

Lucrarea de laborator Nr 2.

Tema 1: Cercetarea și proiectarea Filtrelor: TJ, TS, Banda, Oprește Banda.

Scopul lucrării: Cercetarea și proiectarea filtrelor de diferite caracteristici și parametri. Filtul Trece Jos pentru filtrarea sunetului și a tensiunii de alimentare în blocurile de alimentare ale PC. Filtrele Trece Sus pentru filtrarea semnalelor de comunicare numerică. Filtrele de banda sau selective pentru canalele de comunicare selective. Filtre cu regim de rezonanță.

Surse și dispozitive tehnice necesare pentru elaborarea lucrării de laborator:

1. Calculator PC;
2. Acces la Internet;
3. Mediul de măsurare și testare National Instruments: MultiSim;
4. Îndrumare electronice de descriere a arhitecturii PC, DOS, BIOS.

Sarcina tehnică pentru efectuarea lucrării de laborator:

1. Să se asambleze schema electrică a unui filtru TJ, TS, Bandă în mediul NI:MultiSim ;
2. Să se efectueze simularea și măsurările pentru a identifica proprietățile filtrelor TJ, TS și Banda;
3. Să se extragă grafice și să se elaboreze concluzii care determină calitatea filtrelor TJ, TS, Banda.

Perfectarea raportului de evaluare a rezultatelor efectuării lucrării de laborator:

1. Foaia de titlu;
2. Cuprinsul;
3. Schema electrică de principiu pentru filtrele TJ, TS, Banda;

4. Descrierea sarcinii în parametri tehnici;
5. Algoritmul de efectuare a simulărilor / măsurărilor;
6. Graficele obținute în rezultatul simulărilor / măsurărilor efectuate.

I. Introducere

Filtrarea reprezintă prelucrarea unui semnal (în domeniul timp) având ca rezultat schimbarea spectrului de frecvență original al semnalului. Schimbarea constă în reducerea (filtrarea) unor componente nedorite ale semnalului.

Filtrele se pot împărți în două mari categorii: filtre analogice și filtre numerice.

Filtrele analogice lucrează cu semnale continue, pe când cele numerice lucrează cu secvențe discrete. Teoria modernă a eșantionării și a prelucrării numerice a semnalelor a făcut posibilă înlocuirea filtrelor analogice cu cele numerice în numeroase aplicații, printre avantajele filtrelor numerice numărându-se:

- filtrele numerice sunt programabile software și prin urmare sunt ușor de construit și de testat;
- performanțele filtrelor numerice nu variază în funcție de temperatură sau umiditate;
- filtrele numerice au un raport cost/performanță mai bun decât filtrele analogice;
- filtrele numerice sunt stabile și nu necesită componente de mare precizie;
- filtrele numerice necesită doar operații aritmetice simple de înmulțire și adunare /scădere și prin urmare sunt ușor de implementat.

Din punct de vedere al realizării fizice, un filtru numeric poate fi un program într-un calculator, un microprocesor programabil sau un circuit integrat dedicat. Filtrele numerice liniare clasice sunt de două tipuri: filtre cu răspuns finit la impuls (Finite Impulse Response Filters – FIR) și filtre cu răspuns infinit la impuls (Infinite Impulse Response Filters – IIR).

Indiferent de tipul lor (analogice sau numerice), filtrele reduc sau elimină componentele de frecvență nedorită din semnalul prelucrat. În funcție de domeniul de frecvență în care semnalele sunt lăsate să treacă sau sunt atenuate (rejectate), filtrele pot fi clasificate în patru categorii:

- filtre trece jos (FTJ) – lasă să treacă semnalele de frecvență joasă și atenuază componentele de frecvență înaltă;
- filtre trece sus (FTS) – lasă să treacă semnalele de frecvență înaltă dar atenuază semnalele de frecvență joasă;
- filtre trece bandă (FTB) – lasă să treacă semnalele care au frecvența într-o anumită bandă de frecvență;
- filtre oprește bandă (FOB) – atenuază semnalele care au frecvența într-o anumită bandă de frecvență.

