

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

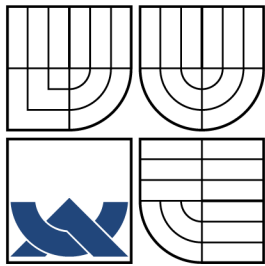
MOŽNOSTI REALIZACE KODEKŮ ZÚŽENÝCH
KONVOLUČNÍCH KÓDŮ

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

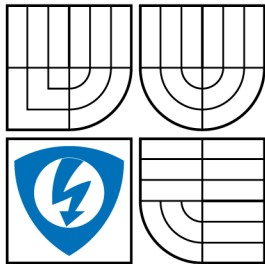
AUTOR PRÁCE
AUTHOR

JAKUB ŠEDÝ

BRNO 2009



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND
COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

MOŽNOSTI REALIZACE KODEKŮ ZÚŽENÝCH
KONVOLUČNÍCH KÓDŮ
REALISATION FEASIBILITIES OF PUNCTURED CONVOLUTIONAL ENCODERS

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

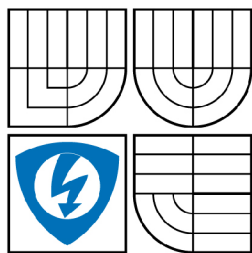
AUTOR PRÁCE
AUTHOR

JAKUB ŠEDÝ

VEDOUCÍ PRÁCE
SUPERVISOR

DOC. ING. KAREL NĚMEC, CSC.

BRNO 2009



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav telekomunikací

Bakalářská práce

bakalářský studijní obor
Teleinformatika

Student: Jakub Šedý

ID: 98460

Ročník: 3

Akademický rok: 2008/2009

NÁZEV TÉMATU:

Možnosti realizace kodeků zúžených konvolučních kódů

POKYNY PRO VYPRACOVÁNÍ:

Navrhněte zúžený konvoluční kód, který zabezpečí digitální přenos dat proti třem nezávislým chybám, tj. $t = 3$, v kódovém ohraničení nA . Pro tento kód vypracujte podrobný návrh realizace kodeku v podobě samostatného elektronického systému. V návrhu realizace využijte skutečnost, že protichybový kodek bude součástí protichybového kódového systému. Ověřte funkční schopnost tohoto kodeku metodou, kterou považujete pro tento návrh za nejvhodnější.

DOPORUČENÁ LITERATURA:

[1] LEE, L.H.CH. Convolutional Coding - Fundamentals and Applications. Artech House, Boston, London, ISBN 0-89006-914-X.

[2] LEE, L.H.CH. New Rate-Compatible Punctured Convolutional Codes for Viterbi Decoding. IEEE Trans. Commun., vol. COM-42, pp. 3073-3079, Dec. 1994.

[3] BEGIN, G.- HACCOUN, D.-CHANTAL, P. Further Results on High Rate Punctured Convolutional Codes for Viterbi and Sequential Decoding. IEEE Trans. Commun., vol. COM-38, No. 11, November 1990, pp. 1922-1928.

Termín zadání: 9.2.2009

Termín odevzdání: 2.6.2009

Vedoucí práce: doc. Ing. Karel Němec, CSc.

prof. Ing. Kamil Vrba, CSc.

Předseda oborové rady

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

ABSTRAKT

Cílem této práce je přiblížit čtenáři problematiku zúžených konvolučních kódů a to v oblasti návrhu a ověření funkčnosti kodeku jako samostatného elektronického systému. Práce je rozdělena na čtyři základní kapitoly. První tři kapitoly se zabývají návrhem a realizací jednotlivých bloků kodeku a následnému ověření těchto bloků. Poslední, čtvrtá kapitola, se zabývá návrhem a realizací kodeku jako samostatného elektronického systému.

KLÍČOVÁ SLOVA

Kodek, kodér, dekodér, zúžovací zařízení, zpětně rozšiřující zařízení, zúžený konvoluční kód.

ABSTRACT

The aim of this thesis is to bring readers the issue punctured convolutional codes and in the design and verification of the functionality of the codec as a separate electronic system. The thesis is divided into four main chapters. The first three chapters deal with design and implementation of individual blocks of the codec and the subsequent verification of these blocks. The fourth and last chapter deals with the design and implementation of the codec as a separate electronic system.

KEYWORDS

Codec, coder, decoder, puncturing device, depuncturing device, punctured convolutional code.

ŠEDÝ, J. *Možnosti realizace kodeků zúžených konvolučních kódů*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2009. 46 s., Vedoucí bakalářské práce doc. Ing. Karel Němec, CSc.

PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Možnosti realizace kodeků zúžených konvolučních kódů“ jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne

.....

(podpis autora)

OBSAH

1	Kódování zúžených konvolučních kódů	10
1.1	Úvod	10
1.2	Zúžené konvoluční kódy	11
1.3	Poměr - přizpůsobení (rate - compatible) zúžené konvoluční kódy . . .	14
1.4	Zvolení kódu, návrh a realizace kodéru	16
2	Zúžení a zpětné rozšíření konvolučních kódů	22
2.1	Zúžení a zpětné rozšíření	22
2.2	Návrh a realizace zúžovacího zařízení	25
2.3	Návrh a realizace zpětně rozšiřujícího zařízení	26
3	Dekódování zúžených konvolučních kódů	29
3.1	Největší pravděpodobnost dekodování zúžených konvolučních kódů . . .	29
3.2	Největší pravděpodobnost dekodování poměr - přizpůsobení (rate - compatible) zúžených konvolučních kódů	31
3.3	Návrh a realizace dekodéru	32
4	Kodek	41
4.1	Návrh a realizace kodeku	41
5	Závěr	44
	Literatura	45

SEZNAM OBRÁZKŮ

1.1	Kódovací obvod pro (2, 1, 2) kód.	11
1.2	Mřížový diagram pro (2, 1, 2) kód.	12
1.3	Kódovací obvod pro (3, 2, 2) kód.	12
1.4	Mřížový diagram pro (3, 2, 2) kód.	13
1.5	Stanovení RCPC kódů pro typický blok zprávy se seskupenými bity podle jejich chybové citlivosti.	15
1.6	Kódovací obvod pro kód (14, 7, 6).	18
1.7	Mřížový diagram pro kód (14, 7, 6).	19
1.8	Návrh a realizace kodéru v programu Matlab Simulink.	20
1.9	Vstupní a výstupní parametry kodéru.	21
2.1	Zúžení.	23
2.2	Zpětné rozšíření.	23
2.3	Zúžení a zpětné rozšíření pro konkrétní příklad.	24
2.4	Návrh a realizace zúžovacího zařízení v programu Matlab Simulink.	25
2.5	Vstupní a výstupní parametry zúžovacího zařízení.	26
2.6	Návrh a realizace zpětně rozšiřujícího zařízení v programu Matlab Simulink.	26
2.7	Vstupní a výstupní parametry zpětně rozšiřujícího zařízení.	27
3.1	Mřížový diagram pro Viterbiho dekodér.	33
3.2	1. stav mřížového diagramu Viterbiho dekodéru.	34
3.3	2. stav mřížového diagramu Viterbiho dekodéru.	35
3.4	3. stav mřížového diagramu Viterbiho dekodéru.	35
3.5	4. stav mřížového diagramu Viterbiho dekodéru.	36
3.6	5. stav mřížového diagramu Viterbiho dekodéru.	37
3.7	Nejpravděpodobnější cesta dekodovacího procesu pro danou zprávu.	38
3.8	Návrh a realizace dekodéru v programu Matlab Simulink.	39
3.9	Vstupní a výstupní parametry dekodéru.	40
4.1	Návrh a realizace kodeku v programu Matlab Simulink.	42
4.2	Vstupní a výstupní parametry kodeku.	43

SEZNAM TABULEK

1.1	Váhová spektra RCPC kódů generovaných z poměru - 1/2 základního kódu.	17
-----	--	----

ÚVOD

Při přenosu dat se díky zvyšování přenosové rychlosti stále častěji setkáváme s chybami. K dispozici jsou konvoluční kódy, které jsou schopny tento přenos dostatečně zabezpečit. V některých případech nejsou konvoluční kódy vhodné a to tehdy, kdy je vyžadována efektivnější redukce dat. Z tohoto důvodu byly vyvinuty zúžené konvoluční kódy.

Tato práce se zabývá problematikou zúžených konvolučních kódů. Cílem práce, jak je uvedeno v zadání, je navrhnout zúžený konvoluční kód, který bude schopen zabezpečit digitální přenos dat proti třem nezávislým chybám a následně pro tento kód navrhnout a realizovat kodek. Práce je rozdělena do čtyř základních částí. V první části je ukázán způsob jak zvolit nejvhodnější kód a ukázka výpočtů parametrů kódu potřebných k realizaci a návrhu kodeku. Následující tři části mají čtenáře seznámit s realizací a návrhem kodeku jako samostatného systému, který je součástí protichybového kódového systému. Jsou zde popsány funkce jednotlivých bloků kodeku a jejich následné ověření funkčnosti.

1 KÓDOVÁNÍ ZÚŽENÝCH KONVOLUČNÍCH KÓDŮ

1.1 Úvod

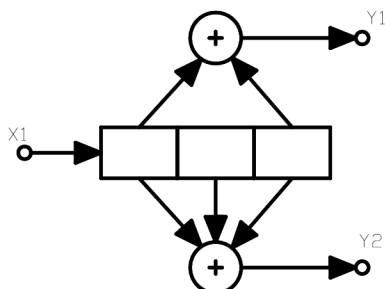
Konvolučním kódování s Viterbiho dekódováním je dobrým prostředkem k dosažení velkých kódovacích zisků na diskretních rušených kanálech. V aplikacích, kde je vysoký přenos dat a dobrá korekční schopnost, jsou požadovány pro kmitočtově omezené kanály a je potřebný efektivní rychlý přenos konvolučních kódů $R_c = k/n$. Pro (n, k, m) konvolučních kódů, kde existuje 2^k cest pro opuštění nebo spojení se z dalším uzlem v mřížovém diagramu kodéru. Ve Viterbiho dekodéru musí být provedeno 2^k metrických výpočtů pro každý uzel v mříži a s největší pravděpodobností bude vybrána cesta pro každý uzel v mřížovém diagramu. To představuje problém pro praktickou realizaci rychlého přenosu konvolučních kódů. Naštěstí je možné najít některé kódy, které snižují počet metrických výpočtů pro každý uzel v mřížovém diagramu. Tato třída kódů se nazývá zúžené konvoluční kódy. Zúžené konvoluční kódy byly představeny Clarkem a Kainem v roce 1979. Více nových zúžených konvolučních kódů bylo nalezeno v posledních letech. V některých oblastech použití, jsou různé úrovně ochrany proti chybám žádoucí v rámci řazení informací nebo jednotek. Je požadováno kanálové kódovací schéma s nerovnoměrnou ochranou proti chybám (UEP unequal error protection). Jediný způsob k vytvoření schématu je seskupení vstupních signálů podle jejich chybové citlivosti. Počet použitých paralelních kodérů a dekodérů kanálu k poskytnutí nutných úrovní ochrany proti chybám. Složitost tohoto schématu je vysoká při velkém počtu úrovní ochrany proti chybám. Jinak struktura jednobanového kodéru a dekodéru užívá zúžených konvolučních kódů. Hagenauer objevil poměr - přizpůsobení (rate - compatibility) zúžené konvoluční kódy, které jsou velmi vhodné pro toto použití [7].

1.2 Zúžené konvoluční kódy

Zúžené konvoluční kódy jsou podtřídou (n, k, m) konvolučních kódů. Jsou to v čase proměnné (time-varying) konvoluční kódy. Uvažme $(2, 1, 2)$ konvoluční kód s generátorem polynomiální matice $\mathbf{G}(D) = (1 + D^2 \ 1 + D + D^2)$ na obr. 1.1 a mřížový diagram kodéru na obr. 1.2. Pokud by každý třetí výstupní kódovací bit byl smazán, kódér vytvoří tři kódovací bity pro každé dva vstupní informační bity. Ve skutečnosti každý následující symbol poměru $1/2$ konvolučních kódů je odstraněn z každé větve a nový kód je generován. Tento nový kód má poměr $R_c = 2/3$ a výsledný kód je identický s poměrem $2/3$ konvolučního kódu s generátorem polynomiální matice [7]

$$\mathbf{G}(D) = \begin{bmatrix} 1 + D & 1 + D & 1 \\ 0 & D & 1 + D \end{bmatrix}. \quad (1.1)$$

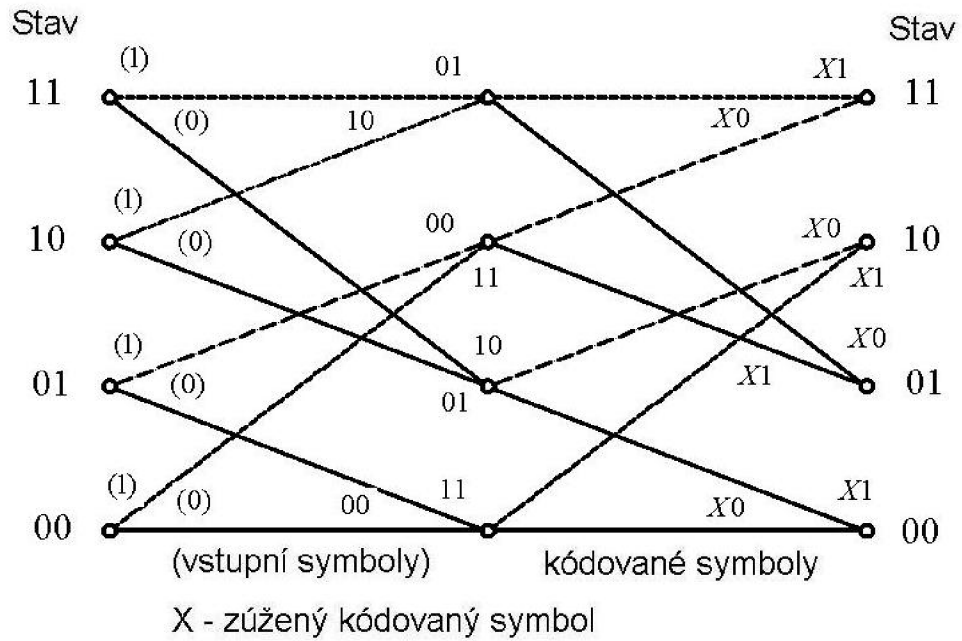
Obvod kodéru tohoto kódu je znázorněn na obr. 1.3, mřížový diagram kodéru na obr. 1.4.



