMA-Mobilkommunikationssystem bereitzustellen.

**[0029]** Diese Aufgabe ist durch den Gegenstand der unabhängigen Patentansprüche gelöst.

**[0030]** Bevorzugte Ausführungsformen sind in den abhängigen Patentansprüchen definiert.

**[0031]** Es ist ein Aspekt der vorliegenden Erfindung, eine Vorrichtung und ein Verfahren zum Abbilden von codierten TFCI-Symbolen auf einen physikalischen Kanal in einem CDMA-Mobilkommunikationssystem anzugeben.

**[0032]** Es ist ein weiterer Aspekt der vorliegenden Erfindung, eine Vorrichtung und ein Verfahren zum Abbilden der codierten TFCI-Symbole für den DCH und der codierten TFCI-Symbole für den DSCH, die in einem bestimmten Verhältnis aufgeteilt sind, für einen physikalischen Kanal in einem CDMA-Mobilkommunikationssystem anzugeben.

**[0033]** Es ist ein weiterer Aspekt der vorliegenden Erfindung, eine Vorrichtung und ein Verfahren zum Empfangen von codierten TFCI-Symbolen, die vor der Übertragung auf einen physikalischen Kanal abgebildet werden, in einem CDMA-Mobilkommunikationssystem anzugeben.

**[0034]** Es ist ein weiterer Aspekt der vorliegenden Erfindung, eine Vorrichtung und ein Verfahren zum Empfangen von codierten TFCI-Symbolen für den DCH und von codierten TFCI-Symbolen für den DSCH, die in einem bestimmten Verhältnis aufgeteilt sind und vor der Übertragung auf einen physikalischen Kanal abgebildet sind, in einem CDMA-Mobilkommunikationssystem anzugeben.

**[0035]** Gemäß einem ersten Aspekt der vorliegenden Erfindung ist ein Verfahren zum Abbilden von ersten codierten TFCI-Symbolen und zweiten codierten TFCI-Symbolen auf einen Funkrahmen in einer Sendevor-

richtung eines Mobilkommunikationssystems zum Codieren von  $k$  ersten TFCI-Bits und  $(10 - k)$  zweiten TFCI-Bits angegeben, wobei die Summe der ersten codierten TFCI-Symbole und der zweiten codierten TFCI-Symbole gleich 32 ist. Das Verfahren umfasst das Multiplexen der codierten Symbole, so dass die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole gleichmäßig in Übereinstimmung mit einem Übertragungsmodus und einer Datenrate des Funkrahmens verteilt sind, das Ausgeben von 32 codierten Symbolen und das Abbilden der 32 gemultiplexten codierten Symbole auf den Funkrahmen, um die Anzahl der codierten Symbole vorzusehen, die auf einen Funkrahmen abgebildet werden kann und in Übereinstimmung mit dem Übertragungsmodus und der Datenrate des Funkrahmens bestimmt wird.

**[0036]** Gemäß einem zweiten Aspekt der vorliegenden Erfindung ist eine Vorrichtung zum Übertragen von ersten TFCI-Bits und zweiten TFCI-Bits über einen Funkrahmen in einer Sendevorrichtung eines Mobilkommunikationssystems angegeben. Die Vorrichtung umfasst wenigstens einen Codierer zum Codieren von  $k$  ersten TFCI-Bits mit einer ersten Codiertrate, um  $(3k + 1)$  erste codierte TFCI-Symbole auszugeben, und zum Codieren von  $(10 - k)$  zweiten TFCI-Bits mit einer zweiten Codiertrate, um  $(31 - 3k)$  zweite codierte TFCI-Symbole auszugeben, sowie einen Codiertsymbol-Anordner zum Multiplexen der codierten Symbole, so dass die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole gleichmäßig in Übereinstimmung mit einem Übertragungsmodus und einer Datenrate des Funkrahmens verteilt werden, und zum Ausgeben der gemultiplexten codierten Symbole in Übereinstimmung mit der Anzahl der codierten Symbole, die über einen Funkrahmen übertragen werden können.

**[0037]** Gemäß einem dritten Aspekt der vorliegenden Erfindung ist ein Verfahren zum Übertragen von ersten TFCI-Bits und zweiten TFCI-Bits über einen Funkrahmen in einer Sendevorrichtung eines Mobilkommunikationssystems angegeben. Das Verfahren umfasst das Codieren von  $k$  ersten TFCI-Bits mit einer ersten Codiertrate, um  $(3k + 1)$  erste codierte TFCI-Symbole auszugeben, das Codieren von  $(10 - k)$  zweiten TFCI-Bits mit einer zweiten Codiertrate, um  $(31 - 3k)$  zweite codierte TFCI-Symbole auszugeben, und einen Codiertsymbol-Anordner zum Multiplexen der codierten Symbole, so dass die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole gleichmäßig in Übereinstimmung mit einem Übertragungsmodus und einer Datenrate des Funkrahmens verteilt werden, und das Ausgeben der gemultiplexten codierten Symbole in Übereinstimmung mit der Anzahl der codierten Symbole, die über einen Funkrahmen übertragen werden können.

**[0038]** Gemäß einem vierten Aspekt der vorliegenden Erfindung ist eine Vorrichtung zum Decodieren von  $k$  ersten TFCI-Bits und  $(10 - k)$  zweiten TFCI-Bits in einer Empfangsvorrichtung eines Mobilkommunikationssystems zum Empfangen von  $(3k + 1)$  ersten codierten TFCI-Symbolen für einen DCH (Dedicated Channel = Dedizierter Kanal) und von  $(31 - 3k)$  zweiten codierten TFCI-Symbolen für einen DSCH (Downlink Shared Channel = Gemeinsamer Abwärtsverbindungskanal) angegeben. Die Vorrichtung umfasst einen Codiertsymbol-Anordner zum Trennen der ersten codierten TFCI-Symbole und der zweiten codierten TFCI-Symbole, die über einen DPCH (Dedicated Physical Channel = Dedizierter physikalischer Kanal) übertragen werden, in Übereinstimmung mit dem Wert von  $k$  für die Neuordnung, und wenigstens einen Decodierer zum Decodieren der ersten codierten TFCI-Symbole, um die  $k$  ersten TFCI-Bits auszugeben, sowie zum Decodieren der zweiten codierten TFCI-Symbole, um die  $(10 - k)$  zweiten TFCI-Bits auszugeben.

**[0039]** Gemäß einem fünften Aspekt der vorliegenden Erfindung ist ein Verfahren zum Decodieren von  $k$  ersten TFCI-Bits und  $(10 - k)$  zweiten TFCI-Bits in einer Empfangsvorrichtung eines Mobilkommunikationssystems zum Empfangen von  $(3k + 1)$  ersten codierten TFCI-Symbolen für einen DCH (Dedicated Channel = dedizierter Kanal) und  $(31 - 3k)$  zweiten codierten TFCI-Symbolen für einen DSCH (Downlink Shared Channel = gemeinsamer Abwärtsverbindungskanal) angegeben. Das Verfahren umfasst das Trennen der ersten codierten TFCI-Symbole und der zweiten codierten TFCI-Symbole, die über einen DPCH (Dedicated Physical Channel = dedizierten physikalischen Kanal) übertragen werden, in Übereinstimmung mit einem Wert von  $k$  für die Neuordnung, das Decodieren der ersten codierten TFCI-Symbole, um die  $k$  ersten TFCI-Bits auszugeben, und zum Decodieren der zweiten codierten TFCI-Symbole, um die  $(10 - k)$  zweiten TFCI-Bits auszugeben.

**[0040]** Vorstehende und andere Aufgaben, Merkmale und Vorteile der vorliegenden Erfindung werden durch die folgende detaillierte Beschreibung mit Bezug auf die beigefügten Zeichnungen verdeutlicht:

**[0041]** [Fig. 1](#) zeigt dem Aufbau eines herkömmlichen Senders auf der Basis eines Hard Split-Modus (HSM),

**[0042]** [Fig. 2](#) zeigt den Aufbau eines Knoten-B-Senders gemäß einer Ausführungsform der vorliegenden Erfindung,

- [0043] [Fig. 3](#) zeigt einen anderen Aufbau eines Knoten-B-Senders gemäß einer Ausführungsform der vorliegenden Erfindung,
- [0044] [Fig. 4](#) zeigt den ausführlichen Aufbau des in [Fig. 2](#) und [Fig. 3](#) gezeigten Codierers,
- [0045] [Fig. 5](#) zeigt den Aufbau eines Abwärtsverbindungs-Funkrahmens, der von einem Knoten B zu einem Benutzergerät übertragen wird,
- [0046] [Fig. 6](#) zeigt den ausführlichen Aufbau des in [Fig. 2](#) gezeigten Symbol-Anordners,
- [0047] [Fig. 7](#) zeigt den ausführlichen Aufbau des in [Fig. 3](#) gezeigten Selektors,
- [0048] [Fig. 8](#) zeigt einen anderen ausführlichen Aufbau des in [Fig. 3](#) gezeigten Symbol-Anordners,
- [0049] [Fig. 9](#) zeigt den Aufbau eines Benutzergerät-Empfängers gemäß einer Ausführungsform der vorliegenden Erfindung,
- [0050] [Fig. 10](#) zeigt einen anderen Aufbau eines Benutzergerät-Empfängers gemäß einer anderen Ausführungsform der vorliegenden Erfindung,
- [0051] [Fig. 11](#) zeigt den ausführlichen Aufbau des Decodierers, der in dem in [Fig. 10](#) gezeigten Empfänger verwendet wird,
- [0052] [Fig. 12](#) zeigt ein Verfahren zum Auswählen von Codes, die für den ersten TFCI und zweiten TFCI gemäß einer Ausführungsform der vorliegenden Erfindung zu verwenden sind,
- [0053] [Fig. 13](#) zeigt eine andere Verbindung zwischen den Codierern und einem Symbol-Anordner gemäß einer Ausführungsform der vorliegenden Erfindung,
- [0054] [Fig. 14](#) zeigt eine weitere Verbindung zwischen einem Codierer und einem Symbol-Anordner gemäß einer Ausführungsform der vorliegenden Erfindung,
- [0055] [Fig. 15](#) zeigt eine weitere Verbindung zwischen einem Codierer und einem Symbol-Anordner gemäß einer Ausführungsform der vorliegenden Erfindung,
- [0056] [Fig. 16](#) zeigt eine Codieroperation gemäß einer Ausführungsform der vorliegenden Erfindung,
- [0057] [Fig. 17](#) zeigt eine Decodieroperation gemäß einer Ausführungsform der vorliegenden Erfindung.
- [0058] [Fig. 18A](#) und [Fig. 18B](#) zeigen zwei unterschiedliche Aufbauten eines Symbol-Anordners gemäß einer Ausführungsform der vorliegenden Erfindung, und
- [0059] [Fig. 19](#) zeigt einen Aufbau eines Codiertsymbol-Anordners gemäß einer Ausführungsform der vorliegenden Erfindung.
- [0060] Im Folgenden wird eine bevorzugte Ausführungsform der vorliegenden Erfindung mit Bezug auf die beigefügten Zeichnungen beschrieben. In der folgenden Beschreibung werden wohlbekannte Funktionen oder Aufbauten nicht im Detail beschrieben, da dadurch die Erfindung durch unnötige Einzelheiten verundeutlicht werden würde.
- [0061] Die vorliegende Erfindung gibt eine Vorrichtung und ein Verfahren zum Unterteilen von insgesamt 10 Eingangsinformationsbits für den DCH und Informationsbits für den DSCH in einem Verhältnis von 11:9, 2:8, 3:7, 4:6, 5:5, 6:4, 7:3, 8:2 und 9:1 in dem HSM-Verfahren sowie zum folgenden separaten Codieren der Informationsbits für den DCH und der Informationsbits für den DSCH an. Wenn die Summe der ersten TFCI-Informationsbits und der zweiten TFCI-Informationsbits kleiner als 10 ist, erhöhen die Vorrichtung und das Verfahren gemäß einer Ausführungsform der vorliegenden Erfindung die Zuverlässigkeit der ersten TFCI-Informationsbits oder der zweiten TFCI-Informationsbits vor der Codierung. Alternativ hierzu erhöhen die Vorrichtung und das Verfahren die Zuverlässigkeit von sowohl den ersten TFCI-Informationsbits als auch den zweiten TFCI-Informationsbits vor der Codierung.

**[0062]** Zuerst wird ein Codierer beschrieben, wenn die Summe der ersten TFCI-Informationsbits und der zweiten TFCI-Informationsbits gleich 10 ist.

**[0063]** Ein Funkrahmen überträgt 30, 120, 32 und 128 codierte TFCI-Symbole jeweils in Übereinstimmung mit den Bedingungen A1, A2, A3 und A4. In jedem Fall ist ohne eine wiederholte Übertragung eine Grundcodier-rate gleich  $10/32$ , wobei die Codierrate bei der Bedingung A1 aufgrund der begrenzten Übertragung des physikalischen Kanals zu  $10/20$  wird. Wenn also die TFCI-Informationsbits für den DSCH und die TFCI-Informationbits für den DCH in einem bestimmten Verhältnis von 1:9, 2:8, 3:7, 4:6, 5:5, 6:4, 7:3, 8:2 oder 9:1 unterteilt werden, kann die Codierrate natürlich aufrechterhalten werden, indem die codierten Symbole in den oben genannten Verhältnissen aufgeteilt werden. Unter Aufrechterhalten der Codierrate ist zu verstehen, dass die Grundcodier-rate von  $(32,10)$  aufrechterhalten wird. In dem HSM muss die Codeverstärkung der verschiedenen codierten TFCI für den DSCH und TFCI für den DCH aufrechterhalten werden, um eine Codeverstärkung durch das entsprechende Aufrechterhalten der Codierrate  $(32,10)$  aufrechtzuerhalten, obwohl der TFCI für den DSCH und der TFCI für den DCH separat codiert werden. Ein Beispiel für die Aufteilung der codierten Bits in Übereinstimmung mit dem Verhältnis der Eingangsbits wird unter der Annahme der Bedingung A1 beschrieben.

**[0064]** Wenn unter der Bedingung A1 10 Eingangsinformationsbits in einem Verhältnis von 1:9 aufgeteilt werden, dann werden 30 codierte Ausgangssymbole in einem Verhältnis von 3:27 aufgeteilt, und wenn die 10 Eingangsinformationsbits in einem Verhältnis von 2:8 aufgeteilt werden, dann werden die 30 codierten Ausgangssymbole in einem Verhältnis von 6:24 aufgeteilt. Wenn weiterhin die 10 Eingangsinformationsbits in einem Verhältnis von 3:7 aufgeteilt werden, dann werden die 30 codierten Ausgangssymbole in einem Verhältnis von 9:21 aufgeteilt, und wenn die 10 Eingangsinformationsbits in einem Verhältnis von 4:6 aufgeteilt werden, dann werden die 30 codierten Ausgangssymbole in einem Verhältnis von 12:18 aufgeteilt. Unter den Bedingungen A2, A3 und A4 jedoch werden die 32 codierten Symbole alle übertragen, oder die 32 codierten Symbole werden wiederholt übertragen, so dass die codierten Symbole nicht wie unter der Bedingung A1 korrekt aufgeteilt werden können.

**[0065]** Deshalb können in der Ausführungsform der vorliegenden Erfindung die in Verbindung mit den Eingangsbits definierten Codierraten der codierten Symbole wie in der Tabelle 1 gezeigt ausgedrückt werden.

Tabelle 1

Verhältnis der Eingangsbits	Verhältnis der codierten Symbole	Verwendete Codiertrate	
		Codiertrate des ersten TCFI	Codiertrate des zweiten TCFI
1:9	3:29	(3:1)	(29:9)
	4:28	(4:1)	(28:9)
	5:27	(5:1)	(27:9)
2:8	6:26	(6:2)	(26:8)
	7:25	(7:2)	(25:8)
	8:24	(8:2)	(24:8)
3:7	9:23	(9:3)	(23:7)
	10:22	(10:3)	(22:7)
	11:21	(11:3)	(21:7)
4:6	12:20	(12:4)	(20:6)
	13:19	(13:4)	(19:6)
	14:18	(14:4)	(18:6)
6:4	18:14	(18:6)	(14:4)
	19:13	(19:6)	(13:4)
	20:12	(20:6)	(12:4)
7:3	21:11	(21:7)	(11:3)
	22:10	(22:7)	(10:3)
	23:9	(23:7)	(9:3)
8:2	24:8	(24:8)	(8:2)
	25:7	(25:8)	(7:2)
	26:6	(26:8)	(6:2)
9:1	27:5	(27:9)	(5:1)
	28:4	(28:9)	(4:1)
	29:3	(29:9)	(3:1)

**[0066]** Im Folgenden wird ein Kriterium zum Bestimmen der Codiertraten von Tabelle 1 in Übereinstimmung mit dem Verhältnis der Eingangsbits beschrieben. Die Ausführungsform der vorliegenden Erfindung setzt die Summe der codierten Symbole auf 30, indem sie den minimal erforderlichen Wert auf die wesentliche Codiertrate (30,10) für den am häufigsten verwendeten Fall A1 der Bedingungen A1, A2, A3 und A4 setzt und die Codiertrate des ersten TCFI und die Codiertrate des zweiten TCFI auf ein Minimum von 1/3 setzt, wobei sie dann die verbleibenden zwei codierten Symbole jeweils auf das codierte Symbol des ersten TCFI und das codierte Symbol des zweiten TCFI setzt. Dadurch erhöht die Ausführungsform der vorliegenden Erfindung sowohl die Codiertrate des ersten TCFI als auch die Codiertrate des zweiten TCFI, oder sie erhöht entweder die Codiertrate des ersten TCFI oder die Codiertrate des zweiten TCFI, indem Sie die verbleibenden zwei codierten Symbole

als codierte Symbole des ersten TFCI oder als codierte Symbole des zweiten TFCI verwendet. Die Ausführungsform erhöht die Codierrate des ersten TFCI oder des zweiten TFCI aus den Kriterien zum Bestimmen der Codierraten, wenn die Leistung erhöht werden muss, indem sie nur die Codierrate des ersten TFCI oder die Codierrate des zweiten TFCI unter der Bedingung erhöht, dass die Summe der Anzahl der codierten Symbole für den ersten TFCI und der Anzahl der codierten Symbole für den zweiten TFCI gleich 32 werden sollte.

**[0067]** Sobald das Verhältnis der Eingangsbits in Tabelle 1 bestimmt wird, wird eines von drei Codierv Verfahren in Übereinstimmung mit dem Verhältnis der codierten Symbole verwendet.

**[0068]** Die vorliegende Erfindung gibt einen Codierer an, der eine Codierung mit allen in Tabelle 1 angegebenen Codierraten durchführen kann. Wenn mit Bezug auf die Tabelle 1 das Verhältnis der Eingangsbits (bzw. das Verhältnis der Informationsmengen, d.h. das Verhältnis der ersten TFCI-Bits zu den zweiten TFCI-Bits) gleich 1:9 ist, wird das Verhältnis der codierten Symbole zu 3:29, 4:28 oder 5:27. Wenn das Verhältnis der Eingangsbits gleich 2:8 ist, wird das Verhältnis der codierten Symbole zu 6:26, 7:25 oder 8:24, und wenn das Verhältnis der Eingangsbits 3:7 ist, wird das Verhältnis der codierten Symbole zu 9:23, 10:22 oder 11:21. Wenn das Verhältnis der Eingangsbits 4:6 ist, wird das Verhältnis der codierten Symbole zu 12:20, 13:19 oder 14:18. Wenn das Verhältnis der Eingangsbits 6:4 ist, wird das Verhältnis der codierten Symbole zu 18:14, 19:13 oder 20:12, und wenn das Verhältnis der Eingangsbits 7:3 ist, wird das Verhältnis der codierten Symbole zu 21:11, 22:10 oder 23:9. Wenn das Verhältnis der Eingangsbits 8:2 ist, wird das Verhältnis der codierten Symbole zu 24:8, 25:7 oder 26:6, und wenn das Verhältnis der Eingangsbits 9:1 ist, wird das Verhältnis der codierten Symbole zu 27:5, 28:4 oder 29:3.

**[0069]** Wenn also das Verhältnis der Eingangsbits 1:9 ist, dann sind {(3,1)-Codierer, (29,9)-Codierer, (4,1)-Codierer und (28,9) Codierer} oder {(5,1)-Codierer und (27,9)-Codierer} erforderlich. Wenn das Verhältnis der Eingangsbits 2:8 ist, dann sind {(6,2)-Codierer, (26,8)Codierer, (7,2)-Codierer und (25,8)Codierer} oder {(8,2)Codierer und (24,8)Codierer} erforderlich. Wenn das Verhältnis der Eingangsbits gleich 3:7 ist, dann sind {(9,3)-Codierer, (23,7)-Codierer, (10,3)-Codierer und (22,7)-Codierer} oder {(11,3)-Codierer und (21,7)-Codierer} erforderlich. Wenn das Verhältnis der Eingangsbits 4:6 ist, dann sind {(12,4)-Codierer, (20,6)-Codierer, (13,4)Codierer und (19,6)-Codierer} oder {(14,4)-Codierer und (18,6)-Codierer} erforderlich. Wenn man also die 24 Codierer und den derzeit verwendeten (16,5)-Codierer und den (32,10)-Codierer betrachtet, besteht also ein Bedarf für einen Codierer, der die Rolle der 18 Codierer in einem einzigen Aufbau erfüllt, um die Leistung zu erhöhen und die Komplexität der Hardware zu reduzieren.

**[0070]** Allgemein kann eine Hamming-Distanzverteilung für Codewörter der Fehlerkorrekturcodes als Maß dienen, das die Leistung von linearen Fehlerkorrekturcodes angibt. Die "Hamming-Distanz" ist die Anzahl der Nicht-Null-Symbole in einem Codewort. Das heißt, für ein bestimmtes Codewort '0111' ist die Anzahl der Einsen in dem Codewort gleich 3, so dass die Hamming-Distanz gleich 3 ist. Der kleinste Wert der Hamming-Distanzwerte wird als „minimale Distanz  $d_{\min}$ “ bezeichnet, und eine Erhöhung der minimalen Distanz des Codeworts verbessert die Fehlerkorrekturleistung der Fehlerkorrekturcodes. Mit anderen Worten ist der „optimale Code“ ein Code mit der optimalen Fehlerkorrekturleistung. Dies ist im Detail in einem Artikel mit dem Titel „The Theory of Error-Correcting Codes“ von F.J. Macwilliams bei N.J.A. Sloane, North-Holland beschrieben.

**[0071]** Um außerdem nur einen Codiereraufbau für die Codierer mit unterschiedlichen Längen vorzusehen und dadurch die Hardware-Komplexität zu reduzieren, wird vorzugsweise der Code mit der längsten Länge, d.h. der (32,10) Code gekürzt. Für die Kürzung müssen die codierten Symbole ausgesondert werden. Bei der Aussonderung variiert jedoch die minimale Distanz des Codes in Übereinstimmung mit den Aussonderungspositionen. Deshalb werden vorzugsweise die Aussonderungspositionen derart berechnet, dass der ausgesonderte Code die minimale Distanz aufweist.

**[0072]** Für die minimale Distanz wird zum Beispiel vorzugsweise ein optimaler (7,2) Code mit einer der Codierraten von Tabelle 1 verwendet, der erhalten wird, indem ein (3,2)-Einfachcode drei Mal wiederholt wird und dann die letzten zwei codierten Symbole ausgesondert werden. Die Tabelle 2 zeigt die Beziehung zwischen den Eingangsinformationsbits des (3,2)-Einfachcodes und den (3,2)-Einfachcodewörtern, die auf der Basis der Eingangsinformationsbits ausgegeben werden.

Tabelle 2

Eingangsinformationsbits	(3,2)-Einfachcodewörter
00	000
01	101
10	011
11	110

**[0073]** Die Tabelle 3 zeigt die Beziehung zwischen den Eingangsinformationsbits und den (7,2)-Einfachcodewörtern, die erhalten werden, indem das (3,2)-Einfachcodewort drei Mal wiederholt wird und dann die letzten zwei codierten Symbole ausgesondert werden.

Tabelle 3

Eingangsinformationsbits	(7,2)-Einfachcodewörter
00	000 000 0
01	101 101 1
10	011 011 0
11	110 110 1

**[0074]** Die durch das dreifache Wiederholen der (3,2)-Einfachcodewörter und das folgende Aussondern der letzten zwei codierten Symbole erhaltenen (7,2)-Einfachcodewörter können implementiert werden, indem der bestehende (16,4)-Reed-Muller-Code gekürzt wird.

**[0075]** Eine Beschreibung des Kürzungsverfahrens wird im Folgenden anhand eines Beispiels gegeben. Der (16,4)-Reed-Muller-Code ist eine lineare Kombination aus vier Grundcodewörtern mit der Länge 16, wobei '4' die Anzahl der Eingangsinformationsbits ist. Das Empfangen von nur zwei Bits der 16 Eingangsinformationsbits entspricht der Verwendung einer linearen Kombination von nur zwei Grundcodewörtern aus den vier Grundcodewörtern mit der Länge 16 und der Nichtverwendung der verbleibenden Codewörter. Indem außerdem die Verwendung der Grundcodewörter eingeschränkt wird und dann neun Symbole aus den 16 Symbolen ausgesondert werden, kann ein (7,2)-Codierer unter Verwendung des (16,4)-Codierers realisiert werden. Die Tabelle 4 stellt das Kürzungsverfahren dar.

Tabelle 4

Eingangs- infor- mations- bits	Codewörter															
0000	0(*)	0	0	0	0(*)	0	0	0	0(*)	0	0(*)	0(*)	0(*)	0(*)	0(*)	0(*)
0001	0(*)	1	0	1	0(*)	1	0	1	0(*)	1	0(*)	1(*)	0(*)	1(*)	0(*)	1(*)

<b>0010</b>	<b>0(*)</b>	<b>0</b>	<b>1</b>	<b>1</b>	<b>0(*)</b>	<b>0</b>	<b>1</b>	<b>1</b>	<b>0(*)</b>	<b>0</b>	<b>1(*)</b>	<b>1(*)</b>	<b>0(*)</b>	<b>0(*)</b>	<b>1(*)</b>	<b>1(*)</b>
0011	0(*)	1	1	0	0(*)	1	1	0	0(*)	1	1(*)	0(*)	0(*)	1(*)	1(*)	0(*)
<b>0100</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>
0101	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0
0110	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0
0111	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1
<b>1000</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>
1001	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0
1010	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0
1011	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1
1100	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0
1101	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1
1110	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1
1111	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0

**[0076]** Wie in [Fig. 4](#) gezeigt ist jedes (16,4)-Codierwort eine lineare Kombination aus den vier fetten Grundcodewörtern mit der Länge 16. Um den (6,2)-Code zu erhalten, werden nur die oberen zwei Codewörter aus den vier Grundcodewörtern verwendet. Dann werden die verbleibenden zwölf unteren Codewörter automatisch nicht verwendet. Es werden also nur die oberen vier Codewörter verwendet. Um übrigens ein Grundcodewort mit der Länge 7 unter den vier Grundcodewörtern zu erzeugen, müssen die neun Symbole ausgesondert werden. Es ist möglich, die (7,2)-Einfachcodewörter der Tabelle 3 zu erhalten, indem die in der Tabelle 4 durch (\*) angegebenen Symbole ausgeondert werden und dann die verbleibenden sieben codierten Symbole gesammelt werden.

**[0077]** Im Folgenden wird der Aufbau eines Codierers zum Erstellen von {(3,1)-Optimalcode, (29,9)-Optimalcode, (4,1)-Optimalcode und (28,9)-Optimalcode} und {(5,1)-Optimalcode und (27,9)-Optimalcode} für die Verwendung für das Informationsbitverhältnis 1:9, der Aufbau eines Codierers zum Erstellen von {(6,2)-Optimalcode, (26,8)-Optimalcode, (7,2)-Optimalcode und (25,8)-Optimalcode} und {(8,2)-Optimalcode und (24,8)-Optimalcode} für die Verwendung für das Informationsbitverhältnis von 2:8, der Aufbau eines Codierers zum Erstellen von {(9,3)-Optimalcode, (23,7)-Optimalcode, (10,3)-Optimalcode und (22,7)-Optimalcode} und {(11,3)-Optimalcode und (21,7)-Optimalcode} für die Verwendung für das Informationsbitverhältnis von 3:7, der Aufbau eines Codierers zum Erstellen von {(12,4)-Optimalcode, (20,6)-Optimalcode, (13,4)-Optimalcode und (19,6)-Optimalcode} und {(14,4)-Optimalcode und (18,6)-Optimalcode} für die Verwendung für das Informationsbitverhältnis von 4:6 und der Aufbau eines Codierers zum Erstellen eines (16,5)-Optimalcodes und eines (32,10)-Optimalcodes für die Verwendung für das Informationsbitsverhältnis von 5:5 durch das Kürzen eines (32,10)-Teilcodes des Reed-Muller-Codes der zweiten Ordnung beschrieben. Außerdem wird im Folgenden der Aufbau eines Decodierers in Entsprechung zu dem Codierer beschrieben.

1. Erste Ausführungsform des Senders

**[0078]** Eine Ausführungsform der vorliegenden Erfindung umfasst eine Vorrichtung und ein Verfahren zum Aufteilen von 10 Informationsbits in einem Verhältnis von 1:9, 2:8, 3:7, 4:6, 5:5, 6:4, 7:3, 8:2 und 9:1 vor dem Codieren in dem Hard split-Modus, wie es im Logical split-Modus, wenn das Verhältnis der Eingangsinformationsbits 5:5 ist.

