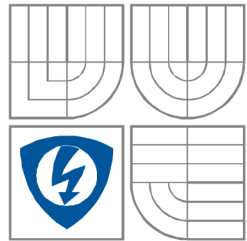


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNOLOGIÍ**
ÚSTAV RADIOELEKTRONIKY

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF RADIO ELECTRONICS

MODULÁTOR S ROZPROSTŘENÝM SPEKTRÉM

SPREAD SPECTRUM MODULATOR

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

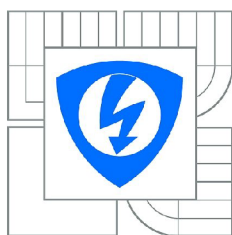
AUTOR PRÁCE
AUTHOR

Bc. ALENA LEJSKOVÁ

VEDOUCÍ PRÁCE
SUPERVISOR

doc. Ing. ROMAN MARŠÁLEK, Ph.D.

BRNO, 2011



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav radioelektroniky

Diplomová práce

magisterský navazující studijní obor

Elektronika a sdělovací technika

Studentka: Bc. Alena Lejsková

ID: 73105

Ročník: 2

Akademický rok: 2010/11

NÁZEV TÉMATU:

Modulátor s rozprostřeným spektrem

POKYNY PRO VYPRACOVÁNÍ:

Prostudujte problematiku přímého rozprostření spektra. Navrhněte principiální strukturu jednoduchého modulátoru a demodulátoru, využívajícího principu rozprostření. Seznamte se s posloupnostmi používanými pro rozprostření signálu, jejich generováním a vlastnostmi.

Simulujte výše navržený modulátor a demodulátor v MATLABu. Analyzujte možnost implementace jednotlivých bloků modulátoru v signálovém procesoru nebo obvodu FPGA.

Navržené bloky modulátoru s rozprostřeným spektrem implementujte dle volby buď v prostředí Xilinx ISE, nebo v simulačním prostředí signálového procesoru řady C6713.

DOPORUČENÁ LITERATURA:

[1] CHIUEH, T. D., TSAI, P. Y OFDM Baseband Receiver Design for Wireless Communications. Singapore: John Wiley & Sons Asia, 2007.

Termín zadání: 7.2.2011

Termín odevzdání: 20.5.2011

Vedoucí práce: doc. Ing. Roman Maršálek, Ph.D.

prof. Dr. Ing. Zbyněk Raida

Předseda oborové rady

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové 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 části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Práce se zabývá metodou rozprostření spektra datových signálů určených k přenosu rádiovým kanálem. Existují tři různé způsoby aplikace rozprostírání, k jejich realizaci se využívají tzv. pseudonáhodné posloupnosti, nebo také vzájemně ortogonální posloupnosti. V těchto systémech nalezneme uplatnění řady modulací, z nichž jedna z nich - čtyřstavové fázové klíčování - je uvedena. Cílem práce je seznámení s principem metody rozprostírání spektra a popis systémů, vysílačů a přijímačů, které pracují na základě této metody.

KLÍČOVÁ SLOVA

multiplex, rozprostřené spektrum, rozprostírání, autokorelace, ortogonální, pseudonáhodná posloupnost, sekvence, Hadamardova matice

ABSTRACT

This thesis deals with a method of spectrum-spreading of the data signals transmitted in the radio channel. There exist three different ways of spreading application, they uses pseudorandom sequences and also orthogonal sequences. In these systems we can find various kinds of modulations, one of them - Quadrature Phase Shift Keying - is shown. The thesis wants to present the principle of the spectrum spreading method and the description of systems, transmitters and receivers that are using this method.

KEYWORDS

multiplex, spread spectrum, signal, spreading, autocorrelation, orthogonal, pseudorandom, sequence, Hadamard matrix

LEJSKOVÁ, A. *Modulátor s rozprostřeným spektrem*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2011. 39 s., 2 příl. Vedoucí diplomové práce doc. Ing. Roman Maršálek, Ph.D.

Prohlášení

Jako autor diplomové práce na téma Modulátor s rozprostřeným spektrem dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušila autorská práva třetích osob, zejména jsem nezasáhla nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědoma následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne 20. kvěna 2011

.....
podpis autora

Poděkování

Děkuji vedoucímu diplomové práce doc. Ing. Romanu Maršálkovi, Ph.D. za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé diplomové práce.

V Brně dne 20. kvěna 2011

.....
podpis autora

OBSAH

Úvod	8
Rozprostírání spektra	9
Kódový multiplex CDM (Code Division Multiplex)	9
Princip rozprostírání signálů	9
Činitel rozprostírání (Spreading Factor)	11
Rozprostření signálu přímou kódovou posloupností (DS-SS)	11
Rozprostírání metodou kmitočtového skákání (FH-SS)	12
Rozprostírání metodou časového skákání (TH-SS)	13
Mnohonásobný přístup MC-DS-CDMA, MC-CDMA	13
Multiplex s Fourierovou transformací	14
Posloupnosti sloužící k rozprostření signálu	15
M-sekvence	15
Gold a Kasami sekvence	16
Walshovy funkce a sekvence	17
Energetická a spektrální účinnost	18
Čtyřstavové klíčování QPSK	19
UMTS	20
Rozprostírání kanálů pro uplink (DPCCH/DPDCH/HS-DPCCH)	21
Krátká scramblingová sekvence pro uplink	22
Dlouhá scramblingová sekvence pro uplink	23
Simulace rozprostření signálu v Matlabu	24
Implementace modulátoru a demodulátoru do obvodu FPGA	27
Obvody FPGA	27
Simulace v prostředí Xilinx ISE (verze 8.2)	27
Jazyk VHDL	27
Použití knihoven	29
Deklarace signálů	29
Součet signálů	29
Posuvný registr LFSR	31
Generátor nosné vlny	32
Modulace datového signálu	33
DCM – Digital clock manager	34
Časové průběhy signálů	34
Závěr	38
Literatura	39

SEZNAM OBRÁZKŮ

Obr.1: Princip rozprostírání při přenosu signálů více uživatelů v jednom kanále ([4]).	10
Obr.2: Odolnost proti úzkopásmovému rušení ([4]): a) signály během přenosu rádiovým kanálem a b) po derozprostření v přijímači.	10
Obr.3: Odolnost proti širokopásmovému rušení ([4]): a) rádiový kanál a b) po derozprostření v přijímači.	10
Obr.4: Příklad rozprostření datového signálu sčítáním mod 2, SF = 4.	11
Obr.5: Schéma vysílače s rozprostřeným spektrem DS-SS ([1]).	12
Obr.6: Schéma přijímače s rozprostřeným spektrem DS-SS s korelátorem ([1]).	12
Obr.7: Frekvenční skákání nosné vlny.	13
Obr.8: Vysílač a přijímač systému s frekvenčním skákáním nosné FH-SS podle [1].	13
Obr.9: Vysílač MC-CDMA, rozprostření frekvenčně [7].	14
Obr.10: Vysílač MC-DS-CDMA [7].	14
Obr.11: Autokorelace obecné posloupnosti.	15
Obr.12: Posuvný registr LFSR, m = 3.	15
Obr.14: Příklad generátoru Goldovy sekvence [3].	17
Obr.15: Strom prvků Walshovy posloupnosti.	18
Obr.16: Blokové schéma modulátoru QPSK [4].	19
Obr.17: Vektorový diagram a) QPSK a b) O-QPSK.	19
Obr.18: Blokové schéma demodulátoru QPSK [4].	20
Obr.19: Rozprostírání v systému UMTS (uplink).	20
Obr.20: Rozprostření datových kanálů a kontrolního kanálu pro uplink UMTS [7].	21
Obr.21: Rozprostření kanálů pro uplink UMTS [7].	22
Obr.22: Generátor krátké scamblovací sekvence pro uplink UMTS [7].	23
Obr.23: Generátor dlouhé scamblovací sekvence pro uplink UMTS [7].	24
Obr.24: Autokorelační funkce zvolené m-sequence.	25
Obr.25: Vzájemná korelace dvou různých m-sequencí.	26
Obr.26: Autokorelace zvolené Walshovy funkce a vzájemná korelace dvou různých Walshových funkcí.	26
Obr.27: Blokové znázornění modulátoru a demodulátoru s rozprostřením.	27
Obr.28: Schématický symbol komponenty Adder/Subtracter [9].	30
Obr.29: Schématický symbol LFSR [10].	31
Obr.30: Schématický symbol DDS [11].	32
Obr.31: Schématický symbol DCM [12].	34
Obr.32: Generování dvou různých hodinových kmitočtů pomocí bloku DCM.	35
Obr.33: Počáteční průběh signálů po restartu.	35
Obr.34: Časový průběh vstupních signálů.	36
Obr.35: Časový průběh výstupního signálu (Cout).	36
Obr.36: Průběh modulovaného signálu.	37

SEZNAM TABULEK

Tab.1: Násobení signálů a součet modulo 2 binárních signálů.	11
Tab.2: Mapování sekvence $z_n(i)$.	22

Úvod

Metoda rozprostření spektra je základním postupem k zabezpečení přenosu signálů v moderních telekomunikačních systémech, zejména mobilních. Chrání užitečný signál proti rušení vlivem šíření volným prostorem, dokáže pomoci proti příjmu nežádoucích signálů jiných uživatelů, které se v přijímači jeví také jako rušení. Samotným rozprostřením dochází k navýšení objemu dat, v konečném důsledku ale znamená kvalitnější příjem, než jaký by byl bez použití této metody.

K rozprostření slouží speciální kódy, zvané rozprostírací posloupnosti. Jedná se o binární posloupnosti, vytvořené pomocí daných matematických operací. Součástí této práce je rozdělení a popis jednotlivých druhů těchto kódů a ukázky jejich generování v matematickém programu Matlab a také ve vývojovém prostředí Xilinx ISE.

Rozprostírání spektra

Způsoby přenosu v komunikačních systémech

V telekomunikacích se využívá řada postupů, jak přenášet signály jednotlivých uživatelů tak, aby se vzájemně neovlivňovaly a daly se rozlišit v přijímači. Protože pro přenos více signálů je obvykle k dispozici jedna cesta, musí být signály sloučeny do jednoho signálu a po přenesení rádiovým kanálem jsou původní signály zpět obnoveny. Metoda sdružování signálů se nazývá multiplex a obnovení původních signálů demultiplex.

Rozlišujeme tři druhy multiplexů: multiplex s časovým dělením TDM (Time Division Multiplex), s frekvenčním dělením FDM (Frequency Division Multiplex) a multiplex s kódovým dělením CDM (Code Division Multiplex). Časový multiplex TDM přiděluje každému uživateli jiný časový úsek v průběhu komunikace, uživatelé se průběžně střídají. K přenosu stačí jediná nosná vlna a odpadá problém s nežádoucími intermodulacemi, jako u více frekvencí. Nevýhodou TDM mohou být problémy spojené s požadavkem na přesnou synchronizaci. Při frekvenčním multiplexu FDM je frekvenční pásmo rozděleno a každý uživatel využívá vlastní kanál. Uživatelé mohou tedy komunikovat současně. Toto řešení je náročné na šířku spektra a všechny intermodulační produkty je třeba odstranit, potlačit. FDM omezuje počet uživatelů v pásmech. Proto se v současných komunikacích využívá kódového multiplexu CDM, který umožňuje komunikaci více uživatelů současně na jedné frekvenci a rozlišení provádí systémy pomocí specifických kódů. Multiplex CDM využívá techniky rozprostírání spektra, která slouží k přenosu dat více uživatelů jedním kanálem.

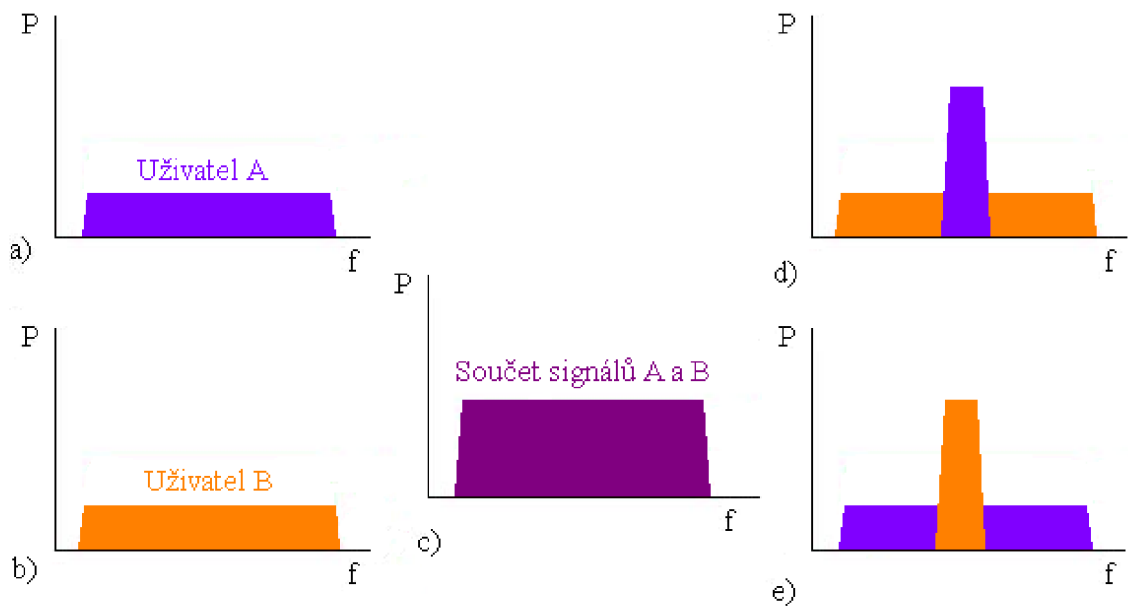
Kódový multiplex CDM (Code Division Multiplex)

Kódový multiplex CDM využívá techniky rozprostření spektra SS (Spread Spectrum). Signály všech uživatelů se přenášejí ve stejném frekvenčním pásmu a ve stejném čase a rozlišeny jsou pomocí pseudonáhodných nebo ortogonálních kódů. Mnohonásobný přístup využívající metody kódového multiplexu se nazývá CDMA (Code Division Multiple Access). Pod mnohonásobným přístupem se rozumí přístup do určitého rádiového prostředí a nerušený provoz všech uživatelů. Aby nedošlo k poruchám, je počet účastníků vždy omezen.

