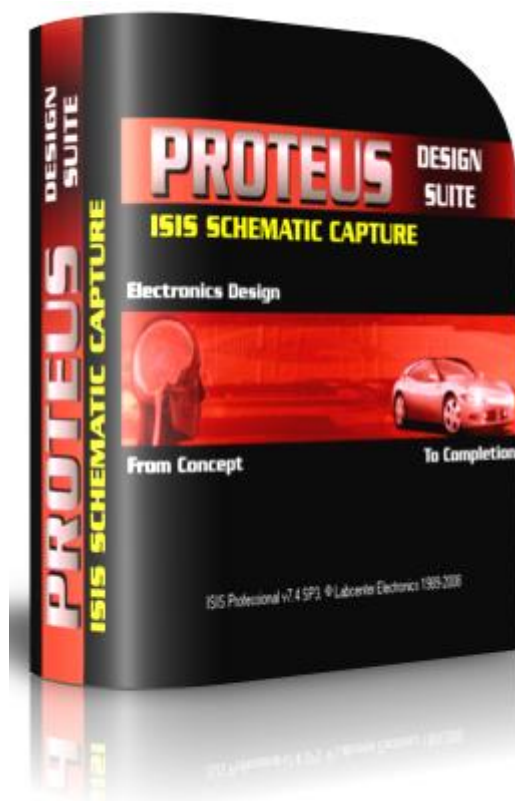


საქართველოს ტექნიკური უნივერსიტეტი

ზ. ჯოჯუა

ვირტუალური მოდელირების სისტემა
PROTEUS VSM
(საფუძვლები)



თბილისი

2021

სახელმძღვანელოში განხილულია ვირტუალური მოდელირები სისტემა **PROTEUS VSM**-ის შემადგენელი ნაწილის **ISIS**-ის ფუნქციონირებისა და გამოყენების ძირითადი საკითხები. **ISIS** განკუთვნილია ელექტრონული სქემების რეალურ დროში მოდელირებისათვის და გააჩნია პირველადი რადიოელემენტებისა და ფუნქციონალურად დასრულებული სქემატექნიკური კვანძების: ლოგიკური ელემენტების, რეგისტრების, მთვლელების, დეშიფრატორების, ამჟამავების, მიკროპროცესორების, მიკროკონტროლერების, ციფრული ინდიკაციის ელემენტების და სხვა მსგავსი ციფრული მოწყობილობების მოდელის მდიდარი ბილიოთეკა. ბილიოთეკა მდიდარია აგრეთვე ანალოგური მოწყობილობების მოდელებითაც: ოპერაციული მაძლიერებლები, ციფრულ-ანალოგური და ანალოგურ-ციფრული გარდამქმნელები და სხვა ანალოგიური კომპონენტები.

სახელმძღვანელოში მოცემულია სისტემის კომპიუტერში ჩამოტვირთვის, გაშვების და ინტერაქტიური ურთიერთქმედების, ე. წ. სამომხმარებლო ინტერფეისის საკითხები. კერძოდ მოცემულია: ბრძანებათა მენიუს აღწერა; საოპერაციო ინსტრუმენტების-პიქტოგრამების აღწერა; ელემენტთა გამოძახება ბილიოთეკიდან და მათი განთავსება სამუშაო დაფაზე; კომპონენტების ურთიერთდაკავშირება როგორც ინდივიდუალური, ისე სალტური პრინციპით; ნებისმიერი კომპონენტის იდენტიფიცირება და პარამეტრების რედაქტირება; პროგრამულად მართვადი კომპონენტებისთვის პროგრამის მიღება, კომპილირება და გამართვა სქემატექნიკასთან კომპლექსურ რეჟიმში, რეალურ დროში.

ამათან, **ISIS** უზრუნველყოფს ელექტრონული კავშირების ცხრილის შექმნას, რომლის საფუძველზე **ARES** (**PROTEUS VSM**-ის მეორე ნაწილი) ადგენს ბეჭდური ფირის ნახაზს და დაპროექტებული მოწყობილობის ნახაზს 3D ფორმატში.

სახელმძღვანელო განკუთვნილია საქართველოს ტექნიკური უნივერსიტეტის სტუდენტებისათვის, მაგრამ მიგვაჩნია, რომ ის სასრგებლო იქნება პირთათვის, რომლებიც დაინტერესებული არიან ელექტრონული სქემების დამუშავების და გამოყენების საკითხებით.

შევნიშნავ, რომ წიგნში განხილულია პროტეუსის **Proteus 7.10** ვერსია, რომელსაც ავტორი იყენებდა წლების განმავლობაში. **Proteus**-ის შემდგომი ვერსიის - **Proteus 8**-ის უფასოდ და უსაფრთხოდ გადმოტვირთვა ავტორმა ვერ მოახერხა, მაგრამ **Proteus 7.10** სრულიად საკმარისია სისტემის მუშაობის პრინციპების და გამოყენების თავისებურებების შესასწავლად. მით უმეტეს, რიგი ავტორები სტატიებში მიუთითებენ, რომ მათ სრულიად აკმაყოფილებთ მათთვის ჩვეული, ფუნქციონალურად სრული **Proteus 7.10** ვერსია და არ აპირებენ მერვე ვერსიაზე გადასვლას.

ზოგადი საკითხები

PROTEUS VSM (Virtual System Modelling - ვირტუალური მოდელირების სისტემა) წარმოადგენს ავტომატიზირებული პროექტირების სისტემას, რომელიც გამოიყენება დისკრეტული და ანალოგური სქემების და მათ საფუძველზე აგებული ელექტრული და ელექტრონული მოწყობილობების მოდელირებისათვის.

სისტემა შექმნილია დიდი ბრიტანეთის ფირმა Labcenter Electronics მიერ SPICE3F5 ბირთვის ბაზაზე (**SPICE** - Simulation Program with Integrated Circuit Emphasis-ინტეგრალურ სქემებზე აქცენტირებული სიმულაციის პროგრამა), რომელიც თავის მხრივ დამუშავებულია კალიფორნიის უნივერსიტეტის მიერ ბერკლიში. სისტემა განკუთვნილია გამჭოლი პროექტირებისთვის-დაწყებული პრინციპიალური სქემიდან და დამთავრებული ბეჭდური ფირის ნახაზით.

სისტემა შედგება ორი ნაწილისაგან: **ISIS**(Intelligent Schematic Input System-სქემების შეტანის ინტელექტუალური სისტემა)-ელექტრონული სქემების რეალური დროის რეჟიმში დამუშავებისა და გამართვისათვის და **ARES**(Advanced Routing and Editing Software-განვითარებული პროგრამული უზრუნველყოფა ტრასირებისა და რედაქტირებისათვის) -ბეჭდური ფირების დამუშავებისათვის.

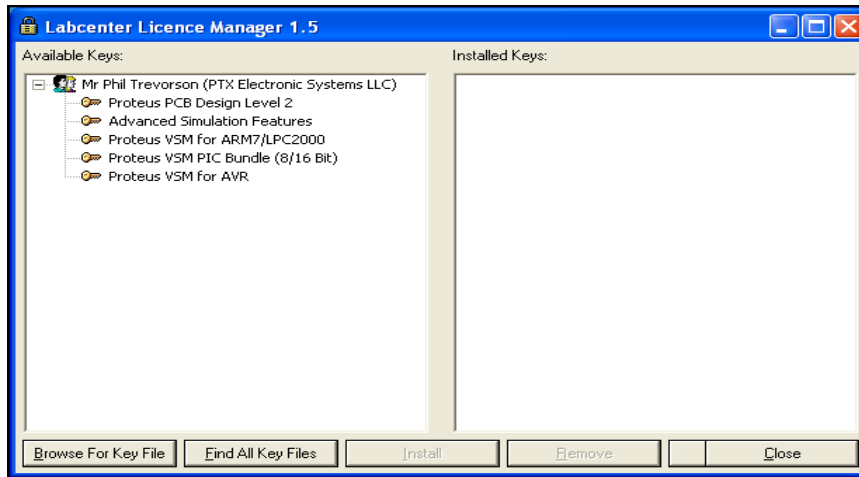
რამდენადაც ჩვენი ძირითადი ამოცანაა ელექტრონული სქემების და მათზე აგებული მოწყობილობების მოდელირება, მიზანშეწონილია **ISIS** პროგრამის ფუნქციონირებისა და გამოყენების შესწავლა. ამასთან, მასში გათვალისწინებულია მონაცემების გადაცემა **ARES** პროგრამისათვის ბეჭდური ფირის დასამუშავებლად.

