

PŘIHLÁŠKA VYNÁLEZU

zveřejněná podle § 31 zákona č. 527/1990 Sb.

(21) Číslo dokumentu:

2000 -308

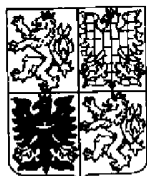
(13) Druh dokumentu: **A3**

(51) Int. Cl.:

H 03 M 5/14

G 11 B 20/14

(19)
ČESKÁ
REPUBLIKA



ÚŘAD
PRŮMYSLOVÉHO
VLASTNICTVÍ

(22) Přihlášeno: **25.05.1999**

(32) Datum podání prioritní přihlášky: **29.05.1998**

(31) Číslo prioritní přihlášky: **1998/105280**

(33) Země priority: **JP**

(40) Datum zveřejnění přihlášky vynálezu: **15.11.2000**
(Věstník č. 11/2000)

(86) PCT číslo: **PCT/IB99/00948**

(87) PCT číslo zveřejnění: **WO99/63671**

(71) Přihlašovatel:

KONINKLIJKE PHILIPS ELECTRONICS N. V.,
Eindhoven, NL;
SONY CORPORATION, Tokyo, JP;

(72) Původce:

Schouhamer Immink Kornelis, Eindhoven, NL;
Kahlman Joseph A. H. M., Eindhoven, NL;
Van den Enden Gijsbert J., Eindhoven, NL;
Nakagawa Toshiyuki, Eindhoven, NL;
Shinpuku Yoshihide, Eindhoven, NL;
Narahara Tatsuya, Eindhoven, NL;
Nakamura Kosuke, Eindhoven, NL;

(74) Zástupce:

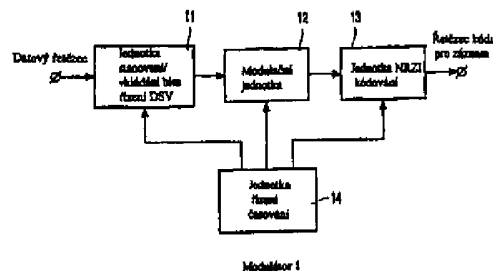
Zelený Pavel JUDr., Hájkova 2, Praha 2, 12000;

(54) Název přihlášky vynálezu:

**Metoda pro modulaci /demodulaci s následným
omezením minimální délky běhu a přístroj k
jejímu provádění**

(57) Anotace:

Jednotka (11) stanovení/vykládání bitů řízení DSV vkládá bity řízení DSV pro provedení řízení DSV do vstupního datového řetězce a vysílá datový řetězec obsahující bity řízení DSV na modulační jednotku (12). Modulační jednotka (12) převádí datový řetězec se základní délkou dat 2 bitů na kód s proměnnou délkou se základní délkou kódu 3 bitů podle převodní tabulky a vysílá kód, který je výsledkem přeměny do jednotky kódování NRZI (13). Převodní tabulka používána modulační jednotkou (12) obsahuje substituční kódy pro omezení počtu následných výskytů minimálního oběhu na předem stanovenou hodnotu a substituční kódy pro udržení limitu délky běhu. Navíc převodní tabulka uplatňuje převodní pravidlo, podle kterého bude zbytek dělení dvěma počtu "1" elementu v datovém řetězci s hodnotou 0 nebo 1 vždy roven zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, který je výsledkem přeměny zmíněného datového řetězce.



Přístroj a Metoda pro modulaci/demodulaci s následným
omezením minimální délky běhu a přístroj k jejímu provádění

Oblast techniky

Tento vynález se obecně týká modulačního přístroje a modulační metody, demodulačního přístroje, demodulační metody a také média, které reprezentuje program. Tento vynález se týká obzvláště výhodného modulačního přístroje, výhodné modulační metody, výhodného demodulačního přístroje, výhodné demodulační metody a také výhodného média, které reprezentuje program, kterých se využívá při operacích záznamu dat na záznamové médium s vysokou záznamovou hustotou a reprodukce dat zaznamenaných na záznamovém médiu s vysokou hustotou záznamu.

Dosavadní stav techniky

Při přenosu dat přenosovou cestou nebo při jejich záznamu na záznamové médium, jakými jsou magnetický disk nebo optický disk, jsou tato data před přenosem nebo záznamem modulována kódem, který je vhodný pro danou přenosovou cestu nebo záznamové médium. Známou modulační metodou je blokové kódování. Při blokovém kódování je datový řetězec rozčleněn do jednotek, kdy každá obsahuje $m \times i$ bitů. Každá jednotka, která bude v dalším označována jako datové slovo, je pak přeměněna na kódové slovo, které obsahuje $n \times i$ bitů podle příslušného kódovacího pravidla.

Pro $i=1$ je toto kódové slovo kódem s pevnou délkou. V případě, že i nabývá několika hodnot, které jsou vybírány z intervalu 1 až $i_{\max}$, což je maximální hodnota i , je výsledné kódové slovo kódem s proměnnou délkou. Obecně je kód, který je výsledkem blokového kódování vyjádřen jako kód s proměnnou délkou ($d, k; m, n; r$).

Zde se i označuje omezení délky a r je $i_{\max}$, maximální omezení délky. d je minimální počet nul, které leží mezi dvěma následujícími jedničkami. Písmenem d se označuje minimální běh nul. Na druhé straně, k je maximální počet nul, které leží mezi dvěma následujícími jedničkami. Proměnná k se nazývá maximální běh nul.

Při operaci záznamu kódu s proměnnou délkou, který byl získán blokovým kódováním, které je popsáno výše, na záznamové médium, jakými jsou např. optický disk nebo magnetooptický disk, např. na kompaktní disk (CD) nebo minidisk (MD), je kód s proměnnou délkou modulován pomocí NRZI (invertovaný non return to zero), kde každá "1" kódu s proměnnou délkou je interpretována jako změna stavu a "0" je interpretována jako žádná změna stavu. Pak je kód s proměnnou délkou, který představuje modulaci NRZI zaznamenán. Kód s proměnnou délkou, který představuje modulaci NRZI je označován jako záznamový sled vln. V případě magnetooptického záznamu, který je v souladu s dřívějšími normami ISO, které stanovovaly nepřiliš velké hustoty záznamu, jsou sledy bitů představující záznamovou modulaci zaznamenány bez jejich modulace pomocí kódu NRZI.

Nechť $T_{\min}$ a $T_{\max}$ označuje minimální, resp. maximální inverzní periody záznamových sled vln. V tomto případě se pro záznam záznamového sledu vln s vysokou záznamovou

.

hustotou v lineárním směru rychlosti dává přednost dlouhé minimální inverzní periodě T_{min} nebo dlouhému minimálnímu běhu d . Navíc z hlediska generování hodinového signálu je žádoucí mít krátkou maximální inverzní periodu T_{max} nebo krátký maximální běh k . Aby se těmto požadavkům vyhovělo, bylo navrženo mnoho různých modulačních metod.

Pro uvedení konkrétního příkladu: pro optický disk, magnetický disk a magneto-optický disk byly navrženy nebo jsou v současnosti používány modulační metody pro generování kódu s proměnnou délkou RLL (1-7), který se též popisuje jako (2,7 m, n; r), a také kód s pevnou délkou RLL (1-7), který se též popisuje jako (1, 7; m, n; 1), který se používá v ISO normě MO. V případě přístrojů využívajících disk, které se v současnosti zkoumají a vyvíjejí, jakými jsou optický disk a magneto-optický disk s vysokou záznamovou hustotou, se běžně používá kód RLL (kód s omezenou délkou běhu) s minimálním během $d = 1$.

Následuje příklad převodní tabulky kódu RLL s proměnnou délkou (1-7).

TABULKA 1

RLL(1, 7; 2,3; 2)

	Data	Kód	
i=1	11	00x	
	10	010	
	01	10x	
i=2	0011	000	00x
	0010	000	010

0001

100

00x

0000

100

010

Znak x použitý v převodní tabulce má hodnotu "1" pro kanál následující bit "0" nebo má hodnotu "00" pro kanál následující bit "1". Maximální omezení délky r je rovna 2.

Parametry kódu s proměnnou délkou RLL (1-7) jsou (1,7; 2,3; 2). Minimální inverzní perioda T_{min} , která se může vyjádřit jako $(d+1)T$ je tak rovna $2 (=1+1)T$ kde T je bitová mezera v záznamovém sledu vln. Minimální inverzní perioda T_{min} , která může být vyjádřena též pomocí $(m/n) \times 2T_{data}$, je tak rovna 1,33 $(=2/3 \times 2) T_{data}$, kde T data je bitová mezera v datovém řetězci. Maximální inverzní perioda T_{max} , která může být vyjádřena pomocí $(k+1)T$ je tak rovna $(7+1)T = 8T = 8 \times (m/n)T_{data} = 8 \times 2/3 T_{data} = 5,33T_{data}$. Šířka detekčního okénka T_w , které může být vyjádřeno také jako $(m/n)T_{data}$ je tak rovna 0,67 $(=2/3) T_{data}$.

Mimoходом ve sledu kanálových bitů představujících modulaci RLL (1-7), který je uveden v tabulce 1, je generující kmitočet s periodou $2T$, která je rovna minimální inverzní periodě T_{min} , většinou následován generujícím kmitočtem, který odpovídá periodám $3T$ a $4T$. Fakt, že hodně okrajové informace je generováno v krátkých intervalech jako $2T$ a $3T$ je v mnoha případech výhodný pro generování hodinového signálu. Avšak když se hustota záznamu dále zvýší, stává se naopak minimální běh problémem. T.j., pokud jsou následně generovány dva běhy $2T$, je záznamový sled vln, náchylný ke zkreslení, které je v něm generováno. To je z důvodu, že výstup $2T$ vlny je menší, než výstupy jiných vln a tudíž je snadno ovlivnitelný takovými faktory, jakými jsou

.

rozostření a tangenciální zešikmení. Navíc, při vysoké hustotě je záznam následných minimálních značek (2T) též snadno ovlivněn poruchami, jako např. šumem. Takže operace přehrávání dat bude též náchylná k chybám. V tomto případě se v mnoha případech pozoruje chybový obrazec při reprodukci dat jako posun předního a zadního kraje minimální značky. Výsledkem je, že se zvýší délka generované bitové chyby. Jak je popsáno výše, když jsou data přenášena přenosovou cestou nebo zaznamenávána na záznamové médium, jsou data modulována před přenosem nebo záznamem kódem, který přísluší dané přenosové cestě nebo záznamovému médiu. Pokud výsledný modulační kód obsahuje složku stejnosměrného proudu, jsou snadno generovány různé chybové signály, např. chyba polohování, která je generována v řídicím servu pohonu disku, náchylné ke změnám nebo časové nestabilitě.

Aby se zabránilo situaci, kdy modulovaný kód obsahuje stejnosměrnou proudovou složku, bylo navrženo řízení DSV (řízení hodnoty digitálního součtu), aby se zabránilo situaci, kdy modulovaný kód má stejnosměrnou proudovou složku. DSV je součet hodnot sledu bitů (znaků dat), kdy hodnotám +1 a -1 jsou ve sledu přiřazeny hodnoty "1", resp. "0", což vytváří modulaci NRZI (tzn. hladinové kódování) sled kanálových bitů. DSV je indikátorem stejnosměrné složky proudu, která je obsažena ve sledu kódů. Zmenšování absolutní hodnoty DSV pomocí řízení DSV je ekvivalentní potlačování velikosti stejnosměrné složky proudu, která je obsažena v sledu kódů.

Řízení DSV není aplikována na modulační kód, který je generován podle kódu s proměnnou délkou RLL (1-7) z tabulky 1, která je uvedena výše. Řízení DSV v takovém případě je

prováděno výpočtem DSV sledu kódových bitů (sled kanálových bitů) po modulaci po předem stanovenou časovou periodu a vložení předem stanoveného počtu bitů řízení DSV do sledu kódových bitů (sled kanálových bitů).

V každém případě jsou bity řízení DSV v zásadě redundantními. Pokud se má brát v úvahu účinnost kódové konverze, je potřeba zredukovat počet bitů řízení DSV na nejmenší možnou hodnotu.

Dále, pokud jsou přidány bity řízení DSV, je též žádoucí neměnit minimální běh d a maximální běh k . To je z toho důvodu, že změna v (d,k) má vliv na záznamové/reprodukční charakteristiky.

Podstata vynálezu

Jak je popsáno výše, při operaci záznamu kódu RLL s vysokou hustotou nebo při operaci přehrávání kódu RLL zaznamenaného s vysokou hustotou vzniká problém, že obrazec následných minimálních běhů d snadno způsobí generování dlouhých chyb.

Navíc v případě kódu RLL, jako např. kód RLL (1-7) vyžaduje řízení DSV přidání bitů řízení DSV do volitelné části řetězce kódových slov (sled kanálových bitů). Protože bity řízení DSV jsou v podstatě redundantní, je však žádoucí snížit počet přidanych bitů řízení DSV na nejmenší možnou hodnotu. Aby se však držel minimální běh d a maximální běh k konstantní, počet bitů řízení DSV je alespoň 2. Je proto žádoucí snížit počet bitů řízení DSV na ještě menší hodnotu.

Tento vynález řeší problémy, které jsou popsány výše.

.

Cílem tohoto vynálezu je umožnit zpracování bitů řízení DSV pro vytvoření vysoké efektivity řídicích bitů na kódu RLL při $(d, k; m, n)$, kde minimální běh $d=1$, to znamená kód RLL při $(1, 7; 2, 3)$ tak, aby počet následných minimálních běhů byl snížen při držení minimálního běhu a maximálního běhu.

Dalším cílem tohoto vynálezu je zabránit rozšíření chyby demodulace pomocí použití převodní tabulky s nejjednodušší možnou konfigurací.

Způsob řešení problému

Modulátor podle patentového nároku 1 se vyznačuje tím, že obsahuje prostředky pro převod vstupních dat do kódu podle převodní tabulky, kdy zmíněná převodní tabulka uplatňuje převodní pravidlo, podle kterého zbytek dělení dvěma počtu "1" elementu v datovém řetězci s hodnotami 0 nebo 1 bude vždy roven zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, které jsou výsledkem konverze zmíněného datového řetězce a konverzní kódy zmíněné konverzní tabulky obsahují:

základní kódy pro $d=1$, $k=7$, $m=2$ a $n=3$ kde d je minimální běh a k je omezení délky běhu.

první substituční kódy pro omezení počtu následných zmíněných minimálních běhů d a

druhé substituční kódy pro udržování zmíněného limitu délky běhu k .

Modulační metoda podle bodu 23 se vyznačuje tím, že obsahuje krok konverze vstupních dat do kódu podle převodní tabulky, kdy zmíněná převodní tabulka uplatňuje převodní

.

pravidlo, podle kterého zbytek dělení dvěma počtu "1" elementu v datovém řetězci s hodnotami 0 nebo 1 bude vždy roven zbytku dělení dvěma počtu "1" elementu v řetězci kódových slovech, které jsou výsledkem konverze zmíněného datového řetězce konverzní kódy zmíněné konverzní tabulky obsahují:

základní kódy pro $d=1$, $k=7$, $m=2$ a $n=3$ kde d je minimální běh a k je omezení délky běhu.

první substituční kódy pro omezení počtu následných zmíněných minimálních běhů d a

druhé substituční kódy pro udržování zmíněného limitu délky běhu k .

Program, který představuje médium podle patentového nároku 24 pro prezentaci programu, který implementuje zpracování včetně kroku přeměny vstupních dat do kódu podle převodní tabulky dat v modulátoru pro převod dat se základní délkou dat m bitů do kódu s proměnnou délkou $(d,k;m,n;r)$ se základní délkou kódu n bitů se vyznačuje tím, že zmíněná převodní tabulka uplatňuje převodní pravidlo, podle kterého bude zbytek dělení dvěma počtu "1" elementu datového řetězce s hodnotou 0 nebo 1 vždy rovný zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, který je výsledkem konverze zmíněného datového řetězce, a převodní kód zmíněné převodní tabulky obsahuje:

základní kódy pro $d=1$, $k=7$, $m=2$ a $n=3$, kde d je minimální běh a k je omezení délky běhu.

první substituční kódy pro omezení počtu následných výskytů zmíněných minimálních běhů d a

druhé substituční kódy pro udržení zmíněné omezené délky běhu k .

.

Demodulátor podle patentového nároku 25 se vyznačuje tím, že obsahuje prostředky pro přeměnu vstupního kódu do dat podle převodní tabulky, kdy zmíněná tabulka uplatňuje převodní pravidlo, podle kterého bude zbytek dělení dvěma počtu "1" elementu datového řetězce s hodnotou 0 nebo 1 vždy rovný zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, který je výsledkem konverze zmíněného datového řetězce, a převodní kód zmíněné převodní tabulky obsahuje:

základní kódy pro $d=1$, $k=7$, $m=2$ a $n=3$, kde d je minimální běh a k je omezení délky běhu.

první substituční kódy pro omezení počtu následných výskytů zmíněných minimálních běhů d a

druhé substituční kódy pro udržení zmíněné omezené délky běhu k .

