

## ლაბორატორია 10

### სიგნალების მოდელირება და დამუშავება SIMULINK-ში

#### 1 SIMULINK-ის გაცნობა

SIMULINK გარემოა რომელიც სხვადასხვა საინჟინრო სისტემების დიზაინის და სიმულირებისთვის გამოიყენება. მას გააჩნია ინტერაქტიული გრაფიკული ინტერფეისი და დაპროგრამებადი ბლოკების უზარმაზარი ბიბლიოთეკა, რომელთა გამოყენებითაც შეგიძლიათ დაადიზაინოთ, სიმულაცია გაუკეთოთ, დააიმპლემენტოთ და გამოცადოთ სისტემები, როგორებიცაა საკომუნიკაციო, სიგნალების დამუშავების, კონტროლის (control systems), ვიდეო და სურათების დამუშავების სისტემები. სისტემების დიზაინი ბლოკების უბრალო მიერთებით ხდება, რაც გვიმარტივებს მუშაობას და დროს გვიზოგავს. ყველა ბლოკს გააჩნია input ან/და output ტერმინალები, რომელთა მეშვეობით შეგვიძლია სიგნალის სტრიმი ავაწყოთ. SIMULINK-ს შეუძლია როგორც ციფრული, ისე უწყვეტი დროის (ანალოგური) სისტემების და სიგნალების მოდელირება.

#### 1.1 მეორე რიგის სისტემის მოდელირება SIMULINK-ში

ახლა, თქვენ SIMULINK-ში მეორე რიგის სისტემის step-response-ის მოდელს შექმნით. გაუშვით MATLAB-იდან SIMULINK-ის გარემო, აირჩიეთ “blank model” და გახსენით ბიბლიოთეკები. მეორე რიგის სისტემის მოდელის ასაგებად დაგვჭირდება ბლოკები SIMULINK-ის სამი ბიბლიოთეკიდან:

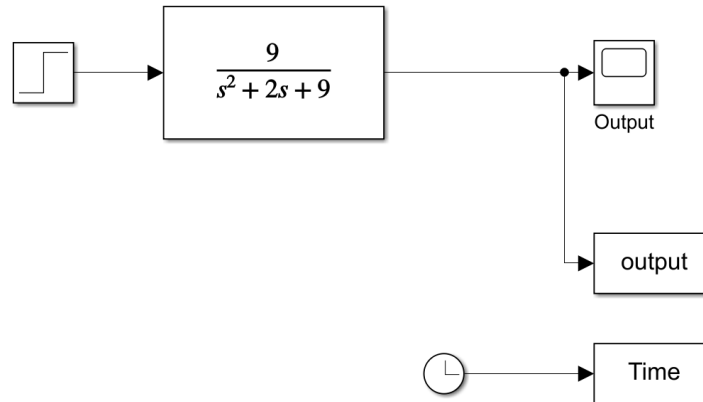
- Sources - რომელშიც მოთავსებულია სხვადასხვა ტიპის სიგნალის დასაგენერირებელი წყაროები;
- Sinks - ბლოკები, რომლების მეშვეობით შეგვიძლია სიგნალის ვიზუალიზაცია (მაგალითად scope, რომელიც ოსცილოგრაფის ანალოგია) და მიღებული output სიგნალების MATLAB-ის workspace-ში დამახსოვრება.
- Continuous - ამ ბლოკებით შეგვიძლია უწყვეტი დროის სხვადასხვა სისტემის სიმულირება. ასევე, გვაქვს ბიბლიოთეკა Discrete, რომელშიც დისკრეტული სისტემების შესაბამისი ბლოკებია მოთავსებული.