FTJ permite trecerea semnalelor care au frecvența mai mică decât frecvența de tăiere f_t , iar FTS lasă să treacă semnalele cu frecvență mai mare decât frecvența de tăiere. FTB lasă să treacă semnalele cu frecvența între cele două frecvențe de tăiere – inferioară f_{t1} , respectiv superioară f_{t2} . FOB atenuază semnalele care au frecvența între cele două frecvențe de tăiere.

Domeniul de frecvență în care filtrul permite trecerea semnalelor se numește bandă de trecere (BT), iar domeniul de frecvență în care filtrul nu lasă să treacă semnalele se numește bandă de oprire (BO) sau bandă de blocare. Frecvențele de tăiere și benzile de trecere și de oprire, alături de alți parametri, fac parte din specificațiile de proiectare ale filtrelor.

Filtrele ideale sunt caracterizate de amplitudine unitară (0 dB) în banda de trecere și de amplitudine zero ($-\infty$ dB) în banda de oprire. În cazul implementărilor practice, reale, aceste performanțe ale filtrului nu sunt complet realizate. La filtrele ne-ideale, între banda de trecere și banda de oprire există o zonă nenulă numită bandă de tranziție (TR), în care amplitudinea se modifică gradual de la amplitudinea unitară (0 dB) la amplitudine nulă ($-\infty$ dB).

Banda de trecere se definește în cazul filtrelor reale ca fiind intervalul de frecvență în care amplitudinea variază de la 0 dB la -3 dB. Pentru anumite tipuri de filtre, amplitudinea în banda de trecere poate prezenta mici oscilații cunoscute sub numele de ripple (riplu). Amplitudinea acestor oscilații se numește ripple al benzii de trecere sau simplu ripple și este de fapt diferența între amplitudinea reală a filtrului în zona de trecere și amplitudinea unitară ideală. Deoarece în practică atenuarea în zona de

oprire nu poate fi infinit de mică, în cadrul proiectării trebuie specificată o limită care trebuie respectată.

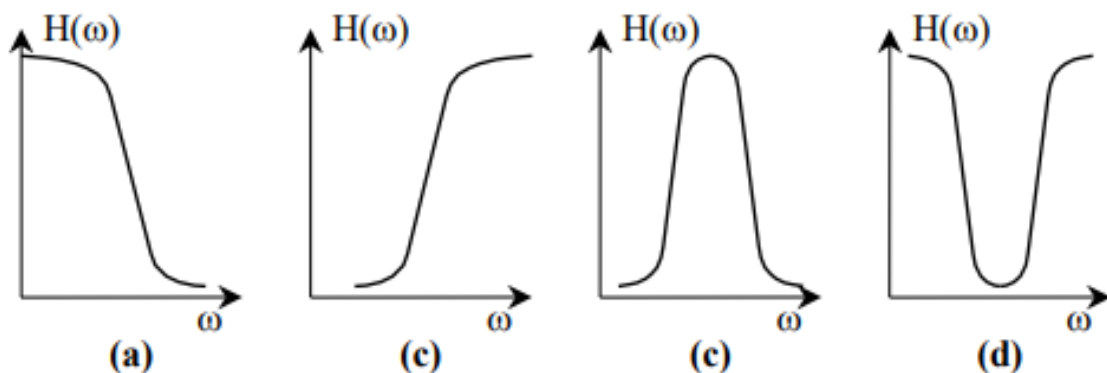
Prin urmare, în cazul filtrelor reale apar ca specificații de proiectare suplimentare banda de tranziție (măsurată în Hz), ripple-ul (dB) și atenuarea în banda de oprire (dB).

Proiectarea filtrelor poate fi considerată o procedură de aproximare a răspunsului în frecvență al unui filtru ideal cu respectarea unor specificații (performanțe) impuse.