Demodulační metoda podle patentového nároku 28 se vyznačuje tím, že obsahuje krok přeměny vstupního kódu na data podle převodní tabulky, kdy zmíněná tabulka uplatňuje převodní pravidlo, podle kterého bude zbytek dělení dvěma počtu "1" elementu v datovém řetězci s hodnotou 0 nebo 1 vždy rovný zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, který je výsledkem konverze zmíněného datového řetězce, a převodní kód zmíněné převodní tabulky obsahuje:

základní kódy pro $d=1$, $k=7$, $m=2$ a $n=3$, kde d je minimální běh a k je omezení délky běhu.

první substituční kódy pro omezení počtu následných výskytů zmíněných minimálních běhů d a

druhé substituční kódy pro udržení zmíněné omezené délky běhu k .

Program, který představuje médium podle patentového nároku 29 pro prezentaci programu, který obsahuje krok přeměny vstupních dat do kódu podle převodní tabulky dat v demodulátoru pro převod kódu s proměnnou délkou $(d,k;m,n;r)$ se základní délkou kódu n bitů na data se základní délkou m bitů se vyznačuje tím, že zmíněná převodní tabulka uplatňuje převodní pravidlo, podle kterého bude zbytek dělení dvěma počtu "1" elementu v datovém řetězci s hodnotou 0 nebo 1 vždy rovný zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, které jsou výsledkem konverze zmíněného datového řetězce, a převodní kód zmíněné převodní tabulky obsahuje:

základní kódy pro $d=1$, $k=7$, $m=2$ a $n=3$, kde d je minimální běh a k je omezení délky běhu.

první substituční kódy pro omezení počtu následných výskytů zmíněných minimálních běhů d a

druhé substituční kódy pro udržení zmíněné omezené délky běhu k .

Podle modulátoru, který je nárokován v bodu 1, modulační metody podle bodu 23, programu představujícího médium podle bodu 24, demodulátor podle bodu 25, demodulační metody podle bodu 28 a média představujícího program podle bodu 29 je proveden proces převodu na základě převodní tabulky, která uplatňuje převodní pravidlo, podle kterého bude zbytek dělení dvěma počtu "1" elementu v datovém řetězci s hodnotou 0 nebo 1 vždy rovný zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, který je výsledkem konverze zmíněného datového řetězce, a převodní kód zmíněné převodní tabulky obsahuje:

základní kódy pro $d=1$, $k=7$, $m=2$ a $n=3$,

první substituční kódy pro omezení počtu následných výskytů zmíněných minimálních běhů d a

druhé substituční kódy pro udržení zmíněné omezené délky běhu k.

Přehled obrázků na výkresech a příklady provedení vynálezu

Před započítím objasňování některých výhodných provedení tohoto vynálezu je nutné, abychom objasnili vztahy, které spojují prostředky tohoto vynálezu popsanych v patentových nárocích s implementacemi, které jsou použity v provedeních, a proto je v následujícím popisu charakterizující vynález každý z prostředků následován typickou implementací v závorce ve formě "prostředky (implementovány např. typickou implementací)". Není třeba však zdůrazňovat, že typická implementace se nemá chápat jako omezení. Tzn. prostředky nejsou nutně omezeny na typickou implementaci s nimi spojenou.

Modulátor podle patentového nároku 1 se vyznačuje tím, že obsahuje převodní prostředky (implementovány např. modulační jednotkou z obr. 12) pro převod vstupních dat do kódu podle převodní tabulky (která je implementována např. tabulkou 2), kdy zmíněná převodní tabulka uplatňuje převodní pravidlo, podle kterého zbytek dělení dvěma počtu "1" elementu v datovém řetězci s hodnotami 0 nebo 1 bude vždy roven zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, který je výsledkem konverze zmíněného datového řetězce a konverzní kódy zmíněné konverzní tabulky obsahují:

10-10-78

druhé substituční kódy pro udržování zmiňného limitu
délky běhu k.

Podle bodu 10, modulátor nárokováný jako bod 10 se dále vyznačuje tím, že dále obsahuje prostředky vkládání synchronizačního signálu (implementované například jednotkou vkládání synchronizačního signálu 212 z obr. 9) pro vkládání synchronizačního signálu, včetně jedinečného obrazce, který není obsažen ve zmíněných konverzních kódech zmíněné převodní tabulky do jakékoliv libovolné pozice ve zmíněném řetězci kódových slov.

Podle bodu 21, modulátor nárokový jako bod 1 se dále vyznačuje tím, že obsahuje prostředky řízení DSV (implementované např. jednotkou stanovení/vkládání bitů řízení DSV 11 z obr. 1) pro řízení DSV vstupních dat a přivedení zmíněných DSV na zmíněné konverzní prostředky.

Podle bodu 22, modulátor nárokovaný jako bod 1 se dále vyznačuje tím, že zmíněné konverzní prostředky obsahují:

prostředky detekce prvního kódu (implementované např. jednotkou detekce kódu omezení následného výskytu minimálních běhů 33 z obr. 3) pro detekci zmíněných prvních substitučních kódů pro omezení počtu následných výskytů zmíněného minimálního běhu d a

druhé detekční prostředky (implementované např. prostředky detekce kódu zabezpečující maximální běh 34 z obr. 3) pro detekci zmíněných substitučních kódů pro udržování omezení délky běhu.

1

1

2

•

vynálezu.

<Tabulka 2>

17PP.RML.32

Data	Kód
11	*0*
10	001
01	010
0011	010 100
0010	010 000
0001	000 100
000011	000 100 100
000010	000 100 000
000001	010 100 100
000000	010 100 000
"110111	001 000 000 (dále 010)
00001000	000 100 100 100
00000000	010 100 100 100
pokud xx1	pak *0* = 000
xx0	pak *0* = 101

Synchronizace a zakončení

#01 000 000 001 (12 kanálových bitů)

nebo

#01 001 000 000 001 000 000 001 (24 kanálových bitů)

= 0: nenastává konec

= 1: nastává konec

Tabulka zakončení

00	000	
0000	010	100

"11111 001 000 000 (dále 010)

Pokud jsou dalšími kanálovými bity "010", přeměň " 01 11" na "001 000 000" po použití hlavní převodní tabulky a tabulky zakončení.

Jak uvádí tabulka 2, ukazuje převodní tabulka kódy, které jsou výsledkem přeměny včetně základních kódů, substitučních kódů a kódů zakončení. Proces přeměny nemůže být proveden bez základního kódu. V převodní tabulce jsou základní kódy ty kódy, které jsou výsledkem převodu datových řetězců (11) až (000000). Přeměna může být provedena dokonce i když neexistuje substituční kód. Pokud však substituční kód existuje, může být provedena efektivnější přeměna.

.

V převodní tabulce jsou substituční kódy ty kódy, které jsou výsledkem přeměny datových řetězců (110111), (00001000) a (00000000). Kód zakončení se používá pro zakončení kódu, který je výsledkem přeměny na jakékoliv pozici. Kódy zakončení v tabulce jsou ty kódy, které jsou výsledkem přeměny datových řetězců (00) a (0000). Navíc tabulka předepisuje také synchronizační signály.

V tabulce 2 je minimální běh d 1 zatímco maximální běh k je 7. Jeden z prvků základních kódů obsahuje neurčitý kód, tzn. kód, který je indikován symbolem asterisk "*". Bit reprezentovaný symbolem "*" v neurčitém kódu se může vybrat buď "0" nebo "1", aby se udržely hodnoty minimálního běhu d a maximálního běhu k bez ohledu na těsně předcházející nebo následující řetězec kódových slov. Abychom to vysvětlili detailně, pokud dvoubitový datový řetězec, který má být přeměněn je roven (11), může být výsledný kód "000" nebo "101" v závislosti na těsně předcházejícím řetězci kódových slov. Podrobněji řečeno, pokud jeden kanálový bit těsně předcházející řetězec kódových slov je "1", je dvoubitový datový řetězec (11) přeměněn na kódové slovo "000", aby se udržel minimální běh d . Na druhé straně, pokud jeden kanálový bit těsně předcházející řetězec kódových slov je "0", je dvoubitový datový řetězec (11) přeměněn na kód "101", aby se udržel maximální běh k .

Základní kódy, které jsou uvedeny v převodní tabulce 2 mají strukturu proměnné délky. Počet základních kódů s konstantní délkou $i = 1$ je 3, což je hodnota menší, než požadovaný počet 4 ($=2^m=2^2$). Tyto 3 základní kódy jsou "*0*", "001" a "010". Výsledkem je, že při operaci přeměny datového řetězce se setkáme s datovým řetězcem, který nemůže

.

být přeměněn pouze s omezením délky $i = 1$. Z tohoto důvodu je nutné se zpětně odkázat na základní kódy s omezením délky i do 3 v tabulce 2 při operaci přeměny všech datových řetězců. To znamená, že základní kód s omezením délky i do 3 je obsažen v tabulce 2, aby tato sloužila jako dostačující převodní tabulka.

Převodní tabulka 2 obsahuje navíc substituční kódy pro omezení následných výskytů minimálního běhu d . Pokud má datový řetězec podobu (110111) a řetězec kódových slov, který následuje kód, který je výsledkem přeměny datového řetězce je "010", je datový řetězec přeměněn na kódové slovo "010 000 000". Na druhé straně, pokud řetězec kódových slov následující kód, který je výsledkem přeměny datového řetězce je jiný než "010", je datový řetězec (110111) přeměněn do dvou bitů jednotek. Pro detailní vyjádření, jsou dvoubitové sledy (11), (01) a (11) v datovém řetězu přeměněny na řetězce kódových slov "*0* 010 a *0*". Výsledkem je, že následné výskyty minimálního běhu d v řetězci kódových slov, který je výsledkem přeměny datového řetězce mohou být potlačeny, při omezení počtu opakování minimálních běhů na maximálně 6. Navíc převodní tabulky 2 uplatňuje převodní pravidlo, podle kterého bude zbytek dělení dvěma počtu "1" elementu v datovém řetězci s hodnotou 0 nebo 1 vždy rovný zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, který je výsledkem přeměny datového řetězce. To znamená, pokud počet "1" elementu v datovém řetězci je sudý, je počet "1" elementu v řetězci kódových slov také sudý a pokud je počet "1" elementu v datovém řetězci lichý, je také počet "1" elementu v řetězci kódových slov lichý. Např. datový řetězec (000001) je přeměněn na řetězec kódových slov

"010 100 000". V tomto případě je zbytek dělení dvěma počtu "1" elementu v datovém řetězci roven 1, což je rovno zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, který je výsledek přeměny datového řetězce. To znamená, že počty "1" elementů datového řetězce a řetězce kódových slov jsou vždy liché. V dalším případě je datový řetězec (000000) přeměněn na řetězec kódových slov "010 100 100". V tomto případě je zbytek dělení dvěma počtu "1" elementu v datovém řetězci roven 0, což je rovno zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, který je výsledkem přeměny datového řetězce. To znamená, že počet "1" datového řetězce a řetězce kódových slov je vždy lichý.

Navíc maximální omezení délky r v převodní tabulce 2 je rovno 4. Kódy v tabulce s omezením délky $i = 4$ jsou substituční kódy pro implementaci maximálního běhu $k = 7$. Taková substituce se označuje jako maximálně zajištěný kód. To znamená, že datový řetězec (00001000) je přeměněn na řetězec kódových slov "000 100 100 100", zatímco datový řetězec (00000000) je přeměněn na řetězec kódových slov "010 100 100 100". Mělo by se poznamenat, že i v tomto případě se udrží hodnota minimálního běhu d rovna 1.

Pokud převodní tabulka 2 neobsahuje substituční kódy, které mají omezení délky $i = 4$, je maximální omezení délky r pro danou tabulku rovna 3, což má za následek generování kódu s maximálním během $k = 8$. Protože však tabulka obsahuje základní kódy s omezením délky 4, může být generován kód s maximálním během $k = 7$.

Obecně řečeno, čím větší je maximální běh k , tím obtížnější je generování hodinového signálu a tím je horší stabilita systému. Takže snížením hodnoty maximálního běhu

.

k z 8 na 7 se může zlepšit charakteristika systému souměřitelně se snížením maximálního běhu k.

To znamená, pokud převodní tabulka 2 je vytvořena tak, že obsahuje pouze základní kódy, je maximální omezení délky r takové tabulky rovna 3. V tomto případě je možné generovat kód, který má minimální běh $d = 1$ a maximální běh $k = 8$. Navíc zbytek dělení dvěma počtu "1" elementu v datovém řetězci s hodnotou 0 nebo 1 bude vždy roven zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, který je výsledkem přeměny datového řetězce.

Pokud je převodní tabulka vytvořena tak, že obsahuje též substituční kódy pro omezení následných výskytů minimálního běhu d navíc k základním kódům, je maximální omezení délky r takové tabulky rovna také 3. V tomto případě je však možné generovat kód, který má minimální běh $d = 1$ a maximální běh $k = 8$, zatímco počet následných minimálních běhů d je omezen na svou horní hodnotu. Navíc zbytek dělení dvěma počtu "1" elementu v datovém řetězci s hodnotou 0 nebo 1 bude vždy roven zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, který je výsledkem přeměny datového řetězce.

Pokud je převodní tabulka vytvořena tak, že obsahuje též substituční kódy pro zabezpečení maximálního běhu $k = 7$ navíc k substitučním kódům pro omezení následných výskytů minimálního běhu d a základním kódům, je maximální omezení délky r pro takovou tabulku rovno 4. V tomto případě je možné generovat kód, který má minimální běh $d = 1$ a maximální běh $k = 7$, zatímco počet následných minimálních běhů d je omezen na horní hodnotu. Navíc zbytek dělení dvěma počtu "1" elementu v datovém řetězci s hodnotou 0 nebo 1

.

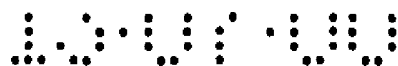
bude vždy roven zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, který je výsledkem přeměny datového řetězce.

Obecně řečeno, čím větší je maximální omezení délky r , tím horší jsou charakteristiky šíření generované demodulační chyby v případě bitového posuvu, to znamená chyba, která je generována, protože krajní bit je posunut v ve směru vpřed nebo vzad ze své obvyklé pozice o jeden bit.

Porovnání tabulky 1 a tabulkou 2 ukazuje, že maximální omezení délky r první je rovna 2, zatímco druhé je rovna 4. Takže tabulka 2 by měla mít horší charakteristiky než tabulka 1. Avšak výsledky simulace, která bude popsána později v odkazu na tabulku 7 ukazují, že charakteristiky tabulky 2 nejsou ve srovnání s tabulkou 1 tak špatné. Např. jak ukazuje tabulka 7, průměrná četnost chyb pro tabulku 1 je 1,014 bytů, zatímco pro tabulku 2 je to 1,67 bytů, což je mnohem větší hodnota než pro tabulku 1. Rozdíl v průměrné četnosti chyb se může přičíst faktu, že počet přeměn kódových skupin v tabulce 1 je menší než v tabulce 2 o rozdíl rovný dvěma.

Mimochodem pro případ, kdy je synchronizační signál zaváděn na libovolnou pozici do řetězce kódových slov (tzn. sled kanálových bitů) generovaných jako výsledek přeměny provedené podle převodní tabulky 2, vytváří převodní tabulka kód se strukturou proměnné délky. To je proto, že převodní tabulka obsahuje zakončovací tabulku, která předepisuje zakončovací kódy pro zakončení kódu, který je výsledkem přeměny na libovolné pozici. Zakončovací kód je použit kdykoliv je to zapotřebí.

Předpokládejme například, že synchronizační signál je



zaveden na určitou pozici do kódu, který je výsledkem přeměny. V tomto případě, nejprve, v bodě spojení mezi řetězcí kódových slov těsně předcházejících danou pozici a řetězec kódových slov těsně následující danou pozici, jsou nastaveny spojovací bity za současného udržení minimálního běhu d a maximálního běhu k a mezi spojovací bity je nastaven jedinečný obrazec, který reprezentuje synchronizační signál. Uvažujme obrazec synchronizačního signálu, který přeruší hodnotu 7 maximálního běhu k. V tomto případě je obrazec synchronizačního signálu s minimální délkou 12-bitové kódové slovo (tzn. 12 kanálových bitů), které je dáno následovně:

"#01 000 000 001"

Znak "#" na začátku obrazce synchronizačního signálu je spojovací bit, který může mít hodnotu "0" nebo "1", jak bude vysvětleno dále. Druhý kanálový bit následující "#" je bit "0", aby se udržel minimální běh d. Třetí kanálový bit a následné bity jsou nastaveny na hodnoty, které tvoří jedinečný obrazec 9T, což je kódový obrazec, který není předepsán tabulkou 2, aby se získal maximální běh k = 8. Jak ukazuje výše uvedený obrazec, třetí kanálový bit a poslední kanálový bit mají mezi sebou 8 následných bitů "0". Mělo by se poznamenat, že i když poslední kanálový bit v obrazci synchronizačního signálu je nastaven pomocí převodní tabulky 2 na hodnotu "1", může se udržet minimální běh d.