Obr. 1.1: Kódovací obvod pro $(2, 1, 2)$ kód.

Pro dva stejné vstupní a tři výstupní symboly mezi dvěma stavy obr. 1.4, je vidět, že přechod na větví v mříži, znázorněné na obr. 1.2, se provádí prostřednictvím přechodných stavů (dvě větve v příkladu). Je zřejmé, že stejný kód se generuje odlišným způsobem.

Obecně lze říci, že poměr P/Q' zúžených konvolučních kódů lze získat z $(n, 1, m)$ konvolučního kódu vypuštěním $n \cdot P - Q'$ kódovacích symbolů z každého $n \cdot P$ kódovacího symbolu odpovídajícího kódování P informačních symbolů, podle $(n, 1, m)$

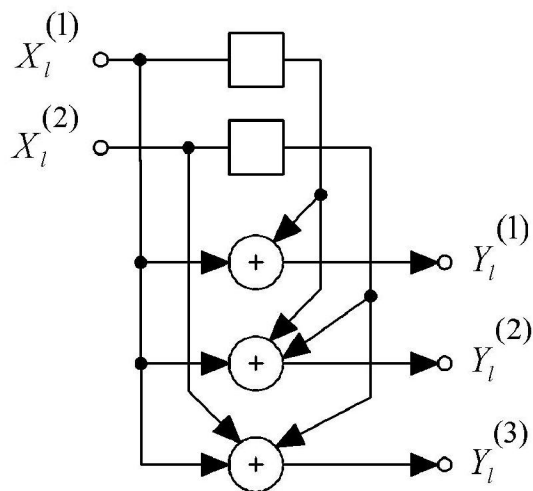


Obr. 1.2: Mřížový diagram pro (2, 1, 2) kód.

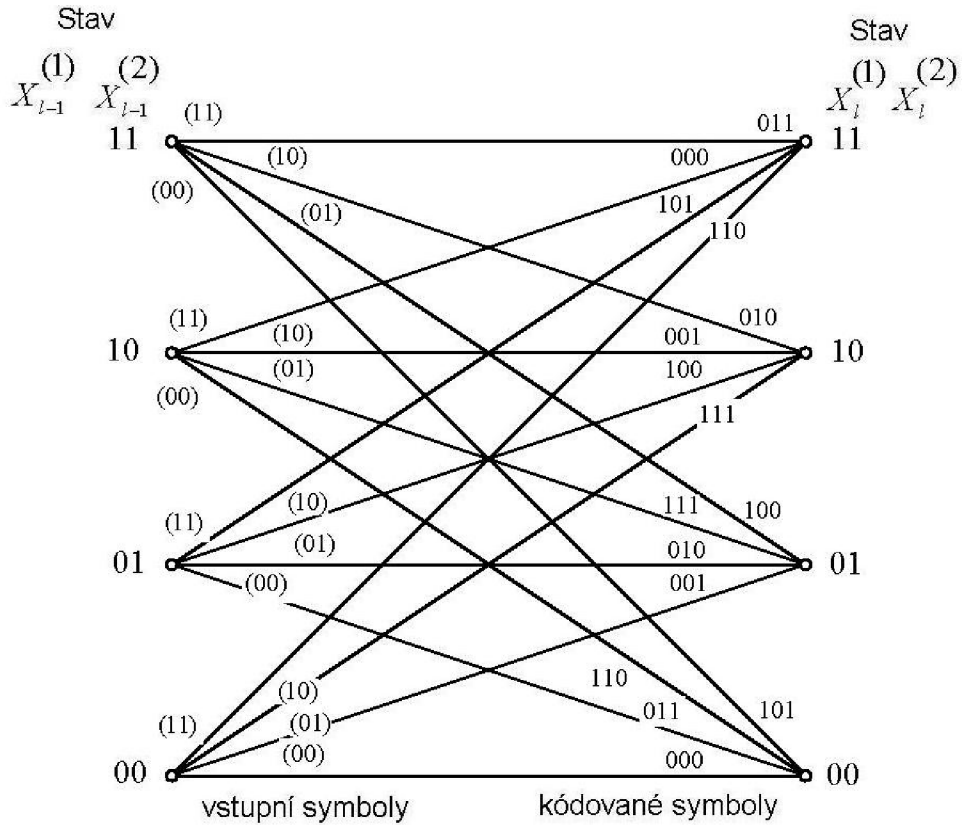
konvolučního kódu. Tento nízký přenos $R_c = 1/n$ kódování je nazýván základním kódem (mother code), s generátorem polynomiální matice [7]

$$\mathbf{G}(D) = [G_1^{(1)}(D) G_1^{(2)}(D) \cdots G_1^{(n)}(D)], \quad (1.2)$$

kde



Obr. 1.3: Kódovací obvod pro (3, 2, 2) kód.



Obr. 1.4: Mřížový diagram pro (3, 2, 2) kód.

$$\mathbf{G}_1^{(j)}(D) = [g_{1,0}^{(j)} + g_{1,1}^{(j)}D + \cdots + g_{1,m}^{(j)}D^m], \quad (1.3)$$

s prvky 0 a 1. Lze vidět, že poměr ze zúženého konvolučního kódu je $R_c = P/(n \cdot P - \delta)$, kde $\delta = n \cdot P - Q'$. Vypuštění kódových symbolů je reprezentováno n -do- P zúženého pole \mathbf{P}_δ . Každý sloupec z \mathbf{P}_δ je spojen s jedním z P n -prvkem vektorů kódu odpovídající kódování P informačních symbolů, podle poměru $1/n$ základního (mother) kódu, a každý řádek \mathbf{P}_δ odpovídá P symbolům kódu na výstupu ze základního (mother) kódu. Prvkem zúženého pole jsou 0 a 1, které odpovídají odstraňování nebo zachování odpovídajících symbolů kódu na výstup ze základního (mother) kódu. Zúžené pole je [7]

$$\mathbf{P}_\delta = \begin{bmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,P} \\ p_{2,1} & p_{2,2} & \cdots & p_{2,P} \\ \vdots & \vdots & & \vdots \\ p_{n,1} & p_{n,2} & \cdots & p_{n,P} \end{bmatrix}, \quad (1.4)$$

s $p_{j,l'} = 0$ or 1 , $1 \leq j \leq n$ a $1 \leq l' \leq P$. δ udává počet nul v poli a δ může nabýt hodnoty mezi 1 a $(n-1)P-1$, včetně. Vzhledem k tomu, že zúžené pole se používá pravidelně pro každé $n \cdot P$ symbolů kódu v kódovacím pořadí, se P nazývá *zúžená perioda*.

1.3 Poměr - přizpůsobení (rate - compatible) zúžené konvoluční kódy

Poměr - přizpůsobení (rate - compatible) zúžené konvoluční kódy (RCPC) jsou podtřídu zúžených konvolučních kódů. Vezměme si příklad, kde poměr $4/7$ z kódu se zúženou periodou $P = 4$ a $\delta = 1$, který je generován z poměru $1/2$ základního (mother) kódu uvedeném v odstavci 2.1. Zúžené pole pro poměr $4/7$ zúženého kódu je [7] [8]

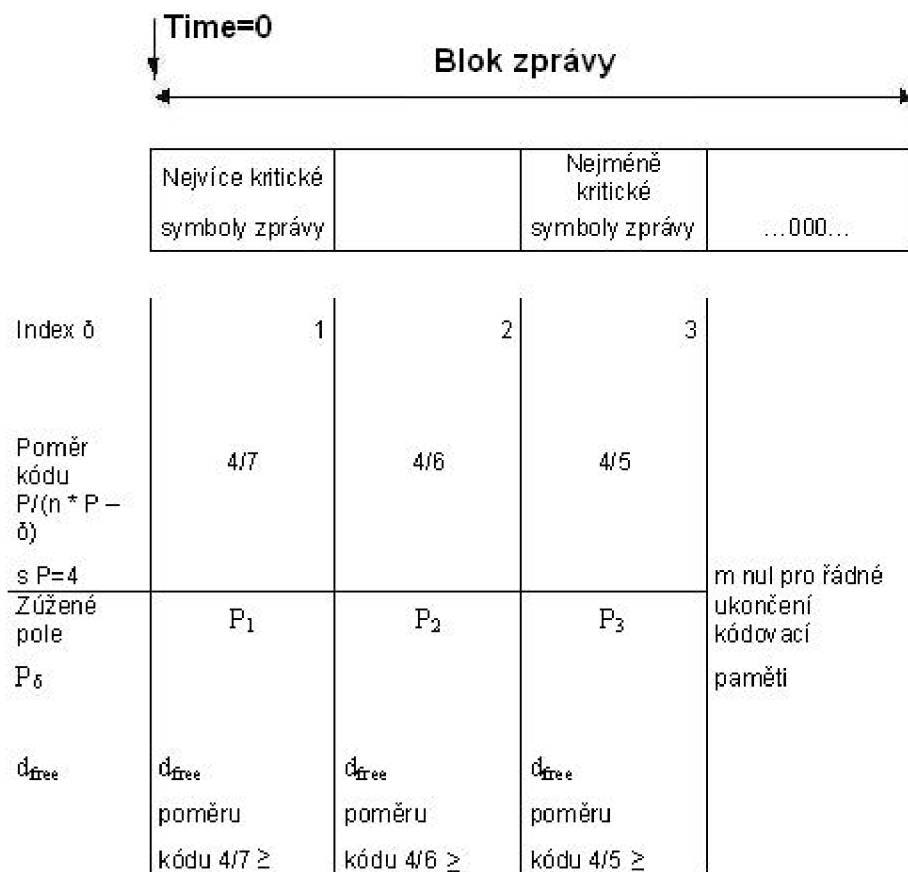
$$\mathbf{P}_1 = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}, \quad (1.5)$$

kde je jeden prvek pole \mathbf{P}_1 nula. Pro vytvoření poměru $4/6$ zúženého kódu se stejnou zúženou periodou P z poměru $1/2$ základního (mother) kódu, máme dva nulové prvky v poli \mathbf{P}_2 . Pro poměr přizpůsobení (rate-compatible), nulový prvek z \mathbf{P}_1 je ponechán v \mathbf{P}_2 . Obecně lze říci, že všechny nulové prvky v \mathbf{P}_δ jsou ponechány v následujícím poli $\mathbf{P}_{\delta+1}$, $\mathbf{P}_{\delta+2}$. Toto omezení naznačuje, že všechny symboly kódu zúžených kódů s vyšším poměrem jsou požadovány kódy s nižším poměrem. V tomto příkladu zúžená pole pro poměr $4/6$ a $4/5$ kódu jsou [7] [8]

$$\mathbf{P}_2 = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}, \quad (1.6)$$

$$\mathbf{P}_3 = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}. \quad (1.7)$$

To znamená že rodina RCPC kódu je generována touto cestou. RCPC kódy jsou vhodné pro použití tam, kde jsou různé stupně chybové ochrany potřebné během posloupnosti zpráv nebo jednotek. Typickou posloupností zpráv nebo jednotek je seskupení podle požadavku na ochranu proti chybám a RCPC kódy jsou přiřazeny do skupiny na obr. 1.5. Lze vidět, že nejvíce kritické symboly zpráv v první skupině, jsou zabezpečeny pomocí RCPC kódů jehož volná vzdálenost je větší nebo se rovná volné vzdálenosti kódů pro následující skupiny. Poměr - přizpůsobení (rate - compatible) kódy zaručují hladký průběh přechodů mezi poměry [7] [8].



Obr. 1.5: Stanovení RCPC kódů pro typický blok zprávy se seskupěnými bity podle jejich chybové citlivosti.

1.4 Zvolení kódu, návrh a realizace kodéru

V této kapitole bude zvolen kód a následně realizován kodér. Jak už bylo zmíněno v zadání, zvolený zúžený konvoluční kód má zabezpečit přenos dat proti třem nezávislým chybám $t = 3$ v kódovém ohraničení n_A . Je nutné vycházet z počtu zadaných chyb $t = 3$. Tuto hodnotu dosadíme do vzorce pro výpočet minimální Hammingovy vzdálenosti d_{free} [9]

$$d_{free} = (2t) + 1. \quad (1.8)$$

Dále je třeba zvolit z tab. 1.4 Váhová spektra RCPC kódů generovaných z poměru $1/2$ základního kódu [7] patřičný kód. Z tabulky je zřejmé, že kód má informační rychlost $R_c = k_0/n_0 = 1/2$, proto $k_0 = 1$, $n_0 = 2$, $K = 6$. Nyní je třeba si vyjádřit kódové ohraničení n_A podle vztahu [9]

$$n_A = n_0 \cdot (K + 1), \quad (1.9)$$

$$k = (K + 1) \cdot k_0, \quad (1.10)$$

vytvářející mnohočleny:

$$\begin{aligned} \mathbf{G}_{(1)}^{(1)} &= (133)_8 = 1011011 \Rightarrow \mathbf{G}_{(1)}^{(1)} = 1 + D^2 + D^3 + D^5 + D^6, \\ \mathbf{G}_{(2)}^{(1)} &= (171)_8 = 1111001 \Rightarrow \mathbf{G}_{(2)}^{(1)} = 1 + D + D^2 + D^3 + D^6, \end{aligned}$$

poměr RCPC kódu P/Q' kde [7]

$$Q' = (n_0 \cdot P - \delta), \quad (1.11)$$

Základní kód		RCPC kód			
K	$\mathbf{G}_{(1)}^{(1)}, \mathbf{G}_{(2)}^{(1)}$	\mathbf{P}_δ	P/Q'	d_{free}	$(a_d, d = d_{free}, d_{free} + 1 \dots)$ $(c_d, d = d_{free}, d_{free} + 1 \dots)$
2	5, 7	1101	4/7	4	(3, 9, 21, 48, 111)
		1111			(4, 19, 72, 224, 651)
		1001	4/6	4	(23, 0, 182, 0, 1530)
		1111			(66, 0, 1108, 0, 13836)
		1001	4/5	2	(1, 15, 63, 262, 1136)
		1110			(3, 54, 387, 2299, 12679)
3	15, 17	1111	4/7	4	(1, 0, 27, 0, 170)
		1101			(2, 0, 80, 0, 903)
		1110	4/6	4	(6, 21, 66, 213, 716)
		1101			(16, 78, 368, 1495, 6064)
		1110	4/5	3	(8, 43, 224, 1212, 6521)
		1001			(31, 264, 1947, 13535, 89172)
4	23, 35	1111	4/7	5	(1, 9, 16, 37, 154)
		1101			(1, 31, 72, 175, 1003)
		1111	4/6	4	(2, 0, 54, 0, 691)
		0101			(2, 0, 248, 0, 5444)
		1111	4/5	3	(4, 20, 137, 844, 4946)
		0100			(21, 137, 1344, 10854, 77549)
5	53, 75	1011	4/7	6	(1, 15, 32, 60, 202)
		1111			(5, 59, 158, 360, 1524)
		1010	4/6	6	(38, 0, 440, 0, 6178)
		1111			(192, 0, 3808, 0, 71872)
		1000	4/5	4	(7, 54, 307, 2005, 12970)
		1111			(40, 381, 3251, 27123, 213451)
6	133, 171	1111	4/7	7	(2, 12, 43, 98, 224)
		1110			(6, 37, 193, 581, 1771)
		1111	4/6	6	(2, 32, 96, 316, 1284)
		1010			(6, 140, 570, 2552, 12320)
		1111	4/5	4	(3, 24, 172, 1158, 7409)
		1000			(12, 188, 1732, 15256, 121372)

Tab. 1.1: Váhová spektra RCPC kódů generovaných z poměru - 1/2 základního kódu.

