

Būvējam OFDM sakaru sistēmu. versija 1.7

Artūrs Āboltniš¹

2011. g. 16. jūnijā

¹Finansēts no Valsts pētījumu programmas

Saturs

| | |
|---|-----------|
| 1. 1.Modelis: Sakaru sistēma ar sinhrono kanālu. | 3 |
| 1.1. Modeļa apraksts | 3 |
| 1.2. Raidītājs | 3 |
| 1.3. Uztvērējs | 4 |
| 1.4. Sakaru sistēmas shēma un galvenie parametri | 5 |
| 1.5. Kanāla kropļojumu ietekme | 5 |
| 2. 2.Modelis: Sistēma ar kadru sinhronizāciju | 6 |
| 2.1. Ievads | 6 |
| 2.2. Sakaru sistēmas blokshēma | 6 |
| 2.3. Raidītājs | 6 |
| 2.3.1. Bloku sinhronizācijas signāla ģenerēšana | 6 |
| 2.3.2. Raidītāja modelis | 7 |
| 2.4. Uztvērēja sinhronizācija | 8 |
| 2.4.1. Īso simbolu detektors | 8 |
| 2.4.2. Uztvērēja strobēšanas modulis | 9 |
| 2.5. Uztvērējs | 9 |
| 2.6. Sakaru sistēma | 10 |
| 2.7. Secinājumi un sakaru sistēmas galvenie parametri | 10 |
| 3. 3.Modelis: Sistēma ar kanāla novērtēšanu un korekciju lietojot pilostsimbo- lus | 11 |
| 3.1. Ievads | 11 |
| 3.2. Kanāla novērtēšana | 11 |
| 3.3. Pilottoņu izvēle | 11 |
| 3.4. Sakaru sistēmas blokshēma | 12 |
| 3.5. Raidītājs | 12 |
| 3.6. Uztvērējs | 13 |
| 3.6.1. Ekvalaizers | 13 |
| 3.7. Frekvenču selektīvs sakaru kanāls | 14 |
| 3.8. Sakaru sistēma | 14 |
| 3.9. Secinājumi un sakaru sistēmas galvenie parametri | 15 |
| 4. 4.Modelis: Sistēma ar kanāla novērtēšanu un korekciju lietojot pilot toņus | 17 |
| 4.1. Ievads | 17 |
| 4.2. Pilottoņi | 17 |
| 4.3. Sakaru sistēmas blokshēma | 18 |
| 4.4. Raidītājs | 18 |
| 4.5. Uztvērējs | 19 |
| 4.5.1. Sinc interpolātors | 19 |
| 4.6. Frekvenču selektīvs sakaru kanāls | 20 |
| 4.7. Sakaru sistēma | 20 |
| 4.8. Simulācijas rezultāti | 20 |

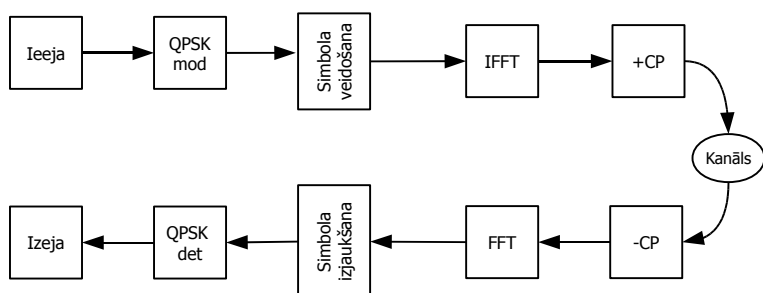
| | |
|--|-----------|
| 4.9. Secinājumi un sakaru sistēmas galvenie parametri | 22 |
| 5. 5.Modelis: Sistēma ar virknes datu pārraidi. | 24 |
| 5.1. Ievads | 24 |
| 5.2. Sakaru sistēmas blokshēma | 24 |
| 5.3. Raidītāja modelis | 25 |
| 5.4. Uztvērēja modelis | 25 |
| 5.5. Simbolu sinhronizācijas bloks | 26 |
| 5.6. Cikliskā prefiksa detektors | 27 |
| 5.7. Skaitliski vadāmais ģenerators | 28 |
| 5.8. Sakaru sistēma | 29 |
| 5.9. Simulācijas rezultāti | 29 |
| 5.10. Secinājumi un sistēmas galvenie parametri | 31 |
| 5.11. Sistēmas uzlabošana | 31 |
| 6. 6.Modelis: Sistēma ar diskretizāciju. | 32 |
| 6.1. Ievads | 32 |
| 6.2. Sakaru sistēmas blokshēma | 32 |
| 6.3. Raidītāja modelis | 33 |
| 6.4. Uztvērēja modelis | 33 |
| 6.5. Virknes - paralēlais pārveidotājs | 35 |
| 6.6. Simbolu sinhronizācijas bloks | 35 |
| 6.7. Laika proporcionāli-integrāli-diferenciālais vadības bloks | 36 |
| 6.8. Skaitliski vadāmais ģenerators | 37 |
| 6.9. Uztvērēja apstrādes bloks | 37 |
| 6.10. Paralēlais-virknes pārveidotājs uztvērēja izejā | 39 |
| 6.11. Sakaru sistēma | 39 |
| 6.12. Simulācijas rezultāti | 39 |
| 6.13. Secinājumi un sistēmas galvenie parametri | 41 |
| 6.14. Sistēmas uzlabošana | 41 |
| 7. 7.Modelis: Sistēma frekvences sinhronizāciju. | 42 |
| 7.1. Ievads | 42 |
| 7.2. Sakaru sistēmas blokshēma | 42 |
| 7.3. Raidītāja modelis | 43 |
| 7.4. Uztvērēja modelis | 43 |
| 7.5. Kanāla novērtējuma fāžu apstrādes bloks | 43 |
| 7.6. Frekvences proporcionāli-integrāli-diferenciālais vadības bloks | 44 |
| 7.7. Uztvērēja apstrādes bloks | 45 |
| 7.8. Sakaru sistēma | 45 |
| 7.9. Trokšņa jaudas aprēķinu bloks | 47 |
| 7.10. Simulāciju rezultāti | 47 |
| 7.11. Secinājumi un sistēmas galvenie parametri | 47 |
| 7.12. Sistēmas uzlabošana | 48 |
| A Saīsinājumi | 49 |
| B Darba žurnāls | 51 |

1. nodaļa

1. Modelis: Sakaru sistēma ar sinhrono kanālu.

1.1. Modeļa apraksts

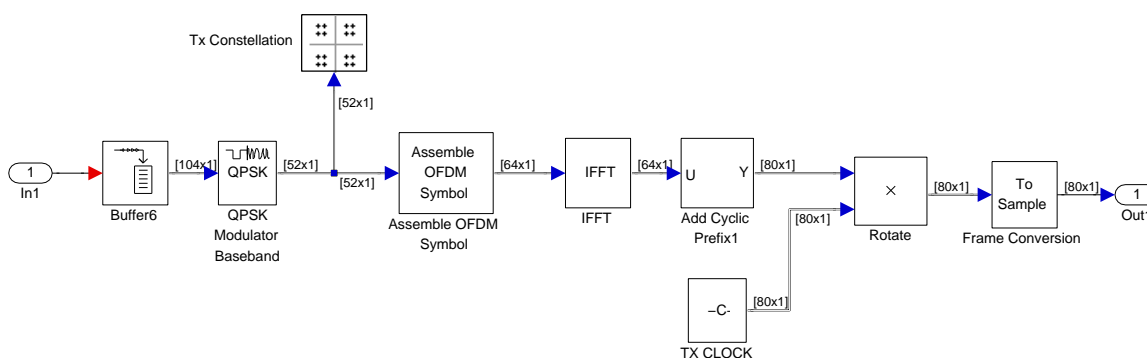
Sāksim ar vienkāršotu, uz *Ortogonalās Frekvenčdales Multipleksāciju* (Orthogonal Frequency Division Multiplexing, OFDM) balstītu sakaru sistēmu. Tās blokhēma ir parādīta attēlā 1.1..



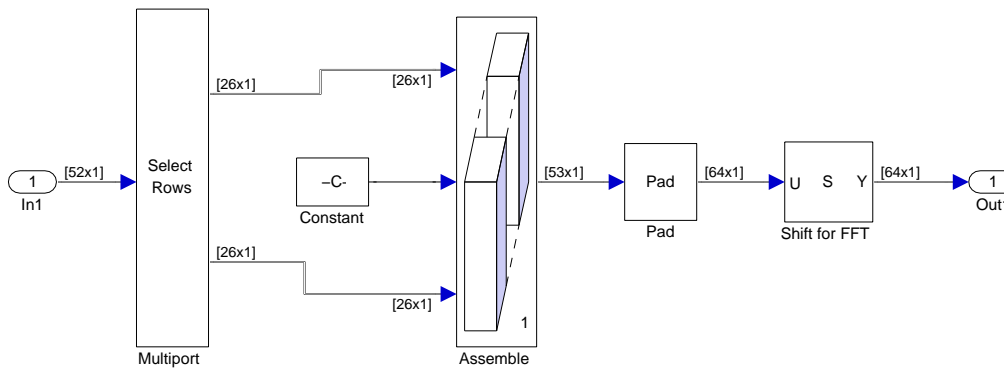
1.1. att. Blokhēma sakaru sistēmai ar sinhrono kanālu

1.2. Raidītājs

Raidītāja Simulink modelis ir parādīts attēlā 1.2. Ar QPSK palīdzību lietotāja datu bināro plūsmu pārveidosim kompleksu skaitļu plūsmā. Pēc tam sagrupēsīm datus pa 52 apakšnesējām un blokā "simbola veidošana" veiksīm apakšnesēju pārvietošanu tā lai iegūtu ātri rimstošu

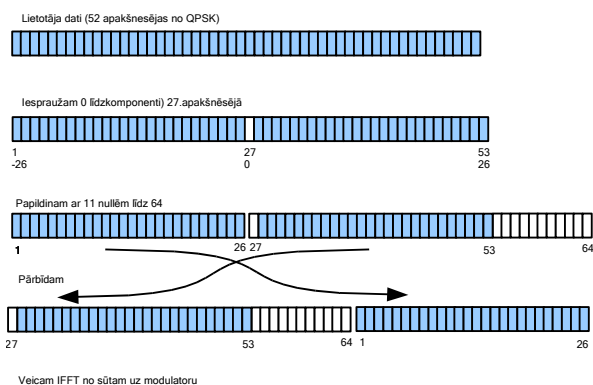


1.2. att. Raidītājs sistēmai ar sinhrono kanālu



1.3. att. OFDM simbola veidošanas bloks

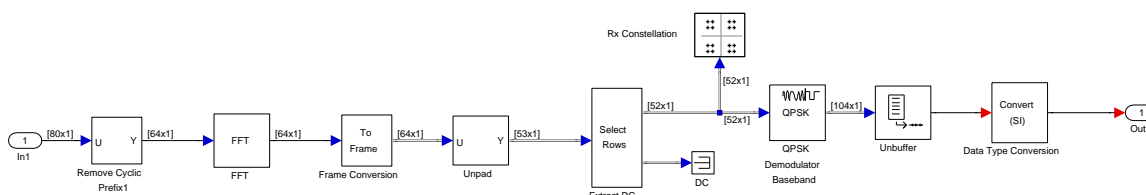
OFDM signāla spektru. Pārgrupēšanas process shematiski ir parādīts attēlā 1.4.. OFDM simbols sastāv no 52 datu apakšnesējām un specifiski novietotām 12 tukšām apakšnesējām. OFDM simbola veidošanas bloka shēma ir dota 1.3. attēlā. Izveidotos OFDM simbolus padodam uz IFFT. Pievienojam ciklisko prefiksu. Rezultātā izveidojās monolīti 80-čipu (laika apgabala) simboli, kas nepārtrauktā plūsmā var tikt pārraidīti sakaru kanālā. Takts signāla bloks 'TX CLOCK' nodrošinās noteiktus nolašu laikus raidītājā un uztvērējā.



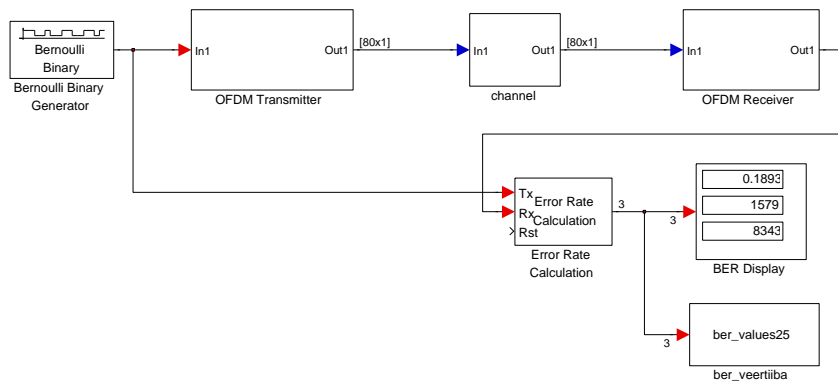
1.4. att. OFDM simbola veidošana

1.3. Uztvērējs

Uztvērējs saņemtajam signāla 80 nolašu vektoram novāc ciklisko prefiksu (pirmās 16 nolases) un veic FFT. Tālāk turpinās darbs frekvenču apgabalā: tiek novāktas tukšās apakšnesējas, kas tika iespraustas (skat 1.4. attēlu) lai uzlabotu OFDM signāla spektru. QPSK demodulators pārveido kompleksās nolases binārā datu plūsmā. Uztvērēja Simulink modelis ir parādīts zemāk:



1.5. att. Uztvērējs sistēmai ar sinhrono kanālu



1.6. att. Sakaru sistēmas ar sinhrono kanālu Simulink blokshēma

| Parametrs | Vērtība |
|-------------------------|-----------|
| Datu apakšnesēju skaits | 52 |
| Takts frekvence | 20 kHz |
| Pārraides ātrums | 26 kbit/s |
| SNR pie BER=10e-3 | 35dB |

1.1. tabula Sakaru sistēmas ar sinhrono kanālu galvenie parametri.

1.4. Sakaru sistēmas shēma un galvenie parametri

Savienosim raidītāju un uztvērēju ar sakaru kanālu. Kā datu avotu pieslēgsim Bernulli ģeneratoru. Sakaru sistēmas efektivitātes mērīšanai izmantosim bitu kļūdu intensitātes (BER) aprēķinu bloku no Simulink bibliotēkas. Sakaru sistēmas Simulink modelis ir parādīts attēlā 1.6..

Sakaru sistēmas galvenie parametri:

1.5. Kanāla kropļojumu ietekme

Augstāk aprakstīto sistēmu var izmantot informācijas pārraidei kanālos ar aditīvu troksni. Taču kanālos pat ar salīdzinoši nelieliem Releja pamirumiem (Rayleigh fading) vai frekvenču-selektīvu vājinājumu un aizturi, datu pārraide ir apgrūtināta. Lai cīnītos ar kanāla kropļojumiem, ir nepieciešams izmantot kanāla raksturlīknes korekciju. Šim nolūkam kanālā ir jāpārraida pilottoņi un uztvērējā ir jāmēra pilottoņu līmenis.

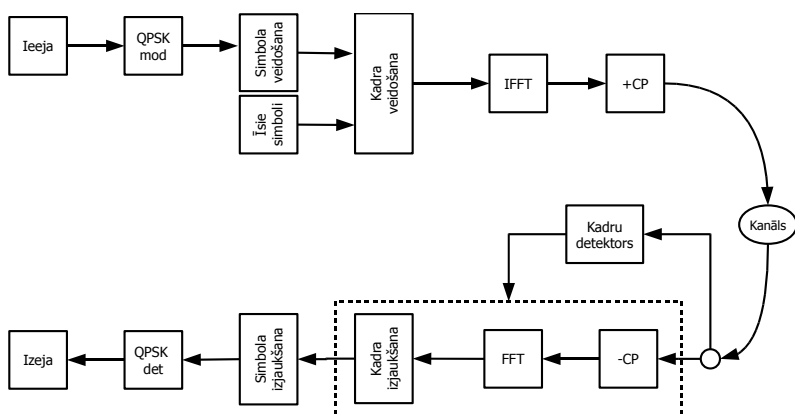
2. nodaļa

2. Modelis: Sistēma ar kadru sinhronizāciju

2.1. Ievads

Lai izveidotu sistēmu, kas nodrošinās kanāla pārvades funkcijas (KPF) korekciju izmantojot speciālus simbolus, ir jāveic pāreja no nepārtrauktas simbolu plūsmas uz noteikta garuma kadriem. Izveidosim sistēmu, kas pārraidīs informāciju kadros pa 20 simboliem.

2.2. Sakaru sistēmas blokshēma



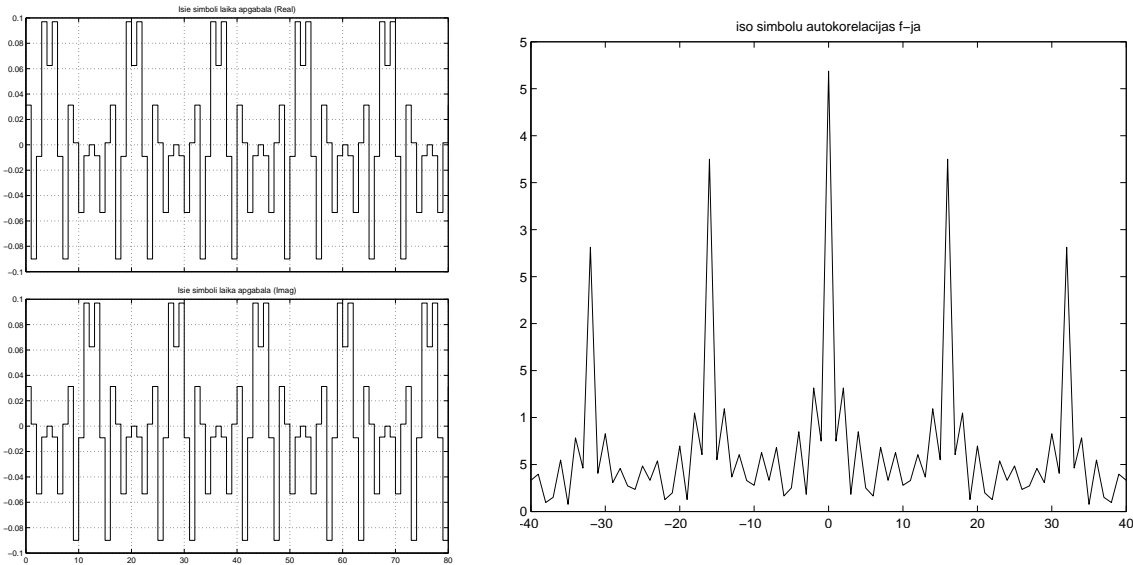
2.1. att. Sakaru sistēma ar kadru sinhronizāciju

Lai nodrošinātu kadru sinhronizāciju, papildināsim mūsu iepriekšējo modeli ar attiecīgiem blokiem. Raidītāju papildināsim ar kadru salikšanas moduli. Tam pieslēgsim kadru sinhronizācijas simbolu ģeneratoru. Uztvērējā būs jāizveido kadru sinhronizācijas simbolu detektors. Jaunās sakaru sistēmas blokshēma ir parādīta attēlā 2.1.

2.3. Raidītājs

2.3.1. Bloku sinhronizācijas signāla ģenerēšana

Ņemot vērā, ka informācija tiek pārraidīta 20 simbolu kadros, ir jānodrošina mehānisms kā ziņot uztvērējam par kārtējā kadra sākumu. Pievienosim vēl divus vienādus dienesta simbolus - *īsos simbolus*, kas signalizēs par kadra sākumu. Katra īsā simbola signālu izvēlēsimies tādu,



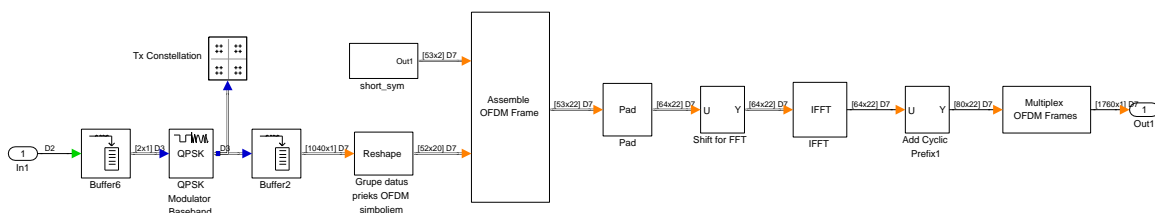
2.2. att. Īsie simboli: laika apgabala signāls (pa kreisi); autokorelācijas f-ja (pa labi)

lai pēc tā transformācijas no frekvenču apgabala uz laika apgabalu (IFFT) tam būtu izteiktas auto-korelācijas īpašības. Paņemsim čipu virkni, kas tiek izmantota IEEE 802.11: 0, 0, 1+j, 0, 0, 0, -1-j, 0, 0, 0, 1+j, 0, 0, 0, -1-j, 0, 0, 0, -1-j, 0, 0, 0, 1+j, 0, 0, 0, 0, 0, 0, 0, 0, -1-j, 0, 0, 0, -1-j, 0, 0, 0, 1+j, 0, 0, 0, 1+j, 0, 0, 0, 1+j, 0, 0, 0, 1+j, 0, 0.

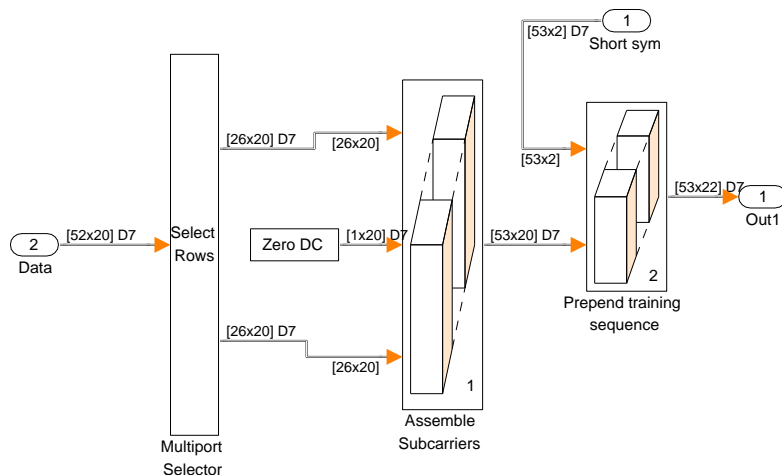
Pēc dotās simbolu virknes IFFT transformācijas, veiksīm apakšnesēju 'pārbīdīšanu' kā bija parādīts 1.4. attēlā un veiksīm cikliskā prefiksa pievienošanu. 2.2. attēlā ir redzama iegūtā simbolu virkne. No viena 80-čipu īsā simbola frekvenču apgabalā veidojas 5 identiski 16 nolašu gari īsie simboli laika apgabalā. Šādā veidā, divas reizes pēc kārtas tie tiks pārraidīti sakaru kanālā. Simbolu virknes auto-korelācijas īpašības ir redzamas attēlā 2.2. pa labi. Redzam ka auto-korelācijas pīķi atkārtojas ik pēc 16 nolāsēm. Tas nozīmē, ka uztvērēja īso impulsu detektorā var izmantot 16 nolases garu korelācijas paraugu.

2.3.2. Raidītāja modelis

Ievietosim augstāk aprakstīto sinhrosignālu ģeneratoru raidītājā, kuru esam izgatavojuši iepriekšējā nodaļā. Pirms sinhronizācijas signālu pievienošanas lietotāja datu plūsmas QPSK 52 nolašu vektori ir jāsagrupē pa 20. Jaunais raidītāja Simulink modelis ir dots zemāk. OFDM apakšnesēju un sinhronizācijas simbolu salikšanas bloka 'Assemble OFDM frames' Simulink modelis ir dots attēlā 2.4.