Podle způsobu rozprostření a přístupu se systémy CDMA dělí na DS-CDMA (Direct Sequence Spread Spectrum CDMA), FH-CDMA (Frequency Hopping CDMA), TH-CDMA (Time Hopping CDMA) a na hybridní systémy CDMA, které kombinují předchozí systémy (např. DS/FH-CDMA), nebo jsou kombinacemi základních protokolů (TDMA/CDMA apod.).

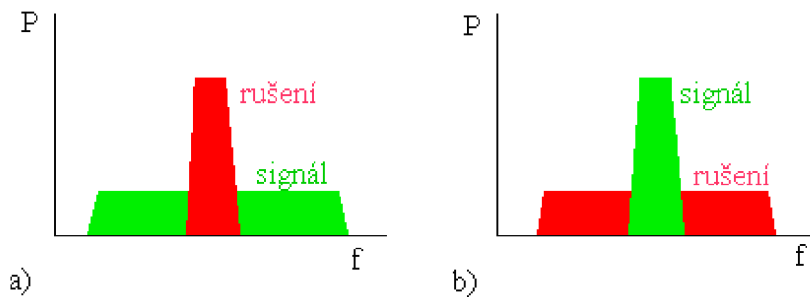
Princip rozprostírání signálů

Užitečný signál, který je přenášen v bezdrátových technologiích, má úzkopásmové spektrum. Na vysílací straně je signál vynásoben rozprostírající posloupností s širokým spektrem, výsledný signál má šířku rozprostírajícího signálu a je přenášen jako širokopásmový. Rozprostřené spektrum se označuje jako SS (Spread Spectrum). V přijímači jsou přijaty všechny signály, ale pouze užitečný signál se derozprostře stejnou posloupností zpět na úzkopásmový. V CDMA se tedy používá rozprostírání k identifikaci a oddělení jednotlivých uživatelů. Na přijímací straně je derozprostřen pouze signál konkrétního uživatele a signály od ostatních uživatelů jsou vnímány jako šum. Na Obr.1 a) a b) je znázorněn již rozprostřený signál uživatele A a B. Po projití rádiovým kanálem (Obr.1 c)) se v přijímači aplikuje příslušný kód a dostaneme zpět užitečný úzkopásmový signál (Obr.1 d) a e)).

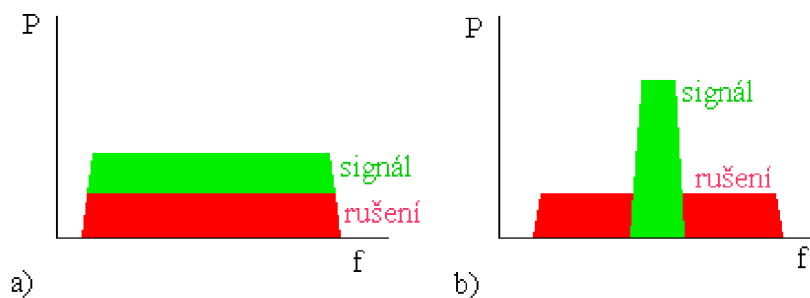


Obr.1: Princip rozprostírání při přenosu signálů více uživatelů v jednom kanále podle [4].

Na stejném principu pomáhá rozprostření chránit signál proti úzkopásmovému i širokopásmovému rušení během přenosu.



Obr.2: Odolnost proti úzkopásmovému rušení ([4]): a) signály během přenosu rádiovým kanálem a b) po derozprostření v přijímači.



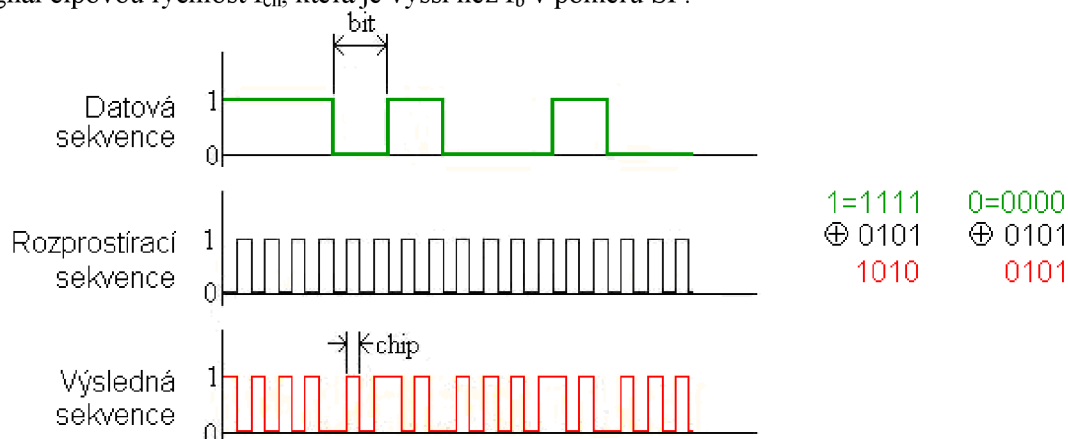
Obr.3: Odolnost proti širokopásmovému rušení ([4]): a) rádiový kanál a b) po derozprostření v přijímači.

Činitel rozprostírání (Spreading Factor)

Datová posloupnost užitečného signálu je rozdělena na bity. Během rozprostírání se mění bitová rychlost, proto se zavádí pojem čipy (též chipy) – prvky rozprostírající posloupnosti. Každý bit se transformuje na N čipů, poměr čipů a bitů je nazýván činitel rozprostření SF (Spreading Factor):

$$SF = \frac{N \text{ čipů}}{\text{bit}} \quad [-].$$

Původní přenosová rychlost dat je označována jako bitová rychlost f_b , po rozprostření má signál čipovou rychlost f_{ch} , která je vyšší než f_b v poměru SF.



Obr.4: Příklad rozprostření datového signálu sčítáním mod 2, SF = 4.

Při zpracování signálů je důležité rozlišit datové funkce a datové sekvence. Zatímco funkce jsou obvykle vyjádřeny opačnými úrovněmi -1 a 1, sekvence představují prvky binární soustavy (bity) 0 a 1. Vzájemně se dá vyjádřit úroveň signálu -1 jako bit 1 a úroveň 1 jako bit 0. Prvky funkcí se během sčítání dvou signálů mezi sebou násobí. Stejnému procesu s použitím bitů odpovídá součet modulo 2 (mod 2). Pokud signál nabývá pouze hodnot +1 a -1 (pouze kladné a záporné napětí), je označován jako NRZ (Non Return To Zero).

$s = s_1 \cdot s_2$			$s = s_1 \oplus s_2$		
s_1	s_2	s	s_1	s_2	s
1	1	1	1	1	0
1	-1	-1	1	0	1
-1	1	-1	0	1	1
-1	-1	1	0	0	0

Tab.1: Násobení signálů a součet modulo 2 binárních signálů.

Rozprostření signálu přímou kódovou posloupností (DS-SS)

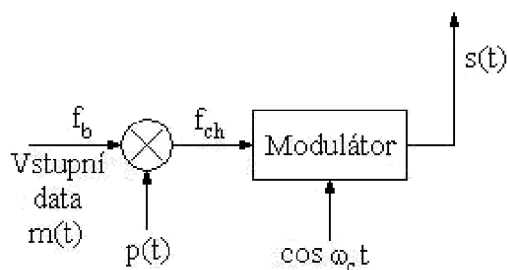
Metoda má anglický název Direct Sequence Spread Spectrum (DS-SS). Vstupní signál je vynásoben rozprostírající posloupností, a tím převeden přímo na signál s rozšířeným spektrem v poměru činitele rozprostření SF. Vysílač pro DS-SS se skládá z násobícího členu (resp. modulo

2), kde je binární signál $m(t)$ o bitové rychlosti f_b vynásoben rozprostírací posloupností $p(t)$ o chipové rychlosti f_{ch} . Signál s širokým spektrem se pak moduluje na nosnou vlnu v modulátoru BPSK, QPSK... Jinou variantou může být opačné zapojení, kdy se nejprve signál moduluje na nosnou vlnu a poté se rozprostře (viz [1]). Modulovaný signál

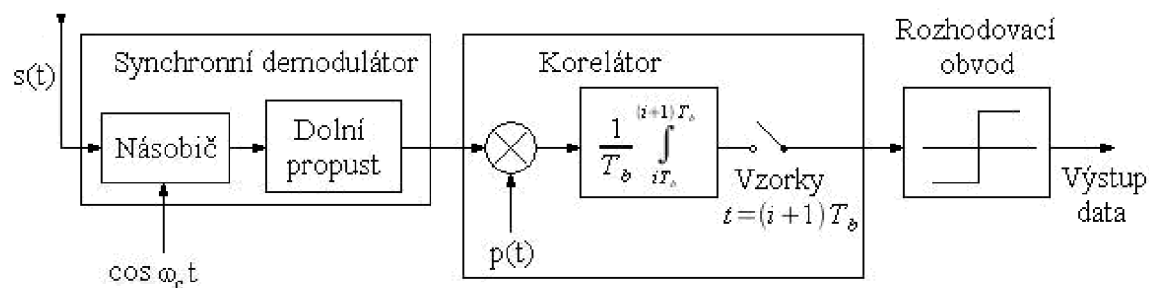
$$s(t) = m(t)p(t) \cos \omega_c t$$

se před samotným vysíláním v praxi obvykle ještě konvertuje do vyšších pásem a výkonově zesílí. V mobilních komunikacích se DS-SS používá v systémech GPS, GNSS, Wi-fi, či buňkových telefonech.

Přijímač s rozprostřeným spektrem se skládá z demodulátoru příslušné digitální modulace a korelátoru, kde se provádí zpětné zúžení spektra. Kódová posloupnost v přijímači musí být vždy v synchronizaci s přijímaným signálem.



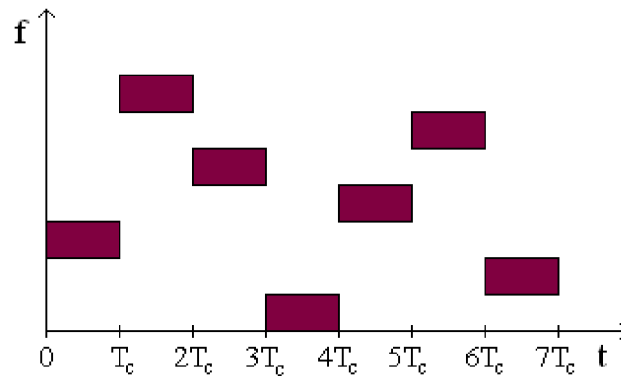
Obr.5: Schéma vysílače s rozprostřeným spektrem DS-SS ([1]).



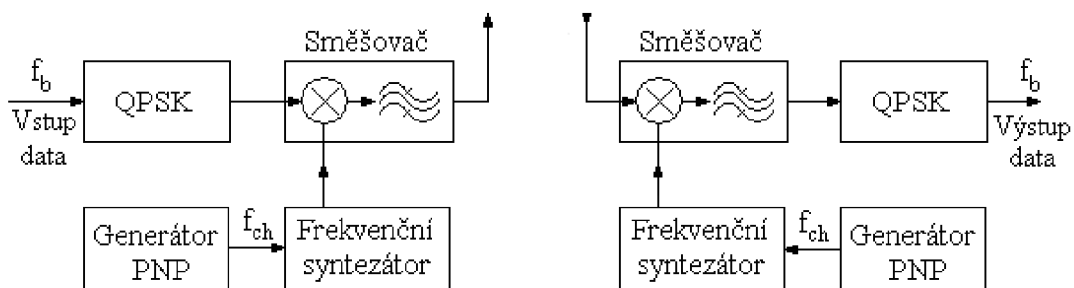
Obr.6: Schéma přijímače s rozprostřeným spektrem DS-SS s korelátorem ([1]).

Rozprostírání metodou kmitočtového skákání (FH-SS)

Frequency Hopping Spread Spectrum (FH-SS) je metoda frekvenčního skákání nosného kmitočtu. Data s bitovou rychlostí f_b jsou nejprve modulovány na nosnou vlnu (GFSK, QPSK...) a ta je ve směšovači násobena s další nosnou vlnou, u které se kmitočet mění v širokém pásmu. Tyto změny jsou dány pseudonáhodnou posloupností s přenosovou rychlostí f_{ch} . Podle poměru rychlosti, v jaké dochází ke kmitočtovému skákání a symbolové rychlosti dat původní informace f_s můžeme rozlišit FFH-SS (Fast Frequency Hopping-SS) - rychlé frekvenční skákání ($f_{ch} = k f_s$, kde k je celé kladné číslo) a SFH-SS (Slow Frequency Hopping-SS) - pomalé ($f_s = k f_{ch}$). Pro přenos může být k dispozici až několik stovek kmitočtů. Metoda je využívána jak v buňkových telefonech, tak v dalších komunikačních systémech.



Obr.7: Frekvenční skákání nosné vlny.



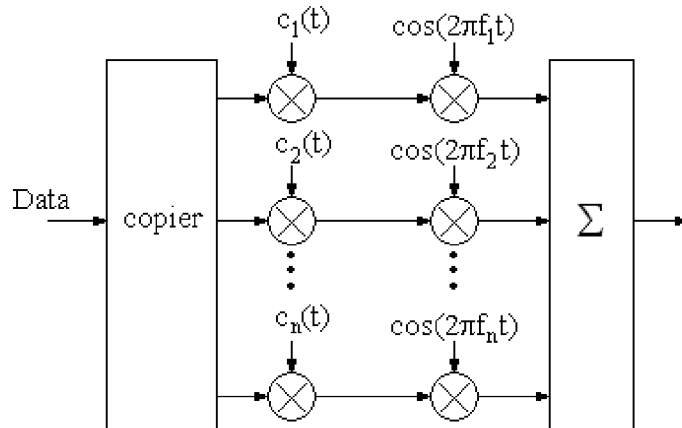
Obr.8: Vysílač a přijímač systému s frekvenčním skákáním nosné FH-SS podle [1].