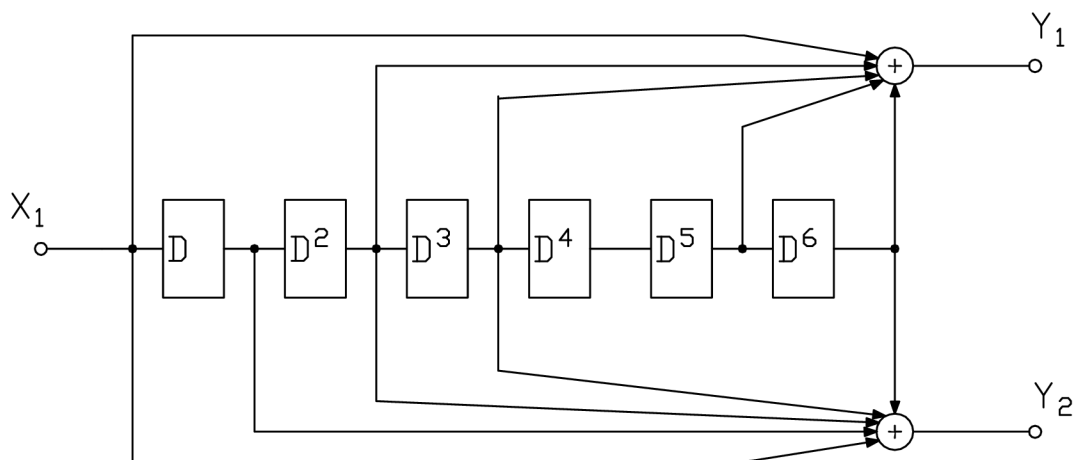
a zužovací pole \mathbf{P}_δ pro poměr RCPC kódu 4/7

$$\mathbf{P}_1 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}.$$

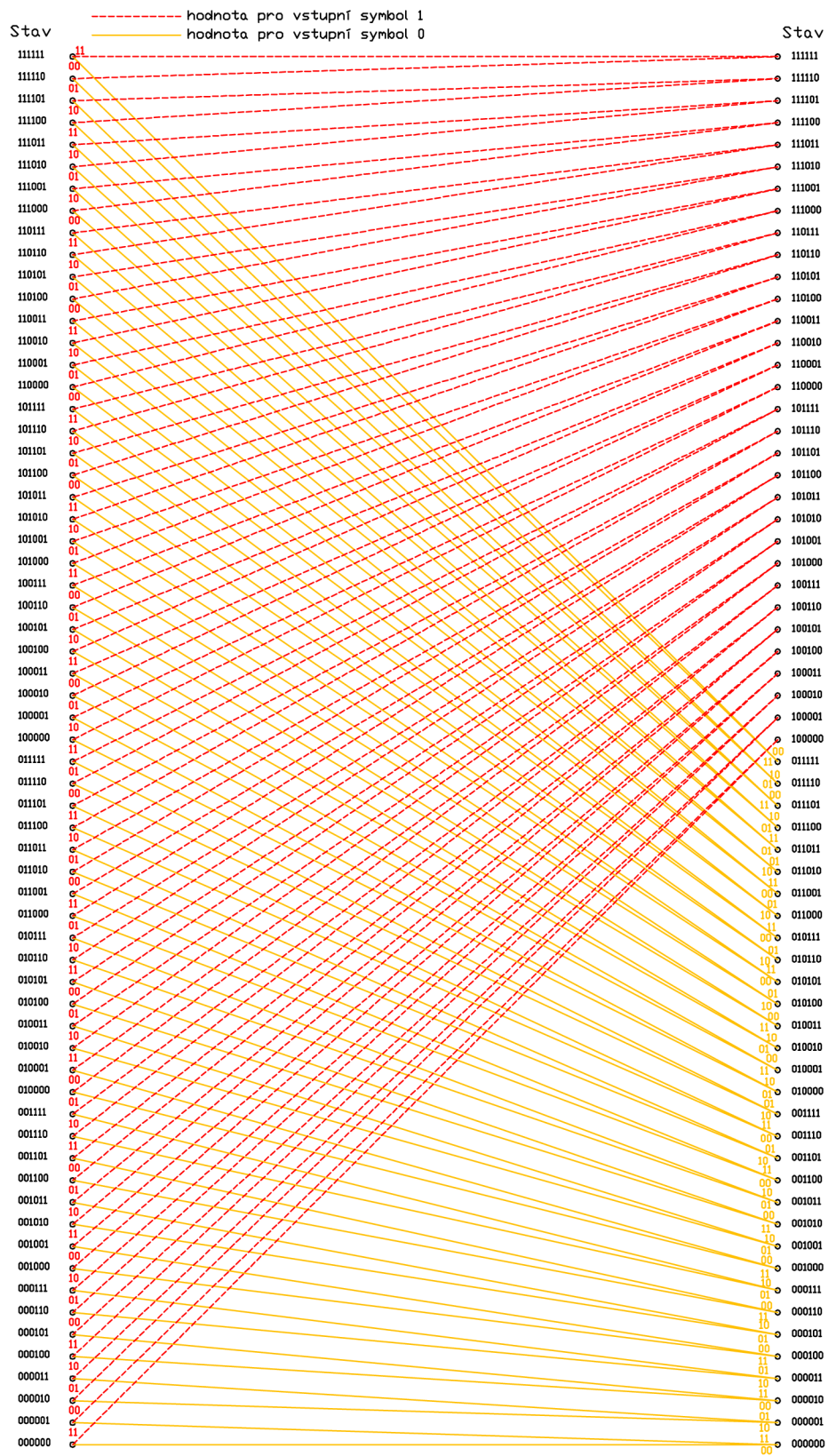
Vytvářecí mnohočleny nám tedy určují konvoluční kód (14, 7, 6) s parametry:

$$n_0 = 2, k_0 = 1, t = 3, K = 6, n_A = 14, P/Q' = 4/7.$$

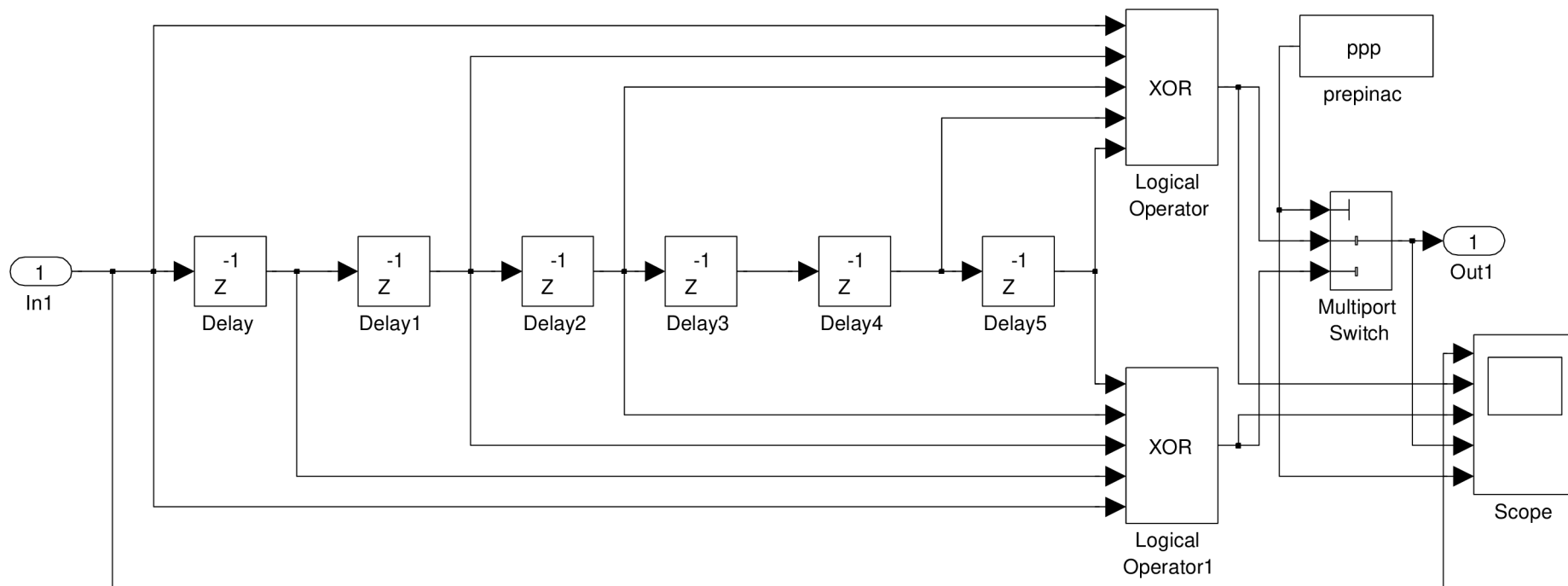
Obvod kodéru tohoto kódu je znázorněn na obr. 1.6 a mřížový diagram kodéru na obr. 1.7.



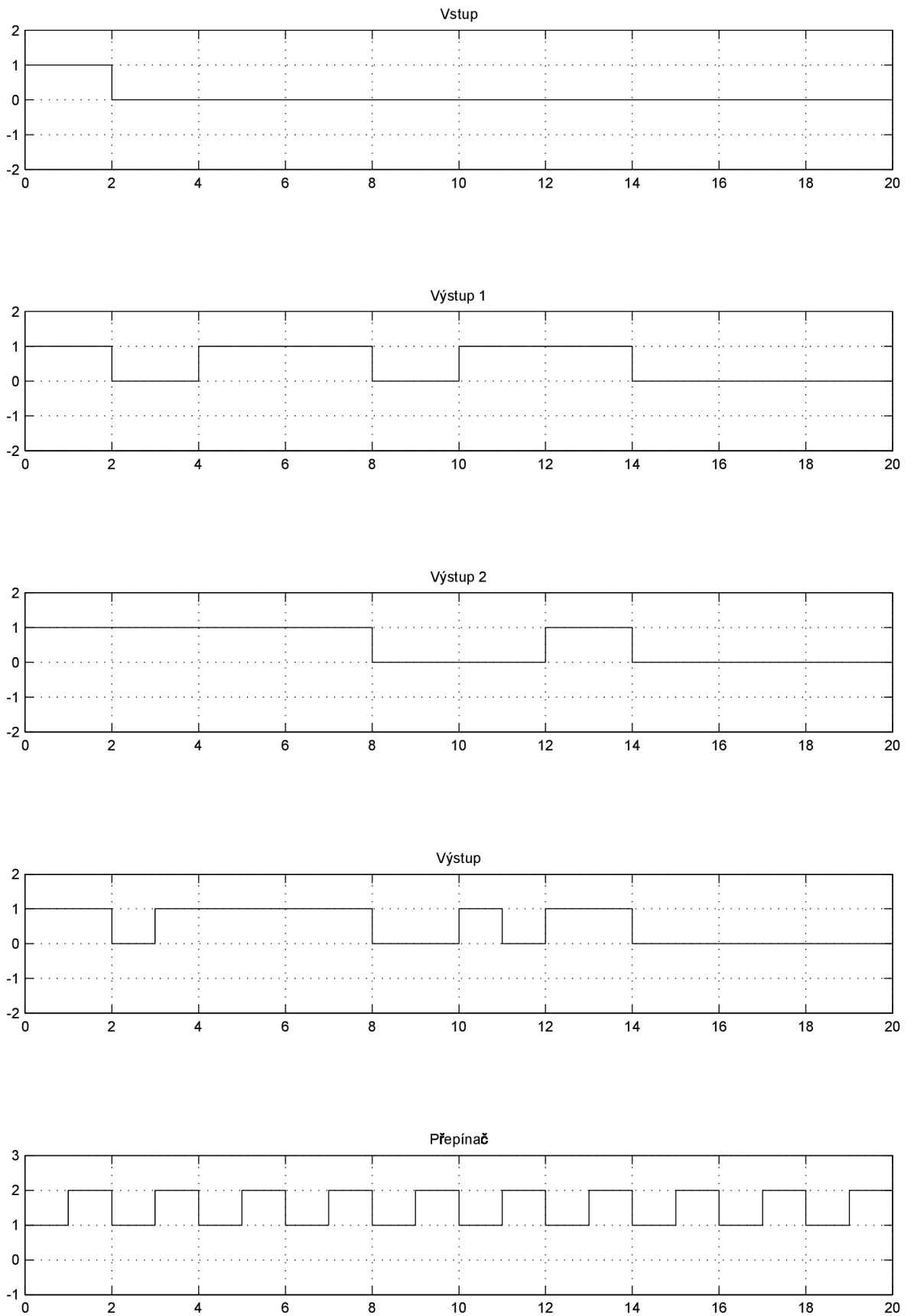
Obr. 1.6: Kódovací obvod pro kód (14, 7, 6).



Obr. 1.7: Mřížový diagram pro kód (14, 7, 6).



Obr. 1.8: Návrh a realizace kodéru v programu Matlab Simulink.



Obr. 1.9: Vstupní a výstupní parametry kodéru.