2.3. att. Raidītāja Simulink modelis sistēmā ar kadru sinhronizāciju



2.4. att. Simbolu un kadru veidošanas bloks

2.4. Uztvērēja sinhronizācija

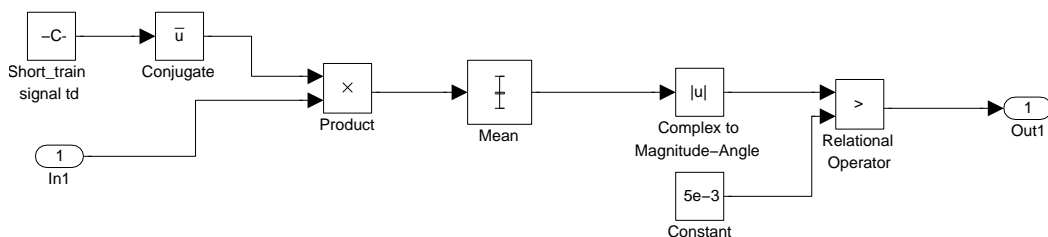
Uztvērējam ir jāspēj noteikt kārtējā kadra sākumu. Pateicoties tam, ka katra kadra sākumā esam izvietojusi divus kadru sinhronizācijas simbolus (īsos simbolus), kadra sākumu var noteikt izveidojot iekārtu, kas reaģēs uz īso simbolu kadriem. Atrodot īso kadru, mūsu sinhronizācijas detektors ģenerēs impulsu, kas 'palaidīs' uztvērēju uz noteiktu laiku.

2.4.1. Īso simbolu detektors

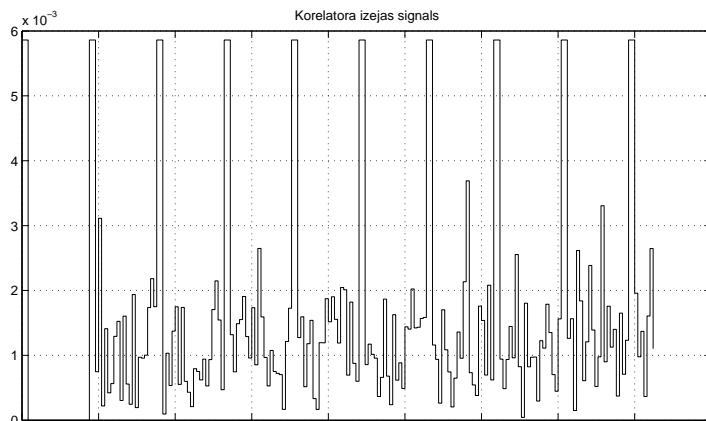
Kā jau iepriekš secinājām, īsos simbolus var atrast izmantojot korellatoru ar 16 nolases garu paraugu. Ņemot vērā, ka mūsu sakaru sistēma strādā sinhronā režīmā (t.i. simboli tiek pārraidīti kā 80 nolašu vektori), uztvērējs vienā taktī saņem uzreiz 5 īsos simbolus. Tādēļ korelācijas paraugu veidosim tādu, kas saturēs uzreiz 5 identiskus 16 nolašu 'īso simbolu' paraugus. Viena 16 nolašu korelācijas parauga vērtības ir šādas:

$$\begin{array}{cccc}
 0.0313-0.0313i, & -0.0900-0.0016i, & -0.0092+0.0533i, & 0.0970+0.0086i, \\
 0.0625, & 0.0970+0.0086i, & -0.0092+0.0533i, & -0.0900-0.0016i, \\
 0.0313-0.0313i, & 0.0016+0.0900i, & -0.0533+0.0092i, & -0.0086-0.0970i, \\
 0-0.0625i, & -0.0086-0.0970i, & -0.0533+0.0092i, & 0.0016+0.0900i
 \end{array}$$

Pirms sareizināšanas ar ieejas signālu, korelācijas paraugam ir jāveic kompleksā konjugācija. Ieejas signāla un kompleksi konjugētā korelācijas parauga reizinājuma vidējā vērtība atspoguļo signālu savstarpējo korelāciju. Šo signālu padosim uz komparatoru, kas izdalīs korelatora izejas



2.5. att. Īso simbolu detektors

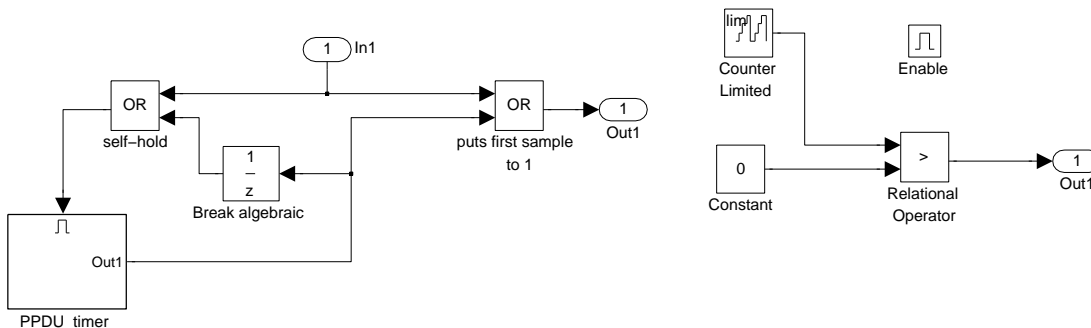


2.6. att. Īso simbolu korelatora izejas signāls.

signāla absolūtās vērtības pīķus un veidos uztvērēja starta impulsu. Īso simbolu detektora Simulink modelis ir dots 2.5.. attēlā, bet korelatora izejas signāls ir parādīts 2.6. attēlā.

2.4.2. Uztvērēja strobēšanas modulis

Kad uztvērējā ienāk kadru sinhronizācijas simbols (īsaiss simbols), īso simbolu filtra izejā parādās loģiskais '1', kas pazūd beidzoties sinhronizācijas simboliem. Tagad mums ir jāizveido sistēma, kas izmantojot šo impulsu ieslēgs uztvērēju uz noteiktu ciklu skaitu. Šim nolūkam izmantosim 'atļauto subsistēmu' (enabled subsystem), kas satur taimeru, ar atgriezenisko saiti. Uztvērēja strobēšanas moduļa Simulink modelis un tā taimeris ir doti 2.7. attēlā.

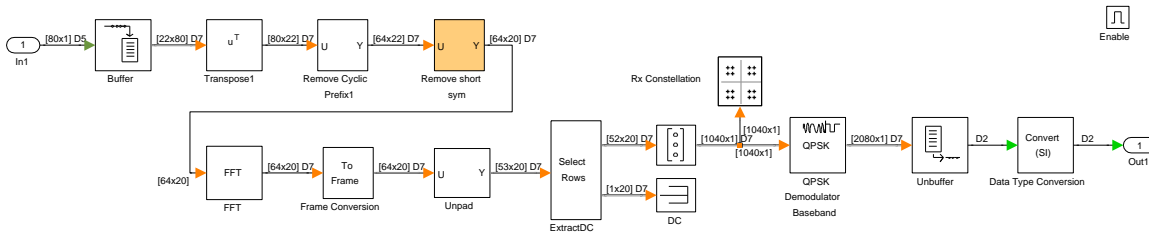


2.7. att.: Uztvērēja strobēšanas modulis (PPDU strobe) (kreisajā pusē) un tā taimeris (PPDU-timer)(labajā pusē)

Tātad: pienākot diviem īsajiem simboliem pēc kārtas, īso simbolu detektors izdod impulsu, kas palaiž uztvērēja strobēšanas moduli. Strobēšanas modulis iedarbina uztvērēju uz tieši 22 cikliem. Pēc tam uztvērējs pāriet gaidīšanas režīmā un gaida nākošo īso simbolu pāri.

2.5. Uztvērējs

Kā sagatavi mēs varam lietot uztvērēju no modeļa ar sinhrono kanālu (1. nodaļu). Uztvērēja ieejā ir jāizveido grupēšanas pa 22 bloks, kas no monolītas 80 nolašu kadru plūsmas veidos blokus pa 22 kadriem. Īso simbolu novākšanas bloks novāks īsos simbolus, jo tie uztvērējam nav nepieciešami (tie bija nepieciešami uztvērēja sinhronizācijai, skat paragrāfu 2.4.). Tālākā apstrāde paliek bez izmaiņām. Uztvērēja blokshēma ir dota 2.8.. attēlā.



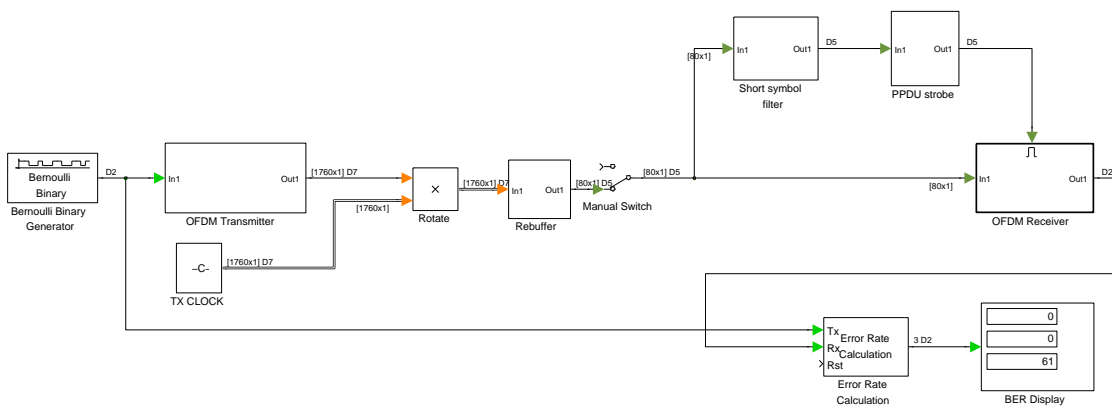
2.8. att. Uztvērēja blokhēma sistēmā ar kadru sinhronizāciju

| Parametrs | Vērtība |
|-------------------------|--------------|
| Datu apakšnesēju skaits | 52 |
| Takts frekvence | 20 kHz |
| Pārraides ātrums | 23.64 kbit/s |
| SNR pie BER=10e-4 | 25dB |

2.1. tabula Sakaru sistēmas ar kadru sinhronizāciju galvenie parametri.

2.6. Sakaru sistēma

Saslēgsim augstāk aprakstītās apakšsistēmas vienotā sakaru sistēmas modelī. Kā iepriekš, raidītājam pieslēgsim nolašu laika uzdošanas bloku "TX CLOCK". Raidītāja izejā novietosim palīgbloku, kas kadru plūsmu (bloki 22x80) pārveidos monolitā 80 nolašu vektoru plūsmā. Ievietosim arī slēdzi ar kura palīdzību varēs pārtraukt kadru plūsmu un 'izjaukt' atsevišķus kadrus. Tādējādi būs iespējams pārlicināties par uztvērēja sinhronizācijas sistēmas darbību. Starp raidītāju un uztvērēju ievietosim AWGN sakaru kanālu. Paralēli uztvērējam izslēgsim uztvērēja sinhronizāciju. Pilnas sakaru sistēmas Simulink modelis ir dots 2.9. attēlā.



2.9. att. Sistēmas ar kadru sinhronizāciju Simulink modelis

2.7. Secinājumi un sakaru sistēmas galvenie parametri

Veicot sakaru sistēmas modelēšanu var konstatēt, ka uztvērējs stabili var sinhronizēties bloku līmenī. Protams, traucējum noturības ziņā dotais modelis ir identisks iepriekšējam, jo kaut arī tika ieviesta bloku sinhronizācija, kanāla pārvades funkcijas korekcija vēl nav realizēta. Tas mums ir jāizdara nākošajā modelī.

3. nodaļa

3. Modelis: Sistēma ar kanāla novērtēšanu un korekciju lietojot pilostsimbolus

3.1. Ievads

Ņemot vērā, ka esam ieviesuši kadru sinhronizāciju, varam ķerties pie kanāla ekvivalēra ievietošanas uztvērējā. Kā jau iepriekš bija minēts, bezvadu kanāla pārvades raksturlīkne ir nevienmērīga. Signāla aizture un vājinājums var svārstīties plašā apgabalā un var būt atkarīga gan no frekvences gan no laika. Lai kompensētu sakaru kanāla, kropļojumus ir jālieto kanāla novērtēšanu un raksturlīknes korekciju.

3.2. Kanāla novērtēšana

Ņemot vērā, ka informācijas pārraide izmantojot daudzas šaurjoslas nesējas, vājinājumu vienas nesējas ietvaros var uzskatīt par konstantu. Kā izmērīt sakara kanāla vājinājumu? Lai izmērītu kanāla komplekso vājinājumu, starp pārraidāmās informācijas simboliem mums ir jāiesprauž īpašus simbolus - pilottoņus. Ja mums ir zināma pilottoņa amplitūda un fāze gan kanāla ieejā $d_p(n)$ gan tā izejā $p_p(n)$, tad kanāla komplekso vājinājumu varam noteikt pēc formulas:

$$\hat{H}(n) = \frac{p_p(n)}{d_p(n)} \quad (3.1)$$

t.i. kanāla vājinājums ir attiecība starp saņemtajiem un pārraidītajiem simboliem.

3.3. Pilottoņu izvēle

Tagad apskatīsim jautājumu par pilottoņu signālu izvēli. Principā par pilottoni var kalpot jebkurš impulss ar ne nulles amplitūdu. Turklāt jo lielāka ir pilottoņa amplitūda, jo precīzāk tiks noteikts kanāla vājinājums. Tādējādi par pilottoņiem parasti izvēlas impulsus ar amplitūdu 1 un fāzi $\pm 90^\circ$.

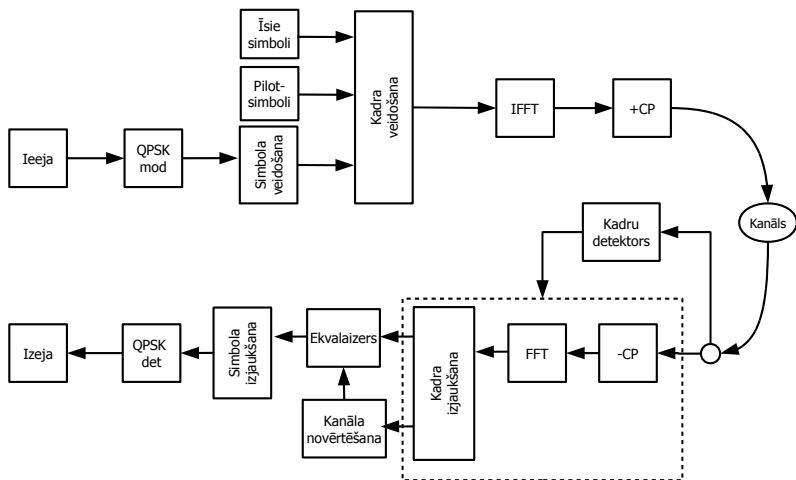
Lai noteiktu cik bieži un kurās kadra pozīcijās ir jānovieto pilottoņi, mums ir jāatceras Naikvista teorēmu, t.i. lai no diskrēta signāla būtu iespējams atjaunot ierobežota spektra kontinuālu signālu, nolasēm ir jābūt ņemtām ne retāk kā apgrieztais signāla divpusējā spektra platums. Tātad, rēķinot pilottoņu attālumu laikā, ir jāņem Doplera frekvenču nobīdes diapazonu. Turpretī lai noteiktu pilottoņu maksimālo attālumu pa frekvenču asi, ir jāņem vērā signāla aiztures deviāciju.

Izvēlēsimies vienkāršāko pilottoņu izvietojumu - izveidosim pilostsimbolu, t.i. simbolu kur visas apakšnesējas ir aizņemtas ar pilottoņiem. Novietosim šādus četrus vienādus simbolus uz-

reiz aiz kadru sinhronizācijas simboliem. Sauksim šos simbolus par *garajiem simboliem*. Ņemot vērā, ka mums ir kadru sinhronizācija, uztvērējs varēs viegli atdalīt pilottoņus no lietderīgajiem signāliem ¹ Izvēlēsimies tādu pilottoņu secību, kāda tiek lietota IEEE 802.11: 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, 0, 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.

3.4. Sakaru sistēmas blokhēma

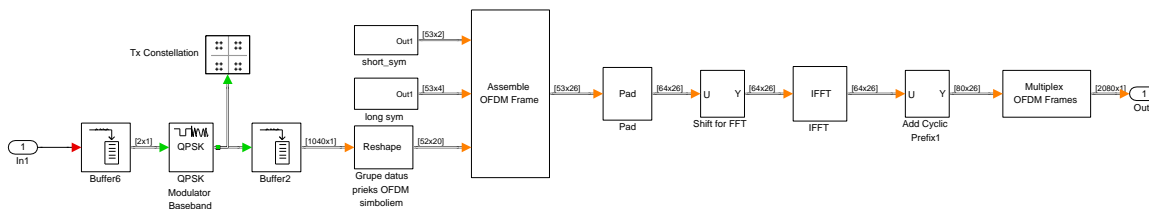
Jaunā modeļa blokhēma ir parādīta 3.1. attēlā. Iepriekšējās shēmas (2.1. attēls) uztvērēja daļa tika papildināta ar kanāla noteikšanas (channel estimation) bloku un ekvalaizeru. Savukārt raidītāja daļā parādās pilottoņu ģenerators.



3.1. att. Sakaru sistēma ar kanāla noteikšanu lietojot pilotsimbolus

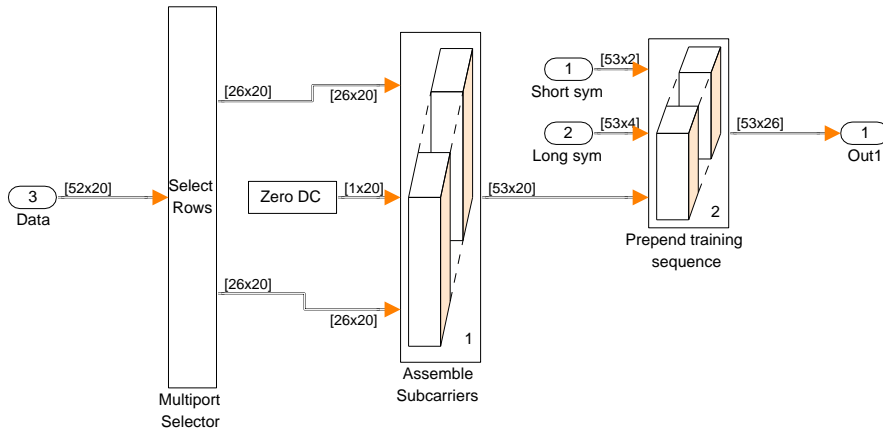
3.5. Raidītājs

Papildināsim iepriekšēja modeļa raidītāju (sk. attēlu 2.3.) ar pilottoņu (garo simbolu) ģeneratoru. Rezultātā rodas shēma, kas parādīta 3.2. attēlā. Signālu apvienošanu veiks blokā "Assemble OFDM frame". Šī bloka Simulink modelis ir dots attēlā 3.3.



3.2. att. Raidītāja modelis sistēmā ar kanāla noteikšanu

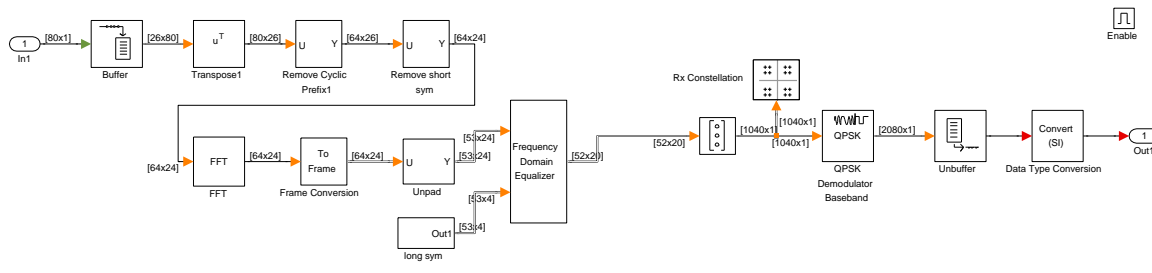
¹Jāpiebilst, ja mēs pārraidītu pilottoņus tikai uz dažām apakšnesējām, tad lai veiktu visa kanāla novērtēšanu, būtu jālieto interpolācija pa frekvenču asi. Turpretī, ja pilottoņi netiktu pārraidīti uz noteiktām nesējām pastāvīgi, tad būtu jāveic interpolācija arī laika ass virzienā.



3.3. att. Simbolu un kadru veidošanas bloks

3.6. Uztvērējs

Par pamatu ņemsim uztvērēju, kas bija uzkonstruēts nodaļā 2.5.. Jaunā uztvērēja modelī būs jāizmaina kadra izjaukšanas shēma. Īso simbolu izvākšanu no kadra atstāsim vecajā vietā. Starp FFT izeju un QPSK modulatoru iespraudīsim jaunu bloku, ko nosauksim par “Frequency domain equalizer”. Jaunā uztvērēja modelis ir dots attēlā 3.4..



3.4. att. Uztvērēja blokshēma sistēmā ar kanāla novērtēšanu

3.6.1. Ekvalaizers

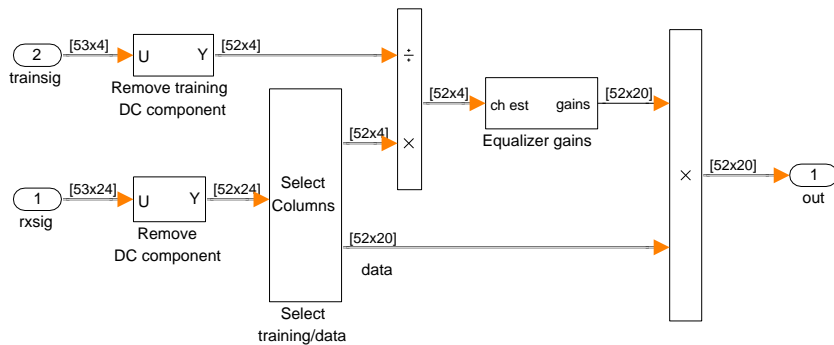
Iespražamā bloka Simulink modelis ir dots attēlā 3.5.. Dotā bloka uzdevumi būs:

- Pilottoņu simbolu atdalīšana no informācijas simboliem (Select columns)
- Kanāla novērtēšana (euqlizer gains)
- Ekvalaizers (reizinātājs izejā)

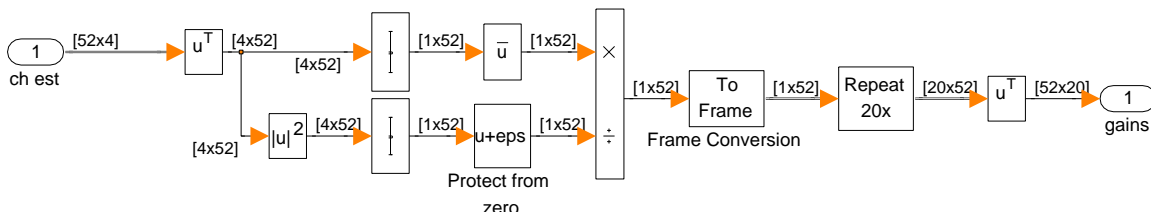
Sarežģītākais uzdevums - kanāla novērtēšana un ekvalaizera koeficientu atrašana tiks veikts atsevišķā Simulink subsistēmā. Kā jau bija minēts iepriekš, komplekso kanāla vājinājumu var atrast izmantojot formulu (3.1), t.i. kā attiecību starp pilotsignāliem kanāla ieejā un izejā. Ekvalaizera koeficienti ir apgriezti proporcionāli nomērītajam kanāla vājinājumam, tādējādi

$$K(n) = \frac{1}{\hat{H}(n)} = \frac{\hat{H}(n)^*}{|\hat{H}(n)|^2} \quad (3.2)$$

Dotās subsistēmas modelis ir dots attēlā 3.6..