Rozprostírání metodou časového skákání (TH-SS)

Obdobně k frekvenčnímu skákání existuje metoda časového skákání Time Hopping Spread Spectrum (TH-SS). Namísto kmitočtu nosné se zde však mění časové úseky, ve kterých jsou impulsy vysílány. Časy vysílání jsou určeny předem pseudonáhodnou posloupností. V mobilních komunikacích se metoda nevyužívá, spíše např. pro LAN, vojenskou komunikaci apod.

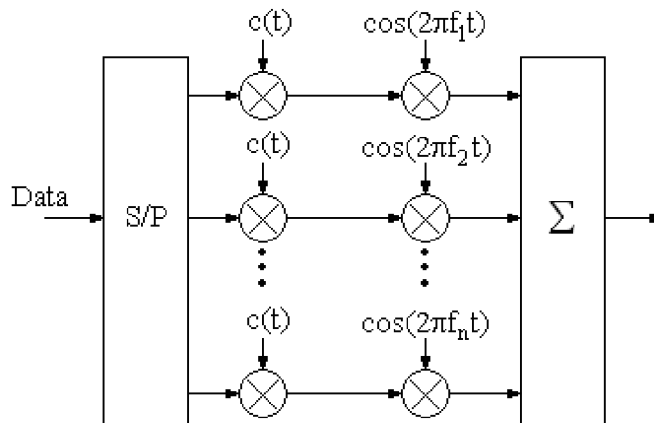
Mnohonásobný přístup MC-DS-CDMA, MC-CDMA

Zpracování více nosných vln se označuje MC-CDMA (Multi-carrier CDMA). Vysílač rozprostírá původní data přes různé subnosné, s použitím daného kódu. Signál je nejprve převeden v S/P převodníku, pak rozprostřen. Je potřeba vhodně zvolit počet subnosných a délku ochranného intervalu. Rozlišujeme dva typy MC-CDMA: první rozšíří datový tok z jednoho do n kanálů a moduluje jednotlivé kanály jednotlivými čipy $c_1(t), c_2(t), \dots, c_n(t)$ rozprostíracího kódu $c(t)$. Jedná se o tzv. rozprostření frekvenčně (Obr.9).



Obr.9: Vysílač MC-CDMA, rozprostření frekvenčně ([7]).

Druhý typ je MC-DS-CDMA: data jsou rozdělena v sériově paralelním převodníku a schéma jednotlivých větví odpovídá DS-CDMA. Každý bit datového toku je rozprostřen rozprostíracím kódem $c(t)$ daného uživatele na čipy a datový tok modulován na subnosnou. Rozprostření je časové. Spektrum subnosných je vzájemně ortogonální.



Obr.10: Vysílač MC-DS-CDMA ([7]).

Multiplex s Fourierovou transformací

Počet kanálů je obvykle příliš velký, může dosahovat až tisíců. To by vyžadovalo ve vysílači i přijímači velký počet modulátorů a demodulátorů. Proto se využívá metoda implementace za pomoci inverzní a přímé diskrétní Fourierovy transformace – DFT. Vzorkovaný digitální signál ve vysílači má diskrétní Fourierovo spektrum. Každá subnosná vlna odpovídá jednomu elementu diskrétního spektra. Soustava modulátorů je nahrazena procesorem inverzní diskrétní Fourierovy transformace IDFT (Inverse Discrete Fourier Transform). Ta je však výpočetně náročná, v praxi se tedy používá inverzní rychlá Fourierova transformace IFFT (Inverse Fast Fourier Transform).

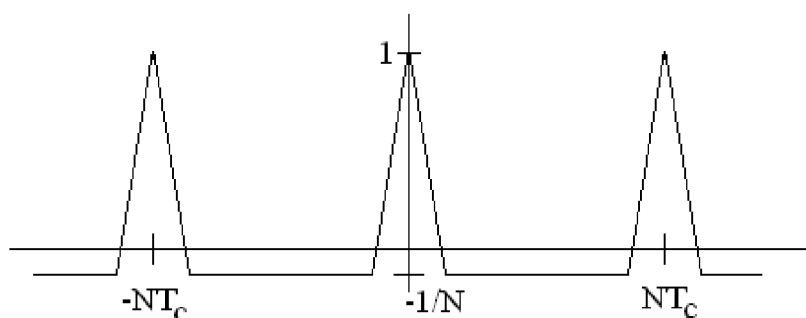
V přijímači se provede opačná operace – přímá rychlá Fourierova transformace FFT. Ve své podstatě se tedy frekvenční vzorky dat převedou na časové, kterými se moduluje výstupní nosná vlna, a přijímač je převede zase zpět.

Posloupnosti sloužící k rozprostření signálu

Použití různých posloupností je dáno jejich vlastnostmi. Ideální posloupnost by měla mít nulovou střední hodnotu a dobré autokorelační vlastnosti:

$$k=0: R(k) = \frac{1}{N} \sum_{n=1}^N c_n c_{n-k} = 1$$

$$k \neq 0: R(k) = 0.$$

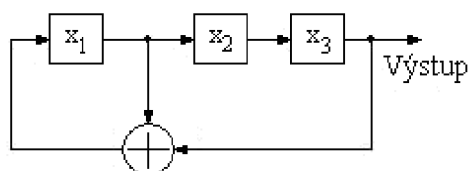


Obr.11: Autokorelace obecné posloupnosti.

Autokorelace slouží k synchronizaci na přijímací straně. V systému CDMA se využívají tzv. pseudonáhodné posloupnosti PN (Pseudo Noise, resp. Pseudo Random PR) a posloupnosti OVFSF (Orthogonal Variable Spreading Factor). Pseudonáhodné posloupnosti se vyznačují právě dobrou autokorelací. Jsou to posloupnosti s vlastnostmi náhodných sekvencí, ale náhodné nejsou. Je důležité, aby danou posloupnost použitou ve vysilači znal i přijímač, protože signál je obnoven pomocí stejné posloupnosti. PN posloupností se využívá u rozprostírání DS-SS. Mezi nejpoužívanější patří např. m-sequence, Gold a Kasami sekvence.

M-sequence

M-sequence jsou PN posloupnosti s maximální délkou periody (maximum-length sequences). Generují se pomocí posuvného registru LFSR (Linear Feedback Shift Register). Ten obsahuje m paměťových buněk, spojených určitými zpětnými vazbami, na jejichž výstupu je požadovaná posloupnost. Periody posloupnosti je $N = 2^m - 1$. Aby mohla být posloupnost generována, nesmí registr na začátku procesu obsahovat samé nuly.



Obr.12: Posuvný registr LFSR, $m = 3$.

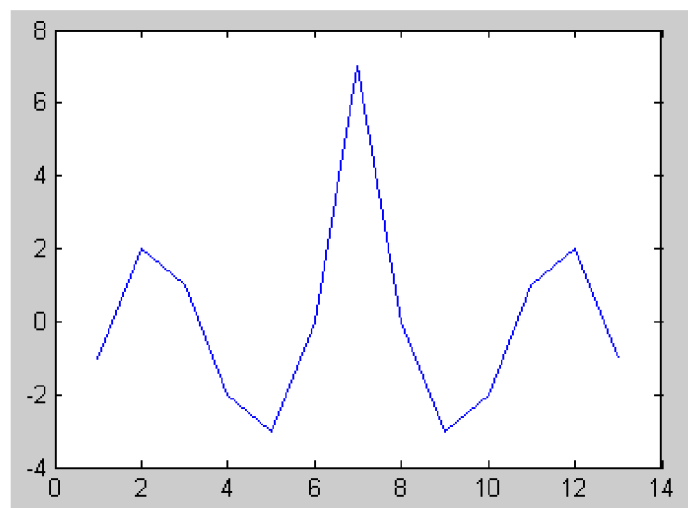
Nejjednodušší LFSR registr se skládá ze tří paměťových buněk. Vytvoření sekvence pomocí registru z Obr.12 bude následující:

010
 001
 100
 110
 111
 011
 101

Zpětná vazba je tvořena odbočkami 1 a 3. Perioda této posloupnosti je $N = 2^3 - 1 = 7$ a výstupní sekvence 1010011. Při výpočtu autokorelace se posloupnost binárních dat sečte se stejnou posloupností posunutou v čase a autokorelace je rovna rozdílu počtu shod a neshod, shody = 0, neshody = 1. Autokorelace je zobrazena na Obr..

$m=3$; $N = 7$

S =	1010011	S =	1010011	S =	1010011
S =	1010011	S(1) =	0100111	S(2) =	1001110
výsledná data =	0000000	výsledná data =	1110100	výsledná data =	0011101
autokorelace =	7	autokorelace =	3-4 = -1	autokorelace =	3-4 = -1

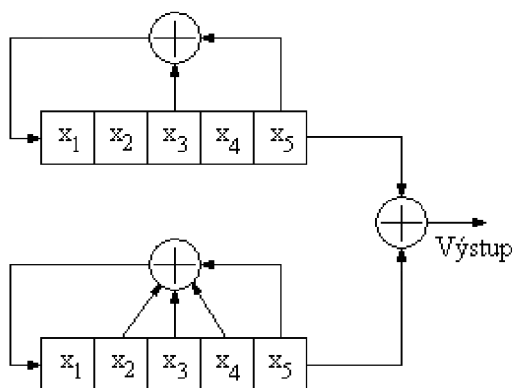


Obr. 13: Autokorelace vybrané posloupnosti, $m=3$.

Provedeme-li autokorelaci přes všechna posunutí, zjistíme, že maximální hodnoty dosahuje pro nulové posunutí a v ostatních případech je rovna -1. To je důležitá vlastnost m-sekvencí.

Gold a Kasami sekvence

Nevýhodou m-sekvencí je někdy příliš vysoká vzájemná korelace, při které nemusejí být jednotlivé kanály dostatečně rozlišený. Ideální vzájemná korelace je nulová, v praxi by měla být co nejmenší. Proto se v praxi používají modifikace m-sekvencí, sekvence Gold a Kasami, které se vytvoří např. ze dvou různých m-sekvencí jejich posouváním a sečtením, popsáným v [3]. Tyto sekvence vykazují dobré autokorelační vlastnosti a nižší vzájemnou korelaci než m-sekvence. Použity jsou v systému UMTS.



Obr.14: Příklad generátoru Goldovy sekvence ([3]).

Walshovy funkce a sekvence

Walshovy sekvence patří mezi posloupnosti OVVSF (Orthogonal Variable Spreading Factor), posloupnosti navzájem ortogonální. Na rozdíl od PN sekvencí nevykazují Walshovy sekvence dobré autokorelační vlastnosti, ale zato dobré vzájemné korelační vlastnosti. V ideálním prostředí vzájemně neinterferují a jsou tedy odolnější proti rušení. Walshovy sekvence se používají v systému UMTS s kombinací dalších PN posloupností sloužících k synchronizaci, činitel rozprostírání dosahuje hodnot okolo 10 až 1000.

Walshovy funkce se generují s použitím Hadamardových matic. Základem je matice hodnoty 1, další matice (H_{2n}) se vytvoří vždy z předchozí matice (H_n) podle dané rovnice:

$$H_1 = [1]$$

$$H_{2n} = \begin{bmatrix} H_n & H_n \\ H_n & H_n \end{bmatrix}.$$

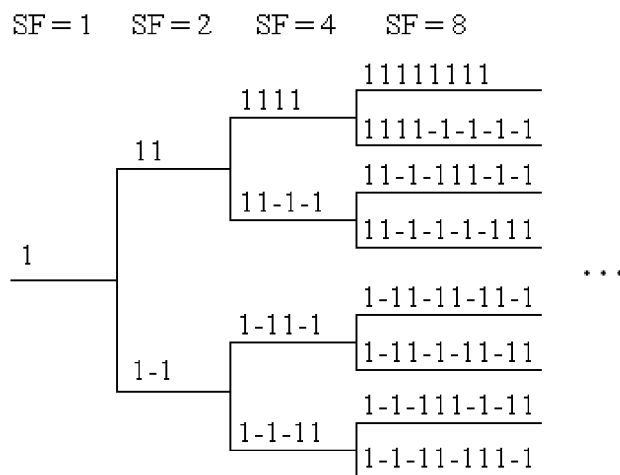
Vznikne soustava matic s ortogonálními řádky i sloupci:

$$H_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, H_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}, H_8 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix} \text{ atd.}$$

Pro Walshovy sekvence (posloupnosti 0 a 1) se tvoří matice obdobně, $H_1 = 0$:

$$H_2 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \text{ atd.}$$

Walshova funkce z matice osmého řádu bude 1 -1 1 -1 -1 1 -1 1, což odpovídá binární sekvenci 01011010. K vytvoření Walshových funkcí se využívají i různé jiné postupy, obecně jsou založeny na principu skládání symetrických funkcí. Dalším způsobem vyjádření prvků je strom (Obr.15), kde obdobně jako v matici symetricky přidáváme k posloupnosti jednu stejnou část a jednu převrácenou, tím dostáváme hodnoty dalšího SF.



Obr.15: Strom prvků Walshovy posloupnosti.