2 ZÚŽENÍ A ZPĚTNÉ ROZŠÍŘENÍ KONVO- LUČNÍCH KÓDŮ

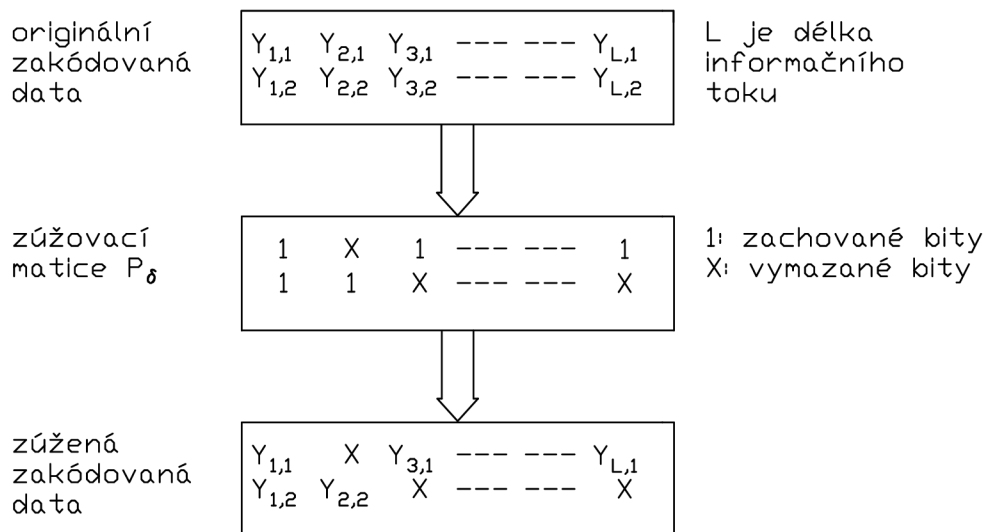
2.1 Zúžení a zpětné rozšíření

Podle periodického odstraňování bitů pomocí zúžovací matice. Tyto kódy umožňují vyšší poměr kódu, který dává vyšší dekodovací zisk, zatím co nedochází k problémům při realizaci z velké hodnoty k . Pokud je struktura kodéru s nízkým poměrem kódu $1/n$, pak existuje 2^k výpočtů pro každý uzel v dekodovací mříži, která je vhodná pro praktickou realizaci. Zúžené kódy jsou lépe pochopitelné z obrázku, pro detailnější příklad [3]. Předpokládejme základní (mother) kód, nebo standardní stavbu kodéru, kde je poměr $1/2$. Pro k vstupních bitů je n výstupních bitů. Předpokládejme posloupnost bitů s délkou L , tedy blok představuje L výstupních párů bitů. Poměr P/Q zúženého konvolučního kódu může být získán z poměru $1/n$ nezúženého konvolučního kódu vymazáním $n \cdot P - Q$ symbolů z každého $n \cdot P$ kódovaného bitu odpovídající kódování P informačních bitů podle poměru $1/n$. Poměr konvolučního kódu je $R_c = \frac{P}{n \cdot P - \delta}$, kde $\delta = n \cdot P - Q$. Vymazání bitů kódu je popsáno jako $n \cdot P - Q$ zúžené matice P_δ . Prvky zúžovacího pole jsou nuly a jedničky, odpovídající zachování nebo vymazání zakódovaných bitů podle pořadí v poli. Zúžovací zařízení vymaže symboly z kódované posloupnosti podle zúžovací matice. Na obr. 2.1 je ukázán příklad jak zařízení funguje [12].

Výstupní posloupnost z obr. 2.1 je:

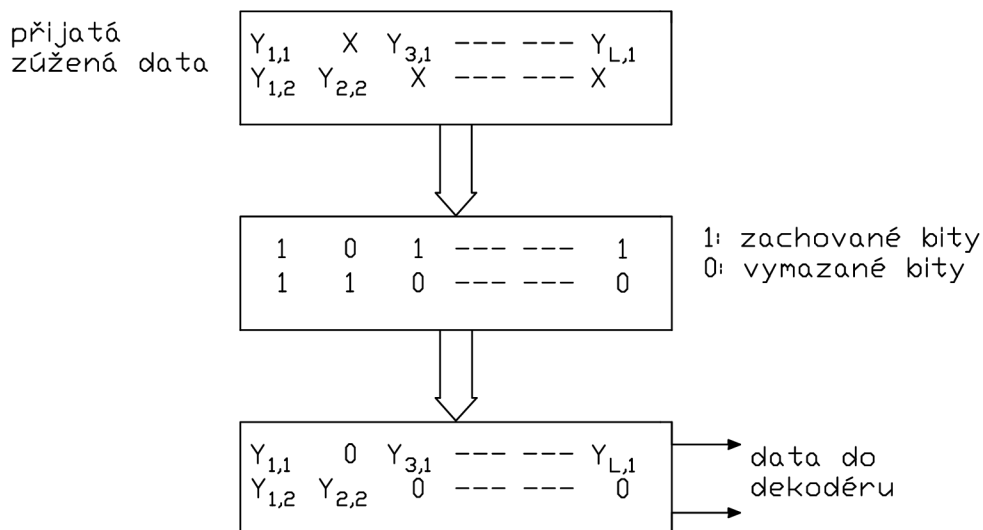
$$\mathbf{y} = (Y_{1,1}Y_{1,2}, XY_{2,2}, Y_{3,1}X, \dots, Y_{L,1}X), \quad (2.1)$$

Nezapomeňme, že X se značí data, která jsou vymazána. Podívejme se na první dvě výstupní posloupnosti $\mathbf{y}' = (Y_{1,1}Y_{1,2}, XY_{2,2})$, pro každé dvě vstupní posloupnosti (vstupní hodnota pro výstupní hodnotu $Y_{1,1}Y_{1,2}$ a vstupní hodnota pro výstupní hodnotu $Y_{1,1}X$), kde jsou tři výstupní posloupnosti (přenesené hodnoty y'). Všechny zúžené konvoluční kódy mohou být dekodovány použitím Viterbiho dekodéru. V postupu využijeme výhody jednoduchého $1/n$ základního (mother) kódu.



Obr. 2.1: Zúžení.

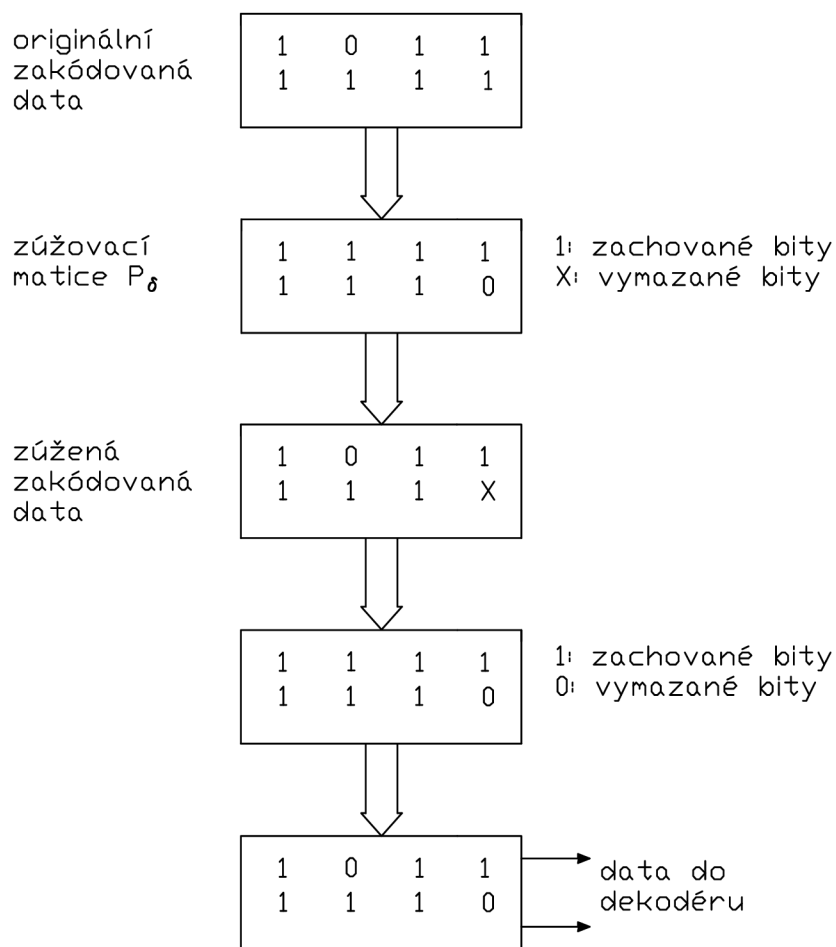
Zúžená kódovaná data musí být vrácena zpět do poměru $1/n$ před vstupem do kodéru. Vymazaná data jsou zpět přidána do přijaté zúžené posloupnosti k umožnění dekodéru pracovat na základním $1/n$ kódu nikoliv na zúženém poměru P/Q . Z předchozí znalosti o zúžovací matici P_δ . Vymazané bity jsou zpět vloženy do pozic dříve vymazaných bitů zúžené zakódované zprávy, přicházející z vysílače. Na obr. 2.2 je zobrazeno, jakým způsobem jsou bity přidány do posloupnosti [12]. Vstupní po-



Obr. 2.2: Zpětné rozšíření.

sloupnost do dekodéru je $\mathbf{r} = (Y_{1,1}Y_{1,2}, 0Y_{2,2}, Y_{3,1}0, \dots, Y_{L,1}0)$, kde nuly představují nějaké konstantní hodnoty pro vymazané bity. Dekodér operuje na základním $1/n$ kódu.

Nyní si teoreticky ukážeme, jak budou data zúžena a zpětně rozšířena pro náš konkrétní příklad obr. 2.3 [12]. Výstupní data z kodéru jsou 11, 01, 11, 11, 00, 10, 11, zúžíme



Obr. 2.3: Zúžení a zpětné rozšíření pro konkrétní příklad.

a následně je zpětně rozšíříme podle zúžovacího pole \mathbf{P}_1

$$\mathbf{P}_1 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}.$$

Podle výše uvedených pravidel budeme zúžovat a zpětně rozšiřovat první čtyři páry bitů z výstupu kodéru.

Výstupní posloupnost po zúžení čtyř vstupních párů bitů:

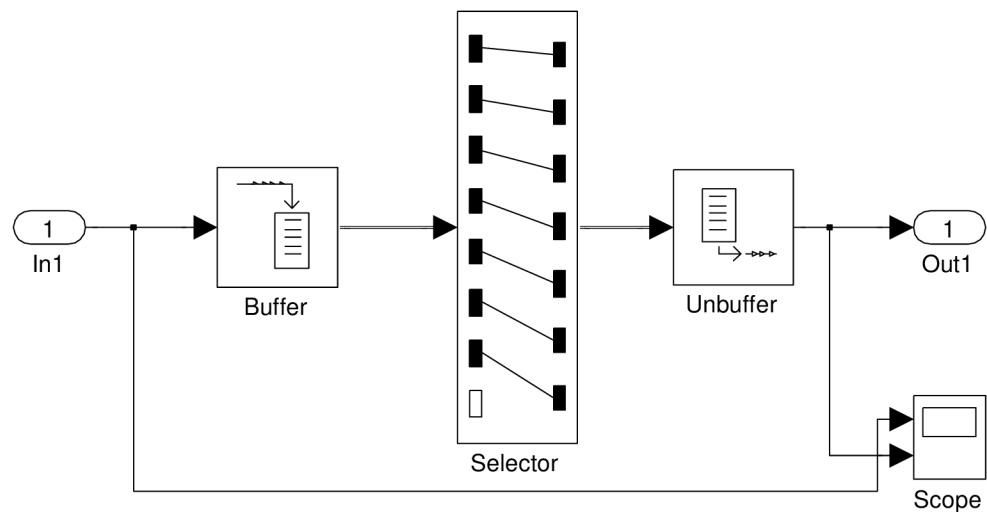
$$\mathbf{y} = (11, 01, 11, 1X).$$

Výstupní posloupnost po zpětném rozšíření čtyř vstupních párů bitů:

$$\mathbf{r} = (11, 01, 11, 10).$$

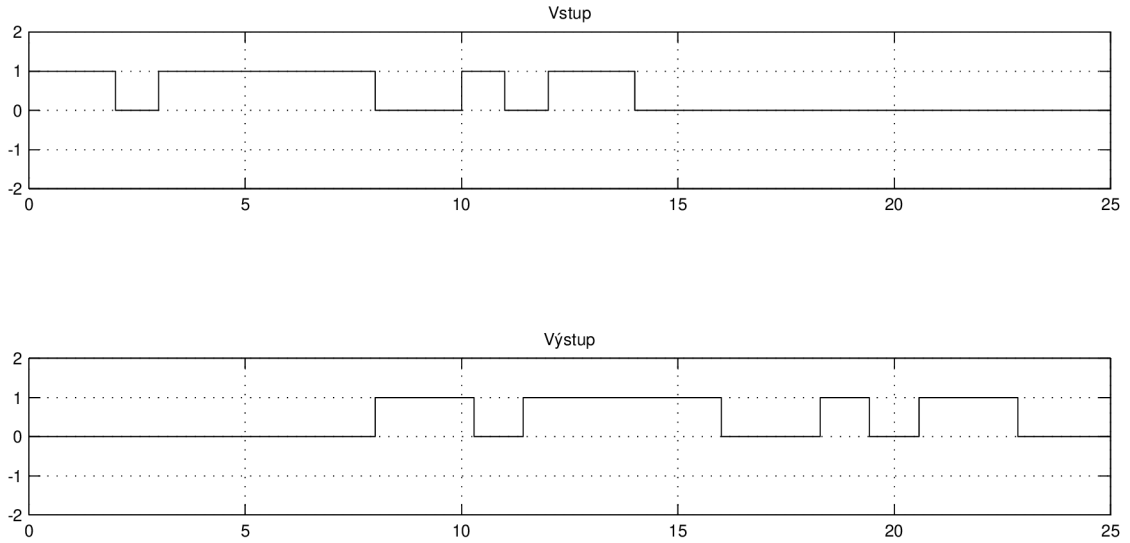
2.2 Návrh a realizace zúžovacího zařízení

Metoda vymazávání bitů [12] vyžaduje zvětšení zúžovací matice na vektor o délce $2 \cdot P$. Vstupní bity jsou poslány skrz blok selektoru, který napodobí pouze zvolené hodnoty dat odpovídající pozicím bitů nesoucích hodnotu 1 v zúžovací matici.