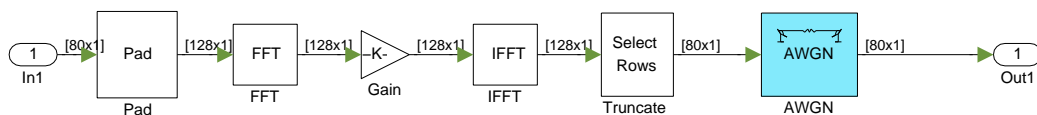
3.5. att. Kanāla novērtēšanas un ekvailaizera bloks



3.6. att. Ekvailaizera koeficientu atrašanas bloks (equalizer gains)

3.7. Frekvenču selektīvs sakaru kanāls

Līdz šim mēs izmantojām tiešo savienojumu starp raidītāju un uztvērēju. Vienīgā ierīce, ko saturēja mūsu “sakaru kanāls” bija slēdzis, kas ļāva izjaukt raidītāja un uztvērēja darbības koherenci un tādējādi testēt kadru sinhronizāciju. Lai novērtētu jaunbūvētās kanāla novērtēšanas un korekcijas sistēmas efektivitāti, starp uztvērēju un raidītāju ieslēgsim ķēdi ar stacionāru, no frekvences atkarīgu komplekso vājinājumu. Piemēram, var būt izmantots modelis, kas ir parādīts attēlā 3.7. Šajā modelī ir arī ieslēgts aditīva baltā Gausa trokšņa ģenerators.



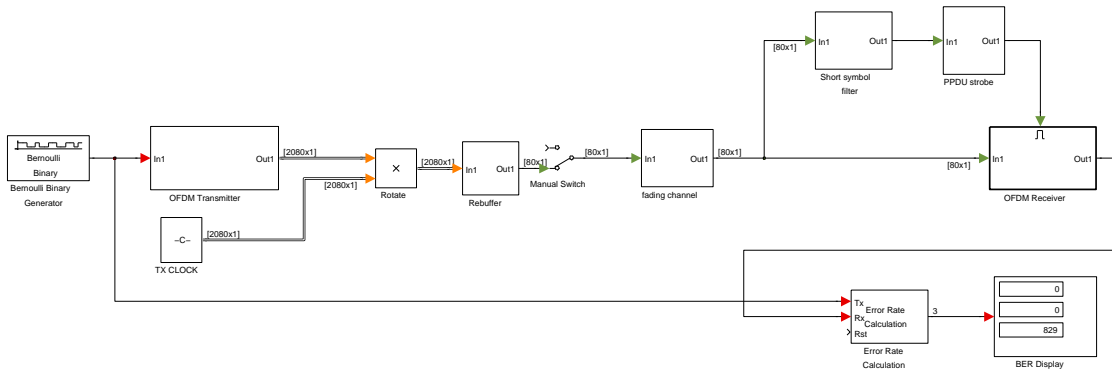
3.7. att. Kanāls ar frekvenču-selektīvu vājinājumu un aditīvu Gausa troksni

3.8. Sakaru sistēma

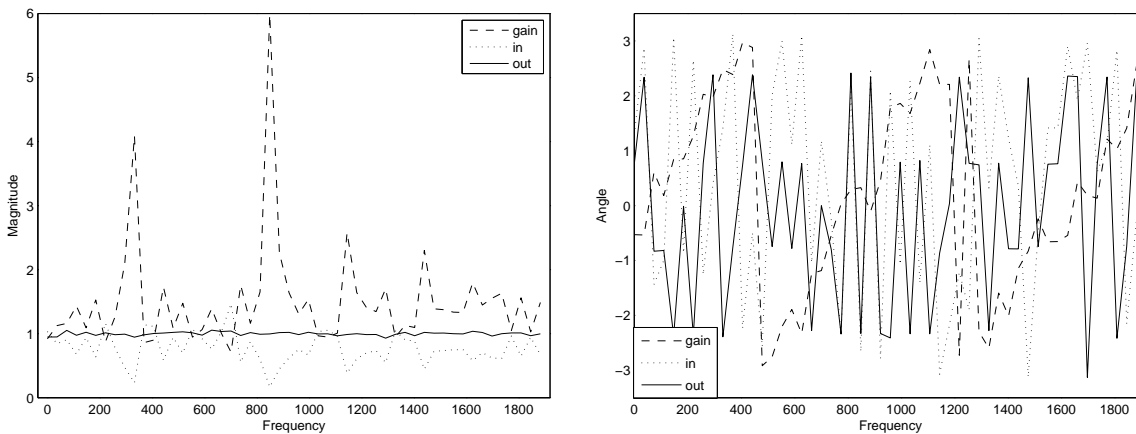
Saslēgsim OFDM raidītāju, sakaru kanālu un uztvērēju vienotā sakaru sistēmā. Neaizmirsīsim arī kadru sinhronizācijas ķēdes. Jaunās sakaru sistēmas Simulink modelis ir dots 3.8. attēlā.

Sākumā pārlicināsimies, ka kanāla novērtēšana un ekvailaizers darbojas. Šim nolūkam izmērīsim kanāla pārvades raksturlīkni, ekvailaizera koeficientus un pilotsignālus pēc korekcijas. Mērījumu rezultāti ir doti attēlā 3.9.

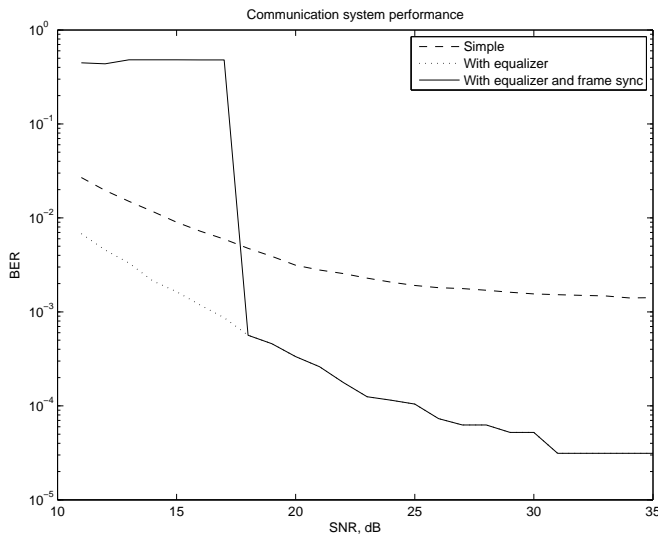
Tagad pamēģināsim nomērīt bitu kļūdas intensitāti (BER) atkarībā no attiecības signāls / troksnis uztvērēja ieejā. Šim nolūkam veiksime virkni mērījumu pie dažādiem kanāla bloka “AWGN” parametra SNR lielumiem. Ieslēgsime frekvenču selektīvo sakaru kanālu un veiksime šādus mērījumus Sakaru sistēmai ar sinhrono kanālu (skat 1..nodaļu) un sistēmai ar kadru sinhronizāciju (skat 2.. nodaļu). Uzņemtās atkarības ar dota attēlā 3.10.



3.8. att. Sistēmas ar kanāla novērtēšanu un korekciju Simulink modelis



3.9. att.: Piektā OFDM datu simbola korekcijas simulācija. *Gain, in, out* nozīmē atbilstoši ekvivalenta koeficientus, ieejas signāla raksturlīkni, izejas signāla raksturlīkni.



3.10. att.: BER atkarība no SNR sistēmai ar sinhrono kanālu, sistēmai ar kadru sinhronizāciju, sistēmai ar kanāla novērtēšanu.

3.9. Secinājumi un sakaru sistēmas galvenie parametri

Veicot sakaru sistēmas modelēšanu konstatējām, ka kanāla novērtēšanas un korekcijas ieviešana ievērojami samazina bitu kļūdas intensitāti. Pie lieliem SNR (>30dB) vinnests sasniedz

| Parametrs | Vērtība |
|-------------------------|-----------|
| Datu apakšnesēju skaits | 52 |
| Takts frekvence | 20 kHz |
| Pārraides ātrums | 20 kbit/s |
| SNR pie BER=10e-4 | 25dB |

3.1. tabula Sakaru sistēmas ar kanāla novērtēšanu un korekciju galvenie parametri

10 reizes un vairāk. 3.10. arī ir redzams, ka pie aptuveni 17dB pārstāj darboties kadru sinhronizācija. Tātad būtiska nozīme ir arī kadru sinhronizācijas sistēmas noturībai pret traucējumiem.

4. nodaļa

4. Modelis: Sistēma ar kanāla novērtēšanu un korekciju lietojot pilot toņus

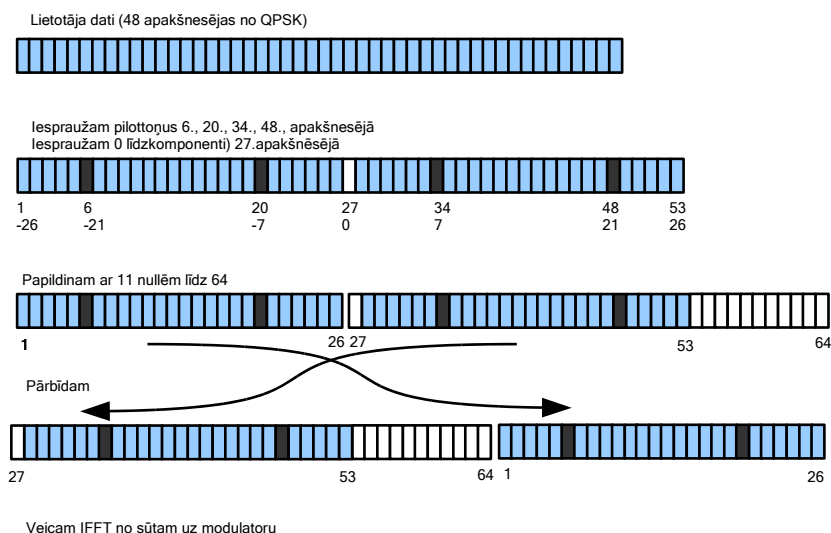
4.1. Ievads

Esam izveidojuši uztvērēju ar kanāla novērtēšanu un korekciju, kas balstīta uz pilot-simboliem t.i. periodiski, katra kadra sākumā izvietotiem OFDM simboliem, kas sastāv vienīgi no pilottoņiem. Šāds kanāla novērtēšanas un korekcijas mehānisms strādā, ja sakaru kanāla izmaiņas ir relatīvi lēnas. Lai nodrošinātu uztvērēja adaptāciju pret straujām sakaru kanāla fāzes un amplitūdas raksturlīknes izmaiņām, pilottoņi ir jāiesprauž starp derīgās informācijas nolasēm.

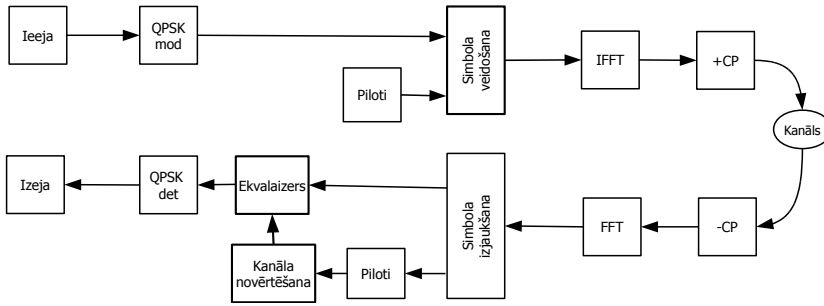
4.2. Pilottoņi

Lai nodrošinātu pastāvīgu kanāla novērtēšanu, derīgās informācijas plūsmā ir jāiesprauž pilot toņus. Pilottoņi ir specifiskas, kompleksas, iepriekš zināmas nolases. Tos var iespraust OFDM simbolos starp derīgās informācijas nolasēm darbojoties gan frekvenču apgabalā gan arī laika apgabalā. Detalizēts apraksts par pilottoņu izvēli ir izklāstīts punktā 3.3..

Izvēlēsimies pilottoņu struktūru, kas ir līdzīga izmantojamajai WLAN standartā 802.11a[3]. Izvietosim nolases "1+0j" sekojošās frekvenču apgabala apakšnesējās: -21,-7,7,21 (6., 20., 34., 48., ja skaita no 1). OFDM simbola struktūra ir redzama 4.1.. attēlā.



4.1. att. Simbola veidošana sakaru sistēmā ar pilottoņiem



4.2. att. Sakaru sistēma ar kanāla noteikšanu lietojot pilot toņus

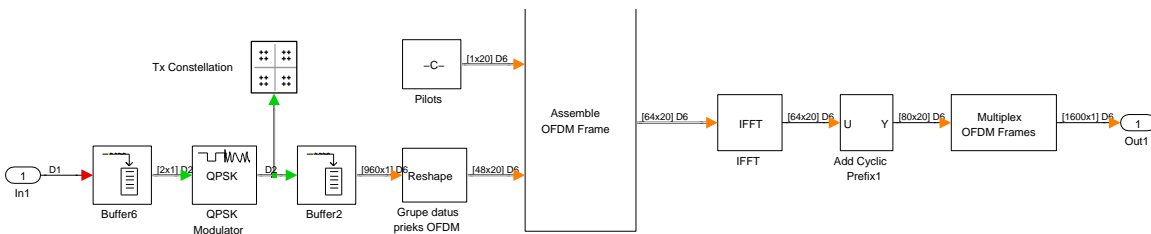
4.3. Sakaru sistēmas blokhēma

Kanāla novērtēšanai izmantojot pilot toņus ir nepieciešama pavisam jauna pieeja. Tāpēc par pamatu ņemsim sakaru sistēmu, ko uzkonstruējam 1.. nodaļā. Mums ir jāpārveido raidītājs tā lai tiktu pārraidīti pilottoņi un uztvērējs tā lai pilottoņi tiktu pareizi atdalīti un novērtēti. Lai pārbaudītu sakaru sistēmas darbību izmantosim frekvenču selektīvu sakaru kanālu, tādu kāds ir aprakstīts punktā 3.7.. Jaunā sakaru sistēmas blokhēma ir dota 4.2.. attēlā.

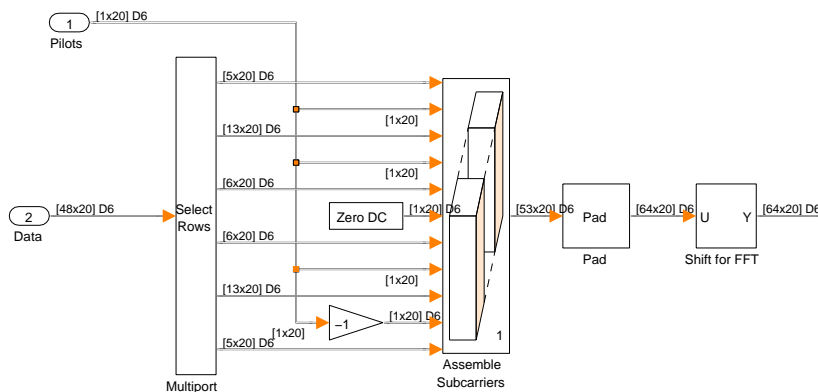
Lai maksimāli vienkāršotu sakaru sistēmu, no tās ir izslēgta kadru veidošana un detektēšana. Pilottoņi tiek pievienoti simbolu līmenī. Uztvērēja daļā parādās simbola izjaukšanas bloks, kas atdala 4 pilot toņus no pārējām 48 derīgās informācijas nolasēm.

4.4. Raidītājs

Par pamatu ņemsim raidītāju, kas aprakstīts punktā 1.2.. Pilottoņu ģeneratoru realizēsim kā vienkāršu "Constant" bloku ar bezgalīgu nolases laiku. Raidītāja modelis ir redzams 4.3.. attēlā. Pilottoņus ievadīsim OFDM simbola 6., 20., 34., un 48. nolasēs (apakšnesējās). Pēdējo pilottoņi invertēsim. Simbolu salikšana bloka uzbūve ir redzama 4.4.. attēlā.



4.3. att. Raidītāja modelis

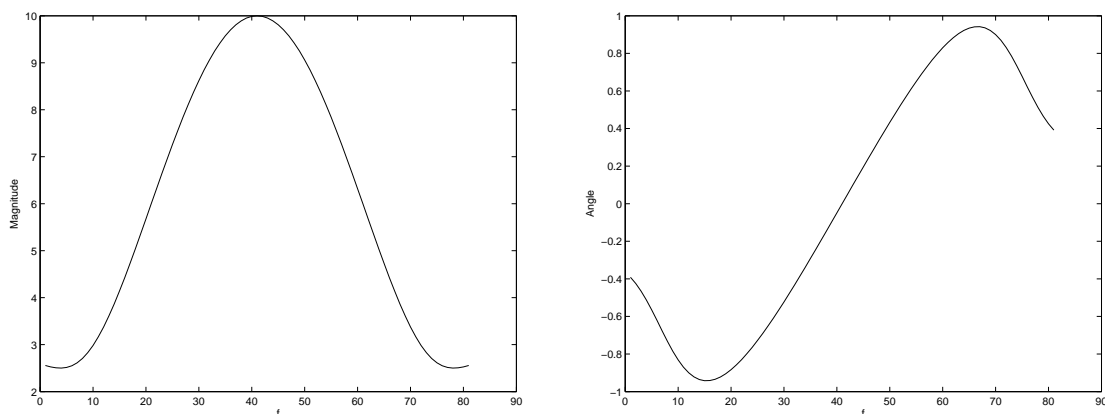


4.4. att. Simbolu veidošanas bloks

uzlabotie algoritmi [4]. Kā alternatīvu mēs varam izmantot interpolāciju no 4 uz 64 un pēc tam nobīdīt rezultātu pa kreisi par 4 pozīcijām. Rezultātā iegūsim aptuvenu interpolāciju, kas dos vismazāko kļūdu intervāla vidū. Jāņem vērā arī tas, ka pirmā pilottoņu nolase atrodas uz 6. apakšnesējas, tādēļ rezultātu būtu jānobīda pa labi par 5 pozīcijām. Nobīdes atņemas un faktiski ir jāievieš nobīde tikai par 1 pozīciju pa labi. Interpolatora modelis ir dots 4.6. attēlā.

4.6. Frekvenču selektīvs sakaru kanāls

Lai notestētu mūsu izveidoto sakaru sistēmu, mums ir nepieciešams frekvenču-selektīvs sakaru kanāls. Par pamatu ņemsim sakaru kanāla modeli no 3.7. punkta. Ņemsim tādu kanāla komplekso pārvades raksturlīkni, kura mainās lēzeni. Tas ir nepieciešams nosacījums šai sakaru sistēmai, jo pilottoņi atrodas tikai uz dažām apakšnesējām. Pieņemsim, ka mūsu kanāla impulsa reakcija ir “1, 2, 3, 4, 0, 0, 0...”. Tā ir pietiekoši īsa un pietiekoši slikta raksturlīkne, lai mēs varētu pārliecināties par ekvalaizera darbību. Pie šādas impulsa reakcijas ir neizbēgama starpsimblu un starpnesēju interference. Izmantojot FFT iegūsim kanāla frekvenču raksturlīkni. Šādas raksturlīknes grafiki ir doti 4.7. attēlā. Frekvenču-selektīva sakaru kanāla modelis bija dots 3.7. attēlā.



4.7. att. Kanāla pārvades raksturlīknes modulis un fāze

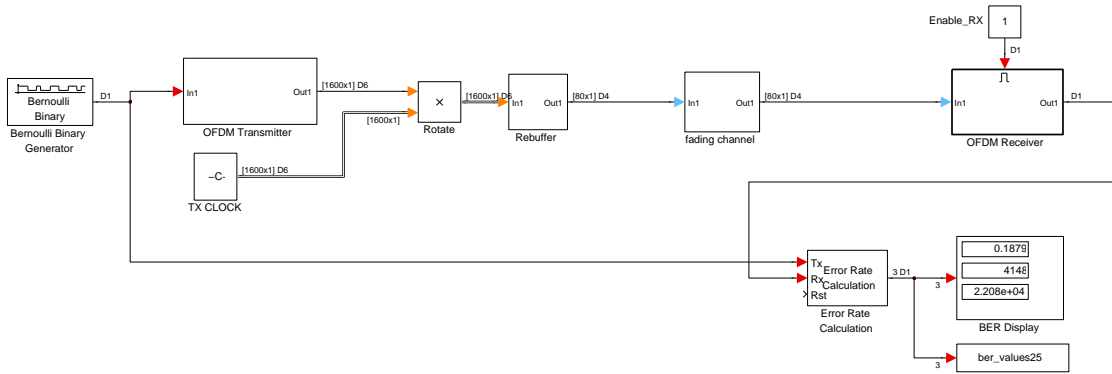
4.7. Sakaru sistēma

Lai notestētu jaunās sakaru sistēmas iespējas, savienosim raidītāju, sakaru kanālu un uztvērēju vienotā shēmā. Sakaru sistēmas modelis ir dots 4.8. attēlā.

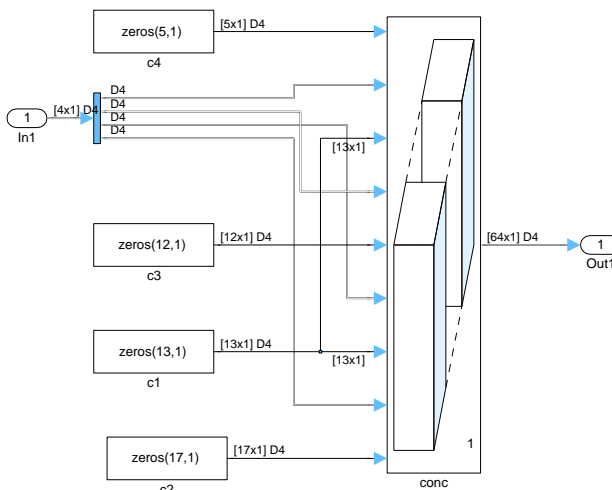
4.8. Simulācijas rezultāti

Pārbaudīsim kā darbojas interpolators. Šim nolūkam interpolatora ieejā ieslēgsim bloku, kas no 4 nolašu pilotsignālu vektora izveidos tādu 64 nolašu vektoru, kur pilotsignāli atradīsies vajadzīgajās pozīcijās (6., 20., 34., un 48. nolasēs), bet pārējās pozīcijās būs 0. Šo signālu arī salīdzināsim ar interpolatora izejas signālu. Pilotsignālu nolasēm būtu jāsakrīt. Šī bloka modelis redzams 4.9. attēlā. Bloka ieeju pieslēgsim interpolatora ieejai, bet izejas signālu izvadīsim uz “Vector scope”. Uz otru “Vector scope” ieeju padosim signālu no interpolatora izejas. Interpolatora ieejas un izejas signālu amplitūdas un fāzes salīdzinājums ir redzams 4.10. attēlā.