**[0079]** [Fig. 2](#) stellt den Aufbau eines Senders gemäß einer Ausführungsform der vorliegenden Erfindung dar. Wie in [Fig. 2](#) gezeigt, werden die TFCI-Bits für den DSCH und die TFCI-Bits für den DCH, die mit einem der oben genannten Informationsbitverhältnisse aufgeteilt sind, jeweils zu dem ersten und dem zweiten Codierer **200** und **205** gegeben. Dabei werden die TFCI-Bits für den DSCH als TFCI(Feld 1) oder erste TFCI-Bits bezeichnet, während die TFCI-Bits für den DCH als TFCI(Feld 2) oder zweite TFCI-Bits bezeichnet werden. Die

TFCI-Bits für den DSCH werden in einem ersten TFCI-Biterzeuger **250** erzeugt, und die TFCI-Bits für den DCH werden in einem zweiten TFCI-Biterzeuger **255** erzeugt. Die Anzahl der ersten TFCI-Bits unterscheidet sich von der Anzahl der zweiten TFCI-Bits in Übereinstimmung mit den oben genannten Informationsbitverhältnissen. Außerdem wird ein Steuersignal, das Codelängeninformation, d.h. Information zu einem Längenwert des in Übereinstimmung mit dem Informationsbitverhältnis gesetzten Codeworts angibt, zu dem ersten und dem zweiten Codierer **200** und **205** gegeben. Die Codelängeninformation wird in einem Codelängeninformationserzeuger **260** erzeugt und weist einen Wert auf, der in Übereinstimmung mit den Längen der ersten TFCI-Bits und der zweiten TFCI-Bits variiert.

**[0080]** Wenn das Informationsbitverhältnis 6:4 ist, empfängt der Codierer **200** ein Längensteuersignal, damit der Codierer **200** bei Empfang von 6 TFCI-Bits für den DSCH als (20,6)-Codierer, (19,6)-Codierer oder (18,6)-Codierer dienen kann, während der Codierer **205** ein Längensteuersignal empfängt, damit der Codierer **205** bei Empfang von 4 TFCI-Bits für den DCH als (12,4)-Codierer, (13,4)-Codierer oder (14,4)-Codierer dienen kann. Wenn das Informationsbitverhältnis 7:3 ist, empfängt der Codierer **200** ein Längensteuersignal, damit der Codierer **200** bei Empfang von 7 TFCI-Bits für den DSCH als (23,7)-Codierer, (22,7)-Codierer oder (21,7)-Codierer dienen kann, während der Codierer **205** ein Längensteuersignal empfängt, damit der Codierer **205** bei Empfang von 3 TFCI-Bits für den DCH als (9,3)-Codierer, (10,3)-Codierer oder (11,3)-Codierer dienen kann. Wenn das Informationsbitverhältnis 8:2 ist, empfängt der Codierer **200** ein Längensteuersignal, damit der Codierer **200** bei Empfang von 8 TFCI-Bits für den DSCH als (26,8)-Codierer, (25,8)-Codierer oder (24,8)-Codierer dienen kann, während der Codierer **205** ein Längensteuersignal empfängt, damit der Codierer **205** bei Empfang von 2 TFCI-Bits für den DCH als (6,2)-Codierer, (7,2)-Codierer oder (8,2)-Codierer dienen kann. Wenn das Informationsbitverhältnis 9:1 ist, empfängt der Codierer **200** ein Längensteuersignal, damit der Codierer **200** bei Empfang von 9 TFCI-Bits für den DSCH als (29,9)-Codierer, (28,9)-Codierer oder (27,9)-Codierer dienen kann, während der Codierer **205** ein Längensteuersignal empfängt, damit der Codierer **205** bei Empfang von 1 TFCI-Bit für den DCH als (3,1)-Codierer, (4,1)-Codierer oder (5,1)-Codierer dienen kann. Das Längensteuersignal sollte derart erzeugt werden, dass die Summe der ersten TFCI-Bits und der zweiten TFCI-Bits gleich 32 wird. Das heißt, wenn der erste TFCI-Codierer ein (4,1)-Codierer ist, sollte der zweite TFCI-Codierer ein (28,9)-Codierer und nicht ein (29,9)- oder ein (27,9)-Codierer sein. Wenn der zweite TFCI-Codierer der (29,9)-Codierer ist, wird die Anzahl der codierten Bits  $b_1$  zu 33, und wenn der zweite TFCI-Codierer der (27,9)-Codierer ist, wird die Anzahl der codierten Bits  $b_1$  zu 31. In diesem Fall ist der Sender nicht mit dem herkömmlichen Sender kompatibel, der zwei (16,5)-Codierer oder einen (32,10)-Codierer verwendet. Außerdem ist der Sender nicht mit dem herkömmlichen Sender kompatibel, da die  $b_1$  Bits auf die  $d_m$  Bits abgebildet werden.

**[0081]** [Fig. 4](#) stellt einen detaillierten Aufbau der Codierer **200** und **205** dar. Das heißt, der Codierer **200** zum Codieren des ersten TFCI-Codierers und der Codierer **205** zum Codieren des zweiten TFCI-Codierers weisen den Aufbau von [Fig. 4](#) auf. Wenn jedoch die ersten TFCI-Codewörter und die zweiten TFCI-Codewörter mit einer Zeitverzögerung erzeugt werden, können der erste TFCI-Codierer und der zweite TFCI-Codierer durch einen einzigen Codierer realisiert werden. Ein Aufbau eines Senders zum Erzeugen der ersten TFCI-Codewörter und der zweiten TFCI-Codewörter mit einer Zeitverzögerung ist in [Fig. 3](#) dargestellt.

**[0082]** Zuerst wird mit Bezug auf [Fig. 2](#) eine detaillierte Beschreibung eines Codierers in Übereinstimmung mit der vorliegenden Erfindung für den Fall gegeben, dass das Verhältnis der ersten TFCI-Bits zu den zweiten TFCI-Bits gleich 1:9 ist.

**[0083]** Wenn das Informationsbitverhältnis 1:9 ist, dient der Codierer **200** als ein (3,1)-Codierer und dient der Codierer **205** als ein (22,9)-Codierer, dient der Codierer **200** als ein (4,1)-Codierer und dient der Codierer **205** als ein (28,9)-Codierer, oder dient der Codierer **200** als ein (5,1)-Codierer und dient der Codierer **205** als ein (27,9)-Codierer.

**[0084]** Im Folgenden werden die Operationen des (3,1)-Codierers, des (29,9)-Codierers, des (4,1)-Codierers, des (28,9)-Codierers, des (5,1)-Codierers und des (27,9)-Codierers im Detail mit Bezug auf [Fig. 4](#) beschrieben.

**[0085]** Zuerst wird eine Operation des (3,1)-Codierers beschrieben. Wie in [Fig. 4](#) gezeigt, wird ein Eingangsbit  $a_0$  normal zu dem Codierer gegeben, wobei die verbleibenden Eingangsbits  $a_1$ ,  $a_2$ ,  $a_3$ ,  $a_4$ ,  $a_5$ ,  $a_6$ ,  $a_7$ ,  $a_8$  und  $a_9$  mit Nullen aufgefüllt werden. Das Eingangsbit  $a_0$  wird zu einem Multiplizierer **410** gegeben, das Eingangsbit  $a_1$  zu einem Multiplizierer **412**, das Eingangsbit  $a_2$  zu einem Multiplizierer **414**, das Eingangsbit  $a_3$  zu einem Multiplizierer **416**, das Eingangsbit  $a_4$  zu einem Multiplizierer **418**, das Eingangsbit  $a_5$  zu einem Multiplizierer **420**, das Eingangsbit  $a_6$  zu einem Multiplizierer **422**, das Eingangsbit  $a_7$  zu einem Multiplizierer **424**,

das Eingangsbit a8 zu einem Multiplizierer **426** und das Eingangsbit a9 zu einem Multiplizierer **428**. Gleichzeitig erzeugt ein Walsh-Code-Erzeuger **400** ein Grundcodewort  $W1 = 101010101010101101010101010100$  und gibt das erzeugte Grundcodewort  $W1$  an den Multiplizierer **410** aus. Der Multiplizierer **410** multipliziert dann das Eingangsbit a0 mit dem Grundcodewort  $W1$  in einer Symboleinheit und gibt seine Ausgabe an einen Exklusiv-ODER-(XOR)-Operator **440** aus. Weiterhin erzeugt der Walsh-Code-Erzeuger **400** andere Grundcodewörter  $W2, W4, W8$  und  $W16$  und gibt diese jeweils an die Multiplizierer **412, 414, 416** und **418** aus. Ein Nur-Einsen-Codeerzeuger **402** erzeugt ein Nur-Einsen-Grundcodewort (bzw. eine Nur-Einsen-Sequenz) und gibt das erzeugte Nur-Einsen-Grundcodewort an den Multiplizierer **420** aus. Ein Maskenerzeuger **404** erzeugt Grundcodewörter  $M1, M2, M4$  und  $M8$  und gibt die erzeugten Grundcodewörter  $M1, M2, M4$  und  $M8$  jeweils an die Multiplizierer **422, 424, 426** und **428** aus. Weil jedoch die Eingangsbits a1, a2, a3, a4, a5, a6, a7, a8 und a9, die an den Multiplizierern **412, 414, 416, 418, 420, 422, 424, 426** und **428** angelegt werden, alle nur aus Nullen bestehen, geben die Multiplizierer **412, 414, 416, 418, 420, 422, 424, 426** und **428** Nullen an den Exklusiv-ODER-Operator **440** aus, so dass die Ausgabe des Exklusiv-ODER-Operators **440** nicht beeinflusst wird. Das heißt, ein durch die Exklusiv-ODER-Verknüpfung der Ausgabewerte der Multiplizierer **410, 412, 414, 416, 418, 420, 422, 424, 426** und **428** durch den Exklusiv-ODER-Operator **440** bestimmter Wert ist gleich dem Ausgabewert des Multiplizierers **410**. Die aus dem Exklusiv-ODER-Operator **440** ausgegebenen 32 Symbole werden zu einem Aussonderer **460** gegeben. Zu diesem Zeitpunkt empfängt eine Steuereinrichtung **450** Codelängeninformation und gibt an den Aussonderer **460** ein Steuersignal aus, das die Aussonderungspositionen auf der Basis der Codelängeninformation angibt. Der Aussonderer **460** sondert dann das 1-te, 3-te, 5-te, 6-te, 7-te, 8-te, 9-te, 10-te, 11-te, 12-te, 13-te, 14-te, 15-te, 16-te, 17-te, 18-te, 19-te, 20-te, 21-te, 22-te, 23-te, 24-te, 25-te, 26-te, 27-te, 28-te, 29-te, 30-te und 31-te codierte Symbol aus den insgesamt 32 codierten Symbolen (0-tes bis 31-tes Symbol) in Übereinstimmung mit dem Steuersignal aus der Steuereinrichtung **450** aus. Mit anderen Worten sondert der Aussonderer **460** 29 Symbole aus den 32 codierten Symbolen aus und gibt also drei nicht-ausgesonderte codierte Symbole aus.

[0086] Zweitens wird die Operation des (29,9)-Codierers beschrieben. Wie in [Fig. 4](#) gezeigt, werden normalerweise neun Eingangsbits a0, a1, a2, a3, a4, a5, a6, a7 und a8 an den Codierer ausgegeben, während das verbleibende Eingangsbit a9 mit einer Null aufgefüllt wird. Das Eingangsbit a0 wird an dem Multiplizierer **410** angelegt, das Eingangsbit a1 an dem Multiplizierer **412**, das Eingangsbit a2 an dem Multiplizierer **414**, das Eingangsbit a3 an dem Multiplizierer **416**, das Eingangsbit a4 an dem Multiplizierer **418**, das Eingangsbit a5 an dem Multiplizierer **420**, das Eingangsbit a6 an dem Multiplizierer **422**, das Eingangsbit a7 an dem Multiplizierer **424**, das Eingangsbit a8 an dem Multiplizierer **426** und das Eingangsbit a9 an dem Multiplizierer **428**. Gleichzeitig gibt der Walsh-Code-Erzeuger **400** an den Multiplizierer **410** das Grundcodewort  $W1 = 101010101010101101010101010100$  aus, an den Multiplizierer **412** das Grundcodewort  $W2 = 01100110011001100110011001100$ , an den Multiplizierer **414** das Grundcodewort  $W4 = 000111100001111000011110000111100$ , an den Multiplizierer **416** das Grundcodewort  $W8 = 000000011111110000000111111100$  und an den Multiplizierer **418** das Grundcodewort  $W16 = 00000000000000011111111111101$ . Dann multipliziert der Multiplizierer **410** das Grundcodewort  $W1$  mit dem Eingangsbit a0 in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus, multipliziert der Multiplizierer **412** das Grundcodewort  $W2$  mit dem Eingangsbit a1 in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus, multipliziert der Multiplizierer **414** das Grundcodewort  $W4$  mit dem Eingangsbit a2 in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus, multipliziert der Multiplizierer **416** das Grundcodewort  $W8$  mit dem Eingangsbit a3 in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus und multipliziert der Multiplizierer **418** das Grundcodewort  $W16$  mit dem Eingangsbit a4 in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus. Außerdem erzeugt der Nur-Einsen-Codeerzeuger **402** ein Nur-Einsen-Grundcodewort mit der Länge 32 und gibt das erzeugte Nur-Einsen-Grundcodewort an den Multiplizierer **420** aus. Der Multiplizierer **420** multipliziert dann das Nur-Einsen-Grundcodewort mit dem Eingangsbit a5 in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus. Der Maskenerzeuger **404** gibt an den Multiplizierer **422** das Grundcodewort  $M1 = 0101\ 0000\ 1100\ 0111\ 1100\ 0001\ 1101\ 1101$  aus, an den Multiplizierer **424** das Grundcodewort  $M2 = 0000\ 0011\ 1001\ 0100\ 1011\ 0111\ 0001\ 1100$  und an den Multiplizierer **426** das Grundcodewort  $M4 = 0001\ 0101\ 1111\ 0010\ 0110\ 1100\ 1010\ 1100$ . Dann multipliziert der Multiplizierer **422** das Grundcodewort  $M1$  mit dem Eingangsbit a6 in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus, multipliziert der Multiplizierer **424** das Grundcodewort  $M2$  mit dem Eingangsbit a7 in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus und multipliziert der Multiplizierer **426** das Grundcodewort  $M4$  mit dem Eingangsbit a8 in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus. Weiterhin erzeugt der Maskenerzeuger **404** das andere Grundcodewort  $M8$  und gibt das erzeugte Grundcodewort  $M8$  an den Multiplizierer **428** aus. Weil jedoch das an den Multiplizierer **428** angelegte Eingangsbit a9 eine Null ist, gibt der Multiplizierer **428** eine Null an den Exklusiv-ODER-Operator **440** aus, so dass die Ausgabe des Exklusiv-ODER-Operators **440** nicht beeinflusst

wird. Das heißt, ein durch die Exklusiv-ODER-Verknüpfung der Ausgabewerte aus den Multiplizierern **410**, **412**, **414**, **416**, **418**, **420**, **422**, **424**, **426** und **428** durch den Exklusiv-ODER-Operator **440** ist gleich einem durch die Exklusiv-ODER-Verknüpfung der Ausgabewerte der Multiplizierer **410**, **412**, **414**, **416**, **418**, **420**, **422**, **424** und **426** bestimmten Wert. Die 32 aus dem Exklusiv-ODER-Operator **440** ausgegebenen Symbole werden an den Aussonderer **460** gegeben. Zu diesem Zeitpunkt empfängt die Steuereinrichtung **450** Codelängeninformation und gibt an den Aussonderer **460** ein Steuersignal aus, das die Aussonderungspositionen auf der Basis der Codelängeninformation angibt. Der Aussonderer **460** sondert dann das 6-te, 10-te und 11-te codierte Symbol aus den insgesamt 32 codierten Symbolen (0-tes bis 31-tes Symbol) in Übereinstimmung mit dem Steuersignal aus der Steuereinrichtung **450** aus. Mit anderen Worten sondert der Aussonderer **460** drei Symbole aus den 32 codierten Symbolen aus und gibt dann 29 nicht-ausgesonderte codierte Symbole aus.

**[0087]** Drittens wird die Operation des (4,1)-Codierers beschrieben. Wie in [Fig. 4](#) gezeigt, wird normalerweise ein Eingangsbit a0 an den Codierer ausgegeben, während die verbleibenden Eingangsbits a1, a2, a3, a4, a5, a6, a7, a8 und a9 mit einer Null aufgefüllt werden. Das Eingangsbit a0 wird an dem Multiplizierer **410** angelegt, das Eingangsbit a1 an dem Multiplizierer **412**, das Eingangsbit a2 an dem Multiplizierer **414**, das Eingangsbit a3 an dem Multiplizierer **416**, das Eingangsbit a4 an dem Multiplizierer **418**, das Eingangsbit a5 an dem Multiplizierer **420**, das Eingangsbit a6 an dem Multiplizierer **422**, das Eingangsbit a7 an dem Multiplizierer **424**, das Eingangsbit a8 an dem Multiplizierer **426** und das Eingangsbit a9 an dem Multiplizierer **428**. Gleichzeitig erzeugt der Walsh-Code-Erzeuger **400** ein Grundcodewort  $W1 = 101010101010101101010101010100$  und gibt das erzeugte Grundcodewort  $W1$  an den Multiplizierer **410** aus. Der Multiplizierer **410** multipliziert dann das Eingangsbit a0 mit dem Grundcodewort  $W1$  in einer Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus. Weiterhin erzeugt der Walsh-Code-Erzeuger **400** die anderen Grundcodewörter  $W2$ ,  $W4$ ,  $W8$  und  $W16$  und gibt diese jeweils an die Multiplizierer **412**, **416** und **418** aus. Der Nur-Einsen-Codiererzeuger **402** erzeugt ein Nur-Einsen-Grundcodewort (bzw. eine Nur-Einsen-Sequenz) und gibt das erzeugte Nur-Einsen-Grundcodewort an den Multiplizierer **420** aus. Der Maskenerzeuger **404** erzeugt die Grundcodewörter  $M1$ ,  $M2$ ,  $M4$  und  $M8$  und gibt die erzeugten Grundcodewörter  $M1$ ,  $M2$ ,  $M4$  und  $M8$  jeweils an die Multiplizierer **422**, **424**, **426** und **428** aus. Weil jedoch die Eingangsbits a1, a2, a3, a4, a5, a6, a7, a8 und a9, die an den Multiplizierern **412**, **414**, **416**, **418**, **420**, **422**, **424**, **426** und **428** angelegt werden, alle Nullen sind, geben die Multiplizierer **412**, **414**, **416**, **418**, **420**, **422**, **424**, **426** und **428** Nullen an den Exklusiv-ODER-Operator **440** aus, so dass die Ausgabe des Exklusiv-ODER-Operators **440** nicht beeinflusst wird. Das heißt, ein durch die Exklusiv-ODER-Verknüpfung der Ausgabewerte der Multiplizierer **410**, **412**, **414**, **416**, **418**, **420**, **422**, **424**, **426** und **428** durch den Exklusiv-ODER-Operator **440** bestimmter Wert ist gleich dem Ausgabewert des Multiplizierers **410**. Die aus dem Exklusiv-ODER-Operator **440** ausgegebenen 32 Symbole werden zu einem Aussonderer **460** gegeben. Zu diesem Zeitpunkt empfängt eine Steuereinrichtung **450** Codelängeninformation und gibt an den Aussonderer **460** ein Steuersignal aus, das die Aussonderungspositionen auf der Basis der Codelängeninformation angibt. Der Aussonderer **460** sondert dann das 1-te, 3-te, 5-te, 7-te, 8-te, 9-te, 10-te, 11-te, 12-te, 13-te, 14-te, 15-te, 16-te, 17-te, 18-te, 19-te, 20-te, 21-te, 22-te, 23-te, 24-te, 25-te, 26-te, 27-te, 28-te, 29-te, 30-te und 31-te codierte Symbol aus den insgesamt 32 codierten Symbolen (0-tes bis 31-tes Symbol) in Übereinstimmung mit dem Steuersignal aus der Steuereinrichtung **450** aus. Mit anderen Worten sondert der Aussonderer **460** 28 Symbole aus den 32 codierten Symbolen aus und gibt also vier nicht-ausgesonderte codierte Symbole aus.

**[0088]** Viertens wird die Operation des (28,9)-Codierers beschrieben. Wie in [Fig. 4](#) gezeigt, werden normalerweise neun Eingangsbits a0, a1, a2, a3, a4, a5, a6, a7 und a8 an den Codierer ausgegeben, während das verbleibende Eingangsbit a9 mit einer Null aufgefüllt wird. Das Eingangsbit a0 wird an dem Multiplizierer **410** angelegt, das Eingangsbit a1 an dem Multiplizierer **412**, das Eingangsbit a2 an dem Multiplizierer **414**, das Eingangsbit a3 an dem Multiplizierer **416**, das Eingangsbit a4 an dem Multiplizierer **418**, das Eingangsbit a5 an dem Multiplizierer **420**, das Eingangsbit a6 an dem Multiplizierer **422**, das Eingangsbit a7 an dem Multiplizierer **424**, das Eingangsbit a8 an dem Multiplizierer **426** und das Eingangsbit a9 an dem Multiplizierer **428**. Gleichzeitig gibt der Walsh-Code-Erzeuger **400** an den Multiplizierer **410** das Grundcodewort  $W1 = 101010101010101101010101010100$  aus, an den Multiplizierer **412** das Grundcodewort  $W2 = 011001100110011001100110011001100$ , an den Multiplizierer **414** das Grundcodewort  $W4 = 000111100001111000011110000111100$ , an den Multiplizierer **416** das Grundcodewort  $W8 = 0000000111111100000001111111100$  und an den Multiplizierer **418** das Grundcodewort  $W16 = 000000000000000111111111111101$ . Dann multipliziert der Multiplizierer **410** das Grundcodewort  $W1$  mit dem Eingangsbit a0 in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus, multipliziert der Multiplizierer **412** das Grundcodewort  $W2$  mit dem Eingangsbit a1 in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus, multipliziert der Multiplizierer **414** das Grundcodewort  $W4$  mit dem Eingangsbit a2 in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus, multipliziert der Multiplizierer **416** das Grundcodewort  $W8$  mit dem Eingangsbit a3 in der Sym-

boleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus und multipliziert der Multiplizierer **418** das Grundcodewort W16 mit dem Eingangsbit a4 in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus. Außerdem erzeugt der Nur-Einsen-Codeerzeuger **402** ein Nur-Einsen-Grundcodewort mit der Länge 32 und gibt das erzeugte Nur-Einsen-Grundcodewort an den Multiplizierer **420** aus. Der Multiplizierer **420** multipliziert dann das Nur-Einsen-Grundcodewort mit dem Eingangsbit a5 in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus. Der Maskenerzeuger **404** gibt an den Multiplizierer **422** das Grundcodewort M1 = 0101 0000 1100 0111 1100 0001 1101 1101 aus, an den Multiplizierer **424** das Grundcodewort M2 = 0000 0011 1001 0100 1011 0111 00001 1100 und an den Multiplizierer **426** das Grundcodewort M4 = 0001 0101 1111 0010 0110 1100 1010 1100. Dann multipliziert der Multiplizierer **422** das Grundcodewort M1 mit dem Eingangsbit a6 in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus, multipliziert der Multiplizierer **424** das Grundcodewort M2 mit dem Eingangsbit a7 in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus und multipliziert der Multiplizierer **426** das Grundcodewort M4 mit dem Eingangsbit a8 in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus. Weiterhin erzeugt der Maskenerzeuger **404** das andere Grundcodewort M8 und gibt das erzeugte Grundcodewort M8 an den Multiplizierer **428** aus. Weil jedoch das an den Multiplizierer **428** angelegte Eingangsbit a9 eine Null ist, gibt der Multiplizierer **428** eine Null an den Exklusiv-ODER-Operator **440** aus, so dass die Ausgabe des Exklusiv-ODER-Operators **440** nicht beeinflusst wird. Das heißt, ein durch die Exklusiv-ODER-Verknüpfung der Ausgabewerte aus den Multiplizierern **410**, **412**, **414**, **416**, **418**, **420**, **422**, **424**, **426** und **428** durch den Exklusiv-ODER-Operator **440** ist gleich einem durch die Exklusiv-ODER-Verknüpfung der Ausgabewerte der Multiplizierer **410**, **412**, **414**, **416**, **418**, **420**, **422**, **424** und **426** bestimmten Wert. Die 32 aus dem Exklusiv-ODER-Operator **440** ausgegebenen Symbole werden an den Aussonderer **460** gegeben. Zu diesem Zeitpunkt empfängt die Steuereinrichtung **450** Codelängeninformation und gibt an den Aussonderer **460** ein Steuersignal aus, das die Aussonderungspositionen auf der Basis der Codelängeninformation angibt. Der Aussonderer **460** sondert dann das 6-te, 10-te, 11-te und 30-te codierte Symbol aus den insgesamt 32 codierten Symbolen (0-tes bis 31-tes Symbol) in Übereinstimmung mit dem Steuersignal aus der Steuereinrichtung **450** aus. Mit anderen Worten sondert der Aussonderer **460** vier Symbole aus den 32 codierten Symbolen aus und gibt dann 28 nicht-ausgesonderte codierte Symbole aus.

[0089] Fünftens wird die Operation des (5,1)-Codierers beschrieben. Wie in [Fig. 4](#) gezeigt, wird normalerweise ein Eingangsbit a0 an den Codierer ausgegeben, während die verbleibenden Eingangsbits a1, a2, a3, a4, a5, a6, a7, a8 und a9 mit einer Null aufgefüllt werden. Das Eingangsbit a0 wird an dem Multiplizierer **410** angelegt, das Eingangsbit a1 an dem Multiplizierer **412**, das Eingangsbit a2 an dem Multiplizierer **414**, das Eingangsbit a3 an dem Multiplizierer **416**, das Eingangsbit a4 an dem Multiplizierer **418**, das Eingangsbit a5 an dem Multiplizierer **420**, das Eingangsbit a6 an dem Multiplizierer **422**, das Eingangsbit a7 an dem Multiplizierer **424**, das Eingangsbit a8 an dem Multiplizierer **426** und das Eingangsbit a9 an dem Multiplizierer **428**. Gleichzeitig erzeugt der Walsh-Code-Erzeuger **400** ein Grundcodewort W1 = 101010101010101101010101010100 und gibt das erzeugte Grundcodewort W1 an den Multiplizierer **410** aus. Der Multiplizierer **410** multipliziert dann das Grundcodewort W1 mit dem Eingangsbit a0 in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus. Weiterhin erzeugt der Walsh-Code-Erzeuger **400** die anderen Grundcodewörter W2, W4, W8 und W16 und gibt die erzeugten Grundcodewörter W2, "4, W8 und W16 jeweils an die Multiplizierer **412**, **416** und **418** aus. Der Nur-Einsen-Codeerzeuger **402** erzeugt ein Nur-Einsen-Grundcodewort mit der Länge 32 und gibt das erzeugte Nur-Einsen-Grundcodewort an den Multiplizierer **420** aus. Weiterhin erzeugt der Maskenerzeuger **404** die Grundcodewörter M1, M2, M4 und M8 und gibt die erzeugten Grundcodewörter M1, M2, M4 und M8 jeweils an die Multiplizierer **422**, **424**, **426** und **428** aus. Weil jedoch die Eingangsbits a1, a2, a3, a4, a5, a6, a7, a8 und a9, die an den Multiplizierern **412**, **414**, **416**, **418**, **420**, **422**, **424**, **426** und **428** angelegt werden, alle Nullen sind, geben die Multiplizierer **412**, **414**, **416**, **418**, **420**, **422**, **424**, **426** und **428** Nullen an den Exklusiv-ODER-Operator **440** aus, so dass die Ausgabe des Exklusiv-ODER-Operators **440** nicht beeinflusst wird. Das heißt, ein durch die Exklusiv-ODER-Verknüpfung der Ausgabewerte der Multiplizierer **410**, **412**, **414**, **416**, **418**, **420**, **422**, **424**, **426** und **428** durch den Exklusiv-ODER-Operator **440** bestimmter Wert ist gleich dem Ausgabewert des Multiplizierers **410**. Die aus dem Exklusiv-ODER-Operator **440** ausgegebenen 32 Symbole werden zu einem Aussonderer **460** gegeben. Zu diesem Zeitpunkt empfängt die Steuereinrichtung **450** Codelängeninformation und gibt an den Aussonderer **460** ein Steuersignal aus, das die Aussonderungspositionen auf der Basis der Codelängeninformation angibt. Der Aussonderer **460** sondert dann das 1-te, 3-te, 5-te, 7-te, 9-te, 10-te, 11-te, 12-te, 13-te, 14-te, 15-te, 16-te, 17-te, 18-te, 19-te, 20-te, 21-te, 22-te, 23-te, 24-te, 25-te, 26-te, 27-te, 28-te, 29-te, 30-te und 31-te codierte Symbol aus den insgesamt 32 codierten Symbolen (0-tes bis 31-tes Symbol) in Übereinstimmung mit dem Steuersignal aus der Steuereinrichtung **450** aus. Mit anderen Worten sondert der Aussonderer **460** 27 Symbole aus den 32 codierten Symbolen aus und gibt also fünf nicht-ausgesonderte codierte Symbole aus.