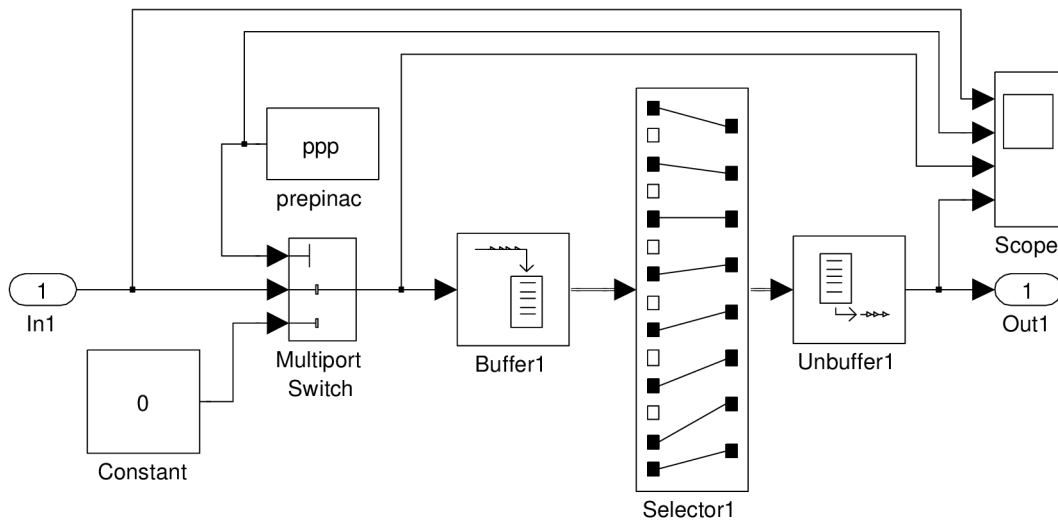
Obr. 2.4: Návrh a realizace zúžovacího zařízení v programu Matlab Simulink.

Podle zvoleného kódu, pokud je zúžovací matice $j = [1\ 1\ 1\ 1; 1\ 1\ 1\ 0]$, získáme vektor $v = [1\ 1\ 1\ 1\ 1\ 1\ 1\ 0]$, kde maximální délka vektoru v je $2 \cdot P$. Na pozicích ve vektoru v , kde jsou „1“ tj. 1. - 7. pozice, budou bity zachovány a ostatní vymazány. Tímto vznikne vektor o délce Q .



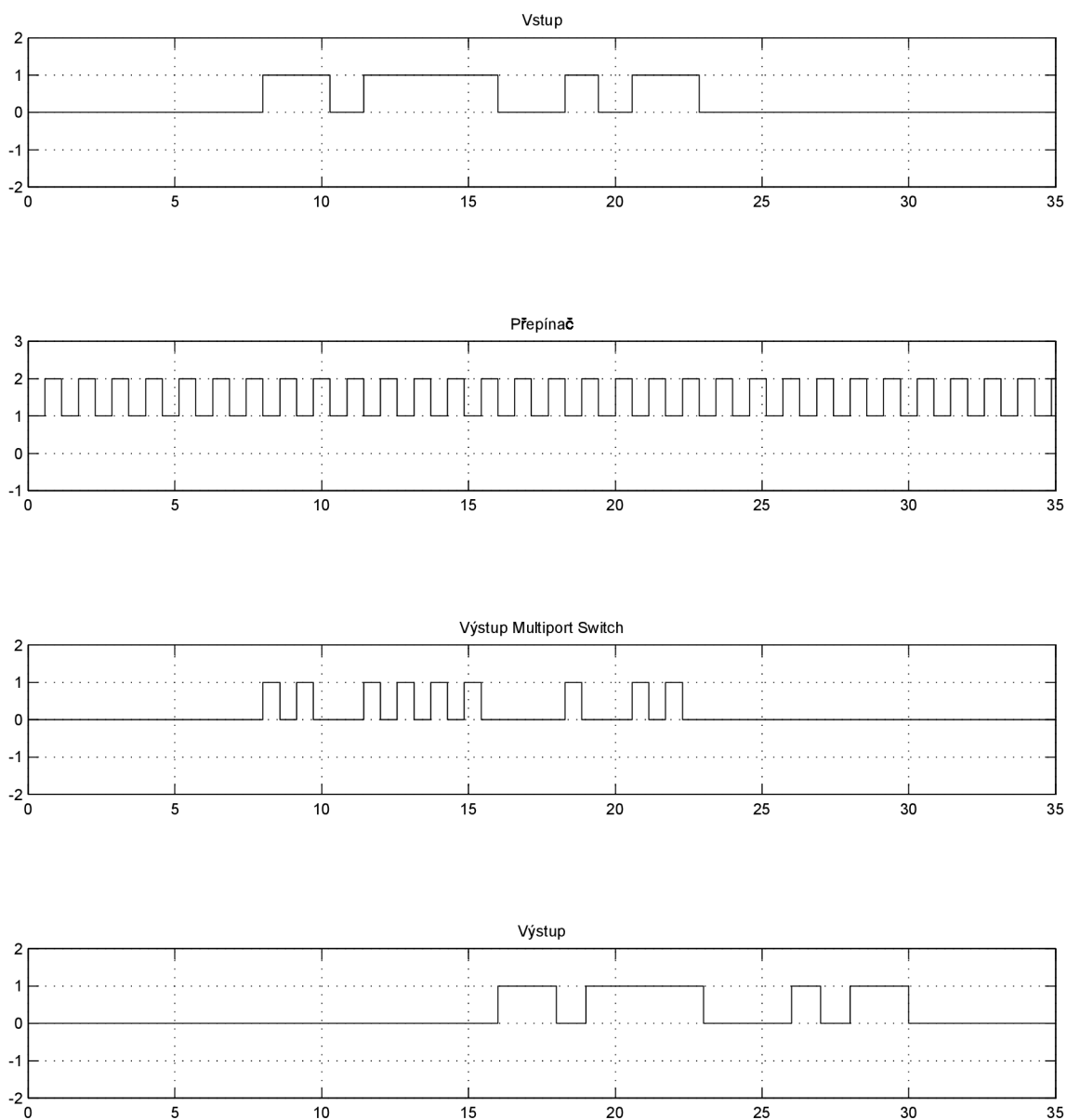
Obr. 2.5: Vstupní a výstupní parametry zúžovacího zařízení.

2.3 Návrh a realizace zpětně rozšiřujícího zařízení



Obr. 2.6: Návrh a realizace zpětně rozšiřujícího zařízení v programu Matlab Simulink.

Blok zpětné rozšíření [12] je v podstatě opak zúžovacího bloku obr. 2.4. Zúžovací blok o délce $2 \cdot P$ a výstupních symbolech o délce $2 \cdot Q$ nám vytváří zúžený po-



Obr. 2.7: Vstupní a výstupní parametry zpětně rozšiřujícího zařízení.

měr P/Q . Tato změna délky nám vytváří proměnné kódovací poměry. Blok zpětné rozšíření nám přidá z , kde z je počet nul v zúžovací matici. V tomto bloku se převádí proměnná délka a poměr kódu generovaného zúžovacím blokem zpět na originální poměr kódu $1/2$ pro Viterbiho dekódování. Použijeme zúžovací matici $p = [1111; 1110]$, $z = 1$, tedy 1 náhodná (dummy) hodnota „0“ je přidána do posloupnosti tak, aby vytvořila délku $2 \cdot P$, která byla na vstupu zúžovacího bloku.

Blok „Selektoru“ přeuspořádá novou délku vstupního vektoru $2 \cdot P$ tak, že umístí nuly na místa, která byla vymazána v zúžovacím bloku.

3 DEKÓDOVÁNÍ ZÚŽENÝCH KONVOLUČNÍCH KÓDŮ

3.1 Největší pravděpodobnost dekódování zúžených konvolučních kódů

Všechny zúžené konvoluční kódy lze dekódovat pomocí Viterbiho dekódovacího algoritmu. Dekódování poměru $P/(n \cdot P - \delta)$ zúženého konvolučního kódu Viterbiho algoritmem, který zahrnuje nalezení nejkratší cesty přes zúženou mříž ze $(n, 1, m)$ základního (mother) kódu. V procesu dekódování, je zakázán metrický výpočet podle odpovídajícího místa zúžení. Ve Viterbiho dekóderu nyní působí na mříži ze $(n, 1, m)$ základního (mother) kódu, spíše než z poměru mříže $P/(n \cdot P - \delta)$ zúženého konvolučního kódu. Z předchozí znalosti o zúženém poli \mathbf{P}_δ víme, že formální data jsou vložena do přijmaného pořadí $\{\mathbf{R}_l\} = \{\mathbf{Y}_l + \mathbf{E}_l\}$ odpovídající pozici symbolů zúženého kódu na vysílač, kde [7]

$$\mathbf{R}_l = [r_l^{(1)} r_l^{(2)} \dots r_l^{(n \cdot P - \delta)}], \quad (3.1)$$

$$\mathbf{Y}_l = [y_l^{(1)} y_l^{(2)} \dots y_l^{(n \cdot P - \delta)}], \quad (3.2)$$

a

$$\mathbf{E}_l = [e_l^{(1)} e_l^{(2)} \dots e_l^{(n \cdot P - \delta)}], \quad (3.3)$$

je dána modifikovaná posloupnost $\{\mathbf{R}'_l\} = \{\mathbf{Y}'_l + \mathbf{E}'_l\}$. Vektory $\{\mathbf{R}'_l\}$, $\{\mathbf{Y}'_l\}$ a $\{\mathbf{E}'_l\}$ mohou být popsány jako P -prvek vektorů

$$\mathbf{R}'_l = [\mathbf{R}'_{l,1} \mathbf{R}'_{l,2} \dots \mathbf{R}'_{l,P}], \quad (3.4)$$

$$\mathbf{Y}'_l = [\mathbf{Y}'_{l,1} \mathbf{Y}'_{l,2} \dots \mathbf{Y}'_{l,P}], \quad (3.5)$$

a

$$\mathbf{E}'_l = [\mathbf{E}'_{l,1} \mathbf{E}'_{l,2} \cdots \mathbf{E}'_{l,P}], \quad (3.6)$$

podle pořadí. Lze vidět, že každý subvektor 3.5 odpovídá na mřížové větvi $(n, 1, m)$ základního (mother) kódu a výstupu binárních symbolů kódu (zúžené a nezúžené symboly) přiřazené k d -té větvi v mříži základního (mother) kódu [7]

$$\mathbf{Y}'_{l,d'} = [y'_{l,d'}{}^{(1)} y'_{l,d'}{}^{(2)} \cdots y'_{l,d'}{}^{(n)}], \quad (3.7)$$

pro $1 \leq d' \leq P$. Optimální dekodér určuje kódovou posloupnost $\{\hat{\mathbf{Y}}'_l\}$, která má největší metrickou cestu k úpravě posloupnosti $\{\mathbf{R}'_l\}$. Je to provedeno pomocí Viterbiho algoritmu, který rekursivně maximalizuje metrickou cestu

$$M_l(\mathbf{R}'_0/\mathbf{Y}'_0, \mathbf{R}'_1/\mathbf{Y}'_1, \cdots, \mathbf{R}'_l/\mathbf{Y}'_l) = M_{l-1}(\mathbf{R}'_0/\mathbf{Y}'_0, \mathbf{R}'_1/\mathbf{Y}'_1, \cdots, \mathbf{R}'_{l-1}/\mathbf{Y}'_{l-1}) + M(\mathbf{R}'_l/\mathbf{Y}'_l), \quad (3.8)$$

v čase l přes celou posloupnost kódu $\{\mathbf{Y}'_l\}$ základního (mother) kódu. $M(\mathbf{R}'_l/\mathbf{Y}'_l)$ je metrická větev mezi vektory $\{\mathbf{R}'_l\}$ a $\{\mathbf{Y}'_l\}$ v čase l .

V praxi dojde k rozhodnutí, když dekodér hledá ψ segmenty v zúžené mříži základního (mother) kódu. Dekódovací délka vyhledávání ψ by měla být dostatečně velká k vyhnutí se podstatnému poklesu ve výkonu pravděpodobnosti chyby. V čase l , dekodér drží 2^K přežívajících cest (surviving path) s pevným vyhledáváním délky v $6x$ omezené délce v základním (mother) kódu a pracovat iterativně na P větvích v mříži základního (mother) kódu pomocí Viterbiho algoritmu. Spojená s každou cestou (přežívající cestou) odpovídající běžící metrické cestě $M_l(\mathbf{R}'_0/\mathbf{Y}'_0, \mathbf{R}'_1/\mathbf{Y}'_1, \cdots, \mathbf{R}'_l/\mathbf{Y}'_l)$. Dekodér nyní vypočítává metrické větve pro všechny cesty vstupující do uzlu v mříži v čase l podle 3.8. Pro každý uzel v mříži základního (mother) kódu si dekodér zvolí cestu s největší metrikou. Vyplývá 2^K přežívajících cest, které jsou uloženy spolu s přiřazenou metrikou. Po P iteracích

mřížových větví, dekodér vytvoří pevné rozhodování (firm-decision) a dekoduje informační bity spojené s cestou s maximální metrikou s vědomím unikátního jednotkového převodového (one-to-one) mapování mezi posloupností zpráv základního (mother) kódu a kódu posloupnosti $\{\mathbf{Y}'_l\}$. Tento proces se opakuje tímto způsobem na následujících vektorech \mathbf{R}_{l+1} . Dále jsou uvedeny následující funkce.

1. Existuje 2^K metrických výpočtů během stavu v mříži, které mají být provedeny u Viterbiho dekodéru operujícího na mříži o poměru $P/(n \cdot P - \delta)$ nezúženého konvolučního kódu, zatím co mají být provedeny pouze dva metrické výpočty za stav na Viterbiho dekodéru operujícího na mříži $(n, 1, m)$ základního (mother) kódu pro dekódovací poměr $P/(n \cdot P - \delta)$ zúženého konvolučního kódu.
2. Pro dekódování zúžených konvolučních kódů používáme mříž ze základního (mother) kódu. Viterbiho dekodér zná zúžené pole \mathbf{P}_δ , δ , periodu P a poměr základního (mother) kódu.

3.2 Největší pravděpodobnost dekódování poměr - přizpůsobení (rate - compatible) zúžených kon- volučních kódů

Dekódování RCPC kódů je velmi podobné dekódování zúžených konvolučních kódů. Pro zúžené konvoluční kódy Viterbiho dekodér operuje na mříži $(n, 1, m)$ základního (mother) kódu. Z předcházející znalosti o zúžených polích \mathbf{P}_δ víme, že formální data jsou vložena do přijaté posloupnosti \mathbf{R}'_l odpovídající pozicím symbolů zúžených kódů, poskytnutým upravené posloupnosti \mathbf{R}'_l vysílači. V dekódovacím postupu je zakázán metrický výpočet odpovídající zúžené pozici. Pro dekódování RCPC kódů používáme mříž základního (mother) kódu. Viterbiho dekodér zná všechna zúžená pole \mathbf{P}_δ , δ , zúženou periodu P a poměr základního (mother) kódu [7] [8].