Lai pārbaudītu kanāla novērtēšanas un korekcijas shēmas efektivitāti, sākumā atslēgsim kanāla novērtēšanas shēmu un ekvalaizeru. QPSK detektora ieejas zvaigznājs ir redzams 4.11.(A) attēlā. Pieslēdzot kanāla novērtēšanu, zvaigznājs izmainās (skat 4.11.(B)). Kaut arī iegūtais



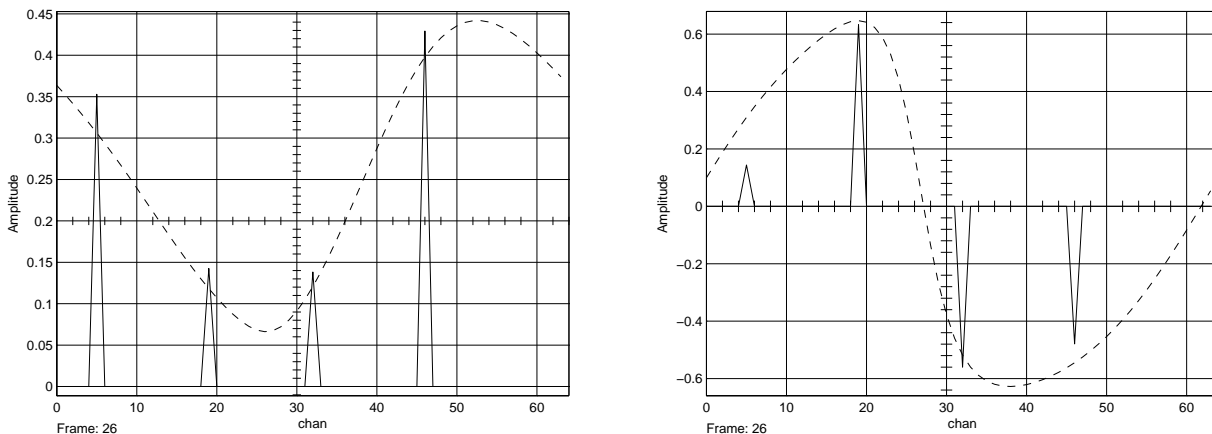
4.8. att. Sistēmas ar kanāla novērtēšanu lietojot pilot toņus Simulink modelis



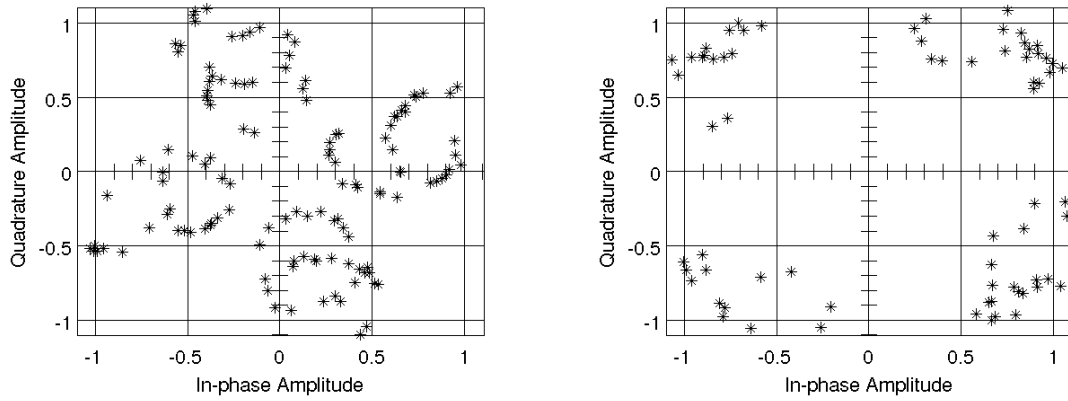
4.9. att. Papildbloks interpolācijas testēšanai

zvaigznājs nav ideāls, signāla svārstības notiek ap attiecīgā kvadranta centru, kas ievērojami uzlabo sakaru sistēmas darbību.

Tagad novērtēsim sistēmas veiktspējas pieaugumu. Šim nolūkam uzņemsim 3 līknes, kas līdzīgi kā iepriekšējā nodaļā, attēlos BER atkarību no attiecības signāls/troksnis (SNR) mūsu frekvenču selektīvajā kanālā ar AWGN. BER atkarība no SNR ir redzama 4.12. attēlā. Redzam, ka parādoties no frekvences atkarīgiem amplitūdas un fāzes kropļojumiem, sakaru sistē-

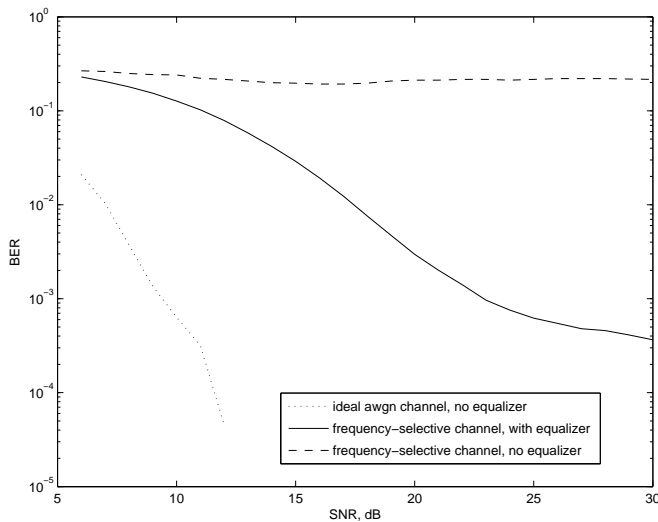


4.10. att.: Interpolatora ieejas un izejas signālu amplitūda(A) un leņķis (B). Izejas signāls parādīts ar raustītu līniju



4.11. att. Zvaigznājs QPSK detektora ieejā bez ekvalaizera (A), ar ekvalaizeru (B)

ma pārstāj darboties ($BER=0.5$). Tas notiek praktiski neatkarīgi no attiecības signāls/troksnis. Kanāla novērtēšana un ekvalaizera ieviešana dramatiski uzlabo situāciju (jau pie $SNR=25dB$ BER samazinās līdz 10^{-4}).



4.12. att.: BER atkarība no SNR sistēmai ar sinhrono kanālu, sistēmai ar kadru sinhronizāciju, sistēmai ar kanāla novērtēšanu.

4.9. Secinājumi un sakaru sistēmas galvenie parametri

Veicot Simulink eksperimentus, varam secināt, ka kanāla novērtēšanas un korekcijas ieviešana dod iespēju strādāt kanālos ar nevienmērīgu amplitūdas-frekvenču un fāzes-frekvenču raksturlīkni. Ja kanāla frekvenču raksturlīkne mainās pietiekoši lēzeni un $SNR = 30dB$, ekvalaizera ieviešana dod iespēju sasniegt $BER = 10^{-4}$.

| Parametrs | Vērtība |
|-------------------------|-----------|
| Datu apakšnesēju skaits | 52 |
| Takts frekvence | 20 kHz |
| Pārraides ātrums | 24 kbit/s |
| SNR pie BER=10e-4 | 25dB |

4.1. tabula Sakaru sistēmas ar kanāla novērtēšanu un korekciju galvenie parametri

5. nodaļa

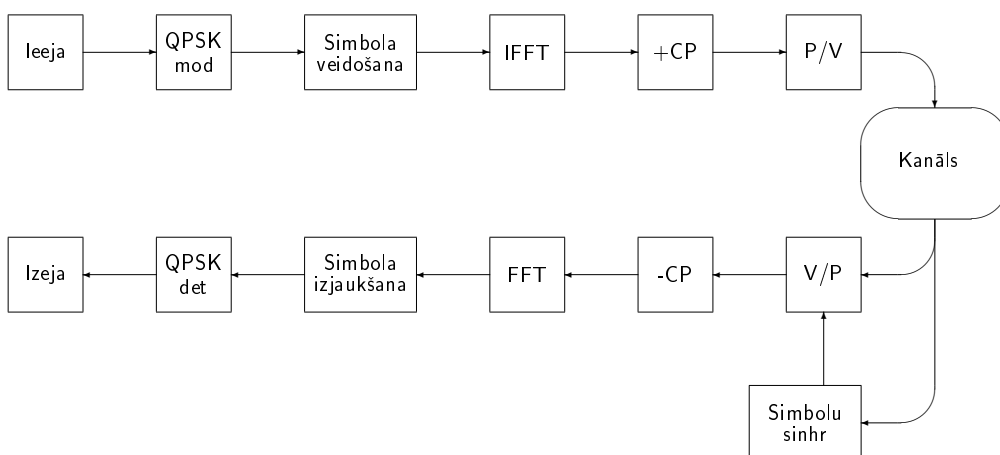
5. Modelis: Sistēma ar virknes datu pārraidi.

5.1. Ievads

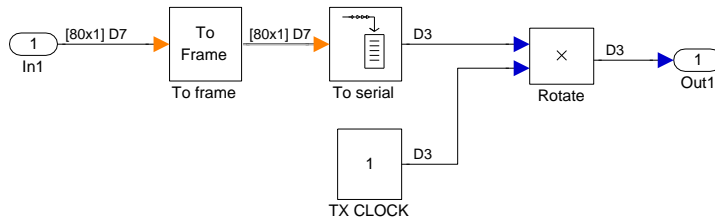
Visi modeļi, kurus mēs esam veidojuši līdz šim, izmantoja sinhrono datu pārraidi, t.i. OFDM laika apgabala simboli no raidītāja uz uztvērēju ceļoja 80-nolašu vektoru veidā. Šajā situācijā simbolu sinhronizācija nebija vajadzīga. Reālā OFDM sakaru sistēmā OFDM laika apgabala simboli tiek pārraidīti virknē. Lai noteiktu kārtējā simbola sākumu ir nepieciešama *simbolu sinhronizācija*.

5.2. Sakaru sistēmas blokshēma

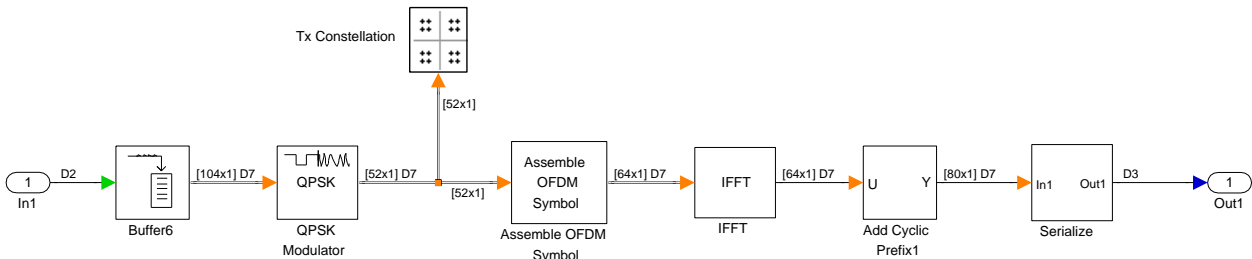
Par pamatu ņemsim blokshēmu no Sakaru sistēmas ar sinhrono kanālu (skat 1. nodaļu). Papildināsim raidītāju ar multiplexeru "MUX", kas veiks OFDM laika apgabala simbolu (80 nolašu vektoru) pārveidošanu par virknēm attiecīgi paaugstinot datu pārraides ātrumu. Bloks uztvērējā - "DMX" veiks pretēju funkciju - demultipleksēšanu, t.i. grupēs skalāras ieejas nolases OFDM simbolos (vektoros pa 80 simboliem). Ņemot vērā, ka signāls uztvērēja ieejā ir nepārtraukts, ir nepieciešams mehānisms kā atrast kārtējā OFDM simbola sākumu. Šo uzdevumu veiks simbolu sinhronizācijas bloks. Sakaru sistēmas blokshēma ir parādīta 5.1. attēlā. Dotā modeļa uzdevums ir simbolu sinhronizācijas testēšana, tādēļ pārējās funkcijas, ko esam izstrādājuši trijās iepriekšējās nodaļās atstāsim uz vēlāku laiku.



5.1. att. Sakaru sistēma ar virknes datu pārraidi



5.2. att. Raidītāja multiplexers (Serialize)



5.3. att. Raidītāja modelis (OFDM Transmitter)

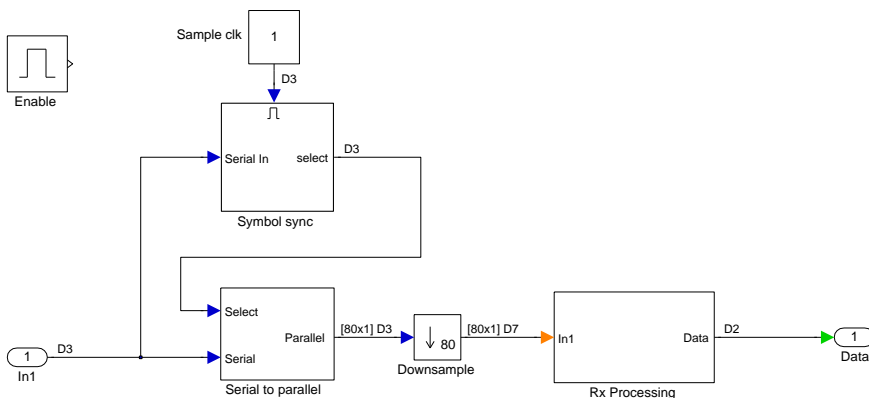
5.3. Raidītāja modelis

Ņemam raidītāja modeli no Sistēmas ar sinhrono kanālu, kas bija parādīts 1.2. attēlā. Pievienojam bloku “Serialize”, kas veiks multiplexēšanu. Tā galvenais elements ir Simulink bloks “Buffer” ar Output Buffer size=1. Šī bloka Simulink shēma ir dota 5.2. attēlā. Jaunā raidītāja Simulink modelis ir dots 5.3. attēlā. Pārējo OFDM raidītāja bloku konfigurācija un saturs paliek tādi paši kādi tie bija aprakstīti 1.2. nodaļā.

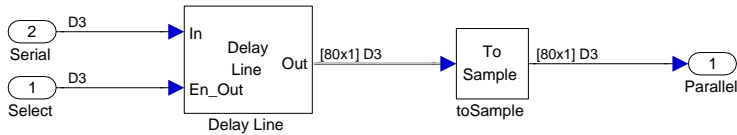
5.4. Uztvērēja modelis

IZveidosim jaunu Simulink subsistēmu ar nosaukumu “OFDM Receiver”. Tā apvienos sevī “vecu” uztvērēja daļu, kas bija aprakstīta 1.3. nodaļā un “jauno”, kas būs saistīta ar simbolu sinhronizāciju. Uztvērēja modelis ir dots 5.4. attēlā. Uztvertās signāla skalārās nolases vienlaicīgi tiek padotas gan uz demultipleksu (Serial tp parallel) gan uz simbolu sinhronizācijas moduli (Symbol sync). Kad simbolu sinhronizācijas bloks detektē simbola sākumu, tas padot strobējošo signālu (select) uz demultipleksu, un tas nomaina izejas vektoru atbilstoši pēdējām 80 nolāsēm, kas ir ienākušas šajā blokā. Demultipleksa uzbūve ir dota 5.5. attēlā. Kā redzam demultipleksers balstās uz aiztures līniju ar atļaujošo ieeju.

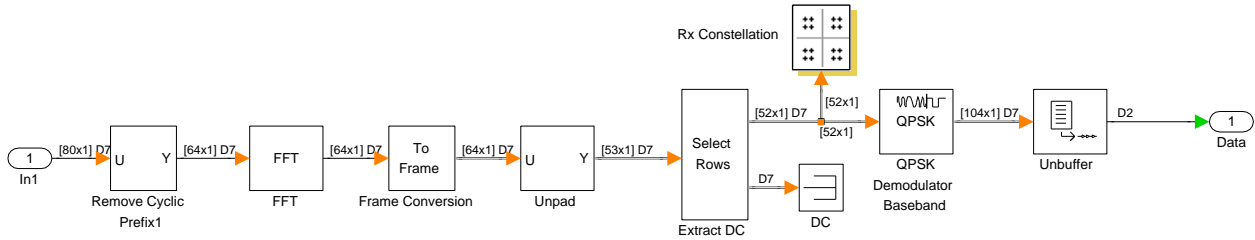
Kad uztvērējs ir nosinhronizējies, demultipleksa izejā periodiski parādās 80 vienādi 80-



5.4. att. Uztvērēja modelis (OFDM Receiver)



5.5. att. Uztvērēja demultipleksers (Serial to parallel)



5.6. att. OFDM Uztvērēja signālapstrādes bloks (Rx Processing)

nolašu vektori. Decimators (Downsample) izvēlas vienu no šiem vektoriem, atkarībā no sinhronizācijas momenta. Principā nav svarīgi kurš no 80 vektoriem tiek paņemts, jo tie ir vienādi. Decimatora izejā mēs iegūstam OFDM signālu paralēlā formā ar attiecīgi samazinātu ātrumu.

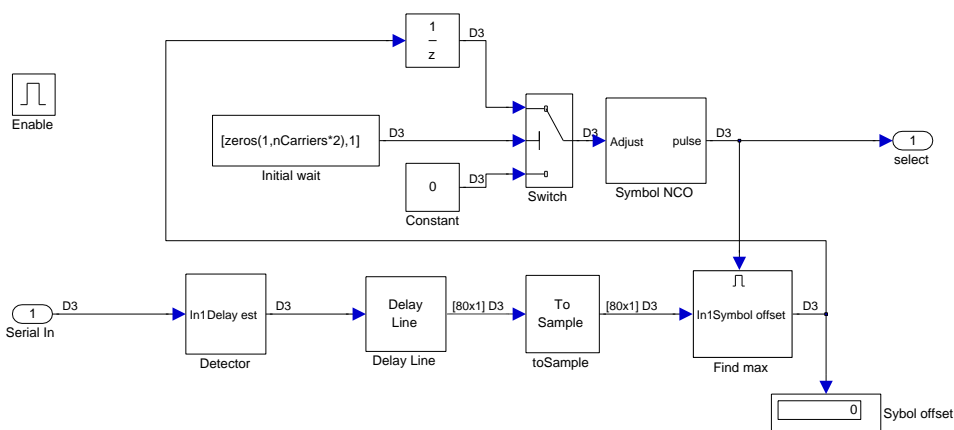
Pēc decimatora signāls nokļūst tajā uztvērēja daļā, kas veic OFDM signālapstrādi (Rx Processing). Faktiski šī daļa mums jau ir labi zināma un ne ar ko neatšķiras no OFDM uztvērēja, kas bija aprakstīts nodaļā 1.3.. Šī bloka modelis ir redzams 5.6. attēlā.

Taktēšanas bloks (Sample clk) nodrošina atbilstošu OFDM uztvērēja nolašu frekvenci. Atļaujas bloks (Enable) ir nepieciešams uztvērēja sinhronizācijas testēšanai, lai uz kādu brīdi izslēgtu uztvērēju izjauktu sinhronizāciju starp uztvērēju un raidītāju.

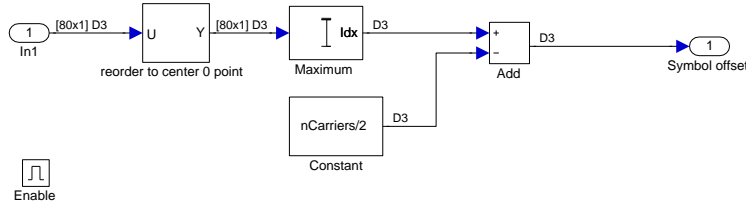
5.5. Simbolu sinhronizācijas bloks

Simbolu sinhronizācijas bloks nodrošina kārtējā OFDM laika apgabala simbola sākuma atrašanu. Tas ir nepieciešams tādēļ, ka lai pareizi veiktu cikliskā prefiksa noņemšanu un FFT. Ja simbola robežas tiks noteiktas nepareizi, starsimbolu interferences dēļ datu pārraide nebūs iespējama. Šī bloka modelis ir redzams 5.7. attēlā. Simbolu sinhronizācijas bloka pamatelementi ir CP detektors (Detector) un skaitliski vadāmais ģenerators (Symbol NCO).

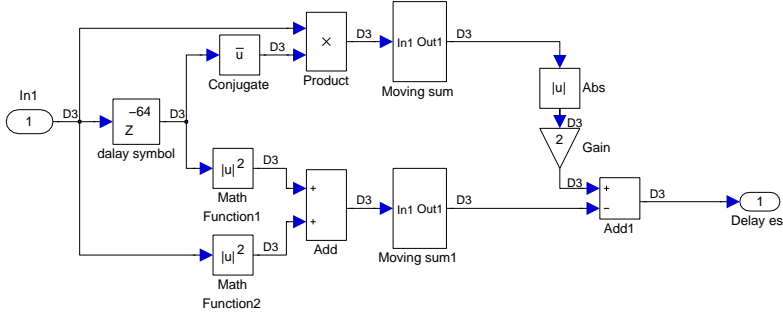
Sinhronizācijas bloks darbojas šādi: CP detektors apstrādā ieejošo nolašu virkni tā, lai mo-



5.7. att. Simbolu sinhronizācijas bloks (Symbol sync)



5.8. att. Maksimuma atrašanas bloks (FindMax)



5.9. att. CP detektors (Detector)

mentos kad tā ieejā ir CP, tā izejā parādītos piķis. CP detektora darbība ir balstīta uz autokorelācijas funkcijas atrašanu un tā detaļās tiks apskatīta nākošajā (5.6.) nodaļā. Ņemot vērā, ka katra OFDM simbola sākumā ir CP, piķi CP detektora izejā atbilst simbolu sākumiem. Bloks “Delay Line” nodrošina skalāras nolašu plūsmas pārveidošanu par 80-nolašu vektoriem. Aiz tā sekojošais Maksimuma noteikšanas bloks (FindMax) nosaka maksimuma nobīdi attiecībā pret pirmo vektora nolasi. Tā uzbūve ir parādīta 5.8. attēlā. Rezultātā šī bloka izejā parādās simbolu nobīde skalārā veidā. Ja mēs šajā vietā pieslēgsim indikatoru (Symbol offset), tas reālā laikā rādīs simbolu sinhronizācijas nobīdi.

Šī nobīde tiek izmantota Skaitliski vadāmā ģenerators (Symbol NCO) stūrēšanai. Pārskatījot šo ģeneratoru, tiek iegūti simbolu sinhroimpulsi pareizos laika momentos. Šie sinhroimpulsi, kā iepriekš jau bija minēts nodaļā 5.4., tiek izmantoti OFDM simbolu atdalīšanai un pārejai no virknes uz paralēlo nolašu plūsmu. Skaitliski vadāmais ģenerators detaļās ir aprakstīts 5.7. nodaļā. Papildbloks $\left[\frac{1}{z}\right]$ nepieciešams algebraiskās cilpas novēršanai. Slēdzis (Switch), nulļu ģenerators (Wait), konstante (Constant) ir nepieciešami lai aizliegtu ģenerators pārskatīšanu uzreiz pēc modeļa palaišanas.