[0090] Sechstens wird die Operation des (27,9)-Codierers beschrieben. Wie in [Fig. 4](#) gezeigt, werden norma-

lerweise neun Eingangsbits  $a_0, a_1, a_2, a_3, a_4, a_5, a_6, a_7$  und  $a_8$  an den Codierer ausgegeben, während das verbleibende Eingangsbit  $a_9$  mit einer Null aufgefüllt wird. Das Eingangsbit  $a_0$  wird an dem Multiplizierer **410** angelegt, das Eingangsbit  $a_1$  an dem Multiplizierer **412**, das Eingangsbit  $a_2$  an dem Multiplizierer **414**, das Eingangsbit  $a_3$  an dem Multiplizierer **416**, das Eingangsbit  $a_4$  an dem Multiplizierer **418**, das Eingangsbit  $a_5$  an dem Multiplizierer **420**, das Eingangsbit  $a_6$  an dem Multiplizierer **422**, das Eingangsbit  $a_7$  an dem Multiplizierer **424**, das Eingangsbit  $a_8$  an dem Multiplizierer **426** und das Eingangsbit  $a_9$  an dem Multiplizierer **428**. Gleichzeitig gibt der Walsh-Code-Erzeuger **400** an den Multiplizierer **410** das Grundcodewort  $W_1 = 1010101010101010101010101010100$  aus, an den Multiplizierer **412** das Grundcodewort  $W_2 = 01100110011001101100110011001100$ , an den Multiplizierer **414** das Grundcodewort  $W_4 = 000111100001111000011110000111100$ , an den Multiplizierer **416** das Grundcodewort  $W_8 = 000000011111110000001111111100$  und an den Multiplizierer **418** das Grundcodewort  $W_{16} = 000000000000000111111111111101$ . Dann multipliziert der Multiplizierer **410** das Grundcodewort  $W_1$  mit dem Eingangsbit  $a_0$  in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus, multipliziert der Multiplizierer **412** das Grundcodewort  $W_2$  mit dem Eingangsbit  $a_1$  in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus, multipliziert der Multiplizierer **414** das Grundcodewort  $W_4$  mit dem Eingangsbit  $a_2$  in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus, multipliziert der Multiplizierer **416** das Grundcodewort  $W_8$  mit dem Eingangsbit  $a_3$  in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus und multipliziert der Multiplizierer **418** das Grundcodewort  $W_{16}$  mit dem Eingangsbit  $a_4$  in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus. Der Nur-Einsen-Codeerzeuger **402** erzeugt ein Nur-Einsen-Grundcodewort mit der Länge 32 und gibt das erzeugte Nur-Einsen-Grundcodewort an den Multiplizierer **420** aus. Der Multiplizierer **420** multipliziert dann das Nur-Einsen-Grundcodewort mit dem Eingangsbit  $a_5$  in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus. Der Maskenerzeuger **404** gibt an den Multiplizierer **422** das Grundcodewort  $M_1 = 0101\ 0000\ 1100\ 0111\ 1100\ 0001\ 1101\ 1101$  aus, an den Multiplizierer **424** das Grundcodewort  $M_2 = 0000\ 0011\ 1001\ 0100\ 1011\ 0111\ 00001\ 1100$  und an den Multiplizierer **426** das Grundcodewort  $M_4 = 0001\ 0101\ 1111\ 0010\ 0110\ 1100\ 1010\ 1100$ . Dann multipliziert der Multiplizierer **422** das Grundcodewort  $M_1$  mit dem Eingangsbit  $a_6$  in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus, multipliziert der Multiplizierer **424** das Grundcodewort  $M_2$  mit dem Eingangsbit  $a_7$  in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus und multipliziert der Multiplizierer **426** das Grundcodewort  $M_4$  mit dem Eingangsbit  $a_8$  in der Symboleinheit und gibt seine Ausgabe an den Exklusiv-ODER-Operator **440** aus. Weiterhin erzeugt der Maskenerzeuger **404** das andere Grundcodewort  $M_8$  und gibt das erzeugte Grundcodewort  $M_8$  an den Multiplizierer **428** aus. Weil jedoch das an den Multiplizierer **428** angelegte Eingangsbit  $a_9$  eine Null ist, gibt der Multiplizierer **428** Nullen an den Exklusiv-ODER-Operator **440** aus, so dass die Ausgabe des Exklusiv-ODER-Operators **440** nicht beeinflusst wird. Das heißt, ein durch die Exklusiv-ODER-Verknüpfung der Ausgabewerte aus den Multiplizierern **410, 412, 414, 416, 418, 420, 422, 424, 426** und **428** durch den Exklusiv-ODER-Operator **440** ist gleich einem durch die Exklusiv-ODER-Verknüpfung der Ausgabewerte der Multiplizierer **410, 412, 414, 416, 418, 420, 422, 424** und **426** bestimmten Wert. Die 32 aus dem Exklusiv-ODER-Operator **440** ausgegebenen Symbole werden an den Aussonderer **460** gegeben. Zu diesem Zeitpunkt empfängt die Steuereinrichtung **450** Codelängeninformation und gibt an den Aussonderer **460** ein Steuersignal aus, das die Aussonderungspositionen auf der Basis der Codelängeninformation angibt. Der Aussonderer **460** sondert dann das 0-te, 2-te, 8-te, 19-te und 20-te codierte Symbol aus den insgesamt 32 codierten Symbolen (0-tes bis 31-tes Symbol) in Übereinstimmung mit dem Steuersignal aus der Steuereinrichtung **450** aus. Mit anderen Worten sondert der Aussonderer **460** fünf Symbole aus den 32 codierten Symbolen aus und gibt dann 28 nicht-ausgesonderte codierte Symbole aus.

**[0091]** Die folgende Tabelle 5 stellt die Aussonderungsmuster dar, mit denen alle Codierer der Tabelle 1 durch den Codierer von [Fig. 4](#) realisiert werden können. Die Aussonderungsmuster der Tabelle 5 werden auf den Aussonderer **460** von [Fig. 4](#) angewandt, um einen  $(n,k)$ -Code (wobei  $n = 3, 4, \dots, 14, 18, 19, \dots, 29$  und  $k = 1, 2, 3, 4, 5, 6, 7, 8, 9$ ) zu realisieren.

Tabelle 5

Code	0 1 2 3 4	5 6 7 8 9	10 11 12 13 14	15 16 17 18 19	20 21 22 23 24	25 26 27 28 29	30 31
(3,1)	10101	00000	00000	00000	00000	00000	00
(4,1)	10101	01000	00000	00000	00000	00000	00
(5,1)	10101	01010	00000	00000	00000	00000	00
(6,2)	11101	11000	00000	00000	00000	00000	00
(7,2)	11101	11010	00000	00000	00000	00000	00
(8,2)	11101	11011	00000	00000	00000	00000	00
(9,3)	11111	11001	10000	00000	00000	00000	00
(10,3)	11111	11011	01000	00000	00000	00000	00
(11,3)	11111	11000	01111	00000	00000	00000	00
(12,4)	00011	11111	11111	00000	00000	00000	00
(13,4)	00011	11111	11111	10000	00000	00000	00
(14,4)	00011	11111	11111	11000	00000	00000	00
(18,8)	01111	11010	10111	10110	11110	00000	00
(19,8)	11111	10111	00100	10010	01010	10111	10
(20,8)	11111	11111	01000	11110	00100	11001	10
(21,7)	00000	01011	11011	11101	10110	11111	11
(22,7)	11111	11101	11011	10100	11101	10011	00
(23,7)	11110	11100	11011	10101	11100	11111	01
(24,8)	10111	11011	11101	01111	01111	01111	00
(25,8)	11110	11111	10110	01111	00011	11111	11
(26,8)	11111	11011	11101	01111	01111	01111	01
(27,9)	01011	11101	11111	11110	01111	11111	11
(28,9)	11111	10111	00101	11111	11111	11111	11
(29,9)	11111	10111	00111	11111	11111	11111	11

[0092] In der Tabelle 5, gibt '0' eine Position wieder, bei der das codierte Symbol ausgesondert wird, während '1' eine Position wiedergibt, bei der das codierte Symbol nicht ausgesondert wird. Indem die Aussonderungsmuster von Tabelle 5 verwendet werden, können die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole auch dann berechnet werden, wenn die Verhältnisse der ersten TFCI-Informationsbits zu den zweiten TFCI-Informationsbits 2:8, 3:7, 4:6, 6:4, 7:3, 8:2 und 9:1 sind. Die Operation der Codierer **200** und **205** sollte aus den Aussonderungsmustern der Tabelle 5 und den vorausgehenden Beschreibungen für den Fall, dass das Verhältnis der ersten TFCI-Informationsbits zu den zweiten TFCI-Informationsbits gleich 1:9 ist, deutlich werden.

[0093] Nach den vorstehenden Operationen werden die codierten Symbole, die aus den Codierern **200** und **205** ausgegeben werden, durch einen Anordner (oder Multiplexer) **210** angeordnet (zeitgemultiplext), um ein gemultiplextes 32-Symbol-Signal zu erzeugen.

[0094] Im Folgenden wird das Verfahren zum Anordnen der ersten codierten TFCI-Symbole und der zweiten codierten TFCI-Symbole durch den Codiertsymbol-Anordner **210** beschrieben. Der Codiertsymbol-Anordner **210** ordnet die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole aus den Codierern **200** und **205** derart an, dass die codierten TFCI-Symbole so gleichmäßig wie möglich in einem Funkrahmen angeordnet sind. Das heißt, der Codiertsymbol-Anordner **210** bildet die Informationsbits  $a_k$  auf den codierten Bits  $b_l$  ab, wie es in der Beschreibung des Standes der Technik definiert wurde. Von den codierten Symbolen, die durch das Codieren der Informationsbits  $a_k$  erhalten werden, wird ein  $x$ -tes codiertes Symbol aus den codierten Symbolen, die durch das Codieren der ersten TFCI-Bits erhalten werden, als  $c_x^1$  definiert, wobei  $x$  eine Ganzzahl einschließlich von '0' ist, und ein  $y$ -tes codiertes Symbol aus den codierten Symbolen, die durch das Codieren der zweiten TFCI-Bits erhalten werden, wird als  $c_y^2$  definiert, wobei  $y$  eine Ganzzahl einschließlich von '0' ist. Die Summe aus einem  $x$ -Wert des letzten Symbols für  $c_x^1$  und einem  $y$ -Wert des letzten Symbols für  $c_y^2$  sollte immer gleich 32 sein. Weiterhin ist die Summe der Anzahl der codierten Symbole  $c_x^1$  und der Anzahl der codierten Symbole  $c_y^2$  gleich 32. Deshalb hat der Codiertsymbol-Anordner **210** eine Funktion zum Abbilden der codierten Symbole  $c_x^1$  und  $c_y^2$  auf die Bits  $b_l$ . Die  $b_l$  Bits werden auf die  $d_m$  Bits abgebildet, bevor sie über den tatsächlichen Funkrahmen für die entsprechenden Bedingungen A1, A2, A3 und A4 übertragen werden.

[0095] Unter den Bedingungen A2, A3 und A4 werden alle 32  $b_l$  Bits in vorteilhafter Weise übertragen. Unter der Bedingung A1 werden jedoch die Bits  $d_{30}(b_{30})$  und  $d_{31}(b_{31})$  nicht übertragen, so dass eines der codierten Symbole  $c_x^1$  und  $c_y^2$  ausgewählt werden muss, um auf die Bits  $d_{30}(b_{30})$  und  $d_{31}(b_{31})$  abgebildet zu werden. Die Regeln für die Abbildung der codierten Symbole  $c_x^1$  und  $c_y^2$  auf die Bits  $d_{30}(b_{30})$  und  $d_{31}(b_{31})$  werden im folgenden genannt.

Regel 1: die zuletzt codierten Symbole der ersten codierten TFCI-Symbole und der zweiten codierten TFCI-Symbole werden auf  $d_{30}(b_{30})$  und  $d_{31}(b_{31})$  abgebildet.

Regel 2: arbiträr codierte Symbole der ersten codierten TFCI-Symbole und der zweiten codierten TFCI-Symbole werden auf  $d_{30}(b_{30})$  und  $d_{31}(b_{31})$  abgebildet.

Regel 3: zwei arbiträr codierte Symbole, die aus einem Codierer mit einer erhöhten Codiertrate ausgegeben werden, werden auf  $d_{30}(b_{30})$  und  $d_{31}(b_{31})$  abgebildet.

Regel 4: zwei arbiträr codierte Symbole, die aus einem Codierer mit einer hohen Codiertrate ausgegeben werden, werden auf  $d_{30}(b_{30})$  und  $d_{31}(b_{31})$  abgebildet.

Regel 5: zwei arbiträr codierte Symbole, die aus einem anderen Codierer als dem Codierer mit einer erhöhten Codiertrate ausgegeben werden, werden auf  $d_{30}(b_{30})$  und  $d_{31}(b_{31})$  abgebildet.

**[0096]** Bei der Anwendung von Regel 1, Regel 2, Regel 3, Regel 4 und Regel 5 sollte Folgendes berücksichtigt werden. Wenn ein oder zwei codierte Symbole aus den codierten Symbolen  $c_x^1$  und  $c_y^2$  jedes Codes übertragen werden, sollte folgendes berücksichtigt werden: (1) wie die Leistung jedes Codes für den ersten TFCI oder den zweiten TFCI geändert wird, (2) welcher TFCI, der erste TFCI oder der zweite TFCI, in seiner Zuverlässigkeit (oder Leistung) erhöht werden sollte, (3) welche codierten Symbole, die codierten Symbole  $c_x^1$  und  $c_y^2$  aus den entsprechenden Codierern, auf  $d_{30}(b_{30})$  und  $d_{31}(b_{31})$  abgebildet werden sollten, um die Leistungsver schlechterung der Codes zu minimieren, und (4) welcher TFCI, der erste TFCI oder der zweite TFCI, während der Übertragung betont werden soll.

**[0097]** Bei der folgenden Beschreibung von Regel 1, Regel 2, Regel 3 und Regel 5 wird angenommen, dass das Informationsbitverhältnis des ersten TFCI zu dem zweiten TFCI 3:7 in dem HSM ist. Weiterhin wird bei der Beschreibung von Regel 4 angenommen, dass das Informationsbitverhältnis des ersten TFCI zu dem zweiten TFCI gleich 3:7 unter der Bedingung A1 ist.

**[0098]** Im Folgenden wird die Regel 1 mit Bezug auf ein Beispiel beschrieben. Ein (9,3)-Code und ein (23,7)-Code oder ein (11,3)-Code und ein (21,7)-Code sind in Übereinstimmung mit dem Informationsbitverhältnis des ersten TFCI zu dem zweiten TFCI verfügbar. Der (9,3)-Code und der (23,7)-Code werden verwendet, um die Codeleistung des zweiten TFCI zu erhöhen, während der (11,3)-Code und der (21,7)-Code verwendet werden, um die Codeleistung des ersten TFCI zu erhöhen. Wenn die Regel 1 angewendet wird, wird das zuletzt codierte Symbol des (9,3)-Codes nicht übertragen, so dass eine tatsächliche Codiertrate des (9,3)-Codes zu (8,3) wird; das zuletzt codierte Symbol des (23,7)-Codes wird nicht übertragen, so dass eine tatsächliche Codiertrate des (23,7)-Codes zu (22,7) wird; das zuletzt codierte Symbol des (11,3)-Codes wird nicht übertragen, so dass eine tatsächliche Codiertrate des (11,3)-Codes zu (10,3) wird; und das zuletzt codierte Symbol des (21,7)-Codes wird nicht übertragen, so dass eine tatsächliche Codiertrate des (21,7)-Codes zu (20,7) wird. In der Regel 1 codieren die Codierer ihre zuletzt codierten Symbole auf  $d_{30}(b_{30})$  und  $d_{31}(b_{31})$  und tragen so zu einer Vereinfachung der Abbildung bei. Unter der Bedingung A1 jedoch wird die tatsächliche Codiertrate des ersten TFCI zu dem zweiten TFCI herabgesetzt, was eine Reduktion der Codeleistung des ersten TFCI und des zweiten TFCI zur Folge hat.

**[0099]** Im Folgenden wird die Regel 2 mit Bezug auf ein Beispiel beschrieben. Ein (9,3)-Code und ein (23,7) Code oder ein (11,3)-Code und ein (21,7)-Code sind in Übereinstimmung mit dem Informationsbitverhältnis des ersten TFCI zu dem zweiten TFCI verfügbar. Wenn die Regel 2 angewendet wird, wird ein willkürliches codiertes Symbol des (9,3)-Codes nicht übertragen, so dass eine tatsächliche Codiertrate des (9,3)-Codes zu (8,3) wird; wird ein willkürliches codiertes Symbol des (23,7)-Codes nicht übertragen, so dass eine tatsächliche Codiertrate des (23,7)-Codes zu (22,7) wird; wird ein willkürliches codiertes Symbol des (11,3)-Codes nicht übertragen, so dass eine tatsächliche Codiertrate des (11,3)-Codes zu (10,3) wird; und wird ein willkürliches codiertes Symbol des (21,7)-Codes nicht übertragen, so dass eine tatsächliche Codiertrate des (21,7)-Codes zu (20,7) wird. Die willkürlichen codierten Symbole können aus den vier Codes ausgewählt werden, so dass die tatsächlichen Codeleistungen nicht vermindert werden, obwohl die tatsächlichen Codiertraten der entsprechenden Codes herabgesetzt werden. Es können jedoch mehrere Codes in der Leistung vermindert werden, unabhängig von den ausgewählten willkürlichen codierten Symbolen. Die Regel 2 ist komplexer als die Regel 1 in einem Verfahren zum Abbilden der codierten Symbole  $c_x^1$  und  $c_y^2$ . Unter der Bedingung A1 können jedoch die Codeleistungen des ersten TFCI und des zweiten TFCI unabhängig von der Reduktion der tatsächlichen Codiertrate der Codierer für den ersten TFCI und den zweiten TFCI aufrechterhalten werden.

**[0100]** Im Folgenden wird die Regel 3 mit Bezug auf ein Beispiel beschrieben. Ein (9,3)-Code und ein (23,7) Code oder ein (11,3)-Code und ein (21,7)-Code sind in Übereinstimmung mit dem Informationsbitverhältnis des ersten TFCI zu dem zweiten TFCI verfügbar. Wenn die Regel 3 angewendet wird, werden zwei willkürliche codierte Symbole des (23,7)-Codes nicht übertragen, so dass eine tatsächliche Codiertrate des (23,7)-Codes zu

(21,7) wird, und werden zwei willkürliche codierte Symbole des (11,3)-Codes nicht übertragen, so dass eine tatsächliche Codierrate des (11,3)-Codes zu (9,3) wird. Die willkürlichen codierten Symbole können derart ausgewählt werden, so dass die tatsächlichen Codeleistungen nicht vermindert werden, obwohl die tatsächlichen Codierraten der entsprechenden Codes herabgesetzt werden. Die meisten der Codes vermindert jedoch ihre Leistung. Bei der Regel 3 wird die tatsächlichen Codierrate der entsprechenden Codes zu (9,3) oder (21,7), so dass die Leistung der TFCI-Codewörter mit einer tatsächlichen Datenrate 1/3 für die Bedingung A1 erfüllt wird. Die erhöhte Anzahl der TFCI-Symbole verursacht jedoch eine Verminderung der Leistung des Codes, für den die codierten Symbole zahlreicher geworden sind, obwohl die Leistung des ersten TFCI-Codes oder des zweiten TFCI-Codes erhöht werden sollte. Die Regel 3 kann die willkürlichen Symbole suchen, welche die Leistung der Codes nicht erhöhen. Ähnlich wie die Regel 2, weist auch die Regel 3 ein komplexes Abbildungsverfahren auf. Um das Abbildungsverfahren zu vereinfachen, werden die letzten zwei Symbole der codierten Symbole aus dem Codierer mit einer erhöhten Anzahl von codierten Symbolen auf  $d_{30}(b_{30})$  und  $d_{31}(b_{31})$  abgebildet.

**[0101]** Im Folgenden wird die Regel 4 mit Bezug auf ein Beispiel beschrieben. Ein (23,7)-Code und ein (9,3) Code oder ein (21,7)-Code und ein (11,3)-Code sind in Übereinstimmung mit dem Informationsbitverhältnis des ersten TFCI zu dem zweiten TFCI verfügbar. Der (21,7)-Code und der (11,3)-Code werden verwendet, um die Codierrate des zweiten TFCI zu erhöhen, während der (23,7)-Code und der (9,3)-Code verwendet werden, um die Codierrate des ersten TFCI zu erhöhen. Wenn die Regel 4 angewendet wird, werden die letzten zwei codierten Symbole des (23,7)-Codes nicht übertragen, so dass eine tatsächliche Codierrate des (23,7)-Codes zu (21,7) wird und die Codierrate des (9,3)-Codes unverändert bleibt; und die letzten zwei codierten Symbole des (21,7)-Codes werden nicht übertragen, so dass eine tatsächliche Codierrate des (21,7)-Codes zu (19,7) wird und die Codierrate des (11,3)-Codes unverändert bleibt. Wenn die Regel 4 angewendet wird, werden die letzten zwei Symbole oder arbiträre zwei Symbole aus den entsprechenden Codierer mit der großen Anzahl von Codewörtern auf  $d_{30}(b_{30})$  und  $d_{31}(b_{31})$  abgebildet. In der Regel 4 werden zwei codierte Symbole des Codes mit einem längeren Codewort nicht übertragen, so dass die Leistung des Codes mit dem längeren Codewort reduziert wird, während die Leistung des Codes mit dem kürzeren Codewort sichergestellt wird.

**[0102]** Im Folgenden wird die Regel 5 mit Bezug auf ein Beispiel beschrieben. Wenn angenommen wird, dass die Informationsbitrate des ersten TFCI zu dem zweiten TFCI 3:7 ist und die Leistung des Codeworts für die Übertragung des zweiten TFCI erhöht wird, dann sind ein (9,3)-Code und ein (23,7)-Code verfügbar. Um in der Regel 5 den zweiten TFCI mit hoher Zuverlässigkeit zu übertragen, werden zwei arbiträr codierte Symbole des (9,3)-Codes auf  $d_{30}(b_{30})$  und  $d_{31}(b_{31})$  abgebildet, so dass die tatsächliche Codierrate zu (7,3) wird. In der Regel 5 wird die Leistung des ersten TFCI-Codierers reduziert, wobei aber die codierten Symbole für den zweiten TFCI nicht beschädigt werden, so dass das zweite TFCI-Codewort sicher übertragen werden kann.

**[0103]** In der vorstehenden Beschreibung von Regel 1, Regel 2, Regel 3 und Regel 4 werden  $c_x^1$  und  $c_y^2$  nur unter der Bedingung A1 auf  $b_i$  abgebildet. Unter den Bedingungen A2, A3 und A4 werden alle 32 codierten Symbole übertragen oder werden die 32 codierten Symbole durch Wiederholung übertragen, so dass keine separate Abbildungsregel erforderlich ist und die intakten Abbildungsregeln für die Bedingung A1 verwendet werden können. Außerdem können Regel 1, Regel 2, Regel 3, Regel 4 und Regel 5 gemäß den Umständen entsprechend verwendet werden.

**[0104]** Die vorliegende Erfindung gibt ein Verfahren zum Abbilden der  $c_x^1$  und  $c_y^2$  Symbole auf die  $b_i$  Bits als Beispiel an. In dem folgenden Beispiel können das auf die Regel 1 anwendbare Verfahren und das Verfahren, das die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole so gleichmäßig wie möglich anordnet, um eine Zeitübertragungsverstärkung zu erhalten, auch auf ein anderes Abbildungsverfahren angewendet werden. Unter der Bedingung A1 werden die letzten codierten Symbole von  $c_x^1$  und  $c_y^2$  auf  $b_{30}$  oder  $b_{31}$  abgebildet.

**[0105]** Von den durch die vorliegende Erfindung vorgesehenen Codierern weisen die 16 Codierer, die die Codierrate des ersten TFCI-Codierers oder des zweiten TFCI-Codierers erhöhen, trotz ihrer Codierrate von 1/3 eine optimale Leistung bei der Codierrate 1/3 auf.

**[0106]** Bevor das Verfahren zum Abbilden der codierten Symbole  $c_x^1$  und  $c_y^2$  auf die codierten Bits  $b_i$  beschrieben wird, wird die Anzahl der ersten codierten TFCI-Symbole  $c_x^1$  als  $n$  definiert (wobei  $n = x + 1$ ) und wird die Anzahl der zweiten codierten TFCI-Symbole  $c_y^2$  als  $m$  definiert (wobei  $m = y + 1$ ). Der einfacheren Erläuterung halber wird angenommen, dass  $n$  gleich oder kleiner als  $m$  ist und dass die Summe aus  $n$  und  $m$  gleich 32 ist. Es gilt also: für jeweils  $n = 4, 7, 10, 13$  und  $16$  ist  $m = 28, 25, 22, 19$  und  $16$ . Die Werte von  $n$  und  $m$  werden wie folgt definiert:

$$b_{\left\lfloor \frac{32-x(i+1)}{n} \right\rfloor - 1} = c_i^1, (0 \leq i \leq n-1) \quad \dots (1)$$

$$b_{i + \left\lfloor \frac{n-x(i+\frac{1}{2})}{32-n} \right\rfloor} = c_i^2, (0 \leq i \leq m-1) \quad \dots (2)$$

**[0107]** In der Gleichung (1) gibt n die Gesamtanzahl der ersten codierten TFCI-Symbole an und gibt i die Indizes der ersten codierten TFCI-Symbole an, wobei  $0 \leq i \leq n - 1$  (oder x). Die Indizes werden in der Reihenfolge ihrer Erzeugung zugewiesen. Die Gleichung (1) gibt die Positionen der  $b_i$  Bits wieder, auf die die ersten codierten TFCI-Symbole abgebildet werden. In der Gleichung (1) gibt  $\lfloor x \rfloor$  eine Ganzzahl an, die durch das Abrufen eines gegebenen Werts x erhalten wird.

**[0108]** In der Gleichung (2) gibt n die Gesamtanzahl der ersten codierten TFCI-Symbole an, gibt m die Gesamtanzahl der codierten zweiten codierten TFCI-Symbole an und gibt i die Indizes der zweiten codierten TFCI-Symbole an, wobei  $0 \leq i \leq m - 1$  (oder y). Die Indizes werden in der Reihenfolge ihrer Erzeugung zugewiesen. Die Gleichung (2) gibt die Positionen der  $b_i$  Bits an, auf die die zweiten codierten TFCI-Symbole abgebildet werden. In der Gleichung (2) gibt  $\lfloor x \rfloor$  einen Maximalwert aus den Ganzzahlen an, der kleiner oder gleich einem bestimmten Wert x ist.

**[0109]** Die ersten codierten TFCI-Symbole werden in Übereinstimmung mit der Gleichung (1) abgebildet (angeordnet), während die zweiten codierten TFCI-Symbole in Übereinstimmung mit der Gleichung (2) abgebildet werden. Was die Reihenfolge der codierten Symbole betrifft, können entweder die ersten codierten TFCI-Symbole oder die zweiten codierten TFCI-Symbole zuerst angeordnet werden. Alternativ hierzu können die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole gleichzeitig angeordnet werden.

**[0110]** Wenn die Anzahl der ersten codierten TFCI-Symbole größer ist als die Anzahl der zweiten codierten TFCI-Symbole ( $n > m$ ), wird die Gleichung (2) verwendet, um die ersten codierten TFCI-Symbole abzubilden, und wird die Gleichung (1) verwendet, um die zweiten codierten TFCI-Symbole abzubilden.

**[0111]** Die folgende Tabelle 6 zeigt die codierten Symbole, die in Übereinstimmung mit der Gleichung (1) und der Gleichung (2) erzeugt werden. In der Tabelle 6 gibt '0' die Positionen an, bei denen die ersten codierten TFCI-Symbole  $c_x^1$  übertragen werden, und gibt '1' die Positionen an, bei denen die zweiten codierten TFCI-Symbole  $c_y^2$  übertragen werden.

Tabelle 6

Code	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
4,1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	0
7,2	1	1	1	0	1	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	1	0	1	1	1	1	1	0
10,3	1	1	0	1	1	0	1	1	0	1	1	0	1	1	1	0	1	1	0	1	1	0	1	1	0	1	1	0	1	1	1	0
13,4	1	0	1	0	1	1	0	1	0	1	1	0	1	0	1	1	0	1	0	1	1	0	1	0	1	1	0	1	0	1	1	0
16,5	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
19,6	0	1	0	1	0	0	1	0	1	0	0	1	0	1	0	0	1	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0
22,7	0	0	1	0	0	1	0	0	1	0	0	1	0	0	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0
25,8	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0
28,9	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

**[0112]** Die Tabelle 6 ist ein Beispiel für die Anordnung der ersten codierten TFCI-Symbole und der zweiten codierten TFCI-Symbole. Um die Positionen für die Übertragung der in Übereinstimmung mit der Tabelle 6 erzeugten codierten Symbole über einen physikalischen Kanal zu wählen, werden verschiedene Verfahren unter den Bedingungen A1, A2, A3 und A4 verwendet.

**[0113]** Unter der Bedingung A1 werden die codierten Symbole  $c_x^1$  und  $c_y^2$ , die auf das Bit  $b_{30}$  oder  $b_{31}$  abgebildet werden, ausgesondert, wobei dann vor der Übertragung 30  $b_i$  Bits auf die Bits  $d_m$  abgebildet werden. Unter der Bedingung A2 werden die unter der Bedingung A1 abgebildeten 32  $b_i$  Bits sequentiell drei Mal wiederholt, noch ein Mal von  $b_0$  bis  $b_{23}$  wiederholt und dann vor der Übertragung auf insgesamt 120  $d_m$  Bits abgebildet. Unter der Bedingung A3 werden die unter der Bedingung A1 angeordneten 32  $b_i$  Bits vor der Übertragung auf die Positionen der Übertragungsbits  $d_m$  abgebildet. Unter der Bedingung A4 werden die unter der Bedingung A1 abgebildeten 32  $b_i$  Bits vier Mal wiederholt, wobei sie dann vor der Übertragung auf die Positionen der 128

Übertagungsbits  $d_m$  abgebildet werden.