3.3 Návrh a realizace dekodéru

Ze zadání jsme vypočítali, že se jedná o konvoluční kód (14, 7, 6), kde vytvářecí mnohočleny jsou $\mathbf{G}_{(1)}^{(1)} = 1 + D^2 + D^3 + D^5 + D^6$, $\mathbf{G}_{(2)}^{(1)} = 1 + D + D^2 + D^3 + D^6$ a zúžovací pole je

$$\mathbf{P}_1 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}.$$

V tomto případě použijeme tvrdé rozhodování tzn. budeme pracovat se dvěma úrovněmi (př. $0V = \log 0$, $1V = \log 1$). Mříž tohoto kódu je znázorněna na obr. 3.1. Za předpokladu, že počet pamětí $K = 6$ bude mít mříž 2^K tedy 64 stavů. Přechody znázorněné čárkovanou (červenou) čarou náleží vstupní informaci 1 a přechody znázorněné plnou (oranžovou) čarou náleží vstupní informaci 0. Všechny přechody znázorněné v mříži, jsou odvozeny ze stavové mříže obr. 1.7 pro kódér (14, 7, 6) [7] [13] [12] [14].

Pomocí následujících obrázků obr. 3.2, 3.3, 3.4, 3.5, 3.6 si ukážeme jakým způsobem pracuje Viterbiho dekodér. Pro jednoduchost byla zvolena zpráva 1 0 0 0 0 0 0.

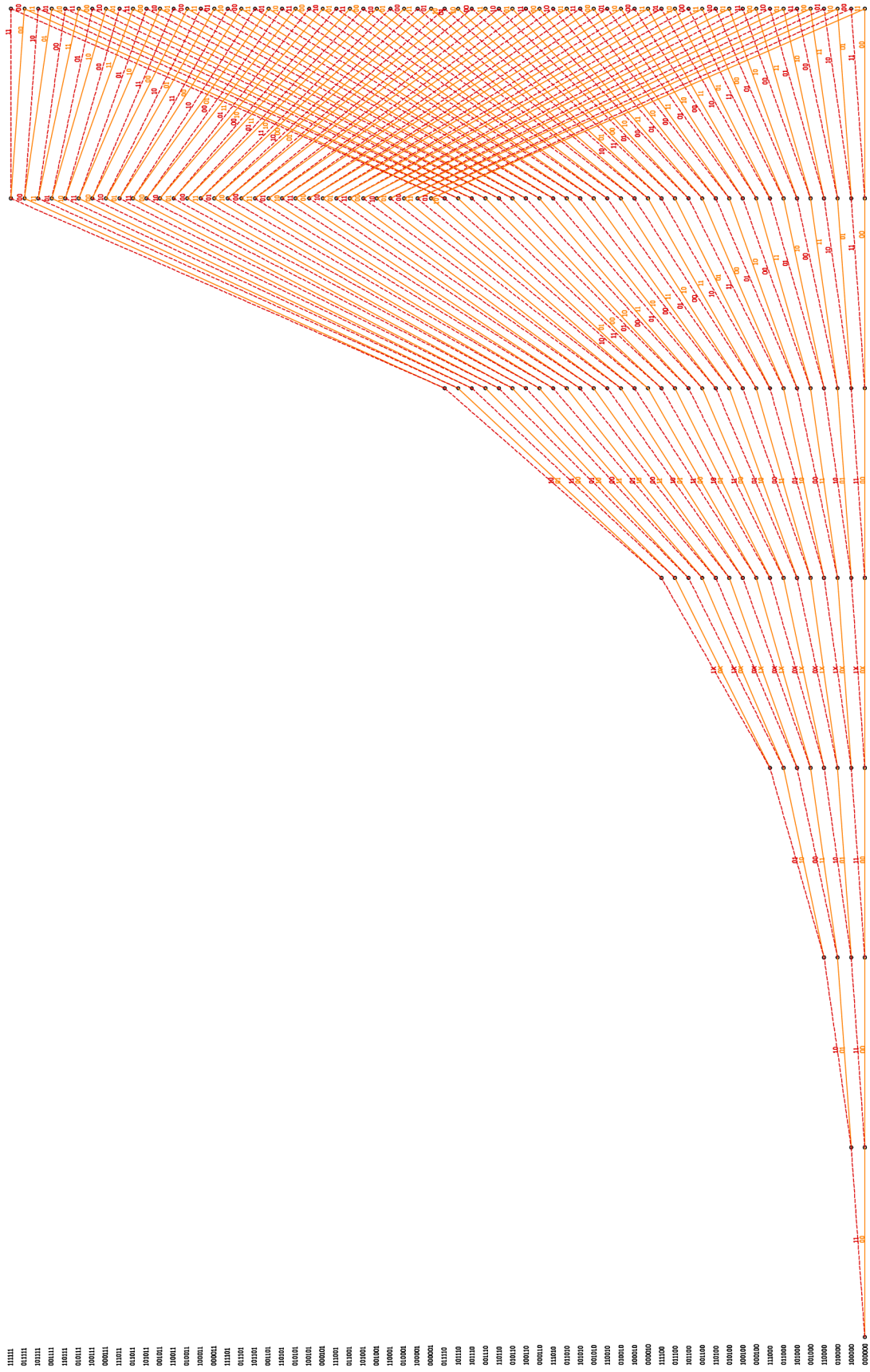
zpráva: 1 0 0 0 0 0 0

kódovaná zpráva: 11 01 11 1X 00 10 11

chyby v přenosu: 01 00 10 0– 01 00 00 1 - chyby v přijaté zprávě

přijatá zpráva: 10 01 01 1X 01 10 11 X - náhodná (dummy) data

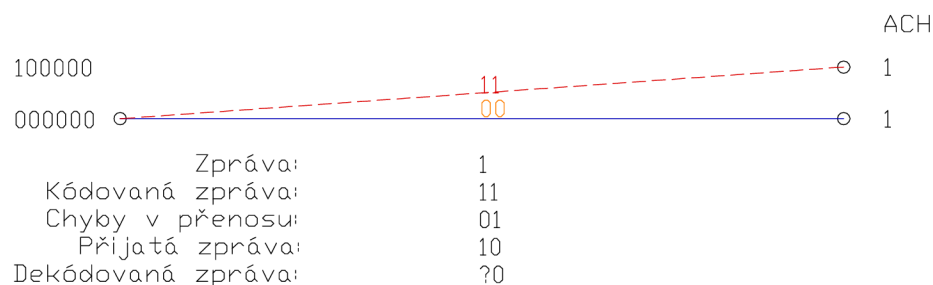
Při každém přijatém páru symbolů budeme počítat „vzdálenost“ mezi přijatým párem symbolů a všemi možnými páry symbolů, které bychom mohli přijmout. Metrikou bude Hammingova vzdálenost mezi přijatým a hypoteticky možným symbolem. Výsledkem tedy může být 0, 1 nebo 2. Tyto vzdálenosti nazývané metrika větve, budeme ukládat jako „akumulovanou chybovou hodnotu“ (ACH) vztaženou k danému stavu. ACH předchůdce + současná chybová hodnota = aktuální ACH.



Obr. 3.1: Mřížový diagram pro Viterbiho dekodér.

Nyní si ukážeme, jak se bude dekódování vyvíjet pro prvních pět časových kroků mřížového diagramu.

Pro přechod ze stavu v čase 0 do stavu v čase 1 máme hypoteticky pouze dva možné symboly, které by mohly být přijaty a to 00 nebo 11. Je to dáno apriorní znalostí inicializovaného stavu kodéru obr. 3.2.



Obr. 3.2: 1. stav mřížového diagramu Viterbiho dekodéru.

Protože nám vyšla ACH pro obě větve mřížového diagramu stejná, zvolíme tedy náhodně spodní větev pro 0 a o správnosti tohoto rozhodnutí rozhodne další krok v mříži Viterbiho dekodéru. Je tedy zřejmé, že v tomto kroku nastala chyba.

Pokračujeme s dalším (druhým) přijatým symbolem obr. 3.3. Vypočteme Hammingovy vzdálenosti jednotlivých přechodů mezi stavy pro přijatou hodnotu 01. Budeme zkoumat pouze vzdálenosti z koncových stavů pro přijetí prvního symbolu, tedy ze stavu 000000 a 100000.

Ze stavu 000000 do 000000 symbol 00 vzdálenost 1.

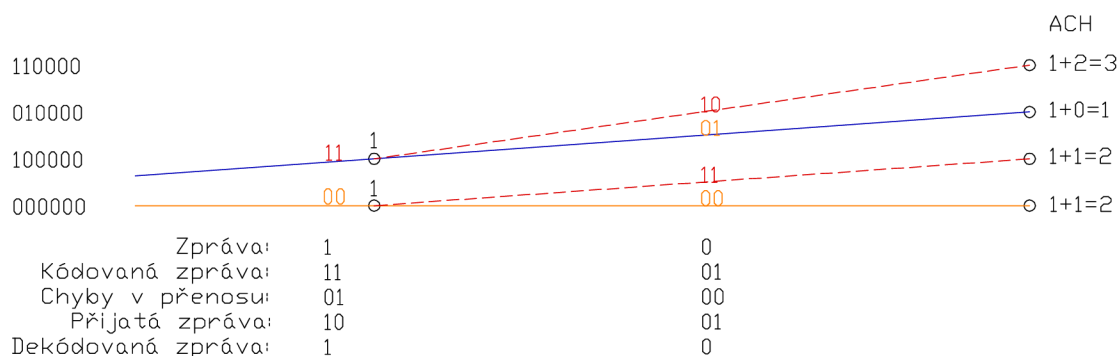
Ze stavu 000000 do 100000 symbol 11 vzdálenost 1.

Ze stavu 100000 do 010000 symbol 01 vzdálenost 0.

Ze stavu 100000 do 110000 symbol 10 vzdálenost 2.

K těmto hodnotám přičteme vzdálenosti přechodů, které byly 1 a 1. Nyní porovnáme

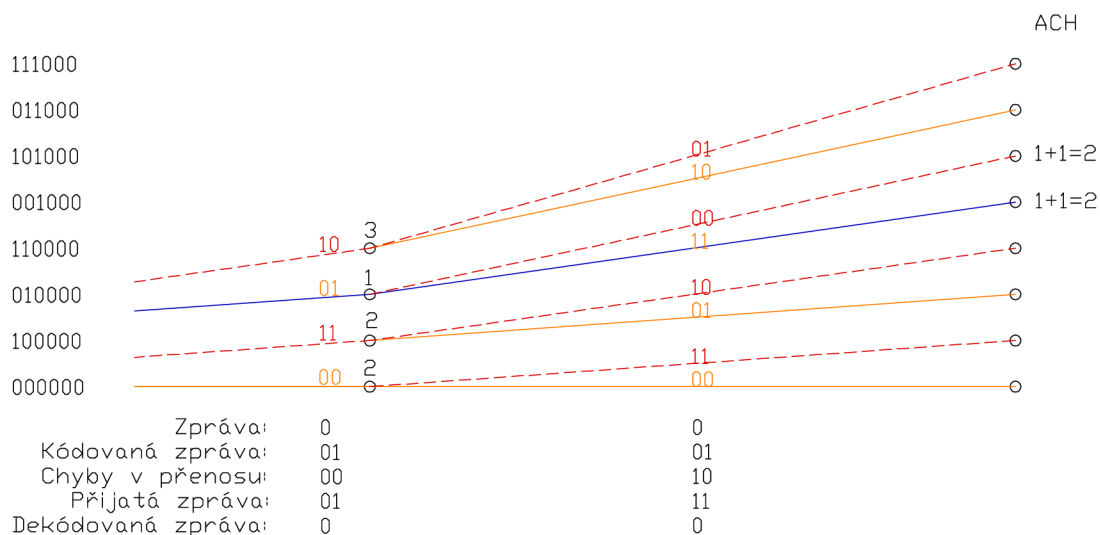
hodnoty ACH, vyšší zahodíme a pokračujeme dál jen s menší hodnotou.



Obr. 3.3: 2. stav mřížového diagramu Viterbiho dekodéru.

Z obrázku je zřejmé, že v předchozím kroku v mříži Viterbiho dekodéru byla zvolena špatná cesta, a proto je nutné opravit dekodovanou zprávu předchozího kroku.

Dalším přijatým symbolem je 01, který byl původně 11, přijímáme ho tedy s chybou. Vycházíme ze stavu 010000, pro který byla hodnota ACH 1. Pro obě



Obr. 3.4: 3. stav mřížového diagramu Viterbiho dekodéru.

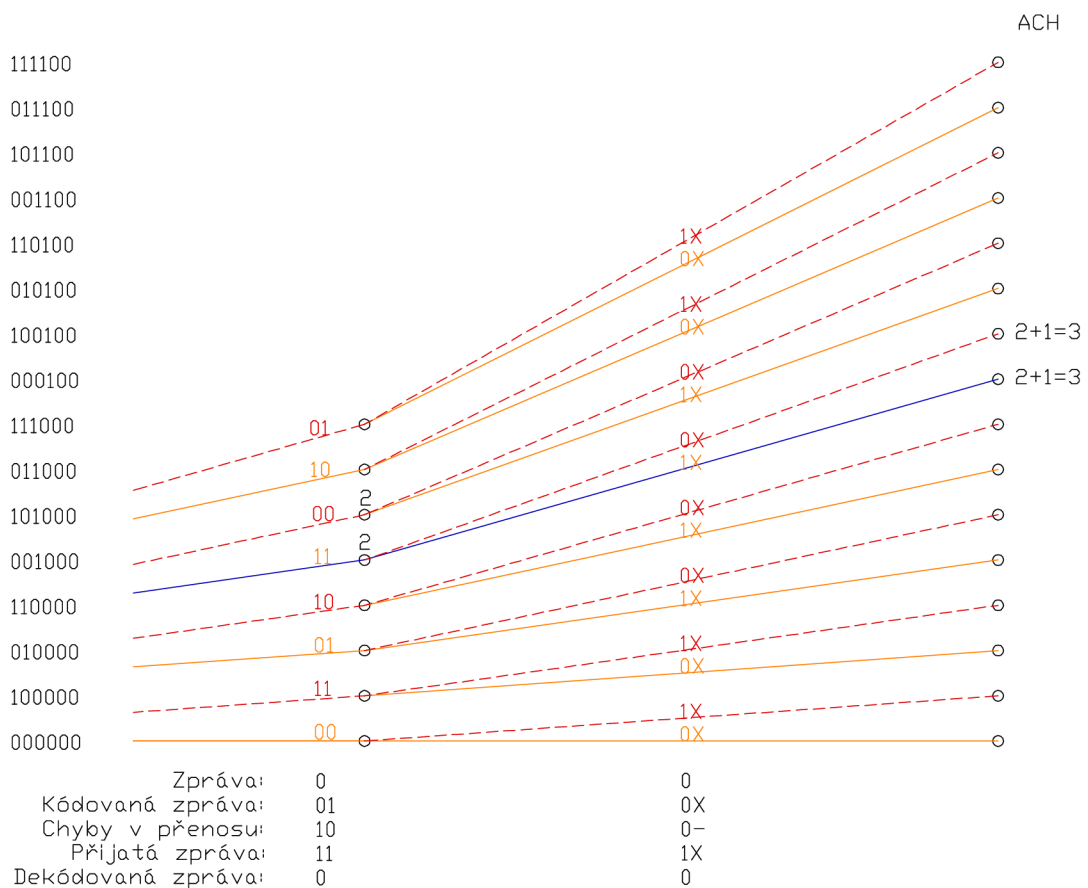
následující cesty nám nyní vyšlo ACH 2. Tento problém lze řešit dvěma způsoby:

- dostaneme-li ve dvou nebo více stavech stejnou hodnotu ACH provedeme náhodný výběr dalšího přechodu cesty s nejnižší Hammingovou vzdáleností

- dostaneme-li ve dvou nebo více stavech stejnou hodnotu ACH provedeme determinovaný výběr podle stanoveného pravidla (např. vždy větev dolů)

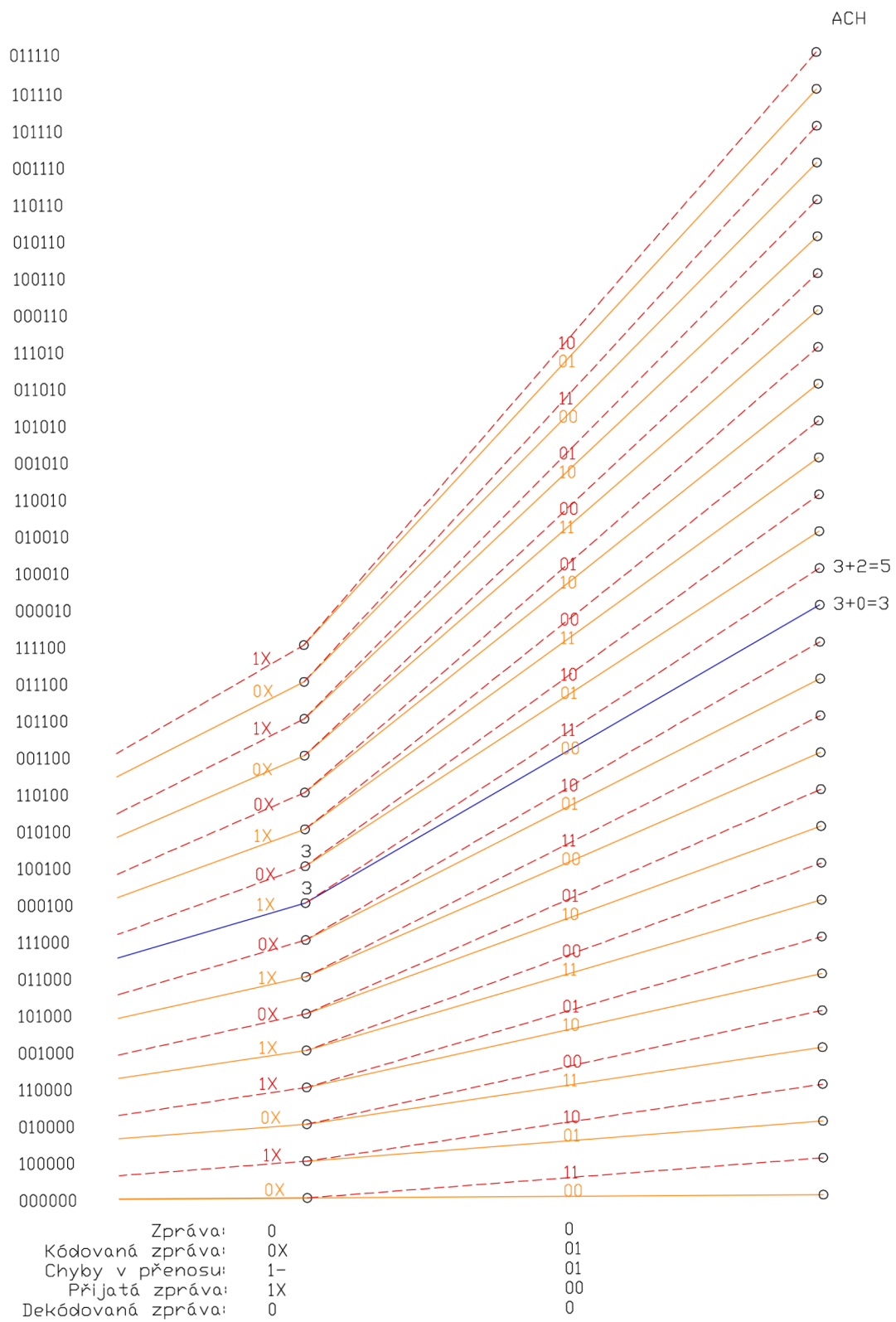
Obě varianty jsou si rovny a je jedno, kterou použijeme. Třetí krok je znázorněn na obr. 3.4.

Ve čtvrtém kroku je jeden z bitů nahrazen písmenem X, a to z důvodu použití zúženého konvolučního kódu. X znázorňují náhodná (dummy) data vložená zpětně rozšiřujícím zařízením. V tomto kroku může hodnota ACH nabýt pouze hodnotu 0 nebo 1, a to z důvodu, že data X jsou vložena až po přenosu dat, tudíž na této pozici nemůže nastat chyba obr. 3.5. V tomto kroku je opět nutné rozhodnout, kterou cestu zvolit z důvodu stejné hodnoty ACH podle výše uvedených pravidel.

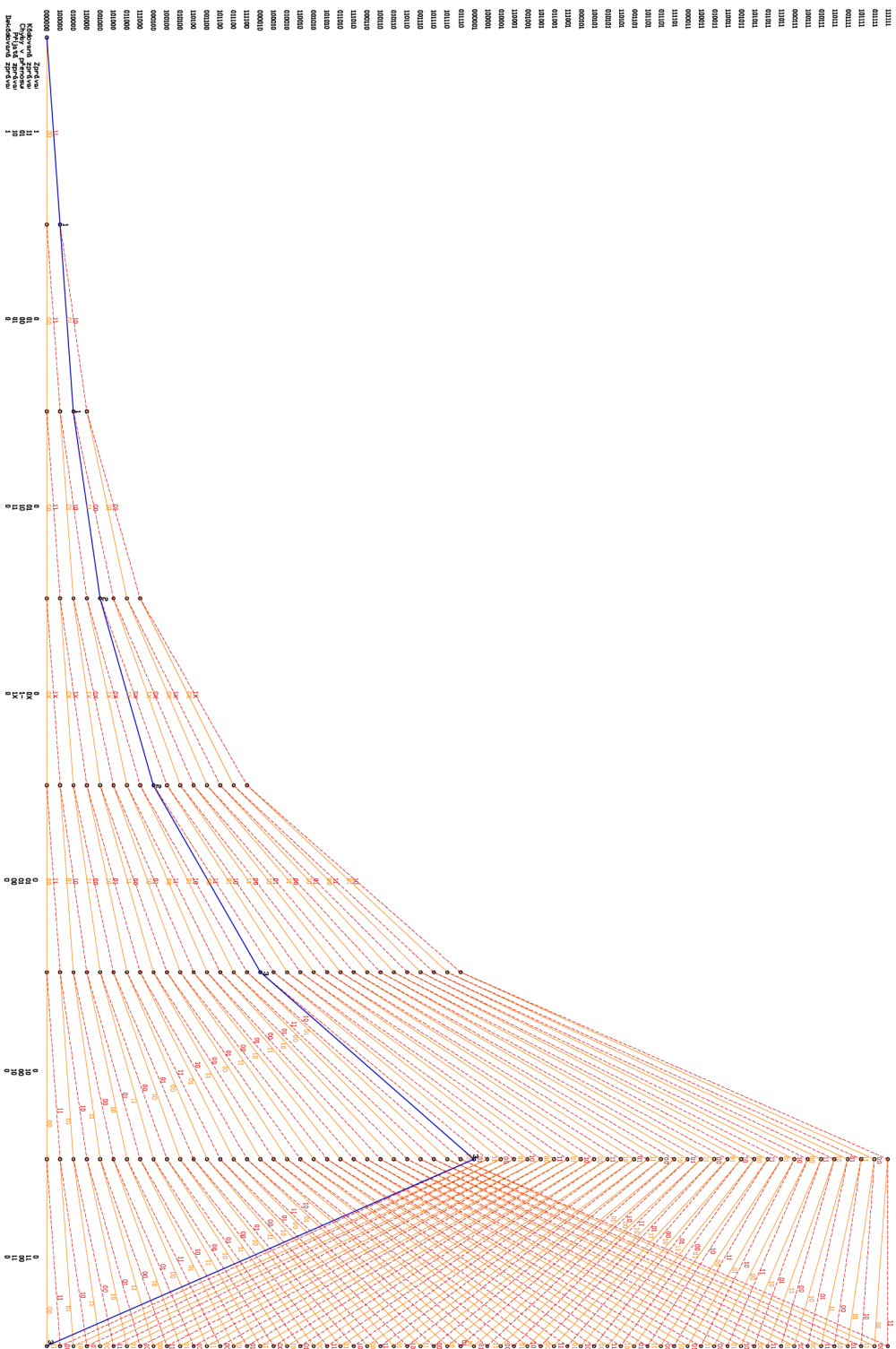


Obr. 3.5: 4. stav mřížového diagramu Viterbiho dekodéru.

Pátý krok je algoritmicky stejný jako kroky předchozí obr. 3.6.



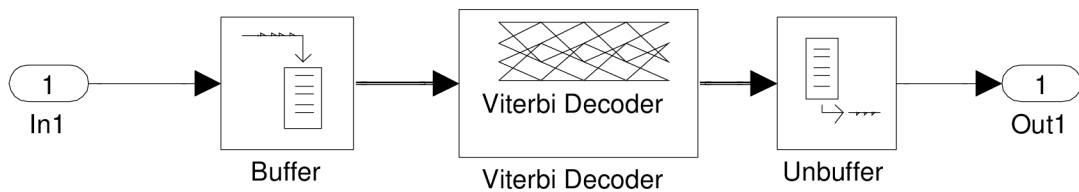
Obr. 3.6: 5. stav mřížového diagramu Viterbiho dekodéru.



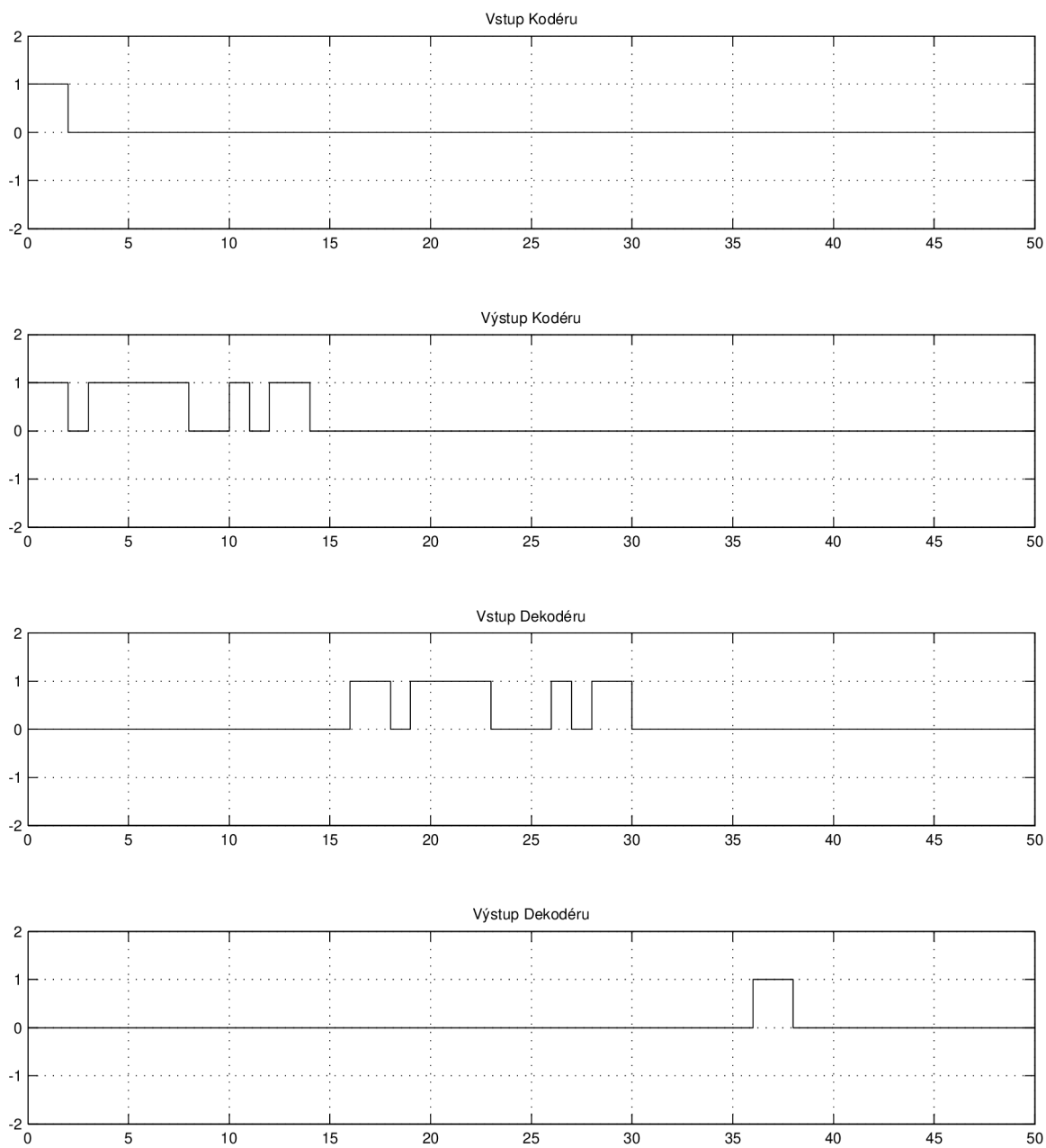
Obř. 3.7: Nejpravděpodobnější cesta dekódovacího procesu pro danou zprávu.