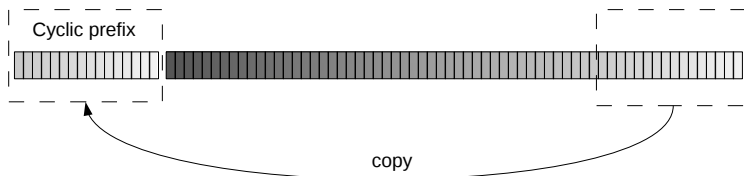
5.6. Cikliskā prefiksa detektors

Cikliskā prefiksa (CP) detektora darbības teorētiskais pamatojums ir atrodams publikācijā [5]. Detektora pamatā ir autokorelators, kas korelē signālu un tā kopiju, kas aizturēta par 64 nolasēm. Autokorelators izmanto *maksimālās līdzības (Maximum likelihood)* algoritmu. Detektora darbībai ir nepieciešami 2 novērtējumi: a)parastais (autokorelācija):

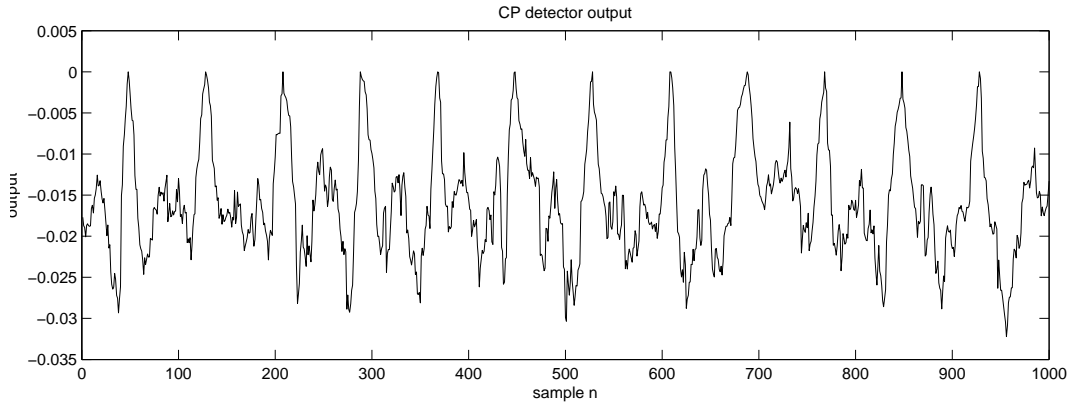
$$y(m) = \sum_{k=m}^{m+L-1} r(k)r^*(k+N), \quad m \in \{0, \dots, \theta, \dots, N+L-1\} \quad (5.1)$$

b)kvadrātiskais (kvadrāta autokorelācija)

$$u(m) = \frac{1}{2} \sum_{k=m}^{m+L-1} |r(k)|^2 + |r(k+N)|^2 \quad (5.2)$$



5.10. att. OFDM simbola cikliskais prefikss



5.11. att. CP detektora izejas signāls

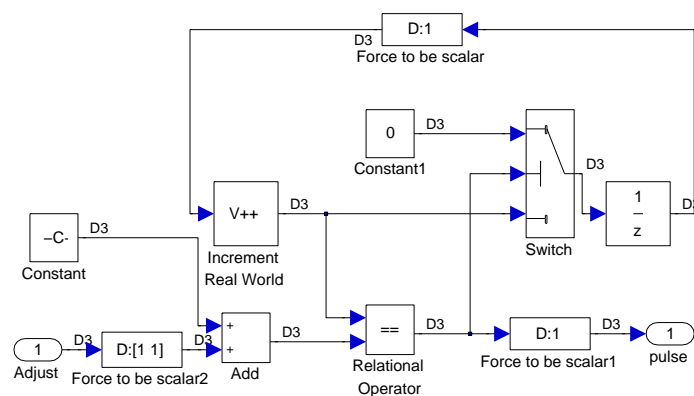
kur r ir ieejas signāla nolases vērtība, N ir OFDM lietderīgā simbola garums (64), L ir CP garums (16). Simbola aizturi var atrast izmantojot sakarību:

$$\hat{\theta}_{ML} = \arg \max_{\theta} \{|y(\theta)| - \rho u(\theta)\} \quad (5.3)$$

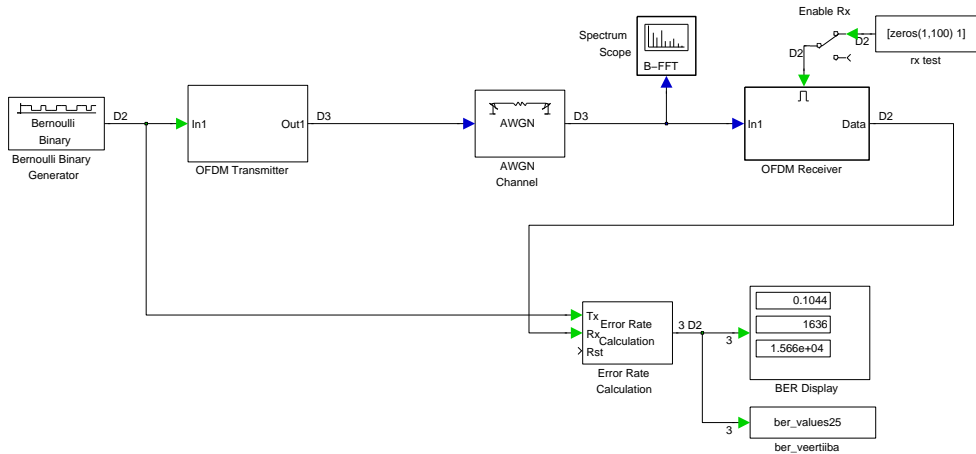
Ņemot vērā OFDM simbola ciklisko dabu (OFDM simbola laika apgabala simbola uzbūve ir dota 5.10. attēlā.), ir pierādīts, ka šāds novērtēšanas mehānisms ir optimāls. CP detektora shēma, kas ir dota 5.9. attēlā, ir izveidota izmantojot augstāk dotās matemātiskās sakarības. CP detektora izejas signāla grafiks ir dots 5.11. attēlā.

5.7. Skaitliski vadāmais ģenerators

Skaitliski vadāmais ģenerators (NCO) izstrādā simbolu sinhronizācijas impulsus ar pareizu frekvenci un pareizā fāzē. Šie sinhronizācijas impulsi nodrošina OFDM uztvērēja virknes-paralēlo pārveidojumu. NCO pamatelementi ir Inkrementa bloks (Increment Real World),



5.12. att. Skaitliski vadāmais ģenerators (Symbol NCO)



5.13. att. Sakaru sistēmas ar virknes datu pārraidi modelis

aizture $\left[\frac{1}{z}\right]$, un salīdzināšanas bloks (Relational Operator). Cilpa Inkrements - Aizture nodrošina periodisku vērtības pieaugumu. Salīdzināšanas bloks nepārtraukti salīdzina šo vērtību ar summu starp skaitli pieskaņošanas ieejā (Adjust) un normālo simbola garumu (80), t.i. Salīdzināšanas bloka otrajā ieejā pienāk informācija par nepieciešamo nākamā simbola garumu. Kad nepieciešamais garums ir sasniegts, Salīdzināšanas bloks izdod impulsu ar kuru vienlaicīgi, izmantojot slēdzi (Switch), nomet arī Inkrementa ieejas vērtību. Tālāk process sākas no jauna.

5.8. Sakaru sistēma

Visas sakaru sistēmas modelis ir redzams 5.13. attēlā. Šeit varam izmantot tipisku slēgumu, kur raidītājs ar uztvērēju ir savienots caur AWGN sakaru kanāla modeli no Simulink standarta bibliotēkas. Papildus esam ielikuši arī spektra analizatora bloku, ar kuru varēsim pārlicināties, ka OFDM signāla spektrs ir ierobežots apmēram ar pusi no takts frekvences (mūsu modelī 20kHz). Bloks $[zeros(1,100) 1]$ aiztur uztvērēja ieslēgšanos simulācijas sākumā, tādējādi panākot tā sākotnējo desinhronizāciju. Ar slēdzi “Enable Rx” mēs varam arī manuāli ieslēgt un izslēgt uztvērēju lai pārbaudītu sinhronizācijas darbību.

5.9. Simulācijas rezultāti

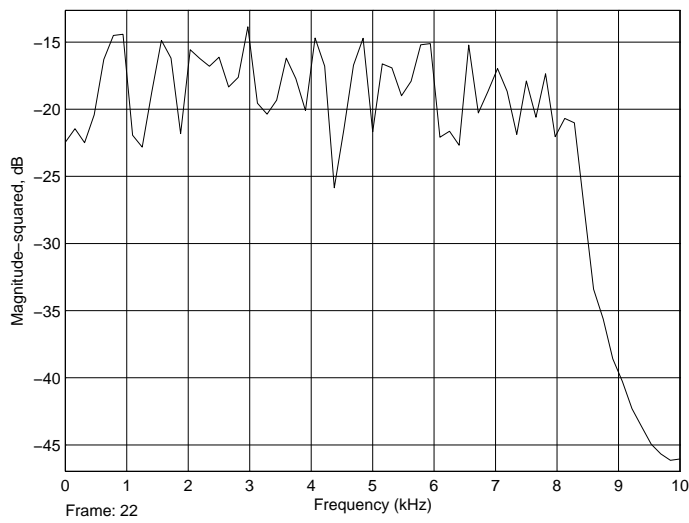
OFDM uztvērēja signāla spektrs ir parādīts 5.14. attēlā. Redzam, ka spektrs ir ierobežots apmēram ar pusi no čipu takts frekvences.

Simbolu sinhronizāciju OFDM uztvērējā raksturojošās laika diagrammas ir redzamas 5.15. attēlā. Augšējais attēls parāda OFDM signālu uztvērēja ieejā. Nākošais attēls parāda atbilstošu mums jau pazīstamo CP detektora izejas signālu. Trešā aile parāda simbolu sinhronizācijas impulsus skaitliski vadāmā ģeneratora (NCO) izejā. Tie norāda simbolu sākumus virknes-paralēlajam pārveidotājam. Apakšējais attēls parāda signālu NCO ieejā, t.i. simbolu sinhronizācijas kļūdu.

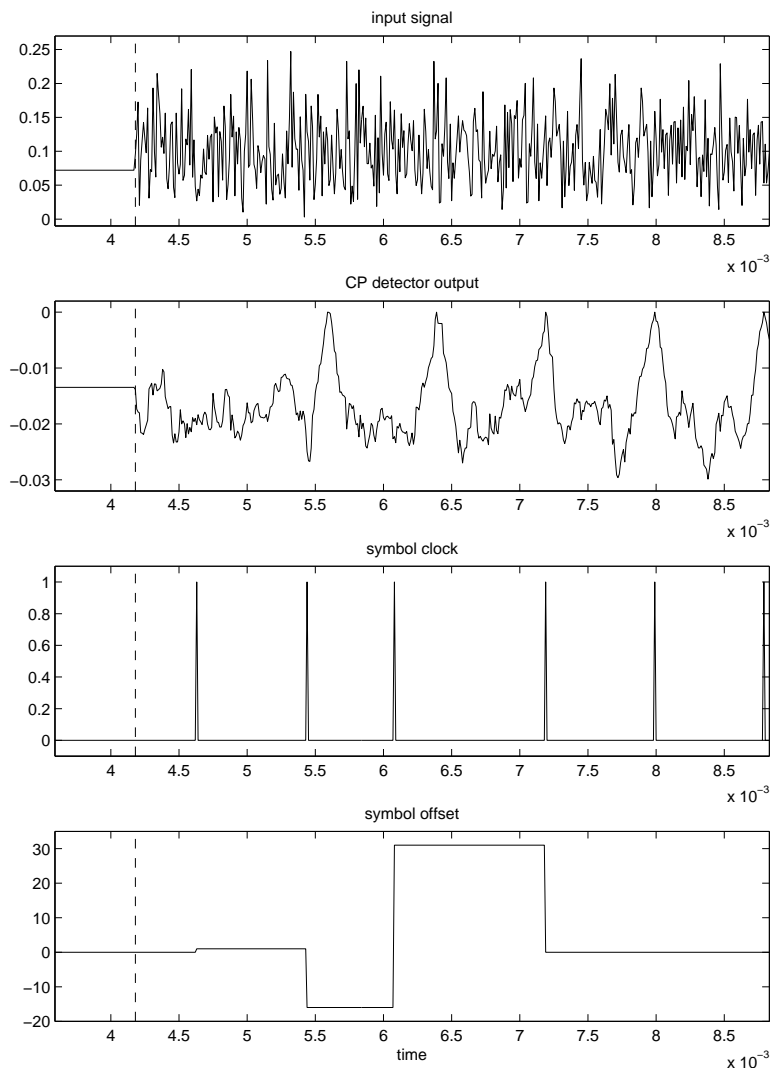
Novērtēsim sakaru sistēmas Bitu kļūdu intensitāti atkarībā no trokšņa līmeņa sakaru kanālā. Šim nolūkam izmantosim Matlab programmu, kas automātiski mainīs attiecību Signāls/Troksnis (SNR) sakaru kanālā un palaidīs simulāciju. Simulācijas beigās iegūtais BER tiks saglabāts mainīgajā. Šī testa veikšanai nepieciešams arī papildbloks raidītāja izejā, kas mēra signāla jaudu. Ierobežotas vietas dēļ augstākminētās programmas un bloku apraksts netiks dots. Iegūtais Bitu kļūdas intensitātes atkarības no Signāls/troksnis ir dots 5.16. attēlā. Salīdzinājumam šajā attēlā ir dots BER sistēmai ideālu simbolu sinhronizāciju, t.i. sistēmu,

kur simbolu pozīcijas uztvērējam jau iepriekš ir zināmas. Kļūdas šajā gadījumā izraisa tikai QPSK detektors.

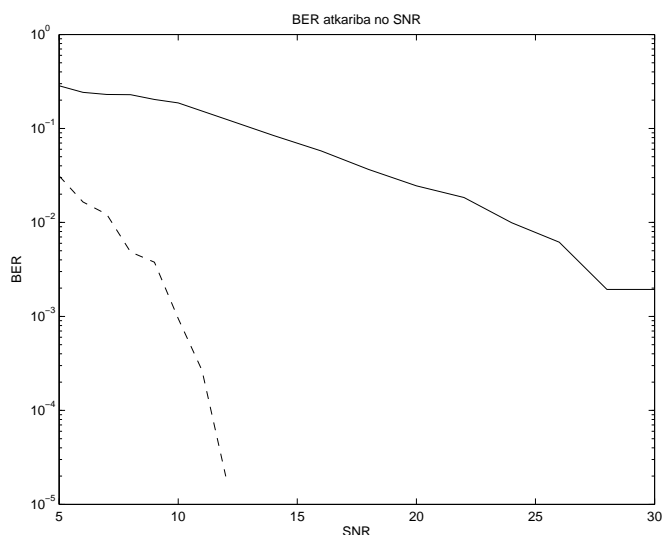
BER atkarībā no nobīdes



5.14. att. OFDM signāla Furjē spektrs



5.15. att.: Simbolu sinhronizācijas procesa laika diagrammas. Ar raustītu līniju ir parādīts OFDM uztvērēja ieslēgšanas brīdis.



5.16. att.: BER atkarība no SNR sistēmai ar virknes datu pārraidi. Ar raustītu līniju parādīta ideāli sinhronizētas sakaru sistēmas veiktspēja.

5.10. Secinājumi un sistēmas galvenie parametri

Pateicoties cikliskajam prefiksam ir iespējams noteikt OFDM simbolu pozīcijas ieejas virknes signāla plūsmā. Izmantojot augstāk aprakstīto sinhronizācijas shēmu, lai sinhronizētos ir nepieciešami 2-4 simbolu periodi. Ņemot vērā, ka katra simbola lietderīgā slodze ir 104 biti, viena simbolu sinhronizācijas kļūda noved pie 200-400 bitu nepareizas uztveršanas. Šāds kļūdu skaits nedod iespēju sasniegt BER mazāku par 10^{-3} .

| Parametrs | Vērtība |
|-------------------------|-----------|
| Datu apakšnesēju skaits | 52 |
| Takts frekvence | 20 kHz |
| Pārraides ātrums | 26 kbit/s |
| SNR pie BER=6.5e-3 | 25dB |

5.1. tabula Sakaru sistēmas ar virknes datu pārraidi galvenie parametri

5.11. Sistēmas uzlabošana

Lai mazinātu trokšņa ietekmi uz sinhronizāciju, simbolu nobīdes korekcijas cilpu vajadzētu padarīt "inertu". Simbolu nobīdes lielumu, kas pienāk no CP detektora uz NCO varētu padot caur kustīgās summas (moving sum) bloku, kas darbotos kā integrātors un bremsētu straujas lieluma izmaiņas.

6. nodaļa

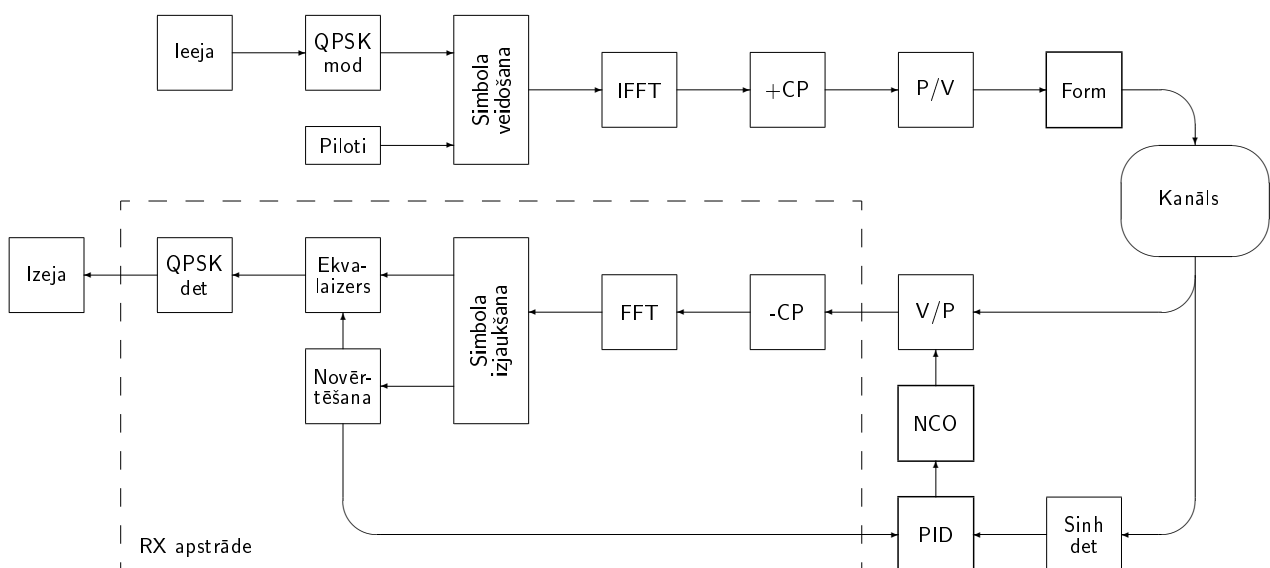
6.Modelis: Sistēma ar diskretizāciju.

6.1. Ievads

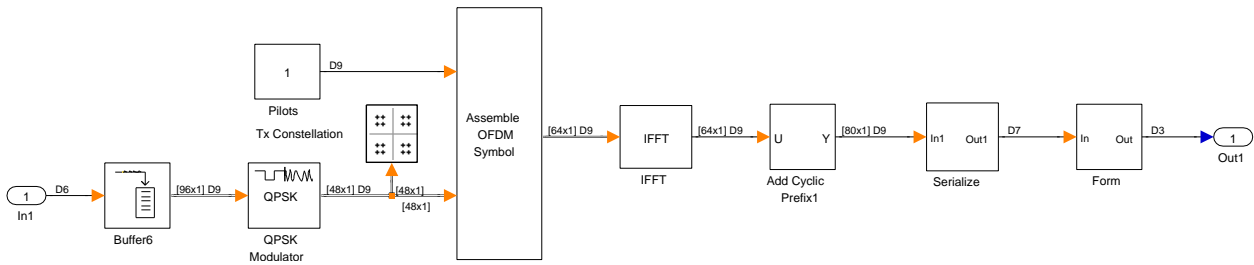
Iepriekšējais sakaru sistēmas modelis nodrošināja virknes datu pārraidi pie nosacījuma, ka dati no raidītāja uz uztvērēju ceļo sinhronu, nedalāmu vienību (čipu) veidā. Reāla uztvērēja ieejā signāls ir analogs un ir jāveic signāla diskretizāciju. Lai sekmīgi veiktu diskretizāciju, ir nepieciešamas ķēdes, kas regulē diskretizācijas fāzi un frekvenci. Zemāk aprakstītais modelis nodrošina precīzu diskretizāciju un simbolu sinhronizāciju, izmantojot jau zināmās simbolu sinhronizācijas metodes, kas ir papildinātas ar jauniem, augstas precizitātes sinhronizācijas mehānismiem. Dotais modelis satur arī kanāla novērtēšanu un korekciju, jo kanāla novērtējuma informācija ir nepieciešama precīzas simbolu sinhronizācijas nodrošināšanai.

6.2. Sakaru sistēmas blokhēma

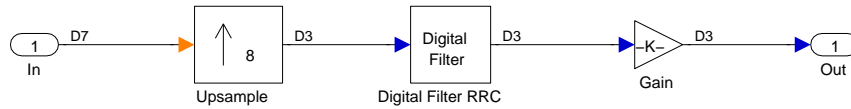
Par pamatu mūsu jaunajai sakaru sistēmai ņemsim divus iepriekšējos modeļus: sistēmu ar kanāla novērtēšanu un korekciju lietojot pilot toņus (skat 4. nodaļu) un sistēmu ar virknes datu pārraidi (skat 5. nodaļu). Ar treknu līniju ir izdalīti bloki, kuri būs jāizstrādā no jauna. OFDM sakaru sistēmas raidītāja daļā parādās impulsu formētājs kas nodrošinās uz 8x augstāku diskretizācijas frekvenci tādējādi imitējot ciparu-analogo pārveidošanu. Savukārt uztvērēja daļa iegūst proporcionāli-integrālo-diferenciālo vadības ķēdi (PID) kura vada ģeneratoru ar skaitliski vadāmu frekvenci (NCO).



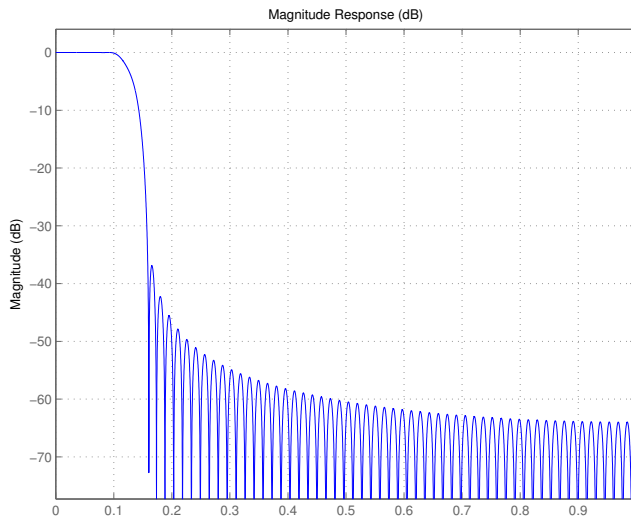
6.1. att. Sakaru sistēma ar diskretizāciju



6.2. att. Raidītāja modelis (OFDM Transmitter)



6.3. att. Impulsu formētājs (Form)



6.4. att. Zemo frekvenču filtra raksturlīkne

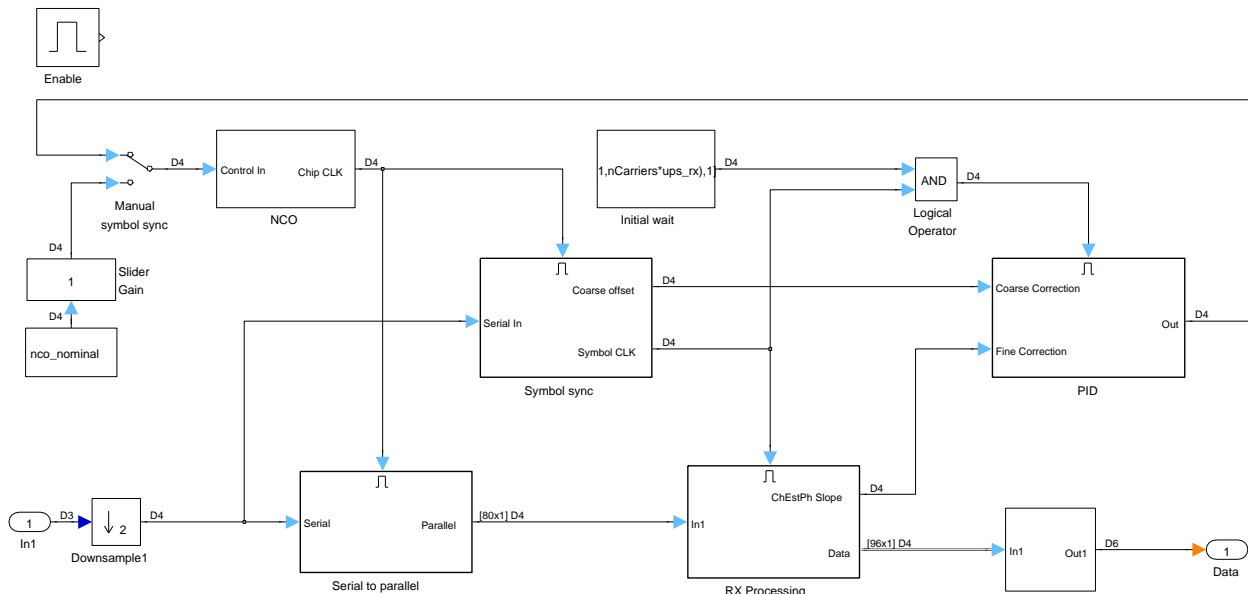
6.3. Raidītāja modelis

Par pamatu izmantosim raidītāja modeli no sistēmas ar kanāla novērtēšanu un korekciju lietojot pilot toņus (skat paragrāfu 4.4.). To papildināsim ar paralēlo-virknes pārveidotāju no 5.3. paragrāfa (skat 5.2. attēlu), saglabājot visus bloku konfigurācijas parametrus. Ciparu-analogā pārveidojuma simulācijai pievienosim impulsu formētāju (Form) (skat 6.3. attēlu). Impulsu formētājs sastāv no bloka, kas palielina diskretizācijas frekvenci 8 reizes (upsample) (līdz 160kHz) un zemo frekvenču filtra (Digital Filter RRC), kurš novāc augstākās signāla harmonikas. Filtra raksturlīkne ir parādīta 6.4. attēlā.