[0114] [Fig. 6](#) stellt den detaillierten Aufbau des Codiertsymbol-Anordners **210** von [Fig. 2](#) dar. In [Fig. 6](#) gibt das Bezugszeichen **601** die zweiten codierten TFCI-Symbole  $c_y^2$  aus dem Codierer **200** von [Fig. 2](#) wieder, und gibt das Bezugszeichen **611** die ersten codierten TFCI-Symbole  $c_x^1$  aus dem Codierer **205** wieder. Die Speichereinrichtungen **603** und **613** sind Einrichtungen zum Speichern der codierten Symbole  $c_x^1$  und  $c_y^2$ , die unter Verwendung eines Speichers realisiert werden. Durch eine Modifikation der Hardwarestruktur können jedoch direkt die zweiten codierten TFCI-Symbole **601** und die ersten codierten TFCI-Symbole **611** zu einem Schalter **620** gegeben werden, ohne dass die codierten Symbole in den Speichereinrichtungen gespeichert werden. Der Schalter **620** wird in Übereinstimmung mit empfangener Codeauswahlinformation zu der Speichereinrichtung **603** oder **613** geschaltet. Die codierten Symbole  $c_x^1$  und  $c_y^2$  aus den Speichereinrichtungen **603** und **613** werden in einer Permanentspeichereinrichtung **621** gespeichert.

[0115] Eine Steuereinrichtung **670** ordnet die empfangenen codierten Symbole  $c_x^1$  und  $c_y^2$  in Übereinstimmung mit der Tabelle 6 an. Die Symbolanordnung kann entweder durch Hardware oder durch Software realisiert werden. Ein Schalter **630** schaltet die  $b_i$  Bits, die in der Speichereinrichtung **621** gespeichert sind, zu einem Ausgabeknoten oder einem Wiederholer **640** in Übereinstimmung mit Information zu der Anzahl der codierten TFCI-Symbole, d.h. mit Information zu den  $d_m$  Bits. Wenn also unter der Bedingung A1 oder A3 die Anzahl der  $d_m$  Bits gleich 30 oder 32 ist, schaltet der Schalter **630** die  $b_i$  Bits aus der Speichereinrichtung **621** zu dem Ausgabeknoten. Unter den Bedingungen A2 oder A4, wenn die Anzahl der  $d_m$  Bits gleich 120 oder 128 ist, schaltet der Schalter **630** die  $b_i$  Bits von der Speichereinrichtung **621** zu dem Wiederholer **640**. Der Wiederholer **640** wiederholt die Bits  $b_i$  aus dem Schalter **630** für eine vorbestimmte Anzahl von Wiederholungen, um die Bits  $d_m$  für die Bedingungen A2 oder A4 zu erhalten. Der Wiederholer **640** ist unter den Bedingungen A2 und A4 aktiviert. Der Wiederholer **640** kann auch durch Software in der Steuereinrichtung **670** realisiert werden.

[0116] Die durch den Codiertsymbol-Anordner **210** in Übereinstimmung mit der Tabelle 6 angeordneten  $b_i$  Bits werden zu einem Multiplexer **220** gegeben, wo sie mit physikalischer Information wie etwa den über den DP-CCH und DPDCH übertragenen TPC-Bits und Pilotbits zeitgemultiplext werden. Der Multiplexer **220** erzeugt den DPCH, dessen Aufbau in [Fig. 5](#) gezeigt ist. [Fig. 5](#) zeigt den Aufbau des von einem Knoten B zu einem Benutzergerät übertragenen DPCH.

[0117] In [Fig. 5](#) gibt das Bezugszeichen **510** den Aufbau eines Funkrahmens mit 15 Zeitschlitz an. Das Bezugszeichen **520** gibt den Aufbau eines Zeitschlitzes eines Abwärtsverbindungskanal wieder, in dem der DPDCH und DPCCH auf einer Zeiteilungsbasis getrennt sind. Das heißt, der Zeitschlitz umfasst zwei Datenfelder **501** und **507** für den DPDCH, und ein TPC-Feld **503**, ein TFCI-Feld **505** und ein Pilotfeld **509** für den DPCCH. Das TPC-Feld **503** wird verwendet, um einen TPC-Befehl für einen Aufwärtsverbindungskanal von dem Benutzergerät zu dem Knoten B zu übertragen, und das Pilotfeld **509** wird verwendet, um eine Änderung in dem Aufwärtsverbindungskanal und der Signalstärke durch das Benutzergerät zu schätzen. Weiterhin wird das TFCI-Feld **505** verwendet, um die codierten TFCI-Übertragungssymbole  $d_m$  aus dem Codiertsymbol-Anordner **210** zu dem Benutzergerät zu übertragen.

[0118] Die DPCH-Ausgabe aus dem Multiplexer **220** wird zu einem Spreizer **230** gegeben, und gleichzeitig wird ein Spreizcode für die Kanaltrennung von einem Spreizcodeerzeuger **235** zu dem Spreizer **230** gegeben. Der Spreizer **230** kanalspreizt den DPCH mit dem Spreizcode in einer Symboleinheit und gibt den kanalspreizten DPCH in einer Chip-Einheit aus. Der kanalspreizte DPCH wird zu einem Verschlüsseler **240** gegeben, und gleichzeitig wird ein Verschlüsselungscode von einem Verschlüsselungscodeerzeuger **245** zu dem Verschlüsseler **240** gegeben. Der Verschlüsseler **240** verschlüsselt den kanalspreizten DPCH mit dem Verschlüsselungscode.

## 2. Zweite Ausführungsform des Senders

[0119] [Fig. 13](#) zeigt den Aufbau eines Senders gemäß einer zweiten Ausführungsform der vorliegenden Erfindung. Wie in [Fig. 13](#) gezeigt, codieren ein Codierer **1303** und ein Codierer **1313** jeweils die TFCI-Informationsbits für den DSCH (zweite TFCI-Informationsbits) und die TFCI-Informationsbits für den DCH (erste TFCI-Informationsbits). Der Codierer **1303** und der Codierer **1313** weisen den gleichen Aufbau auf wie der in [Fig. 4](#) gezeigte Codierer, wobei jedoch kein Aussonderer und keine Steuereinrichtung vorgesehen sind. Die 32 codierten Symbole aus dem Codierer **1303** werden zu einer zweiten Codiert-TFCI-Symbol-Speichereinrichtung **1305** gegeben, und die 32 codierten Symbole aus dem Codierer **1313** werden zu einer ersten Codiert-TFCI-Symbol-Speichereinrichtung **1315** gegeben. Die erste Codiert-TFCI-Symbol-Speichereinrichtung

**1315** und die zweite Codierte-TFCI-Symbol-Speichereinrichtung **1305** können den gleichen Speicher verwenden. In diesem Fall müssen die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole logisch unterscheidbar sein. Die zweite Codierte-TFCI-Symbol-Speichereinrichtung **1305** und die erste Codierte-TFCI-Symbol-Speichereinrichtung **1315** sehen für einen Symbol-Anordner **1350** aus den darin gespeicherten 32 codierten Symbolen die codierten Symbole vor, die in Übereinstimmung mit jeweils zweiter Codiertsymbol-Auswahlinformation **1331** und erster Codiertsymbol-Auswahlinformation **1333** aus einer Steuereinrichtung **1330** ausgewählt werden. Die zweite Codiertsymbol-Auswahlinformation **1331** und die erste Codiertsymbol-Auswahlinformation **1333** sind mit dem in [Fig. 5](#) gezeigten Aussonderungsmuster identisch und werden verwendet, um die gewünschten codierten Symbole aus den 32 codierten Symbolen auszuwählen, anstatt die codierten Symbole in Übereinstimmung mit dem Aussonderungsmuster auszusondern. Die Ausgaben aus der zweiten Codierte-TFCI-Symbol-Speichereinrichtung **1305** und der ersten Codierte-TFCI-Symbol-Speichereinrichtung **1315** sind jeweils gleich  $c_y^2$  und  $c_x^1$ . Der Symbol-Anordner **1350** ordnet die zweiten codierten TFCI-Symbole und die ersten codierten TFCI-Symbole, die in der Form der Tabelle 6 erhalten werden, in Übereinstimmung mit Codiertsymbol-Anordnungsinformation **1335** an, die von der Steuereinrichtung **1330** erhalten wird. Eine Ausgabe aus dem Symbol-Anordner **1350** wird zu  $b_i$ . Die Steuereinrichtung **1330** von [Fig. 13](#) steuert die Symbolspeichereinrichtungen **1305** und **1315** und den Symbol-Anordner **1350** in Übereinstimmung mit jeweils dem Symbol-Aussonderungsmuster der Tabelle 5 und dem Symbol-Anordnungsmuster der Tabelle 6, um dieselbe Ausgabe wie der Codierer und der Symbol-Selektor von [Fig. 4](#), [Fig. 6](#), und [Fig. 8](#) vorzusehen.

[0120] [Fig. 19](#) zeigt den detaillierten Aufbau des Codiertsymbol-Anordners **1350**. Wie in [Fig. 19](#) gezeigt, umfasst der Codiertsymbol-Anordner eine Speichereinrichtung **1901**, eine Steuereinrichtung **1910** und einen Schalter. Die Speichereinrichtung **1901**, die eine Einrichtung zum Speichern der ersten codierten TFCI-Symbole und der zweiten codierten TFCI-Symbole in der Form der Tabelle 6 ist, ordnet die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole unter der Kontrolle der Steuereinrichtung **1910** und gibt dann sequentiell die  $b_i$  Bits aus. Die Steuereinrichtung **1910** steuert den Schalter, um für die Speichereinrichtung **1901** die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole vorzusehen, und steuert die Speichereinrichtung **1901**, um die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole in der Form der Tabelle 6 neu anzuordnen. Die Steuereinrichtung **1910** von [Fig. 19](#) kann durch Software realisiert werden. In diesem Fall kann die Software eine Adress-Steuereinrichtung werden. Alternativ hierzu können der Symbol-Anordner **1350**, die erste Codierte-TFCI-Symbol-Speichereinrichtung **1315** und die zweite Codierte-TFCI-Symbol-Speichereinrichtung **1305** entweder im gleichen Speicher oder in verschiedenen Speichern realisiert werden. Bei einer Realisierung durch Software steuert die Steuereinrichtung **1330** die Adressen für die Speicher des Symbol-Anordners **1350**, der ersten Codierte-TFCI-Symbol-Speichereinrichtung **1315** und der zweiten Codierte-TFCI-Symbol-Speichereinrichtung **1305**, um die Operation der Codierer und des Symbol-Anordners durch Software durchzuführen.

### 3. Dritte Ausführungsform des Senders

[0121] [Fig. 3](#) zeigt den Aufbau eines Senders gemäß einer dritten Ausführungsform der vorliegenden Erfindung. Der Sender codiert die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole unter Verwendung eines einzigen Codierers.

[0122] Wie in [Fig. 3](#) gezeigt, werden zweite codierte TFCI-Bits **301** und erste codierte TFCI-Bits **303** zu einem Selektor **310** gegeben. Der Selektor **310** gibt wahlweise an einen Codierer **311** die zweiten codierten TFCI-Bits **301** oder die ersten codierten TFCI-Bits **303** in Übereinstimmung mit TFCI-Auswahlinformation aus einer Steuereinrichtung **330** aus. Der detaillierte Aufbau des Selektors **310** ist in [Fig. 7](#) beispielhaft gezeigt. Wie in [Fig. 7](#) gezeigt, werden die zweiten TFCI-Bits **301** zu einer Speichereinrichtung **703** gegeben und werden die ersten TFCI-Bits **303** zu einer Speichereinrichtung **713** gegeben. Die Speichereinrichtungen **703** und **713**, die Einrichtungen zum Speichern der zweiten TFCI-Bits **301** und der ersten TFCI-Bits **303** können durch Speicher realisiert werden. Indem der Hardwareaufbau modifiziert wird, können die zweiten TFCI-Bits **301** und die ersten TFCI-Bits **303** direkt zu einem Schalter **720** gegeben werden, ohne die Speichereinrichtungen zu verwenden. Der Schalter **720** wird alternativ zu der Speichereinrichtung **703** oder **713** in Übereinstimmung mit empfangener Codeauswahlinformation geschaltet. Die aus dem Schalter **720** ausgegebenen zweiten TFCI-Bits und ersten TFCI-Bits werden zu dem Codierer **311** gegeben. Der Selektor **310** kann auch durch Software realisiert werden.

[0123] Der Codierer **311** weist den Aufbau von [Fig. 4](#) auf und codiert die TFCI-Bits aus dem Selektor **310** in Übereinstimmung mit Codelängeninformation aus der Steuereinrichtung **330**. Die Steuereinrichtung **330** kann auch durch Software realisiert werden.

[0124] Die aus dem Codierer **311** ausgegebenen codierten Symbole  $c_x^1$  oder  $c_y^2$  werden zu einem Symbol-Anordner **312** gegeben, wo sie in der in Tabelle 6 gezeigten Form angeordnet werden. Der interne Aufbau des Symbol-Anordners **312** ist in [Fig. 8](#) gezeigt.

[0125] Wie in [Fig. 8](#) gezeigt, ordnet eine Speichereinrichtung **801** die empfangenen codierten TFCI-Symbole in der in Tabelle 6 gezeigten Form unter der Kontrolle einer Steuereinrichtung **810** an. Von den codierten Symbolen  $c_x^1$  und  $c_y^2$  werden die ersten empfangenden codierten TFCI-Symbole in der Speichereinrichtung **801** gespeichert, bis die anderen codierten TFCI-Symbole vollständig angeordnet sind. Die Speichereinrichtung **801** gibt an einen Schalter **802**  $b_i$  Bits aus. Der Schalter **803** gibt die intakten codierten TFCI-Symbole aus der Speichereinrichtung **801** aus oder gibt die codierten TFCI-Symbole zu einem Wiederholer **805** aus, was in Übereinstimmung mit der Anzahl der Übertragungen von codierten TFCI-Symbolen geschieht. Der Wiederholer **805** wiederholt die codierten TFCI-Symbole aus dem Schalter **803** entsprechend der Anzahl der codierten TFCI-Symbole  $d_m$ , die über den physikalischen Kanal zu übertragen sind. Der Wiederholer **805** kann realisiert werden, um dieselbe Operation durch Software durchzuführen. Der Wiederholer **805** kann als interner Block der Steuereinrichtung **810** oder als separater Block realisiert werden.

[0126] Die aus dem Symbol-Anordner **312** ausgegebenen codierten TFCI-Symbole  $d_m$  werden zu einem Multiplexer **313** gegeben, wo sie mit der physikalischen Information wie etwa den über den DPCCCH und den DPDCH übertragenen TPC- oder Pilotbits zeitgemultiplext werden. Der gemultiplexte DPCH weist den in [Fig. 5](#) gezeigten Aufbau auf.

[0127] Der DPCH wird zu einem Spreizer **314** gegeben, und gleichzeitig wird ein durch einen Spreizcodeerzeuger **316** erzeugter Spreizcode an den Spreizer **314** ausgegeben. Der Spreizer **314** kanalspreizt den DPCH mit dem Spreizcode in einer Symboleinheit für die Kanaltrennung und gibt den kanalspreizten DPCH in einer Chipeinheit aus. Der kanalspreizte DPCH wird zu einem Verschlüsseler **315** gegeben, und gleichzeitig wird ein durch einen Verschlüsselungscodeerzeuger **317** erzeugter Verschlüsselungscode zu dem Verschlüsseler **315** gegeben. Der Verschlüsseler **315** verschlüsselt den kanalspreizten DPCH mit dem Verschlüsselungscode.

#### 4. Vierte Ausführungsform des Senders

[0128] [Fig. 14](#) zeigt den Aufbau eines Senders gemäß einer vierten Ausführungsform der vorliegenden Erfindung. Der Sender von [Fig. 14](#) unterscheidet sich von dem Sender von [Fig. 13](#) darin, dass ein Codierer sequenziell die ersten TFCI-Informationsbits und die zweiten TFCI-Informationsbits unter Verwendung eines einzigen Codierers codiert. Wie in [Fig. 14](#) gezeigt, werden die ersten TFCI-Informationsbits oder die zweiten TFCI-Informationsbits zu einem Codierer **1403** gegeben, wo sie codiert und dann zu einer Codiertsymbol-Speichereinrichtung **1405** gegeben werden. Die Codiertsymbol-Speichereinrichtung **1405** wählt die codierten Symbole in Übereinstimmung mit Codiertsymbol-Auswahlinformation **1401**, d.h. dem Aussonderungsmuster von Tabelle 5 aus, die sie von einer Steuereinrichtung **1430** empfängt, und gibt die ausgewählten codierten Symbole dann zu einem Codeselektor (oder Codeanordner) **1450**. Die Codiertsymbol-Speichereinrichtung **1405** kann direkt die ausgewählten ersten codierten TFCI-Symbole oder zweiten codierten TFCI-Symbole an den Codeanordner **1450** geben. Alternativ hierzu empfängt der Codierer **1403** die anderen codierten TFCI-Symbole, und die Codiertsymbol-Speichereinrichtung **1405** wählt die empfangenen codierten TFCI-Symbole in Übereinstimmung mit der Codiertsymbol-Auswahlinformation **1401** aus der Steuereinrichtung **1430** und gibt die zwei Typen von codierten TFCI-Symbolen an den Codeanordner **1450**. Der Codewähler **1450** von [Fig. 14](#) bildet die in der Form der Tabelle 6 empfangenden codierten Symbole  $c_x^1$  und  $c_y^2$  auf die Bits  $b_i$  ab. Weiterhin können die Codiertsymbol-Speichereinrichtung, der Codeanordner und die Steuereinrichtung durch Software realisiert werden.

#### 5. Fünfte Ausführungsform des Senders

[0129] [Fig. 15](#) zeigt den Aufbau eines Senders gemäß einer fünften Ausführungsform der vorliegenden Erfindung. Im Gegensatz zu den anderen Sendern führt der Sender von [Fig. 15](#) gleichzeitig eine TFCI-Codierung und eine Symbolanordnung durch.

[0130] Im Folgenden wird eine Operation des Senders mit Bezug auf ein Beispiel beschrieben, in dem die zweiten TFCI-Bits mit einer Rate von (4,1) codiert werden, die ersten TFCI-Bits mit einer Rate von (28,9) codiert werden und die codierten Symbole zu den Bits  $b_i$  angeordnet werden.

[0131] Wie in [Fig. 15](#) gezeigt, speichert ein Grundcodewort-Speicher **1501** die Grundcodewörter  $W_1$ ,  $W_2$ ,



den. Die codierten TFCI-Symbole werden zu einem Codiertsymbol-Neuanordner **910** gegeben. Der Codiertsymbol-Neuanordner **910** trennt die codierten TFCI-Symbole in die codierten Symbole für den DSCH (zweite TFCI-Informationssymbole) und die codierten Symbole für den DCH (erste TFCI-Informationssymbole) in Übereinstimmung mit Codelängeninformation und Positionsinformation. Die Codelängeninformation ist Codelängensteuerinformation auf der Basis eines Verhältnisses der TFCI-Bits für den DSCH zu den TFCI-Bits für den DCH. Die Positionsinformation ist Information, die die Positionen der codierten Symbole für den DSCH und die Positionen der codierten Symbole für den DCH wie in Tabelle 6 gezeigt angibt. Die durch den Codiertsymbol-Neuanordner **910** getrennten zweiten codierten TFCI-Symbole und ersten codierten TFCI-Symbole werden jeweils zu einem ersten Decodierer **900** und einem zweiten Decodierer **905** gegeben. Die Decodierer **900** und **905** bestimmen entsprechende Codes in Übereinstimmung mit Codelängeninformation und decodieren die zweiten codierten TFCI-Symbole und die ersten codierten TFCI-Symbole jeweils mit den bestimmten Codes. Das heißt, ein erster Decodierer **900** decodiert die zweiten codierten TFCI-Symbole und gibt zweite TFCI-Bits (TFCI-Bits für den DSCH) aus, und der zweite Decodierer **905** decodiert die ersten codierten TFCI-Symbole und gibt erste TFCI-Bits (TFCI-Bits für den DCH) aus.

[0139] [Fig. 18A](#) und [Fig. 18B](#) zeigen den detaillierten Aufbau des Codiertsymbol-Neuanordners **910** gemäß anderen Ausführungsformen der vorliegenden Erfindung. Wie in [Fig. 18A](#) gezeigt, umfasst der Codiertsymbol-Neuanordner eine Speichereinrichtung **1801**, eine Steuereinrichtung **1810** und einen Schalter. Die Speichereinrichtung **1801**, die eine Einrichtung zum Speichern der codierten TFCI-Symbole aus dem Demultiplexer **920** ist, trennt die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole unter der Kontrolle der Steuereinrichtung **1810**. Die Steuereinrichtung **1810** steuert die Speichereinrichtung **1801** und den Schalter, um die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole jeweils zu den Decodierern **905** und **900** auszugeben. Wenn alternativ hierzu nur ein Decodierer verwendet wird, gibt die Steuereinrichtung **1810** separat die zwei Typen von codierten TFCI-Symbolen an den einzelnen Decodierer aus. Die Steuereinrichtung **1810** kann durch Software realisiert werden. In diesem Fall kann die Software eine Adress-Steuereinrichtung werden.

[0140] Wie in [Fig. 18B](#) gezeigt, umfasst der Codiertsymbol-Neuanordner eine Speichereinrichtung **1821**, eine Steuereinrichtung **1820**, einen Maskenerzeuger **1830**, einen Multiplizierer **1815** und einen Multiplizierer **1817**. Die Speichereinrichtung **1821** führt dieselbe Operation durch wie die Speichereinrichtung **1801** von [Fig. 18A](#). Die Steuereinrichtung **1820** steuert die Speichereinrichtung **1821**, um die codierten TFCI-Symbole von dem Demultiplexer **920** zu dem ersten Multiplizierer **1815** und dem zweiten Multiplizierer **1817** zu geben. Weiterhin steuert die Steuereinrichtung **1820** den Maskenerzeuger **1830**, um Masken zum Trennen der ersten codierten TFCI-Symbole und der zweiten codierten TFCI-Symbole zu erzeugen. Die durch den Maskenerzeuger **1830** erzeugten Masken werden zu dem ersten Multiplizierer **1815** und zu dem zweiten Multiplizierer **1817** gegeben. Der erste Multiplizierer **1815** multipliziert die codierten TFCI-Symbole aus der Speichereinrichtung **1821** mit der entsprechenden Maske und gibt die ersten codierten TFCI-Symbole aus. Der zweite Multiplizierer **1817** multipliziert die codierten TFCI-Symbole aus der Speichereinrichtung **1821** mit der entsprechenden Maske und gibt die zweiten codierten TFCI-Symbole aus. Der Maskenerzeuger **1830** speichert entweder das Symbolanordnungsmuster der ersten codierten TFCI-Signale und der zweiten codierten TFCI-Signale von Tabelle 6 in der Form einer Maske oder erzeugt die Masken unter Verwendung der Gleichungen (1) und (2). Die Masken werden verwendet, um die codierten TFCI-Symbole aus dem Demultiplexer **920** in die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole zu trennen. Wenn sowohl der Multiplizierer **1815** als auch der Multiplizierer **1817** die zwei Typen von codierten TFCI-Symbolen ausgeben können, wird nur einer der zwei Multiplizierer verwendet, um die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole zu trennen.

[0141] [Fig. 11](#) zeigt den detaillierten Aufbau der Decodierer **900** und **905** von [Fig. 9](#). Wie in [Fig. 11](#) gezeigt, werden empfangene Symbole  $r(t)$  zu einem Null-Einfüger **1100** gegeben, wobei gleichzeitig Codelängeninformation zu einer Steuereinrichtung **1130** gegeben wird. Die Steuereinrichtung **1130** bestimmt Aussonderungspositionen auf der Basis der Codelängeninformation und gibt die Steuerinformation für die bestimmten Aussonderungspositionen an den Null-Einfüger **1100**. Die Codelängeninformation gibt die Codelänge oder die im Codierer verwendete Codierrate an, während die Steuerinformation die Aussonderungspositionen angibt. Die Aussonderungspositionen geben die Positionen der abgeschnittenen Symbole an, um eine gewünschte Codiertsymbollänge in Entsprechung zu den aus dem Codierer empfangenen Bits zu erhalten. Die Tabelle 7 gibt die in Assoziation mit den Codierlängen gespeicherten Aussonderungspositionen an.

Tabelle 7

Codelängeninfo (Codierrate)	Aussonderungsposition
(3,1)	F_29
(4,1)	F_28
(5,1)	F_27
(6,2)	F_26
(7,2)	F_25
(8,2)	F_24
(9,3)	F_23
(10,3)	F_22
(11,9)	F_21
(12,4)	F_20
(13,4)	F_19
(14,4)	F_18
(18,6)	F_14
(19,6)	F_13
(20,6)	F_12
(21,7)	F_11
(22,7)	F_10
(23,7)	F_9
(24,8)	F_8
(25,8)	F_7
(26,8)	F_6
(27,9)	F_5
(28,9)	F_4
(29,9)	F_3

**[0142]** Es wird in Tabelle 7 angenommen, dass die Codelängeninformation die im Codierer verwendete Codierrate angibt. Da eine Codierrate (k,n) angibt, dass n Eingangsbits zu k Symbolen codiert werden, weisen die empfangenen Symbole eine Codierlänge k auf. Weiterhin gibt F<sub>n</sub> in Tabelle 7 n Aussonderungspositionen an. Wie aus der Tabelle 11 bestimmt werden kann, ermöglicht die Steuerinformation (Aussonderungsposition) es dem Null-Einfüger **1100**, die Anzahl (32) der Ausgangssymbole bei einer beliebigen Codierlänge der empfangenen Symbole aufrechtzuerhalten.

**[0143]** Wie in [Fig. 7](#) gezeigt, gibt die Steuereinrichtung **1130** Information zu 29 Aussonderungspositionen für eine Codierrate (3,1), Information zu 28 Aussonderungspositionen für eine Codierrate (4,1), Information zu 27 Aussonderungspositionen für eine Codierrate (5,1), Information zu 26 Aussonderungspositionen für eine Codierrate (6,2), Information zu 25 Aussonderungspositionen für eine Codierrate (7,2), Information zu 24 Aussonderungspositionen für eine Codierrate (8,2), Information zu 23 Aussonderungspositionen für eine Codierrate

(9,3), Information zu 22 Aussonderungspositionen für eine Codierrate (10,3), Information zu 21 Aussonderungspositionen für eine Codierrate (11,9), Information zu 20 Aussonderungspositionen für eine Codierrate (12,4), Information zu 19 Aussonderungspositionen für eine Codierrate (13,4), Information zu 18 Aussonderungspositionen für eine Codierrate (14,4), Information zu 14 Aussonderungspositionen für eine Codierrate (18,6), Information zu 13 Aussonderungspositionen für eine Codierrate (19,6), Information zu 12 Aussonderungspositionen für eine Codierrate (20,6), Information zu 11 Aussonderungspositionen für eine Codierrate (21,7), Information zu 10 Aussonderungspositionen für eine Codierrate (22,7), Information zu 9 Aussonderungspositionen für eine Codierrate (23,7), Information zu 8 Aussonderungspositionen für eine Codierrate (24,8), Information zu 7 Aussonderungspositionen für eine Codierrate (25,8), Information zu 6 Aussonderungspositionen für eine Codierrate (26,8), Information zu 5 Aussonderungspositionen für eine Codierrate (27,9), Information zu 4 Aussonderungspositionen für eine Codierrate (28,9) und Information zu 3 Aussonderungspositionen für eine Codierrate (29,9) aus. Für die jeweiligen Fälle entsprechen die Aussonderungspositionen den bei der Beschreibung der Codierer angegebenen.

**[0144]** Der Null-Einfüger **1100** fügt die Nullen in den Aussonderungspositionen der empfangenen Symbole in Übereinstimmung mit der Steuerinformation ein und gibt dann einen Symbolstrom mit der Länge 32 aus. Der Symbolstrom wird zu einem Invers-Schnell-Hadamard-Transformierer (IFHT) **1120** und den Multiplizierern **1102**, **1104** und **1106** ausgegeben. Der zu den Multiplizierern **1102**, **1104** und **1106** ausgegebene Symbolstrom wird mit den Maskenfunktionen M1, M2 und M15 multipliziert, die jeweils durch den Maskenerzeuger **1110** erzeugt werden. Die Ausgangssymbole der Multiplizierer **1102**, **1104** und **1106** werden jeweils zu den Schaltern **1152**, **1154** und **1156** gegeben. Zu diesem Zeitpunkt gibt die Steuereinrichtung **1130** an die Schalter **1152**, **1154** und **1156** die Schaltsteuerinformation aus, die die Verwendung/Nichtverwendung der Maskenfunktionen auf der Basis der Codelängeninformation angibt. Da beispielsweise die (3,1)-, (4,1)-, (5,1)-, (6,2)-, (7,2)-, (8,2)-, (9,3)-, (10,3)-, (11,3)-, (12,4)-, (13,4)-, (14,4)-, (18,6)-, (19,6)- und (20,6)-Codierer die Maskenfunktionen nicht verwenden, werden die Schalter **1152**, **1154** und **1156** alle in Übereinstimmung mit der Schaltersteuerinformation getrennt. Da jedoch die (21,7)-, (22,7)- und (23,7)-Codierer eine Grundmaskenfunktion verwenden, wird nur der Schalter **1152** verbunden. Auf diese Weise steuert die Steuereinrichtung **1130** die Schalter **1152**, **1154** und **1156** in Übereinstimmung mit der Anzahl der verwendeten Maskenfunktionen auf der Basis der Codierrate. Dann führen die IFHTs **1120**, **1122**, **1124** und **1126** jeweils eine inverse, schnelle Hadamard-Transformation auf den vom Null-Einfüger **1100** empfangenen 32 Symbolen durch und berechnen die Korrelationen zwischen den Symbolen und allen Walsh-Codes, die in dem Sender verwendet werden können. Weiterhin bestimmen die IFHTs die höchste Korrelation unter den Korrelationen und einen Index des Walsh-Codes mit der höchsten Korrelation. Deshalb geben die IFHTs **1120**, **1122**, **1124** und **1126** an den Korrelationsvergleicher **1140** jeweils einen Index der Maskenfunktion, multipliziert mit dem empfangenen Signal, die höchste Korrelation und einen Index des Walsh-Codes mit der höchsten Korrelation aus. Weil das an den IFHT **1120** ausgegebene Signal mit keiner der Maskenfunktionen multipliziert wird, wird eine Kennzeichnung der Maskenfunktion gleich '0'. Der Korrelationsvergleicher **1140** bestimmt die höchste Korrelation, indem er die aus den IFHTs ausgegebenen Korrelationen vergleicht, und kombiniert einen Index einer Maskenfunktion mit der höchsten Korrelation mit einem Index des Walsh-Codes.