SIMULINK-ის მოდელის სამუშაო სივრცეში გადაიტანეთ შემდეგი ბლოკები:

- ბიბლიოთეკა Sources: ბლოკები Step, Clock;
- ბიბლიოთეკა Continuous: ბლოკი Transfer Function;
- ბიბლიოთეკა Sinks: ბლოკები Scope, To Workspace (2 ცალი, ერთს დაარქვით “time”, მეორეს კი “output”);

ბლოკების ერთმანეთზე მიერთებისთვის ბლოკის ტერმინალს დააკლიკეთ მაუსის მარცხენა ღილაკით და მიუერთეთ ის შემდეგი ბლოკის ინფუტს. ერთი სტრიმის გასაყოფათ დააჭირეთ ღილაკს CTRL და ისე დააკლიკეთ სტრიმს.

განათავსეთ ბლოკები ისე, როგორც სურათ 1-ზეა ნაჩვენები



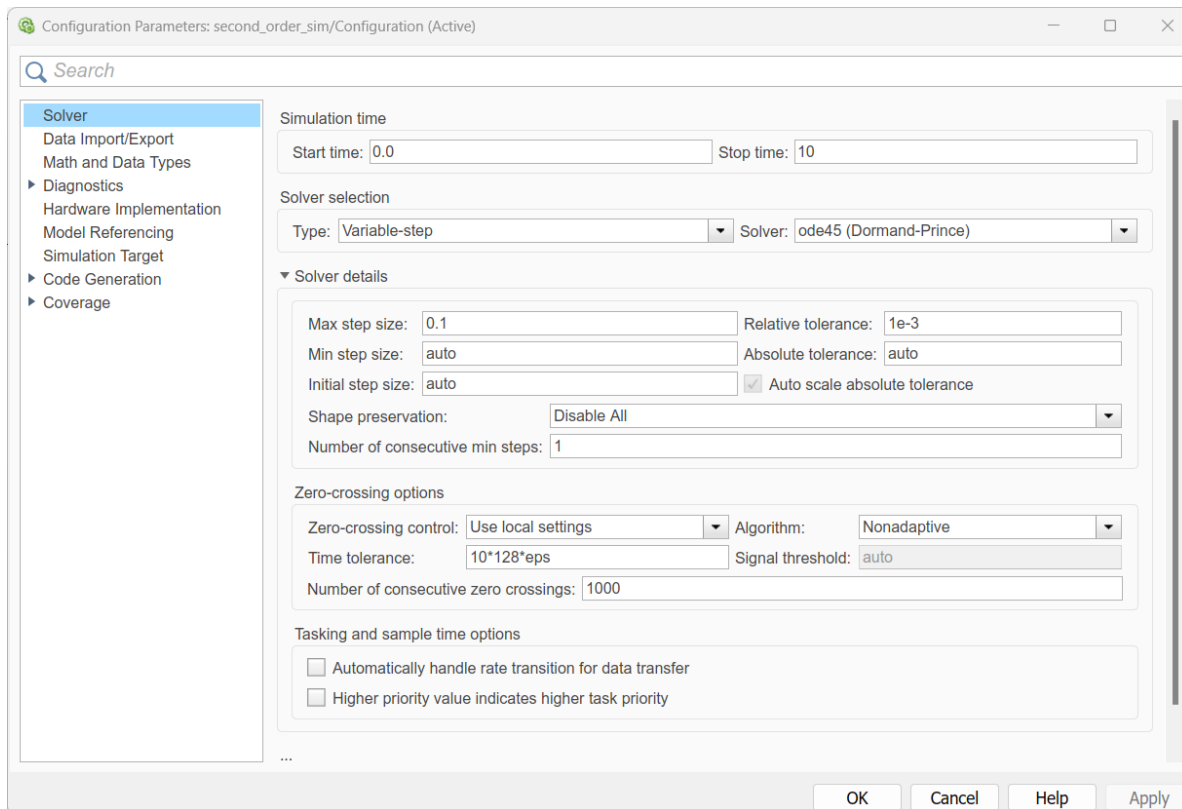
სურათი 1. მეორე რიგის მოდელი SIMULINK-ში.