როგორც აღნიშნავენ, სხვა ანალოგიურ სისტემებთან შედარებით **PROTEUS**-ს გააჩნია რიგი უპირატესობები: სიმულაციის განვითარებული სისტემა გავრცელებული მიკროკონტროლერებისთვის- **8051(Intel)**, **PIC (Microchip)**, **AVR (Atmel)** და სხვა; ელექტრონული და ელექტრული კომპონენტების, პერიფერიული მოწყობილობების მოდელის ფართო ბიბლიოთეკა; გაზომვის, შეფასებების და მონიტორინგის განვითარებული ვირტუალური სისტემები, და ამასთან ეს სისტემა მუდმივ გავითარებაშია.

სისტემა **PROTEUS VSM** ხელმისაწვდომია საიტზე <http://www.labcenter.co.uk/> . მისი მოძიება შეიძლება Google-ში(ან სხვა საძიებო სისტემაში) სახელებით **Proteus VSM** ან **Proteus ISIS**.

მიზანშეწონილია გადმოიტვირთოს პროფესიონალური ვერსიები Proteus v.7 ან შემდგომი. გადმოტვირთული პროგრამა ავტომატურად თავსდება დირექტორიაში: **Program Files\ Labcenter Electronics\ Proteus 7** ან **Program Files(x86)\ Labcenter Electronics\ Proteus 7**, მაგრამ მათი შეცვლა შეიძლება. შემდეგ საჭიროა საინსტალაციო პაკეტის გაშვება **Setup.exe**, რომელიც მოითხოვს ლიცენზიას, რომლის გადალახვა შეიძლება ლიცენზიის მენეჯერის გამოყენებით: **Start=>All Programs =>Proteus x Professional=>Licence Manager**.

მარცხენა ფანჯარაში ღილაკზე **Browse For Key File** (ხელით) ან **Find All Key File** (ავტომატურად) დაჭერა გამოიყვანს ლიცენზიის ფაილების სიას მარცხენა ფანჯარაში, სადაც საჭირო ლიცენზიის ამორჩევა გააქტიურებს ღილაკ **Install**-ს, რომელზეც დაჭერა მარჯვენა ფანჯარაში გამოიტანს შესაბამის ინფორმაციას. საყურადღებოა, რომ



Licence Manager-ის ფანჯარა

გასაღებების პირდაპირ მითითებულია პროგრამის ფუნქციები, რომელზეც ვრცელდება ამორჩეული ლიცენზია.

პროგრამის ნორმალური ფუნქციონირებისთვის მიზანშეწონილია ყველგან ვიხმართ მხოლოდ ინგლისური ასოები. ამასთან, პროტეუსს გააჩნია მრავალსაფეხურიანი დაცვა არალეგალური გამოყენების წინააღმდეგ-დაცულია არა მარტო ფაილები არამედ ცალკეული მოდულები. ამიტომ, არაოფიციალური, ე. წ. **crack**-ვერსიების გამოყენებისას შესაძლებელია პროგრამის მუშაობაში წარმოიშვას პრობლემები. საჭიროა crack-ვერსიების სწორად გამოყენება.

Proteus-ის ახალი ვერსიის **Proteus 8**-ის უფასოდ და უსაფრთხოდ გადმოტვირთვა ავტორმა ვერ მოახერხა, მაგრამ **Proteus 7.10** სრულიად საკმარისია სისტემის მუშაობის ძირითადი პრინციპების და გამოყენების თავისებურებების შესასწავლად. მით უმეტეს, რიგი ავტორები სტატიებში მიუთითებენ, რომ მათ სრულიად აკმაყოფილებთ მათთვის ჩვეული **Proteus 7.10** და არ აპირებენ მერვე ვერსიაზე გადასვლას.

Proteus 7.10-ის დაყენება

მასალა აღებულია რუსული სტატიიდან "Урок 4. Установка Proteus" (ვებგვერდი <http://hamlab.net/mcu/training/proteus.html>), რომელშიც მოცემულია წვდომა **Proteus Professional 7.10 + Crack**. მასზე დაწკაპუნებით გაიხსნება გვერდი <https://dfiles.ru/files/42iio8i9m>, საიდანაც მოწოდებული ინსტრუქციების მიხედვით ჩამოტვირთავთ **Proteus 7.10** - ის არქივს. ვხსნით ამ არქივს და ვღებულობთ: ორ საქაღალდეს-«**Help**» და «**Patch**», ტექსტურ ფაილს «**Установка**» (დაყენება) და დაყენების ფაილს «**P710sp0.exe**», რომელიც უნდა გავუშვათ.

Имя	Тип	Размер	Дата
[..]	<Папка>		16.03.2013 12:06
[Help]	<Папка>		23.01.2012 23:45
[Patch]	<Папка>		27.02.2012 03:05
P710sp0	exe	109 466 544	25.02.2012 16:24
Установка	txt	160	27.02.2012 00:25

გაშვების შემდეგ ვლებულობთ ფანჯარას:



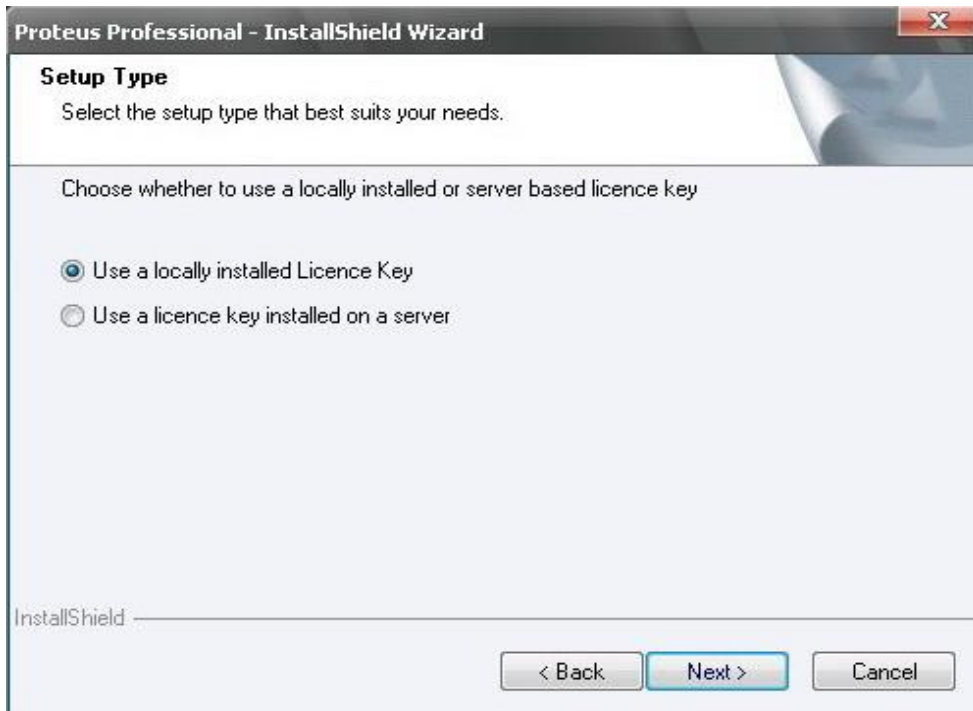
ვაჭერთ ღილაკზე **Next**.

შემდეგ ფანჯარაში მოგეთხოვებათ ლიცენზიის პირობებზე გაცნობა:



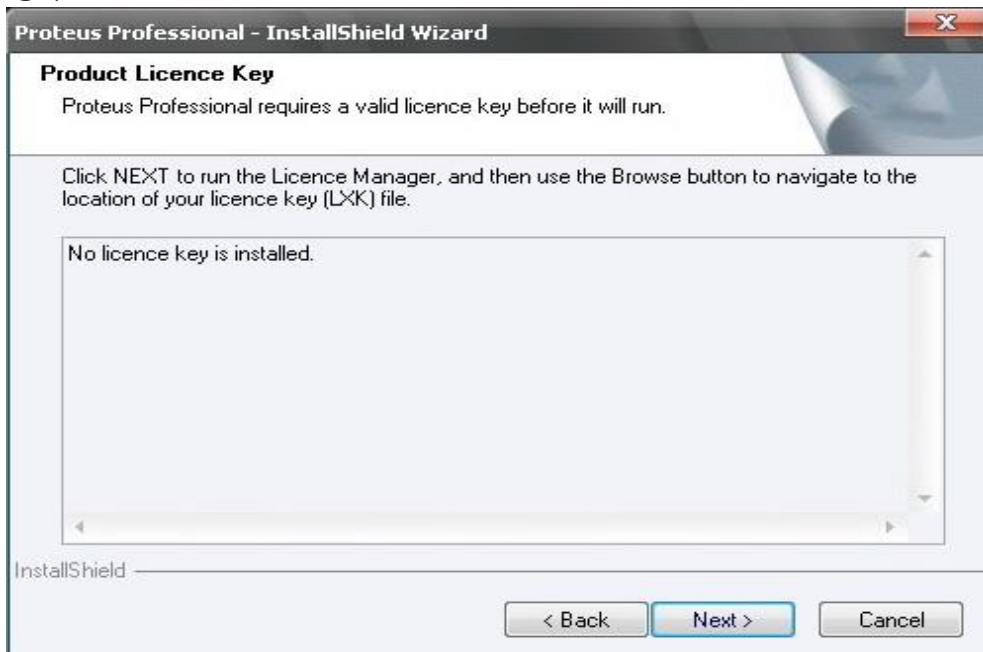
პირობებს ვეთანხმებით და ვადასტურებთ **Yes** ღილაკზე დაჭერით.