## 7. Zweite Ausführungsform des Empfängers

**[0145]** **Fig. 10** zeigt den Aufbau eines Empfängers in Entsprechung zu den Sendern von **Fig. 3** und **Fig. 4** in Übereinstimmung mit einer anderen Ausführungsform der vorliegenden Erfindung. Wie in **Fig. 10** gezeigt, wird ein Abwärtsverbindungs-DPCH zu einem Entschlüsseler **1040** gegeben, wobei gleichzeitig ein durch einen Verschlüsselungscodeerzeuger **1045** erzeugter Verschlüsselungscode zu dem Entschlüsseler **1040** gegeben wird. Der Entschlüsseler **1040** entschlüsselt den Abwärtsverbindungs-DPCH mit dem Verschlüsselungscode. Der entschlüsselte Abwärtsverbindungs-DPCH wird zu einem Entspreizer **1030** gegeben, und gleichzeitig wird ein durch einen Spreizcodeerzeuger **1035** erzeugter Spreizcode zu dem Entspreizer **1030** gegeben. Der Entspreizer **1030** entspreizt den entschlüsselten Abwärtsverbindungs-DPCH durch den Spreizcode in einer Symboleinheit.

**[0146]** Die entspreizten DPCH-Symbole werden zu einem Demultiplexer **1020** gegeben, wo sie in codierte TFCI-Symbole und andere Signale wie etwa die DPDCH-, TPC- und Pilotbits gedemultiplext (getrennt) werden. Die codierten TFCI-Symbole werden zu einem Codiertsymbol-Neuanordner **1010** gegeben. Der Codiertsymbol-Neuanordner **1010** trennt die codierten TFCI-Symbole in die codierten Symbole für den DSCH (zweite TFCI-Informationssymbole) und die codierten Symbole für den DCH (erste TFCI-Informationssymbole) in Übereinstimmung mit Codelängeninformation und Positionsinformation. Die Codelängeninformation ist Codelängensteuerinformation auf der Basis eines Verhältnisses der TFCI-Bits für den DSCH zu den TFCI-Bits für den DCH. Die Positionsinformation ist Information, die die Positionen der codierten Symbole für den DSCH und

die Positionen der codierten Symbole für den DCH wie in Tabelle 6 gezeigt angibt.

**[0147]** Der Codiertsymbol-Neuanordner **1010** weist den Aufbau von [Fig. 18A](#) oder [Fig. 18B](#) auf. Wenn einer der Aufbauten von [Fig. 18A](#) oder [Fig. 18B](#) verwendet wird, sollte der Codiertsymbol-Neuanordner **1010** separat und sequentiell die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole ausgeben. Die separierten zweiten codierten TFCI-Symbole und ersten codierten TFCI-Symbole werden sequentiell zu einem Decodierer **1000** ausgegeben. Der Decodierer **1000** decodiert die ersten codierten TFCI-Symbole oder die zweiten codierten TFCI-Symbole mit einem Code in Übereinstimmung mit der Steuerinformation (Codelängeninformation) für die Codelänge. Deshalb gibt der Decodierer **1000** die ersten TFCI-Bits oder die zweiten TFCI-Bits aus. Der Decodierer **1000** führt dieselbe Operation wie der in [Fig. 11](#) gezeigte Decodierer durch.

**[0148]** Weiterhin sieht die vorliegende Erfindung einen Decodierer vor, der die Decodierung für die entsprechenden Informationsbitverhältnisse durchführt, die dem Codierer für die Codierung der Codes mit verschiedenen Längen entsprechen.

**[0149]** Im Folgenden wird eine Operation des Decodierers gemäß einer Ausführungsform der vorliegenden Erfindung im Detail beschrieben. Wenn der Decodierer als ein Decodierer dient, der den (6,2)-, (7,2)- und (8,2)-Codierern entspricht, verwendet der Decodierer die IFHTs für einen Walsh-Codierer mit einer Länge von 4. Wenn er als ein Decodierer dient, der den (9,3)-, (10,3)- und (11,3)-Codierern entspricht, verwendet er die IFHTs für einen Walsh-Codierer mit einer Länge von 8. Wenn er als ein Decodierer dient, der den (12,4)-, (13,4)- und (14,4)-Codierern entspricht, verwendet er die IFHTs für einen Walsh-Codierer mit einer Länge von 16. Wenn er als ein Decodierer dient, der den (18,6)-, (19,6)-, (20,6)-, (21,7)-, (22,7)-, (23,7)-, (24,8)-, (25,8)-, (27,9)-, (28,9)-, (29,9)- und (32,10)-Codierern entspricht, verwendet er die IFHTs für einen Walsh-Codierer mit einer Länge von 32. Für diese Operation sollte der Decodierer einen IFHT-Aufbau aufweisen, der Codes mit einer variablen Länge unterstützt. Deshalb gibt die vorliegende Erfindung einen Decodierer mit einem IFHT-Aufbau an, der Codes mit einer variablen Länge unterstützt.

## 8. Operationen der Ausführungsformen

**[0150]** Im Folgenden werden die Operationen des Codierers, des Decodierers, des Symbol-Anordners und des Symbol-Neuanordners mit Bezug auf [Fig. 16](#) und [Fig. 17](#) beschrieben.

**[0151]** [Fig. 16](#) zeigt eine Operation des Codierers und des Codiertsymbol-Anordners in dem Sender gemäß einer Ausführungsform der vorliegenden Erfindung. Wie in [Fig. 16](#) gezeigt, entscheidet der Sender in [Fig. 1601](#), die ersten TFCI-Bits (TFCI-Informationbits für den DCH) und die zweiten TFCI-Bits (TFCI-Informationbits für den DSCH) in dem HSM (Hard split-Modus) zu codieren. In Schritt **1602** empfängt der Codierer die ersten codierten TFCI-Bits und die zweiten codierten TFCI-Bits. In Schritt **1603** codiert der Codierer die ersten codierten TFCI-Bits (32 codierte Symbole) und die zweiten codierten TFCI-Bits (32 codierte Symbole) in dem Verfahren gemäß der vorliegenden Erfindung. In Schritt **1604** wählt der Codiertsymbol-Anordner die codierten Symbole mit einer optimalen Leistung aus den ersten codierten TFCI-Symbolen gemäß einem Codeauswahlmuster und wählt weiterhin die codierten Symbole mit einer optimalen Leistung aus den zweiten codierten TFCI-Symbolen gemäß einem Codeauswahlmuster. Die Codeauswahlmuster sind gleich den in [Fig. 5](#) gezeigten Aussonderungsmustern. In Schritt **1605** ordnet der Codiertsymbol-Anordner die ausgewählten ersten codierten TFCI-Symbole und zweiten codierten TFCI-Symbole gemäß einem Symbolanordnungsmuster aus, um eine optimale Zeitdiversitätsverstärkung zu erhalten. Das Symbolanordnungsmuster ist in Tabelle 6 gezeigt. Wie in Verbindung mit [Fig. 15](#) beschrieben, können die Operationen der Schritte **1603**, **1604** und **1605** in nur einem Prozess ausgeführt werden. Nach dem Schritt **1605** werden die  $b_i$  Bits schließlich in Schritt **1606** bestimmt, womit der Codierungs- und Symbolanordnungsprozess abgeschlossen wird.

**[0152]** [Fig. 17](#) zeigt eine Operation des Decodierers und des Codiertsymbol-Neuanordners in dem Empfänger gemäß einer Ausführungsform der vorliegenden Erfindung. Wie in [Fig. 17](#) gezeigt, empfängt der Empfänger in Schritt **1701** die codierten TFCI-Symbole, die über das TFCI-Feld in dem Abwärtsverbindungs-DPCCH des Abwärtsverbindungs-DPCH übertragen werden. In Schritt **1702** fügt der Decodierer die Nullen in den Positionen der zweiten codierten TFCI-Symbole aus den empfangenen codierten TFCI-Symbolen in Übereinstimmung mit Positionsinformation der zweiten codierten TFCI-Symbole ein und erzeugt ein erstes TFCI-Codewort mit 32 codierten Symbolen. Weiterhin fügt der Decodierer Nullen in den Positionen der ersten codierten TFCI-Symbole aus den empfangenen codierten TFCI-Symbolen in Übereinstimmung mit Positionsinformation der ersten codierten TFCI-Symbole ein und erzeugt ein zweites TFCI-Codewort mit 32 codierten Symbolen. Wie in Verbindung mit [Fig. 18A](#) und [Fig. 18B](#) beschrieben, können die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole unter Verwendung der Maske getrennt werden. Die Positionsinformation der

ersten codierten TFCI-Symbole und der zweiten codierten TFCI-Symbole ist gleich dem in Schritt **1604** von **Fig. 16** verwendeten Muster. Der Grund für das Einfügen der Nullen an den Positionen der nicht-ausgesonderten oder nicht-ausgewählten Symbole besteht darin, eine korrekte Operation des Decodierers sicherzustellen. In Schritt **1703** berechnet der Decodierer die Korrelationen des erzeugten ersten TFCI-Codeworts und zweiten TFCI-Codeworts. In Schritt **1704** gibt der Decodierer Werte oder Indizes des ersten TFCI-Codeworts und des zweiten TFCI-Codeworts mit maximaler Korrelation aus. In Schritt **1705** beendet der Decodierer den Prozess zum Decodieren des ersten TFCI-Codeworts und des zweiten TFCI-Codeworts.

**[0153]** Die vorstehende Beschreibung hat auf das Decodierverfahren, das Verfahren zum Abbilden der  $c_x^1$  und  $c_y^2$  Bits auf die  $b_i$  Bits und das Verfahren zum Abbilden der  $b_i$  Bits auf die  $d_m$  Bits für den Fall Bezug genommen, dass die Summe der Anzahl der ersten TFCI-Informationsbits und der Anzahl der zweiten TFCI-Informationsbits in dem HSM gleich 10 wird. Weiterhin wurde eine Beschreibung des Sendeempfängers, des Codierers und des Decodierers gegeben. Wenn herkömmlicherweise die Summe der Anzahl der ersten TFCI-Informationsbits und der Anzahl der zweiten TFCI-Informationsbits kleiner als 10 ist, ist der LSM verfügbar, während der HSM nicht verfügbar ist. Das heißt, der HSM ist nur verfügbar, wenn die Anzahl der ersten TFCI-Informationsbits und die Anzahl der zweiten TFCI-Informationsbits beide kleiner als 5 sind. Herkömmlicherweise wird nur der (16,5)-Codierer in dem HSM verwendet. Deshalb ist der HSM nicht verfügbar, wenn die Anzahl der ersten TFCI-Informationsbits größer als 5 ist oder die Anzahl der zweiten TFCI-Informationsbits größer als 5 ist. Wenn jedoch der neuartige Codierer gemäß der vorliegenden Erfindung verwendet wird, der 24 Arten von Codes erzeugen kann, ist keine Einschränkung bezüglich der Anzahl der TFCI-Informationsbits gegeben, so dass die TFCI-Informationsbits zuverlässig übertragen werden können. Das heißt, es können die Codes bestimmt werden, mit denen die TFCI-Informationsbits zu codieren sind. Es ist dementsprechend möglich, den ersten TFCI-Code oder den zweiten TFCI-Code separat zu übertragen oder den ersten TFCI-Code und den zweiten TFCI-Code gleichzeitig zu übertragen, um eine zuverlässige Übertragung sicherzustellen.

**[0154]** Im Folgenden wird eine ausführliche Beschreibung der Erfindung unter der Annahme gegeben, dass der Codierer den Aufbau von **Fig. 4** aufweist und das Aussonderungsmuster der Tabelle 4 verwendet. Die Erfindung kann auch auf einen anderen Fall angewendet werden, in dem der Codierer einen anderen Aufbau aufweist und ein anderes Aussonderungsmuster verwendet.

Beispiel 1. Verhältnis der ersten TFCI-Informationsbits zu den zweiten TFCI-Informationsbits ist gleich 2:6

**[0155]** Wenn das Verhältnis der ersten TFCI-Informationsbits zu den zweiten TFCI-Informationsbits gleich 2:6 ist, kann das herkömmliche HSM-Verfahren die ersten TFCI-Informationsbits vor der Übertragung codieren, kann aber nicht die zweiten TFCI-Informationsbits übertragen. Wenn jedoch der Codierer gemäß der vorliegenden Erfindung verwendet wird, werden die ersten TFCI-Informationsbits zu 6 Symbolen, 7 Symbolen oder 8 Symbolen codiert und werden die zweiten TFCI-Informationsbits zu 18 Symbolen, 19 Symbolen oder 20 Symbolen codiert. Die Summe der ersten codierten TFCI-Symbole und der zweiten codierten TFCI-Symbole, die durch den Codierer gemäß der vorliegenden Erfindung codiert werden, ist mindestens gleich 24 und höchstens gleich 28. Wenn die Summe kleiner als 32 ist, was der Grundanzahl der codierten Symbole entspricht, besteht das einfachste Verfahren zur Verarbeitung der Symbole darin, nur die 24 Symbole oder die 28 Symbole mittels einer diskontinuierlichen Übertragung (DTX) zu übertragen. Dieses Verfahren trägt zu einer Vereinfachung bei, kann aber keine andere Information in der DTX-Periode übertragen, was eine Verschwendung von Ressourcen darstellt. Außerdem kann die Codierleistung der ersten TFCI-Informationsbits und der zweiten TFCI-Informationsbits aufgrund der diskontinuierlichen Übertragung der codierten Symbole nicht erhöht werden.

**[0156]** In Beispiel 1 kann das Codierverfahren verändert werden, indem dem ersten TFCI Priorität gegeben wird, um die Zuverlässigkeit oder Leistung zu erhöhen, indem dem zweiten TFCI Priorität gegeben wird, um die Zuverlässigkeit oder Leistung zu erhöhen, oder indem die Leistung des ersten TFCI und des zweiten TFCI erhöht wird.

**[0157]** Wenn dem ersten TFCI Priorität gegeben wird, um die Zuverlässigkeit oder Leistung zu erhöhen, werden die zweiten TFCI-Informationsbits unter Verwendung des (18,6)-Codierers, des (19,6)-Codierers oder des (20,6)-Codierers codiert und werden die ersten TFCI-Informationsbits unter Verwendung des (14,4)-Codierers, des (13,4)-Codierers oder des (12,4)-Codierers codiert. Weiterhin besteht ein weiteres Verfahren zum Codieren der ersten TFCI-Informationsbits durch den (6,2)-Codierer, (7,2)-Codierer oder (8,2)-Codierer, wobei dann wiederholt die ersten codierten TFCI-Bits übertragen werden, um dadurch die Zuverlässigkeit oder Leistung zu erhöhen. In dem Verfahren zum Erhöhen der Leistung oder Zuverlässigkeit des ersten TFCI-Codes durch die Codierung der ersten TFCI-Informationsbits unter Verwendung des (14,4)-Codierers, des (13,4)-Codierers

oder des (12,4)-Codierers werden Nullen in den zwei Bits mit Ausnahme der zwei tatsächlichen Informationsbits eingesetzt, bevor diese codiert werden. Nach der Wiederholung des ersten TFCI kann die Summe der wiederholten ersten codierten TFCI-Symbole und zweiten codierten TFCI-Symbole größer als 32 sein. Wenn die Summe der ersten codierten TFCI-Symbole und zweiten codierten TFCI-Symbole größer als 32 ist, ist das System nicht mit dem 3GPP-Standard kompatibel, wodurch die Komplexität der Hardware erhöht wird. Wenn dagegen die Summe der ersten TFCI-Informationsbits und der zweiten TFCI-Informationsbits wie in Beispiel 1 kleiner als 32 ist, ist eine geringere Beschränkung in Bezug auf die Codeauswahl gegeben als in dem Fall, wenn die Summe der ersten TFCI-Informationsbits und zweiten TFCI-Informationsbits gleich 10 ist. Das heißt, wenn die Summe der ersten TFCI-Informationsbits und der zweiten TFCI-Informationsbits gleich 10 ist, ist es nicht erforderlich, die Codes auszuwählen, in denen die Summe der codierten Symbole gleich 32 ist. Wenn jedoch die Summe der codierten Symbole kleiner als 32 ist, obwohl die maximale Codierrate für die in Beispiel 1 gegebenen Informationsbits verwendet wird, kann die Codierrate der TFCI-Informationsbits derart bestimmt werden, dass die Leistung unter der Bedingung verbessert wird, dass die Summe der codierten Symbole kleiner als 32 wird.

**[0158]** Wenn dagegen in Beispiel 1 dem zweiten TFCI Priorität gegeben wird, um die Zuverlässigkeit oder Leistung zu erhöhen, werden die ersten TFCI-Informationsbits unter Verwendung des (6,2)-Codierers, (7,2)-Codierers oder (8,2)-Codierers codiert und werden die zweiten TFCI-Informationsbits unter Verwendung des (28,8)-Codierers, (25,8)-Codierers oder (24,8)-Codierers codiert. Alternativ hierzu können die Informationsbits unter Verwendung des (20,6)-Codierers, (19,6)-Codierers oder (18,6)-Codierers codiert werden, wobei die codierten Bits dann wiederholt übertragen werden, um die Zuverlässigkeit oder Leistung zu erhöhen. Nach der Wiederholung des zweiten TFCI, kann die Summe der ersten codierten TFCI-Symbole und der wiederholten zweiten codierten TFCI-Symbole größer als 32 sein. Wenn jedoch die Summe der ersten codierten TFCI-Symbole und der zweiten codierten TFCI-Symbole größer als 32 ist, ist das System nicht mit dem 3GPP-Standard kompatibel.

**[0159]** Ein Verfahren zum Erhöhen der Zuverlässigkeit oder Leistung des ersten TFCI und des zweiten TFCI in Beispiel 1 besteht darin, vor der Codierung die Anzahl der ersten TFCI-Informationsbits auf 3 zu erhöhen und die Anzahl der zweiten TFCI-Informationsbits auf 7 zu erhöhen. Das heißt, die ersten TFCI-Informationsbits werden durch den (9,3)-Codierer, den (10,3)-Codierer oder den (11,3)-Codierer codiert und die zweiten TFCI-Informationsbits werden durch den (23,7)-Codierer, den (22,7)-Codierer oder den (21,7)-Codierer codiert, bevor sie übertragen werden. Dieses Verfahren kann nur dann verwendet werden, wenn die Summe der codierten Symbole nicht größer als 32 ist. Wenn die Summe der codierten Symbole größer als 32 ist, tritt das oben genannte Problem auf. Ein weiteres Verfahren besteht darin, die ersten TFCI-Informationsbits durch den (6,2)-Codierer, den (7,2)-Codierer oder den (8,2)-Codierer zu codieren, die zweiten TFCI-Informationsbits durch den (18,6)-Codierer, den (19,6)-Codierer oder den (20,6)-Codierer zu codieren, wobei die codierten Bits dann wiederholt übertragen werden. Die Summe der wiederholt übertragenen, codierten Symbole sollte nicht größer als 32 sein. Es gibt drei Arten von Codierern für die Codierung der ersten TFCI-Informationsbits, und es gibt auch drei Arten von Codierern für die Codierung der zweiten TFCI-Informationsbits. Von den Codierern wird der Codierer mit der besten Leistung ausgewählt. Was die Anzahl der Symbole betrifft, die durch die Codierer wiederholt werden, werden die Symbole des ausgewählten Codierers viele weitere Male wiederholt übertragen.

Beispiel 2. Verhältnis der ersten TFCI-Informationsbits zu den zweiten TFCI-Informationsbits ist gleich 3:4

**[0160]** Wenn das Verhältnis der ersten TFCI-Informationsbits zu den zweiten TFCI-Informationsbits gleich 3:4 ist, d.h. wenn die Anzahl der ersten TFCI-Informationsbits und die Anzahl der zweiten TFCI-Informationsbits beide kleiner als 5 sind, nimmt das herkömmliche HSM-Verfahren vor der Übertragung eine (16,5)-Codierung der ersten TFCI-Informationsbits und der zweiten TFCI-Informationsbits separat oder sequentiell vor. Wenn jedoch der Codierer gemäß der vorliegenden Erfindung verwendet wird, werden die ersten TFCI-Informationsbits zu 9 Symbolen, 10 Symbolen oder 11 Symbolen codiert und werden die zweiten TFCI-Informationsbits zu 12 Symbolen, 13 Symbolen oder 14 Symbolen codiert. Die Summe der ersten codierten TFCI-Symbole und der zweiten codierten TFCI-Symbole, die durch den Codierer gemäß der vorliegenden Erfindung codiert werden, ist maximal 25. Wenn die Summe kleiner als 32 ist, was der Grundanzahl der codierten Symbole entspricht, besteht das einfachste Verfahren zur Verarbeitung der Symbole darin, nur die 21 Symbole oder die 24 Symbole mittels einer diskontinuierlichen Übertragung (DTX) zu übertragen. Dieses Verfahren trägt zu einer Vereinfachung bei, kann aber keine anderen Informationen in der DTX-Periode übertragen, was eine Verschwendung von Ressourcen darstellt. Außerdem kann die Codierleistung der ersten TFCI-Informationsbits und der zweiten TFCI-Informationsbits aufgrund der diskontinuierlichen Übertragung der codierten Symbole nicht erhöht werden.

**[0161]** In dem Beispiel 2 kann das Codierverfahren geändert werden, indem dem ersten TFCI Priorität gegeben wird, um die Zuverlässigkeit oder Leistung zu erhöhen, indem dem zweiten TFCI Priorität gegeben wird, um die Zuverlässigkeit oder Leistung zu erhöhen, oder indem die Leistung des ersten TFCI und des zweiten TFCI erhöht wird.

**[0162]** Wenn die Priorität dem ersten TFCI gegeben wird, um die Zuverlässigkeit oder Leistung zu erhöhen, werden die zweiten TFCI-Informationsbits unter Verwendung des (12,4)-Codierers, des (13,4)-Codierers oder des (14,4)-Codierers codiert und werden die ersten TFCI-Informationsbits unter Verwendung des (20,6)-Codierers, des (19,6)-Codierers oder des (18,6)-Codierers codiert. Weiterhin gibt es ein anderes Verfahren zum Codieren der ersten TFCI-Informationsbits durch den (9,3)-Codierer, den (10,3)-Codierer oder den (11,3)-Codierer, wobei dann wiederholt die ersten codierten TFCI-Bits übertragen werden, um die Zuverlässigkeit oder Leistung zu erhöhen. In dem Verfahren zum Erhöhen der Leistung oder Zuverlässigkeit des ersten TFCI-Codes durch das Codieren der ersten TFCI-Informationsbits unter Verwendung des (20,6)-Codierers, des (19,6)-Codierers oder des (18,6)-Codierers werden Nullen in die drei Bits mit Ausnahme der drei tatsächlichen Informationsbits eingesetzt, bevor diese codiert werden. Nach der Wiederholung des ersten TFCI kann die Summe der wiederholten ersten codierten TFCI-Symbole und zweiten codierten TFCI-Symbole größer als 32 sein. Wenn die Summe der ersten codierten TFCI-Symbole und zweiten codierten TFCI-Symbole größer als 32 ist, ist das System nicht mit dem 3GPP-Standard kompatibel, wodurch die Komplexität der Hardware erhöht wird. Wenn dagegen die Summe der ersten TFCI-Informationsbits und der zweiten TFCI-Informationsbits wie in Beispiel 2 kleiner als 32 ist, ist eine geringere Beschränkung in Bezug auf die Codeauswahl gegeben als in dem Fall, wenn die Summe der ersten TFCI-Informationsbits und zweiten TFCI-Informationsbits gleich 10 ist. Das heißt, wenn die Summe der ersten TFCI-Informationsbits und der zweiten TFCI-Informationsbits gleich 10 ist, ist es erforderlich, die Codes auszuwählen, in denen die Summe der codierten Symbole gleich 32 ist. Wenn jedoch die Summe der codierten Symbole kleiner als 32 ist, obwohl die maximale Codierrate für die in Beispiel 2 gegebenen Informationsbits verwendet wird, kann die Codierrate der TFCI-Informationsbits derart bestimmt werden, dass die Leistung unter der Bedingung verbessert wird, dass die Summe der codierten Symbole kleiner als 32 wird.

**[0163]** Wenn dagegen in Beispiel 2 dem zweiten TFCI Priorität gegeben wird, um die Zuverlässigkeit oder Leistung zu erhöhen, werden die ersten TFCI-Informationsbits unter Verwendung des (9,3)-Codierers, (10,3)-Codierers oder (11,3)-Codierers codiert und werden die zweiten TFCI-Informationsbits unter Verwendung des (23,7)-Codierers, (22,7)-Codierers oder (21,7)-Codierers codiert. Alternativ hierzu können die Informationsbits unter Verwendung des (14,4)-Codierers, (13,4)-Codierers oder (12,4)-Codierers codiert werden, wobei die codierten Bits dann wiederholt übertragen werden, um die Zuverlässigkeit oder Leistung zu erhöhen. Nach der Wiederholung des zweiten TFCI, kann die Summe der ersten codierten TFCI-Symbole und der wiederholten zweiten codierten TFCI-Symbole größer als 32 sein. Wenn jedoch die Summe der ersten codierten TFCI-Symbole und der zweiten codierten TFCI-Symbole größer als 32 ist, ist das System nicht mit dem 3GPP-Standard kompatibel.

**[0164]** Schließlich besteht ein Verfahren zum Erhöhen der Zuverlässigkeit des ersten TFCI und des zweiten TFCI in Beispiel 2 darin, die ersten TFCI-Informationsbits und die zweiten TFCI-Informationsbits in ihrer Anzahl zu erhöhen, so dass die Summe der ersten TFCI-Informationsbits und der zweiten TFCI-Informationsbits gleich 10 wird, wobei ein Codierer verwendet wird, der der erhöhen Anzahl von Informationsbits entspricht. Zum Beispiel kann ein Verfahren verwendet werden, um vor der Übertragung die ersten TFCI-Informationsbits durch den (14,4)-Codierer, den (13,4)-Codierer oder den (12,4)-Codierer und die zweiten TFCI-Informationsbits durch den (18,6)-Codierer, (19,6)-Codierer oder (20,6)-Codierer zu codieren. Dieses Verfahren kann nur dann verwendet werden, wenn die Summe der ersten TFCI-Informationsbits und der zweiten TFCI-Informationsbits nicht größer als 10 ist; außerdem sollte die Summe der codierten Symbole nicht größer als 32 sein. Wenn die Summe der codierten Symbole größer als 32 ist, tritt das oben genannte Problem auf. Ein weiteres Verfahren besteht darin, die ersten TFCI-Informationsbits durch den (9,3)-Codierer, den (10,3)-Codierer oder den (11,3)-Codierer zu codieren und die zweiten TFCI-Informationsbits durch den (12,4)-Codierer, den (13,4)-Codierer oder den (14,4)-Codierer zu codieren, wobei die codierten Bits dann wiederholt übertragen werden. Die Summe der wiederholt übertragenen, codierten Symbole sollte nicht größer als 32 sein. Es gibt drei Arten von Codierern für die Codierung der ersten TFCI-Informationsbits, und es gibt auch drei Arten von Codierern für die Codierung der zweiten TFCI-Informationsbits. Von den Codierern wird der Codierer mit der besten Leistung ausgewählt. Was die Anzahl der Symbole betrifft, die durch die Codierer wiederholt werden, werden die Symbole des ausgewählten Codierers viel häufiger wiederholt übertragen. Außerdem kann das Verfahren zum Ändern der Codierrate und das Verfahren der wiederholten Übertragung miteinander kombiniert werden, um die ersten TFCI-Informationsbits und die zweiten TFCI-Informationsbits mit hoher Zuverlässigkeit oder Leistung zu übertragen.

**[0165]** Im Folgenden werden Kriterien für das Codeauswahlverfahren in dem HSM, das in Verbindung mit den Beispielen 1 und 2 beschrieben wird, zusammengefasst.

**[0166]** Kriterium 1: Die Anzahl der ersten TFCI-Informationsbits oder zweiten TFCI-Informationsbits ist höher als 5 Bits.

- Wenn die Priorität dem ersten TFCI gegeben wird, fixiert der Sender den zweiten TFCI-Codierer und ändert dann die Codierate des ersten TFCI während der Übertragung oder codiert den ersten TFCI unter Berücksichtigung der Anzahl der tatsächlichen Informationsbits und überträgt die codierten Bits dann wiederholt.
- Wenn die Priorität dem zweiten TFCI gegeben wird, fixiert der Sender den ersten TFCI-Codierer und ändert dann die Codierate des zweiten TFCI während der Übertragung oder codiert den zweiten TFCI unter Berücksichtigung der Anzahl der tatsächlichen Informationsbits und überträgt die codierten Bits dann wiederholt.
- Wenn die Priorität sowohl dem ersten TFCI als auch dem zweiten TFCI gegeben wird, führt der Sender die Codierung durch, indem er die Codieraten des ersten TFCI und des zweiten TFCI unter Berücksichtigung der Anzahl der tatsächlichen Informationsbits ändert und dann codierten Bits wiederholt überträgt. Das Verfahren zum Ändern der Codierate und das Verfahren für die wiederholte Übertragung können kombiniert werden.