6.4. Uztvērēja modelis

Aplūkojot sakaru sistēmas blokshēmu 6.1. attēlā, redzam, ka uztvērējs būs jāpapildina ar vairākiem blokiem. Arī šoreiz par pamatu izmantosim uztvērēja modeli no sistēmas ar kanāla novērtēšanu un korekciju lietojot pilot toņus (skat paragrāfu 4.4.). Visa OFDM uztvērēja modelis ir parādīts 6.5. attēlā. Viss uztvērējs ir sadalīts 5 lielos blokos, kuriem ir pieslēgti porti un daži papildelementi. Lielākā daļa bloku darbojas uz 4x augstākās takts frekvences (80kHz) salīdzinot ar čipu frekvenci ($D7 = 20\text{kHz}$) Signāla plūsma ir sekojoša:

- Saņemtais analogais signāls pienāk uz portu 'In1'. Saņemtās nolases tiek izretinātas par 50% (downsample 2x) pārejot uz 2x zemāku diskretizācijas frekvenci (80kHz) un padotas

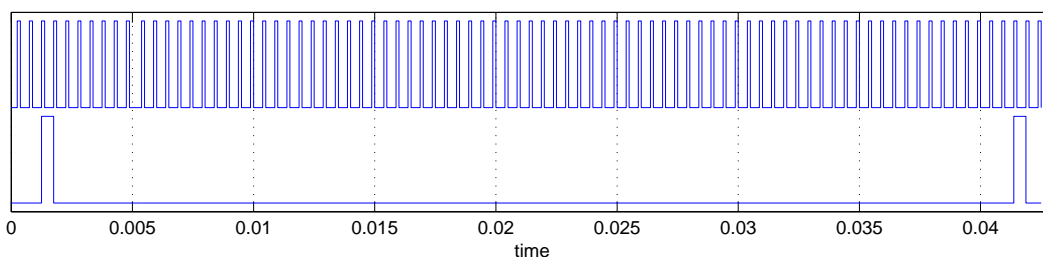


6.5. att. Uztvērēja modelis

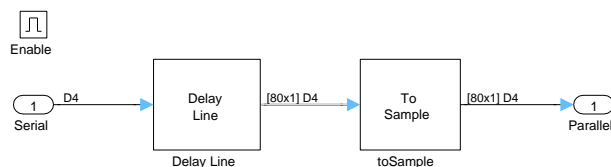
vienlaicīgi gan uz virknes - paralēlo pārveidotāju gan uz simbolu sinhronizācijas detektoru.

- Virknes - paralēlais pārveidotājs (Serial to parallel), kurš tiek stūrēts no skaitliski vadāmā ģeneratora (NCO), vienā no 4 taktīm veic signāla nolasi un nomaina (nobīda) izejas vektoru.
- Tajā pašā laikā simbolu sinhronizācijas bloks (Symbol sync) korelē un “fotografē” ieejošo datu plūsmu laika momentos, kad no NCO ir pienākuši 80 impulsi. Šajos brīžos tiek ģenerēts ‘Simbola sinhroimpulss’ (Symbol CLK), kas vada tālāko uztveršanas gaitu. Analizējot “fotogrāfijas”, simbolu sinhronizācijas bloks nosaka nobīdi starp OFDM simbolu sākumiem un takts impulsiem.
- Konstatētā (rupjā) simbolu nobīde pienāk uz proporcionāli-integrāli-diferencējošo vadības bloku (PID), kas koriģē skaitliski vadāmo ģeneratoru, tādējādi noslēdzot automātiskās pieskaņošanas cilpu. Precīzā simbolu nobīde tiek iegūta kanāla novērtējuma fāzes atvasinājumu.
- Simbolu sinhronizācijas impulsi periodiski ieslēdz uztvērēja apstrādes bloku (RX Processing), kas arī veic visu nepieciešamo signālapstrādi un informācijas nodošanu lietotājam.

OFDM uztvērēja takts signāli ir doti 6.6. attēlā.



6.6. att.: OFDM uztvērēja takts signāli: čipu takts signāls (augšā), simbolu takts signāls (apakšā)



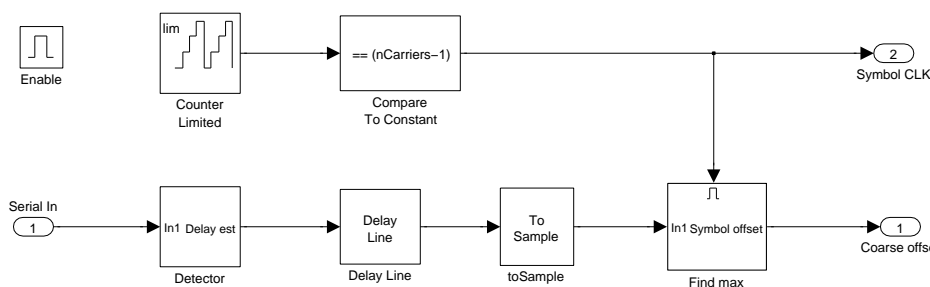
6.7. att. Virknes paralēlais pārveidotājs (Serial to parallel)

6.5. Virknes - paralēlais pārveidotājs

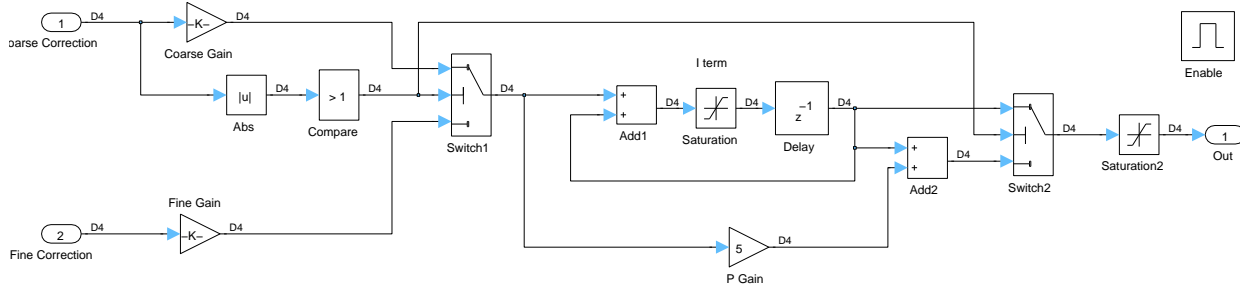
Virknes paralēlais pārveidotājs ir uzbūvēts uz aiztures līnijas (Delay line) pamata. Kad pienāk strobējošais '1' vienā no 4 taktīm, pārveidotājs nolasa tekošo ieejas vērtību un novieto to izejas vektora 1. nolāsē, pārbīdot jau esošās nolases uz leju un pazaudējot pašu vecāko vērtību. Tādējādi mūsu sistēmas virknes - paralēlā pārveidotāja izejas vektors vienmēr satur pēdējās 80 nolases. Virknes paralēlā pārveidotāja shēma ir dota 6.7.

6.6. Simbolu sinhronizācijas bloks

Simbolu sinhronizācijas bloka uzdevums ir simbolu robežu atrašana, simbolu sinhroimpulsu ģenerācija, kā arī simbolu robežu un sinhroimpulsu savstarpējās nobīdes mērīšana. Iepriekšējā nodaļā mēs jau apskatījām kādu simbolu sinhronizācijas bloku (skat paragrāfu 5.5.). Minētais modelis sastāvēja no divām daļām: cikliskā prefiksa (CP) detektora un NCO ar vadāmu fāzi. Mūsu jaunajā sinhronizācijas detektorā saglabāsim minētā modeļa detektējošo daļu, bet NCO nomainīsim uz vienkāršu impulsu skaitītāju. Jaunā simbolu sinhronizācijas bloka modelis ir parādīts 6.8. attēlā. Detektors (Detector) un maksimuma atrašanas (Find max) bloki paliek bez izmaiņām, to uzbūvi un darbības principus var apskatīt attiecīgi paragrāfos 5.6. un 5.5.. Uz simbolu sinhronizācijas bloka atļaujošo portu (Enable) pienāk čipu takts impulsi no NCO (tiks aprakstīts paragrāfā 6.8.) ar pamata takts frekvenci 80kHz. Kad ir iestājusies simbolu sinhronizācija, šie impulsi pienāk ar aizpildījumu 0.25, t.i. vieninieks ir katrs ceturtais impulss, un simbolu sinhronizācijas bloks ieslēdzas katru ceturto takti. Impulsu skaitītājs (Counter limited) skaita ieslēgšanas momentus un pakāpeniski palielina izejas lielumu. Kad skaitītājs ir saskaitījis līdz 79, nostrādā salīdzināšanas bloks (Compare To Constant) un ģenerē kārtējo OFDM simbola sinhroimpulsu, kas ir nepieciešams OFDM uztvērēja darbībai. Maksimuma atrašanas bloks (Find max) mēra nobīdi starp simbola sinhroimpulsu un pīķi, kas pienāk no CP detektora. No nulles atšķirīga nobīde netieši izmaina OFDM uztvērēja NCO parametrus (par to nākošajā paragrāfā), kā rezultātā čipu takts impulsu aizpildījums uz atļaujošā porta (Enable) var būt kļūt mazāks vai lielāks par 0.25 (0, 0.33, 0.5, 1), tādējādi nodrošinot simbolu sinhroimpulsu 'slidēšanu' attiecībā pret uztvertu OFDM simbolu stāvokli.



6.8. att. Simbolu sinhronizācijas detektors (Symbol sync)



6.9. att. Proporcioniāli-integrāli-diferenciālais vadības bloks (PID)

6.7. Laika proporcioniāli-integrāli-diferenciālais vadības bloks

PID vadības bloks ir automātiskās vadības sistēma (*control system, eng*) ar atgriezenisko saiti, kas nodrošina nezināma vadāmā procesa automātisku parametru pieskaņošanu tā, lai panāktu nepieciešamo izejas signāla lielumu. Regulēšana balstās uz principu, kur tiek mērīta atšķirība starp nepieciešamo procesa izejas lielumu un reālo procesa izejas lielumu, un šī atšķirība tiek reizināta / integrēta / diferencēta un padota atpakaļ uz procesa regulējošo ieeju. Mūsu gadījumā vadāmais process ir skaitliski vadāmais ģenerators (NCO) (skat 6.8. paragrāfu). PID bloks strādās 2 režīmos:

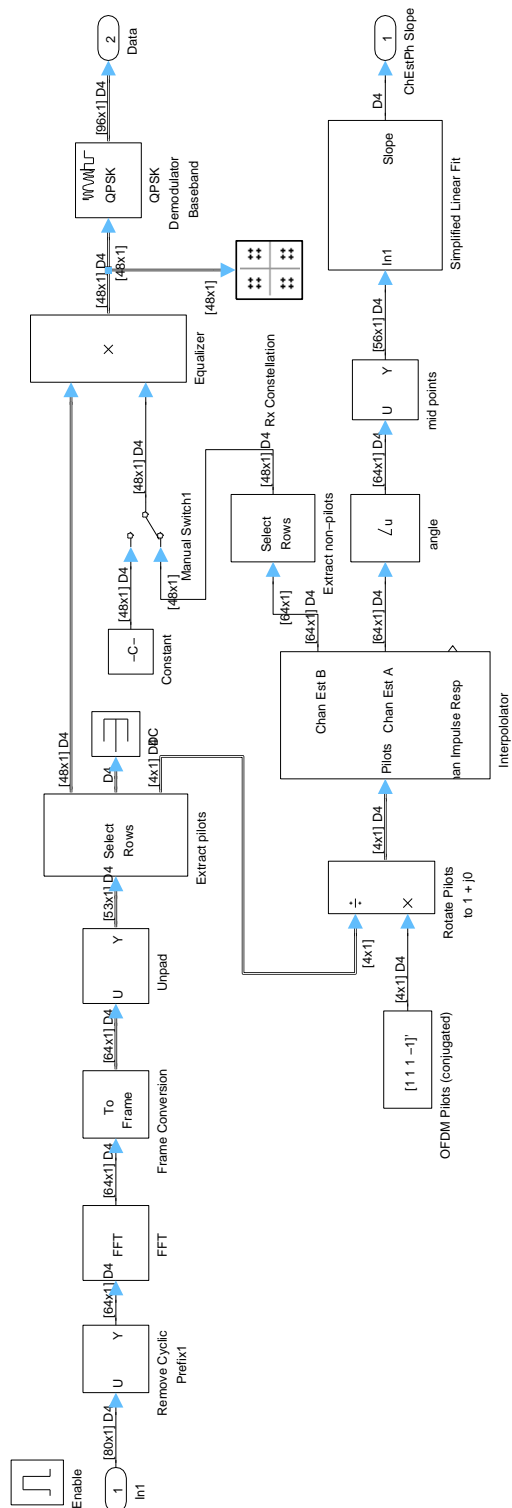
- Rupjās pieskaņošanas režīms: vadības signāls (kļūda) ir simbolu sinhroimpulsu nobīde. Šajā režīmā tiek ņemts vērā tikai kļūdas signāla integrālis, un mūsu sistēma strādā kā I vadības bloks.
- Precīzās pieskaņošanas režīms: vadības signāls (kļūda) ir kanāla novērtējuma fāzes atvasinājums. Tiek ņemts vērā gan kļūdas signāls gan tā integrālis un visa sistēma strādā kā PI vadības bloks.

Pārslēgšanos starp režīmiem nodrošina slēdži 'Switch1' un 'Switch2', kurus vada komparators 'Compare', kurš ieslēdz rupjās pieskaņošanas režīmu, ja simbolu sinhroimpulsu nobīde ir lielāka par 1. Bloka integrālais zars sastāv no blokiem 'Add1', 'Saturation' un 'Delay', savukārt bloka proporcioniālais zars sastāv no pastiprinātāja 'P Gain'. 'Saturation2' bloks ierobežo izejas vērtības diapazonā no 0.2482 līdz 0.2517, tādējādi nodrošinot stabilu NCO darbību.

Jebkurš PID vadības bloks prasa rūpīgu noskaņošanu. Variējot pastiprināšanas koeficientus var vērot dažādu sistēmas uzvedību - nejutību, ierosināšanos, nepareizu darbību. Piedāvātie pastiprinājuma koeficienti tika atrasti eksperimentāli, un ir pilnīgi iespējams, ka tie nav pilnīgi optimāli. Šo koeficientu vērtības ir dotas 6.1. tabulā.

| Bloks | Parametrs | Lielums |
|-------------|-------------|-------------|
| Coarse Gain | Gain | -6.2500e-06 |
| Fine Gain | Gain | -0.0025 |
| Saturation | Lower Limit | 0.2487 |
| | Upper Limit | 0.2512 |
| Saturation2 | Lower Limit | 0.2482 |
| | Upper Limit | 0.2517 |

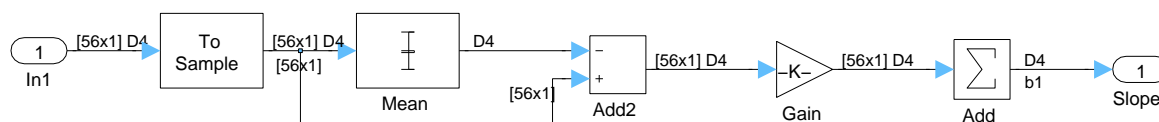
6.1. tabula Kritiskie bloka parametri



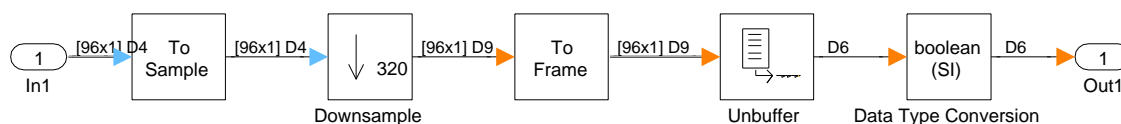
6.12. att. Uztvērēja apstrādes bloks (RX Processing).

sinājuma iegūšanai. Šim nolūkam kanāla novērtējuma informāciju nolasīsim no interpolatora (Interpolator) porta 'Chan Est A', kurš atšķirībā no porta 'Chan Est B' satur nobīdītu novērtējumu (kāpēc tas ir nobīdīts, skat paragrāfu 4.5.1.). Pēc kompleksā argumenta fāzes atdalīšanas blokā 'angle', ar bloka 'mid points' tiek atņemtas 5 vērtības no katra gala. Fāzes novērtējuma atvasinājums tiek atrasts ar vienkāršotas lineārās pielaiķošanas (Simplified linear fit) bloka palīdzību, kurš izmanto mazāko kvadrātu metodi. Šī bloka shēma ir parādīta 6.13. attēlā.

Fāzes novērtējuma atvasinājums ir nepieciešams, lai nodrošinātu precīzu simbolu sinhronizāciju un pareizu diskretizācijas momentu izvēli. Kad uztvērējs ir tuvu sinhronismam, pilottoņu fāzes kļūst lineāri augošas vai dilstošas. Momentā, kas uztvērējs ir pilnīgi nosinhronizējies, visu



6.13. att. Vienkāršotās lineārās pielaikošanas bloks (Simplified linear fit)



6.14. att. Vienkāršotās lineārās pielaikošanas bloks (Simplified linear fit)

pilottoņu fāzes ir vienādas ar 0.

6.10. Paralēlais-virknes pārveidotājs uztvērēja izejā

Ņemot vērā, ka OFDM uztvērējā apstrāde notiek pa blokiem (simboliem), pēc QPSK demodulācijas, apstrādes bloka izejā mēs iegūstam $[96 \times 1]$ izmēra matricas. Turklāt šīs matricas pienāk ar $80 \times 4 = 320$ reižu atkārtosanos, kura ir nepieciešama sinhronizācijas mehānismu darbībai. Papildus bloks 'Serialize' (skat 6.14. attēlu) nodrošina nolašu ātruma samazināšanu (Downsample) kā arī paralēlo-virnes pārveidojumu. Šī bloka izejā dati parādās ar tādu pašu kā ti pienāca sakaru sistēmas raidītāja ieejā.

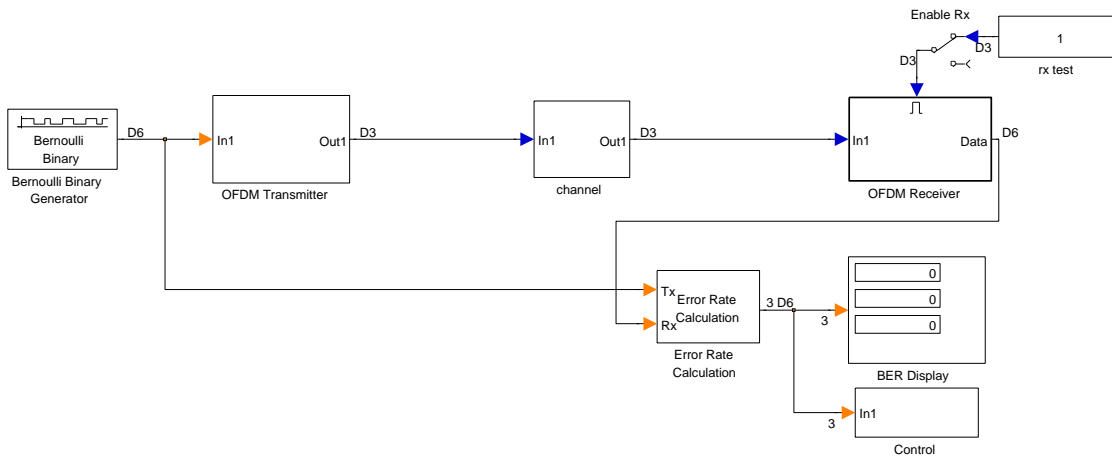
6.11. Sakaru sistēma

Visas sakaru sistēmas modelis ir dots 6.15. attēlā. Tas ir tipisks slēgums no raidītāja, sakaru kanāla un uztvērēja. Tāpat kā iepriekšējos modeļos, slēdzis 'Enable Rx' ir nepieciešams lai notestētu uztvērēja sinhronizāciju brīvi izvēlētā laika momentā. Atšķirībā no citiem modeļiem, šajā attēlā ir redzams Simulink bloks 'Find Delay', kas ir ļoti noderīgs sakaru sistēmas aiztures novērtēšanai. Turklāt, šī aizture mums ir nepieciešama blokā 'Error Rate Calculation'.

6.12. Simulācijas rezultāti

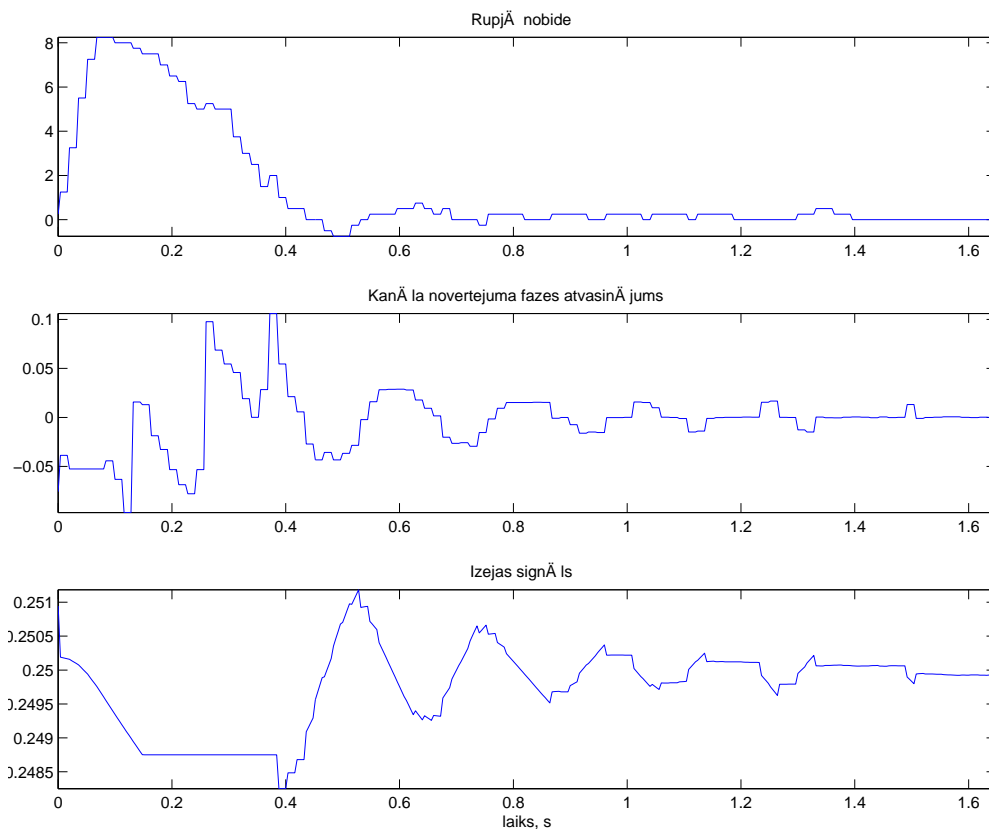
Viens no galvenajiem simulācijas uzdevumiem bija izpētīt sinhronizācijas procesu. Tika veikta virkne eksperimentu variējot PID vadības bloka bloka parametrus. Signāli PID bloka ieejās un izejā pie parametriem, kas ir doti 6.1. tabulā, ir attēloti 6.16. attēlā. Redzam, ka sinhronizācijas process ir iteratīvs un ir nepieciešami vairāki cikli lai noregulētos automātiskās vadības sistēma. Augšējais attēls parāda simbolu nobīdi, ko dod CP korelators. Vidējais attēls parāda kanāla novērtējuma fāzes atvasinājumu. Trešais attēls parāda signālu PID vadības bloka izejā un stūrējošajā NCO ieejā.