თითოეული ბლოკის ორჯერ დაკლიკებით იხსნება ფანჯარა, სადაც ბლოკის პარამეტრიზაციაა შესაძლებელი. მაგალითად, step function ბლოკზე ორჯერ დაკლიკებისას იხსნება ფანჯარა, რომელშიც შეგიძლიათ მიუთითოთ ჰევისაიდის ფუნქციის „ჩართვის“ დრო (მიუთითეთ start time: 1) და მისი ამპლიტუდა.

სასურველი მეორე რიგის სისტემის სიმულირებისთვის, გვჭირდება შესაბამისი ტრანზერ ფუნქციის მითითება. ამისთვის გახსენით Transfer Function ბლოკის პარამეტრები და მიუთითეთ შესაბამისი მრიცხველის და მნიშვნელის პარამეტრები.

To Workspace ბლოკების პარამეტრების დამახსოვრების ფორმატში (save format) აირჩიეთ "array". SIMULINK-ში სისტემის სიმულირების შემდეგ MATLAB-ის workspace-ში დაემატება სტრუქტურა out, რომელშიც ვექტორები time და output იქნება მოთავსებული.

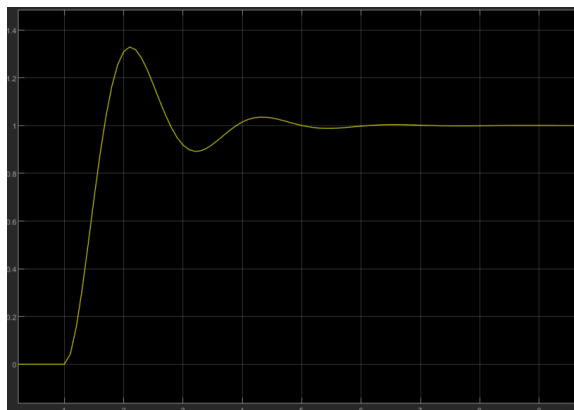
სიმულაციის პარამეტრები: სიმულაციის პარამეტრების შესარჩევად ზუდა მენიუდან აირჩიეთ Simulation → Model Configuration Parameters. გამოჩნდება ფანჯარა, რომელიც სურათ 2-ზეა ნაჩვენები. აქ შეგიძლიათ მიუთითოთ სიმულაციის დაწყების და დასრულების დროები, სიმულაციის შესაბამისი დიფერენციალური განტოლების ამოხსნის მეთოდები (მიუთითეთ ode45 (Dormand-Prince)). ბიჯის ზომა სისტემის სიმულირებისთვის (ამ შემთხვევაში მიუთითეთ Max step size: 0.1). სიმულაციის მაქსიმალური დასაშვები ცდომილება და სხვა პარამეტრები.



სურათი 2. სიმულაციის პარამეტრები

### სიმულაციის გაშვება

ყველაფერი მზადაა სიმულაციის გასაშვებად: დააჭირეთ მწვანე (Run) ღილაკს და სიმულაციის დასრულებისას დააკლიკეთ ოსცილოგრაფს. ოსცილოგრაფზე მეორე რიგის სისტემისთვის დამახასიათებელი რეაქცია გამოჩნდება, რომელიც სურათ 3-ზეა ნაჩვენები.



სურათი 3. მეორე რიგის სისტემის output სიგნალი.

შეგიძლიათ თქვენს სიმულაციაში სხვა ტრანსფერ ფუნქცია და სისტემაში შემავალი ფუნქციები სცადოთ.