შემდგომ ფანჯარაში მისათითებელია ადგილი, სადაც ინახება სალიცენზიო გასაღები:



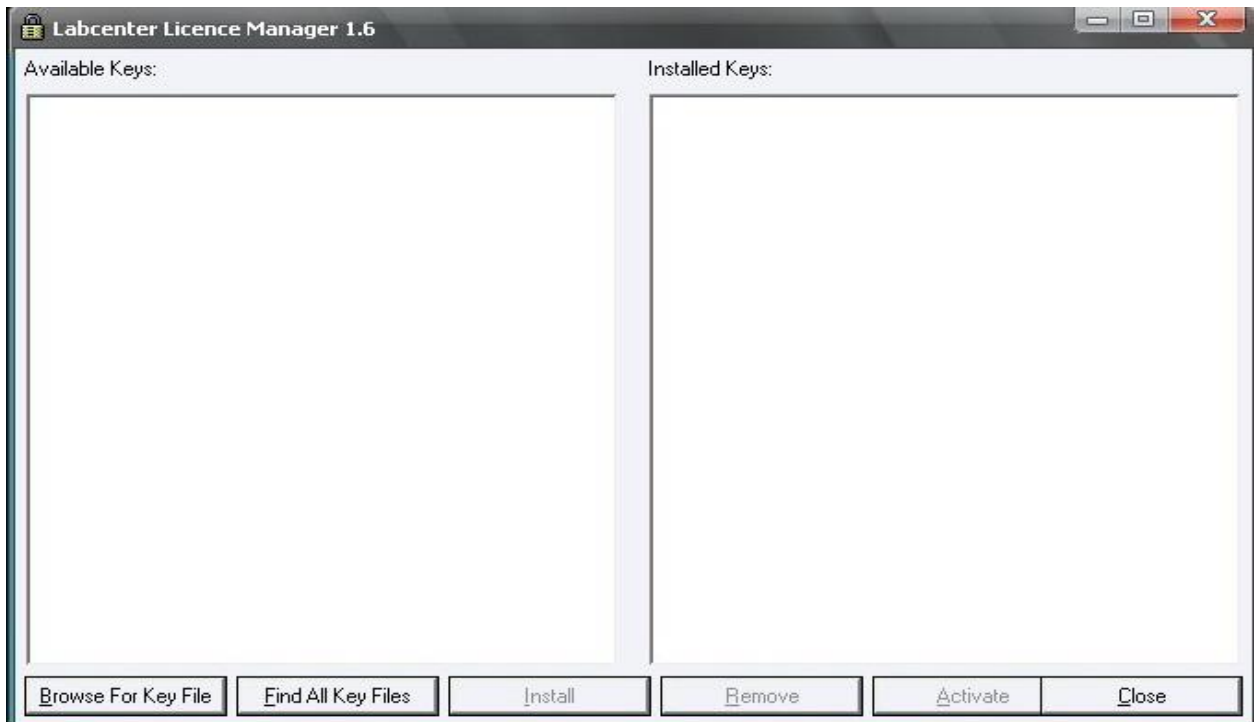
აქ გადამრთველი უნდა დაყენდეს ნიშანზე: «**Use a locally installed Licence Key**», რათა გამოყენებული იქნეს გასაღები შენახული საქალაქო «Patch». ვაჭერთ ლილაკზე **Next**.

გამოჩნდება ფანჯარა, რომელიც გვაცნობებს, რომ სალიცენზიო გასაღები დაყენებული არ არის.



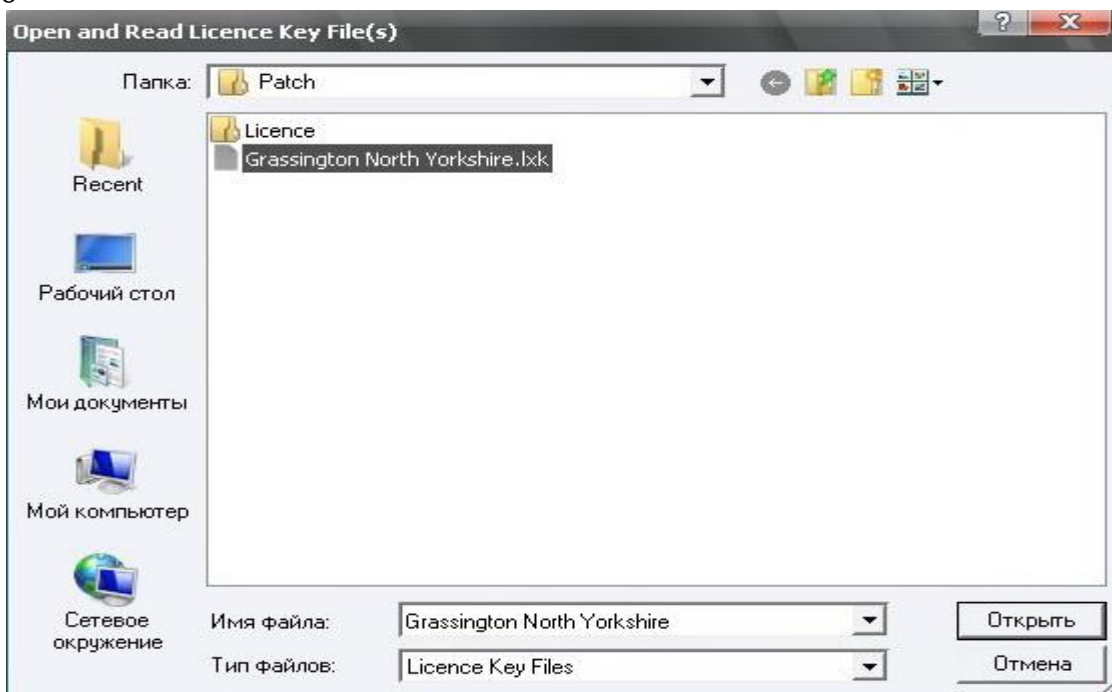
ვაჭერთ ლილაკზე **Next**.

გამოვა ფაჯარა **Labcenter Licence Manager**, რომელშიც შესაძლებელია საკუთარი სალიცენზიო გასაღების დამატება.

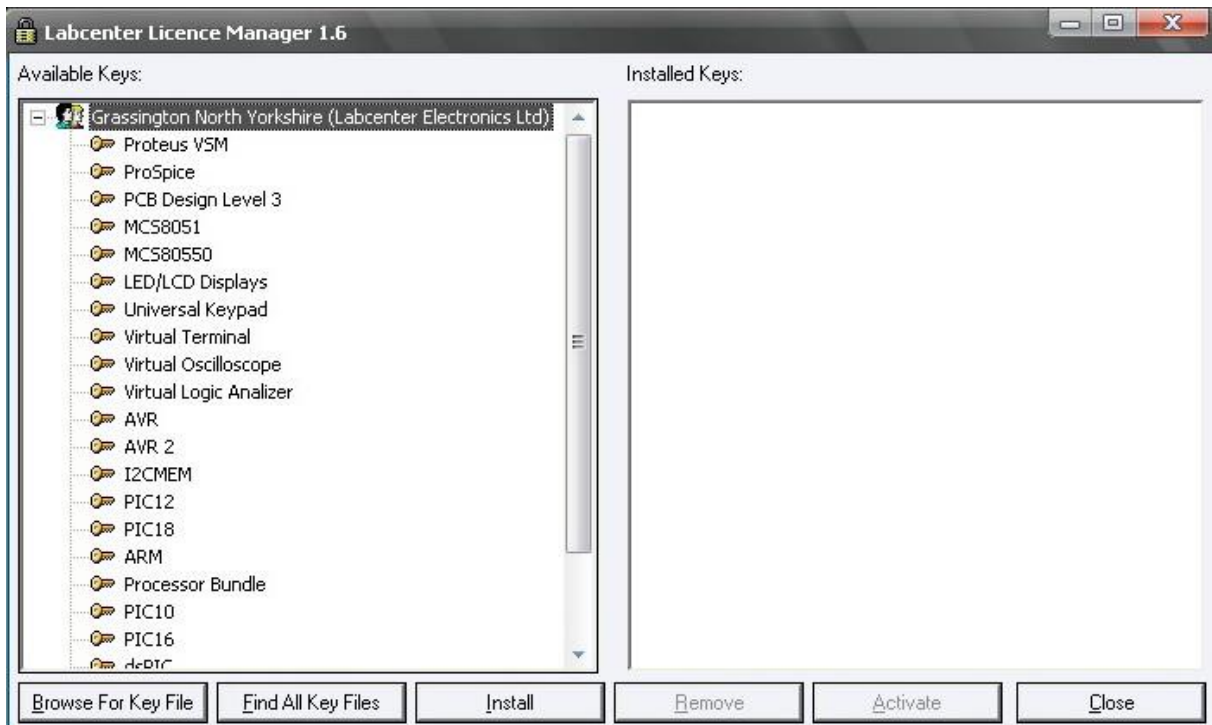


ვაჭერთ ღილაკზე **Browse For Key File**.