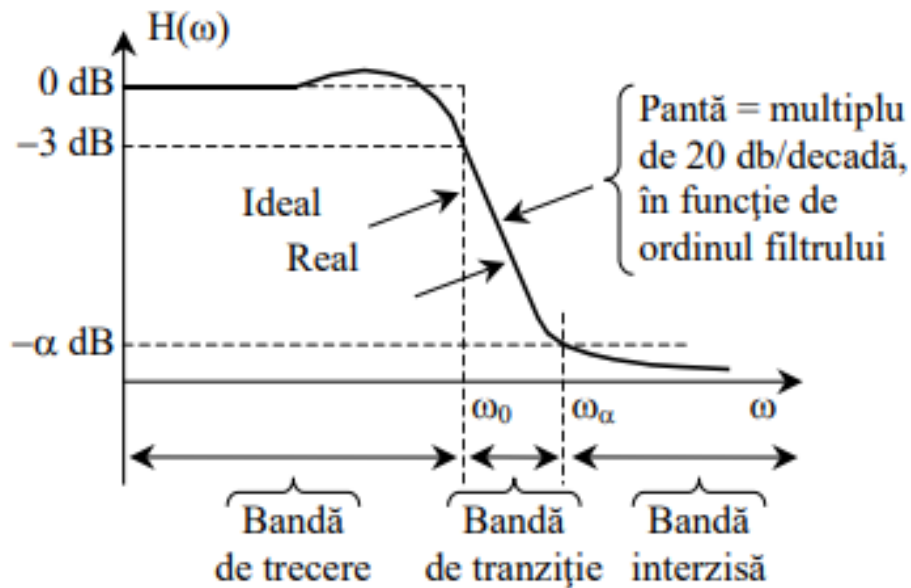
După cum se observă din analiza Fig. 2.20 și Fig. 2.21, între filtrele ideale și cele reale există o serie de deosebiri. Amplitudinea în banda de trecere și în banda de oprire în cazul ideal este plată și constantă, dar în cazul filtrelor reale apare fenomenul de ripple. Filtrele ideale nu au bandă de tranziție, care apare însă la implementarea practică. Proiectarea filtrelor trebuie să realizeze un compromis pentru obținerea unor valori ale parametrilor (ripple, atenuarea în banda de oprire, lățimea benzii de tranziție) cât mai apropiate de cazul ideal, compromis care depinde de structura filtrului și de algoritmul de proiectare.

II. Filtre analogice

Clasificarea filtrelor.

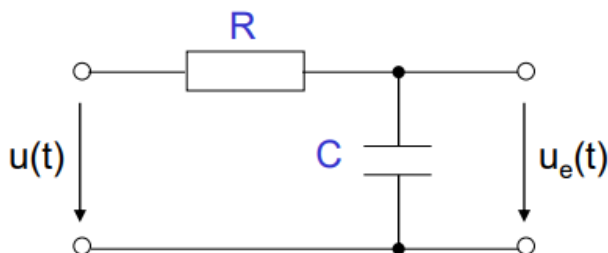


(a) – trece-jos; (b) – trece-sus; (c) – trece-bandă; (d) – oprește-bandă.



Caracteristica amplitudine-frecvență a unui filtru trece-jos.