V UMTS slouží Walshova sekvence k rozproštění signálu, proces se označuje jako *channelization* – rozlišení kanálů jednoho uživatele. Poté je signál násoben tzv.scramblovacím kódem a tím je PN posloupnost, kterou jsou vzájemně rozlišeni jednotliví uživatelé:

Vysílač:
 Data 11100010
 PN sekvence 01011010
 Vyslaná data 10111000

Přijímač:
 Přijatá data 10111000
 PN sekvence 01011010
 Data 11100010

Energetická a spektrální účinnost

Energetická účinnost η_e (Energetic Efficiency) je definována jako poměr střední energie připadající na 1 bit E_b modulovaného signálu a výkonové spektrální hustoty šumu N_0 :

$$\eta_e = E_b / N_0 \text{ [-]},$$

stanovuje se pro určitou hodnotu chybovosti BER (nejčastěji $BER = 10^{-5}$) a vyjadřuje se obvykle v decibelech

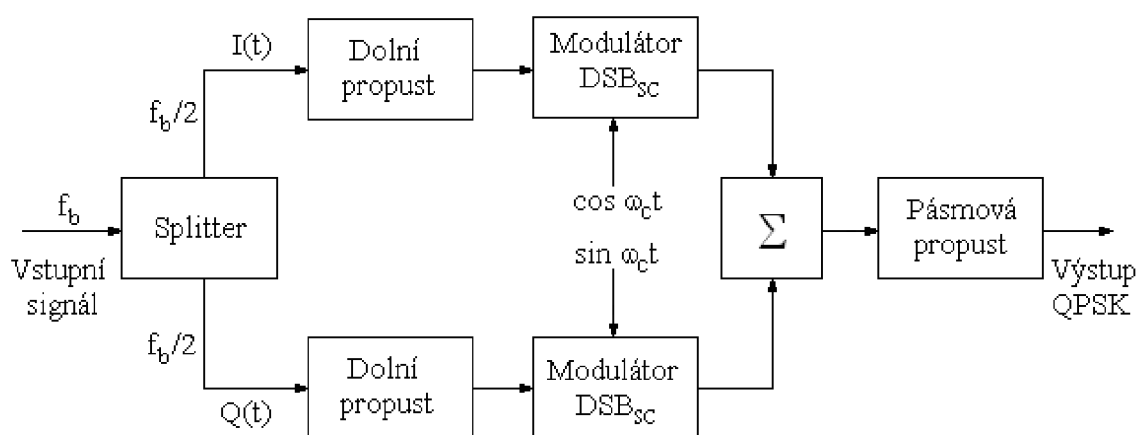
$$\eta_e = 10 \cdot \log (E_b / N_0) \text{ [dB]}.$$

Spektrální účinnost η_s (Spectral Efficiency) se vypočítá jako poměr přenosové rychlosti signálu a šířky pásma rádiového kanálu:

$$\eta_s = f_b / B_{vf} \text{ [bit/s/Hz]}.$$

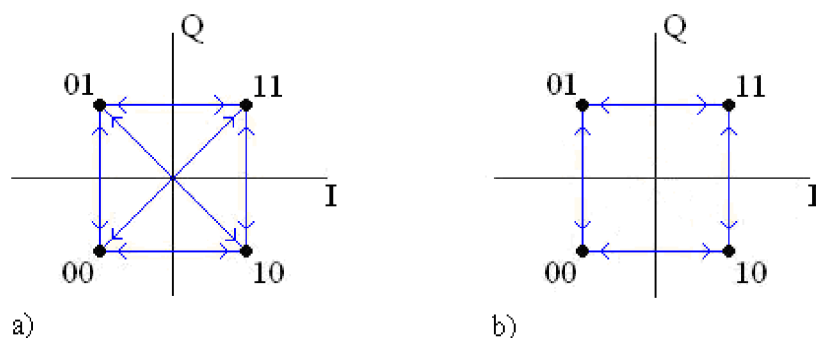
Čtyřstavové klíčování QPSK

Jedná se o modulaci čtyřstavové fázové klíčování 4PSK, neboli QPSK (Quadrature Phase Shift Keying, [6]), která používá k vyjádření symbolů čtyři možné stavy. Blokové schéma je znázorněno na Obr.16 [4]. Vlastnostmi tvoří QPSK kompromis mezi dobrou spektrální a energetickou účinností.



Obr.16: Blokové schéma modulátoru QPSK ([4]).

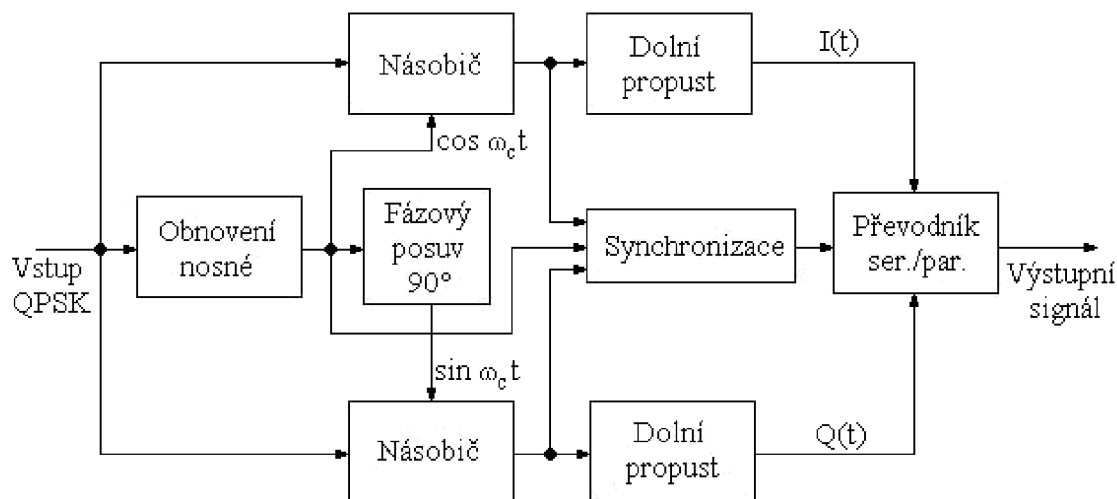
Vstupní signál je tvořen binárními daty (0,1), která mají přenosovou rychlost f_b . Datový tok je rozdělen v obvodu zvaném splitter na dva toky o poloviční přenosové rychlosti $f_b/2$ do větví I a Q a dále se zpracovávají samostatně. Vznikají tak dvojice bitů (dibity), tvoří konstelační diagram QPSK. Dibity jsou oproti vstupnímu signálu zpožděny o bitovou periodu T_b . Ve větvích jdou signály do dolní propusti a do modulátoru, kde jsou součinně modulovány na dvě nosné vlny vzájemně posunutě o 90° .



Obr.17: Vektorový diagram a) QPSK a b) O-QPSK.

Konstelační diagram znázorňuje všechny stavy bitů. Šipky označují možné změny mezi stavy a takový diagram se nazývá vektorový (Obr.17a). Mění-li se současně oba bity (šipky procházející

nulou), mění se amplituda signálu na nulovou hodnotu a poté se vrací s fází opačnou (posun o 180°). Vzniká největší parazitní amplitudová modulace. Ostatní změny stavů způsobí menší pokles amplitudy, změnu fáze jen o 90° a menší parazitní modulaci. Proto se využívá kromě QPSK dalších modulací (O-QPSK, Obr.17b), jejichž změny ve vektorovém diagramu nulou neprochází.

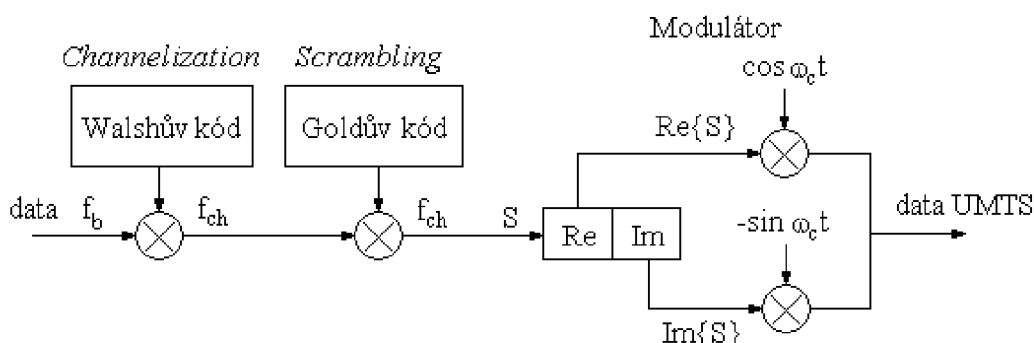


Obr.18: Blokové schéma demodulátoru QPSK ([4]).

Demodulátor QPSK je na Obr.18 [4]. Signál je rozdělen do dvou totožných složek a přiveden do dvou násobičů, kde je pomocí synfázní a kvadraturní nosné zpět demodulován. Oba signály jsou filtrovány dolní propustí a složky I a Q poté sloučeny v paralelně-sériovém převodníku. Synfázní a kvadraturní nosná vlna musí být koherentní s odpovídajícími nosnými vlnami modulátoru QPSK ve vysílači. Modulace QPSK se používá např. u terestriálního digitálního vysílání.

UMTS

System UMTS (Universal Mobile Telecommunications System) využívá Walshovy kódy a Goldovy kódy. Obecné schéma pro uplink systému UMTS je na Obr.19.

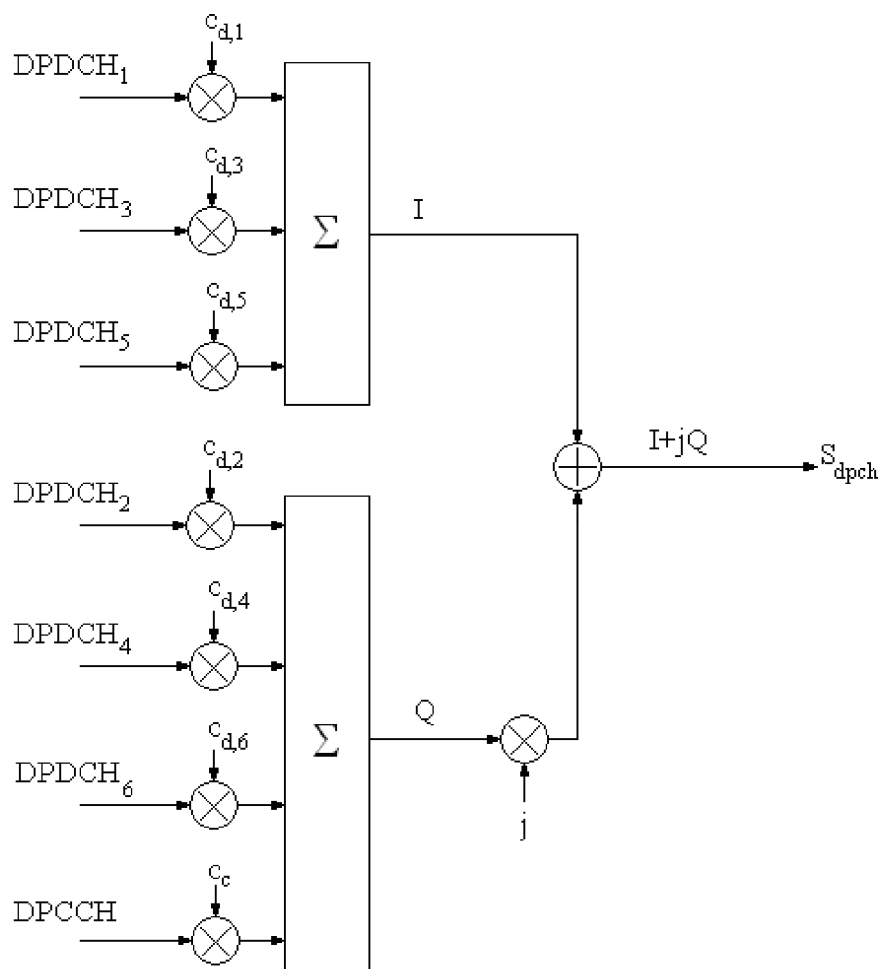


Obr.19: Rozprostírání v systému UMTS (uplink).

Walshův kód mění bitovou rychlost na chipovou s daným rozprostíracím faktorem. Gold kód slouží k zabezpečení přenosu a čipová rychlost se dál nemění.

Rozprostírání kanálů pro uplink (DPCCH/DPDCH/HS-DPCCH)

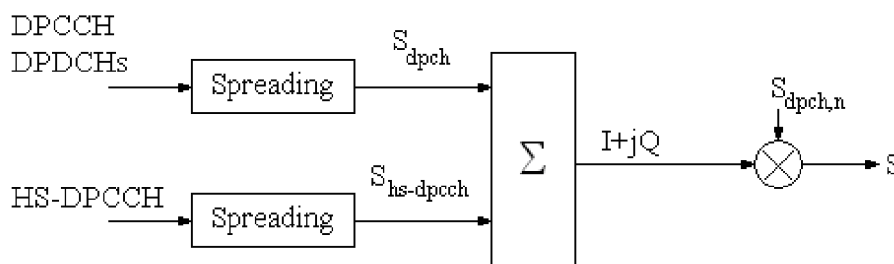
Rozprostření kanálů pro uplink znázorňuje Obr.20. DPCCH (Dedicated physical control channel) je kontrolní kanál rozprostíraný kódem c_c , DPDCH_n jsou jednotlivé datové kanály (data channel), rozprostřeny kódem $c_{d,n}$. Poté jsou hodnoty ještě váhovány. Nakonec jsou signály z větve I a Q sečteny a je z nich vytvořen komplexní tok chipů. Kontrolní kanály bývají ve větvi Q.



Obr.20: Rozprostření datových kanálů a kontrolního kanálu pro uplink UMTS ([7]).

Dalším typem kanálu je HS-DPCCH (high speed DPCCH), vysokorychlostní DPCCH. Rozprostírá se kódem c_{hs} . Komplexní tok je scamblován komplexním kódem $S_{dpch,n}$ (Obr.21). Při modulaci pro uplink je komplexní datový tok získán rozprostřením rozdělen na imaginární a reálnou složku. Pro obě složky je konfigurována jedna frekvence. Reálná část se moduluje na nosnou vlnu $\cos(\omega t)$, imaginární na $-\sin(\omega t)$ (Obr.19).

V downlinku probíhá rozprostírání obdobně, ale trochu složitěji, systém pracuje s komplexními kódy a pouze jedním komplexním datovým tokem. Větev Q je násobena imaginární jednotkou j a sečtena s odpovídajícími prvky větve I, výsledkem je jedna komplexní sekvence, která se dále scrambluje, viz [7].



Obr.21: Rozprostření kanálů pro uplink UMTS ([7]).