ახალ ფაჯარაში, **Patch** საქალაქდემი ვირჩევთ ფაილს **Grassington North Yorkshire.lxk** და ვხსნით მას.

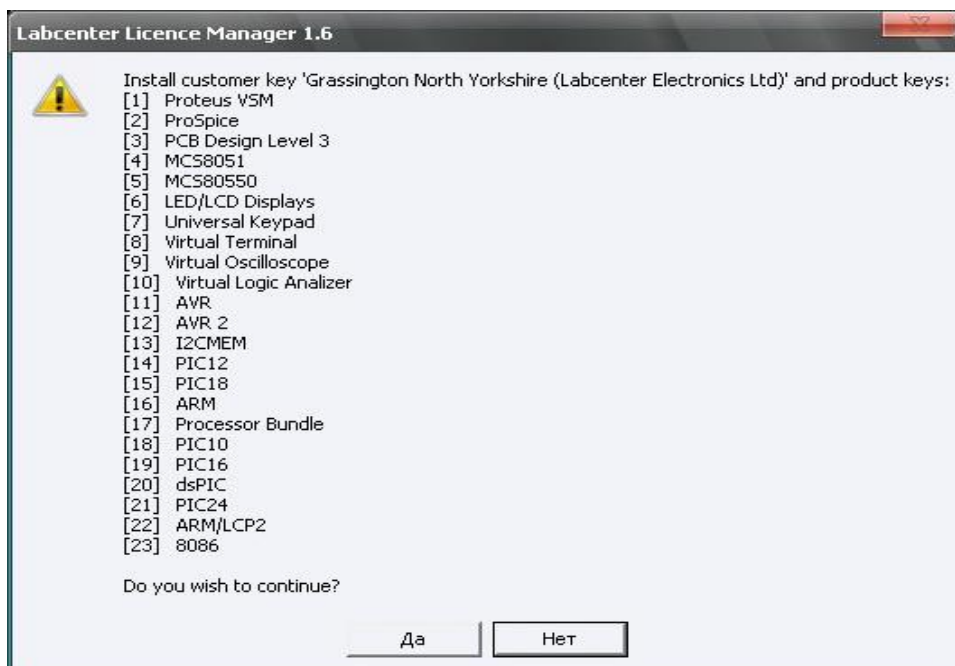


Licence Manager-ის მარცხენა ფაჯარაში- **Available Keys**, გამოჩნდება ინფორმაცია გამოყენებულ გასაღებზე.



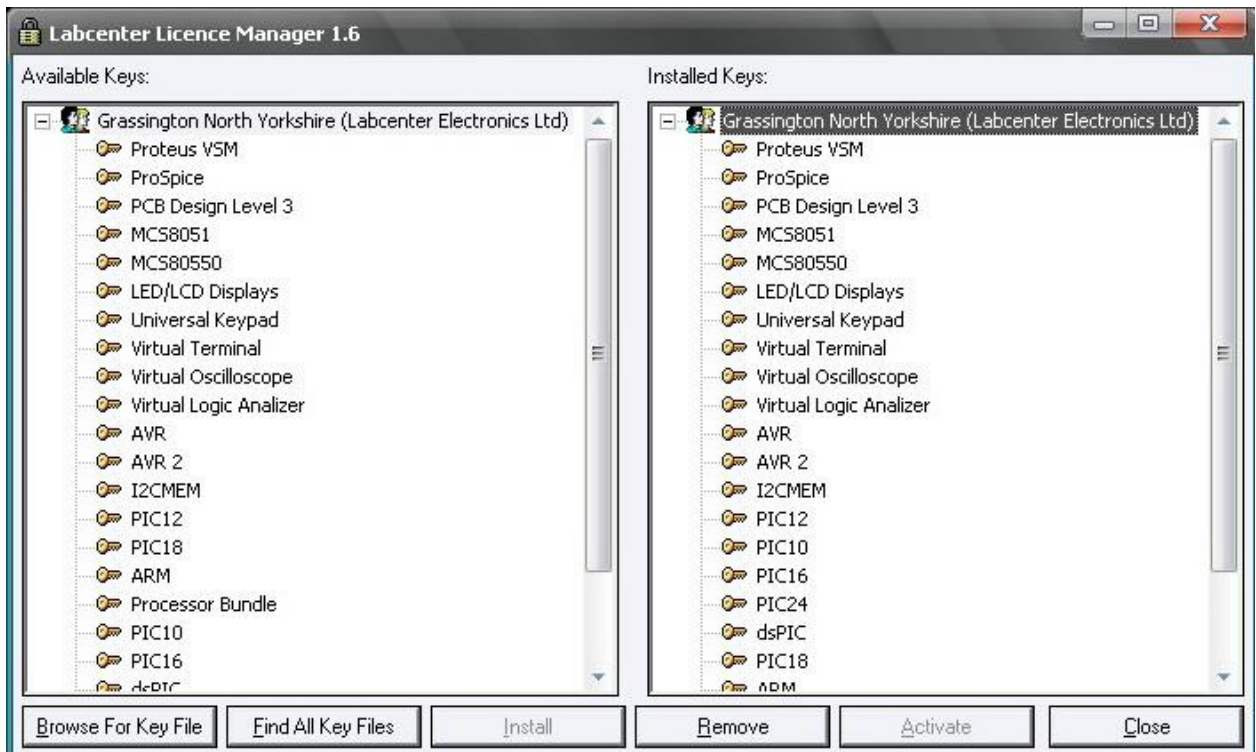
ვაჭერთ **Grassington North Yorkshire (Labcenter Electronics Ltd)** სახელზე, რაც იწვევს **Install** ღილაკის გააქტიურებას. ვაჭერთ მასზე.

შემდეგ ფაჯარაში გვეკითხებიან: მართლა გვინდა ამ გასაღების დაყენება.



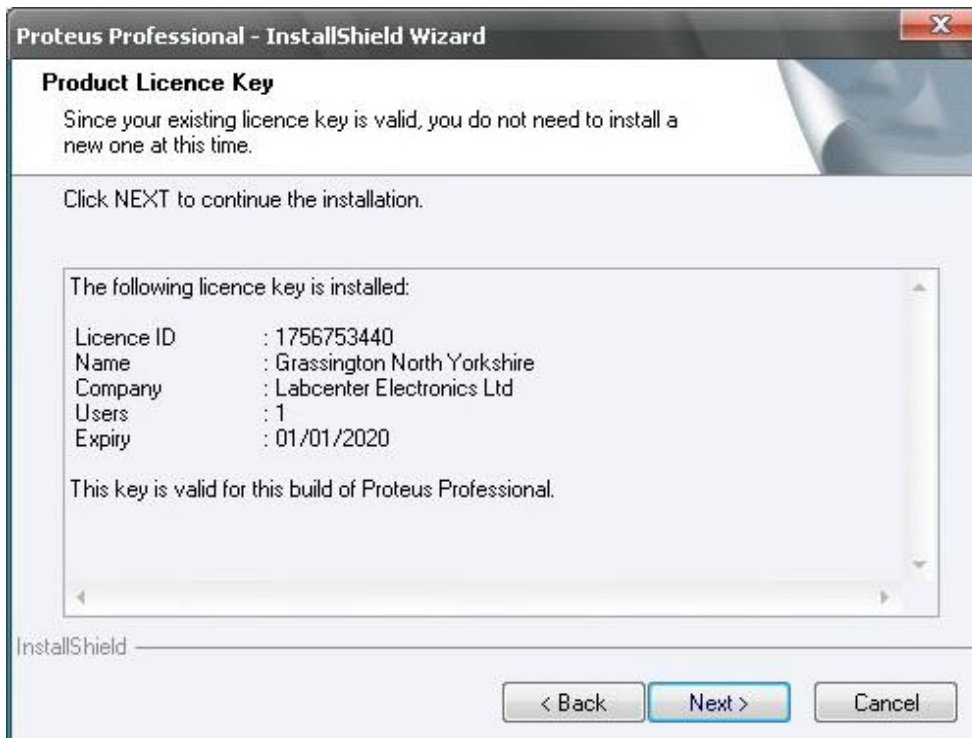
ვაჭერთ კი-ს.