Následující kroky přeskočíme, protože postup při hledání cesty v mříži Viterbiho dekodéru se stále algoritmicky opakuje. Nejpravděpodobnější cesta dekódovacího procesu je znázorněna na obr. 3.7. Z tohoto obrázku je zřejmé, že zvolený kód vyhovuje zadání, protože je schopen zabezpečit přenos proti třem nezávislým chybám $t = 3$ v kódovém ohraničení $n_A = 14$.

Realizace Viterbiho dekodéru bývá zpravidla řešena pomocí integrovaného obvodu. Z tohoto důvodu je v této práci použit pouze blok Viterbiho dekodéru, protože návrh tohoto obvodu by byl velice složitý a sahal by za rámec této práce.



Obr. 3.8: Návrh a realizace dekodéru v programu Matlab Simulink.



Obr. 3.9: Vstupní a výstupní parametry dekodéru.

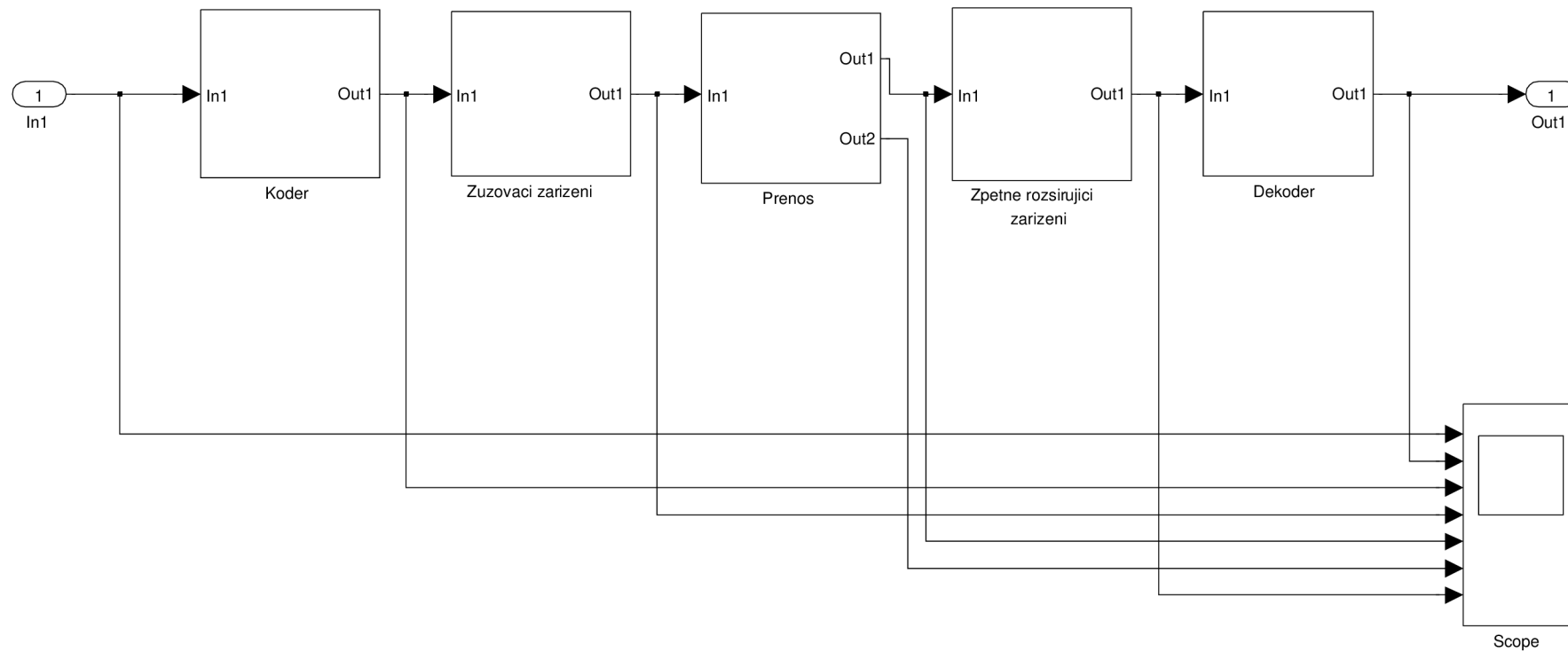
4 KODEK

4.1 Návrh a realizace kodeku

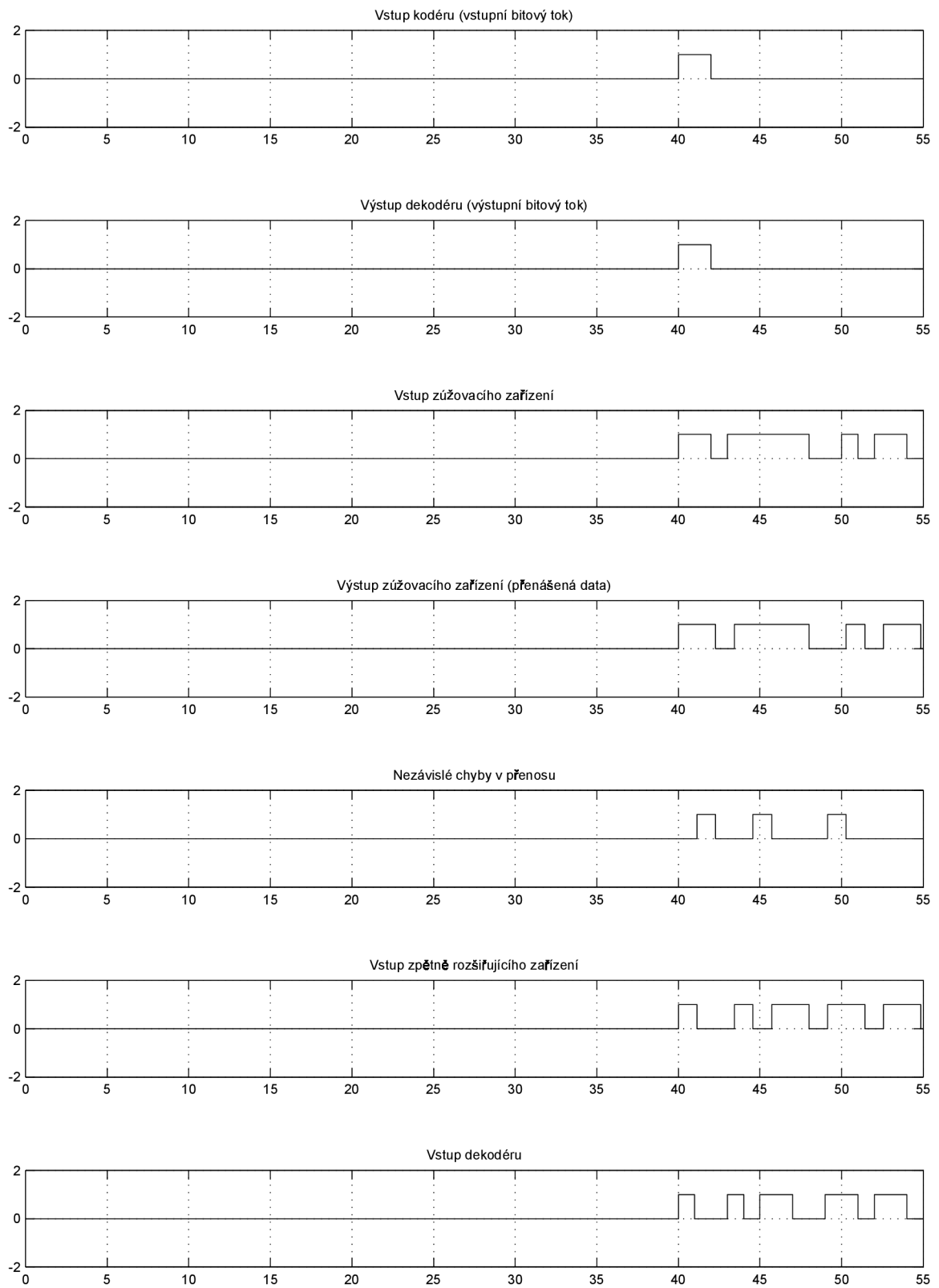
Návrh a realizace kodeku je znázorněna na obr. 4.1. Je zde použito pouze blokové schéma, a to z důvodu přehlednosti. Bloky jsou složeny z jednotlivých návrhů uvedených v předchozích kapitolách. Kodér obr. 1.8, zúžovací zařízení obr. 2.4, zpětně rozšiřující zařízení obr. 2.6 a dekodér obr. 3.8. Mezi blok zúžovacího zařízení a blok zpětně rozšiřujícího zařízení byl přidán blok přenosu. V tomto bloku jsou navozeny tři nezávislé chyby $t = 3$ v kódovém ohraničení $n_A = 14$. Vstupní a výstupní parametry kodeku byly zobrazeny na osciloskopu obr. 4.2. Z tohoto obrázku je zřejmé, že kodek je schopen zabezpečit přenos proti výše uvedeným chybám, tedy splňuje zadání. Vstupní a výstupní parametry byly zpožděny tak, aby bylo možné porovnávat vstupy a výstupy jednotlivých bloků, z tohoto důvodu se tyto parametry liší od parametrů uvedených v předchozích kapitolách.

Vstupy a výstupy jednotlivých bloků znázorněných na obr. 4.2:

- Vstup kodéru (vstupní bitový tok) – 1, 0, 0, 0, 0, 0, 0
- Výstup dekodéru (výstupní bitový tok) – 1, 0, 0, 0, 0, 0, 0
- Vstup zúžovacího zařízení – 11, 01, 11, 11, 00, 10, 11
- Výstup zúžovacího zařízení – 11, 01, 11, 10, 01, 01, 1
- Nezávislé chyby v přenosu – 01, 00, 10, 00, 10, 00, 0
- Vstup zpětně rozšiřujícího zařízení – 10, 01, 01, 10, 11, 01, 1
- Vstup dekodéru – 10, 01, 01, 10, 01, 10, 11



Obr. 4.1: Návrh a realizace kodeku v programu Matlab Simulink.



Obr. 4.2: Vstupní a výstupní parametry kodeku.

5 ZÁVĚR

V této práci bylo úkolem navrhnout zúžený konvoluční kód, který zabezpečí digitální přenos dat proti třem nezávislým chybám tj. $t = 3$, v kódovém ohraničení n_A a následně pro tento kód vypracovat návrh a realizaci kodeku v podobě samostatného systému, který je součástí protichybového kódového systému. Následně ověřit funkčnost tohoto kodeku metodou, která je pro tento návrh nejvhodnější.

Návrh zúženého konvolučního kódu je řešen převážně v první kapitole. Je zde znázorněno jakým způsobem byl kód vybrán a následně vypočítány parametry nutné k návrhu kodeku.

Samostatný kodek je navrhnout a realizován v programu Matlab Simulink a tvoří jej čtyři základní bloky. Jednotlivé bloky (kodér, zúžovací a zpětně rozšiřující zařízení, Viterbiho dekodér) jsou popsány ve třech kapitolách. Tyto kapitoly obsahují teoretický rozbor jednotlivých zařízení, návrh a realizaci a také ověření funkčnosti dané převážně vstupními a výstupními parametry jednotlivých bloků.

Ověření, blokový návrh a realizace kodeku je uvedena v poslední kapitole. Ověření bylo provedeno v programu Matlab Simulink tak, že byl simulován přenos pro kódové ohraničení n_A , ve kterém byly navozeny tři nezávislé chyby. Funkčnost je prokázána vstupními a výstupními parametry zobrazenými na osciloskopu. Z těchto parametrů je zřejmé, že zvolený kód i návrh kodeku splňují výše uvedené zadání.

LITERATURA

- [1] BEGIN, G., HACCOUN, D., CHANTAL, P. Further Results on High Rate Punctured Convolutional Codes for Viterbi and Sequential Decoding. IEEE Trans. Commun., vol. COM-38, No. 11, November 1990, pp. 1922-1928.
- [2] ČÍKA, P., KŘIVÁNEK, V. Výběr nejvhodnějšího konvolučního kódu [online]. 2007. Dostupný z URL: <<http://access.feld.cvut.cz>>.
- [3] CAIN, J., CLARK, G., GEIST, J. Punctured Convolutional Codes of Rate $(n-1)/n$ and Simplified Maximum Likelihood Decoding. IEEE Transactions on Information Theory, Vol. IT-25, No.1, pp. 97-100, Jan. 1979.
- [4] FARELL, P.G., MOREIRA, J.C. Essentials of Error-Control Coding. West Sussex, England: John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, 2006. ISBN-13 978-0-470-02920-6.
- [5] HOFFMAN, D.G., LEONARD, D.A, LINDNER, C.C, PHELPS, K.T., RODGER, C.A., WALL, J.R. Coding Theory The Essentials. New York, Basel, Hong Kong: Marcel Dekker Inc., 1991. ISBN 0-8247-8611-4
- [6] HUFFMAN, W.C., LEONARD, D.A, LINDNER, C.C, PHELPS, K.T., RODGER, C.A., WALL, J.R. Fundamentals of Error-Correcting Codes. United Kingdom: Cambridge University Press, 2003. ISBN 0-521-78280-5.
- [7] LEE, L.H.C. Convolutional Coding - Fundamentals and Applications. Boston, London: Artec House, 1997. ISBN 0-89006-914-X.
- [8] LEE, L.H.C. New Rate-Compatible Punctured Convolutional Codes for Viterbi Decoding. IEEE Trans. Commun., vol. COM-42, pp. 3073-3079, Dec. 1994.
- [9] LIN, S., COSTELLO, D.J. Error Control Coding: Fundamentals and Applications. New Jersey: Prentice Hall, Inc. Englewood Cliffs, 1983. ISBN 0-13-283796-X.
- [10] NĚMEC, K. Datová komunikace. Skripta. VUT FEKT, Brno 2007.

- [11] PURSER, M. Introduction to Error-correcting codes. Boston, London: Artec House, 1995. ISBN 0-89006-784-8.
- [12] TAYLOR, C.L. Punctured Convolutional Coding Scheme for Multi-Carrier Multi-Antenna Wireless Systems. 2001. Dostupné z URL:
http://www.bwrc.eecs.berkeley.edu/publications/2001/THESES/punct_convolut_cod_scheme/ctaylor_thesis.pdf.
- [13] VUCETIC, B., JINGHONG, Y. Turbo Codes - Principles and Applications. Boston, Dortrecht, London: Kluwer Academic Publisher, 2004. ISBN 0-7923-7868-7.
- [14] YASUDA, Y., KASHIKI, K., HIRATA, Y. High rate punctured convolutional codes for soft-decision Viterbi decoding, IEEE Trans. Commun., vol. COM -32, pp. 315-319, Mar. 1984.