Krátká scramblovací sekvence pro uplink

Krátké sekvence $c_{short,1,n}(i)$ a $c_{short,2,n}(i)$ podle [7] vznikají mapováním sekvence $z_n(i)$, $0 \leq n \leq 16777215$, která je součtem modulo 4 tří generovaných posloupností, binární posloupnosti $b(i)$ a $d(i)$ a ternární posloupnosti $a(i)$. sekvence $z_n(i)$ má délku 255 a je popsána vztahem:

$$z_n(i) = a(i) + 2b(i) + 2d(i) \text{ modulo } 4, i = 0, 1, \dots, 254.$$

Generátor sekvence $a(i)$ představuje polynom $g_0(x) = x^8 + 3x^5 + x^3 + 3x^2 + 2x + 3$:

$$a(i) = 3a(i-3) + a(i-5) + 3a(i-6) + 2a(i-7) + 3a(i-8) \text{ modulo } 4, i = 8, 9, \dots, 254;$$

binární sekvence $b(i)$ je dána polynomem $g_1(x) = x^8 + x^7 + x^5 + x + 1$:

$$b(i) = b(i-1) + b(i-3) + b(i-7) + b(i-8) \text{ modulo } 2, i = 8, 9, \dots, 254$$

a binární sekvence $d(i)$ je určena polynomem $g_2(x) = x^8 + x^7 + x^5 + x^4 + 1$:

$$d(i) = d(i-1) + d(i-3) + d(i-4) + d(i-8) \text{ modulo } 2, i = 8, 9, \dots, 254.$$

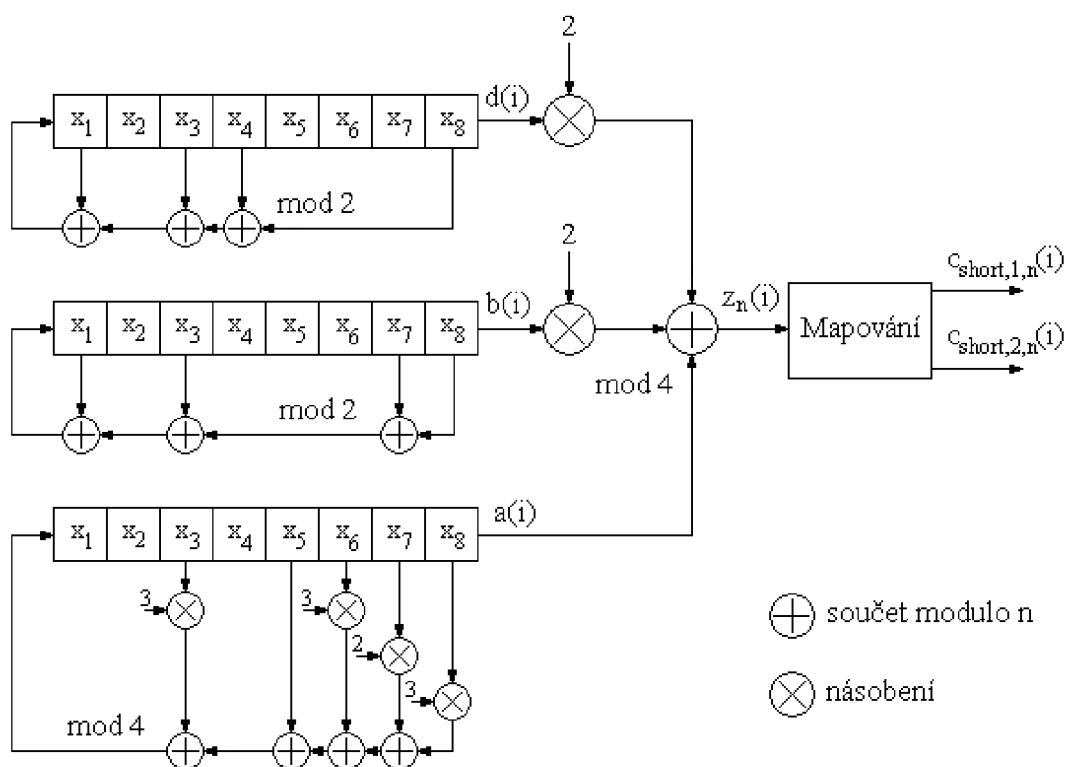
Sekvence $z_n(i)$ je rozšířena na 256 čipů stanovením $z_n(255) = z_n(0)$. Mapování probíhá podle Tab.2. Výsledná komplexní scramblovací sekvence $C_{short,n}$ je definována takto:

$$C_{short,n}(i) = c_{short,1,n}(i \bmod 256) \left(1 + j(-1)^i c_{short,2,n} \left(2 \lfloor (i \bmod 256) / 2 \rfloor \right) \right),$$

kde $i = 0, 1, 2 \dots$ a $\lfloor \cdot \rfloor$ označuje zaokrouhlení na nejbližší nižší celé číslo.

$z_n(i)$	$c_{short,1,n}(i)$	$c_{short,2,n}(i)$
0	+1	+1
1	-1	+1
2	-1	-1
3	+1	-1

Tab.2: Mapování sekvence $z_n(i)$ dle [7].



Obr.22: Generátor krátké scamblovací sekvence pro uplink UMTS ([7]).

Dlouhá scamblovací sekvence pro uplink

Sekvence $c_{long,1,n}$ a $c_{long,2,n}$ pro UMTS vznikají použitím segmentů dvou binárních m-sekvencí, dlouhých 38400 čipů, sečtených funkcí modulo 2. Polynomické generátory m-sekvencí jsou stupně 25. Označíme-li první m-sekvenci x a druhou y , je x vytvořena polynomem $X^{25}+X^3+1$ a y polynomem $X^{25}+X^3+X^2+X+1$. Sekvence $c_{long,2,n}$ představuje 16777232 čipovou verzi sekvence $c_{long,1,n}$. Obě sekvence jsou reálné a skládají se do komplexní scamblovací sekvence $C_{long,n}$.

Pro vytvoření m-sekvencí platí počáteční podmínky:

$$x_n(0)=n_0, x_n(1)=n_1, \dots, x_n(22)=n_{22}, x_n(23)=n_{23}, x_n(24)=1, y(0)=y(1)=\dots=y(24)=1.$$

$$x_n(i+25) = x_n(i+3)+x_n(i) \pmod{2}, i=0, \dots, 2^{25}-27.$$

$$y(i+25) = y(i+3)+y(i+2)+y(i+1)+y(i) \pmod{2}, i=0, \dots, 2^{25}-27.$$

Binární Gold sekvence je definovaná jako:

$$z_n(i) = x_n(i) + y(i) \pmod{2}, \quad i = 0, 1, 2, \dots, 2^{25} - 2.$$

Funkční hodnoty odpovídají reálné Gold sekvenci:

$$Z_n(i) = \begin{cases} +1 & \text{if } z_n(i) = 0 \\ -1 & \text{if } z_n(i) = 1 \end{cases} \quad i = 0, 1, 2, \dots, 2^{25} - 2.$$

Dlouhé scamblovací sekvence $c_{long,1,n}$ a $c_{long,2,n}$ se definují následně:

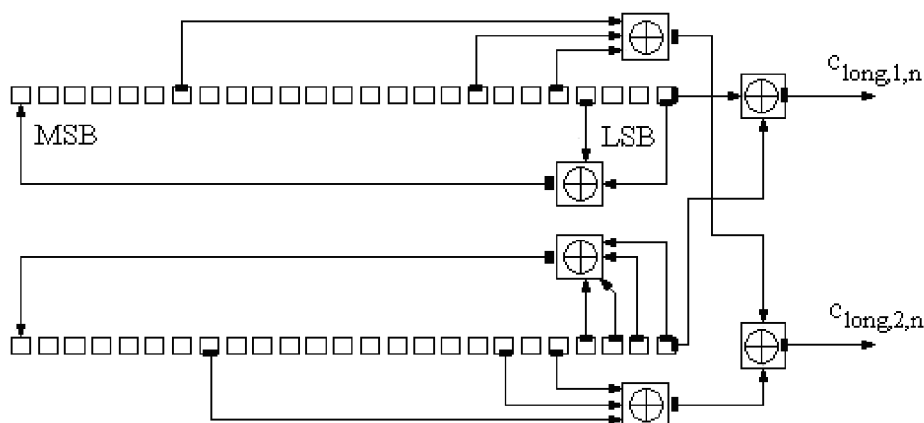
$$c_{long,1,n}(i) = Z_n(i), \quad i = 0, 1, 2, \dots, 2^{25} - 2 \text{ a}$$

$$c_{long,2,n}(i) = Z_n((i + 16777232) \pmod{2^{25} - 1}), \quad i = 0, 1, 2, \dots, 2^{25} - 2.$$

Výsledkem je komplexní dlouhá scamblovací sekvence $C_{long,n}$:

$$C_{long,n}(i) = c_{long,1,n}(i) \left(1 + j(-1)^i c_{long,2,n}(2 \lfloor i/2 \rfloor) \right), \quad i = 0, 1, 2, \dots, 2^{25} - 2,$$

kde $\lfloor \cdot \rfloor$ označuje zaokrouhlení na nejbližší nižší celé číslo.



Obr.23: Generátor dlouhé scamblovací sekvence pro uplink UMTS ([7]).

Simulace rozptřeni signálu v Matlabu

Vstupními signály b (b_1, b_2, \dots) jsou náhodně generované posloupnosti hodnot 1 a -1 délky N_b . Náhodné generování provádí funkce `randsrc`, která vytvoří pole požadované délky (N_b), složené z vybraných prvků (1 a -1), generované se stejnou pravděpodobností:

```
b1=randsrc(1,Nb,[1 -1])
```

Dalším vstupním údajem je rozptřiací kód c (c_1, c_2, \dots) délky N . Každý bit je rozptřen na N čipů. Walshovy funkce získáme z Hadamardovy matice. N -rozměrnou matici vygenerujeme funkcí `hadamard(N)`. Pseudonáhodné posloupnosti jsou tvořeny LFSR registry. Cyklicky se opakující průběh posunu bitů v registru představuje cyklus `for`, počet provedení cyklu odpovídá právě délce výstupního kódu. Na vstup registru se během posuvu uloží hodnota určená zpětnými vazbami.

Výsledkem rozptření je vektor v , jehož prvky jsou vytvořeny násobením jednotlivých prvků vstupního signálu a daného kódu c . Prvky jsou postupně přidávány do vektoru podle daného toku dat. Vektor v je N -krát delší než signály b :

```
v1=[];
for i=1:Nb
    v1=[v1 c1*b1(i)];
end
```


Průchod komunikačním kanálem představuje součet signálů od jednotlivých uživatelů $o1=v1+v2$. Při demodulaci se tento datový tok rozdělí na úseky, jejichž délka odpovídá délce kódu N . K tomu slouží funkce `reshape`(původní matice, počet řádků, počet sloupců):

```
M=reshape(o1,N,length(v1)/N)'
```

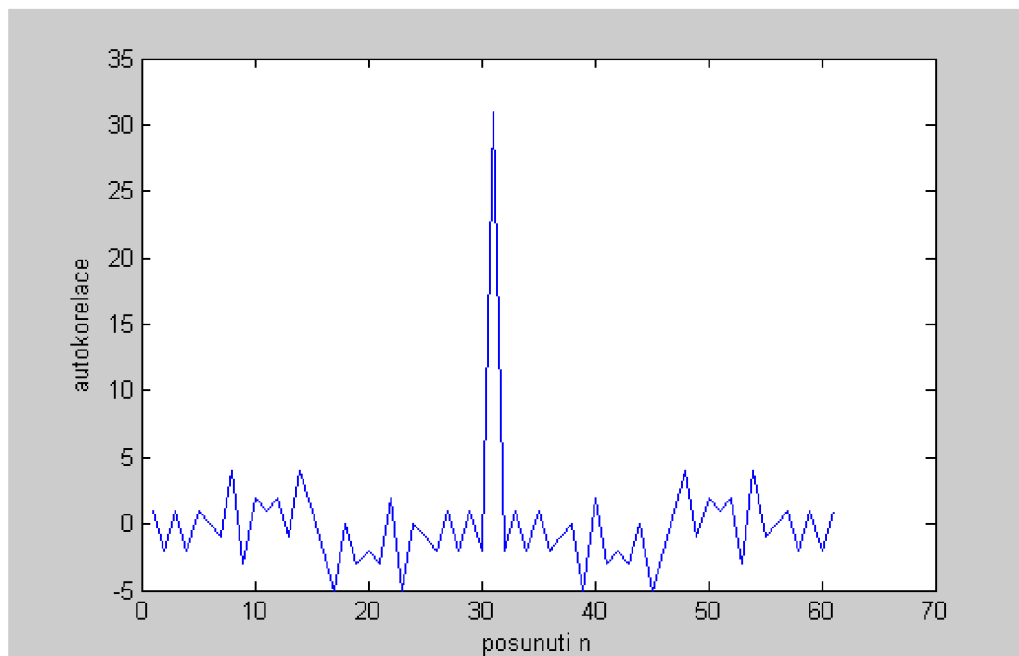
Každý úsek je poté znovu násoben kódem daného uživatele. Jeho výsledné prvky jsou sečteny a funkce `sign` rozhodne, zda byla přijata hodnota 1 nebo -1. Pokud se hodnota neshoduje s hodnotou v původním signálu, počet chyb se zvětší o jedničku. Uvedená simulace odpovídá rozptřeni DS-SS.

```
for i=1:Nb
    d1(i)=sign(sum(M(i,:).*c1));
end
```

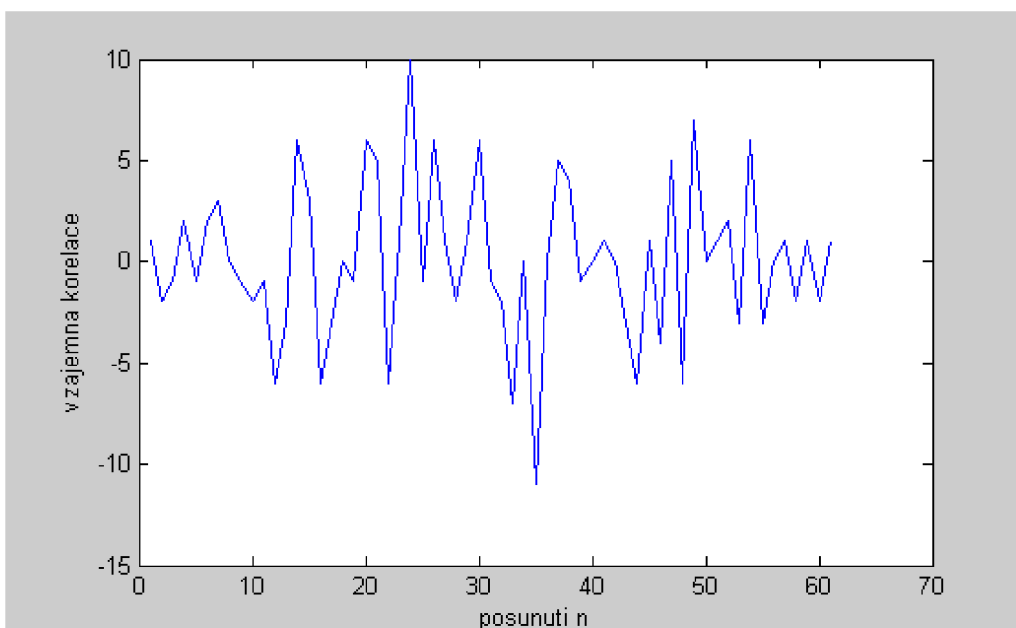
K výpočtu korelačních vlastností vybraných posloupností slouží funkce `xcorr`(název sekvence), funkce `plot` zobrazí výslednou křivku do grafu. Zobrazit lze autokorelační křivku nebo také vzájemná korelační funkce dvou různých posloupností:

```
plot(xcorr(msekvence))
plot(xcorr(msekvence1,msekvence2))
```

Analýzu je třeba provádět z funkčních hodnot odpovídajících prvků. Z grafu na Obr.24 a 25 jsou patrné dobré autokorelační vlastnosti m -sekvence a vzájemná korelace dvou různých zvolených m -sekvencí. Obdobné průběhy vykazují i Gold sekvence.