Dále je vysvětlena zakončovací tabulka a spojovací bit "#" v bitovém obrazci synchronizačního signálu. Jak ukazuje tabulka 2, zakončovací tabulka je vytvořena následovně:

00 000



Zakončovací tabulka je požadovaná pro základní kódy s omezením délky i , vytvářející určitý počet párů, kdy každý obsahuje datový řetězec a řetězec kódových slov menší, než je požadovaný počet 4 ($=2^m = 2^2$).

Detailněji řečeno, v případě tabulky 2 pro omezení délky $i = 1$ je požadovaná zakončovací tabulka, protože počet párů, které obsahují datový řetězec a řetězec kódových slov, je roven 3. Pro omezení délky $i = 2$ je zakončovací tabulka požadovaná, protože počet párů, které obsahují datový řetězec a řetězec kódových slov je také roven 3. Pro omezení délky $i = 3$ je počet párů obsahujících datový řetězec a řetězec kódových slov roven 5 včetně páru obsahujícího substituční kód. Zbývají 4 páry, které obsahují základní kód. Protože požadovaný počet 4 je splněn, není potřeba zakončovací tabulky. Pro omezení délky $i = 4$ není třeba brát zakončovací kód v úvahu, protože řetězce kódových slov jsou všechno substituční kódy. Takže zakončovací tabulka je požadována pro omezení délky $i = 1$, kdy zakončovací kód je použit pro datový řetězec (00). Ze stejného důvodu je zakončovací tabulka požadována pro omezení délky $i = 2$, kdy zakončovací kód je použit pro datový řetězec (0000). Podle zakončovací tabulky jsou datové řetězce (00) a (0000) přeměněny do řetězců kódových slov "000", resp. "010100". Výsledkem je, že při operaci vkládání synchronizačního signálu je možné předejít situaci, při které data předcházející obrazci synchronizačních signálů už nemohou být přeměněna. To znamená, že systém přeměny je schopný eliminovat situaci, kdy není možné nechat kód, který těsně

1. *Phragmites* (common)

•

•

■

•

•

•

11111111

$k = 8$ (9T). Takový synchronizační signál může být detekován s vyšším stupněm spolehlivosti. Pak v případě vytvoření 24-bitového kódového slova může být vytvořen následující synchronizační signál.

"#01 001 000 000 001 000 000 001"

Výše uvedený synchronizační signál, který má obrazec ve formě "3T - 9T - 9T", snižuje pravděpodobnost výskytu dlouhých běhů (T) před a/nebo po dvou následných obrazcích, kdy každý dává maximální běh $k = 8$ (9T) a také zvyšuje detekční výkon. Je možné vybrat podle systémových požadavků, detekční energii, kterou má synchronizační signál zabezpečit.

Tabulka 3 je další typickou převodní tabulkou, která je vytvořena podle tohoto vynálezu.

<Tabulka 3>

17PP.RML.52

i=1 Hlavní tabulka:

Data	Kód
00	101
01	100
10	001
11	000

i=2 Substituční tabulka A (omezuje d na 1)

0000 100 010
 0001 101 010
 1000 000 010
 1001 001 010

i=3 Substituční tabulka B (omezuje k na 8)

111111 000 010 010
 111110 001 010 010
 011110 101 010 010
 011111 100 010 010

i=4 Substituční tabulka C (omezuje RMTR na 6)

00010001 100 010 010 010
 kanál -- 0 10010001 100 000 010 010
 kanál -- 1 10010001 000 010 010 010

r=4 Substituční tabulka D (omezuje k na 7)

kanál 010 11100000 000 001 010 010
 kanál 010 11100010 100 001 010 010
 kanál 010 11100001 001 010 010 010
 kanál 010 11100011 101 010 010 010

Sync

data: x1 ----- 0x

kanál: xx0 100 000 000 10x (12
 kanálových bitů)

data: x1 -----0x

100-424861-100

Zakončení:

přidat datové bity "01" nebo "11" na začátek
a "00" nebo "01" na konec

Převodní tabulka 3 má strukturu, kde pro minimální běh $d = 1$, maximální běh $k = 7$ a omezení délky $i = 1, 4$ ($= 2^1 m = 2^2$) se vytvoří základní kódy. To znamená, že pro omezení délky $i = 1$ jsou do hlavní tabulky dány 4 základní kódy. Pro omezení délky $i = 2$ nebo více jsou vytvořeny tabulky substitučních kódů pro omezení parametrů, např. minimálního běhu d a maximálního běhu k . Pro bližší vysvětlení, tabulka A pro omezení délky $i = 2$ předepisuje substituční kódy pro omezení minimálního běhu d na 1. Tabulka B pro omezení délky $i = 3$ předepisuje substituční kódy pro omezení maximálního běhu k na horní limit 8. Tabulka C předepisuje pro omezení délky $i = 4$ substituční kódy pro omezení následných výskytů minimálního běhu d s hodnotou 1. Tabulka D předepisuje pro omezení délky $i = 4$ substituční kódy pro omezení maximálního běhu k na horní limit 7. Takže v převodní tabulce 3 je maximální omezení délky $r = 4$.

Jak je popsáno výše, převodní tabulka 3 obsahuje substituční kódy pro omezení následných výskytů minimálního běhu d. Např. datový řetězec (0001001) je přeměněn na řetězec kódových slov "100 010 010 010". Jako pro případ datového řetězce (10010001), je dán odkaz na těsně předcházející kódové slovo pro zjištění, zda těsně

Condition	Control (%)	Mild (%)	Severe (%)
1	65	45	25
2	70	55	35
3	75	65	45
4	85	85	85

100

Navíc kódy v převodní tabulce 3 s omezením délky $i = 4$, která je rovna maximálnímu omezení délky r , jsou substituční kódy pro implementaci hodnoty maximálního běhu $k = 7$. V případě převodu s použitím takového substitučního kódu je dán odkaz na těsně předcházející řetězec kódových slov. Detailněji řečeno, pokud těsně předcházející řetězec kódových slov je "010", je převod proveden. Pokud je např. datový řetězec (11100000) a těsně předcházející řetězec kódových slov je "010", je provedena přeměna, jejímž výsledkem je řetězec kódových slov "000 001 010 010". Jako další příklad, pokud je datový řetěz (11100010) a těsně předcházející řetězec kódových slov je "010", je datový řetězec přeměněn na řetězec kódových slov "100 001 010 010". Převodní tabulka 3 uvedená výše nemůže být vystavěna pouze ze základních kódů, aby se provedlo kódování RLL. Kód RLL se zajištěným minimálním během d a zajištěným maximálním během k může být vytvořen použitím základních kódů z hlavní tabulky, stejně jako substitučních kódů z tabulky A pro omezení délky $i = 2$ a tabulky B pro omezení délky $i = 3$. V tomto případě je maximální omezení délky $r = 3$ a je možné generovat kód, který má minimální běh $d = 1$ a maximální běh $k = 8$. Navíc, zbytek dělení dvěma počtu "1" elementu datového řetězce s hodnotou "0" nebo "1" bude vždy roven zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, který je výsledkem přeměny datového řetězce.

Pokud tabulka C, která předepisuje substituční kódy pro omezení následných výskytů minimálního běhu d , je obsažena v konfiguraci převodní tabulky 3 navíc k hlavní tabulce a tabulkám A a B, je maximální omezení délky $r = 4$ a je možné generovat kód, který má minimální běh $d = 1$, maximální

běh $k = 8$ a omezený počet následných výskytů minimálních běhů d . Navíc, zbytek dělení dvěma počtu "1" elementu datového řetězce s hodnotou "0" nebo "1" bude vždy roven zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, který je výsledkem přeměny datového řetězce. Mělo by se poznamenat, že v tomto případě není vždy potřebné odkazovat se na těsně předcházející řetězec kódových slov, jako je to v případě používajícím tabulku C z tabulky 3.

Pokud tabulka D, která předepisuje substituční kódy pro zabezpečení minimálního běhu $k = 7$, je obsažena v konfiguraci převodní tabulky 3 navíc k hlavní tabulce a tabulkám A a B a tabulce C, která předepisuje substituční kódy pro omezení následných výskytů minimálního běhu d , je maximální omezení délky $r = 4$ a je možné generovat kód, který má minimální běh $d = 1$, maximální běh $k = 7$ a omezený počet následných výskytů minimálních běhů d . Navíc, zbytek dělení dvěma počtu "1" elementu datového řetězce s hodnotou "0" nebo "1" bude vždy roven zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, který je výsledkem přeměny datového řetězce.

Zakončovací tabulka tabulky 2 není při zpracování požadovaná pro zavedení synchronizačního signálu na určitou pozici v řetězci kódových slov (nebo sled kanálových bitů), který je výsledkem přeměny s použitím převodní tabulky 3. To je z toho důvodu, že podle převodní tabulky 3 může být kód zakončený s omezením délky $i = 1$.

Aby se provedlo zavedení obrazce synchronizačního signálu tak, aby byl výsledkem co nejefektivnější kód, je tento obrazec určen následovně. Tříbitové kódové slovo těsně předcházející jedinečný obrazec (nebo řetězec kódových

slov), který slouží jako synchronizační signál, a tříbitové kódové slovo, které těsně následuje jedinečný obrazec, jsou kódová slova, která jsou výsledkem přeměny podle tabulky 3. Tři bity kódových slov, které těsně předcházejí a těsně následují jedinečný obrazec mají vždy takový formát, ve kterém jsou datové bity a spojovací bity smíšeny následovně.

Nejdříve 3 bity těsně předcházející kódové slovo jsou stanoveny následovně. Nejdříve prozkoumáme m -bitovou jednotku datového slova před převodem, kde $m = 2$. První bit 2-bitové jednotky na začátku datového slova před přeměnou je informační bit, zatímco druhý bit má hodnotu "1" pro indikaci synchronizačního signálu. 2-bitová jednotka na začátku datového slova je přeměněna na kódové slovo (kanálové bity) podle tabulky 3. Pro konkrétnost, m bitů (2 bity) datového slova (x_1) jsou přeměněny na n bitů (3 bity) kódového slova " $xx0$ ".

Pak jsou tři bity těsně následující kódové slovo stanoveny následovně. Ze stejného důvodu je zkoumána m -bitová jednotka datového slova před přeměnou, kde $m = 2$. V tomto případě však má první bit dvoubitové jednotky na začátku datového slova před přeměnou hodnotu "0" -ro indikaci synchronizačního signálu, zatímco druhý bit je informační bit. Dvoubitová jednotka na začátku datového slova je přeměněna do kódového slova (kanálových bitů) podle tabulky 3. Pro konkrétnost, m bitů (2 bity) datového slova ($0x$) je přeměněno na n bitů (3 bity) kódového slova " $10x$ ".

Pokud je jedinečný obrazec synchronizačního signálu nastaven jako obrazec, který přerušuje hodnotu 7 maximálního běhu k , obsahuje synchronizační signál obrazec, který může být implementován s nejkratší možnou délkou kódové slovo

"xx0 100 000 000 10x"

•

2

"xx0 100 000 000 100 10x"

"Xxxo 100 000 000 100 000 000 1ox"

■

následné obrazce, kdy každý z nich tvoří maximální běh $k = 8$ (9T). Podle synchronizačního signálu může být zvýšen detekční výkon. Je možné vybrat detekční výkon, který by měl synchronizační signál vytvořit podle požadavků systému.

Podobně jako u konvenční metody, poté, co je datový řetězec přeměněn pomocí převodní tabulky, jako třeba tabulky 2 nebo 3, může být provedeno řízení DSV pomocí přidání bitů řízení DSV v předem stanovených intervalech do sledu kanálových bitů, který je výsledkem přeměny. Pomocí použití vztahu mezi datovým řetězcem a řetězcem kódových slov, který je výsledkem přeměny založené na tabulce 2 a 3 však může být řízení DSV provedeno s vyšším stupněm efektivity.

Abychom to uvedli detailněji, je použito převodní pravidlo, že zbytek dělení dvěma počtu "1" elementu datového řetězce s hodnotou "0" nebo "1" bude vždy roven zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, který je výsledkem přeměny datového řetězce. Takže zavedení bitů řízení DSV s hodnotou "1" pro indikaci inverze a s hodnotou "0" pro indikaci ne-inverze do sledu kanálových bitů je ekvivalentní zavedení bitů řízení DSV s hodnotou "1" pro indikaci inverze a s hodnotou "0" pro indikaci ne-inverze do sledu datových bitů.

Uvažujme například případ, ve kterém jsou 3 bity (001) dat přeměněny podle tabulky 2. Nechť bit řízení DSV x následující dané 3 bity je umístěn mezi data. V tomto případě data včetně bitu řízení DSV budou (001- x), kde x je 1-bitový řídicí bit, který má hodnotu "0" nebo "1". Pro $x = 0$ jsou data včetně bitu řízení DSV přeměněna podle tabulky 2 následovně:

datového řetězce odpovídá provedení řízení DSV zavedením 1,5 bitu řízení DSV do sledu kanálových bitů, kde hodnota 1,5 je reciproká k poměru přeměny $m/n = 2/3$ tabulek 2 a 3. Aby se provedlo řízení DSV pro tabulka RLL (1-7), jak je znázorněno v tabulce 1, je potřebné použít řízení ve sledu kanálových bitů. V tomto případě jsou požadovány alespoň 2 kanálové bity, aby se udržel minimální běh d tím, že se zvýší relativní redundance ve srovnání s řízením DSV aplikovaného na datový řetězec pro tabulky 2 a 3. Jinými slovy, v tomto systému může být pomocí provedení řízení DSV v datovém řetězci zlepšena efektivita řízení DSV.

Dále je s odkazem na obr. 1 vysvětleno provedení modulátoru podle tohoto vynálezu. V tomto provedení je datový řetězec přeměněn do kódu s proměnnou délkou $(d, k; m, n; r) = (1, 7; 2, 3; 4)$ pomocí tabulky 2.

Jak je znázorněno na obr. 1, obsahuje modulátor 1 jednotku 11 pro stanovení/vkládání bitu řízení DSV, která má za úkol stanovit, zda je hodnota bitu řízení DSV "1" nebo "0" a pro vkládání bitu řízení DSV v jakýchkoliv volitelných intervalech do datového řetězce, který je na ní přiveden, modulační jednotku 12, která moduluje datový řetězec bity DSV, které jsou do něho zavedeny a kódovací NRZI jednotku 13 pro přeměnu výstupu modulační jednotky 12 do záznamového sledu vln. Navíc obsahuje modulátor 1 také jednotku řízení časování 14 pro generování časovacích signálů a vedení těchto signálů k různým částem.

Na obr. 2 je schéma vysvětlující zpracování, které je prováděno jednotkou 11 pro stanovení/vkládání bitu řízení DSV. Jak je na obrázku znázorněno, stanoví se hodnoty bitů řízení DSV a bity řízení DSV jsou vloženy do datového

řetězce v libovolných intervalech. Aby se vložil bit řízení DSV například na pozici mezi částí DATA1 a DATA2 příchozího datového řetězce, vypočte jednotka 11 pro stanovení/vkládání řídicího bitu kumulativní DSV až k DATA1. Celkové DSV se vypočte provedením kroků:

- přeměny DATA1 do sledu kanálových bitů;
- modulace NRZI sledu bitů;
- přiřazení hodnoty +1 hladině H (high) (1) a hodnoty -1 hladině L (low) (0) výsledku modulace NRZI a
- sečtení hodnot, které jsou přiřazeny výsledkům modulace NRZI.

Podle stejného principu vypočítá jednotka 11 pro stanovení/vkládání bitu řízení DSV celkové DSV pro část DATA2 následující DATA1. Nechť x_1 je bit řízení DSV, který má být vložen na pozici mezi částí dat DATA1 a DATA2. Jednotka 11 pro stanovení/vkládání bitu řízení DSV stanoví hodnotu bitu řízení DSV x_1 tak, že absolutní hodnota součtu DSV pro DATA1, x_1 a DATA2 se blíží nule.

Pokud je bit řízení DSV x_1 nastaven na (1), jsou hladinové kódy části DATA2 následující DATA1 invertovány. Pokud je naopak bit řízení DSV x_1 nastaven na (0), hladinové kódy části DATA2 následující DATA1 invertovány nejsou. To je z důvodu, že v každém elementu převodních tabulek 2 a 3 bude zbytek dělení dvěma počtu "1" v datovém řetězci s hodnotou 0 nebo 1 vždy roven zbytku dělení dvěma počtu "1" elementu v řetězu kódových slov, který je výsledkem přeměny daného datového řetězce. Takže vložení bitu (1) do datového řetězce musí být doprovázeno vložení "1" do řetězce kódových slov, který je výsledkem přeměny daného datového řetězce, ve kterém se provádí inverze.

Poté, co byla stanovena hodnota bitu řízení DSV x1 z obr. 2 způsobem popsaným výše, je bit řízení DSV x2 vložen mezi DATA2 a DATA3 tak, že se stejným způsobem provede řízení DSV s předem stanoveným datovým intervalem mezi x1 a x2. V tomto případě je kumulativní DSV součtem kumulativních DSV pro data až k DATA1, hodnota DSV pro x1 a DSV data pro část DATA2.

Jak je popsáno výše, bity řízení DSV jsou vloženy do datového řetězce předtím, než je datový řetězec modulován modulační jednotkou 12 pro generaci sledu kanálových bitů.