**[0167]** Kriterium 2: Die Anzahl der ersten TFCI-Informationsbits oder der zweiten TFCI-Informationsbits ist nicht größer als 5 Bits.

- Der Sender codiert die ersten TFCI-Informationsbits und die zweiten TFCI-Informationsbits unter Verwendung eines (16,5)-Codierers vor der Übertragung.
- Ansonsten wie bei Kriterium 1.

**[0168]** Im Folgenden wird ein Codeauswahlverfahren auf der Basis der oben genannten Kriterien unter Verwendung des Aussonderungsmusters von Tabelle 5 und der Codierate von Tabelle 1 mit Bezug auf [Fig. 12](#) beschrieben.

**[0169]** Wie in [Fig. 12](#) gezeigt, tritt das Erfordernis zur Übertragung des ersten TFCI (der ersten TFCI-Informationsbits) und des zweiten TFCI (der zweiten TFCI-Informationsbits) in Schritt **1201** auf. Das heißt, wenn ein Knoten B erforderlich ist, um den DSCH zu einem Benutzergerät zu übertragen, empfängt ein Sender den TFCI für den DSCH und den TFCI für den DCH. In Schritt **1202** wird bestimmt, ob die Summe aus den ersten Informationsbits und den zweiten Informationsbits gleich 10 ist. Wenn die Summe der ersten Informationsbits und der zweiten Informationsbits gleich 10 ist, bestimmt der Sender in Schritt **1208** einen Code, der für die ersten Informationsbits und die zweiten Informationsbits zu verwenden ist.

**[0170]** Der Codeauswahlprozess von Schritt **1208** wird für den Fall beschrieben, dass das Verhältnis der ersten Informationsbits zu den zweiten Informationsbits gleich 3:7 ist. In diesem Fall ist der Codierer für die ersten Informationsbits ein (9,3)-Codierer, ein (10,3)-Codierer oder ein (11,3)-Codierer und ist der Codierer für die zweiten Informationsbits ein (23,7)-Codierer, ein (22,7)-Codierer oder ein (21,7)-Codierer. Dabei sollte die Summe der codierten Symbole gleich 32 sein. Das Kriterium für die Auswahl der drei Arten von Codieraten in Übereinstimmung mit den Typen der Informationsbits sieht vor, (1) Priorität zu den ersten Informationsbits zu geben, um zwei überzählige Symbole hinzuzufügen, (2) Priorität zu den zweiten Informationsbits zu geben, um zwei überzählige Symbole hinzuzufügen, oder (3) ein überzähliges Symbol zu sowohl den ersten Informationsbits als auch den zweiten Informationsbits hinzuzufügen. Nachdem die Codierate für die Verwendung für die ersten Informationsbits und die zweiten Informationsbits in Schritt **1208** bestimmt wurde, codiert der Sender in Schritt **1209** die ersten Informationsbits und die zweiten Informationsbits mit der bestimmten Codierate. Der Sender multiplext die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole in Schritt **1210**.

**[0171]** Wenn jedoch in Schritt **1202** bestimmt wird, dass die Summe der ersten Informationsbits und der zweiten Informationsbits kleiner als 10 ist, bestimmt der Sender in Schritt **1203**, ob die Anzahl der ersten Informationsbits größer als 5 ist oder ob die Anzahl der zweiten Informationsbits größer als 5 ist. Wenn die Anzahl der ersten Informationsbits größer als 5 ist oder die Anzahl der zweiten Informationsbits größer als 5 ist, schreitet der Sender zu Schritt **1204** fort. Wenn jedoch weder die Anzahl der ersten Informationsbits noch die Anzahl der zweiten Informationsbits größer als 5 ist, schreitet der Sender zu Schritt **1221** fort. In Schritt **1221** bestimmt der Sender, ob er einen (16,5)-Codierer zum Codieren der ersten Informationsbits und der zweiten Informationsbits verwenden soll. Wenn der Sender bestimmt, dass der (16,5)-Codierer nicht verwendet werden soll, schreitet er zu Schritt **1206** fort. Andernfalls bestimmt der Sender, dass der (16,5)-Codierer verwendet werden soll, und schreitet zu Schritt **1209** fort.

**[0172]** In Schritt **1204** bestimmt der Sender, ob DTX für die Übertragung der ersten Informationsbits oder die zweiten Informationsbits verwendet werden soll. Wenn der Sender die Verwendung von DTX bestimmt, schreitet er zu Schritt **1208** fort. Wenn der Sender dagegen bestimmt, dass DTX nicht verwendet wird, schreitet er zu Schritt **1205** fort.

**[0173]** Der Prozess von Schritt **1208** wird für den Fall beschrieben, dass das Verhältnis der ersten Informationsbits zu den zweiten Informationsbits gleich 3:4 ist. In diesem Fall ist der Codierer für die ersten Informationsbits ein ausgewählter (9,3)-Codierer, (10,3)-Codierer oder (11,3)-Codierer und ist der Codierer für die zweiten Informationsbits ein ausgewählter (12,4)-Codierer, (13,4)-Codierer oder (14,4)-Codierer. Wenn DTX in Schritt **1208** verwendet wird und sowohl die Anzahl der ersten Informationsbits als auch die Anzahl der zweiten Informationsbits nicht größer ist als 5, ist keine Beschränkung bezüglich der Auswahl der Codierer gegeben, wobei aber die Summe der codierten Symbole nicht größer als 32 sein sollte.

**[0174]** In Schritt **1205** bestimmt der Sender, ob die Zuverlässigkeit oder Leistung sowohl des ersten TFCI als auch des zweiten TFCI vor der Übertragung erhöht werden soll. Wenn der Sender bestimmt, die Zuverlässigkeit oder Leistung sowohl des ersten TFCI als auch des zweiten TFCI vor der Übertragung zu erhöhen, wählt er in Schritt **1207** das Verfahren zum Erhöhen der Codierrate, das Verfahren für die wiederholte Übertragung oder eine Kombination aus den beiden zuvor genannten Verfahren. In Schritt **1208** bestimmt der Sender den für den ersten TFCI und den zweiten TFCI zu verwendenden Code in Übereinstimmung mit dem in Schritt **1207** gewählten Verfahren. Der Sender codiert in Schritt **1209** die ersten TFCI-Informationsbits und die zweiten TFCI-Informationsbits in dem gewählten Verfahren und multiplext dann in Schritt **1210** die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole. Wenn der Sender in Schritt **1207** bestimmt, die Zuverlässigkeit oder Leistung des ersten TFCI und des zweiten TFCI in dem Verfahren für die wiederholte Übertragung vor der Übertragung zu verbessern, wiederholt der Sender die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole in Schritt **1209** und multiplext diese dann in Schritt **1210**. Alternativ hierzu wiederholt der Sender in Schritt **1210** die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole, die in Schritt **1209** codiert werden.

**[0175]** Wenn der Sender in Schritt **1205** bestimmt, die Zuverlässigkeit oder Leistung von entweder dem ersten TFCI oder dem zweiten TFCI vor der Übertragung zu erhöhen, wählt der Sender in Schritt **1206** den ersten TFCI oder den zweiten TFCI, dem er Priorität gibt. Der Sender gibt dem ersten TFCI Priorität, wenn die ersten TFCI-Informationsbits mit hoher Zuverlässigkeit unabhängig von der Anzahl der Informationsbits übertragen werden sollen. Der Sender gibt dem zweiten TFCI Priorität, wenn die zweiten TFCI-Informationsbits mit hoher Zuverlässigkeit in Vorbereitung für den Fall übertragen werden sollen, dass die Knoten Bs neben dem Knoten B, der den DSCH empfängt, die zweiten TFCI-Informationsbits für den DSCH nicht übertragen können, wenn das Benutzergerät in dem weichen Übergabebereich lokalisiert ist. Weiterhin gibt der Sender dem zweiten TFCI Priorität, wenn die zweiten TFCI-Informationsbits mit hoher Zuverlässigkeit unabhängig von der Anzahl der Informationsbits übertragen werden sollen. Wenn der Sender in Schritt **1206** bestimmt, die Zuverlässigkeit oder Leistung von entweder dem ersten TFCI oder dem zweiten TFCI vor der Übertragung zu erhöhen, bestimmt der Sender in Schritt **1207** ein Verfahren zum Erhöhen der Zuverlässigkeit oder Leistung von entweder dem ersten TFCI oder dem zweiten TFCI vor der Übertragung, indem er das Verfahren zum Erhöhen der Codierrate, das Verfahren für die wiederholte Übertragung oder eine Kombination aus diesen beiden Verfahren nutzt. In Schritt **1208** bestimmt der Sender den für den ersten TFCI und den zweiten TFCI zu verwendenden Code in Übereinstimmung mit dem in Schritt **1207** bestimmten Verfahren. Der Sender codiert in Schritt **1209** die ersten TFCI-Informationsbits und die zweiten TFCI-Informationsbits in dem bestimmten Verfahren und multiplext dann in Schritt **1210** die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole. Wenn der Sender in Schritt **1207** bestimmt, die Zuverlässigkeit oder Leistung von entweder dem ersten TFCI oder dem zweiten TFCI in dem Verfahren zum Erhöhen der Codierrate zu erhöhen, multiplext der Sender in Schritt **1210** die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole, die in Schritt **1209** codiert wurden. Wenn der Sender in Schritt **1207** bestimmt, die Zuverlässigkeit oder Leistung von entweder dem ersten TFCI oder dem zweiten TFCI vor der Übertragung in dem Verfahren zur wiederholten Übertragung zu erhöhen, wiederholt der Sender in Schritt **1209** die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole und multiplext diese dann in Schritt **1210**. Alternativ hierzu wiederholt der Sender in Schritt **1210** die ersten codierten TFCI-Symbole und die zweiten codierten TFCI-Symbole, die in Schritt **1209** codiert wurden.

**[0176]** Wie oben beschrieben, kann die Ausführungsform der vorliegenden Erfindung verschiedene Typen von TFCI-Bits unter Verwendung eines einzigen Codierer/Decodierer-Aufbaus codieren/decodieren. Außerdem multiplext die Ausführungsform die TFCI-Symbole, die in den verschiedenen Codiertechniken codiert werden, derart, dass die TFCI-Symbole gleichmäßig verteilt sein sollten, bevor sie übertragen werden. Für die 10

Eingangsinformationsbits wird die TFCI-Codierung in einem ausgewählten Verhältnis von 1:9, 2:8, 3:7, 4:6, 5:5, 6:4, 7:3, 8:2 und 9:1 in Abhängigkeit von dem Typ und den Eigenschaften der über den DSCH und den DCH übertragenen Daten durchgeführt, um so zu der Flexibilität des HSM beizutragen, das dem LSM in Bezug auf die Signalisierung und Zeitverzögerung überlegen ist. Außerdem codiert der Codierer die TFCI-Bits für den DCH und die TFCI-Bits für den DSCH und speichert dann die codierten TFCI-Symbole für den DCH und die codierten TFCI-Symbole für den DSCH in einer Speichereinrichtung, um eine schnelle Informationsverarbeitung sicherzustellen.

### Patentansprüche

1. Verfahren zum Abbilden von ersten kodierten TFCI (Transportformat-Kombinationsindikator)-Symbolen und zweiten kodierten TFCI-Symbolen auf einen Funkrahmen in einem Mobilkommunikationssystem zum Kodieren von  $k$  ersten TFCI-Bits und  $(10 - k)$  zweiten TFCI-Bits, wobei  $k$  eine variable Integerzahl mit einem Wert in einem Bereich von  $1 \leq k \leq 9$  ist, wobei das Verfahren die folgenden Schritte umfasst:

Multiplexen (**1210**) wenigstens einiger der ersten und zweiten kodierten TFCI-Symbole zum Erzeugen einer kombinierten Ausgabe mit einer durch die variable Integerzahl  $k$  beeinflussten Reihenfolge von Symbolen, und Abbilden aller oder weniger als aller der gemultiplexten, kodierten Symbole auf den Funkrahmen, um eine Anzahl der kodierten Symbole vorzusehen, die auf einen Funkrahmen abgebildet werden können.

2. Verfahren nach Anspruch 1, wobei die ersten kodierten TFCI-Symbole an Positionen gemultiplext werden, die wie folgt berechnet werden:

$$b_{\lfloor \frac{32}{n} \cdot x(i+1) \rfloor - 1} = c_i^1, (0 \leq i \leq n - 1)$$

wobei  $c_i^1$  ein erstes kodiertes TFCI Symbol darstellt,  $b$  ein gemultiplextes kodiertes Symbol darstellt,  $\lfloor \rfloor$  eine Integerzahl anzeigt, die sich durch Abrundung des in der Klammer  $\lfloor \rfloor$  gegebenen Wertes ergibt,  $n$  die Gesamtanzahl der ersten kodierten TFCI-Symbole wiedergibt und  $i$  einen Index wiedergibt, der ein willkürliches kodiertes Symbol aus den ersten kodierten TFCI-Symbolen angibt.

3. Verfahren nach Anspruch 1, wobei die zweiten kodierten TFCI-Symbole an Positionen gemultiplext werden, die wie folgt berechnet werden:

$$b_{\lfloor \frac{n}{32-n} \cdot x(i + \frac{1}{2}) \rfloor} = c_i^2, (0 \leq i \leq m - 1)$$

wobei  $c_i^2$  ein zweites kodiertes TFCI-Symbol darstellt,  $b$  ein gemultiplextes kodiertes Symbol darstellt,  $\lfloor \rfloor$  einen Maximalwert der Integerzahlen angibt, der kleiner oder gleich dem gegebenen Wert in den Klammern  $\lfloor \rfloor$  ist,  $n$  die Gesamtanzahl der ersten kodierten TFCI-Symbole wiedergibt,  $m$  die Gesamtanzahl der zweiten kodierten TFCI-Symbole wiedergibt und  $i$  einen Index wiedergibt, der ein willkürliches kodiertes Symbol aus den ersten kodierten TFCI-Symbolen angibt.

4. Verfahren nach Anspruch 1, wobei wenn die Anzahl der kodierten Symbole, die auf einen Funkrahmen abgebildet werden können, gleich 30 ist, dann werden 32 gemultiplexte kodierte Symbole mit Ausnahme eines willkürlichen der ersten kodierten TFCI-Symbole und eines willkürlichen der zweiten kodierten TFCI-Symbole auf den Funkrahmen abgebildet.

5. Verfahren nach Anspruch 4, wobei das willkürliche der ersten kodierten TFCI-Symbole ein letztes kodiertes Symbol der ersten kodierten TFCI-Symbole ist und wobei das willkürliche der zweiten kodierten TFCI-Symbole ein letztes kodiertes Symbol der zweiten kodierten TFCI-Symbole ist.

6. Verfahren nach Anspruch 1, wobei wenn die Anzahl der kodierten Symbole, die auf einen Funkrahmen abgebildet werden können, gleich 30 ist, dann werden 32 gemultiplexte kodierte Symbole mit Ausnahme von zwei willkürlichen kodierten Symbolen aus den ersten kodierten TFCI-Symbolen oder zwei willkürlichen kodierten Symbolen aus den zweiten kodierten TFCI-Symbolen auf einen Funkrahmen abgebildet.

7. Verfahren nach Anspruch 1, wobei wenn die Anzahl der kodierten Symbole, die auf einen Funkrahmen abgebildet werden können, gleich 120 ist, dann werden 32 gemultiplexte kodierte Symbole drei Mal wiederholt und die ersten 24 kodierten Symbole aus den 32 gemultiplexten kodierten Symbole weiter wiederholt, wobei sie dann auf den Funkrahmen abgebildet werden.

8. Verfahren nach Anspruch 1, wobei wenn die Anzahl der kodierten Symbole, die auf einen Funkrahmen abgebildet werden können, gleich 32 ist, dann werden 32 gemultiplexte kodierte Symbole auf den Funkrahmen abgebildet.

9. Verfahren nach Anspruch 1, wobei wenn die Anzahl der kodierten Symbole, die auf einen Funkrahmen abgebildet werden können, gleich 128 ist, dann werden 32 gemultiplexte kodierte Symbole vier Mal wiederholt und dann auf den Funkrahmen abgebildet.

10. Vorrichtung zum Übertragen von ersten TFCI (Transportformat-Kombinationsindikator)-Bits und zweiten TFCI-Bits über einen Funkrahmen in einem Mobilkommunikationssystem, wobei  $k$  eine variable Integerzahl mit einem Wert in einem Bereich von  $1 \leq k \leq 9$  ist, wobei die Vorrichtung umfasst:  
 wenigstens einen Kodierer (**200, 205, 311, 1303, 1313, 1403**) zum Kodieren von  $k$  ersten TFCI-Bits mit einer ersten Kodiertrate, um  $(3k + 1)$  erste kodierte TFCI-Symbole auszugeben, sowie zum Kodieren von  $(10 - k)$  zweiten TFCI-Bits mit einer zweiten Kodiertrate, um  $(31 - 3k)$  zweite kodierte TFCI-Symbole auszugeben, und einen Kodiertsymbol-Anordner (**210, 312, 910, 1010, 1350, 1450**) zum Multiplexen wenigstens einiger der kodierten Symbole zum Erzeugen einer kombinierten Ausgabe mit einer durch die variable Integerzahl  $k$  beeinflussten Reihenfolge von Symbolen sowie zum Ausgeben aller oder weniger als aller der gemultiplexten kodierten Symbole, um eine Anzahl der kodierten Symbole vorzusehen, die auf einen Funkrahmen abgebildet werden können.

11. Vorrichtung nach Anspruch 10, die weiterhin einen Selektor (**310**) zum Auswählen der ersten TFCI-Bits und der zweiten TFCI-Bits in Übereinstimmung mit einem Wert von  $k$  sowie zum Ausgeben der ausgewählten TFCI-Bits zu dem Kodierer umfasst.

12. Vorrichtung nach Anspruch 10, wobei der Kodiertsymbol-Anordner (**210, 312, 910, 1010, 1350, 1450**) die kodierten Symbole derart multiplext, dass die ersten kodierten TFCI-Symbole an Positionen ausgegeben werden, die wie folgt be

$$b_{\left\lfloor \frac{32}{n} \cdot (i+1) \right\rfloor - 1} = c_i^1, (0 \leq i \leq n - 1)$$

rechnet werden:

wobei  $c_i^1$  ein erstes kodierte TFCI-Symbol darstellt,  $b$  ein gemultiplextes kodierte Symbol darstellt,  $\lfloor \cdot \rfloor$  eine Integerzahl anzeigt, die sich durch Abrundung des in der Klammer  $\lfloor \cdot \rfloor$  gegebenen Wertes ergibt,  $n$  die Gesamtanzahl der ersten kodierten TFCI-Symbole wiedergibt und  $i$  einen Index wiedergibt, der ein willkürliches kodierte Symbol aus den ersten kodierten TFCI-Symbolen angibt.

13. Vorrichtung nach Anspruch 10, wobei der Kodiertsymbol-Anordner (**210, 312, 910, 1010, 1350, 1450**) die kodierten Symbole derart multiplext, dass die zweiten kodierten TFCI-Symbole an Positionen ausgegeben werden, die wie folgt berechnet werden:

$$b_{i + \left\lfloor \frac{n}{32-n} \cdot \left( i + \frac{1}{2} \right) \right\rfloor} = c_i^2, (0 \leq i \leq m - 1)$$

wobei  $c_i^2$  ein zweites kodierte TFCI-Symbol darstellt,  $b$  ein gemultiplextes kodierte Symbol darstellt,  $\lfloor \cdot \rfloor$  einen Maximalwert der Integerzahlen angibt, der kleiner oder gleich dem gegebenen Wert in den Klammern  $\lfloor \cdot \rfloor$  ist,  $n$  die Gesamtanzahl der ersten kodierten TFCI-Symbole wiedergibt,  $m$  die Gesamtanzahl der zweiten kodierten TFCI-Symbole wiedergibt und  $i$  einen Index wiedergibt, der ein willkürliches kodierte Symbol aus den ersten kodierten TFCI-Symbolen angibt.

14. Verfahren zum Übertragen von ersten TFCI (Transportformat-Kombinationsindex) -Bits und zweiten TFCI-Bits über einen Funkrahmen in einem Mobilkommunikationssystem, wobei  $k$  eine variable Integerzahl mit einem Wert in einem Bereich von  $1 \leq k \leq 9$  ist, wobei das Verfahren folgende Schritte umfasst:  
 Kodieren von  $k$  ersten TFCI-Bits zum Ausgeben von  $(3k + 1)$  erste kodierte TFCI-Symbole,  
 Kodieren von  $(10 - k)$  zweiten TFCI-Bits zum Ausgeben von  $(31 - 3k)$  zweite kodierte TFCI-Symbole,  
 Multiplexen (**1210**) wenigstens einiger der ersten und zweiten kodierten TFCI-Symbole zum Erzeugen einer kombinierten Ausgabe mit einer durch die variable Integerzahl  $k$  beeinflussten Reihenfolge von Symbolen, und Ausgeben aller oder weniger als aller der gemultiplexten kodierten Symbole, um eine Anzahl der kodierten Symbole vorzusehen, die auf einem Funkrahmen abgebildet werden können.

15. Verfahren nach Anspruch 14, wobei die ersten kodierten TFCI-Symbole an Positionen ausgegeben

werden, die wie folgt berechnet werden:

$$b_{\lfloor \frac{32-n}{n} \cdot x(i+1) \rfloor - 1} = c_i^1, (0 \leq i \leq n-1)$$

wobei  $c_i^1$  ein erstes kodiertes TFCI-Symbol darstellt,  $b$  ein gemultiplextes kodiertes Symbol darstellt,  $\lfloor \cdot \rfloor$  eine Integerzahl anzeigt, die sich durch Abrundung des in der Klammer  $\lfloor \cdot \rfloor$  gegebenen Wertes ergibt,  $n$  die Gesamtzahl der ersten kodierten TFCI-Symbole wiedergibt und  $i$  einen Index wiedergibt, der ein willkürliches kodiertes Symbol aus den ersten kodierten TFCI-Symbolen angibt.

16. Verfahren nach Anspruch 14, wobei die zweiten kodierten TFCI-Symbole an Positionen ausgegeben werden, die wie folgt berechnet werden:

$$b_{\lfloor \frac{n}{32-n} \cdot x(i+\frac{1}{2}) \rfloor} = c_i^2, (0 \leq i \leq m-1)$$

wobei  $c_i^2$  ein zweites kodiertes TFCI-Symbol darstellt,  $b$  ein gemultiplextes kodiertes Symbol darstellt,  $\lfloor \cdot \rfloor$  einen Maximalwert der Integerzahlen angibt, der kleiner oder gleich dem gegebenen Wert in den Klammern  $\lfloor \cdot \rfloor$  ist,  $n$  die Gesamtanzahl der ersten kodierten TFCI-Symbole wiedergibt,  $m$  die Gesamtanzahl der zweiten kodierten TFCI-Symbole wiedergibt und  $i$  einen Index wiedergibt, der ein willkürliches kodiertes Symbol aus den ersten kodierten TFCI-Symbolen angibt.

17. Vorrichtung zum Dekodieren von  $k$  ersten TFCI-Bits und  $(10 - k)$  zweiten TFCI-Bits in einem Mobilkommunikationssystem zum Empfangen von  $(3k + 1)$  ersten kodierten TFCI-Symbolen und von  $(31 - 3k)$  zweiten kodierten TFCI-Symbolen, wobei  $k$  eine variable Integerzahl mit einem Wert in einem Bereich  $1 \leq k \leq 9$  ist, wobei die Vorrichtung umfasst:

einen Kodiertsymbol-Neuanordner (**910, 1010**) zum Trennen der ersten kodierten TFCI-Symbole und der zweiten kodierten TFCI-Symbole, die über einen DPCH (dedizierten physikalischen Kanal) übertragen werden, in Übereinstimmung mit der variablen Integerzahl, um diese neu anzuordnen, und wenigstens einen Dekodierer zum Dekodieren der ersten kodierten TFCI-Symbole, um die  $k$  ersten TFCI-Bits auszugeben, sowie zum Dekodieren der zweiten kodierten TFCI-Symbole, um die  $(10 - k)$  zweiten TFCI-Bits auszugeben.

18. Vorrichtung nach Anspruch 17, wobei der Kodiertsymbol-Neuanordner (**910, 1010**) die ersten kodierten TFCI-Symbole an Positionen, die durch die folgende Formel berechnet werden, von den kodierten Symbolen trennt, die durch das Multiplexen der ersten kodierten TFCI-Symbole und der zweiten kodierten TFCI-Symbole erhalten werden:

$$b_{\lfloor \frac{32-n}{n} \cdot x(i+1) \rfloor - 1} = c_i^1, (0 \leq i \leq n-1)$$

wobei  $c_i^1$  ein erstes kodiertes TFCI-Symbol darstellt,  $b$  ein gemultiplextes kodiertes Symbol darstellt,  $\lfloor \cdot \rfloor$  eine Integerzahl anzeigt, die sich durch Abrundung des in der Klammer  $\lfloor \cdot \rfloor$  gegebenen Wertes ergibt,  $n$  die Gesamtanzahl der ersten kodierten TFCI-Symbole wiedergibt und  $i$  einen Index wiedergibt, der ein willkürliches kodiertes Symbol aus den ersten kodierten TFCI-Symbolen angibt.

19. Vorrichtung nach Anspruch 17, wobei der Kodiertsymbol-Neuanordner (**910, 1010**) die zweiten kodierten TFCI-Symbole an Positionen, die durch die folgende Formel berechnet werden, von den kodierten Symbolen trennt, die durch das Multiplexen der ersten kodierten TFCI-Symbole und der zweiten kodierten TFCI-Symbole erhalten werden:

$$b_{\lfloor \frac{n}{32-n} \cdot x(i+\frac{1}{2}) \rfloor} = c_i^2, (0 \leq i \leq m-1)$$

wobei  $c_i^2$  ein zweites kodiertes TFCI-Symbol darstellt,  $b$  ein gemultiplextes kodiertes Symbol darstellt,  $\lfloor \cdot \rfloor$  einen Maximalwert der Integerzahlen angibt, der kleiner oder gleich dem gegebenen Wert in den Klammern  $\lfloor \cdot \rfloor$  ist,  $n$  die Gesamtanzahl der ersten kodierten TFCI-Symbole wiedergibt,  $m$  die Gesamtanzahl der zweiten kodierten TFCI-Symbole wiedergibt und  $i$  einen Index wiedergibt, der ein willkürliches kodiertes Symbol aus den ersten kodierten TFCI-Symbolen angibt.

20. Verfahren zum Dekodieren von  $k$  ersten TFCI-Bits und  $(10 - k)$  zweiten TFCI-Bits in einem Mobilkom-

munikationssystem zum Empfangen von  $(3k + 1)$  ersten kodierten TFCI-Symbolen und  $(31 - 3k)$  zweiten kodierten TFCI-Symbolen, wobei  $k$  eine variable Integerzahl mit einem Wert in einem Bereich  $1 \leq k \leq 9$  ist, wobei das Verfahren folgende Schritte aufweist:

Trennen der ersten kodierten TFCI-Symbole und der zweiten kodierten TFCI-Symbole, die über einen DPCH (dedizierten physikalischen Kanal) übertragen werden, in Übereinstimmung mit der variablen Integerzahl, um diese neu anzuordnen,

Dekodieren der ersten kodierten TFCI-Symbole, um die  $k$  ersten TFCI-Bits auszugeben, und

Dekodieren der zweiten kodierten TFCI-Symbole, um die  $(10 - k)$  zweiten TFCI-Bits auszugeben.

21. Verfahren nach Anspruch 20, wobei die ersten kodierten TFCI-Symbole an Positionen, die durch die folgende Formel berechnet werden, von den kodierten Symbolen getrennt werden, die durch das Multiplexen der ersten kodierten TFCI-Symbole und der zweiten kodierten TFCI-Symbole erhalten werden:

$$b_{\left\lfloor \frac{32-n}{n} \cdot x(i+1) \right\rfloor} = c_i^1, (0 \leq i \leq n-1)$$

wobei  $c_i^1$  ein erstes kodiertes TFCI Symbol darstellt,  $b$  ein gemultiplextes kodiertes Symbol darstellt,  $\lfloor \rfloor$  eine Integerzahl anzeigt, die sich durch Abrundung des in der Klammer  $\lfloor \rfloor$  gegebenen Wertes ergibt,  $n$  die Gesamtanzahl der ersten kodierten TFCI-Symbole wiedergibt und  $i$  einen Index wiedergibt, der ein willkürliches kodiertes Symbol aus den ersten kodierten TFCI-Symbolen angibt.

22. Verfahren nach Anspruch 20, wobei die zweiten kodierten TFCI-Symbole an Positionen, die durch die folgende Formel berechnet werden, von den kodierten Symbolen getrennt werden, die durch das Multiplexen der ersten kodierten TFCI-Symbole und der zweiten kodierten TFCI-Symbole erhalten werden:

$$b_{\left\lfloor \frac{n}{32-n} \cdot x(i+\frac{1}{2}) \right\rfloor} = c_i^2, (0 \leq i \leq m-1)$$

wobei  $c_i^2$  ein zweites kodiertes TFCI-Symbol darstellt,  $b$  ein gemultiplextes kodiertes Symbol darstellt,  $\lfloor \rfloor$  einen Maximalwert der Integerzahlen angibt, der kleiner oder gleich dem gegebenen Wert in den Klammern  $\lfloor \rfloor$  ist,  $n$  die Gesamtanzahl der ersten kodierten TFCI-Symbole wiedergibt,  $m$  die Gesamtanzahl der zweiten kodierten TFCI-Symbole wiedergibt und  $i$  einen Index wiedergibt, der ein willkürliches kodiertes Symbol aus den ersten kodierten TFCI-Symbolen angibt.