შედეგად **Licence Manager**-ის მარჯვენა ფაჯარაში გამოჩნდება ისეთივე სია როგორც მარცხენაში.



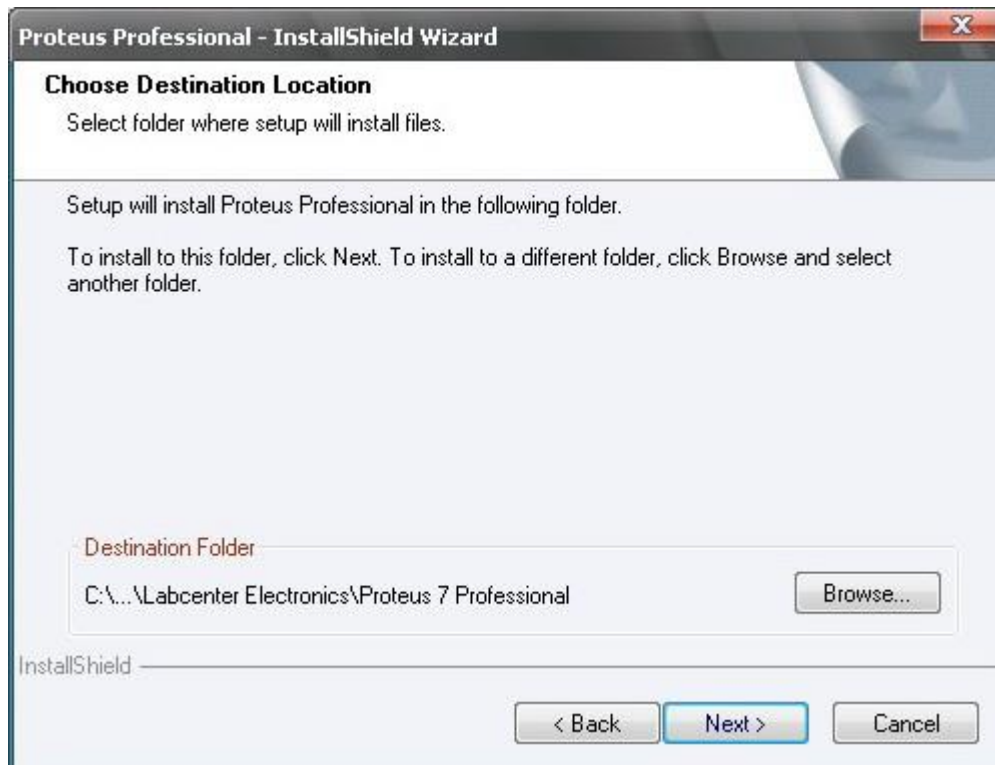
ვაჭერთ ღილაკზე **Close**.

მომდევნო ფაჯარაში ვნახავთ ინფორმაციას დაყენებული ფანჯრის შესახებ.



ვაჭერთ ღილაკზე **Next**.

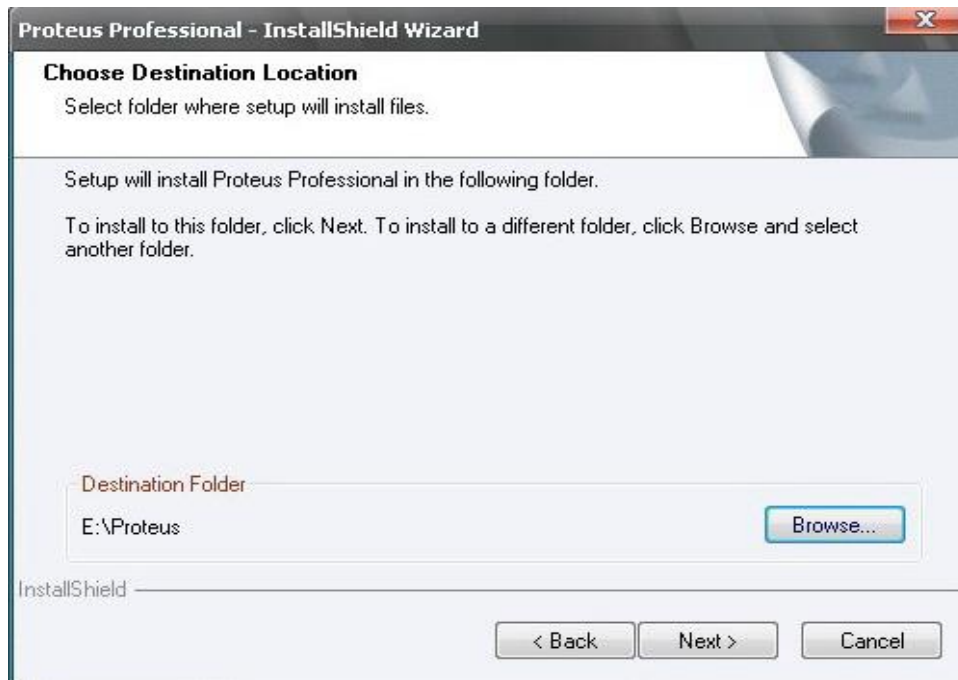
შემდეგ ჯერზე ვირჩევთ პროგრამის დასაყენებელ გზას.



მიუთითებლად(ავტომატურად) პროგრამა დგება საქალაქდებში: **C:\Program Files\Labcenter Electronics\Proteus 7 Professional** ან **C:\Program Files(x86)\Labcenter Electronics\Proteus 7 Professional**. შეიძლება მისი შეცვლა, მაგ. დაყენება **E** დისკზე. ამისთვის ვაჭერთ ღილაკზე **Browse**. დიალოგურ ფაჯარაში ვუზღვენებთ დაყენების ადგილს.

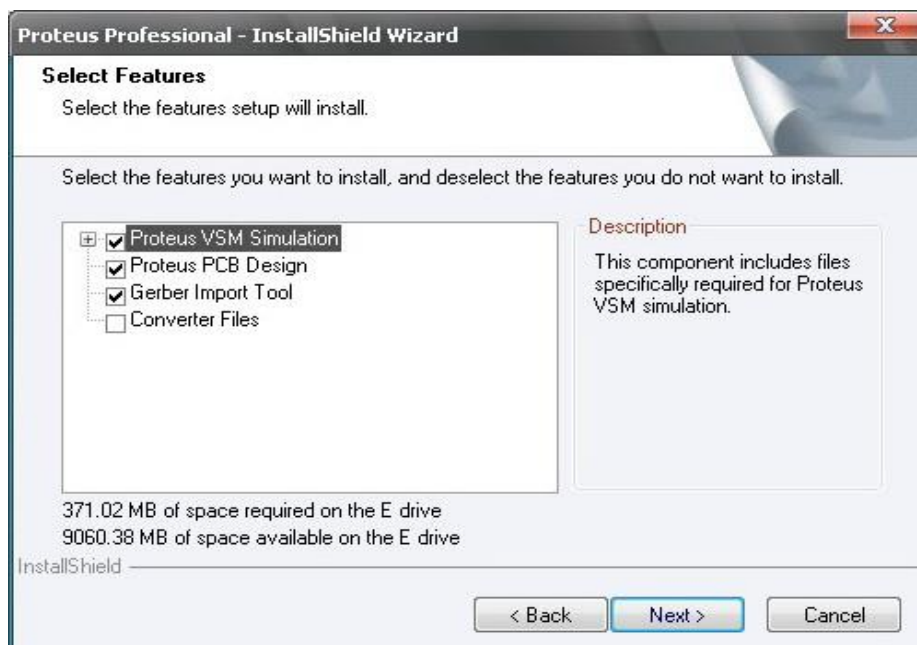


ეს არის საქალაქდებ **Proteus E** დისკზე. ვაჭერთ ღილაკზე **OK** რითაც გზა ნაჩვენებია.