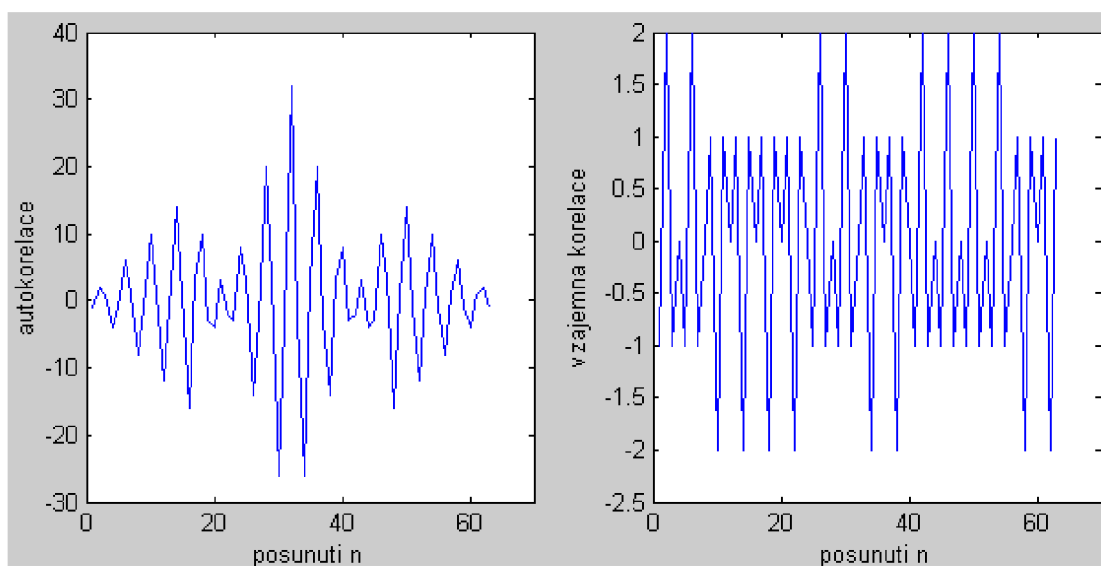


Obr.24: Autokorelační funkce zvolené m -sekvence.



Obr.25: Vzájemná korelace dvou různých m-sekvencí.

U Walshových sekvencí není autokorelační maximum tak výrazné a může být nesprávně určena jako maximální úroveň některého z předchozích vrcholků (Obr.26 vlevo). Stačí k tomu, aby překročily nebo dosáhly rozhodovací úrovně, která je stanovena. Naopak vzájemné korelační vlastnosti dvou různých Walshových sekvencí jsou velmi výhodné. Symetrické vlastnosti původních funkcí, ze kterých jsou složeny, zaručují, že se neobjevuje žádné ostré maximum a funkce jsou plně ortogonální, což je zřetelné na Obr.26 vpravo.



Obr.26: Autokorelace zvolené Walshovy funkce a vzájemná korelace dvou různých Walshových funkcí.

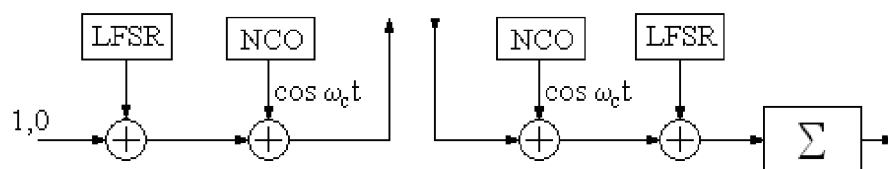
Implementace modulátoru a demodulátoru do obvodu FPGA

Obvody FPGA

Obvod FPGA (Field-programmable Gate Array) je integrovaný obvod, který se skládá z programovatelných logických obvodů. Obsahuje velké množství hradel (současně až 6 milionů dle [13]), která mohou představovat logické funkce AND, NAND, XOR apod. a obsahují také paměťové jednotky. Části obvodu se dělí na bloky IOB (Input/Output Block) a LB (Logic Block). LB jsou vlastní programovatelné logické bloky. Ke konfiguraci se používá jazyk HDL (Hardware Description Language). Softwarovým prostředkem, ve kterém mohou být implementovány struktury do FPGA, je prostředí Xilinx ISE. V něm jsou k dispozici dva programovací jazyky: VHDL a Verilog.

Simulace v prostředí Xilinx ISE (verze 8.2)

Model modulátoru s rozptřením DS-SS pro simulaci je zobrazen na Obr.27. Jednotlivé funkční bloky lze vložit pomocí komponent z IP Core Generatoru. Vlastnosti komponent jsou specifikovány v [9], [10] a [11]. Po vytvoření nového projektu je nejprve třeba vygenerovat nový zdroj (New Source) a tím je VHDL Module. Je to v podstatě soubor s příponou .vhd. Do tohoto zdroje se vkládají jednotlivé části komponenty. Komponentu lze vytvořit opět jako New Source a dále je třeba zvolit IP (Coregen & Architecture Wizard). Následuje výběr komponent, který je v dané verzi prostředí k dispozici. Parametry každé komponenty se nastavují v zobrazeném okně. Po jejím vygenerování se zobrazí potřebný programový kód v záložce Language Templates.



Obr.27: Blokové znázornění modulátoru a demodulátoru s rozptřením.

Jazyk VHDL

Základní struktura programu ve VHDL se skládá alespoň ze dvou samostatně překládaných jednotek: entity a architecture. Entita specifikuje vnější vstupní a výstupní porty modelu. Tyto porty jsou jediným prostředkem pro komunikaci modelu s jeho okolím. Architecture definuje chování modelu.

Funkční architektura je taková, která nemá vnořené další entity a architektury. Vstupně-výstupní závislosti jsou definovány paralelními příkazy, které přiřazují hodnoty signálům. Při změně hodnoty některého signálu dochází k tzv. události, tj. aktivaci všech paralelních příkazů, které jsou na daný signál citlivé. Strukturní architektura má vnořené další entity a architektury. Porty vnořených entit mohou být připojeny na porty příslušné entity nebo na lokální signály této architektury. Pokud architektura neobsahuje žádné lokální deklarace, její příkazy operují pouze nad porty či signály příslušné entity.

VHDL není objektově orientovaný jazyk, termín objekt v něm označuje prostředek pro reprezentaci nějakého atributu modelu. Rozlišovány jsou zde třídy objektů: signal, variable a constant. Signály jsou objekty třídy signal, proměnné jsou objekty třídy variable. Z hlediska směru přenosu informace existují pro jednotlivé prvky (parametry, porty) módy in, out, inout a buffer. Porty označují speciální signály, u kterých se dovolené operace v příslušné architektuře řídí jejich módy. Vstupní porty (mód in) mohou být pouze čteny, výstupní porty (mód out) je možné pouze budit, a to v strukturách. Uvnitř entity nelze porty budit. Porty v módu buffer jsou výstupní porty, které mohou být zároveň také čteny (vrací jim poslední zapsanou hodnotu).

Deklarace entity a architektury podle [8]:

```
entity <identifikátor entity> is
  [ generic.....; ] -- parametry entity
  [ port.....; ] -- vstupy a výstupy
  [ --přípustné deklarace:
    datové typy a podtypy,
    konstanty,
    paralelní podprogramy,
    signály,
    soubory,
    klausule alias a use. ]
  [ begin
    -- operační část entity, paralelní prostředí, přípustné jsou paralelní příkazy:
      příkaz assert
      příkaz pasivního procesu,
      příkaz pasivní procedury ]
end [ <identifikátor entity> ] ;

architecture <identifikátor architektury> of <identifikátor entity> is
--deklarační část
  [ datové typy a podtypy,
    konstanty,
    paralelní podprogramy,
    signály,
    lokální komponenty,
    soubory
    klausule alias a use ]
begin
--operační část architektury reprezentující paralelní prostředí:
  signálový paralelní příkaz,
  příkaz procesu,
  příkaz assert,
  příkaz paralelní procedury,
  příkaz generate,
  příkaz bloku,
  příkaz instalace komponenty.
end [ <identifikátor architektury> ];
```

Použití knihoven

Předem nastaveny bývají obvykle standardní knihovny IEEE, IEEE.STD_LOGIC_1164.ALL, work a UNISIM. Pro simulaci je důležitá knihovna UNISIM, k práci s komponentami slouží knihovna UNISIM.vcomponents.all. Použití aritmetických operací vyžaduje knihovnu IEEE.STD_LOGIC_ARITH.ALL. Při simulaci pomocí VHDL Test Bench se objeví také knihovna IEEE.NUMERIC_STD.ALL. Podle požadovaných hodnot binárních signálů se zvolí jedna z knihoven IEEE.STD_LOGIC_UNSIGNED.ALL nebo IEEE.STD_LOGIC_SIGNED.ALL.

Deklarace signálů

Důležitou částí programu je samotné stanovení vstupů, výstupů a vnitřních signálů. Mezi vstupní signály bude náležet hodinový signál (*Clk*) a datový signál (*a*) - libovolná sekvence jedniček a nul. Další vstupy jsou volitelné. Doplnit lze např. vstup do posuvného registru LFSR (signál *b*) a tím nastavit jeho počáteční stav. Výstupní signál (*Cout*) představuje výsledek celé modulace.

Na výstupu LFSR se objeví první vnitřní signál (*s1*). Spolu s datovým signálem vstupuje do sčítačky. Výstupem sčítačky je další vnitřní signál (*s2*) a dále je zpracován na vnitřní signálový vektor (*s3*). Rovněž výstup generátoru harmonického signálu (*cos*) je vnitřní signál, protože je dále použit spolu se signálem *s3* jako vstup do násobičky.

V případě demodulátoru by byl postup analogický. Z výstupního signálu *Cout* by se stal vstupní signál, demodulace proběhne stejným signálem *cos* a derozprostřednění signálem *s1*. Výsledný datový tok je třeba obdobně jako v předchozích kapitolách nakonec ještě rozčlenit na periody. Je-li suma hodnot jedné periody větší jak 0,5, byl přijat bit ‚1‘, je-li menší jak 0,5, přijatý bit je ‚0‘. Porovnáním výstupu demodulátoru a původního signálu *a* dostaneme chybovost BER.

Součet signálů

Součet dvou signálů je možné provést pouhým sečtením signálů, deklarací nového procesu v těle architektury. V Core Generatoru jde o komponentu Adder/Subtractor v7.0 podle [9]. Součástka dovede sčítat i odečítat a obsahuje další doplňující signály.

Základními parametry komponenty uvedené v [9] jsou šířky vstupních portů A a B, které mohou nabývat rozsahu od 1 do 256. Hodnoty dat mají tvar znaménkový (Signed) nebo bez použití znaménkového bitu (Unsigned), přednastavena je hodnota Unsigned. Na port B je možné přivádět konstantní hodnotu: zatržením položky „Constant Value“ a zadáním hodnoty v hexadecimálním kódu do textového pole.

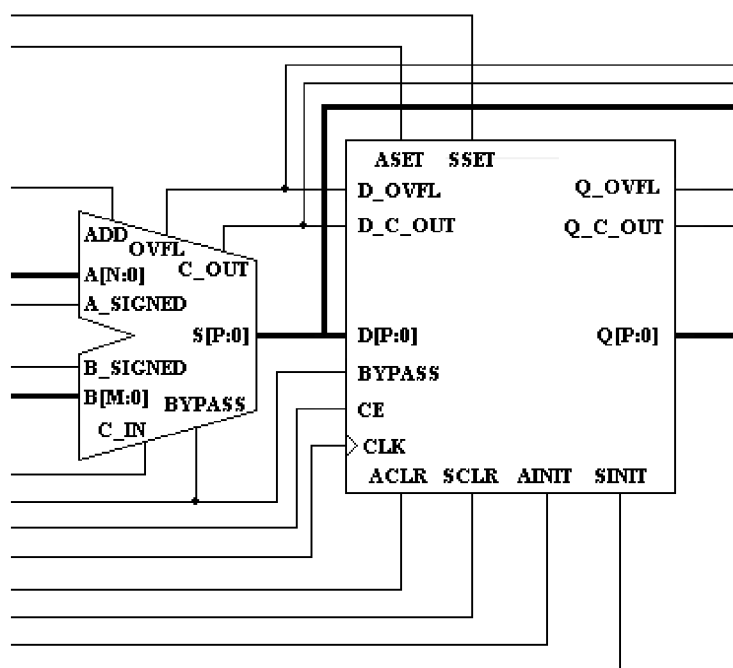
Při součtových operacích může dojít k přetečení, neboli překročení rozsahu původních hodnot sčítanců. Bit, který je vyššího řádu než rozsah daných hodnot, se přenáší a je označen Carry. Při odečítání obdobně bit, který si „půjčí“ menšitel pro provedení operace, aby výsledek odečtu nebyl záporný, se označuje Borrow. Z toho plyne i použití pinů.