Es folgen 20 Blatt Zeichnungen

Anhängende Zeichnungen

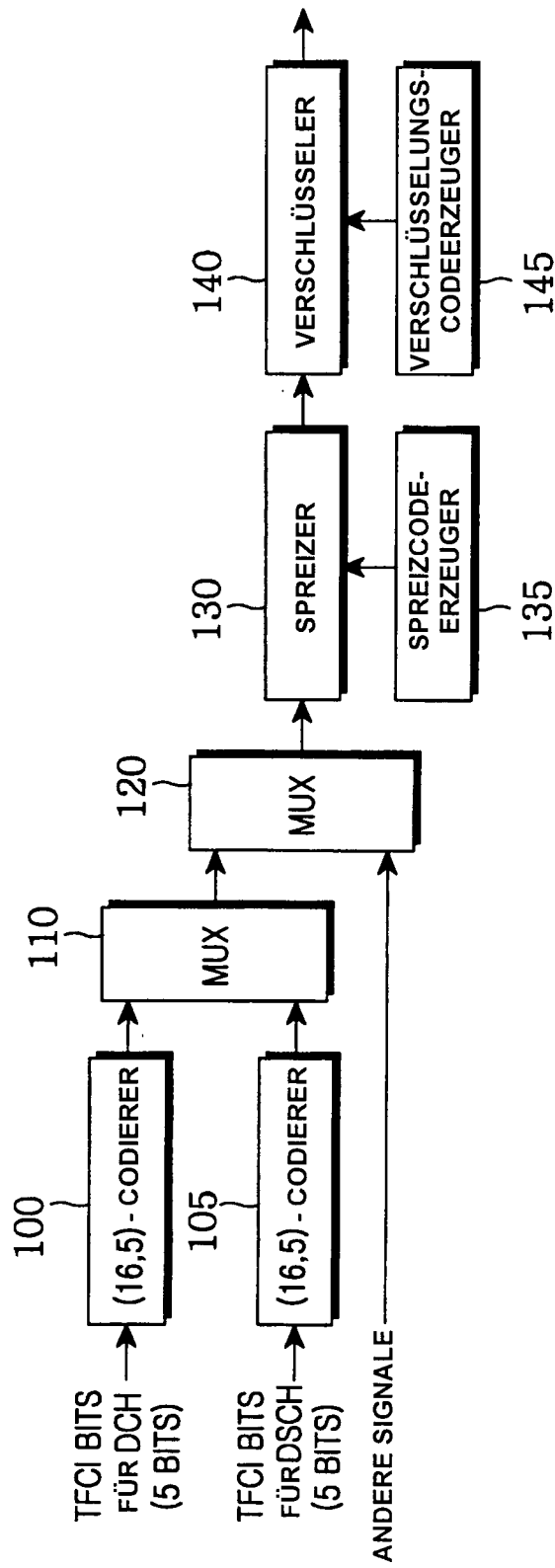


FIG.1

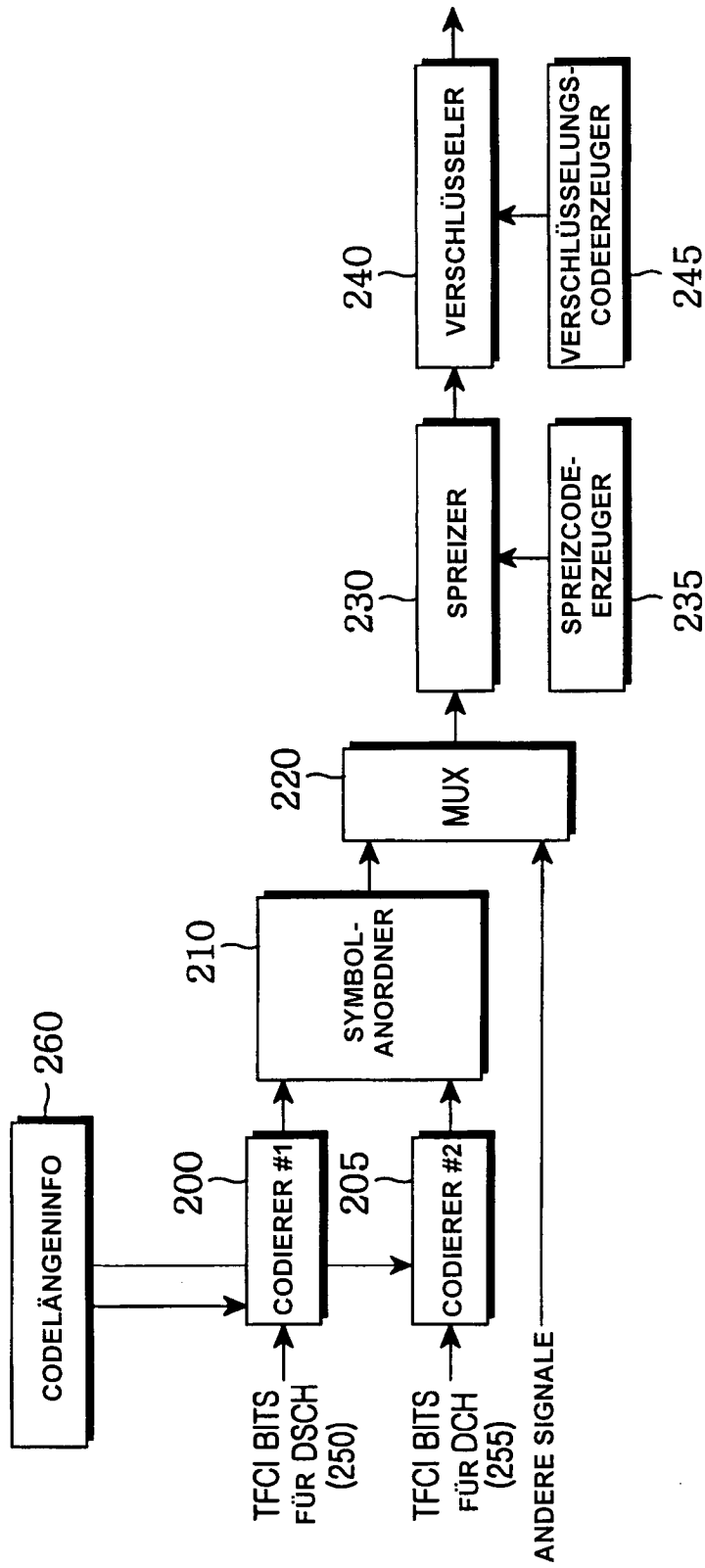


FIG.2

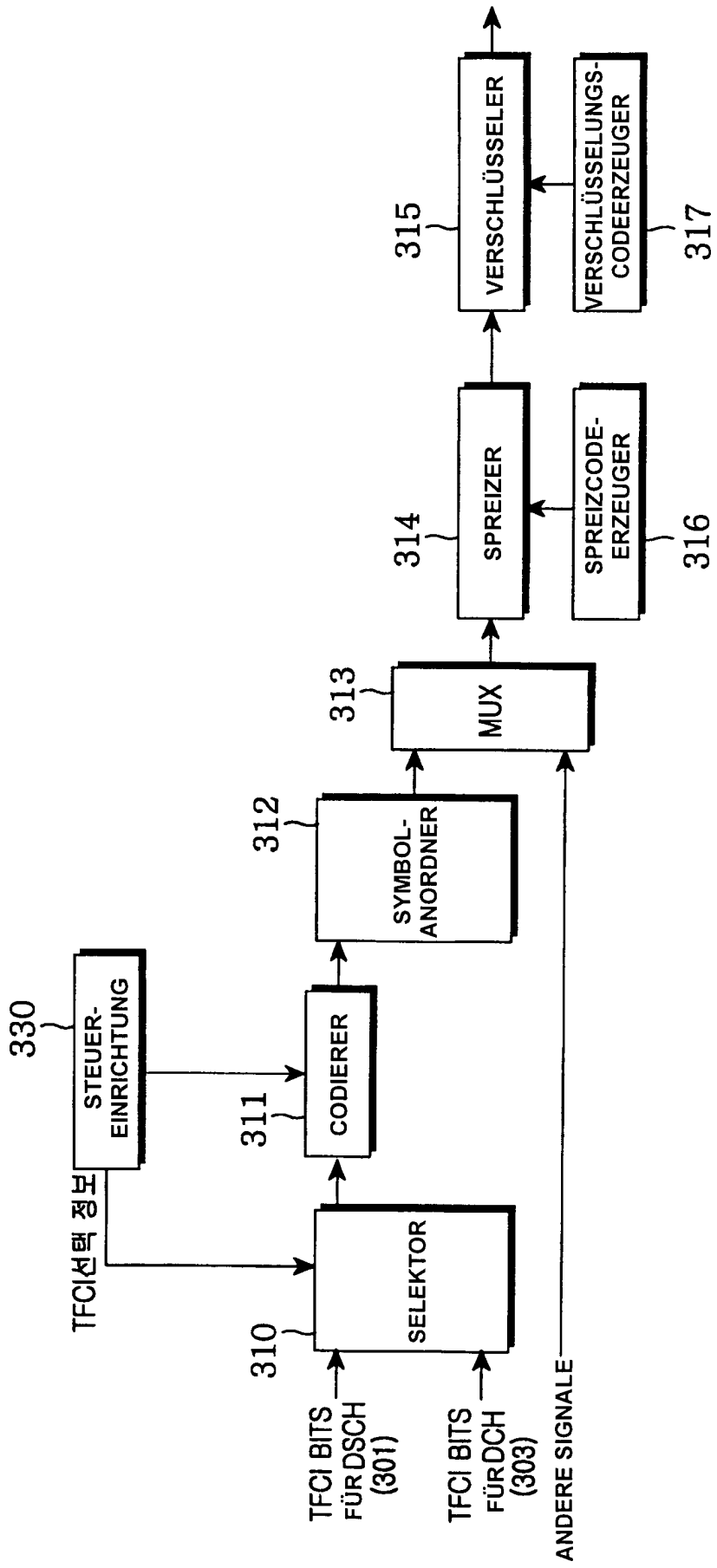


FIG.3

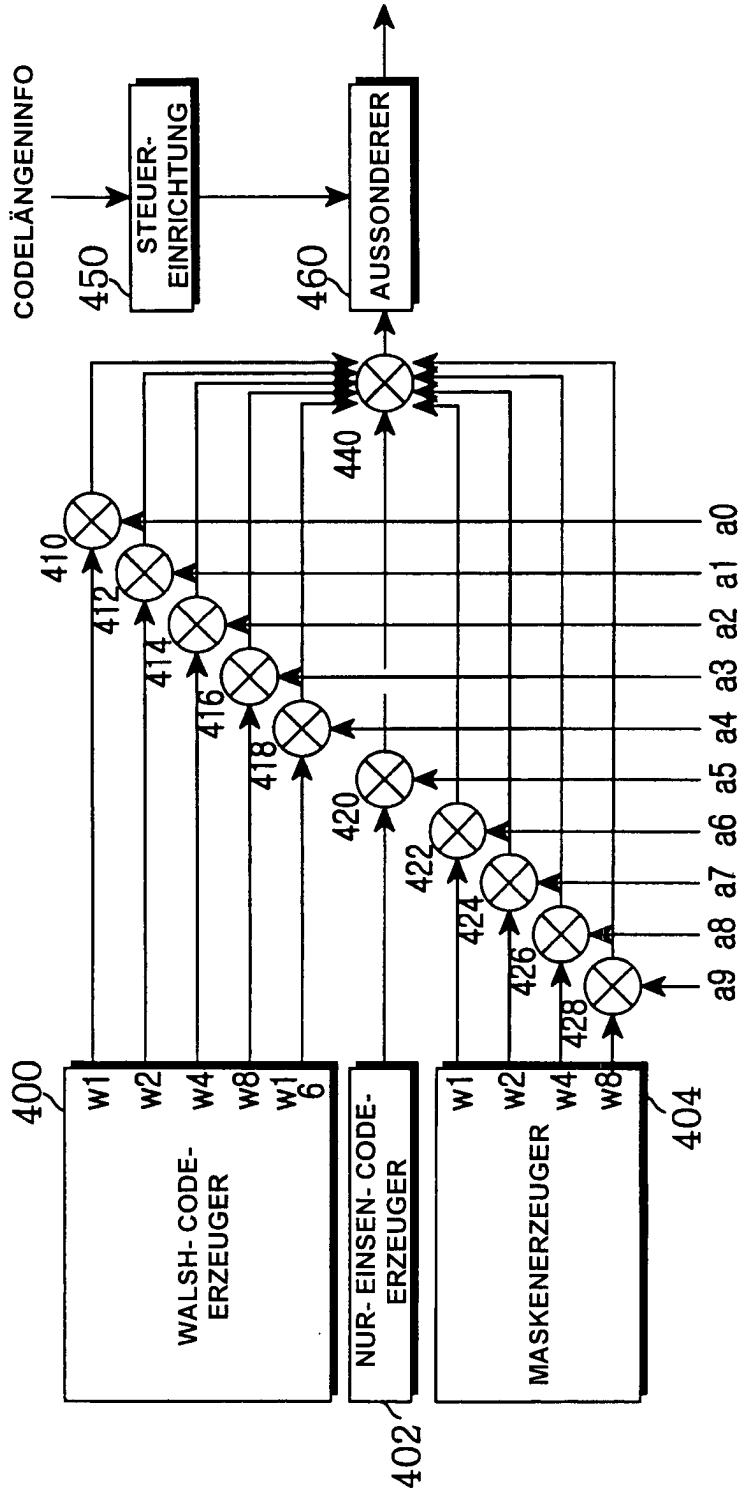


FIG.4

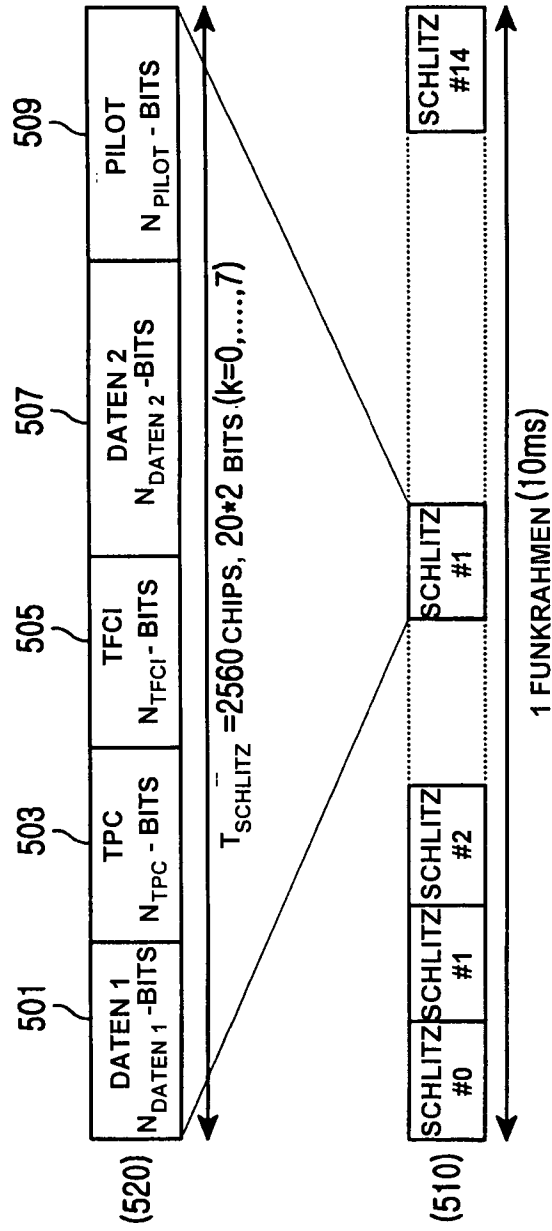


FIG.5

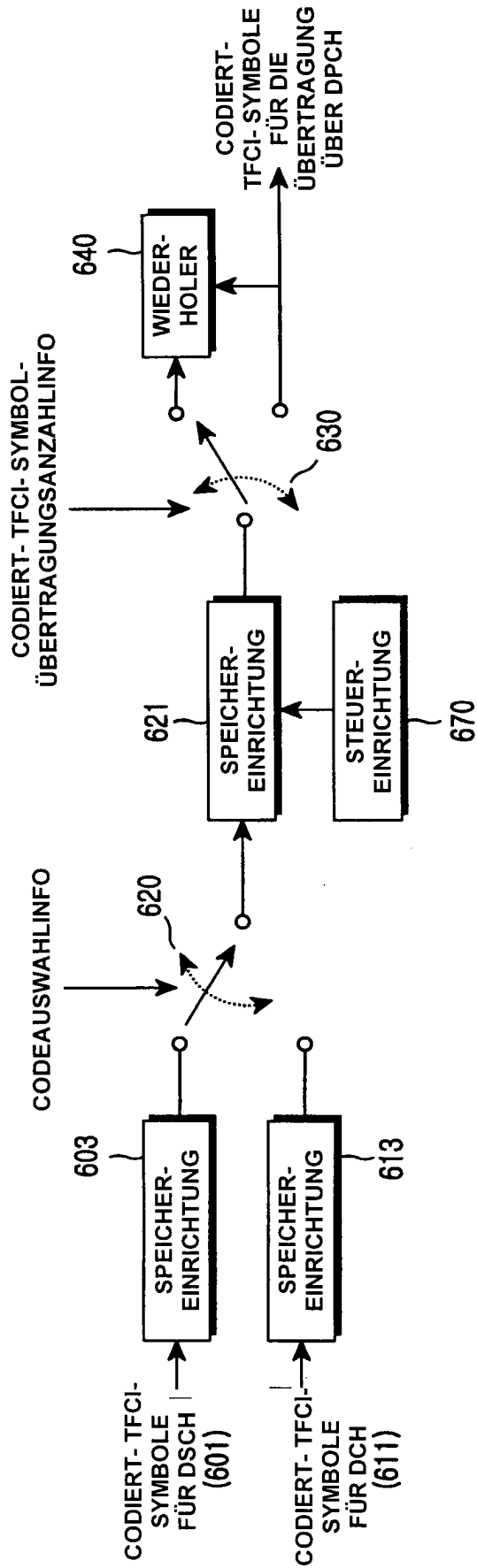


FIG.6

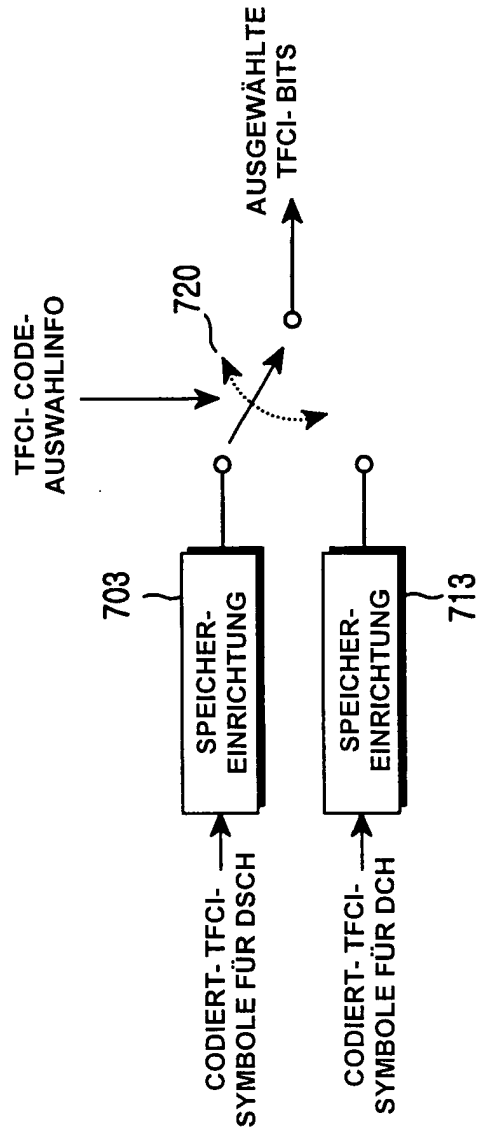


FIG.7

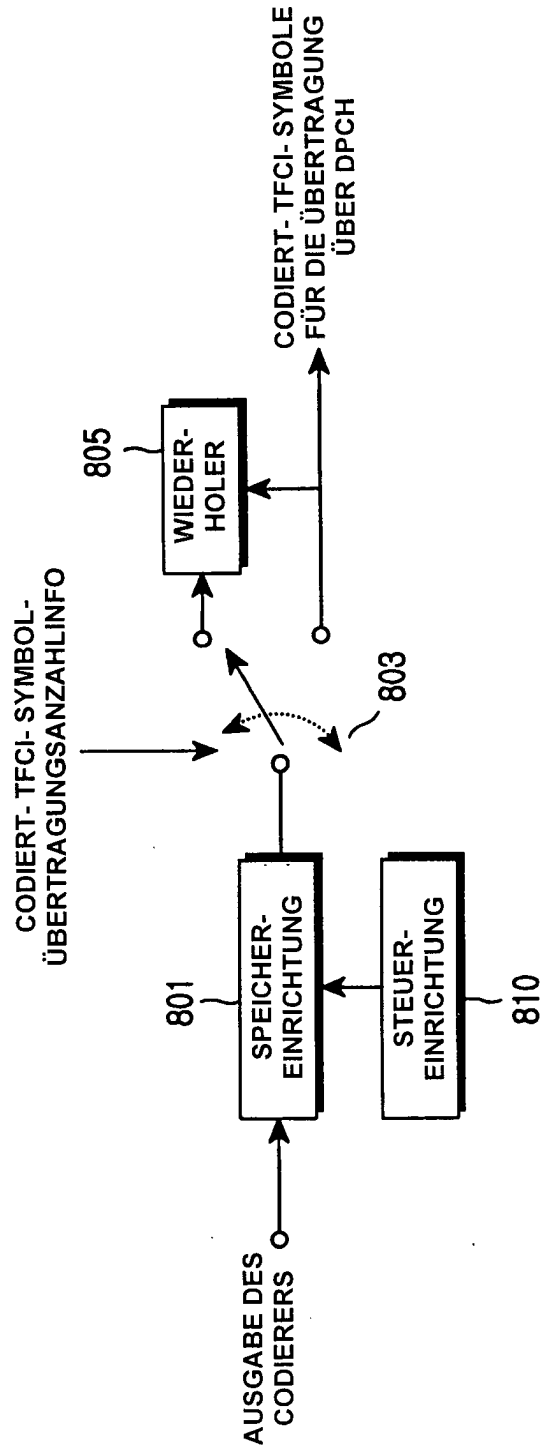


FIG. 8

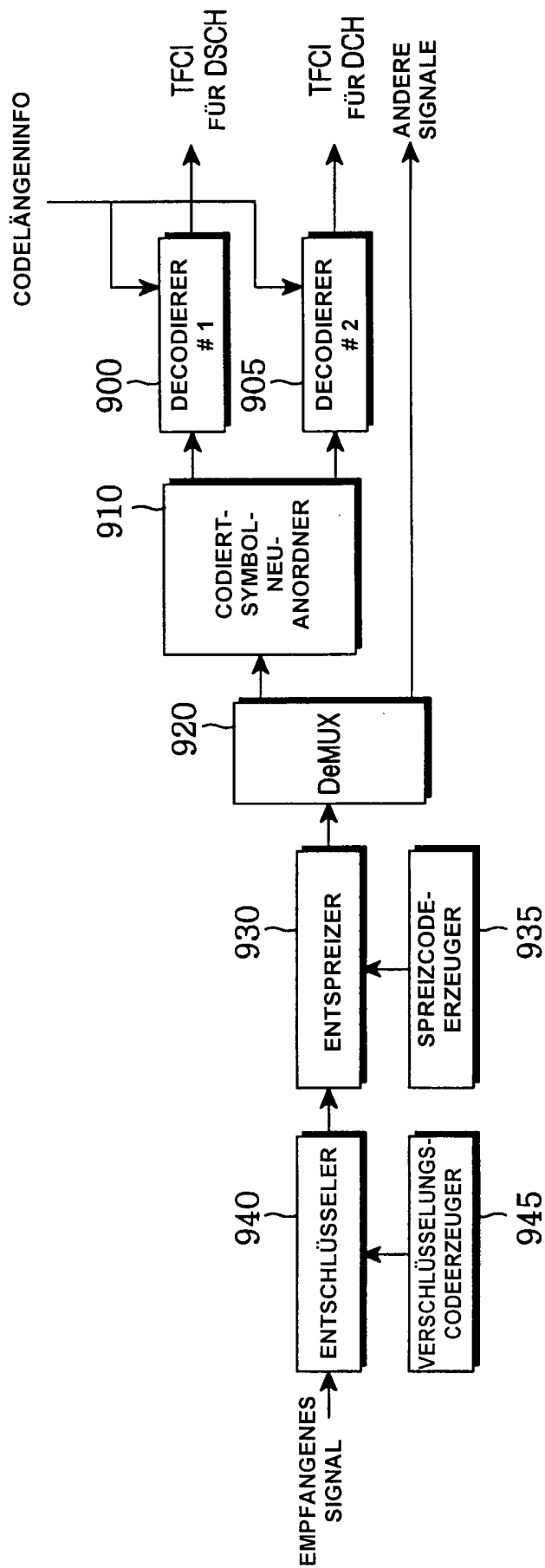


FIG.9

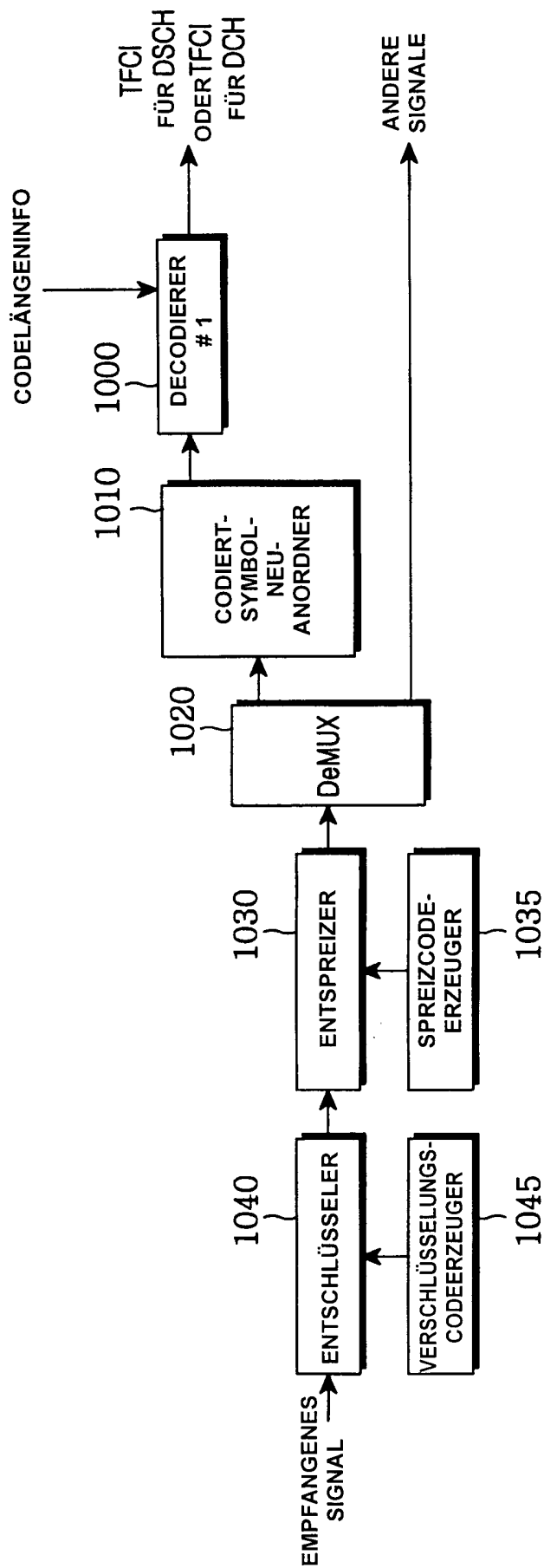