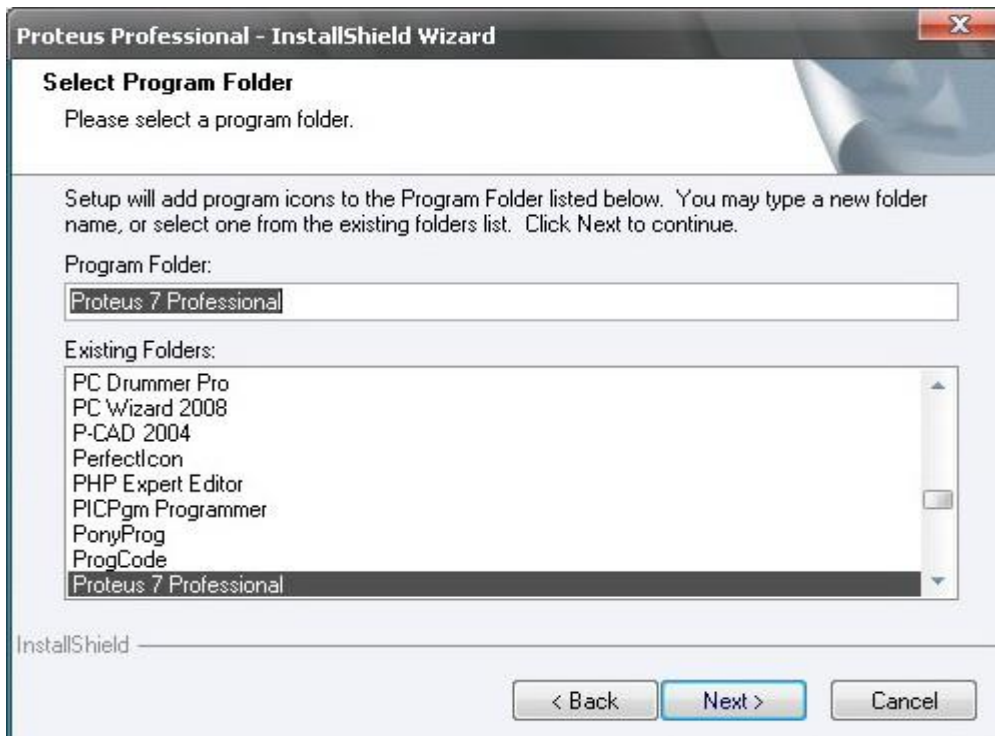
ვაჭერთ ღილაკზე **Next**.

შემდეგ ფანჯარაში ვირჩევთ საჭირო ინსტრუმენტებს. გაჩუმებით ამორჩეულია სამი, მეოთხე- **Converter Files** ჯერ საჭირო არ არის, საჭიროებისას შეიძლება დამატება. აქვე ნაჩვენებია **E** დისკზე დაკავებული და თავისუფალი ადგილების რაოდენობა.



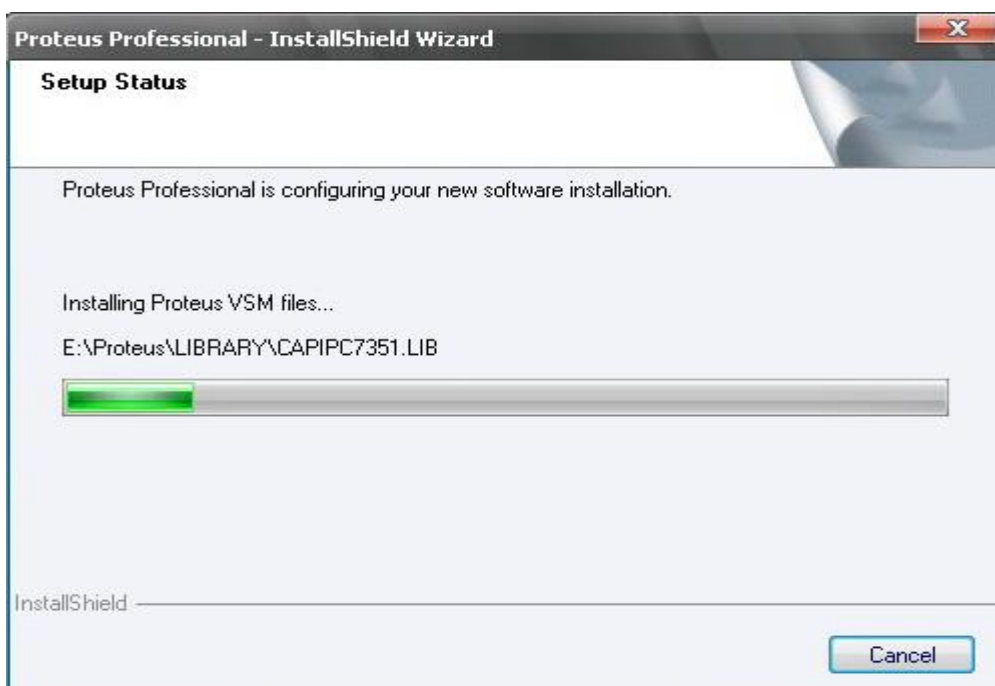
ვაჭერთ ღილაკზე **Next**.

შემდეგ ფანჯარაში ამორჩევა საქალაქის სახელი სადაც განთავსებული იქნება **Proteus**-ი, მენიუ **Start**-ის **All Programs** განყოფილებაში. ვტოვებთ შემოთავაზებულს.

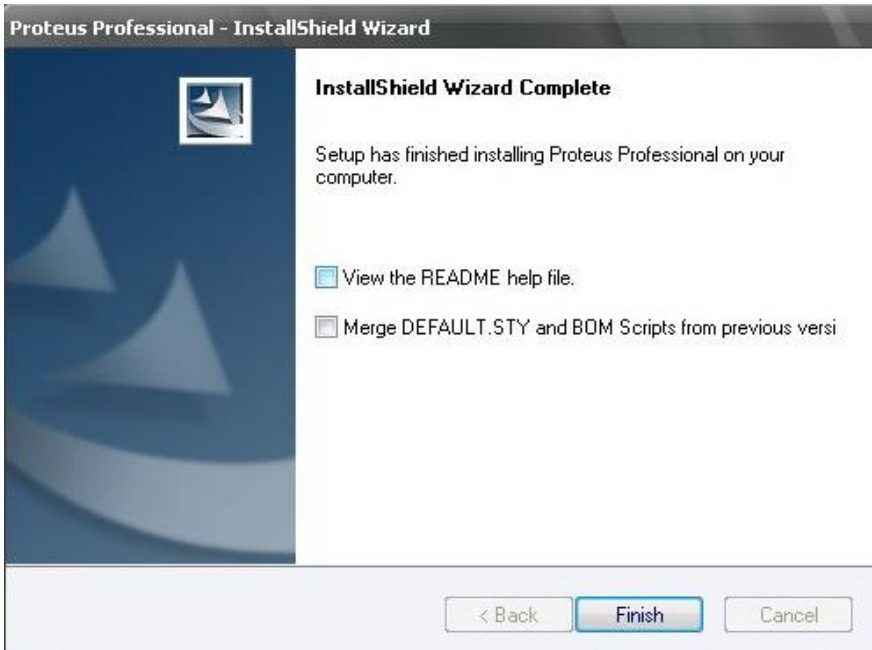


ვაჭერთ ღილაკზე **Next**.

დაიწყება პროგრამის დაყენება.



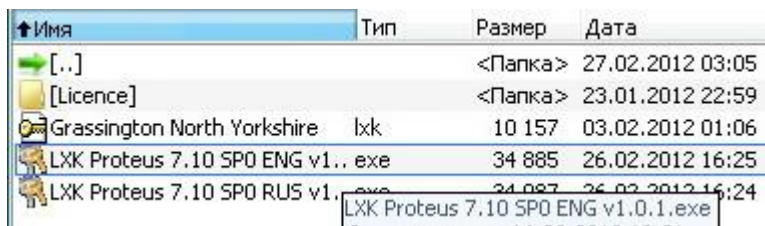
დაყენების დამთავრების შემდეგ, გამოჩნდება ფანჯარა



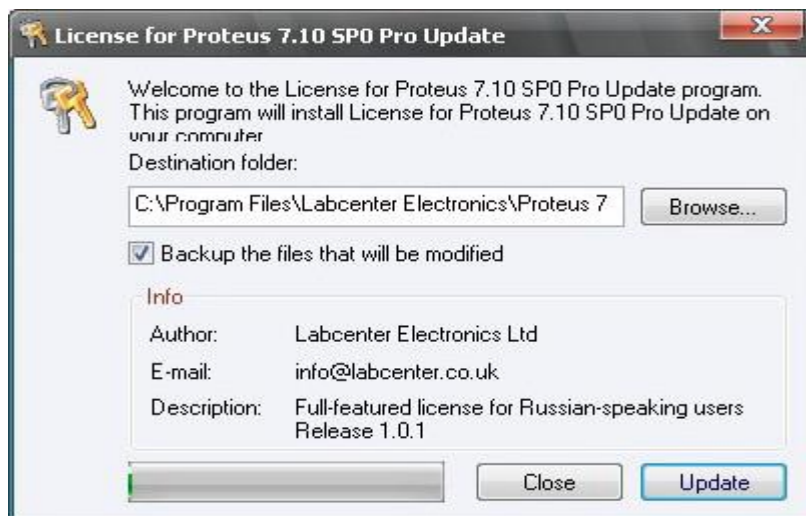
ვაჭერთ ღილაკზე **Finish**.