Vstup Carry/Borrow: pin C_IN je generovaný pro volbu sčítání a sčítání/odečítání, B_IN pro odečítání. Výstup Carry/Borrow: platí pouze pro hodnoty Unsigned. Pro součet nebo součet/odečet se používá pin C_OUT. Pro pouhý odečet slouží pin B_OUT. Přítomnost přetečení signalizuje výstup OVFL. Platí pouze pro znaménkové hodnoty vstupů. V původním nastavení není generován.

BYPASS pin dovoluje hodnotám na portu B s následující aktivní hodinovou hranou přejít logikou do výstupního registru. CE Override for Bypass kontroluje, zda je vstup BYPASS omezen signálem CE. Není-li tato možnost zvolena, registr potřebuje k nahrávání dat portu B aktivní signál CE. Parametr CE Overrides kontroluje, zda jsou vstupy SSET, SCLR a SINIT stanoveny signálem CE, volba je dostupná pouze za přítomnosti požadovaného vstupu CE. Signál CE je vygenerován po zaškrtnutí položky Clock Enable.

Položka Asynchronous Settings slouží k nastavení asynchronních vstupů. Při ponechání volby „None“ není zapojen žádný ze vstupů ASET, ACLR, nebo AINIT. Obdobně lze nastavovat synchronní vstupy SSET, SCLR a SINIT (v položce Synchronous Settings). V původním (default) nastavení jsou opět vypnuty.

Při realizaci součtu dvou binárních signálů pomocí operace XOR (modulo 2) je potřeba ponechat tvar čísla jako Unsigned, protože vstupní datový signál a signál, se kterým je sečten, je považován v podstatě za jednobitový logický vektor. Součet provede sčítačka bit po bitu a nedochází k přetečení (zbytek se zahazuje a další bit se počítá zvlášť, na rozdíl od vícebitových vektorů). Vstupy Carry/Borrow či výstup OVFL tak nejsou využity.



Obr.28: Schématický symbol komponenty Adder/Subtractor [9].

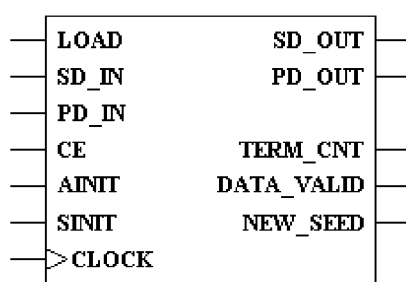
Posuvný registr LFSR

Linear Feedback Shift Register (v3.0) podle [10] je univerzální posuvný registr řízený hodinovým signálem Clock, s nastavitelnou délkou a volitelným zapojením vstupů. Zadáání všech parametrů probíhá v okně zobrazeném po otevření součástky. Propojení vodičů je třeba uvést ve tvaru polynomu odpovídajícímu jednotlivým výstupům LFSR.

Zapojení vstupů je možné provést dvěma metodami ([10]). Metoda Fibonacci znamená, že výstupy z vybraných registrů jsou vzájemně sečteny pomocí modulo 2 (XOR) a tento součet zaveden přímo na vstup posuvného registru. Přednastavené hodnoty polynomů pro konkrétní délky registrů a délku výsledné posloupnosti uvádí tabulky v [10]. Maximální délka LFSR dosahuje až 168 registrů. Metodou Galois lze dosáhnout stejné výstupní sekvence jako v předchozím případě už po 1 cyklu. Jednotlivé brány jsou umístěny mezi registry.

Nastavovanými parametry jsou LFSR size, což je velikost posuvného registru, pro polynom 5. řádu je 5. Způsob součtu (Feedback Gate Type) XOR nebo XNOR. Pro Fibonacciho strukturu použijeme součet XOR. Polynom $X^5+X^2+X^0$ určuje sečtení vazeb z 1., 3. a 5. registru, kde 1. je označuje LSB a 5. registr MSB.

Vstupními piny posuvného registru jsou: LOAD, SD_IN, PD_IN, CE, AINIT, SINIT a CLOCK. Výstupní piny: SD_OUT, PD_OUT, TERM_CNT, DATA_VALID a NEW_SEED.



Obr.29: Schématický symbol LFSR [10].

Do posuvného registru lze zadávat data sériově (pomocí signálu SD_IN) nebo paralelně (signálem PD_IN). Registr je možné naplnit libovolně jedničkami a nulami, kromě kombinace samých nul v celém LFSR (při použití logických operátorů XOR). Z tohoto stavu se do jiného stavu registr sám od sebe nedostane. LFSR je však na počátku již naplněn samými jedničkami, a proto pracuje správně, i když není zapojen ani jediný datový vstup.

Při použití sériového vstupu SD_IN dochází k plnění po jednom bitu do vstupní buňky LFSR s každým hodinovým cyklem během doby, kdy je aktivní signál LOAD. Vzestupná hrana signálu Load zahájí plnění vstupního registru, sestupná ukončí. Obdobně probíhá plnění signálem PD_IN, zde však díky paralelnímu plnění lze posuvný registr naplnit celý najednou již po jednom hodinovém cyklu. Šířka portu PD_IN[N-1:0] odpovídá právě délce LFSR.

Výstup LFSR je obdobně jako vstup sériový i paralelní. Data lze číst postupně pomocí signálu SD_OUT, nebo zobrazit stavy všech registrů najednou díky paralelnímu portu PD_OUT. Signály DATA_VALID, NEW_SEED a TERM_CNT nesou doplňující informace.

Vstupní signál CE mění časový průběh generování. Není-li zapojen, nebo $CE = 1$, posuvný registr obnovuje svůj stav při každé vzestupné hraně hodinového signálu. Při $CE = 0$ se stav

nemění až do uplatnění asynchronního resetu. K synchronnímu a asynchronnímu resetu LFSR slouží signál SINIT a AINIT [10].

Generátor nosné vlny

K modulaci datového signálu na nosnou kosinusovou vlnu s průběhem $\cos \omega_c t$ je nejprve potřeba vložit do projektu generátor harmonického signálu, nebo-li napětím řízený oscilátor. Ten se v IP Core skrývá pod označením DDS - Direct digital synthesizer v5.0 (v Xilinx ISE 8.2) podrobně popsany v [11], jde o generátor sinusových a kosinusových vln a jejich kombinací.

DDS jsou obecně důležitými komponentami v mnoha komunikačních systémech. Kvadraturní syntezátory se používají při konstrukci digitálních konvertorů, demodulátorů a při využití různých typů modulačních schémat, např. PSK (phase shift keying), FSK (frequency shift keying) a MSK (minimum shift keying).

Výstupní frekvence f_{out} podle [11] je funkcí frekvence hodinového signálu systému f_{clk} , počtu bitů $B_{\Theta(n)}$ ve fázovém akumulátoru a fázového inkrementu $\Delta\theta$:

$$f_{out} = \frac{f_{clk} \Delta\theta}{2^{B_{\Theta(n)}}} \text{ Hz}.$$

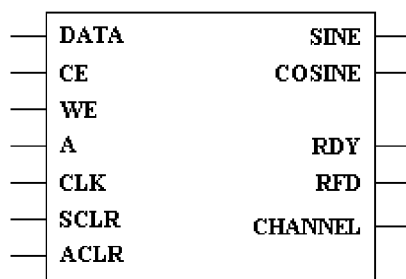
Při zadané výstupní frekvenci se dle tohoto vztahu určuje fázový inkrement $\Delta\theta$.

Rozlišení frekvence Δf syntezátoru je funkcí frekvence hodinového signálu a počtu bitů $B_{\Theta(n)}$ fázového akumulátoru [11]:

$$\Delta f = \frac{f_{clk}}{2^{B_{\Theta(n)}}}.$$

Schématický symbol DDS je na Obr.30. Známořňuje vstupní signály: CLK, A, WE, CE, DATA, ACLR, SCLR a výstupní signály: RDY, RFD, CHANNEL, SINE, COSINE.

Hlavní hodinový signál CLK má nastavenou aktivní náběžnou hranu. Pin A označuje adresu vybranou pro zápis do paměti fázového inkrementu (PINC) a do paměti fázového offsetu (POFF).



Obr.30: Schématický symbol DDS [11].

DDS může mít kvadraturní výstup (sinus a kosinus), nebo jediný výstupní port (sinus nebo kosinus). Označení „Negative Sine“ a „Negative Cosine“ převrací znaménko hodnot výstupních

signálů. Je-li například zvolen kvadratický výstup (políčko „Sine and Cosine“) a „Negative Sine“ i „Negative Cosine“ nejsou zatrženy, bude výstupní signál $s(n)$ z [11]:

$$s(n) = e^{j\Theta(n)} = \cos\Theta(n) + j \sin\Theta(n).$$

Při zatržení volby „Negative Sine“ se změní $s(n)$ na:

$$s(n) = e^{-j\Theta(n)} = \cos\Theta(n) - j \sin\Theta(n).$$

Pro generování signálu kosinus je potřeba zvolit v okně parametrů výstup „Cosine“. Ostatní vstupní a výstupní signály nejsou nutné. Základními parametry jsou DDS Clock Rate a Output frequency. DDS Clock Rate určuje rychlost vzorkování, čím je větší, tím podrobnější průběh v čase dostaneme. Podle DDS Clock Rate a výstupní frekvence je stanoven počet vzorků na periodu a tím ovlivněn i kmitočet výstupní vlny. Zvýšení DDS Clock Rate při stejné výstupní frekvenci vede ke snížení kmitočtu výstupní vlny. Výstupní frekvence musí splnit vzorkovací teorém, nesmí překročit polovinu kmitočtu DDS Clock Rate. Aby byl průběh kosinu zřetelný v tomto simulátoru, je vhodné použít alespoň deset vzorků na periodu. Pak je možné generovat nosnou vlnu o frekvenci přibližně 15 MHz (generování vyšších frekvencí naráží na omezení volné verze). Tomu odpovídá např. DDS Clock Rate 155MHz a Output Frequency 15MHz.

Dalším parametrem je rozlišení vertikálních úrovní signálu, neboli Spurious Free Dynamic Range. Čím vyšší je hodnota, tím je amplituda generované vlny rozdělena na více hladin, výstupní šířka dat a tedy i přesnost se zvětšuje. Rovněž ale dosažitelná frekvence kosinu se zároveň zmenšuje. Hodnota 36dB generuje rozsah úrovní od -32 do 31 (6 bitů).

Modulace datového signálu

Modulací datového signálu na nosnou vlnu se rozumí vzájemné vynásobení jejich průběhu. Zatímco modulovaná data mají binární průběh velikosti jednoho bitu, nosná kosinová vlna představuje logický vektor o délce několika bitů, protože vyjadřuje reálný průběh. Aby bylo možné signály násobit, binární signál je nutné rozšířit na vektor o velikosti vektoru představujícího kosinus. Formát čísla v tomto případě je Signed, vzorky signálu kosinus mají kladné i záporné hodnoty.

Pro binární číslo ,0‘ bude mít 6-bitový vektor hodnotu 100000. Bit ,1‘ lze vyjádřit jako vektor 011111. Hodnota není přímo rovna jedné, ale nejvýše 0,99999.... Hodnota 1 není pro tvar Signed definována. Pro rozlišení úrovní tento způsob zcela postačuje. Vektor 100000 lze v okně Test Bench WaveForm zobrazit v dekadické podobě, jeho hodnota je -32. Vektor 011111 má dekadickou hodnotu 31. Převod může být proveden vložení následující podmínky do procesu:

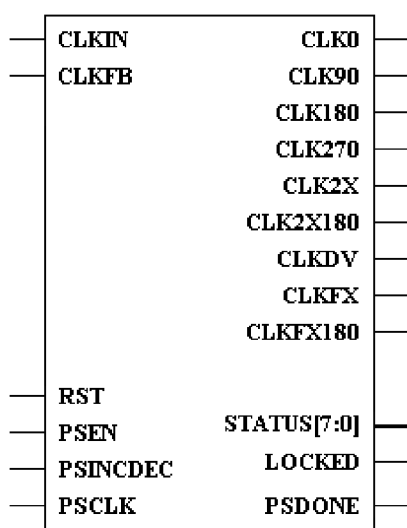
```
if s2='1' THEN
    s3<="011111";--01111111...etc (signed)
end if;
if s2='0' THEN
    s3<="100000";--10000000...etc (signed)

end if;
```

Signál s_2 a s_3 jsou výše uvedené vnitřní signály. Obdobnou podmínkou lze dosáhnout zpětného převodu. Jestliže první bit vektoru je 1, výsledkem bude bit ,0'. V opačném případě bit ,1'.

DCM – Digital clock manager

Každá komponenta obvodu má mezi vstupními piny vstup pro hodinový signál. Za aktivní je obvykle považována vzestupná (náběžná) hrana. Uvažujeme-li kódové rozprostření 31 chipů na 1 bit, bude vstupní signál modulátoru s bitovou rychlostí 100kbit/s rozprostřen na signál s frekvencí 3,1MHz. Nosná vlna by měla mít patřičně vyšší kmitočet, ve volné verzi prostředí ISE však nastávají problémy s určitými frekvenčními omezeními. Pro simulaci postačuje vygenerovaná kosinová vlna o frekvenci 15MHz. Vyžadují-li komponenty více jak jeden hodinový kmitočet, je výhodné použití bloku DCM (Obr.31) podle [12].



Obr.31: Schématický symbol DCM [12].

Vstupním signálem DCM z [12] je CLKIN a CLKFB. Signál CLKIN je systémový čas, který může být generován automaticky z vnějšího zdroje anebo ze samotného DCM. Na pin CLKFB je zavedena smyčka zpětné vazby (z výstupu CLK0 nebo CLK2X).

Výstupními signály jsou jednak CLK0, což je nezměněný signál ze vstupu CLKIN, a dále fázově posunuté signály CLK90, CLK180, CLK270, výstupní signál s dvojnásobnou frekvencí CLK2X a CLK2X180, dělič kmitočtu CLKDV a signál CLKFX, určený ke zlomkovému vyjádření kmitočtů.