## 2 სინუსოიდური სიგნალიდან ხმაურის მოშორება

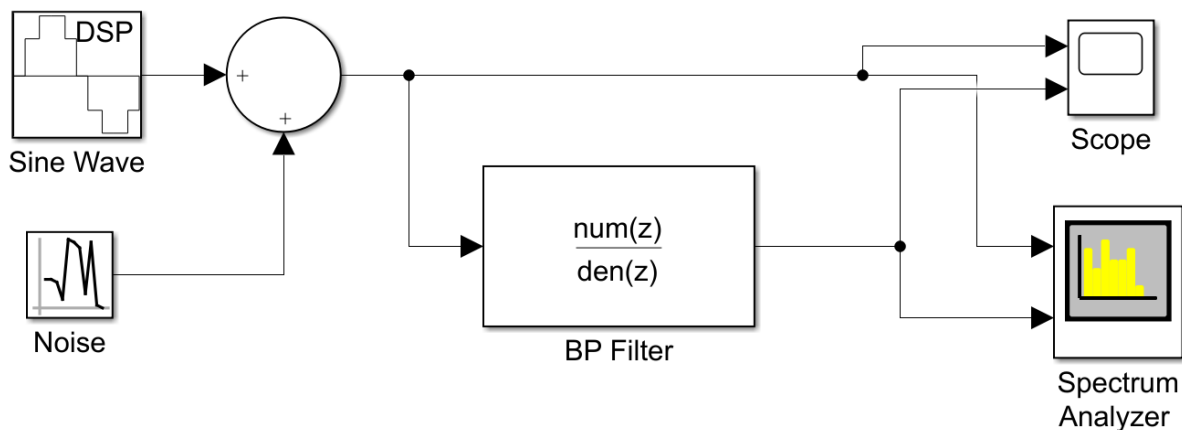
ამჯერად SIMULINK-ში ხმაურიანი სინუსოიდური სიგნალიდან ხმაურის გაფილტვრის სიმულაციას შექმნით.

### 2.1 მეორე რიგის სისტემის მოდელირება SIMULINK-ში

სამუშაო სივრცეში გადაიტანეთ შემდეგი ბლოკები:

- DSP System Toolbox, Sources ბიბლიოთეკიდან: Sine Wave;
- SIMULINK, Sources ბიბლიოთეკიდან: Random Number;
- SIMULINK, Commonly Used Blocks ბიბლიოთეკიდან: Sum;
- SIMULINK, Discrete ბიბლიოთეკიდან: Discrete Filter;
- SIMULINK, Sink ბიბლიოთეკიდან: Scope,
- DSP System Toolbox, Sinks ბიბლიოთეკიდან: Spectrum Analyzer.

დააკავშირეთ ბლოკები, როგორც სურათ 4-ზე არის გამოსახული. სინუსოიდა და ხმაური sum ბლოკის გამოყენებით იჯამება. Scope და spectrum analyzer-ში გამოგვაქვს ორივე, გაფილტრული და გაუფილტრავი სიგნალები. Discrete Filter ბლოკში დისკრეტული ფილტრის ტიპის და a, b პარამეტრების მითითებაა საჭირო, რომლებსაც თქვენი შექმნილი ფილტრის პარამეტრებს გამოიყენებთ.



სურათი 4. ხმაურიანი სინუსოიდური სიგნალი

Sine Wave ბლოკის პარამეტრებში აირჩიეთ ამპლიტუდა **1**, სიხშირე **100 Hz**, და დისკრეტიზაციის სიხშირე (sample rate) **1/8000**. დანარჩენი პარამეტრები უცვლელად დატოვეთ.

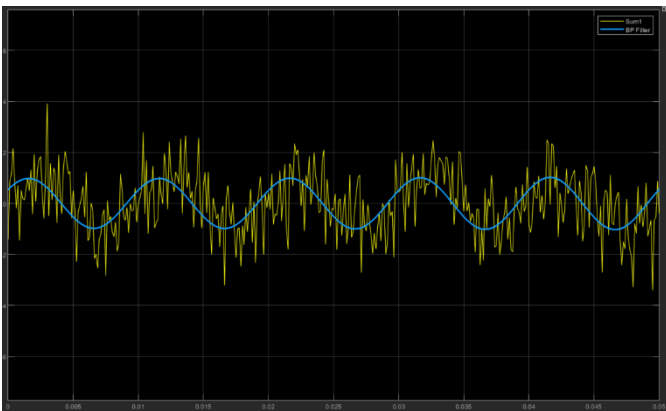
Random Number ბლოკს ხმაურის დასაგენერირებლად ვიყენებთ. გახსენით მისი პარამეტრები და მიუთითეთ შემდეგი: *Mean: 0; Variance: 1; Seed: 0* და *Sample time: 1/8000*.