ამით **Proteus 7 Professional** -ის დაყენება დამთავრებულია.

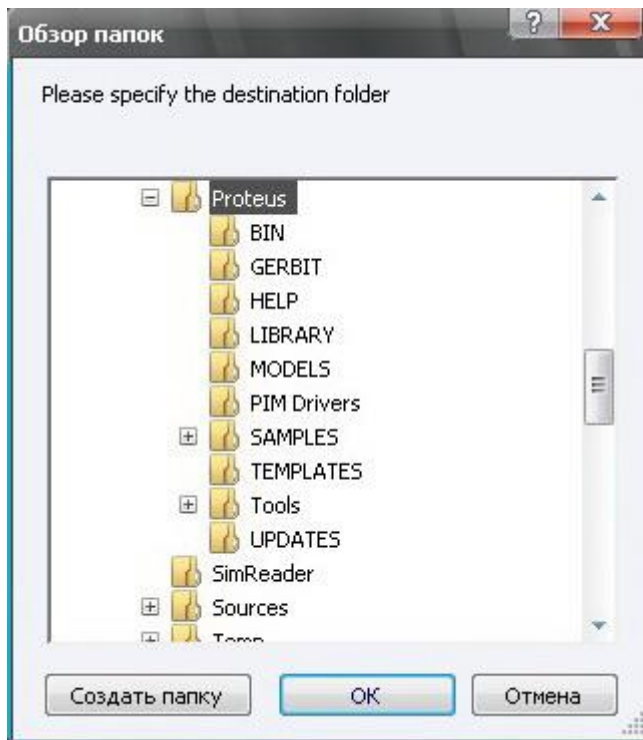
ახლა საჭიროა **Proteus** -ის გააქტიურება. შევდივართ საქალაქო Patch და ვუშვებთ პროგრამას **LXK Proteus 7.10 SP0 ENG v1...exe** .



გამოჩნდება ფანჯარა

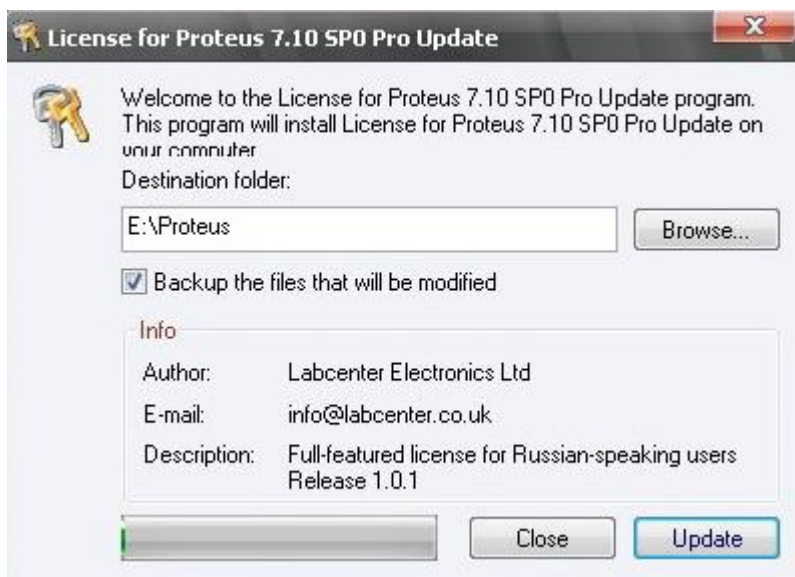


ვაჭერთ ღილაკზე **Browse**, რათა შევცვალოთ სტანდარტული გზა ახლით:



ვაჭერთ **OK**.

Destination Folder გამოიხატება ახალი გზა პროგრამისკენ-**E:\Proteus**.



ვაჭერთ ლილაკზე **Update**. დაიწყება განახლების პროცესი.

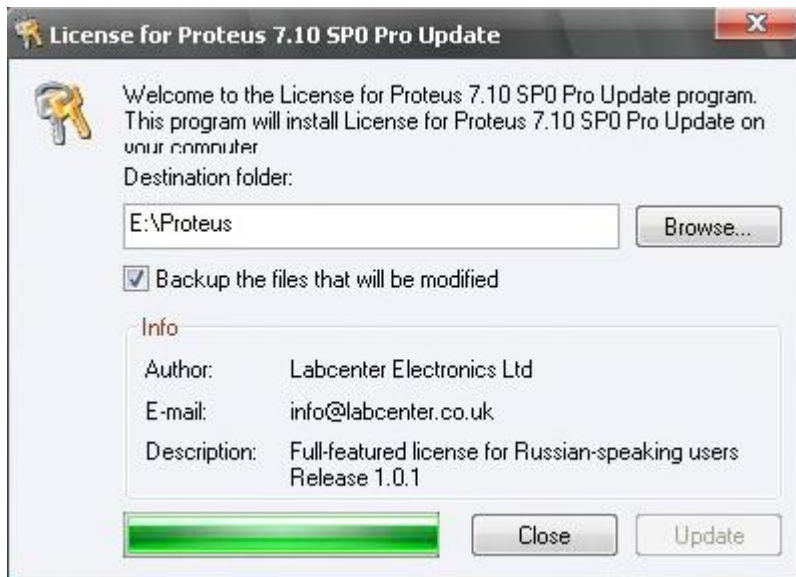


ბოლოს გამოჩნდება პატარა ფანჯარა შეტყობინებით, რომ განახლების პროცესი წარმატებით დამთავრდა.



ვადასტურებთ-ვაჭერთ **OK**.

სრულად გამოჩნდება ძირითადი, ბოლო ფანჯარა.



ვაჭერთ **Close** და ვხურავთ ამ ფანჯარას.

ამით **Proteus 7**-ის დაყენების პროცესი დამთავრებულია.

ახლა საჭიროა გავუშვათ მოდელირების პროგრამა **ISIS** შემდეგი გზით:

Start - All Programs - Proteus 7 Professional– ISIS 7 Professional. ამის შემდეგ **ISIS 7 Professional** ავტომატურად დაჯდება **Start**-ის ფანჯარაში.

ISIS-ის გაშვების შემდეგ გამოჩნდება პროექტების დიალოგური ფანჯარა.

დიალოგური ფანჯარასთან მუშაობის წესები მოცემულია ქვევით.