FIG.10

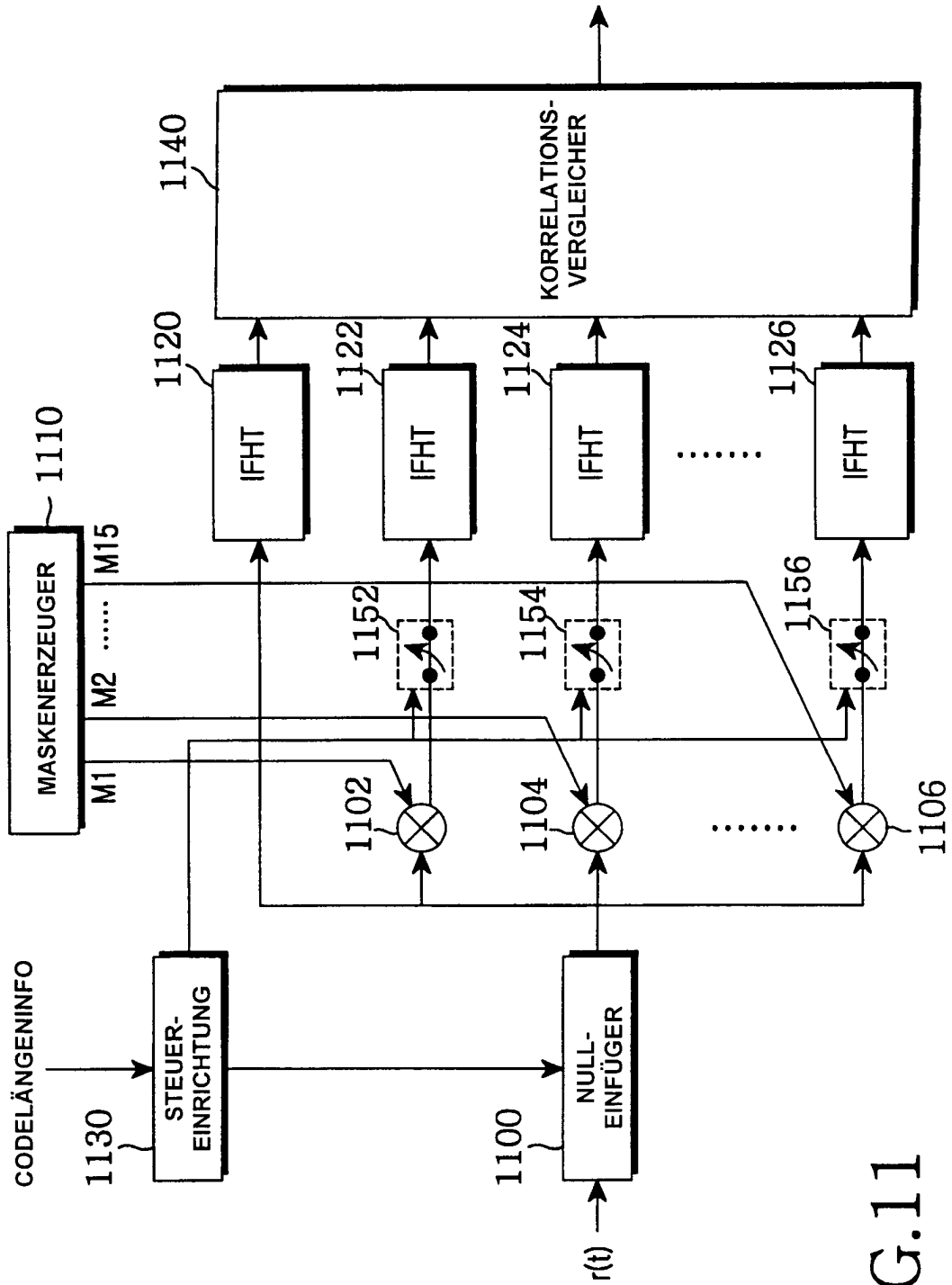


FIG.11

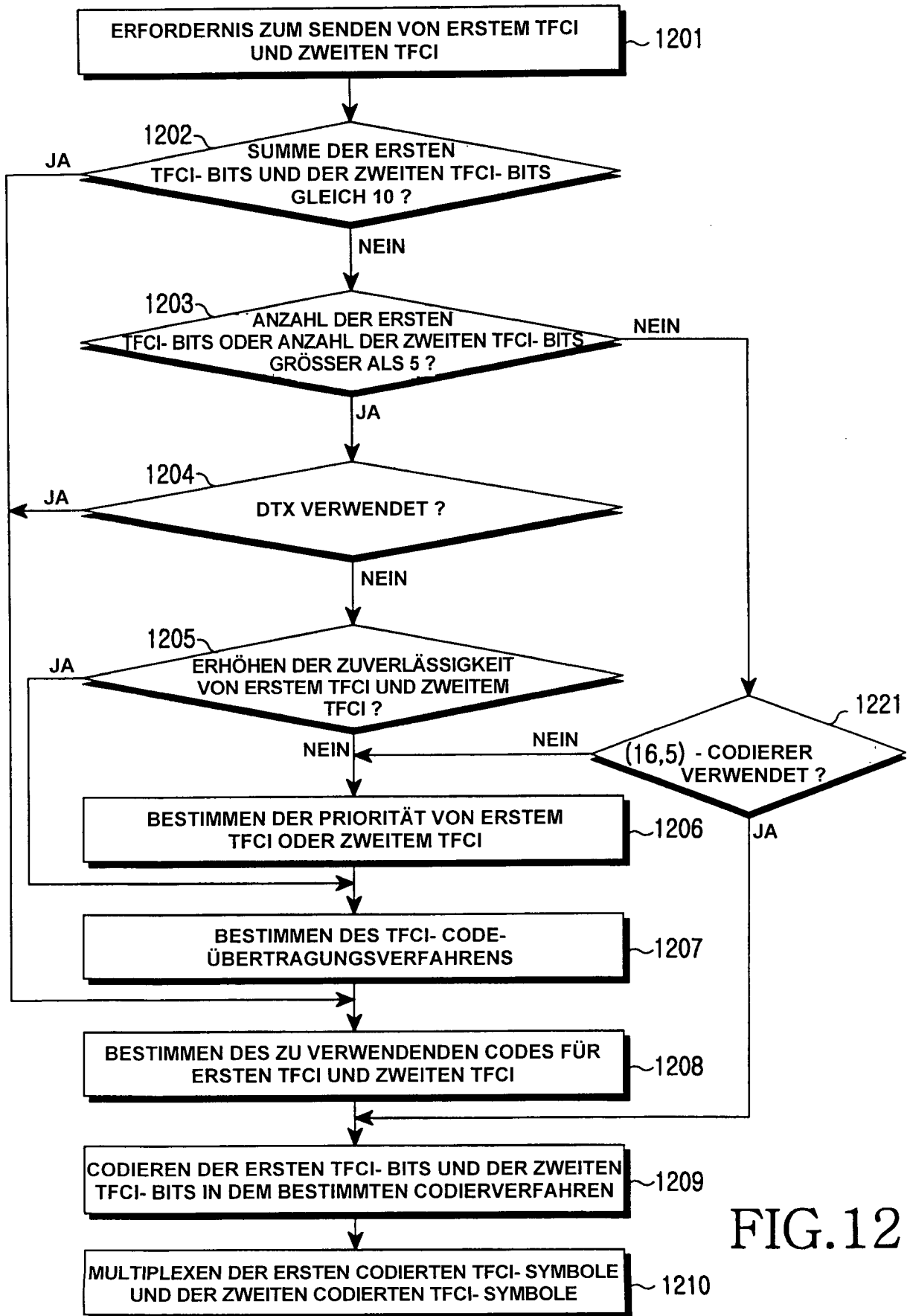


FIG.12

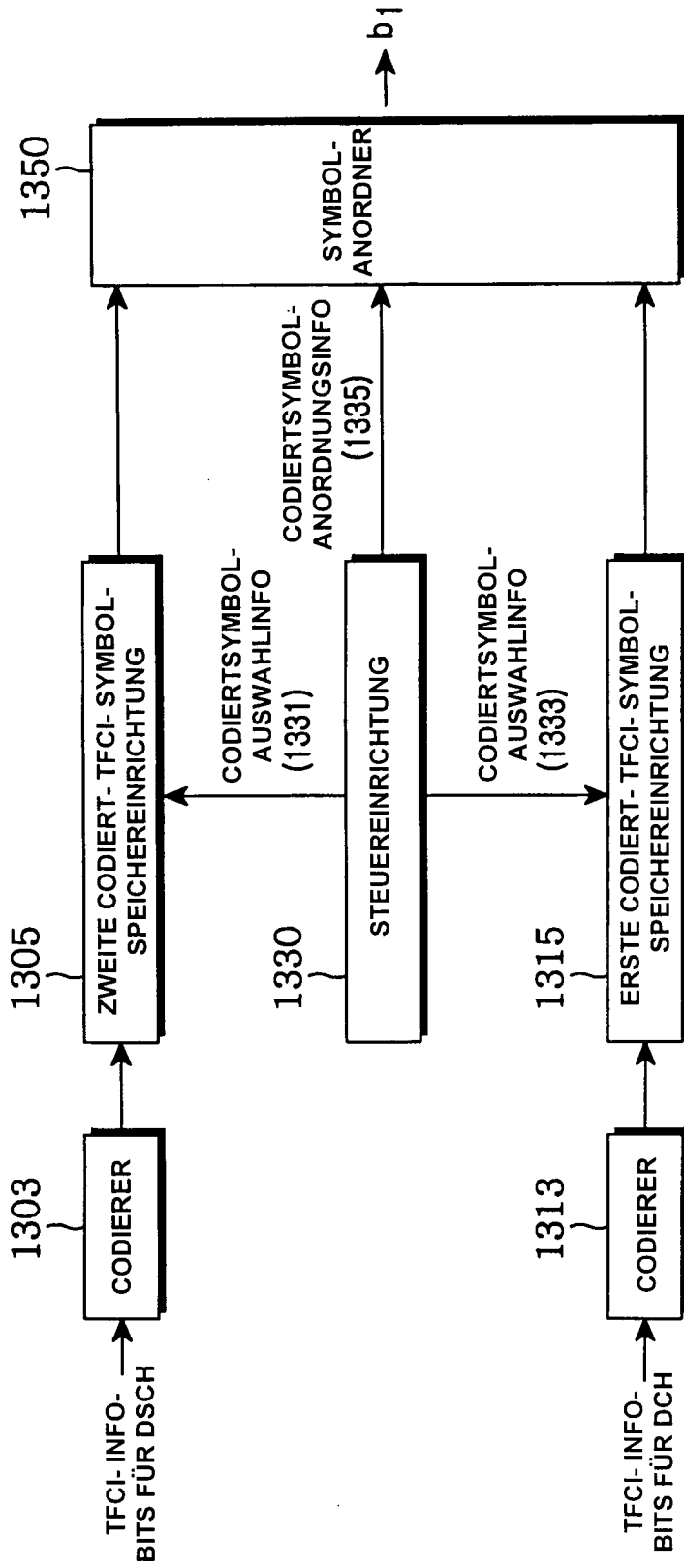


FIG.13

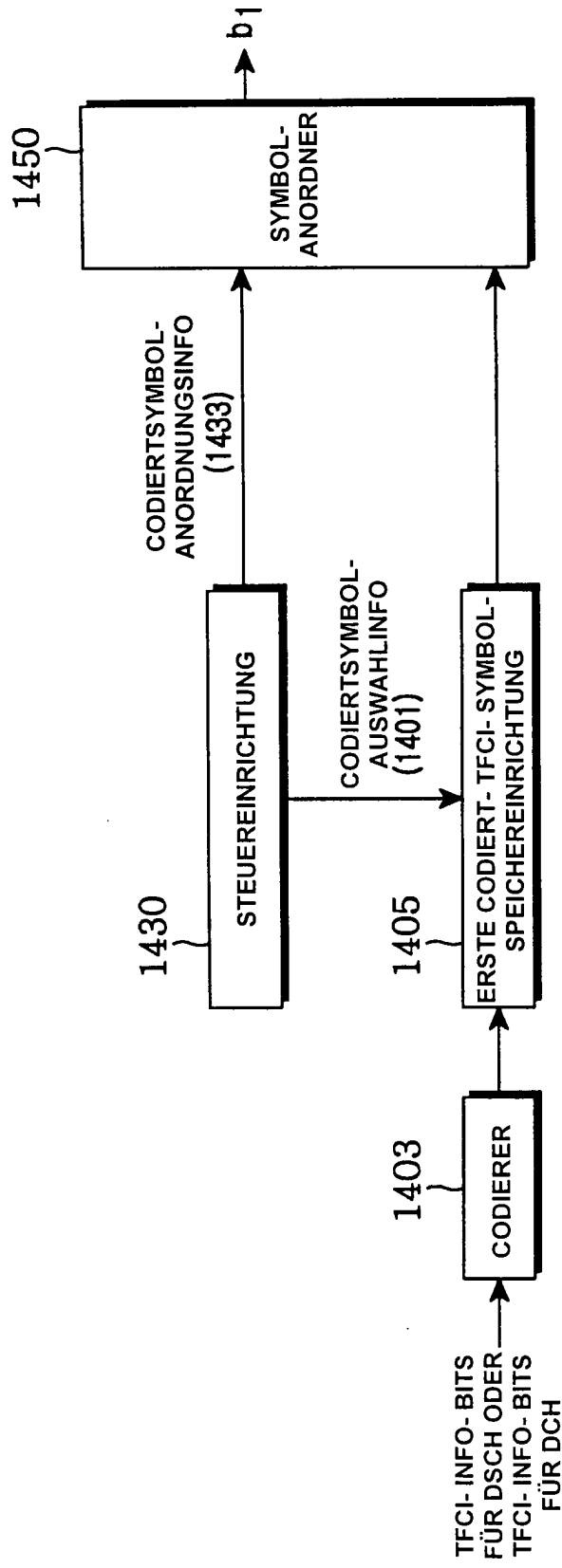


FIG.14

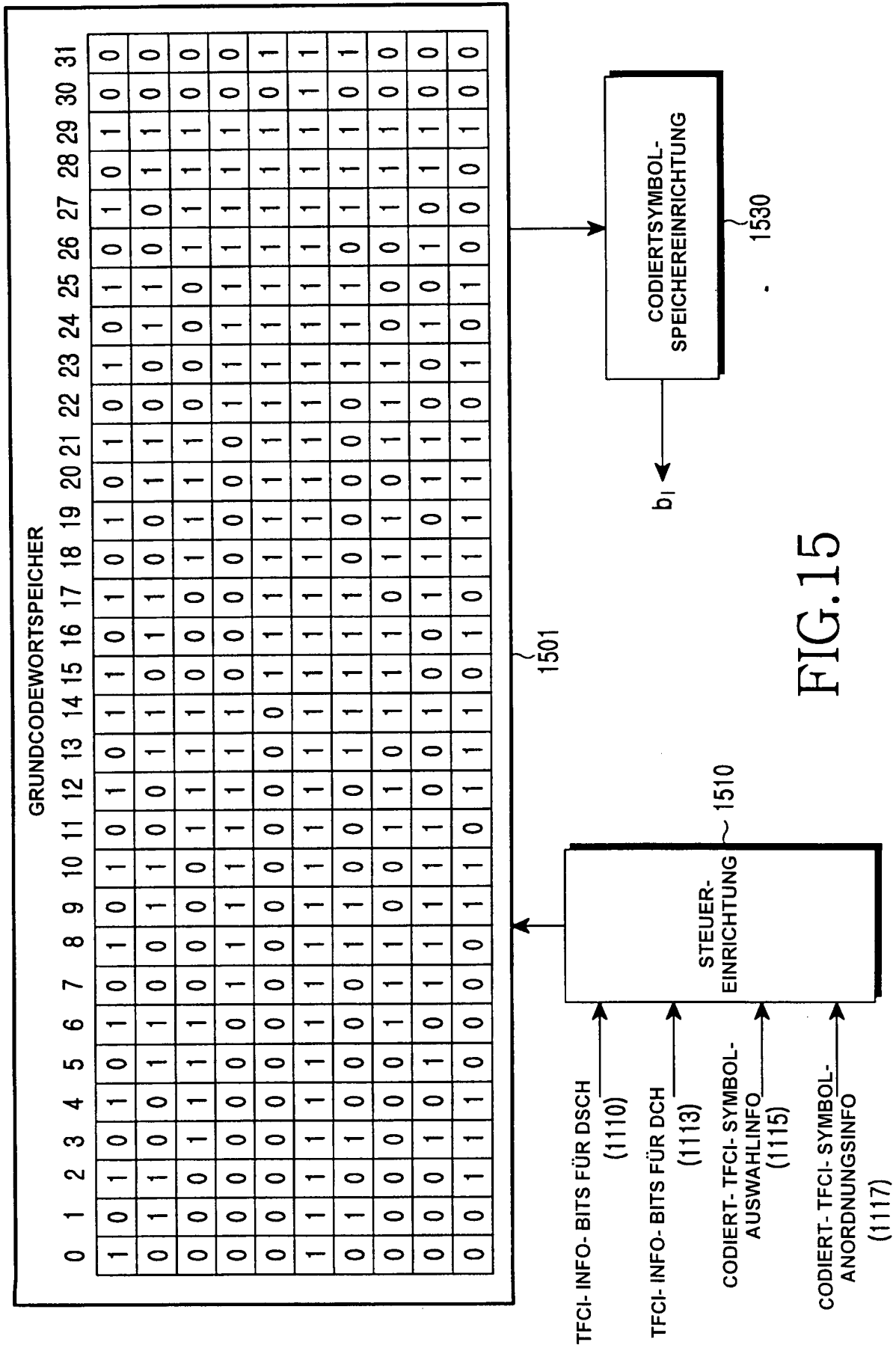


FIG.15

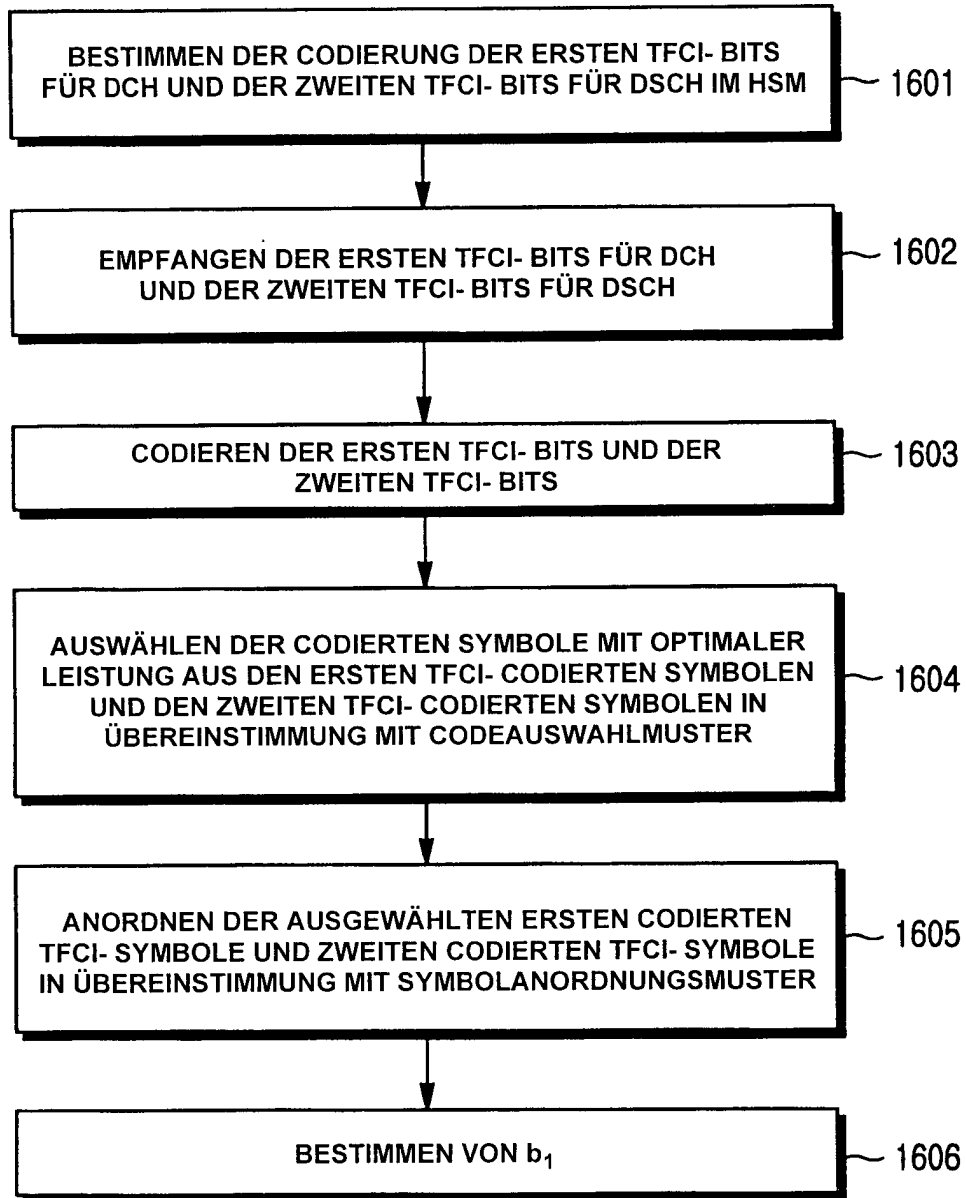


FIG.16

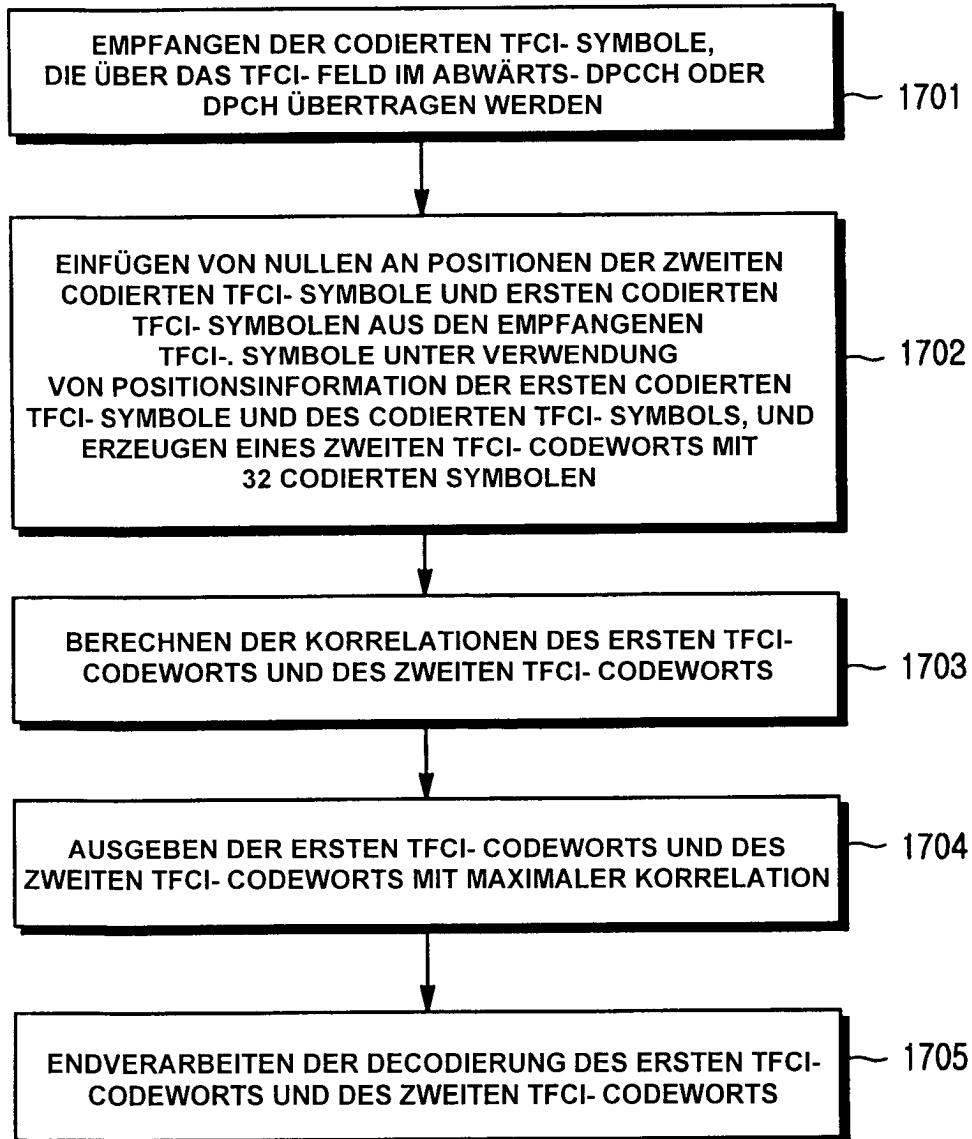


FIG.17

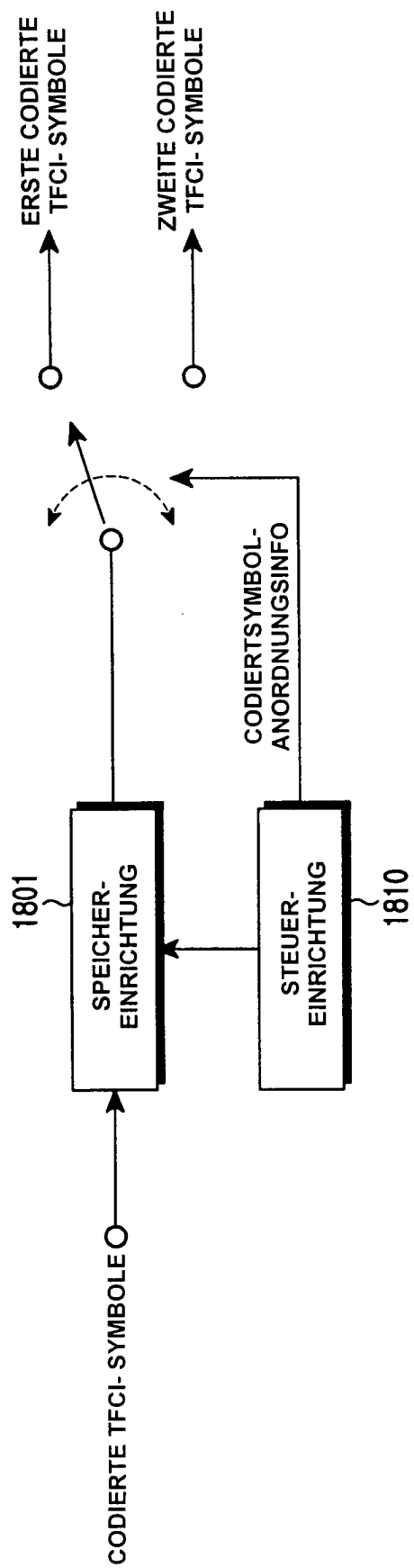


FIG.18a

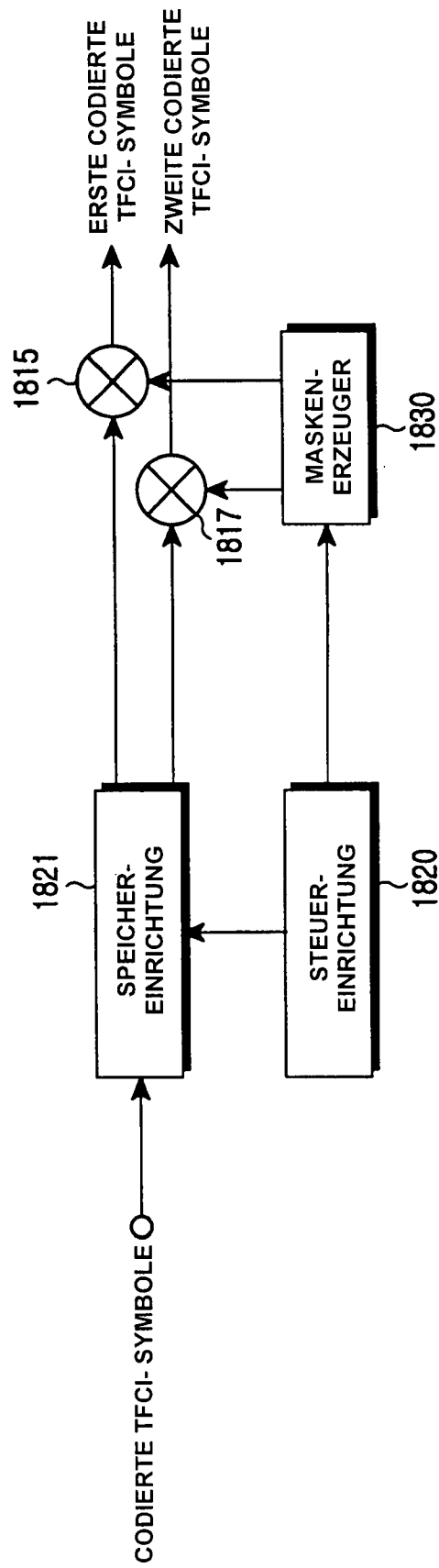


FIG.18b

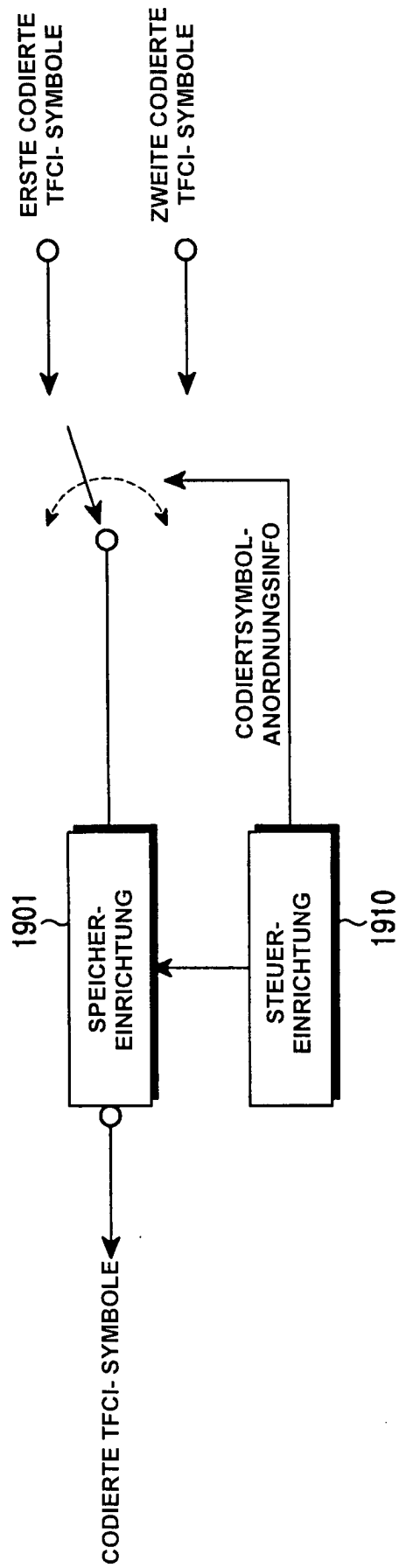


FIG.19