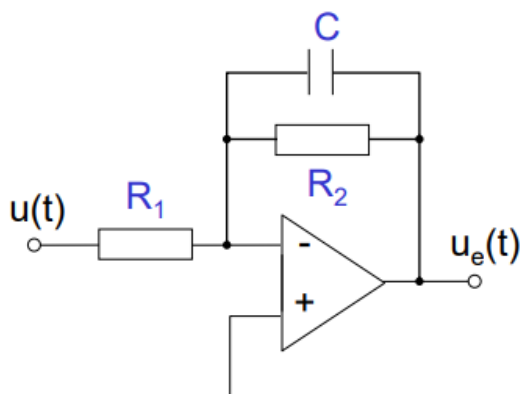
Filtru analogic pasiv trece jos



$$u(t) = RC \frac{du_e(t)}{dt} + u_e(t)$$

$$H(s) = \frac{1}{1 + RCs}$$

Filtru analogic activ trece jos

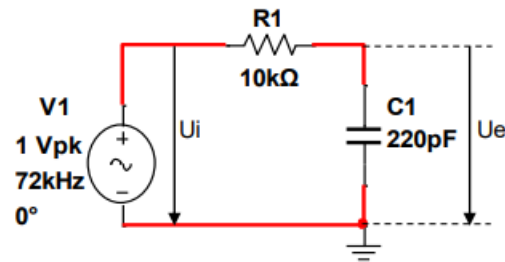


$$u(t) = R_1 C \frac{du_e(t)}{dt} + \frac{R_1}{R_2} u_e(t)$$

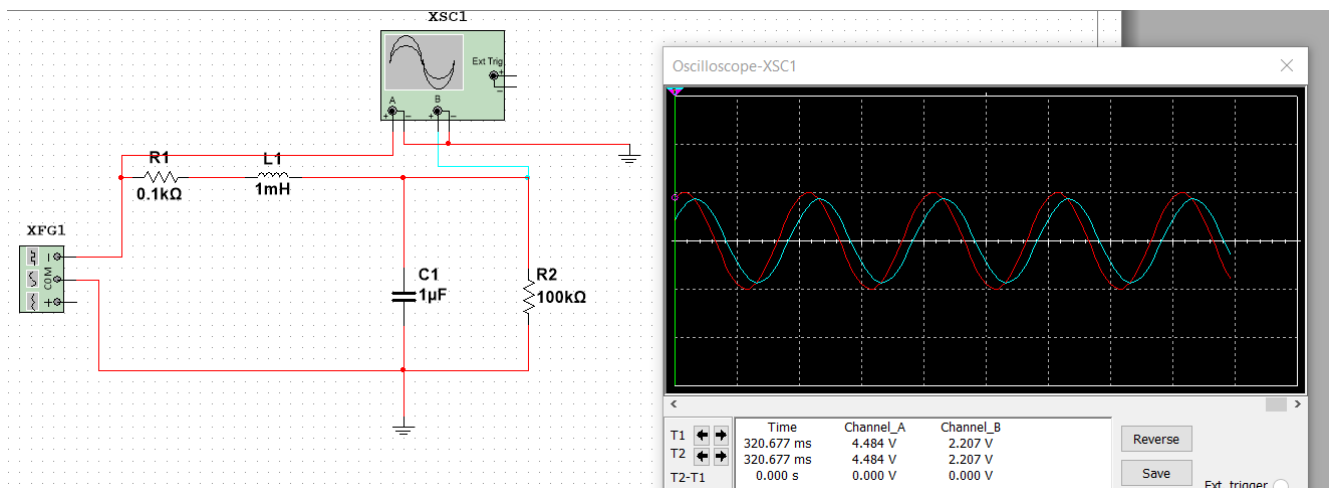
$$H(s) = \frac{R_2}{R_1} \frac{1}{1 + R_2 C s}$$

III. Scheme pentru cercetare

a) Sa se cerceteze schema Filtrului TJ.



Realizarea scheme în NI MultiSim.



Sa se cerceteze schema in condițiile:

$$U=5V,$$

$$f=1KHz - 100MHz,$$

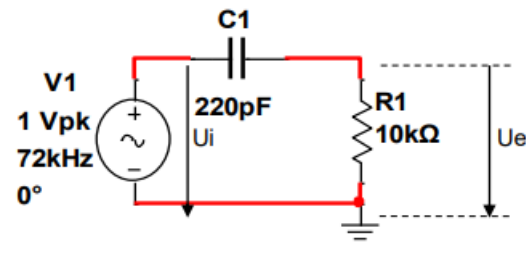
$$L=0.001mH - 10mH,$$

$$C=0.0001uF - 10uF,$$

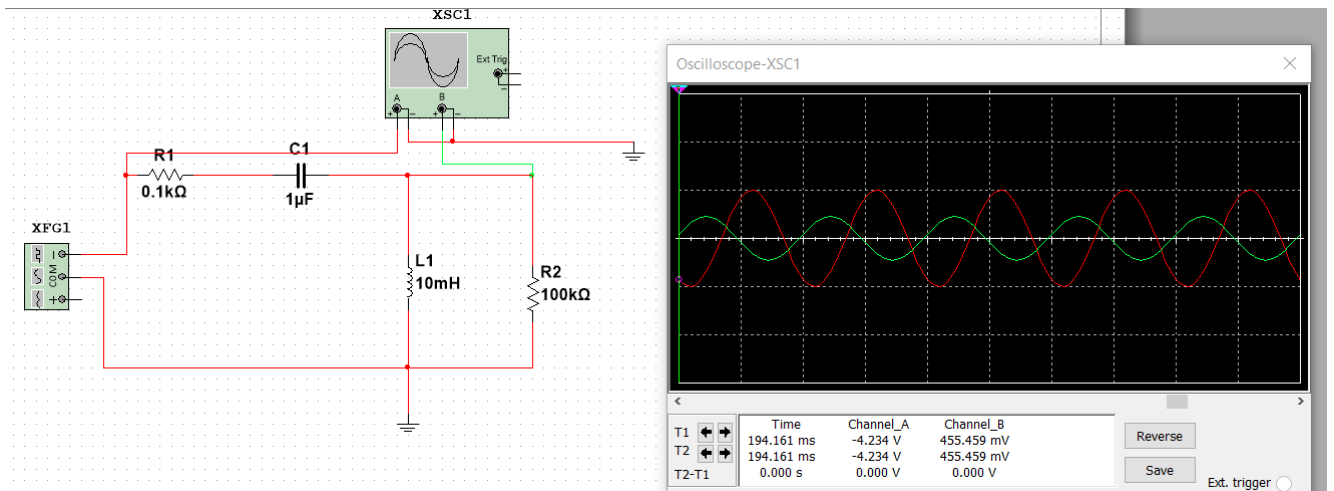
$$R=1K.$$

Pentru fiecare condiție să se obțină graficile respective ce demonstrează funcționalitatea filtrului TJ.

b) Sa se cerceteze schema Filtrului TS.



Realizarea scheme în NI MultiSim.



Sa se cerceteze schema in condițiile:

$$U=5V,$$

$$f=1\text{KHz} - 100\text{MHz},$$

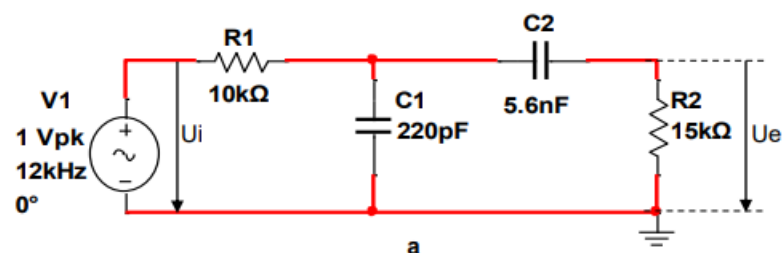
$$L=1\text{mH} - 1000\text{mH},$$

$$C=0.0001\mu\text{F} - 10\mu\text{F},$$

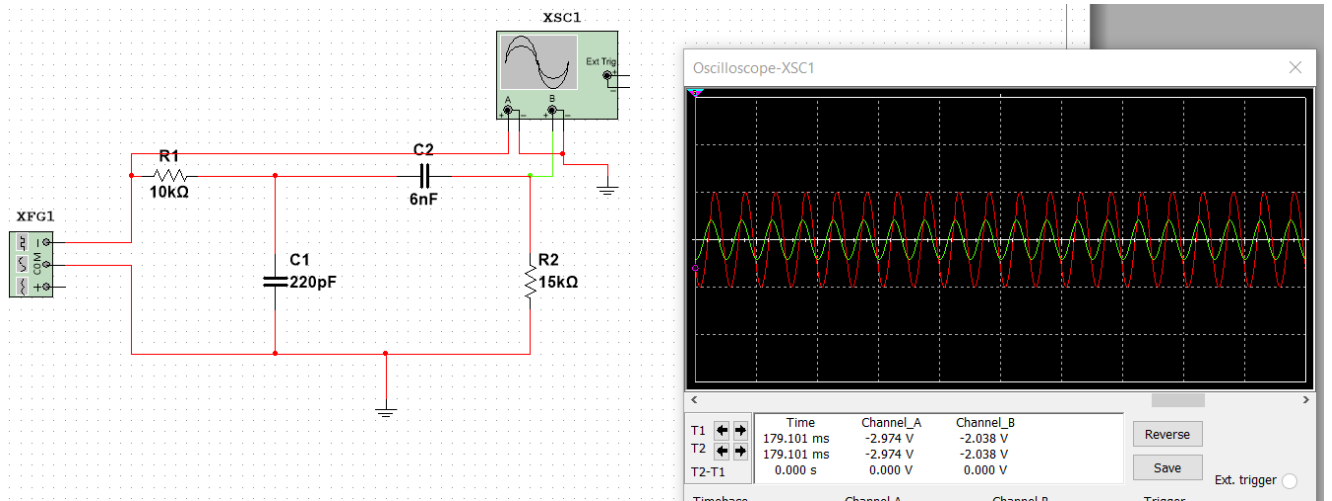
$$R=1\text{K}.$$

Pentru fiecare condiție să se obțină graficile respective ce demonstrează funcționalitatea filtrului TS.

c) Sa se cerceteze schema Filtrului Trece Bandă.



Realizarea scheme în NI MultiSim.



Sa se cerceteze schema in condițiile:

$$U=5V,$$

$$f=1KHz - 100MHz,$$

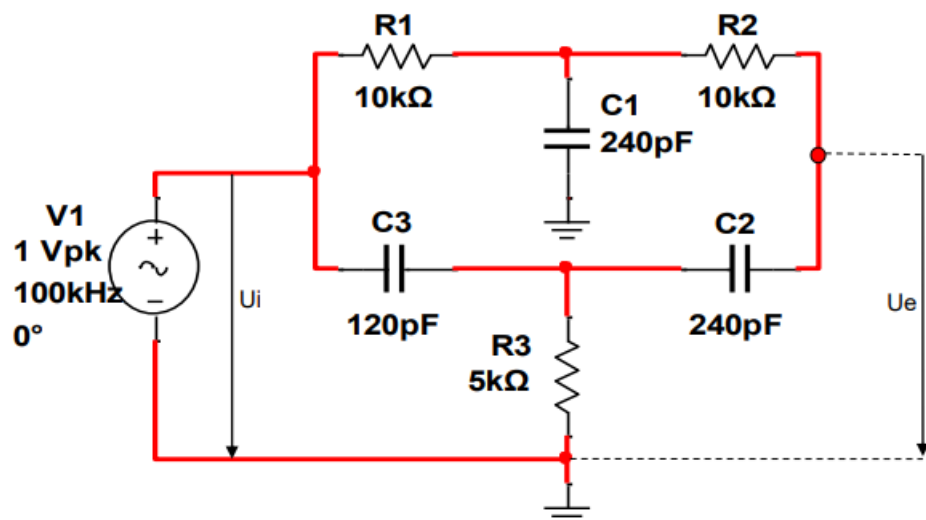
$$L=0.001mH - 10mH,$$

$$C=0.0001\mu F - 10\mu F,$$

$$R=1K.$$

Pentru fiecare condiție să se obțină graficile respective ce demonstrează funcționalitatea filtrului Trece Bandă.

d) Sa se cerceteze schema Filtrului Oprește Bandă.