Na obr. 3 je blokové schéma, které představuje typickou konfiguraci modulační jednotky 12. Jak obrázek ukazuje, posuvný registr 31 posouvá data v něm uložená o dva bity najednou, svůj výstup vysílá na jednotku stanovení omezení délky 32, jednotku detekce kódu omezení výskytu následních minimálních běhů 33, jednotku detekce kódu zabezpečení omezení délky běhu 34 a všechny převodní jednotky 35-1 až 35-4. Posuvný registr 31 přivádí tolik bitů, kolik je jich požadováno, na zpracování do každé ze součástí 32 až 35.

Jednotka stanovení omezení délky 32 stanovuje omezení délky i dat a vede délku i na multiplexer 36. Když jednotka detekce kódu omezení výskytu následních minimálních běhů 33 detekuje zvláštní datové slovo, které závisí na omezení následného výskytu minimálního běhu, jednotka detekce kódu omezení výskytu následních minimálních běhů 33 vede příslušný detekční signál (i=3) pro indikaci omezení délky i jednotce stanovení omezení délky 32. V případě tabulky 2 je zvláštním datovým slovem (110111). Ze stejného důvodu, když jednotka detekce kódu zabezpečení omezení délky běhu 34 detekuje zvláštní datové slovo, které vyžaduje

zabezpečení maximálního běhu k , přivede jednotka detekce kódu zabezpečení omezení délky běhu 34 příslušný detekční signál ($i=4$), který indikuje omezení délky i , na jednotku stanovení omezení délky 32. V případě tabulky 2 je zvláštním datovým slovem (00001000) nebo (00000000).

Když je detekováno jednotkou detekce kódu omezení výskytu následních minimálních běhů 33 nebo jednotkou detekce kódu zabezpečení omezení délky běhu 34, předává jednotka stanovení omezení délky 32 dál omezení délky i zvláštního datového slova na multiplexer 36. V tom okamžiku může jednotka stanovení omezení délky 32 stanovit také další hodnotu omezení délky sama. V tom případě však jednotka stanovení omezení délky 32 dá přednost omezení délky, která je dodaná jednotkou detekce kódu omezení výskytu následních minimálních běhů 33 nebo jednotkou detekce kódu zabezpečení omezení délky běhu 34, před tou, kterou stanovila sama. Jinými slovy, je vybráno větší omezení délky.

Každá z převodních jednotek 35-1 až 35-4 vytváří rozhodnutí, zda základní kód pro data, které jsou na ně přiváděna, je obsažen v převodních tabulkách, které jsou v nich obsaženy či nikoliv. Pokud se shledá, že základní kód je součástí katalogu, jsou data přeměněna na kódové slovo, které je reprezentováno daným základním kódem a kódové slovo, které je výsledkem přeměny je vedeno na multiplexer 36. Pokud naopak základní kód součástí převodní tabulky netvoří, pak převodní jednotky 35-1 až 35-4 data neuvažují.

Měli bychom poznamenat, že z důvodu, že modulátor 12 je navržen pro převodní tabulku 2 je každá z konverzních jednotek 35-i navržena pro přeměnu dat s omezením délky i do

hodnoty 4. To znamená, že každá z převodních jednotek 35-1 je navržena pro převod dat s maximálním omezení délky $r = 4$.

Multiplexer 36 vybírá kód, který je výsledkem přeměny, která je provedena jednou z převodních jednotek 35-i odpovídající omezení délky i , která je poskytována jednotkou stanovení omezení délky 32. Vybraný kód pak vystupuje pomocí bufferu 37 jako sériová data.

Operace časování všech součástí je řízeno synchronizací s časovacími signály, které jsou generovány jednotkou řízení časování 14.

Dále je popsána činnost provedení.

Nejdříve přivádí posuvný registr takový počet bitů, který je požadovaný zpracováním, jakým je vytváření posouzení v dvoubitových jednotkách na jednotku stanovení omezení délky 32, jednotka detekce kódu omezujícího výskyt následných minimálních běhů, jednotka detekce kódu zabezpečující omezení délku běhu 34 a všechny převodní jednotky 35-1 až 35-4.

S využitím převodní tabulky, jakou je tabulka 2, stanovuje jednotka stanovení omezení délky 32 hodnotu omezení délky i pomocí odkazu na převodní tabulku a přivedení hodnoty na multiplexer 36.

V jednotce detekce kódu omezení výskytu následných minimálních běhů 33 jsou obsažena data, která mají být nahrazena substitučním kódem pro omezení následných výskytů minimálního běhu d z tabulky 2 za předpokladu, že následujícím kódovým slovem je "010". V případě tabulky 2 je datovým slovem (110111). Pokud jsou detekována data požadující omezení následných výskytů minimálního běhu d jako výsledek odkazu na tuto část převodní tabulky, tak

jednotka detekce kódu omezení výskytu následních minimálních běhů 33 přivede na výstup detekční signál, který indikuje, že omezení délky $i = 3$, který je veden dále na jednotku stanovení omezení délky 32.

V jednotce detekce kódu zabezpečující omezení délku běhu 34 jsou obsažena data, která mají být nahrazena substitučním kódem pro omezení následných výskytů minimálního běhu d z tabulky 2. V případě tabulky 2 jsou datová slova (00001000) a (00000000). Když jsou detekována data požadující omezení následných výskytů minimálního běhu d jako výsledek odkazu na tuto část převodní tabulky, tak jednotka detekce kódu zabezpečující omezení délku běhu 34 přivede na výstup detekční signál, který indikuje, že omezení délky $i = 4$, který je veden dále na jednotku stanovení omezení délky 32.

Pokud jednotka stanovení omezení délky 32 přijme od jednotky detekce kódu omezení výskytu následních minimálních běhů 33 detekční signál, který indikuje, že omezení délky $i = 3$ v případě tabulky 2, vede hodnotu $i = 3$ na multiplexer 36, i když v tom okamžiku jednotka stanovení omezení délky 32 stanoví hodnotu omezení délky i sama namísto předání hodnoty, kterou sama stanovila na multiplexer 36. Ze stejného důvodu, když jednotka stanovení omezení délky 32 přijme od jednotky detekce kódu zabezpečující omezení délku běhu 34 detekční signál, který indikuje, že omezení délky $i = 4$ v případě tabulky 2, vede hodnotu $i = 4$ na multiplexer 36, i když v tom okamžiku jednotka posouzení omezení délky 32 stanoví hodnotu omezení délky i sama namísto předání hodnoty, kterou sama stanovila na multiplexer 36.

To znamená, že jednotka stanovení omezení délky 32

Figure 1 consists of four diagrams labeled (a), (b), (c), and (d). Each diagram shows a central point x surrounded by a grid of points. The number of points in the neighborhood of x is indicated by the number of points in the grid. In (a), the neighborhood contains 1 point. In (b), the neighborhood contains 2 points. In (c), the neighborhood contains 3 points. In (d), the neighborhood contains 4 points.

—

•

—

které je výsledkem přeměny následující datové slovo je "010", vyše jednotka detekce kódu omezení výskytu následných minimálních běhů 33 jednotce stanovení omezení délky 32 detekční signál, který indikuje, že omezení délky $i = 3$. Mělo by se poznamenat, že 3-bitové kódové slovo je výsledkem přeměny datového řetězce s hodnotou (01), (001) nebo (00000) před přeměnou. Jinými slovy, funkce obsahuje datový řetězec (110111) + (01 / 001 / 00000). Když je detekováno 6-bitové slovo, které odpovídá datovému slovu (110111), jsou data až do 5 bitů následující 6-bitová data porovnána s datovým slovem (01) nebo (001) nebo (00000) pro stanovení, zda si odpovídají. Pokud jsou příchozí data (11011101), (11011001) nebo (11011100000), vyše jednotka detekce kódu omezení výskytu následných minimálních běhů 33 jednotce stanovení omezení délky 32 detekční signál, který indikuje, že omezení délky $i = 3$.

V jednotce stanovení omezení délky 32 jsou obsaženy datové řetězce z převodní tabulky 2. Když jsou 6-bitová data, která odpovídají datovým slovům (000011), (000010), (000001) nebo (000000), přivedena na jednotku posouzení omezení délky 32, určí tato jednotka, že hodnota omezení délky $i = 3$. Když jsou 4-bitová data, která odpovídají datovým slovům (0011), (0010) nebo (0001) přivedena na jednotku stanovení omezení délky 32, určí tato jednotka, že hodnota omezení délky $i = 2$. Když jsou 2-bitová data, která odpovídají datovým slovům (11), (10) nebo (01), přivedena na jednotku stanovení omezení délky 32, určí tato jednotka, že hodnota omezení délky $i = 1$.

Předpokládejme, že je přivedeno 6 bitů dat (000010).
V tomto případě určí jednotka stanovení omezení délky 32, že

hodnota omezení délky je 3. Předpokládejme též, že těmto 6 bitům dat následují 2 bity dat (00). Výsledkem je, že na jednotku detekce kódu zabezpečující omezení délku běhu 34 je přivedeno 8-bitové datové slovo (00001000), takže jednotka detekce kódu zabezpečující omezení délku běhu 34 vyše jednotce stanovení omezení délky 32 detekční signál, který indikuje, že omezení délky $i = 4$. V tomto případě dá jednotka stanovení omezení délky 32 přednost detekčnímu signálu, který nese hodnotu 4 z jednotky detekce kódu zabezpečující omezení délku běhu 34, před hodnotou 3, kterou stanovila sama a tak určí hodnotu omezení délky $i = 4$.

Jak je popsáno výše, může být omezení délky dat, která obsahují sled bitů (1) a (0), stanoveno podle převodní tabulky 2 s odkazem na přivedené datové slovo s délkou až 8 bitů, což odpovídá maximálnímu omezení délky, a pokud je to zapotřebí na 3-bitové kódové slovo. Alternativně, omezení délky dat obsahující sled bitů (1) a (0) může být stanoveno pouze pomocí odkazu na přivedené datové slovo s maximální délkou 11 bitů.

Jednotka stanovení omezení délky 32 vysílá hodnotu omezení délky i stanovenou tímto způsobem na multiplexer 36.

Mělo by se poznamenat, že jednotka stanovení omezení délky 32 může též stanovit hodnotu omezení délky i ve vzestupné řadě hodnot i , která začíná nejmenší hodnotou, tzn. v pořadí $i=1$, $i=2$, $i=3$ a $i=4$, což je v protikladu k příkladu z obr. 4.

Každá z převodních jednotek 35-1 až 35-4 má převodní tabulku, která odpovídá hodnotě omezení délky, která je jí přiřazena. Detailněji řečeno, převodní jednotky 35-1 až 35-4 mají převodní tabulky pro $i=1$, $i=2$, $i=3$, resp. $i=4$.

Pokud je převodní pravidlo pro data přivedená na kteroukoliv z převodních jednotek 35-1 až 35-4 obsaženo v katalogu v tabulkách převodních jednotek, je $2 \times i$ bitů přivedených dat přeměněno na $3 \times i$ bitů kódu podle katalogového převodního pravidla. Výsledný kód je pak vyslán na multiplexer 36.

Multiplexer 36 vybírá kód, který je výsledkem přeměny provedené jednou z převodních jednotek 35-i odpovídajícím omezení délky i , vyslané na jednotku stanovení omezení délky 32. Vybraný kód je pak vyslán pomocí vyrovnávací paměti (bufferu) 37 jako sériová data.

Jak je uvedeno v tabulce 2, pro omezení délku $i = 3$ převodní tabulka neobsahuje substituční kód pro datový řetězec (110111), což vyžaduje omezení na následném opakovaném výskytu minimálního běhu d . Předpokládejme, že je přiveden následující datový řetězec:

(1101110111011101).

V tomto případě se přeměna provede v následujícím pořadí datových slov: (11), (01), (11), (01) atd. Výsledkem přeměny je generování následného řetězce kódových slov (sled kanálových bitů):

"101 010 101 010 101 010 101 010"

Pak je obvykle na generovaný řetězec kódových slov aplikovaná modulace NRZI, aby se provedlo hladinové kódování. Protože logická inverze se v signálu uskutečňuje spolu s časováním "1", je výše zmíněné kódové slovo přeměněno na následující řetězec kódových slov:

"110 011 001 100 110 011",

kde minimální intervaly inverze $2T$ probíhají podél řetězce. Když je takový řetězec zaznamenán nebo reprodukován

s vysokou lineární hustotou, vytvoří se obrazec, který snadno způsobí při záznamu nebo reprodukci chybu.

Předpokládejme, že převodní tabulka 2 také předepisuje substituční kód pro datový řetězec (110111), což požaduje omezení na opakovaných následných výskytech minimálního běhu d. Nyní uvažujme, že je přiveden následující datový řetězec:

(1101110111011101).

V tomto případě obsahuje první datové slovo (11011101) v datovém řetězci datové slovo (110111), které je následováno datovým slovem (01), které bude přeměněno na řetězec kódových slov "010". Pak je první datové slovo přeměněno na následující řetězec kódových slov:

"001 000 000 010"

Podobně druhé datové slovo (11011101) v datovém řetězci také obsahuje datové slovo (110111), které je následováno datovým slovem (01), které bude přeměněno na řetězec kódových slov "010". Pak je první datové slovo přeměněno na následující řetězec kódových slov:

"001 000 000 010 001 000 000 010.. ",

kde je zamezeno opakovaným následným výskytům minimálního běhu d. To znamená, že je eliminován obrazec, který snadno způsobuje chybu při záznamu nebo reprodukci při vysoké lineární hustotě. Mělo by se poznamenat, že při přeměně datového řetězce na řetězec kódových slov popsané výše, jsou drženy hodnoty minimálního běhu d a maximálního běhu k na svých příslušných hodnotách.

Jak je popsáno výše, převod provedený modulátorem 1 je založen na převodní tabulce 2. Měli bychom poznamenat, že převod může být proveden též pomocí převodní tabulky 3. V tomto případě obsahuje jednotky detekce kódu omezení

1

4

•

1

•

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 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100

vynálezu s odkazem na obr. 5. V tomto provedení je zpětně demodulován kód s proměnnou délkou $(d,k;m,n;r) = (1,7;2,3;4)$ na datový řetězec pomocí tabulky 2.

Jak ukazuje obr. 5, demodulátor 100 obsahuje demodulační jednotku 111 pro demodulaci signálu přijatého z přenosové cesty nebo signálu reprodukováného ze záznamového média, pomocí demodulační tabulky nebo inverzní převodní tabulky a jednotky odstranění bitů řízení DSV 112 pro odstranění bitů řízení DSV, které byly zavedeny do datového řetězce, který je výsledkem demodulace v libovolných intervalech z datového řetězce, aby se obnovil původní datový řetězec. Vyrovnávací paměť (buffer) 113 je použita pro dočasné ukládání sériových dat, která jsou generována jednotkou odstranění bitů řízení DSV 112. Data uložená v bufferu 113 jsou vyčítána později předem stanovenou přenosovou rychlostí pro vytvoření výstupního signálu. Jednotka řízení časování 114 generuje časovací signály a vysílá je na různé součásti pro řízení časování jejich činností.

Na obr. 6 je blokové schéma, které ukazuje uspořádání demodulační jednotky 111. Jak ukazuje obrázek, demodulační jednotka 111 obsahuje komparátor 121 pro přeměnu signálu přijatého z přenosové cesty nebo signálu reprodukováného ze záznamového média na binární data. Pokud signál, který je přiváděn na komparátor 121, je signál modulovaný NRZI (tzn. hladinovým kódováním), projde tento signál inverzním kódovacím procesem NRZI (tzn. hranový kódovací proces). Jednotka stanovení omezení délky 122 stanovuje omezení délku i digitálního signálu přijímaného komparátorem 121. Když jednotka detekce kódu omezení výskytu následných minimálních

běhů 123 detekuje zvláštní kód pro omezení následných výskytů minimálního běhu d v digitálním signálu generovaném komparátorem 121, vyšle jednotka detekce kódu omezení výskytu následných minimálních běhů 123 na jednotku stanovení omezení délky 122 detekční signál ($i=3$), který indikuje omezení délky. V případě tabulky 2 je zvláštním kódem "001 000 000". Podle stejného principu, když jednotka detekce kódu zabezpečující omezení délku běhu 124 detekuje zvláštní kód pro zabezpečení maximálního běhu k , vyšle jednotka detekce kódu zabezpečující omezení délku běhu 124 na jednotku stanovení omezení délky 122 detekční signál ($i=4$) indikující omezení délky. V případě tabulky 2 je zvláštním kódem "000 100 100 100" nebo "010 100 100 100".

Každá z inverzních převodních jednotek 125-1 až 125-4 obsahuje tabulku, kterou používá pro inverzní přeměnu $n \times i$ bitový kód s proměnnou délkou zpět na $m \times i$ bitová data. V případě tabulky 2 obsahují inverzní převodní jednotky 125-1 až 125-4 inverzní převodní tabulky pro omezení délky $i=1$ až 4 resp., které jsou v zásadě stejné jako převodní tabulky obsažené v převodních jednotkách 35-1 až 35-4, které byly popsány dříve. Multiplexer 126 vybírá jeden z výstupů, které jsou generované inverzními převodními jednotkami 125-1 až 125-4 v závislosti na výsledku přijatého z jednotky stanovení omezení délky 122.