Matlab-ის FilterDesigner-ის გამოყენებით შექმენით ზუღურგამტარი (bandpass) ფილტრი, რომელიც სინუსოიდას გაატარებს. გამოიყენეთ შემდეგი ფილტრის სპეციფიკაციები:

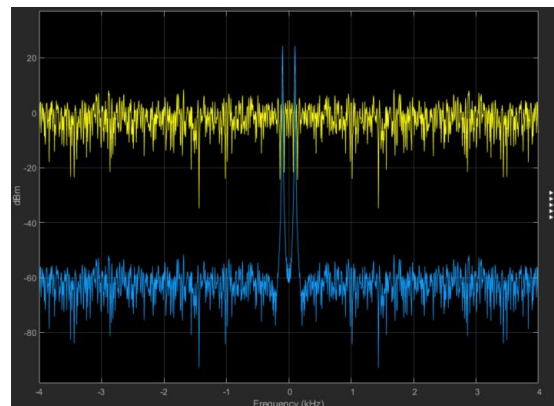
ფილტრის გადაცემის ზოლი იქნება  $50\text{ Hz}$ -დან  $150\text{ Hz}$ -მდე. უკუგდების ზოლში მიუთითეთ სიგნალის ჩახშობა  $60\text{ dB}$ . რა თქმა უნდა, ეს სპეციფიკაციები  $8000\text{ Hz}$ -თ დისკრეტიზირებული სიგნალისთვის. დააგენერირეთ ამ ფილტრის  $a$  და  $b$  კოეფიციენტები პირდაპირი ფორმა II იმპლემენტაციისთვის. მიუთითეთ მიღებული კოეფიციენტები SIMULINK-ის მოდელის Discrete Filter პარამეტრებში.

გაუშვით მოდელი და გახსენით ოსცილოგრაფი და სპექტრის ანალიზატორი. უნდა ნახოთ, რომ აუთფუტ სიგნალიდან ხმაური საგრძნობლად არის მოშორებული.

სურათები 5 და 6 გიჩვენებთ სიმულაციის სასურველ შედეგს.



სურათი 5. შემავალი და გამომავალი სიგნალები (დროში)



სურათი 6. შემავალი და გამომავალი სიგნალების სპექტრი

შეცვალეთ სინუსოიდური სიგნალის სიხშირე ( $20\text{ Hz}$ ,  $50\text{ Hz}$ , ...) და დააკვირდით მიღებულ აუთფუტს.

## 2.2 ხმაურის ალბათური განაწილების ფუნქცია

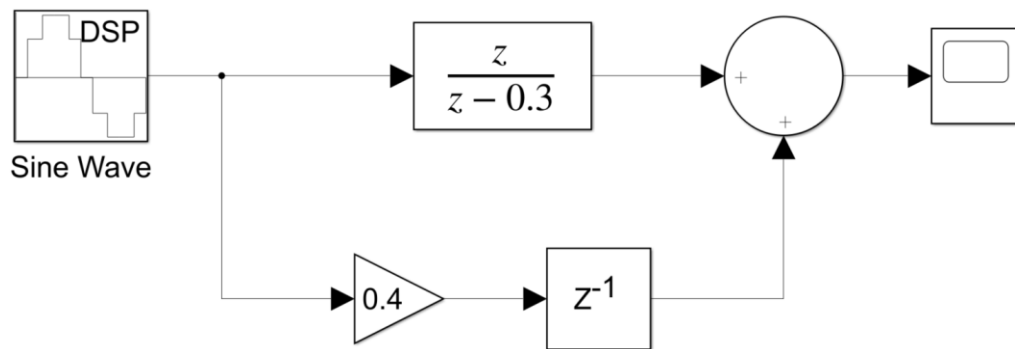
როგორც იცით, თუ სიგნალში ხმაურის ტიპი ჩვენთვის უცნობია, შეგვიძლია მისი განსაზღვრა სიგნალის ჩაწერით და მისი მნიშვნელობების ჰისტოგრამის აგებით.