Sa se cerceteze schema in condițiile indicate in schema:

	INDRUMAR DE LABORATOR	COD: S.03.O.031
	BAZELE TRANSMITERII DE DATE	DATA: 25.08.2020 PAGINA: 9/10

Pentru fiecare condiție să se obțină graficile respective ce demonstrează funcționalitatea filtrului Oprește Bandă.

Să se constuească graficile $U_{out} \rightarrow f$, $U_{out} \rightarrow L$, $U_{out} \rightarrow C$.

	INDRUMAR DE LABORATOR	COD: S.03.O.031
	BAZELE TRANSMITERII DE DATE	DATA: 25.08.2020 PAGINA: 10/10

Referințe bibliografice:

1. Turbo Help 3/5: <http://calc.fcim.utm.md/biblioteca/index.php>.
2. Terminal.exe: <http://calc.fcim.utm.md/biblioteca/index.php>.
3. National Instruments. MultiSim: <http://sine.ni.com/psp/app/doc/p/id/psp-412>.
4. Mediul Arduino: <https://www.arduino.cc/>.
5. Cristian Colonati. Radiocomunicații digitale. Galați, 2004, 273 p. (Sursă electronică: http://www.asrr.org/attachments/*.*).
6. Mihai V. Micea. Telecomunicații digitale moderne. Suport de curs. Timișoara, 2008, 137 p. (Suport electronic: http://dsplabs.cs.upt.ro/~micha/publications/pdfs/2008_CNbk__Telecom_BookInfo.pdf).
7. Noi tehnologii pentru comunicații digitale. (Sursă electronică: <http://alexserbanescu.ro/wp-content/uploads/2013/10/Carte-APLICATII-HAOS-in-COMUNICATII.pdf>).
8. Sistemele de Comunicatii Digitale. (Sursă Electronică: <https://ru.scribd.com/document/47271030/Sistemele-de-Comunicatii-Digitale>).
9. Comunicatii digitale. (Sursă Electronică: <http://www.radioamator.ro/articole/1003/>).
10. Generalitati privind sistemele de comunicatii digitale. (Sursă Electronică: <http://www.scribub.com/stiinta/informatica/GENERALITATI-PRIVIND-SISTEMELE12127.php>).
11. Arpad Gellert, Rodica Baci. Programare în limbaj de asamblare. Aplicații. Universitatea Lucian Blaga din Sibiu, 2001. 39 p. (Sursă electronică: <http://webpace.ulbsibiu.ro/arpad.gellert/html/ASM.pdf>).
12. Gabriel Rădulescu. Elemente de arhitectură a sistemelor de calcul. Programare în limbaj de asamblare. Matrix ROM, București, 2007. 368 p. (Sursă electronică: http://ace.upg-ploiesti.ro/cursuri/pla/curs_pla.pdf).
13. Bjarne Stroustrup. The C++ Programming Language (second edition), Addison Wesley, 1991.