Dále je vysvětlena činnost demodulační jednotky 111, která je znázorněna na obr. 6. Signál přijatý z přenosové cesty nebo signál reprodukováný ze záznamového média je veden na komparátor 121. Signál na výstupu komparátoru 121 je digitální signál inverzního kódu NRZI, tzn. "1" indikuje hranu. Digitální signál je pak vyslán na jednotku stanovení

omezení délky 122 pro stanovení omezení délky i signálu použitím převodní tabulky 2 (přesně řečeno, inverzní převodní tabulky). Výsledek, tzn. hodnota omezení délky i, která je výstupem jednotky stanovení omezení délky 122, je veden na multiplexer 126.

Dále je výstupní digitální signál komparátoru 121 vyslán také na jednotku detekce kódu omezení výskytu následných minimálních běhů 123. Jednotka detekce kódu omezení výskytu následných minimálních běhů 123 obsahuje inverzní převodní část, která obsahuje substituční kód inverzní tabulky 2 pro omezení následných výskytů minimálního běhu d. V případě tabulky 2 je substitučním kódem kódové slovo "001 000 000". Když je detekován kód "001 000 000 not 100", který je v katalogu inverzní převodní tabulky pro omezení následných výskytů minimálního běhu d z digitálních dat, vyšle jednotka detekce kódu omezení výskytu následných minimálních běhů 123 na jednotku stanovení omezení délky 122 hodnotu omezení délky i=3.

Dále je výstupní digitální signál komparátoru 121 vyslán také na jednotku detekce kódu zabezpečující omezení délku běhu 124. Jednotka detekce kódu zabezpečující omezení délku běhu 124 obsahuje inverzní převodní části, která obsahuje substituční kódy inverzní tabulky 2 pro udržení maximálního běhu k. V případě tabulky 2 jsou substitučními kódy kódová slova "000 100 100 100" a "010 100 100 100". Když je některé z kódových slov "000 100 100 100" a "010 100 100 100", které jsou v katalogu inverzní převodní tabulky pro udržení maximálního běhu k, detekováno v digitálních datech, vyšle jednotka detekce kódu zabezpečující omezení délku běhu 124 na jednotku stanovení

omezení délky 122 hodnotu omezení délky $i=4$.

Na obr. 7 je celkové schéma stanovení omezení délky i modulovaného kódu, který je přiváděn na demodulátor 100. Jak ukazuje obrázek, jednotka detekce kódu zabezpečující omezení délku běhu 124 obsahuje inverzní převodní část, která obsahuje kódová slova "000 100 100 100" a "010 100 100 100" z inverzní tabulky 2. Když 12-bitový řetězec kódových slov, který je přiveden na jednotku detekce kódu zabezpečující omezení délku běhu 124 odpovídá jednomu z daných kódových slov v inverzní převodní část, vyšle jednotka detekce kódu zabezpečující omezení délku běhu 124 na jednotku stanovení omezení délky 122 omezení délky 122.

Podle stejného principu, jednotka detekce kódu omezení výskytu následných minimálních běhů 123 obsahuje inverzní převodní část obsahující kódové slovo "001 000 000" z inverzní převodní tabulky 2. Když 12-bitový řetězec kódových slov, který je přiveden na jednotku detekce kódu omezení výskytu následných minimálních běhů 123 odpovídá "001 000 000 not 100", vyšle jednotka detekce kódu omezení výskytu následných minimálních běhů 123 na jednotku stanovení omezení délky 122 omezení délky $i=3$. Měli bychom poznamenat, že detekovaných 12 bitů řetězce kódových slov je vlastně "001 000 000 010", ačkoliv to nemá v zásadě nic společného se stanovením omezení délky.

Jednotka stanovení omezení délky 122 obsahuje inverzní převodní tabulku 2. Pokud je přivedeno na jednotku stanovení omezení délky 122 9-bitový řetězec kódových slov "000 100 100" nebo "010 100 100" nebo pokud je na ni přivedeno 12-bitové kódové slovo "000 100 000 not 100" nebo "010 100 000 not 100", jednotka stanovení omezení délky 122 stanoví

omezení délky na $i=3$. Pokud je na jednotku stanovení omezení délky 122 přivedeno 6-bitový řetězec kódových slov "010 100" nebo "000 100", nebo pokud je na ní přiveden 9-bitový řetězec kódových slov "010 000 not 100", stanoví jednotka stanovení omezení délky 122 hodnotu omezení délky na $i = 2$. Jinak pokud je na jednotku stanovení omezení délky 122 přiveden 3-bitový řetězec kódových slov "000", "101", "001" nebo "010", stanoví jednotka stanovení omezení délky 122 hodnotu omezení délky na $i=1$.

Měli bychom poznamenat, že jednotka stanovení omezení délky 122, jednotka detekce kódu omezení výskytu následných minimálních běhů 123 i jednotka detekce kódu zabezpečující omezení délku běhu 124 mohou též provádět zpracování ve vzestupné řadě hodnot i s počátkem na nejmenší, tzn., v řadě $i=1$, $i=2$, $i=3$ a $i=4$, což je v protikladu s příkladem z obr. 7.

Předpokládejme, že jednotka stanovení omezení délky 122 může též stanovit omezení délky v pořadí $i=2/i=3$ a $i=4$ a na jednotku stanovení omezení délky 122 je přiveden řetězec kódových slov "000 100 100 100". Jednotka stanovení omezení délky 122 porovná řetězec kódových slov, který je na ní přiveden s kódovým slovem v převodní tabulce ve vzestupném pořadí hodnot omezení délky i s počátkem na nejmenší hodnotě a vytvoří rozhodnutí, zda ano či nikoliv řetězec kódových slov odpovídá kódovým slovům. Řetězec kódových slov "000 100 100 100", který je přiveden na jednotku stanovení omezení délky 122 odpovídá jednomu z kódových slov pro všechny omezení délky $i=1$, $i=2$, $i=3$ a $i=4$. V takovém případě se vybere podle pravidla největší omezení délky a je vyslána na multiplexer 126.

Inverzní převodní tabulka inverzní převodní jednotky 125-1 je provedena jako paměť, ve které je úsek dat (11) uložen na adresách "101" a "000", zatímco úseky dat (10) a (01) jsou uloženy na adresách "001", resp. "010". Inverzní převodní tabulky inverzních převodních jednotek 125-2 a 125-4 jsou vždy provedeny jako paměti pro uchovávání dat stejným způsobem, jako v případě inverzní převodní jednotky 125-1. Řetězec kódových slov ze 3 x i bitů přivedený na inverzní převodní jednotku 125-i je zpětně přeměněn na datový řetězec 2 x i bitů, který je pak vyslán na multiplexer 126.

Multiplexer 126 vybírá jeden z datových řetězců, který je dodán inverzními převodními jednotkami 125-1 až 125-4 podle výsledku stanovení hodnoty omezení délky i na výstupu jednotky stanovení omezení délky 122.

Tabulka 4 je inverzní převodní tabulka k tabulce 2.

<Tabulka 4>

Inverzní převodní tabulka (1,7;2,3;4)

Řetězec kódového slova		Demodulovaný řetězec dat
i=1	101	11
	000	11
	001	10
	010	01
i=2	010 100	0011

	010 000 (not 100)	0010
	000 100	0001
i=3	000 100 100	000011
	000 100 000 (not 100)	000010
	010 100 100	000001
	010 100 000 (not 100)	000000

i=3: Zákaz minimální přechodné délky běhu

001 000 000 (not 100)	110111
-----------------------	--------

i=4: Limituje k na 7

000 100 100 100	00001000
010 100 100 100	00000000

Na obr. 8 je vývojový diagram, který je použit jako odkaz na vysvětlení činnosti, která je prováděna jednotkou odstraňování bitu řízení DSV 112. Jednotka odstraňování bitu řízení DSV 112 obsahuje vnitřní čítač. Jak je znázorněno na obrázku, začíná vývojový diagram v kroku S1, při kterém je čítán vnitřním čítačem počet bitů v datovém řetězci, který je vysílán demodulační jednotkou 111. Zpracování dále pokračuje v kroku S2, kde se rozhoduje, zda počet bitů přesáhl hodnotu, která reprezentuje předem stanovený datový interval, ve kterém jsou zaváděny bity řízení DSV. Pokud výsledek rozhodování říká, že počet bitů neodpovídá volitelnému datovému intervalu, pokračuje proces na krok S3, kde jsou data vysílána demodulační jednotkou 111 předávána na buffer 113 beze změny. Pokud výsledek rozhodování říká, že aktuální bit je bitem řízení DSV, je

A 5x5 grid of dots forming the letters 'L', 'U', and 'U'. The first row has dots at (1,1), (1,2), (1,3), (1,4), and (1,5). The second row has dots at (2,1), (2,2), (2,3), (2,4), and (2,5). The third row has dots at (3,1), (3,2), (3,3), (3,4), and (3,5). The fourth row has dots at (4,1), (4,2), (4,3), (4,4), and (4,5). The fifth row has dots at (5,1), (5,2), (5,3), (5,4), and (5,5).

•

•

.

1

—

r=1 Hlavní tabulka

101	00
100	01
001	10
000	11

r=2 Substituční tabulka A (omezuje d na 1)

100 010	0000
101 010	0001
000 010	1000
001 010	1001

r=3 Substituční tabulka B (omezuje k na 8)

000 010 010	111111
001 010 010	111110
101 010 010	011110
100 010 010	011111

r=4 Substituční tabulka C (omezuje RMTR na 6)

100 010 010 010	00010001
100 000 010 010	10010001
000 010 010 010	10010001

r=4 Substituční tabulka D (omezuje k na 7)

000 001 010 010	11100000
100 001 010 010	11100010
001 010 010 010	11100001
101 010 010 010	11100011

Existují případy, při kterých je nutné vložit do dat synchronizační signál (Sync). Dále jsou spolu s odkazem na obr. 9, resp. 10 popsána provedení modulátoru 1, resp. demodulátoru 100, která mohou zpracovávat data se synchronizačními signály. Také v případě těchto provedení je datový řetězec modulován na kód s proměnnou délkou $(d,k;m,n;r)=(1,7,;2,3;4)$.

V dalším provedení modulátoru podle tohoto vynálezu, které je znázorněno na obr. 9, kde jsou vloženy synchronizační signály na předem stanovených intervalech, je vysílán výstup jednotky stanovení/zavádění bitu řízení DSV 11 na jednotku stanovení synchronizačního signálu 211. Dále je na jednotku stanovení synchronizačního signálu 211 přiváděn výstup modulační jednotky 12. Když jednotka stanovení synchronizačního signálu 211 detekuje synchronizační signál v signálu, který je přiváděn od jednotky stanovení/vkládání bitu řízení DSV 11 a modulační jednotky 12, vyśle jednotka stanovení synchronizačního signálu 211 na jednotku vkládání synchronizačního signálu 212 synchronizační signál. Jednotka vkládání synchronizačního signálu 212 vkládá synchronizační signál, který je přiváděn od jednotky stanovení synchronizačního signálu 211 do modulovaného signálu, který je přiváděn od modulační jednotky 12, a vysílá svůj výstup na jednotku

kódování NRZI 13. Zbytek uspořádání je stejný jako v případě modulátoru 1 z obr. 1.

V případě obrazce 24-bitového kódového slova, který slouží jako synchronizační signál, je synchronizační signál přeměněn jednotkou stanovení synchronizačního signálu 211 podle tabulky 2 na následující kód:

"#01 001 000 000 001 000 000 001"

kde znak # označuje bit, který je závislý na těsně předcházejícím datovém řetězci, včetně bitu řízení DSV, pokud existuje, ohraničený vložením synchronizačního signálu. Pro uvedení konkrétních hodnot: když je použita zakončovací tabulka pro zakončení úkonu modulace vymezeného datového řetězce použitím převodní tabulky, "#" = "1". Když se namísto zakončovací tabulky použije pro zakončení tabulka 2, "#" = "0". Takže modulační jednotka 12 vysílá na jednotku stanovení synchronizačního signálu 211 "#" = "1" nebo "#" = "0", když se zakončovací tabulka použije, resp. nepoužije. Po přijetí hodnoty "#" od modulační jednotky 12 přidá jednotka stanovení synchronizačního signálu 211 hodnotu "#" na začátek synchronizačního signálu a pak vyšle synchronizační signál na jednotku vkládání synchronizačního signálu 212.

Jednotka vkládání synchronizačního signálu 212 vkládá synchronizační signál, který je přijímán od jednotky stanovení synchronizačního signálu 211, do modulovaného signálu, který je přijímán od modulační jednotky 12 a vysílá svůj výstup na jednotku kódování NRZI 13. Zbytek zpracování je identický jako v případě modulátoru 1 z obr. 1.

První data, která následují vložený synchronizační signál se začnou převádět od jejich hlavičky bez uvážení

"Synchronizační"

2

2

■

•

.....
kódu, jenž je výsledkem modulace, která byla provedena modulátorem 1 z obr. 9. Jak ukazuje obr. 10, v tomto provedení je příchozí signál, který je přenášen předem stanovenou přenosovou cestou, přiváděn na demodulační jednotku 111 a jednotku identifikace synchronizačního signálu 221. Jednotka identifikace synchronizačního signálu 221 používá vstupní signál a signál přijatý z demodulační jednotky 111 pro identifikaci synchronizačního signálu, vysílá synchronizační signál na jednotky odstraňování synchronizačního signálu 222. Jednotka odstraňování synchronizačního signálu 222 odstraňuje synchronizační signál z demodulovaného signálu, který je dodáván demodulační jednotkou 111, podle signálu, který přichází od jednotky identifikace synchronizačního signálu 221. Demodulovaný signál bez synchronizačních signálů je pak vyslán na jednotku odstraňování bitu řízení DSV 112. Zbytek uspořádání je stejný jako u demodulátoru 100 z obr. 5.

Jednotka identifikace synchronizačního signálu 221 obsahuje čítač pro čítání počtu kódových slov. Obsah čítače je používán pro stanovení polohy každého synchronizačního signálu, který je vložen do řetězce datových slov v předem stanovených intervalech. Po identifikaci polohy obrazce synchronizačního signálu se přečte bit "#", který je stanovený při modulaci. To znamená, že bit na začátku synchronizačního signálu je přečten a vyslán na demodulační jednotku 111. Pokud je prvním bitem "1", použije demodulační jednotka 111 zakončovací tabulku 2 pro demodulaci kódu, který těsně předchází synchronizační signál. Pokud je první bit "0", použije demodulační jednotka 111 tabulku převodních kódů 2 pro demodulaci kódu, který těsně předchází

•

•

2

signálu 222. Tak může jednotka odstraňování synchronizačního signálu 222 odstranit pouze bity synchronizačního signálu, tzn. bity v jedinečném obrazci, který je specifikován signálem přijatým z jednotky identifikace synchronizačního signálu 221.

Na obr. 11 je schéma, které ukazuje příklad kódu pro záznam se synchronizačními signály a bity řízení DSV, které jsou do nich zavedeny. V tomto příkladu je použito jako synchronizačního signálu 24-bitové kódové slovo. Řízení DSV je prováděno v intervalech 56 datových bitů a synchronizační signály jsou vkládány vždy po provedení 5 řízení DSV. Takže počet kódových slov, tzn. počet kanálových bitů na každý synchronizační signál je: $24 + (1 + 56 + 1 + 56 + 1 + 56 + 1 + 56 + 1) \times 1,5 = 453$ kódových slov (kanálových bitů). Relativní redundance, která je zavedena do datových slov je kolem 7,3%, jak je zřejmé z následujícího výpočtu:

$$\text{Množství dat} = (56 \times 5) \times 1,5 - 453 = 420 - 453 = -0,927$$

$$\text{Takže relativní redundance} = 1 - 0,927 = 0,0728 = 7,3\%.$$

Autoři vynálezu a další provádějí simulace, při kterých se využívají převodní výše uvedené tabulky, aby se získaly výsledky modulace. Výsledky modulace datového řetězce včetně vložených bitů řízení DSV s následnými výskyty omezení T_{min} jsou popsány dále. V simulaci byly použity tabulky 2 a 3. Simulace používající tabulku 1 pro běžnou modulaci RLL (1-7) byla také simulována pro srovnání.

V simulacích bylo prováděno řízení DSV pomocí vložení 1 bitu řízení DSV na každých 56 datových bitů náhodných dat, která obsahovala 13,107,200 náhodně zvolených bitů a data byla přeměněna na řetězec kódových slov (nebo sled kanálových bitů) pomocí převodního kódu tabulky 2 nebo 3.

Group	Condition A	Condition B	Condition C	Condition D
Control	~85	~85	~85	~85
Mild	~75	~75	~75	~75
Severe	~65	~65	~65	~65

.....

Číselné hodnoty na řádcích 2T až 10T tabulky 6 jsou

číselné hodnoty rozdělení běhů.

Číselné hodnoty rozdělení následných minimálních běhů:

$(Ren_cnt[i] * (i)) / T_size[2T]$ kde $i = 1, 2, 3, 4, \dots, 10$.