- To workspace ბლოკის გამოყენებით ჩაიწერეთ ორივე, გაფილტრული და გაუფილტრავი სიგნალის 10 წამიანი შუალედები.
- გამოაკელით გაუფილტრავ სიგნალს გაფილტრული და ააგეთ მიღებული სხვაობის სიგნალის მნიშვნელობების ჰისტოგრამა. რა ტიპის განაწილებაა გამოსახული ჰისტოგრამაზე?

### 3 დისკრეტული დროის LTI სისტემების სიმულირება

SIMULINK-ში ასევე შეგვიძლია დისკრეტული დროის სტრუქტურების სიმულირება, როგორც სურათ 7-ზე არის ნაჩვენები. ყურადღება მიაქციეთ, რომ დისკრეტული დროის ბლოკებს მითითებული უნდა ქონდეთ შესაბამისი დისკრეტიზაციის სიხშირე ან დრო. (თუ სტრიმის წინა ბლოკზე დისკრეტიზაციის სიხშირე ან დრო მითითებულია, დანარჩენ ბლოკებს იგივე დისკრეტიზაციის სიხშირე ავტომატურად გადაეცემათ, თუ დისკრეტიზაციის დროის პარამეტრში (sample time) –1 არის მითითებული.

შექმენით შემდეგი დისკრეტული დროის სისტემის მოდელი SIMULINK-ში. დისკრეტიზაციის დროდ მიუთითეთ 0.5 წამი.



სურათი 7. დისკრეტული დროის LTI სისტემა

დაყოვნების და გამრავლების ბლოკები SIMULINK, Discrete ბიბლიოთეკაშია მოთავსებული.

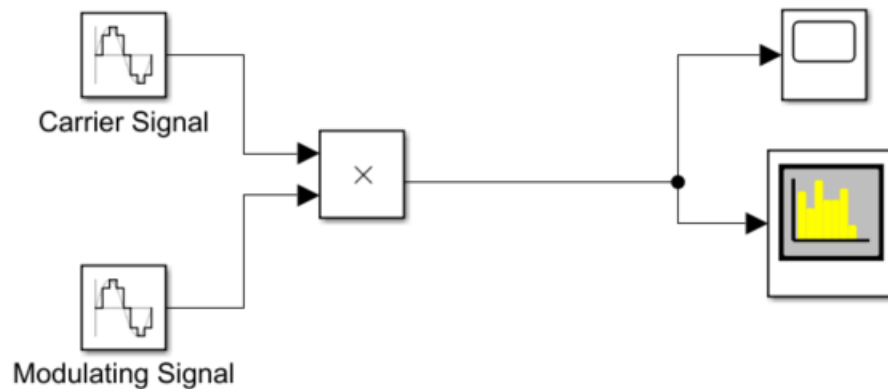
რა ტიპის ფილტრია მოცემული LTI სისტემა? არის რაიმე სიხშირე რომელსაც ფილტრი არ გაატარებს? გამოსცადეთ თქვენი ჰიპოთეზა სხვადასხვა დისკრეტიზებული სინუსოიდური სიგნალის გამოყენებით.

### 4 AM (amplitude modulated) სიგნალის მოდულაცია და დემოდულაცია

რადიო სიგნალების გადაცემის ერთერთი ყველაზე მარტივი მეთოდი AM მოდულაციაა. SIMULINK-ში თქვენ შექმენით AM მოდულაციის და დემოდულაციის სიმულაციას.

#### 4.1 სიგნალის მოდულაცია

შექმენით სურათ 8-ზე მოცემული სისტემის სიმულაცია SIMULINK-ში. Carrier signal, იგივე გადამტანი სიგნალი, სიგნალია, რომელიც გადამცემის რადიო-სიხშირეს შეესაბამება. მის მოდულირებას ვახდენთ მესიჯის სიგნალით, რომელიც გადასაცემ ინფორმაციას შეიცავს.