	INDRUMAR DE LABORATOR	COD: S.03.O.031
	BAZELE TRANSMITERII DE DATE	DATA: 25.08.2020 PAGINA: 11/10

14. Программирование на языке ассемблера. (Sursă electronică: <http://natalia.appmat.ru/c&c++/assembler.html>).
15. Н.В. Максимов, Т.Л. Партыка, И.И. Попов. Архитектура ЭВМ и вычислительных систем. М.: 2005. 512 с. (Sursă electronică: http://elib.ict.nsc.ru/jspui/bitstream/ICT/1346/1/Arhitektyra_EBM.pdf).
16. Е.Д. Жиганов, А.П. Мощевикин. Передача данных в компьютерных сетях. ПетргУ, 2007 156 с. (Sursă electronică: http://plasma.karelia.ru/~alexmu/nets_tele/nets.pdf).
17. Н.А. Руденков, Л.И. Долинер. Основы сетевых технологий. Екатеринбург, 2011, 377 с. (Sursă electronică: <http://urtk.su/net/books/Rudenkov.pdf>).
18. А.Л. Гельгор. Технологии LTE мобильной передачи данных. СПб. 2011, 204 с. (Sursă electronică: <http://window.edu.ru/resource/169/75169/files/popov3.pdf>).
19. Э. Таненбаум, Т. Остин. Архитектура компьютера, 6-е издание, М.: - 2013. 810 с.
20. Ozten Chelai. Arhitectura Calculatoarelor. Suport de curs și laborator. Universitatea Ovidius Constanța, 2012. 160 p. (Sursă electronică: <https://fmidragos.files.wordpress.com/2012/07/arhitectura-sistemelor-de-calcul.pdf>).
21. Horea Oros. Arhitectura sistemelor de calcul. Suport de curs. Universitatea din Oradea, 2010. 147 p. (Sursă electronică: <http://webhost.uoradea.ro/horos/files/ASC.pdf>).
22. Nani Viorel. Echipamente periferice. Note de curs. Universitatea Ioan Slavici, Timișoara, 2013. 53 p. (Sursă electronică: http://www.islavici.ro/articole/Notite%20Curs_EchipPeriferice.pdf).
23. Mihai Romanca. Microprocesoare și microcontrolere. Universitatea Transilvania din Brașov, 2015. 319 p. (Sursă electronică: <http://vega.unitbv.ro/~romanca/Carte-MpMc%202015/Microprocesoare%20si%20microcontrolere-978-606-19-0683-3.pdf>).
24. Sever Spânulescu. Programarea în limbajul de asamblare a microprocesoarelor. Îndrumar de laborator. Editura Victor, 2004. 256 p. (Sursă electronică: ...)

	INDRUMAR DE LABORATOR	COD: S.03.O.031
	BAZELE TRANSMITERII DE DATE	DATA: 25.08.2020 PAGINA: 12/10

<http://automatica.cch.ro/Laboratoare/Laborator%20sisteme%20cu%20microprocesoare.pdf>).

25. Arpad Gellert, Rodica Baciuc. Programare în limbaj de asamblare. Aplicații. Universitatea Lucian Blaga din Sibiu, 2001. 39 p. (Sursă electronică: <http://webpace.ulbsibiu.ro/arpad.gellert/html/ASM.pdf>).
26. Gabriel Rădulescu. Elemente de arhitectură a sistemelor de calcul. Programare în limbaj de asamblare. Matrix ROM, București, 2007. 368 p. (Sursă electronică: http://ace.upg-ploiesti.ro/cursuri/pla/curs_pla.pdf).
27. Bjarne Stroustrup. The C++ Programming Language (second edition), Addison Wesley, 1991.
28. Программирование на языке ассемблера. (Sursă electronică: <http://natalia.appmat.ru/c&c++/assembler.html>).
29. Н.В. Максимов, Т.Л. Партыка, И.И. Попов. Архитектура ЭВМ и вычислительных систем. М.: 2005. 512 с. (Sursă electronică: http://elib.ict.nsc.ru/jspui/bitstream/ICT/1346/1/Arhitektyra_EBM.pdf).
30. Note de curs – Introducere în rețelele de calculator. (Sursă electronică: http://www.afahc.ro/ro/facultate/cursuri/retele_note_curs.pdf).
31. Mihai Micea. Comunicații digitale moderne. Timișoara, 2008.
32. Ștefan Burlacu. Comunicații analogice și numerice. Sibiu, 2000.
33. В.Г. Баула. Введение в архитектуру ЭВМ и системы программирования. М.: 2003. 144 с. (Sursă electronică: <http://cmcstuff.esyr.org/vmkbotva-r15/>).
34. Э. Таненбаум, Т. Остин. Архитектура компьютера, 6-е издание, М.: - 2013. 810 с.
35. Руденков Н.А., Долинер Л.И. Основы сетевых технологий. Екатеринбург, 2011.
36. А.М. Пуртов. Системы и Сети Передачи Данных. Омск, 2010.