Číselné hodnoty na řádcích RMTR(1) až RMTR(9) tabulky 6 jsou číselné hodnoty rozdělení následných minimálních běhů.

Max_RMRT: maximální počet maximálních opakování minimálních běhů DSV: Maxima vypočtených hodnot DSV na kladné a záporné straně pozorovaných v procesu provádění řízení DSV na sledu kanálových bitů.

Výpočet relativní redundance zavedené vložením 1 bitu řízení DSV pro každých 56 datových bitů je založen na skutečnosti, že jeden bit řízení DSV se vyskytuje pro každých 56 datových bitů. Takže relativní redundance se vypočítá následovně:

$$\text{Relativní redundance} = 1 / (1 + 56) = 1,75\%.$$

Výpočet relativní redundance, která je zavedena vložením 2 bitů DSV pro každých 112 kódových bitů je založen na skutečnosti, že pro každých 112 bitů kódových slov se vyskytnou 2 bity DSV. Takže relativní redundance se vypočítá následovně:

$$\text{Relativní redundance} = 2 - (2 + 112) = 1,75\%.$$

Takže je dosaženo stejné redundance v obou případech.

Porovnání PP17

<Tabulka 2 > <Tabulka 3> <Tabulka 1>

	17PP-32	17-PP-52	+2bity - DC (řízení DSV)	Bez - DCC (bez řízení)
Průměr				
Běh	3,3665	3,4048	3,3016	3,2868
Sum	20011947	20011947	20011788	19660782
Total	5944349	5877654	6061150	5981807
2T	0,2256	0,2246	0,2417	0,1419
3T	0,2217	0,2069	0,2234	0,2281
4T	0,1948	0,1935	0,1902	0,1915
5T	0,1499	0,1491	0,1502	0,1511
6T	0,1109	0,1904	0,1135	0,1141
7T	0,0579	0,0814	0,0561	0,0544
8T	0,0392	0,0351	0,0218	0,0188
9T	-----	0,0023	-----	-----
10T	-----	0,0009	-----	-----
RMTR(1)	0,3837	0,3890	0,3628	0,3641
RMTR(2)	0,3107	0,3137	0,2884	0,2883
RMTR(3)	0,1738	0,1906	0,1717	0,1716
RMTR(4)	0,0938	0,0806	0,0909	0,0907
RMTR(5)	0,0299	0,0228	0,0456	0,0452
RMTR(6)	0,0081	0,0033	0,0219	0,0217

RMTR(7)	-----	-----	0,0100	0,0099
RMTR(8)	-----	-----	0,0047	0,0046
RMTR(9)	-----	-----	0,0022	0,0022
Max_RMTR	6	6	18	18

maximální DSV #-36 až 36 #-35 až 40 *-46 až 43 * 1783 až 3433

("#": 56 datových bitů + 1 dc bit, 1,75%)

("*": 112 cbitů + 2 dc bity, 1,75%)

Výše uvedené výsledky potvrzují, že použitím tabulek 2 a 3 se provede systém RLL(1,7) a současně se udržují minimální a maximální běhy a počet následných výskytů minimálního běhu je omezen na 6. Výsledky DSV potvrzují, že řízení DSV může být provedeno v datovém řetězci (tzn. hodnoty maximální DSV se nacházejí v předem stanoveném rozsahu) a v tomto případě je možné, protože účinnost bitů řízení DSV je velká, obdržet nízkofrekvenční komponenty, které jsou uspokojivější, než běžné metody vkládání bitů DSV do řetězce kódových slov (sled kanálových bitů). Výsledky DSV potvrzují, že v případě tabulky 1 je rozdíl mezi kladným a záporným maximem DSV 89 (=46+43), zatímco v případě tabulek 2 a 3 je rozdíl 72 (=36+36), resp. 75 (=35+40), což je v obou případech menší, než hodnota v případě tabulky 1.

Z výše uvedeného popisu je zřejmé, že ve srovnání s obvyklými systémy RLL(1-7), tzn. systémy založenými na tabulce 1, je systém tzv. 17PP používající tabulku 2 nebo 3 schopný omezit počet opakování minimálních běhu nejvíce na 6. Výsledkem je, že se d8 předpokládat zlepšení chybové

charakteristiky při vysoké hustotě.

Navíc, protože účinnost řízení DSV je výborná, má provedení řízení DSV v systému 17PP se stejnou redundancí 1,75% jako v případě obvyklých systému RLL(1-7) za následek menší rozdíl mezi zápornými a kladnými maximálními hodnotami. Protože nízkofrekvenční složky mohou být potlačeny, je výsledkem stabilní chod záznamu/reprodukce dat.

Dále byla provedena také simulace pro zkoumání šíření demodulační chyby, která je způsobena posuvem bitů ve sledu kanálových bitů, které jsou generovány ze stejných náhodných dat jako ve výše popsaném případě. výsledek zkoumání naznačuje, že nejhorší šíření chyb v systému 17PP jsou 3 byty. Avšak tento výsledek potvrzuje, že kmitočet vlastní generace chyb je rozdílná od nuly, což je hodnota indikující nevelké zhoršení ve srovnání s obvyklým systémem RLL(1-7). Byly potvrzeny průměrné četnosti chyb 1,014 bytů pro tabulku 1, 1,167 bytů pro tabulku 2 a 1,174 bytů pro tabulku 3. Měli bychom poznamenat, že výsledné číselné hodnoty pro převodní tabulky podle tohoto vynálezu obsahují bity řízení DSV, ale v případě obvyklého systému RLL(1-7) číselné hodnoty neobsahují bity řízení DSV. To znamená, že se nedá úplně říci, že měření byla provedena za stejných podmínek. Rozdíl v podmínkách měření může ovlivnit číselné hodnoty a proto je nutné vzít v úvahu vliv tohoto rozdílu na výsledky.

<Tabulka 7>

Chybová charakteristika vlivu posuvu bitů

	<Tabulka 2>	<Tabulka 3>	<Tabulka 1>
	17PP-32	17PP-52	+2bit-DC
nejhorší případ (dc bity)	3 byty obsaženy	3 byty obsaženy	2 byty neobsaženy
bytová chyba(0)	0,028	0,096	0,080
bytová chyba(1)	0,777	0,0635	0,826
bytová chyba(2)	0,195	0,268	0,094
bytová chyba(3)	0,000	0,001	-----
Průměr			
četnost chyb bytů	1,167 bytů	1,174 bytů	1,014bytů

Jak bylo popsáno výše, obsahují v tomto provedení převodní tabulky s minimálním během $d=1$, maximálním během $k=7$ a převodním poměrem $m/n = 2/3$ substituční kódy pro omezení počtu následných výskytů minimálních běhů, což má za následek následující:

- (1) Záznam a reprodukce s vysokou hustotou a tolerance na tangenciální zešikmení impulzu jsou vylepšeny.
- (2) Je možné redukovat počet nízkofrekvenčních složek, aby se zvýšila přesnost zpracování vlny, např. AGC a PLL, a tudíž i zlepšila celkové charakteristiky.
- (3) Ve srovnání s obvyklým systémem je možné navrhnout

s krátkou délkou paměti kód abi bitů nebo podobně a tím zmenšit velikost obvodu.

Navíc zbytek dělení dvěma počtu "1" elementu v datovém řetězci s hodnotou 0 nebo 1 bude vždy rovný zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, který je výsledkem přeměny daného datového řetězce, což má za následek následující:

(4) Počet redundandních bitů pro řízení DSV může být redukován.

(5) Řízení DSV může být provedeno s minimálním během $d=1$ a převodními parametry $(m,n) = (2,3)$ na 1,5 bitovém kódovém slovu.

(6) Navíc k nízké relativní redundanci mohou být udrženy minimální a maximální běh d a k .

Navíc obsahují převodní tabulky substituční kódy pro udržení omezení délky běhu, což má za následky následující:

(7) Tabulky jsou kompaktní.

(8) Šíření modulační chyby zapříčiněné posuvem bitů může být vyjádřeno stejně jako v případě obvyklého systému založeného na tabulce 1.

Měli bychom poznamenat, že jako médium pro prezentaci počítačového programu, který se spustí pro provedení výše popsaného zpracování, může být použit CD-ROM a paměti v pevné fázi, komunikační média jako sítě a satelitní spojení navíc k záznamovým médiím, jakým je např. magnetický disk.

Přínos vynálezu

Jak je popsáno výše, podle modulátoru popsaného

v patentovém nároku 1, modulační metoda popsaná v patentovém nároku 23, médium pro prezentaci programu popsané v patentovém nároku 24, demodulátor popsaný v patentovém nároku 25, demodulační metoda popsaná v patentovém nároku 28, médium pro prezentaci programu popsané v bodu 29 je provedeno zpracování na základě převodní tabulky, která uplatňuje převodní pravidlo, podle kterého je zbytek dělení dvěma počtu "1" v datovém řetězci s hodnotou 1 nebo 0 vždy roven zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, který je výsledkem přeměny daného datového řetězce a převodní tabulka obsahuje:

základní kódy pro $d=1$, $k=7$, $m=2$ a $n=3$

první substituční kódy pro omezení počtu následných výskytů minimálního běhu d a

druhé substituční kódy pro udržení omezení délky běhu k

Výsledkem je, že může být prováděno řízení DSV pomocí malého množství redundandních bitů a řetězec kódových slov může být zaznamenán a reprodukován s méně chybami s vyšší lineární hustotou. Navíc může být omezen vzrůst šíření demodulační chyby, která je způsobena bitovým posuvem.

Přehled obrázků na výkresech

Provedení tohoto vynálezu byla popsána s odkazem na následující schémata, kde

Obr. 1 je blokové schéma ukazující typické uspořádání provedení modulátoru podle tohoto vynálezu.

Obr. 2 je schéma popisující zpracování, které je

prováděno jednotkou stanovení/vkládání bitů řízení DSV, která je použita v modulátoru z obr. 1.

Obr. 3 je blokové schéma typického uspořádání modulační jednotky 12, která je použita v modulátoru z obr. 1.

Obr. 4 je schéma, které je příkladem provádění zpracování modulační jednotkou 12 z obr. 3.

Obr. 5 je blokové schéma, které znázorňuje typické uspořádání provedení demodulátoru podle tohoto vynálezu.

Obr. 6 je blokové schéma typického uspořádání demodulační jednotky 111, která je použita v demodulátoru z obr. 5.

Obr. 7 je schéma popisující zpracování, které je prováděno demodulační jednotkou 111 z obr. 6.

Obr. 8 je vývojový diagram, který vysvětluje činnost jednotky odstraňování bitů řízení DSV 112, která je použita v demodulátoru z obr. 5.

Obr. 9 je blokové schéma, které znázorňuje další typické uspořádání provedení modulátoru podle tohoto vynálezu.

Obr. 10 je blokové schéma, které znázorňuje další typické uspořádání provedení demodulátoru podle tohoto vynálezu a

Obr. 11 je schéma, které ukazuje příklad kódu pro záznam se synchronizačními signály a bity řízení DSV, které jsou do nich vloženy.

Seznam hlavních číselných odkazů

<u>11</u>	Jednotka stanovení/vkládání bitu řízení DSV
<u>12</u>	Modulační jednotka

- 100000
- 13 Jednotka kódování NRZI
 - 31 Posuvný registr
 - 32 Jednotka stanovení omezení délky
 - 33 Jednotka detekce kódu omezení výskytu následných
 minimálních běhů
 - 34 Jednotka detekce kódu zabezpečující omezení délku
 běhu
 - 35-1 až 35-4 Převodní jednotky
 - 36 Multiplexer
 - 37 Vyrovňovací paměť (buffer)
 - 111 Demodulační jednotka
 - 112 Jednotka odstraňování bitů řízení DSV
 - 121 Komparátor
 - 123 Jednotka stanovení omezení délky
 - 124 Jednotka detekce kódu omezení výskytu následných
 minimálních běhů
 - 125-1 až 125-4 Inverzní převodní jednotky
 - 126 Multiplexer

1. Modulátor pro převod dat se základní délkou m bitů do kódu s proměnnou délkou $(d, k; m, n; r)$ se základní délkou kódu n bitů, kde d je limit minimálního běhu a k je limit maximálního běhu, zmíněný modulátor se vyznačuje tím, že obsahuje prostředky pro převod vstupních dat na kód podle převodní tabulky, kdy zmíněná tabulka uplatňuje převodní pravidlo, podle kterého bude zbytek dělení dvěma počtu "1" elementu v datovém řetězci s hodnotou 0 nebo 1 vždy roven zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, který je výsledkem přeměny zmíněného datového řetězce a převodní kódy zmíněné převodní tabulky obsahují:

základní kódy pro $d=1$, $k=7$, $m=2$ a $n=3$

první substituční kódy pro omezení počtu následných zmíněných minimálních běhů d a

druhé substituční kódy pro udržování zmíněného limitu délky běhu k .

2. Modulátor podle bodu 1 vyznačující se tím, že počet párů, které obsahují vždy jeden datový řetězec a kódový řetězec, které tvoří zmíněné základní kódy pro omezení délky $i=1$ je menší než 4 ($=2^m=2^2$).

3. Modulátor podle bodu 1 vyznačující se tím, že zmíněné základní kódy zmíněných převodních tabulek mají strukturu proměnné délky.

4. Modulátor podle bodu 1 vyznačující se tím, že základní kódy zmíněných převodních tabulek obsahují kód $"*0*"$, kde

znak * je neurčitý kód, který má hodnotu "0", pokud těsně předcházející nebo těsně následující kódové slovo je "1" a hodnotu "1" pokud zmíněné těsně předcházející nebo těsně následující kódové slovo je "0", což znamená, že zmíněný kód "*0*" je buď "000" nebo "101".

5. Modulátor podle bodu 1 vyznačující se tím, že zmíněné převodní kódy zmíněných převodních tabulek obsahují kódy, které jsou určeny odkazem na těsně následující řetězec kódových slov nebo těsně následující datový řetězec.

6. Modulátor podle bodu 1 vyznačující se tím, že zmíněné převodní kódy zmíněných převodních tabulek obsahují kódy, které jsou určeny odkazem na těsně následující řetězec kódových slov nebo řetězec kódových slov určitého typu.

7. Modulátor podle bodu 5 vyznačující se tím, že zmíněné kódy určené odkazem na těsně následující řetězec kódových slov nebo těsně následující datový řetězec je zmíněný první nebo druhý substituční kód.

8. Modulátor podle bodu 1 vyznačující se tím, že počet párů obsahující datový řetězec a kódový řetězec tvořící zmíněné základní kódy pro omezení délky $i=1$ je roven 4 ($=2^m=2^2$).

9. Modulátor podle bodu 1 vyznačující se tím, že pro omezení délky $i=2$ a větší jsou všechny zmíněné převodní kódy zmíněnými prvním a druhým substitučním kódem.

10. Modulátor podle bodu 1 vyznačující se tím, že zmíněné

11. Modulátor podle bodu 1 vyznačující se tím, že zmíněné převodní kódy zmíněných převodních tabulek obsahují kódy, které jsou určeny odkazem na těsně předcházející řetězec kódových slov.

12. Modulátor podle bodu 1 vyznačující se tím, že zmíněný přístroj obsahuje dále prostředky pro vkládání synchronizačního signálu včetně jedinečného obrazce, který není obsažen ve zmíněných převodních kódech zmíněných převodních tabulek na libovolnou pozici do zmíněného řetězce kódových slov.

13. Modulátor podle bodu 12 vyznačující se tím, že zmíněný jedinečný obrazec je obrazec, který přerušuje zmíněný minimální běh k .

14. Modulátor podle bodu 12 vyznačující se tím, že zmíněný jedinečný obrazec je obrazec, který udržuje zmíněný minimální běh d.

15. Modulátor podle bodu 12 vyznačující se tím, že jedinečný obrazec ve zmíněném synchronizačním signálu obsahuje 1 kódové slovo na jeho začátku, které slouží jako spojovací bit s kódovým slovem, které je výsledkem přeměny až do těsně předcházejících dat, druhý bit pro udržení zmíněného minimálního běhu d a třetí bit.

16. Modulátor podle bodu 12 vyznačující se tím, že zmíněný synchronizační signál má velikost alespoň 12 kódových slov.

17. Modulátor podle bodu 12 vyznačující se tím, že na synchronizační signál s alespoň 21 kódovými slovy obsahuje zmíněný synchronizační signál alespoň 2 obrazce s maximálním během $k=8$.

18. Modulátor podle bodu 12 vyznačující se tím, že zmíněné převodní kódy zmíněné převodní tabulky obsahují zakončovací kódy pro zakončení zmíněného kódu, který je výsledkem přeměny.

19. Modulátor podle bodu 18 vyznačující se tím, že zmíněné zakončovací kódy jsou předepsány pro zmíněné základní kódy s omezením délky i , pro které počet párů, který každý obsahuje datový řetězec a kódový řetězec tvořící zmíněné základní kódy, je menší než 4 ($=2^m=2^2$), a uplatňují převodní pravidlo, podle kterého bude zbytek dělení dvěma počtu "1" elementu v datovém řetězci s hodnotou 0 nebo 1 vždy roven zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, který je výsledkem přeměny zmíněného datového řetězce.