Kļūdu intensitātes mērījumu rezultāti modelim ar AWGN kanālu ir parādīti 6.17. attēlā. Šeit dota kļūdu intensitātes (BER) atkarība no attiecības signāls / troksnis (SNR) uztvērēja ieejā. Mērījumi ir veikti kad sistēma ir nosinhronizējusies (pirmie 3000 biti netiek ņemti vērā).

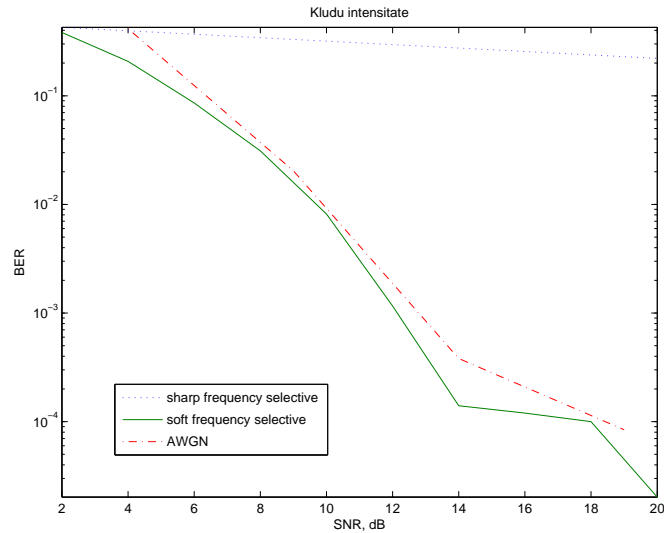


6.15. att. Sakaru sistēmas ar diskretizāciju modelis

Attēlā ir dots sistēmas veiktspējas salīdzinājums pie dažādiem kanāla tipiem: AWGN, ar statistisku līdzenu kanāla raksturlīknes formu (skat 4.7.) attēlu un ar statistisku gadījumveida kanāla raksturlīknes formu. Kā redzam, kanālā ar lēzeni nevienmērīgu raksturlīkni (soft channel), sistēma parāda izcilu veiktspēju un nodrošina BER=10e-4 jau pie attiecības signāls/troksnis 14dB.



6.16. att.: PID vadības bloka oscilogrammas. Augšējais attēls parāda simbolu nobīdi, ko dod CP korelators. Vidējais attēls parāda kanāla novērtējuma fāzes atvasinājumu. Trešais attēls parāda signālu PID vadības bloka izejā un stūrējošajā NCO ieejā.



6.17. att. Kļūdu intensitāte pie dažādiem kanālu tiem

6.13. Secinājumi un sistēmas galvenie parametri

Izmantojot automātiskās vadības sistēmu uz PID bāzes, ir iespējams nodrošināt ļoti precīzu divpakāpju simbolu sinhronizāciju OFDM uztvērējā. Tajā pašā laikā PID vadības bloks ir iekārta, kas prasa rūpīgu noskaņošanu izmantojot lielu skaitu manuālu eksperimentu vai arī sarežģītu programmatūru un patentētas metodes.

| Parametrs | Vērtība |
|-------------------------|-----------|
| Datu apakšnesēju skaits | 52 |
| Takts frekvence | 20 kHz |
| Pārraides ātrums | 26 kbit/s |
| SNR pie BER=10e-4 | 14dB |

6.2. tabula Sakaru sistēmas ar diskretizāciju galvenie parametri

6.14. Sistēmas uzlabošana

Galvenos uzlabošanas pūliņus būtu jāvērs uz sinhronizācijas ātruma palielināšanu. Vienkāršākais risinājums varētu būt PID vadības bloka noskaņošana. Papildus iespējas parādās, ja uztvērēja ieejā mēs novietotu bloku, kas nodrošina variējamu aizturi un līdz ar to dod iespēju precīzāk izvēlēties diskretizācijas momentu. Šāda tipa sistēma būs aprakstīta vienā no nākošajām nodaļām.

7. nodaļa

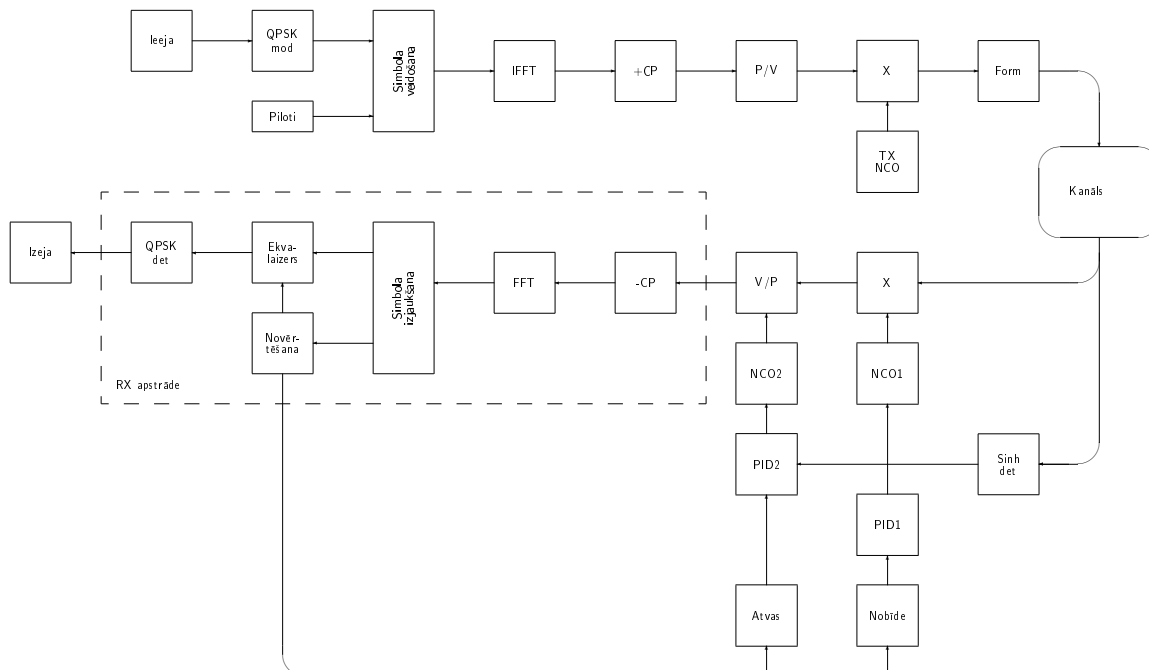
7. Modelis: Sistēma ar frekvences sinhronizāciju.

7.1. Ievads

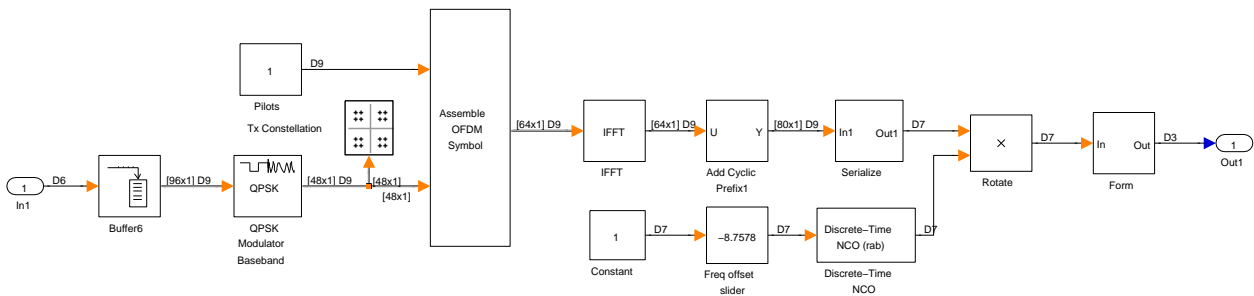
Visas līdz šim veidotās sakaru sistēmas darbojās režīmā, kad raidītāja un uztvērēja takts frekvences ir absolūti vienādas. Reālajā dzīvē raidītājs un uztvērējs atrodas kaut kādā attālumā viens no otra un tādējādi tie izmanto katrs savu takts ģeneratoru, kas neizbēgami noved pie takts frekvenču atšķirības. Kādas sekas izraisa takts frekvenču nesakritība? Pirmkārt, ja ir notikusi simbolu sinhronizācija un ir atrasts pareizs nolasīšanas moments (skat 6. paragrāfu), takts frekvenču atšķirības dēļ, nolasīšanas moments ar katru nolasi nobīdās. Tas noved pie situācijas, kad pēc brīža nojūk simbolu sinhronizācija. Otrkārt, pasliktinās nesējsvārstību savstarpējā ortogonalitāte, jo diskretizētajā signālā parādās papildus locekļi $e^{j2\pi(f^2-f^1)}$, kas rodas frekvenču nesakritības dēļ.

7.2. Sakaru sistēmas blokshēma

Par pamatu izmantosim modeli no 6. paragrāfa. Mums būs jāmodificē raidītājs tā, lai būtu iespējams mainīt raidītāja CAP diskretizācijas frekvenci, jeb pamatjoslas nulles frekvenci.



7.1. att. Sakaru sistēma ar frekvences sinhronizāciju



7.2. att. Raidītāja modelis (OFDM Transmitter)

Pamatjoslas nulles frekvences maiņu var realizēt caur reizinātāju, kas pierēzinās attiecīgas frekvences (kas vienāda ar nobīdi) sinusoīdu. Šis sinusoīdas ģeneratoru apzīmēsim kā bloku “TX NCO”.

Uztvērēja daļa veiktās izmaiņas ir ievērojami lielākas. Frekvenču nobīdes kompensācijai tiek izmantots jauns skaitliski vadāmais ģenerators (NCO1), kuru vada vēl viens PID vadības bloks (PID1). Bloks “Nobīde” nodrošina kanāla fāzes raksturlīknes vidējās nobīdes aprēķināšanu. Pārējā uztvērēja daļa paliek praktiski bez izmaiņām.

7.3. Raidītāja modelis

Ņemot vērā, ka CAP tiek imitēts ar upsampling, lai saglabātu vienotu simulēšanas pamatfrekvenci, CAP frekvences maiņu realizēsim caur reizinātāju, kas pierēzinās mūsu “analogajam” signālam attiecīgas frekvences komplekso eksponenti. Ievērojot Furjē transformāciju īpašības tas nobīdīs visu signāla spektru un efekts būs līdzīgs kā mainot diskretizācijas frekvenci.

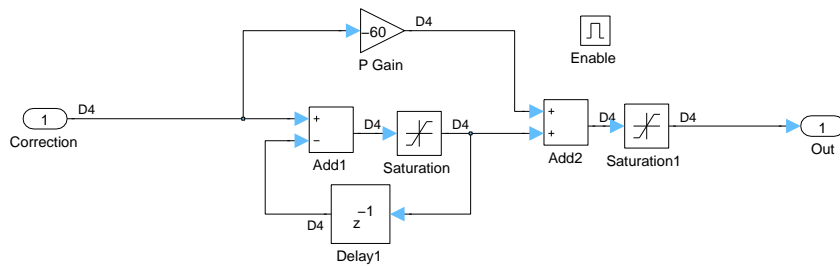
Raidītāja modelis ir parādīts 7.2. attēlā. Kā jau bija minēts iepriekšējā paragrāfā, frekvenču nobīdes imitācijai tiks izmantots reizinātājs (Rotate) ar skaitliski vadāmu ģeneratoru (Discrete time NCO). Bloki “Constant” un “Freq offset slider” nodrošina skaitliski vadāmā ģenerators (Discrete time NCO) vadību.

7.4. Uztvērēja modelis

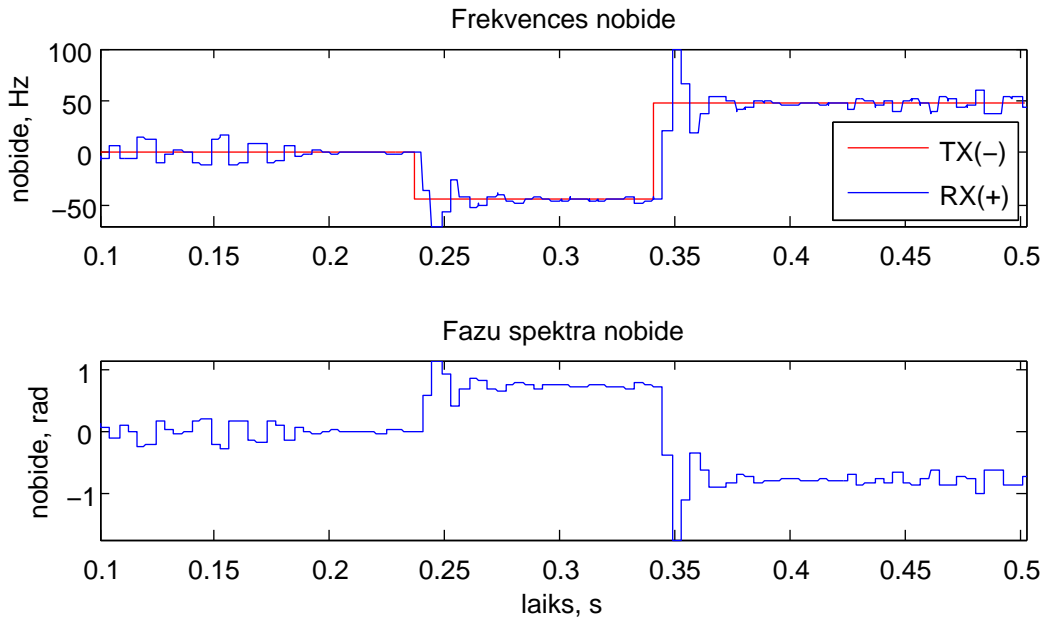
Uztvērēja modelis ir parādīts 7.3. attēlā. Ieejas signāls, pēc diskretizācijas frekvences pazemināšanas, nonāk uz reizinātāju (Shift spectrum), kas veic frekvenču nobīdes kompensatora funkciju. Uz otru šī reizinātāja ieeju pienāk kompleksas eksponentes signāls ar 0-100Hz frekvenci no skaitliski vadāmā ģenerators (Sampling NCO). Šo ģeneratoru vada frekvences proporcionāli-integrāli-diferenciālais (PID) vadības bloks (Freq PID)(skat 7.6. paragrāfu). Uz šo vadības bloku pienāk informācija par kanāla novērtējuma fāžu vidējo nobīdi. Frekvences PID uzdevums ir koriģējot bloka “Sampling NCO” frekvenci tā lai kanāla novērtējuma fāžu vidējā nobīde būtu 0. Frekvences pieskaņošanas operācija tiek strobēta ar signālu ”Symbol CLK” no simbolu sinhronizācijas bloka (Symbol sync) un tiek atļauta tikai tad, kad ir iestājusies simbolu sinhronizācijas smalkās pieskaņošanas fāze un ir fiksētas kārtējā OFDM simbola beigas (to nodrošina bloks ”Logical Operator 1” ar saitēm uz Simbolu sinhronizācijas PID vadības bloku (Time PID) un simbolu sinhronizācijas ģeneratoru (Symbol sync)). Pārējā signālu apstrāde OFDM uztvērējā notiek tāpat kā sistēmā ar diskretizāciju (skat paragrāfu 6.4.).

7.5. Kanāla novērtējuma fāžu apstrādes bloks

Kanāla novērtējuma fāžu apstrādes bloks (Phase Fit) nodrošina kanāla novērtējuma fāzes raksturlīknes aproksimāciju ar taisni $y = an + b$. Bloka ieejā pienāk 56 nolašu novērtējuma



7.5. att. Frekvences PID vadības bloks



7.6. att. Frekvences sinhronizāciju raksturojošās oscilogrammas.

Dotajā modelī pastiprinājums “P Gain” tika atrasts eksperimentālā ceļā.

Frekvences PID darbību raksturojošās oscilogrammas ir dotas 7.6. attēlā. Oscilogrammās ir labi redzams uztvērēja frekvences sinhronizācijas process. Kad izmainās raidītāja pamatjoslas nulles frekvence (parādīta ar sarkano līniju augšējā grafikā), uztvērējā tas izpaužas kā visu pilottonu fāzes spektra nobīde un līdz ar to arī kanāla novērtējuma fāžu nobīde (līkne apakšējā grafikā). Šī nobīde ir vadības signāls PID vadības bloka ieejā, kurš cenšas ģenerēt tādu izejas signālu (parādīts ar zilo līniju augšējā grafikā) lai kompensētu raidītāja frekvences nobīdi.

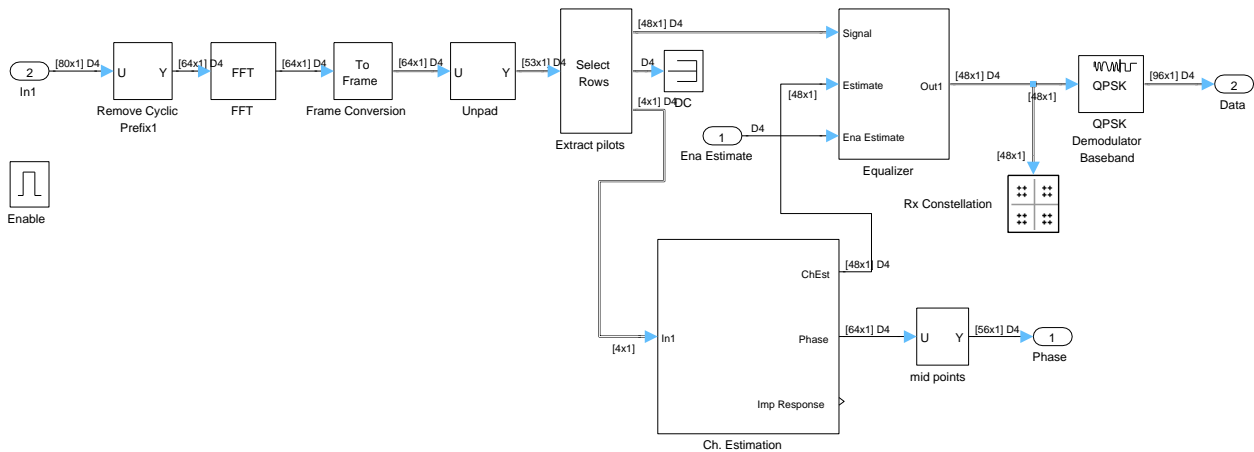
7.7. Uztvērēja apstrādes bloks

Saglabājot tādu pašu funkcionalitāti kā iepriekšējā modelī (skat paragrāfu 6.9.), tika nedaudz pārveidots uztvērēja apstrādes bloks. Jaunā bloka shēma ir dota 7.7. attēlā. Kā redzam atsevišķā blokā “Ch. estimation” ir iznestas darbības, kas ir saistītas ar kanāla novērtēšanu. Savukārt bloki, kas bija saistīti ar kanāla novērtējuma fāžu apstrādi tagad ir vispār iznesti ārā (skat paragrāfu 7.5.). Visu pārējo bloku funkcijas ir saglabājušās (skat paragrāfu 6.9.)

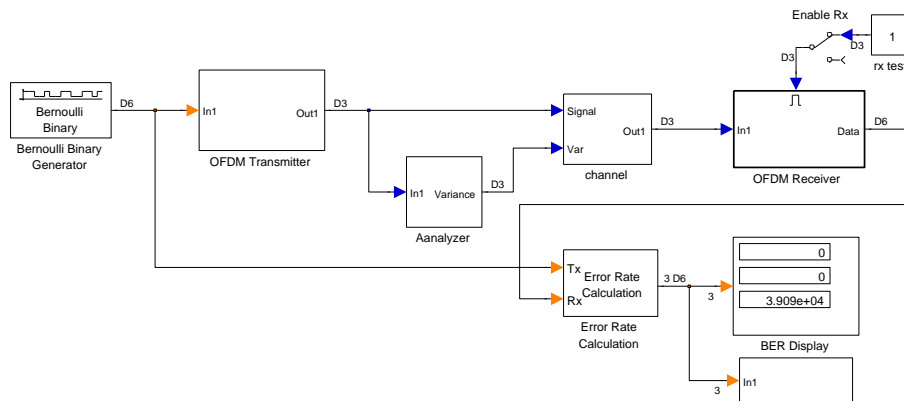
7.8. Sakaru sistēma

Sakaru sistēmas blokshēma ir dota 7.8. attēlā. Tā ir līdzīga iepriekšējā modeļa ar diskretizāciju blokshēmai.

Šajā modelī ir izmantoti uzlaboti sakaru kanāla simulācijas un trokšņa jaudas aprēķinu

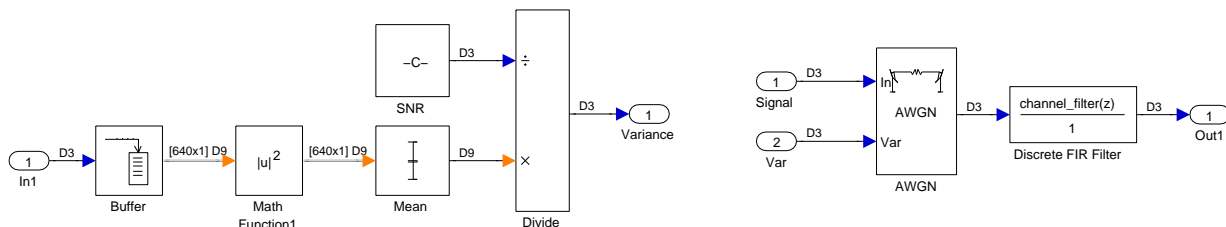


7.7. att. Uztvērēja apstrādes bloks

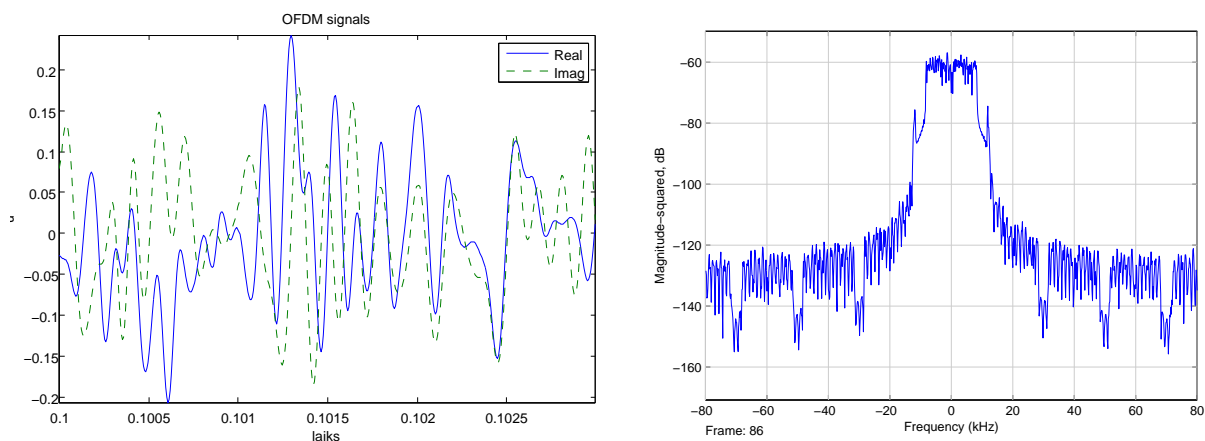


7.8. att. Sakaru sistēma ar frekvences sinhronizāciju

(Analyzer) bloki. Pareizai attiecības signāls/troksnis (SNR) iestatīšanai ir milzīga nozīme, jo sakaru sistēmas kļūdu intensitātes (BER) simulācijās SNR parasti tiek atlikts pa abscisu asi. Līdz šim nepieciešamās signāls/troksnis attiecības (SNR) nodrošināšanai tika izmantota M-S funkcija, kas analizēja ieejas signālu, atrada signāla vidējo jaudu un netieši mainīja AWGN bloka parametru. Jaunajā shēmā ir ļoti redzams, ka uz bloku AWGN kanāla modeli tiek padota trokšņa dispersija, kuras aprēķināšanu veic bloks “Analyzer”.