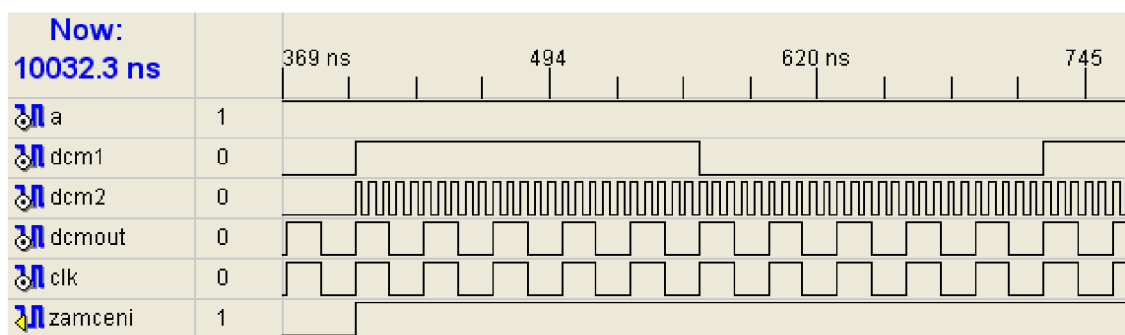
Při použití bloku DCM je třeba dát pozor na jistá kmitočtová omezení vstupního signálu CLKIN. Při použití děliče musí být kmitočet CLKIN v rozmezí 18 až 280MHz. Frekvenční syntezátor vyžaduje rozsah vstupního hodinového signálu 1 až 280MHz.

Časové průběhy signálů

Původní datový signál a má nejnižší kmitočet a lze ho nastavit např. při simulaci pomocí Testbench WaveForm. Signál a je rozprostřen vnitřním signálem s_1 s periodou 31krát kratší,

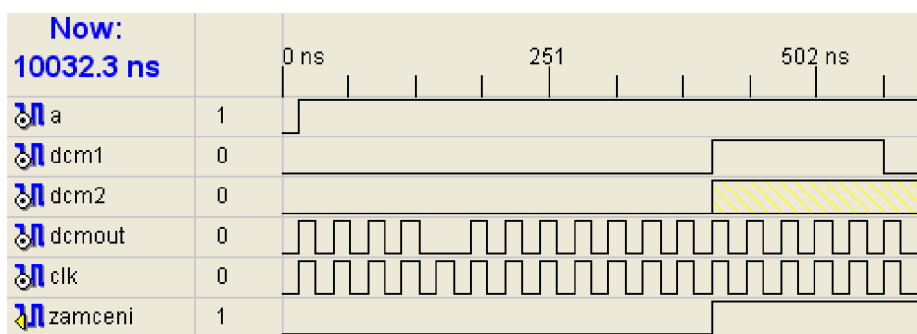
jenž byl získán pomocí LFSR 5.řádu. V tomto případě kódová posloupnost $s1$ vznikla Fibonacciho metodou zadáním polynomu $x^5+x^2+x^0$, vazby byly sečteny operací XOR. LFSR musí mít na vstup přiveden hodinový signál o frekvenci 3,1MHz (při zmiňovaném předpokládaném datovém toku 100kbit/s). Generátor nosné vlny DDS má na vstup přiveden kmitočet 155MHz.

Hodinový kmitočet signálu Clk pro použití DCM může být zvolen libovolně s ohledem na parametry DCM, nejlépe tak, aby hodnoty pro dělení a násobení byly co nejmenší. Optimální hodnota byla zvolena 31MHz. Pro dosažení 3,1MHz pak stačí použít dělič kmitočtu 10ti (signál $dcm1$), požadovaný vysoký kmitočet 155MHz vznikne násobením 5ti (signál $dcm2$). Srovnání časových průběhů je na Obr.32. Signál $dcmout$ je výstupem pinu CLK0, tedy nezměněným hodinovým signálem.



Obr.32: Generování dvou různých hodinových kmitočtů pomocí bloku DCM.

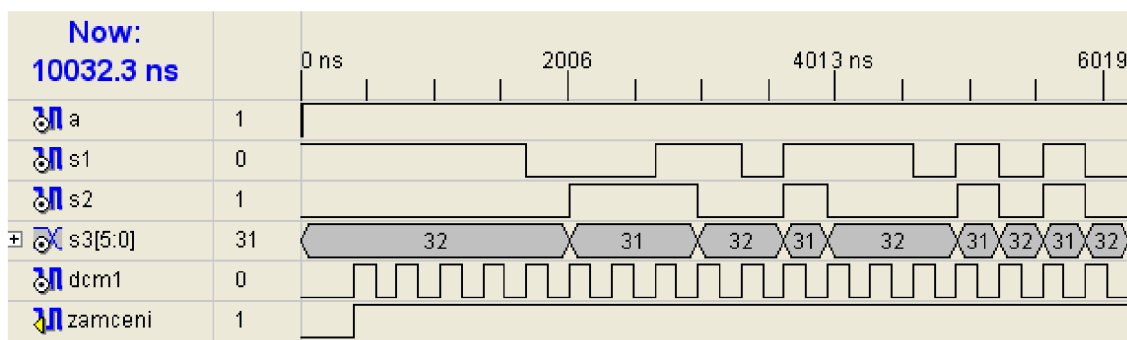
Na počátku nějakou dobu trvá, než jsou potřebné funkce DCM, závislé na CLKIN, na výstupu dostupné pro další aplikace. Tento stav ukazuje signál $Zamceni$ z portu LOCKED. Jakmile jsou hodinové signály dostupné, změní se hodnota $Zamceni$ z 0 na 1 (Obr.33).



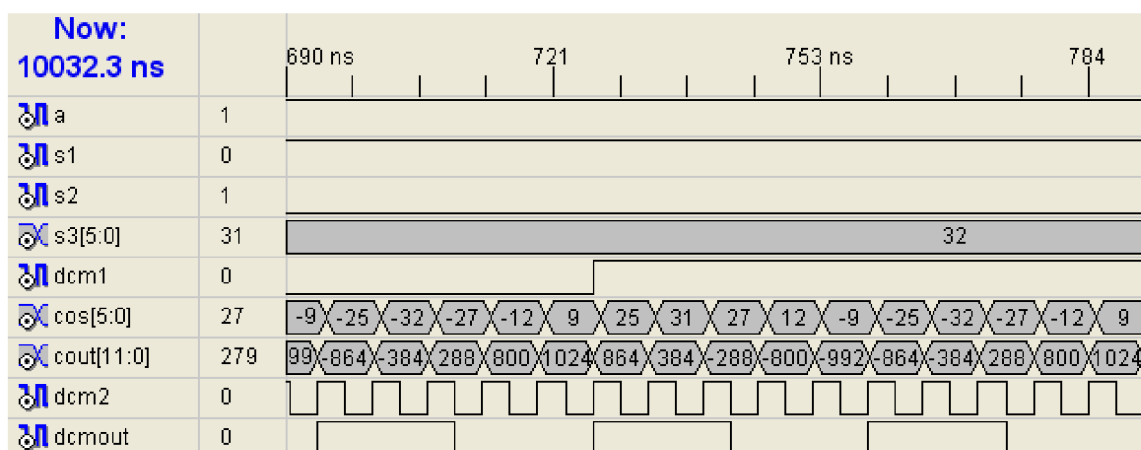
Obr.33: Počáteční průběh signálů po restartu.

Signál a má periodu (délku) bitů 31krát větší než je perioda hodinového signálu $dcm1$. Perioda signálu kódovací posloupnosti $s1$ odpovídá hodinovému signálu $dcm1$. Součtem signálu a a $s1$ vzniká vnitřní signál $s2$, s posunutím 1 periody signálu $dcm1$ (viz. Obr.34). Signál $s3$ je již zmiňované převedení binárního signálu $s2$ na reálný vícebitový vektor, potřebný k rozlišení průběhu kosinusové vlny. Perioda se při tomto kroku nemění (Obr.34).

Signál cos je generován dle vstupního hodinového signálu $dcm2$ (Obr.35). Součin signálů $s3$ a cos představuje výstup modulátoru $Cout$. Výstup z násobičky je posunut o 3 periody $dcm2$.



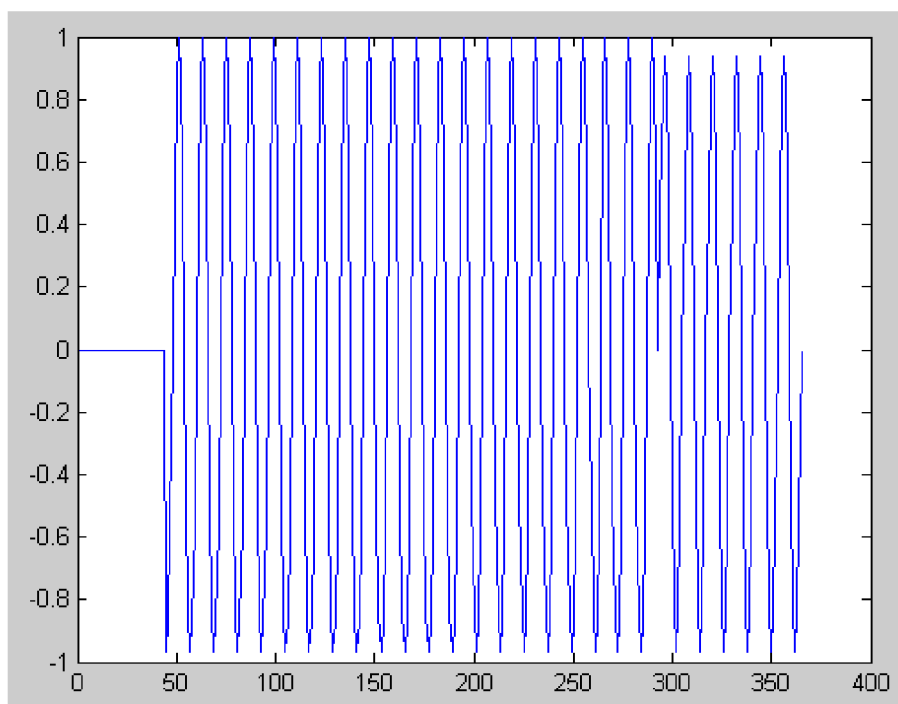
Obr.34: Časový průběh vstupních signálů.



Obr.35: Časový průběh výstupního signálu (Cout).

Výsledný průběh *Cout* lze dále zobrazit v Matlabu jako graf spojitě funkce. Na Obr.36 jsou zobrazeny vzorky hodnot pro prvních 2500ns, 5 bitů ,0' a další bity ,1'. Výstupní hodnoty byly vygenerovány pomocí source VHDL Test Bench. Tento soubor s příponou .vhd obsahuje podobně jako vlastní modul deklarace vstupních a výstupních signálů, dále přiřazení signálů portům a nastavení parametrů simulace. Důležité je nastavení vlastní hodinové frekvence signálu *Clk* stanovením poloviny doby periody na 16,1 ns. Pomocí dalších procesů jsou hodnoty zvoleného signálu zapsány do textového souboru a nasypány do Matlabu ve formě prvků vektoru. K zápisu do souboru jsou zapotřebí knihovny `ieee.std_logic_signed.ALL` a `std.textio.ALL`.

Velikost prvků vektoru byla nakonec upravena na max. rozsah hodnot od -1 do +1 dělením hodnotou celkového rozsahu (1024). Vykreslení provedla funkce `plot()`.



Obr.36: Průběh modulovaného signálu.

Závěr

System s rozprostřeným spektrem lze realizovat nejjednodušeji tak, že aplikujeme rozprostírací posloupnost přímo na datový signál. Základní variantu tvoří pseudonáhodné posloupnosti, které se dají použít bez dalších kódů, výhodnější však je využití kombinací ortogonálních a pseudonáhodných kódů, jako v systému UMTS. Rozdílné vlastnosti umožňují další zlepšení kvality přenosu dat.

Vlastnosti rozprostíracích kódů jsou přehledně čitelné z grafů korelačních a autokorelačních křivek. Ukázány jsou generátory krátkých PN sekvencí, ortogonálních sekvencí a sekvencí ze systému UMTS vytvořené v programu Matlab.

Při implementaci v obvodu FPGA v prostředí Xilinx ISE byl k vytvoření jednotlivých bloků použit Core Generator a Xilinx Architecture Wizard. K rozprostření signálu byla využita krátká pseudonáhodná posloupnost, rozprostřený signál byl převeden z binárního jednobitového průběhu na reálný vícebitový vektor a modulován na nosnou vlnu.

Literatura

- [1] DOBEŠ, J., ŽALUD, V., *Moderní radiotechnika*. BEN, Praha 2006. ISBN 80-7300-132-2.
- [2] Maxim Integrated Products, *Spread-Spectrum* (2005): <http://www.maxim-ic.com/an1890>.
- [3] MARŠÁLEK, R. *Teorie rádiové komunikace - rozpracovaná verze II*. El.verze. VUT, Brno.
- [4] HANUS, S. *Rádiové a mobilní komunikace*. El.verze. VUT, Brno.
- [5] BLÁZSOVITS, G. *DIP - Digital Image Processing (Hadamardova transformace)*, 2003-06: <http://dip.sccg.sk/transf/transf.htm>
- [6] MOLNÁR, J. *Modulace QPSK*. <http://www.umts.wz.cz>.
- [7] ETSI TS 125 213 V9.2.0 (2010-10).
- [8] DOUŠA, J. *Jazyk VHDL*. Vydavatelství ČVUT, Praha 2003. ISBN 80-01-02670-1
- [9] *Adder/Subtractor v7.0*. <http://www.xilinx.com>.
- [10] *Linear Feedback Shift Register v3.0*. <http://www.xilinx.com>.
- [11] *DDS v5.0*. <http://www.xilinx.com>.
- [12] *Using Digital Clock Managers (DCMs) in Spartan-3 FPGAs*. <http://www.xilinx.com>.
- [13] <http://hw.cz/Teorie-a-praxe/Dokumentace/ART365-Nebojte-se-FPGA.html>