20. Modulátor podle bodu 18 vyznačující se tím, pro identifikaci zmíněného zakončovacího kódu je nastaveno 1 kódové slovo na začátku zmíněného obrazce synchronizačního signálu, který slouží jako spojovací bit, na hodnotu "1" když je zmíněný zakončovací kód použit a na hodnotu "0", když zmíněný zakončovací kód použit není.

21. Modulátor podle bodu 12 vyznačující se tím, že zmíněný jedinečný obrazec je sevřen mezi 3 bity na začátku zmíněného synchronizačního signálu a 3 bity na konci zmíněného synchronizačního signálu a 3 bity na zmíněném začátku a 3 bity na zmíněném konci jsou použity jako spojení obsahující směr dat a spojovacích bitů.

22. Modulátor podle bodu 12 vyznačující se tím, že:
první ze 3 bitů na začátku zmíněného synchronizačního signálu má hodnotu reprezentující datové slovo před přeměnou rozdělené do m-bitových jednotek,
následující druhý ze zmíněných 3 bitů je nastaven na "1" z předpisu pro synchronizační signál,
první ze 3 bitů na konci zmíněného synchronizačního signálu je nastaven na "0" z předpisu pro synchronizační signál a
druhý ze zmíněných 3 bitů na zmíněném konci má hodnotu reprezentující zmíněné datové slovo před přeměnou rozdělené na m-bitové jednotky.

23. Modulátor podle bodu 1 vyznačující se tím, že zmíněný přístroj dále obsahuje prostředky pro řízení DSV pro řízení DSV vstupních dat a vyslání zmíněného DSV na zmíněné převodní prostředky.

24. Modulátor podle bodu 1 vyznačující se tím, že zmíněné převodní prostředky obsahují:
prostředky detekující první kód pro detekci zmíněných substitučních kódů pro omezení počtu následných výskytů zmíněného minimálního běhu d a

I
O
U

prostředky pro detekci druhého kódu pro detekci zmíněných druhých substitučních kódů pro udržení limitu délky běhu.

25. Modulační metoda, kterou používá modulátor pro přeměnu dat se základní délkou dat m bitů na kód s proměnnou délkou $(d,k;m,n;r)$ se základní délkou kódu n bitů, kde d je minimální běh a k je omezení délky běhu, zmíněná modulační metoda se vyznačuje tím, že obsahuje krok přeměny vstupních dat na kód podle převodní tabulky, kdy zmíněná převodní tabulka uplatňuje převodní pravidlo, podle kterého bude zbytek dělení dvěma počtu "1" elementu v datovém řetězci s hodnotou 0 nebo 1 vždy roven zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, který je výsledkem přeměny zmíněného datového řetězce a převodní kódy zmíněné převodní tabulky obsahují:

základní kódy pro $d=1$, $k=7$, $m=2$ a $n=3$

první substituční kódy pro omezení počtu následných zmíněných minimálních běhů d a

druhé substituční kódy pro udržování zmíněného limitu délky běhu k .

26. Médium pro prezentaci programu, který provádí zpracování včetně kroku přeměny vstupních dat na kód podle převodní tabulky dat v modulátoru pro přeměnu dat se základní datovou délkou m bitů do kódu s proměnnou délkou $(d,k;m,n;r)$ se základní délkou kódu n bitů, kde d je minimální běh a k je omezení délky běhu, zmíněné médium pro prezentaci programu se vyznačuje tím, že zmíněná převodní tabulka uplatňuje převodní pravidlo, podle kterého bude zbytek dělení dvěma počtu "1" elementu v datovém řetězci s hodnotou 0 nebo 1

vždy roven zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, který je výsledkem přeměny zmíněného datového řetězce a převodní kódy zmíněné převodní tabulky obsahují:

základní kódy pro $d=1$, $k=7$, $m=2$ a $n=3$

první substituční kódy pro omezení počtu následných zmíněných minimálních běhů d a

druhé substituční kódy pro udržování zmíněného limitu délky běhu k .

27. Demodulátor pro přeměnu kódu s proměnnou délkou $(d,k;m,n;r)$ se základní délkou kódu m bitů na data se základní délkou dat m bitů, kde d je minimální běh a k je omezení délky běhu, zmíněný demodulátor se vyznačuje tím, že obsahuje převodní prostředky pro přeměnu vstupního kódu na data podle převodní tabulky, která uplatňuje převodní pravidlo, podle kterého bude zbytek dělení dvěma počtu "1" elementu v datovém řetězci s hodnotou 0 nebo 1 vždy roven zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, který je výsledkem přeměny zmíněného datového řetězce a převodní kódy zmíněné převodní tabulky obsahují:

základní kódy pro $d=1$, $k=7$, $m=2$ a $n=3$

první substituční kódy pro omezení počtu následných zmíněných minimálních běhů d a

druhé substituční kódy pro udržování zmíněného limitu délky běhu k .

28. Demodulátor podle bodu 27 vyznačující se tím, že zmíněný přístroj dále obsahuje prostředky pro odstraňování redundandních bitů, které byly vloženy do zmíněného kódu v předem určených intervalech.

29. Demodulátor podle bodu 28 vyznačující se tím, že zmíněné redundandní bity jsou bity DSV nebo synchronizační signály.

30. Demodulační metoda, kterou používá demodulátor pro přeměnu kódu s proměnnou délkou $(d,k;m,n;r)$ se základní délkou kódu n bitů na data se základní délkou dat m bitů, kde d je minimální běh a k je omezení délky běhu, zmíněná demodulační metoda se vyznačuje tím, že obsahuje krok přeměny pro přeměnu vstupního kódu na data podle převodní tabulky, která uplatňuje převodní pravidlo, podle kterého bude zbytek dělení dvěma počtu "1" elementu v datovém řetězci s hodnotou 0 nebo 1 vždy roven zbytku dělení dvěma počtu "1" elementu v řetězci kódových slov, který je výsledkem přeměny zmíněného datového řetězce a převodní kódy zmíněné převodní tabulky obsahují:

základní kódy pro $d=1$, $k=7$, $m=2$ a $n=3$

první substituční kódy pro omezení počtu následných zmíněných minimálních běhů d a

druhé substituční kódy pro udržování zmíněného limitu délky běhu k .

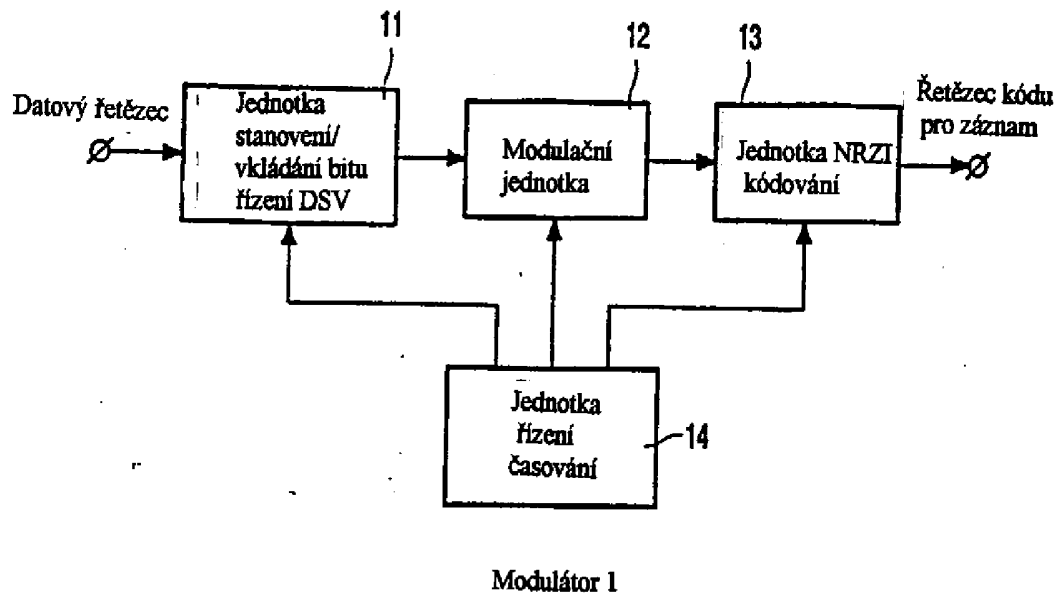
31. Médium pro prezentaci programu, který obsahuje krok přeměny vstupního kódu na data podle převodní tabulky v demodulátoru pro přeměnu kódu s proměnnou délkou $(d,k;m,n;r)$ se základní délkou kódu n bitů na data se základní datovou délkou m bitů, kde d je minimální běh a k je omezení délky běhu, zmíněné médium pro prezentaci programu se vyznačuje tím, že zmíněná převodní tabulka uplatňuje převodní pravidlo, podle kterého bude zbytek

.....
dělení dvěma počtu "1" elementu v datovém řetězci s hodnotou
0 nebo 1 vždy roven zbytku dělení dvěma počtu "1" elementu
v řetězci kódových slov, který je výsledkem přeměny
zminěného datového řetězce a převodní kódy zminěné převodní
tabulky obsahují:

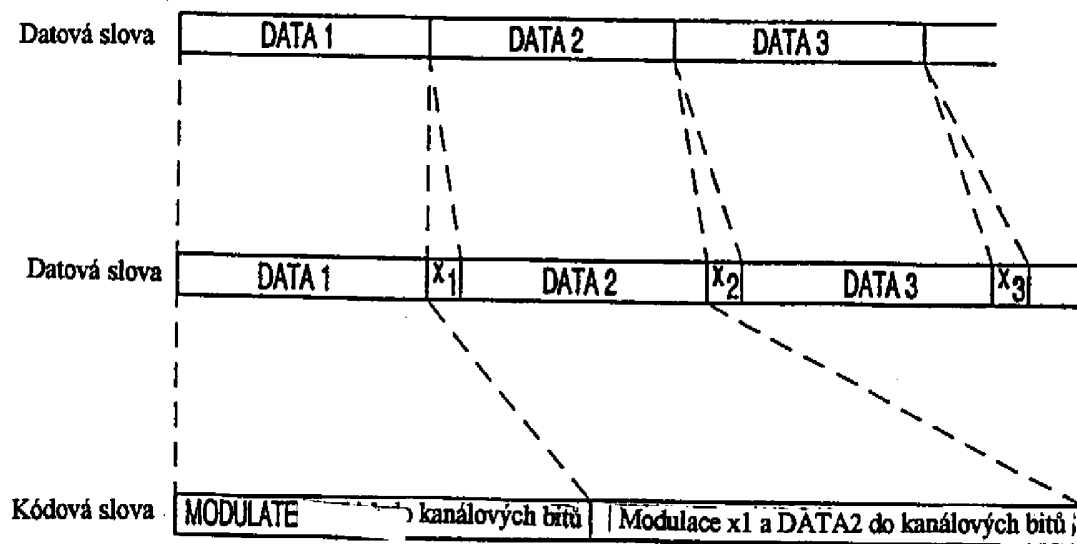
základní kódy pro $d=1$, $k=7$, $m=2$ a $n=3$

první substituční kódy pro omezení počtu následných
zminěných minimálních běhů d a

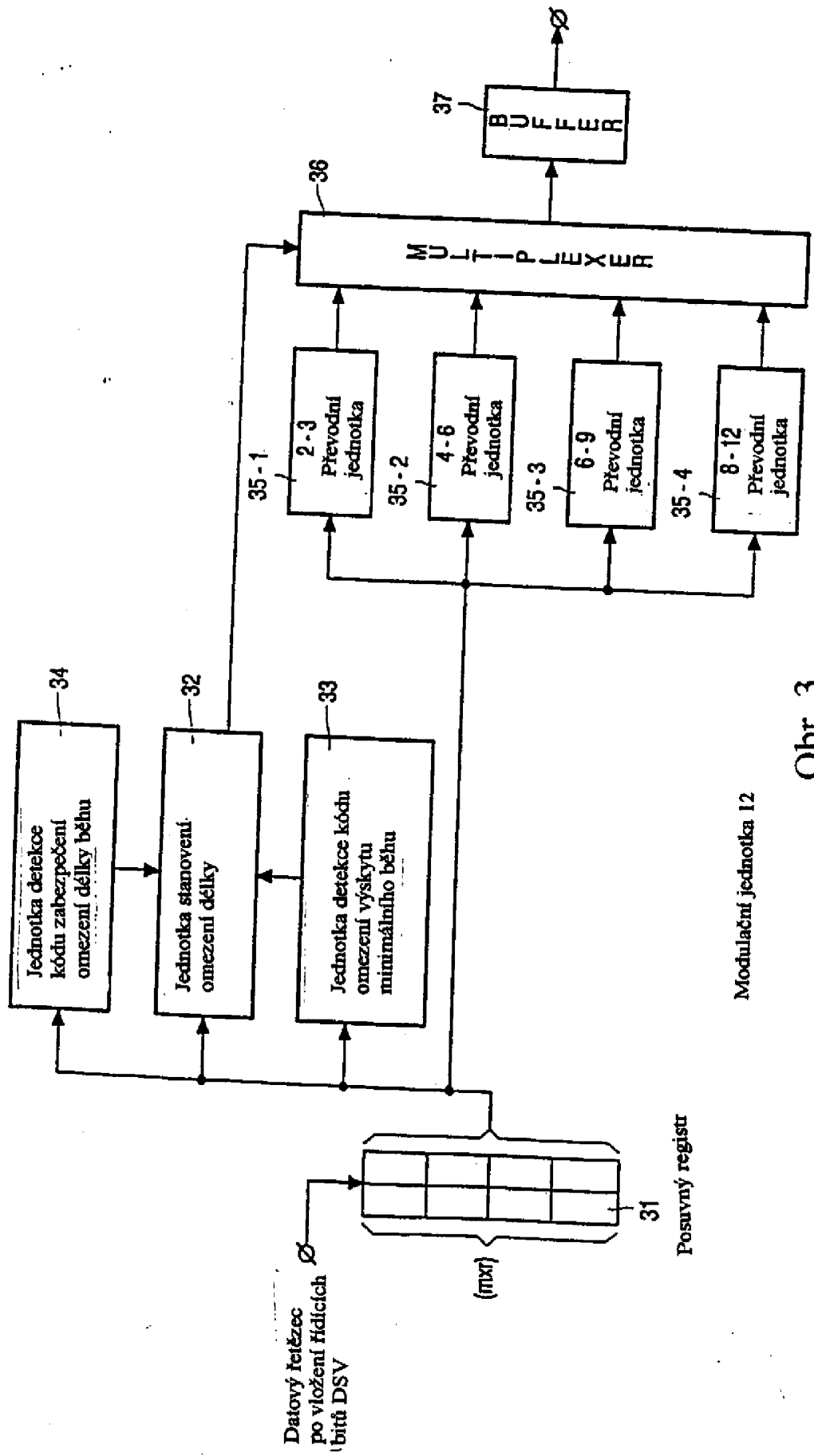
druhé substituční kódy pro udržování zminěného limitu
délky běhu k .