7.9. att. Trokšņa jaudas aprēķinu bloks (pa kreisi), sakaru kanāla modelis (pa labi)



7.10. att. OFDM signāls (pa kreisi) un tā spektrs (pa labi)

7.9. Trokšņa jaudas aprēķinu bloks

Bloks ir nepieciešams lai atrastu signāla vidējo jaudu un pēc tam izrēķinātu nepieciešamo baltā Gausa trokšņa vidējo kvadrātisko novirzi (dispersiju) lai sasniegtu doto attiecību signāls/troksnis sakaru kanālā. Bloka modelis ir dots 7.9. attēlā, tur pat blakus ir redzams arī skaru kanāla modelis. OFDM pārraidītā signāla nolases nonāk 640 nolašu, kas atbilst vienam 80-nolašu OFDM simbolam ar 8x upsampling, buferi. Bufera saturs tiek kāpināts kvadrātā un tiek atrasta vidējā vērtība, tādējādi tiek atrasta pārraidāmā signāla vidējā jauda. Blokā “SNR” iekš wokspace uzdotais SNR lielums tiek pārvērsts no decibelēm uz reizēm. Signāla jauda tiek izdalīta ar šo lielumu un tiek iegūta trokšņa dispersija. Atbilstoši Matlab dokumentācijai, formula, kas apraksta trokšņa dispersiju ir:

$$\text{ja } P = E[u^2] \text{ un } T = \tau \text{ tad } V = \frac{PT}{\tau 10^{\frac{SNR}{10}}} = \frac{E[u^2]}{10^{\frac{SNR}{10}}} \quad (7.1)$$

kur V ir trokšņa dispersija, u ir spriegums, P ir signāla jauda, T ir simbola garums, τ ir diskretizācijas laiks.

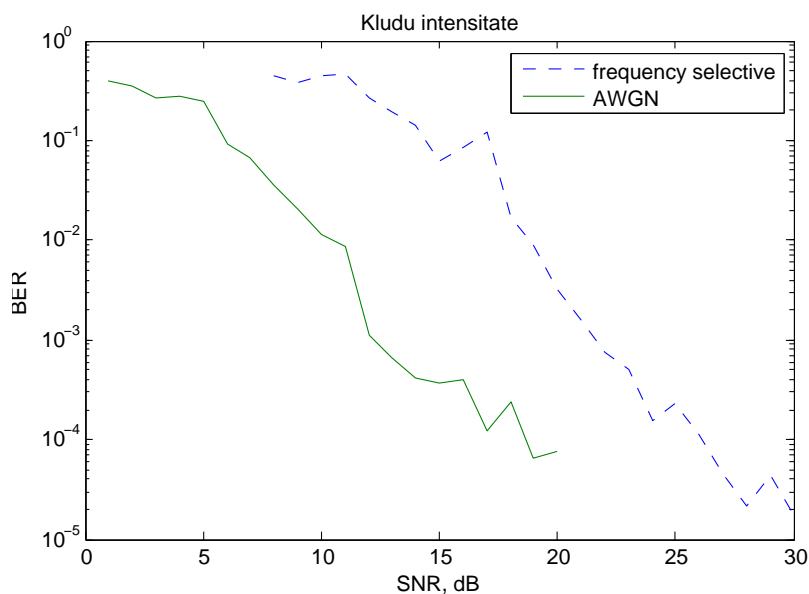
7.10. Simulāciju rezultāti

Uzskatāmībai 7.10. attēlā ir doti OFDM raidītāja izejas signāla spektrs un oscilogramma.

Galvenais veikto simulāciju mērķis ir panākt stabilu frekvences sinhronizāciju kā arī izpētīt frekvences sistēmas veiktspējas samazināšanos frekvences sinhronizatora kļūdu dēļ. Šajā modelī tika izmantots arī savādāks SNR iestatīšanas veids tāpēc atšķirsies arī rezultātu interpretācija. Uzņemta bitu kļūdu intensitātes (BER) atkarības no attiecības signāls/troksnis dažādos kanālos ir redzama 7.11. attēlā. Kā redzam no simulāciju rezultātiem, frekvenču selektīvā kanālā frekvenču sinhronizācija strādā ievērojami sliktāk (SNR zaudējums aptuveni 5dB).

7.11. Secinājumi un sistēmas galvenie parametri

- Izmantojot sistēmu ar atgriezenisko saiti ir iespējams nodrošināt frekvences sinhronizāciju 1% robežās no pamatjoslas platumā (pamatjoslas platumas ir 20kHz, frekvence tiek pieskaņota ± 100 Hz diapazonā).
- Nevienmērīga kanāla frekvenču raksturlīkne atstāj lielu negatīvu iespaidu uz frekvenču sinhronizāciju.



7.11. att. Kļūdu intensitāte pie dažādiem kanālu tiem

| Parametrs | Vērtība |
|-------------------------|-----------|
| Datu apakšnesēju skaits | 52 |
| Takts frekvence | 20 kHz |
| Pārraides ātrums | 26 kbit/s |
| SNR pie BER=10e-4 | 16dB |

7.1. tabula Sakaru sistēmas ar frekvences sinhronizāciju galvenie parametri

7.12. Sistēmas uzlabošana

Ja OFDM sistēmas frekvence ir relatīvi augsta, ir iespējamas situācijas, ka frekvenču nobīde pārsniedz 1% no pamatjoslas platuma. Šādās situācijās būtu jāizmanto kādu no rupjās frekvenču sinhronizācijas algoritmiem [6].

A Pielikums

Saīsinājumi

| | |
|------|--------------------------------------|
| CP | Cyclic prefix |
| DFT | Discrete Fourier transform |
| FFT | Fast Fourier transform |
| IFFT | Inverse Fast Fourier transform |
| NCO | Numerically controlled oscillator |
| PID | Proportional - integral - derivative |
| SNR | Signal to Noise ratio |

Bibliogrāfija

- [1] Rohit Negi, John Cioffi 'Pilot Tone Selection for Channel Estimation in a Mobile OFDM System', Information Systems Laboratory, Stanford University, Stanford, CA 94305, IEEE Transactions on Consumer Electronics, Vol 44, No 3, august 1998. pp 1122-1128
- [2] Heinrich Meyr, Marc Moeneclaey, Stefan A. Fechtel 'Digital communication receivers', 1998 John Wiley & Sons, Inc
- [3] IEEE Standard IEEE Std 802.11a-1999, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications High-speed Physical Layer in the 5 GHz Band.
- [4] Jaakko Astola, Leonid Yaroslavsky, "Advances in Signal Transforms: Theory and Applications", EURASIP Book Series on Signal Processing and Communications, Volume 7, Hindawi Publishing Corporation, 2007
- [5] Jan-Jaap van de Beek, Magnus Sandell, Per Ola B̄rjesson, 'ML Estimation of Time and Frequency Offset in OFDM Systems', IEEE Trans. on Signal Processing, vol. 45, no. 7, pp. 1800-1805, July 1997.
- [6] Fischer, V., Kurpiers, A. (2002). Frequency synchronization strategy for a PC-based DRM receiver. 7th International OFDM-Workshop (InOWo02).
- [7] wikipedia.org
- [8] google.com

B Pielikums

Darba žurnāls

| Datums | h | Komentāri |
|------------|---|---|
| 2008-10-16 | * | Sanjemts no Dainja. Sanjemts no Dainja (turpin) |
| 2008-10-17 | * | Edited model. learning Simulink. |
| 2008-10-20 | * | Rx Gain3 now is calculated from Tx Gain1. Removed scope crowd (still working at this point). Introduced new model os'fft (FFT with sync)Updated old model. Shotcuted buffers. Delay=4. Tested QPSK. OK. Clenup. |
| 2008-10-21 | * | Testing 960sample buffer. Delay=(2+960*2)*2. Testing Disassemble OFDM frames. Estimating delay. Unfinished. |
| 2008-10-22 | * | Successful shortcut after "Assemble OFDM frame" |
| 2008-10-23 | * | Playing. |
| 2008-10-24 | * | All model working. Tx/Rx Delay 7684 samples. BER=0 if SNR=35dB. Switched user io to boolean. |
| 2008-10-30 | * | Replaced FFT with PHI. Unimportant updates |
| 2008-10-31 | * | Adding convolution. Unfinished. |
| 2008-11-07 | * | Conv-based Correlator. Created auto correlation based frame sync detector Created OFDM signal samples |
| 2008-12-04 | * | tested ifft of QPSK added averaging of sync gen (unfinished) |
| 2008-12-05 | * | Introduced Frame Sync Averaging unit (working OK). Short symbol detector. (Unfinished) Short symbol generator. (Unfinished) |
| 2008-12-10 | * | newline test1. push test. short sequence IFFT simulation added. Updated short OFDM symbol IFFT test (successful) Disconnected non-working short symbol detector in main model |
| 2008-12-11 | * | thx |
| 2008-12-12 | * | Modelii A'sakaru'sistema'fft integreeju iistos iisos signaalus, vari apskatiit korelaaciju un autokorelaaciju arhiveets. arhiveets. Merge branch 'master' of git213.175.92.220:ofdm. Modelim pievienots īso simbolu ģenerators. Izlabota kļūda Add cyclic prefix blokā. |
| 2009-01-03 | * | Signed-off-by: dklavins <Administrator.(none)> |
| 2009-01-08 | * | Merge branch 'master' of git@213.175.92.220:ofdm |
| 2009-01-13 | * | Pievienots korelaācijas bloks. papild. |
| 2009-01-14 | * | autokorelacija short symbol |
| 2009-01-30 | * | Fixinf freq eq |
| 2009-02-02 | * | Apvienoju TX un RX parametrus Short simbolu novākšanas bloks Kosmētiski uzlabojumi. Cosmetics. Freq equalizer testi. |
| 2009-02-07 | * | Aizsardziiba pret daliishanu uz 0 FreqDom ekvalaizeraa Nomainiits sample time konstanshu gen blokos no inheriteduz skaitli. Summas aizvietotas ar Mean blokiem |

turpinājums nākošajā lapā

| <i>turpinājums no iepriekšējās lapas</i> | | |
|--|---|---|
| Datums | h | Komentāri |
| 2009-02-10 | * | Ielādēti faili no "Matlab Exchange". Autors Dick Benson. 2008.g. ievienots D.Benson modelis UWB sinhro. Labs paraugs sinhro izpētei. |
| 2009-02-19 | * | Improved cyclic prefix processor based on "On Synchronization in OFDM Systems Using the Cyclic Prefix", Jan-Jaap van de Beek, Magnus Sandell, Per Ola Borjesson. |
| 2009-02-23 | * | Pārslēdzos uz globāliem mainīgajiem nPhi, nFrame, nCyclic. Pievienots aiztures detektēšanas bloks (nestrādā) izmanīta jaudas kalkulācija |
| 2009-02-23 | * | Rx un Tx pārvietoti uz subsistēmām Sākts veidot sinhronizācijas cilpu |
| 2009-02-25 | * | Pielikti timing control bloki Rx. Nestrādā. |
| 2009-02-27 | * | Izveidots atsevišķs modelis sync testēšanai. |
| 2009-02-28 | * | Sinhronizācija (TX ok, RX-problēmas ar līdzīgiem sample reitiem. PPDU catcher etc. Working model. Timing loop doesnt work. |
| 2009-03-01 | * | Sampling unit Still has problems Suspicious signal from TX |
| 2009-03-02 | * | Timing |
| 2009-03-03 | * | Izmainīti taimingi TX |
| 2009-03-04 | * | Pievienots RX downsample. Izskatās, ka ir nepareizs (pārāk rets) sampling (strobe)signāls. |
| 2009-03-14 | * | Rate converters for scope |
| 2009-03-16 | * | Salaboju takts frekvences RX. Problemas ar averaging. Strobe deej tas pazaude datus. |
| 2009-03-18 | * | Beidzot! Palaists Averaging, saregulēti pareizi frame reiti u.c. Averaging blokā Buffer nomainīts uz deley line, ieviesta strobēšana |
| 2009-03-31 | * | minor |
| 2009-04-10 | * | Removed short,long sym |
| 2009-04-14 | * | Farrow. Atrasts iemesls kaapeec netraadaa cilpa. Vajadzīgs PLL |
| 2009-04-16 | * | Uztvereja parbuvesana. Saakts buuveet pilottonju detektoru. Rx paarveide. Re-buffer bloks Tx-Rx sedariibas testiem |
| 2009-04-17 | * | Pievienots TRX.MDL Tx-Rx saderiibas testiem Pievienots pilot-signaalu baazeets channel estimators ar interpolaaciju 4- >64 Meeginaats iesleegt atgriezenisko saiti no faazu spektra uz PI kontrolieri. Sinhronizaacija nenotiek |
| 2009-04-21 | * | signaala dimensijas prob (neatris). prob.turp. |
| 2009-04-22 | * | References modelis: raiditajs, uztverejs bez sinhro. Paralelais savienojums. Formatting. Saakts short'sync. Straadaajoss iiso simbolu filtrs. Short symbol sync. stradaaa. Cosmetics. Pielikti BER meeriitaji: tx-rx modelim iso simbolu sinhronizaācijas modelim |
| 2009-04-28 | * | Pievienots BER meeriitaaajs. Pabeigts Rx. AWGN kanaals. AWGN kanaals |
| 2009-05-02 | * | Pielikts "to workspace" bloks |
| 2009-05-04 | * | Short-sync vecais modelis ar pilotiem paarkopeets uz jaunu modeli short-sync-pilots Modelim bez sinhronizaācijas nonemti pilottonji. Izmesti pilottonji no short-sync modelja |
| 2009-05-08 | * | Paareja uz 1 iso simbolu ciklu test'short'sym. Izmesti pilottonji no short'sync (izlabota problema). Mazsvarigas izmaiņas |
| 2009-05-11 | * | Vienkaarsots tx'rx'parallel nosync: novaakta grueessana pa blokiem. Cosmetics |
| 2009-05-26 | * | Comparison of estimators. Fixed tittle |
| 2009-05-29 | * | symbol timing simulation equalizer simulation |
| 2009-06-05 | * | short long sync BER simulācijas |
| 2009-06-06 | * | Working system with equalizer, fading channel and sync. BER vs SNR simulaacija: 1)nosync 2)nosync+equalizer 3)frame'sync+equalizer |
| 2009-07-17 | * | Izmaiņas prieks raksta Izlabot BPS formula |
| 2009-07-21 | * | Jauns modelis - subcarrier pilottonju testeshanai. Saakts darbs ar subcarrier-tipa pilotsignaalu izmantosanu kanaala noverteeshanai. |
| <i>turpinājums nākošajā lapā</i> | | |

| <i>turpinājums no iepriekšējās lapas</i> | | |
|--|---|---|
| Datums | h | Komentāri |
| 2009-08-17 | * | Straadaa estimators. bet Nepareizi. |
| 2009-08-20 | * | Konstantu pilotu generaacija un uztvershana |
| 2009-09-01 | * | Interpolaācijas testi |
| 2009-09-03 | * | Interpolators straadaa |
| 2009-09-07 | * | Pabeigta estimator cilpa. nestraadaa. Grafiku ziimeeshana rakstam. |
| 2009-09-14 | * | Interpolaācijas vizualizacija |
| 2009-09-22 | * | eksparimenti ar interpolaaciju. Nestraadaa. |
| 2009-09-25 | * | Eksperimenti ar interpolaaciju |
| 2009-09-28 | * | interpollacija |
| 2009-09-29 | * | Pabeigts uz pilotiem bazeets ekvvalaizers. Simulaaciju rezultaati mat failos. Pilot ekvvalaizers. Siikas izmaiņas |
| 2009-10-05 | 1 | |
| 2009-10-06 | 2 | |
| 2009-10-09 | 6 | Mainīts uztvēreja elementu izvietojums |
| 2009-10-20 | 4 | Impulse response based channel gain. Interpolation, angle unwrap problems. |
| 2009-10-21 | 6 | Izlabota kļūda interpolatorā. SUPER. Sākts jauns modelis: simbolu sinhronizācija |
| 2009-10-23 | * | Uzlikti bloki seriaalajai apstraadei. Nepabeigts. |
| 2009-10-28 | 2 | Jauna modeļa būve. nepabeigts |
| 2009-10-30 | * | Modeļa parametru mod |
| 2009-11-02 | 2 | |
| 2009-11-03 | 4 | Eksperimenti ar NCO. Saaka straadaat sampling. Palaists downsampling to tChip |
| 2009-11-04 | * | Novaakts upsample/downsample. Straadaa symbol offset detector. Simbolu impulsu gen (nepabeigts) |
| 2009-11-09 | * | Symbolu takts generators. (Straadaa nepareizi) |
| 2009-11-11 | * | Symbol sync OK. Fully working model with BER=0. Model makeover |
| 2009-11-17 | 3 | Optimizēts modelis ar sinhrono kanālu. |
| 2009-11-20 | 6 | Kosmētiskas izmaiņas. Assemble symbol bloks. |
| 2009-11-27 | 2 | Apraksts. |
| 2009-12-02 | 5 | Novākts liekais |
| 2009-12-04 | 5 | Laika diagrammu uzņemšana |
| 2009-12-08 | 2 | Laika diagrammu zīmēšana. |
| 2009-12-11 | 3 | BER kalkulācija u.c. |
| 2009-12-12 | 4 | Atskaite. BER grafiki |
| 2009-12-14 | 1 | Atskaite. |
| 2009-12-16 | * | BER modelēšana |
| 2009-12-21 | 2 | Sākts darbs pi Phi OFDM chan estimation |
| 2009-12-22 | 5 | Uzlabots modelis ar pilotsimboliem. Būvēsim uz tā pamata Phi-est. Pāreja uz M S-Funkcijām ar parametriem |
| 2009-12-28 | 1 | |
| 2010-01-05 | * | BER salīdzināšana NoTrans, FFT, Phi pie dažādām kanāla raksturliknēm. |
| 2010-01-15 | * | Kanāli pārvietoti uz vienu mapi. |
| 2010-01-19 | 8 | Jaunais šoftkanāls. Pāreja uz ārējiem kanāliem. |
| 2010-01-20 | 8 | Hadamard model. BER graphs. |
| 2010-01-22 | 6 | Hadamard transform |
| 2010-01-26 | 8 | Short symbol design. |
| 2010-01-27 | * | Izveidots modelis jaunu pilotu struktūru. phi pilots |
| 2010-01-29 | * | BER merijumi phi pilots |
| 2010-02-03 | * | Performance simulācijas sistēmām ar ekvvalizāciju. |
| 2010-02-04 | * | Kanāla implementācija caur FIR filtru |
| 2010-02-05 | * | Simulācijas ar kanālu ar FIR filtru |
| <i>turpinājums nākošajā lapā</i> | | |

| <i>turpinājums no iepriekšējās lapas</i> | | |
|--|---|---|
| Datums | h | Komentāri |
| 2010-02-16 | 2 | Sākts darbs ar sampling modeli. AWGN pārslēgts uz RMS bloka lietošanu. Darbs pie upsamplera |
| 2010-02-17 | 2 | Pabeigts TX upsamplers (TX RF). Variable fractional delay bloks |
| 2010-02-19 | * | Farrow Fractional delay izpēte |
| 2010-03-12 | 4 | pub |
| 2010-03-31 | * | color and likelihood. Likelihood |
| 2010-04-1 | 1 | |
| 2010-04-19 | 3 | Eksperimenti ar variable delay. |
| 2010-04-21 | * | Pievienots ekvalaizers. Paareja uz 4x samplerate RX. Nestaadaa. |
| 2010-04-26 | 2 | RX ar 4x takti. Straadaa. Nesinhronizeejas |
| 2010-04-28 | 4 | Aotom cilpas testi. Nekonverģē. |
| 2010-05-19 | * | -39 offset problem. Nestraadaa |
| 2010-05-31 | * | Izlabota dalisana ar 0. PID oscilāciju prob |
| 2010-06-28 | 6 | PID simplification. Fixed PID init. Fixed PID timing issue. PID strādā. Jāuzlabo ātrums. Eksperimenti ar PID konvergenci. Majas modelis |
| 2010-07-20 | 6 | Modelis bez Farrow filtra. Novakst farrow delay |
| 2010-07-21 | 6 | Mainīts TX/Assemble symbol bloku izvietojums |
| 2010-07-27 | 4 | Kosmētika |
| 2010-08-03 | 9 | Oscilogrammas. clean. Cosmetics |
| 2010-08-04 | * | Fix |
| 2010-08-05 | * | Izlabota P-V konversijas kluda |
| 2010-09-24 | * | BER merījumi. FFT'SAM'NOF modelis. Stadaajoss modelis |
| 2010-10-08 | * | Paarsauktas mapes fft'sam'nof par fft'sam |
| 2010-10-22 | * | Unerscores, url atskaitē. filtru koef fft'farrow modelis (straadaa). Neveiksmīgs meginājums uzlabot PID. |
| 2010-11-05 | * | Salikts Freq sinhro modelis. Nav palaists. Paareja uz R2010b. |
| 2010-11-12 | * | Freq sync. Salikts, bet nestrādā. |
| 2010-12-03 | * | Cieša ofdm13 kopija. Nestrādā. |
| 2011-02-10 | * | Perf sim labels |
| 2011-02-22 | * | Freq sync strādā |
| 2011-03-01 | * | Power calculation lietojot simulink blokus. PAPR calculation. slprj ignore |
| 2011-03-02 | * | HAD, Phi un Notrans (QPSK) pāreja uz jauno kanāla modeli ar PAPR. Notrans vienkāršošana |
| 2011-03-04 | * | Pārdēvēts long'nosync -> fft'long. Dzēsts fft'sam'old. Jauns modelis: FFT ar CP, virknes pārraidi un polotsimbolu ekvelizāciju. Vienkāršota simbolu un pilotsimbolu veidošana (netbilst 802.11). OFDM raksturliknes. Uzbūvēta SC-FDE sistēma. Darbojas! |
| 2011-03-09 | * | SC TDE modelis (nepabeigts) |
| 2011-03-10 | * | SC TDE modelis (darbojas). Simulations enhancements. HAD and PHI sim |
| 2011-03-15 | * | Freq sync modelis ar jauno kanālu un PAPR. |
| 2011-04-14 | * | Uzlaboti modeļi (bufferis); Complex Hadamard |
| 2011-04-15 | * | Complex Hadamard simulācijas |
| 2011-04-20 | * | plot ber |
| 2011-05-09 | * | Uzlabojumi priekš Buvejam |
| 2011-05-10 | * | Uztvērēja moduļa kosmetika. |
| 2011-05-17 | * | Color maps |
| 2011-06-14 | * | Uzlabojumi priekš Buvejam SS. |
| 2011-06-15 | * | Mainīts jaudas analīzes bloks |