სურათი 8. სიგნალის AM მოდულაცია

გადამტან სიგნალად გამოიყენეთ სინუსოიდა ამპლიტუდით 10 და სიხშირით 50 kHz. გადასაცემ სიგნალადაც, ამ შემთხვევაში სინუსოიდა გამოიყენეთ, რომლის ამპლიტუდა არის 0.1, სიხშირე კი 2 kHz. ორივე შემთხვევაში მიუთითეთ sampling time:  $10^{-6}$ . ორივე სინუსოიდა უწყვეტი დროის სინუსოიდაა, დისკრეტიზაციის დროის მითითება spectrum analyzer-ის გამოსაყენებლად გვჭირდება.

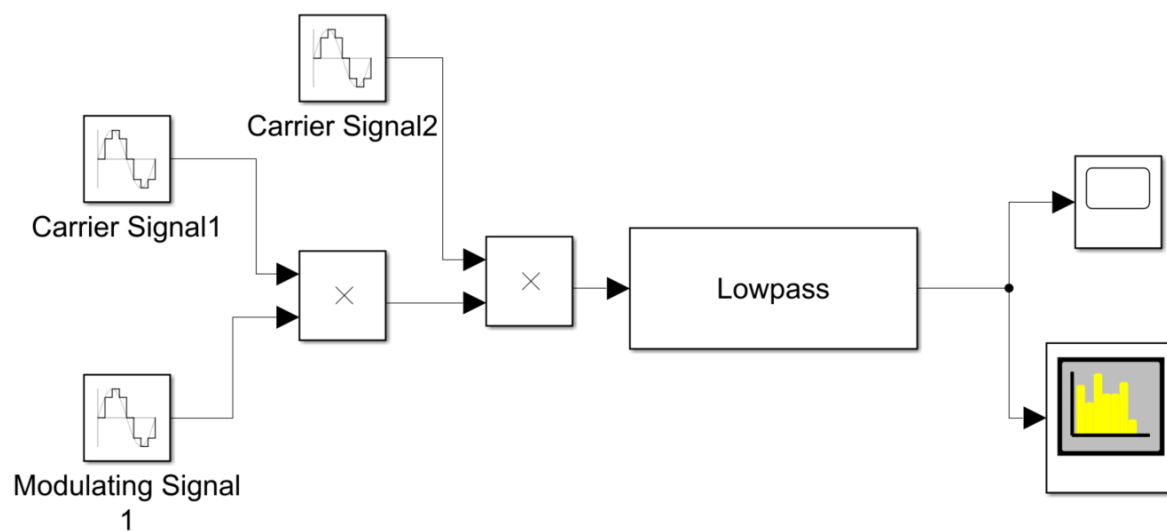
დააკვირდით ოსცილოგრაფის და სპექტრის სიგნალებს. რატომ აქვს თითოეულ მათგანს ასეთი ფორმები? რა იწვევს მოდულირებული სიგნალის სპექტრის წვეროების ორად გაყოფას?

## 4.2 სიგნალის დემოდულაცია

გადაცემული სიგნალის აღდგენისთვის გვჭირდება სიგნალის დემოდულაცია. დემოდულაციის სქემა სურათ 9-ზეა ნაჩვენები. დემოდულაცია, მოდულაციის მსგავსად, გადამტან სიგნალზე გამრავლებით არის შესაძლებელი. შემდეგ დაბალსიხშირული ფილტრის გამოყენებით გადაცემულ (baseband) სიგნალს აღვადგენთ.

დისკრეტიზატორის სიხშირედ მიუთითეთ გადამტანი სიგნალის სიხშირე (50 kHz).

გამოიყენეთ დაბალსიხშირული ფილტრი, რომელიც 2 kHz სიხშირეზე მცირედით მაღალ სიხშირეებს გაატარებს. გამოიკვლიეთ დემოდულირებული სიგნალის სპექტრი. შეადარეთ ის გადაცემული სიგნალის სპექტრს.



სურათი 9. AM სიგნალის დემოდულაცია.