Obr. 1

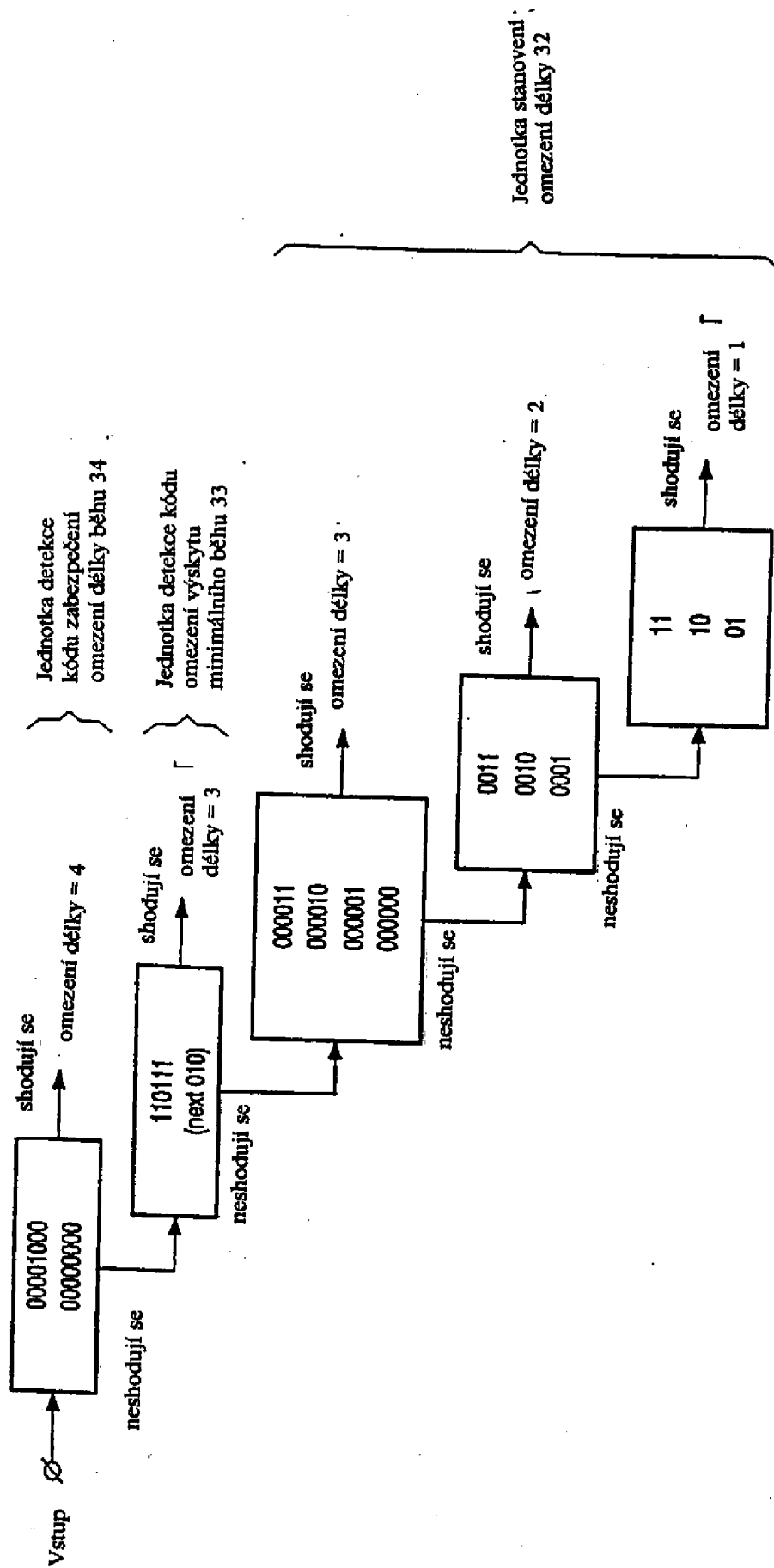


Obr. 2

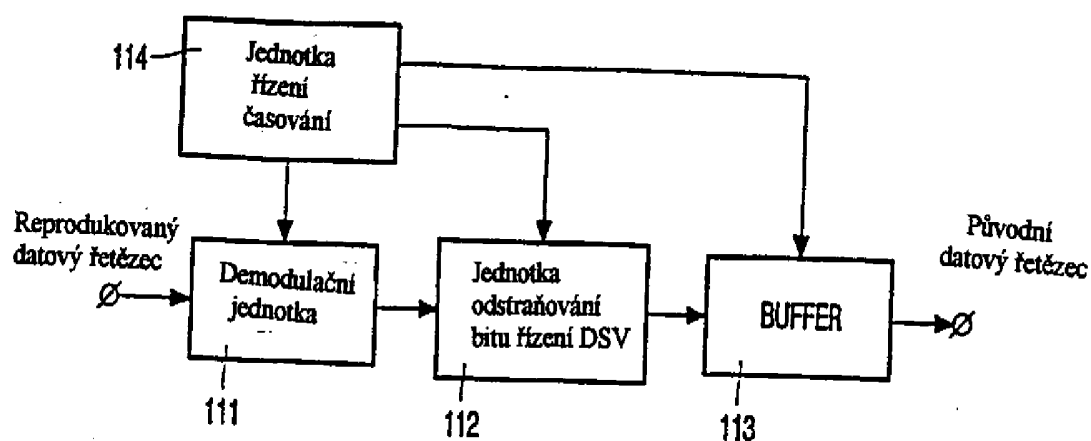


Obr. 3

Modulační jednotka 12

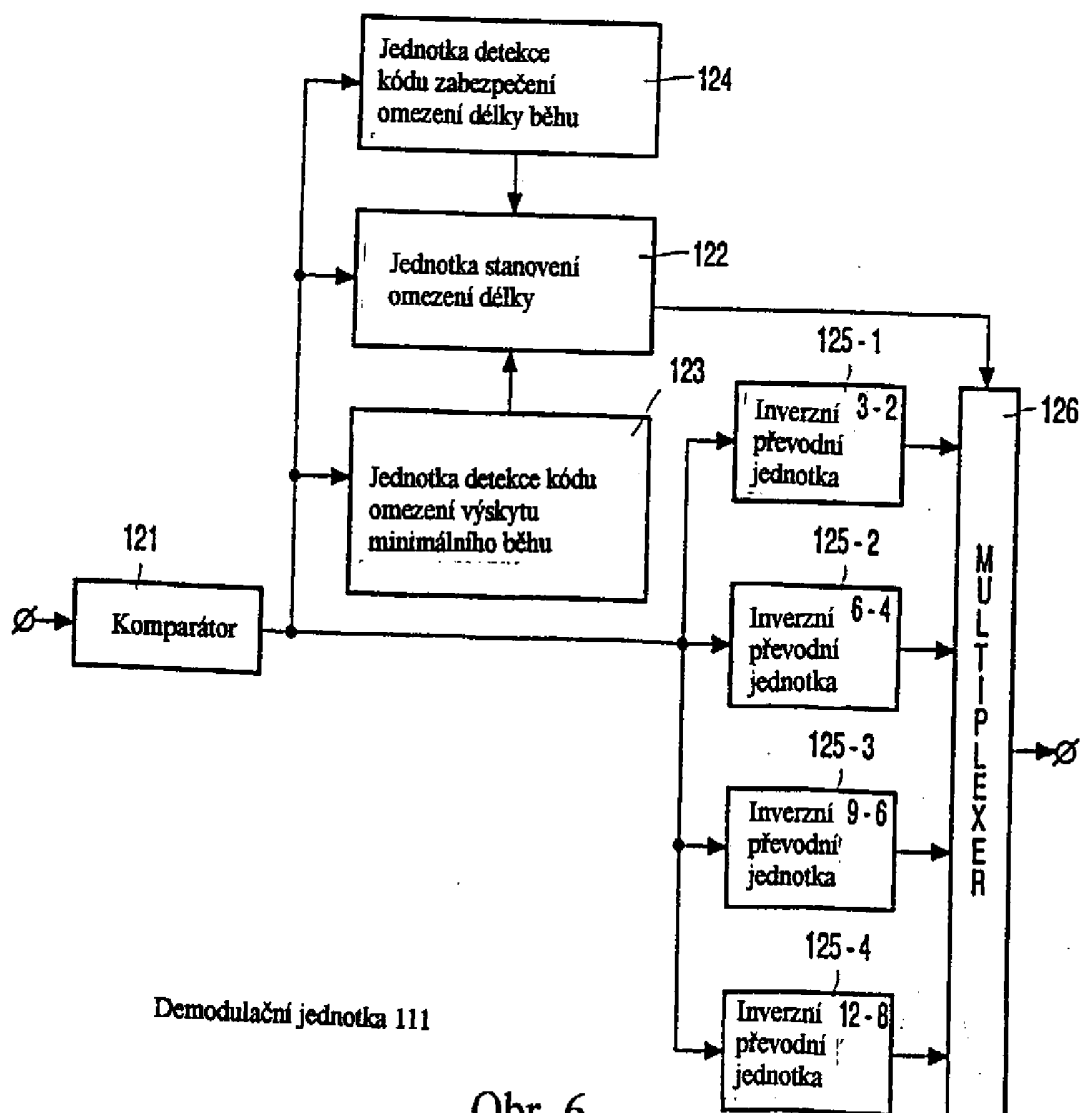


Obr. 4



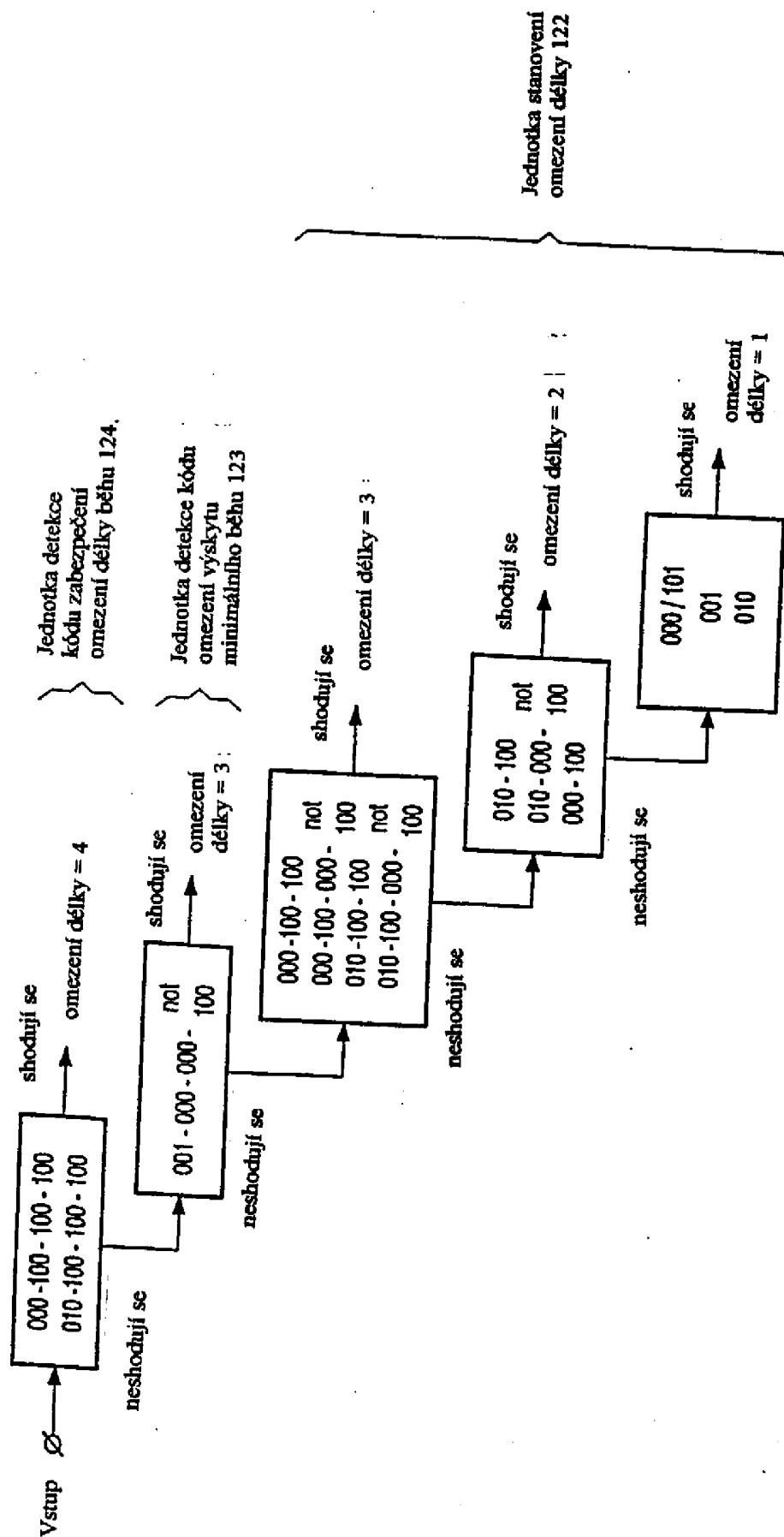
Demodulátor 100

Obr. 5



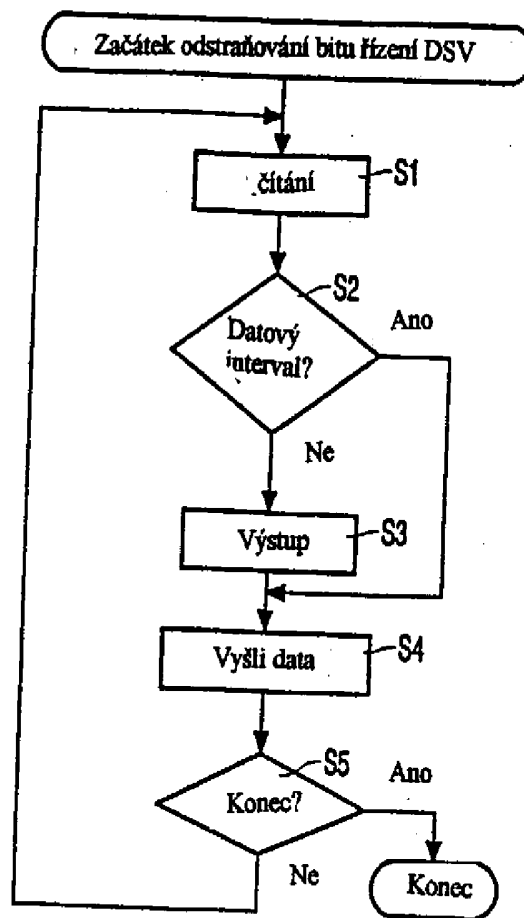
Demodulační jednotka 111

Obr. 6

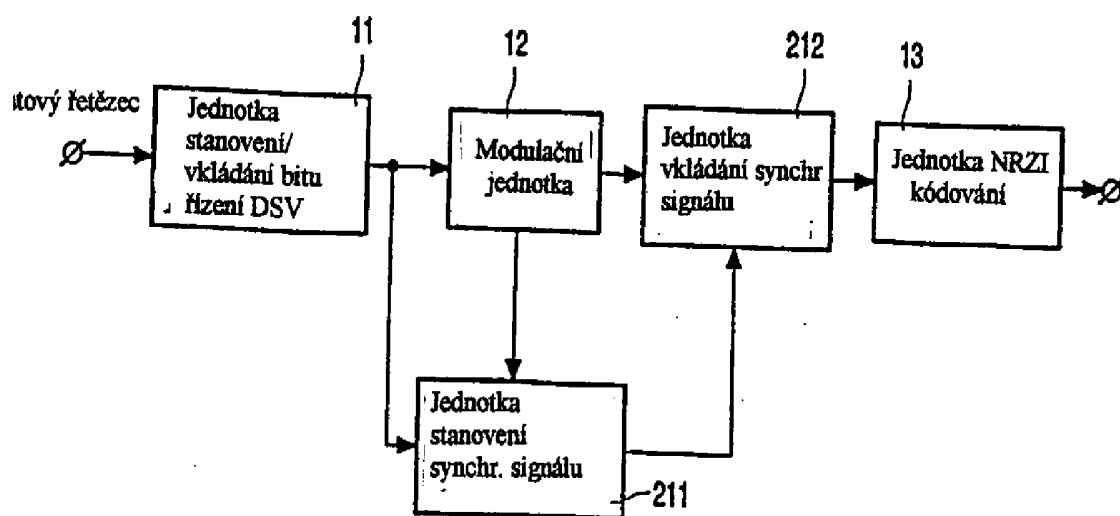


Obr. 7

6/7

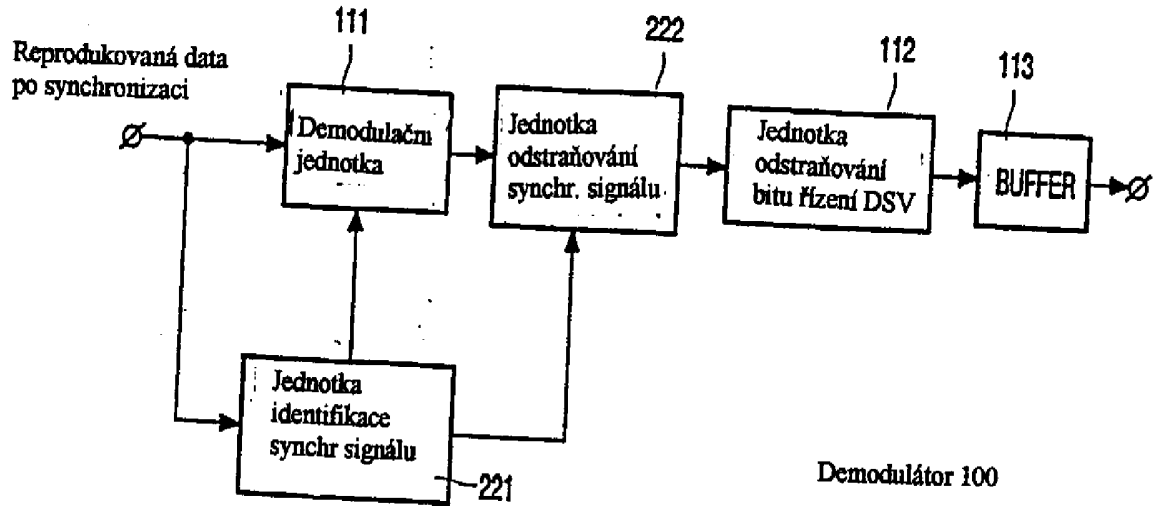


Obr. 8

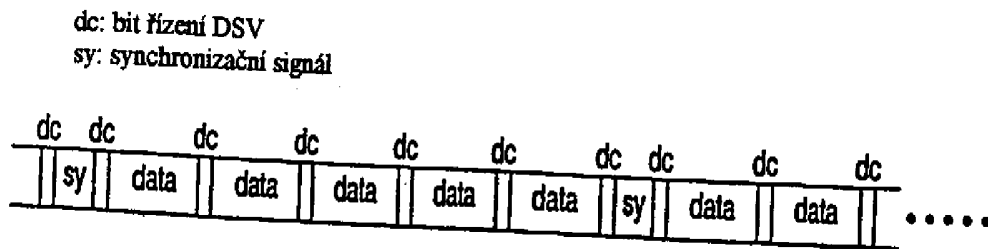


Obr. 9

Modulátor 1



Obr. 10



Obr. 11