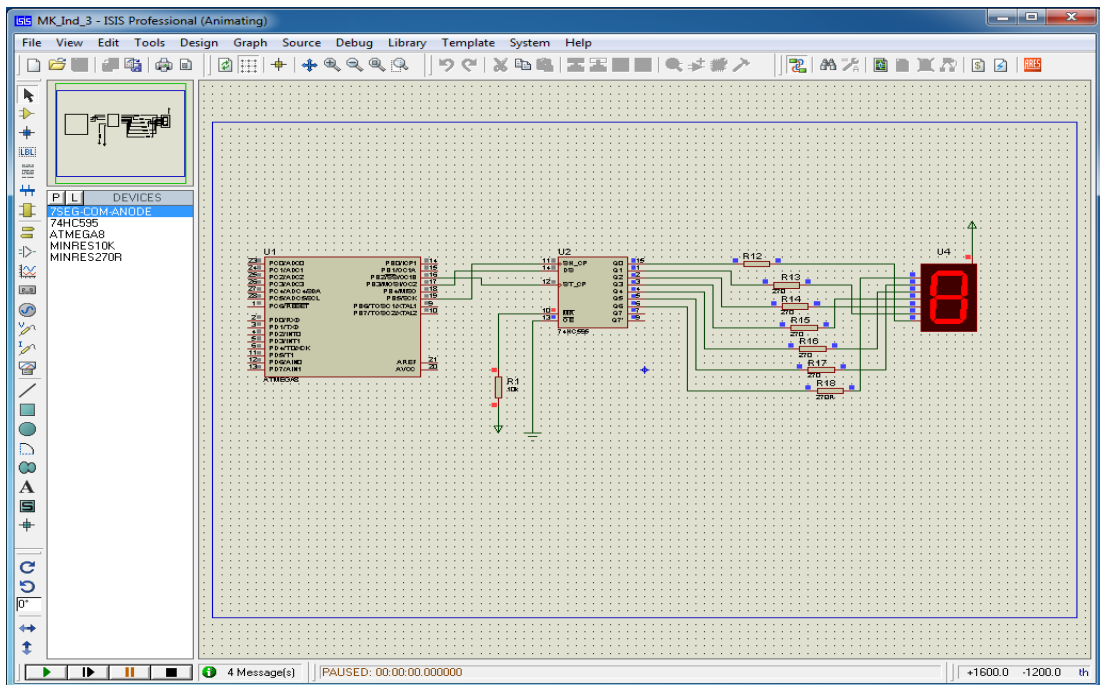
PROTEUS VSM არ არის კრიტიკული კომპიუტერული რესურსების მიმართ. თავისუფლად მუშაობს **Windows 2k/XP/7** ვერსიებში, პროცესორის სიხშირე საკმარისია 1Ghz-იც კი, მაგრამ დიდი ამოცანებისთვის უკეთესია უფრო მაღალი სიხშირეები; ოპერატიული მეხსიერების მოცულობა საკმარისია 500 MB-ის ფარგლებში, მაგრამ რაც მეტი მით უკეთესი; მონიტორის გარჩევადობა საკმარისია 1024x768 წერტილი. რამდენადაც თანამედროვე კომპიუტერები ყოველმხრივ აკმაყოფილებენ ამ მოთხოვნებს მომხმარებელს ამ მხრივ პრობლემები არ უნდა შეექმნას.

თავი 1. ISIS პროგრამის ინტერფეისი

1.1 ეკრანის ველების ორგანიზაცია

ISIS პროგრამის გაშვების შემდეგ ეკრანზე გამოჩნდება ძირითადი ფანჯარა-სურ. 1.2, სადაც ყველაზე დიდი ადგილი უჭირავს რედაქტირების ფანჯარას - **Edit Window**, (რედაქტირების ძირითად ფანჯარას(ველს)). მასზე თავსდება დასამუშავებელი სქემა ან მისი ფრაგმენტი ყველა კომპონენტებით და შესაძლო საკონტროლო ვირტუალური ხელსაწყოებით. ეს არის ერთი ფურცელი. თუ სქემა დიდია ის შეიძლება განთავსდეს რამდენიმე ურთიერთდაკავშირებულ ფურცელზე.

ფანჯარაში სამუშაო ველი შეესაბამება გარკვეულ ფორმატს, რომელიც შემოსაზღვრულია ლურჯი ხაზით, ხოლო მისი ცენტრი მონიშნულია ამავე ფერის წერტილით-კოორდინატები 00:00.



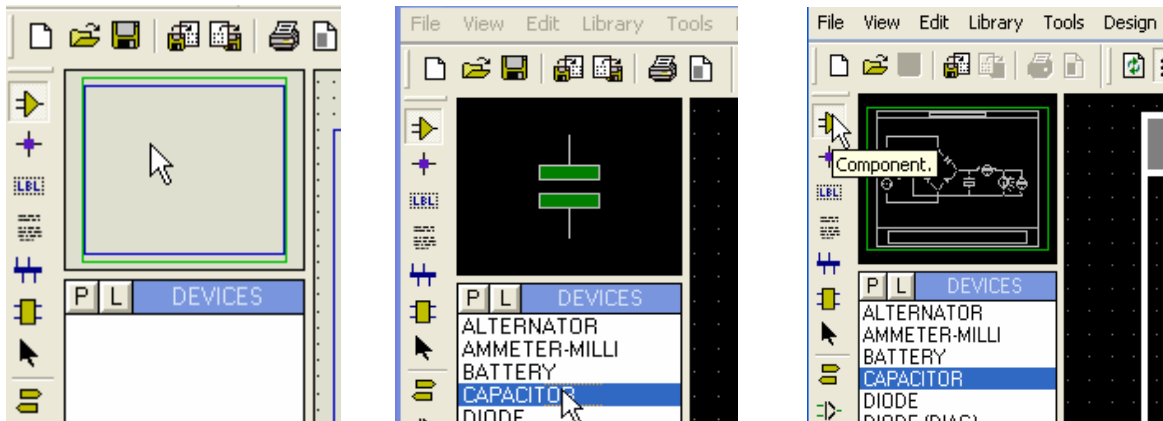
სურ. 1.2.

ეკრანის ზედა მარცხენა მხარეს განთავსებულია პატარა ფანჯარა სქემის ან მისი კომპონენტების თვალიერებისთვის-**Overview Window** - სურ. 1.3.

სურ. 1.3. ა)-ზე ნაჩვენებია თავისუფალი ფანჯარა; ბ)-ზე-კომპონენტის თვალიერების რეჟიმი; გ)-ზე-მთლიანი სქემა შეკუმშული სახით.

რედაქტირების და თვალიერების ფანჯარები ურთიერთდაკავშირებულია ფუნქციონალურად, ამიტომ მათი განხილვა მიზანშეწონილია ერთდროულად.

რედაქტირების ფანჯარაზე შეიძლება მანიპულაციების ჩატარება: ნახაზის მასშტაბის შეცვლა ან ნახაზის ნაწილის მასშტაბის შეცვლა (პანორამირება); ელემენტის ამოღება ან ჩამატება და სხვა.



ა)

ბ)

გ)

სურ. 1.3.


მასშტაბირება ანუ სქემის და შესაბამისად კომპონენტების ზომებში გადიდება ან შემცირება, შეიძლება განხორციელდეს რამდენიმე გზით:


- დააყენეთ თავის მაჩვენებელი გასადიდებელ/შესაკუმშ ადგილზე და დაატრიალეთ თავის გორგოლაჭი წინ(გადიდება) ან უკან(შეკუმშვა).
- დააყენეთ თავის მაჩვენებელი გასადიდებელ/შესაკუმშ ადგილზე და დააჭირეთ **F6** კლავიშას (გადიდება) ან **F7** კლავიშას(შეკუმშვა).
- დაჭერილი **SHIFT** კლავიშის პირობებში თავის მარცხენა კლავიშით გაჭიმეთ სქემის უბნის შემომსაზღვრელი ხაზი(კონტეინერი)-ე. წ. **SHIFT** მასშტაბირება.
- გამოვიყენოთ ინსტრუმენტები გადიდება, შემცირება, გამოვსახოთ მთელი სქემა, გავადიდოთ მონიშნული უბანი.
- კლავიშა **F8** განათავსებს მთელ სქემას სამუშაო ველში.

SHIFT მასშტაბირება და თავის გორგოლაჭი შეიძლება გამოვიყენოთ თვალთვლების ფანჯრის მიმართაც იგივე ეფექტით.

პანორამირება ანუ ეკრანის გადატანა რედაქტირების ფანჯრის შერჩეულ ნაწილზე, ხორციელდება ვარიანტებით:

- თავის შუა ღილაკით დაწკაპუნება ISIS შეიყვანს მიმყოლი პანორამირების რეჟიმში, რასაც აჩვენებს კურსორის შეცვლილი სახე. თავის შემდგომი გადაადგილება დაჭერის გარეშე გამოიწვევს სქემის გადაადგილებას იმავე მიმართულებით. რეჟიმიდან გამოსვლა ხდება მარცხენა ღილაკზე დაჭერით. ამ რეჟიმში შეიძლება მასშტაბის შეცვლა გორგოლაჭით.
- რედაქტირების ფანჯრის ცენტრისკენ გადასაწევა-დააყენეთ კურსორი იმ ადგილზე რომელმაც უნდა გადმოიწიოს ცენტრისკენ და დააჭირეთ **F6**.
- დაჭერილი **SHIFT** კლავიშის პირობებში მიიტანეთ თავის ისარი ფანჯრის კიდესთან-სქემა კონტეინერიანად დაიძვრის საწინააღმდეგო მიმართულებით ანუ, უფრო სწორად, ეკრანი სქემაზე დაიძვრის თავის მოძრაობის მიმართულებით. ისრის ყოველი მიდება კიდესთან გამოიწვევს ძვრას. ეს არის **SHIFT** პანორამირება.

- თუ საჭიროა ეკრანის გადატანა სქემის დაშორებულ ადგილზე - თვალიერების ფანჯრის ნებისმიერ წერტილზე თავის მარცხენა კლავიშის დაწკაპუნება გამოიწვევს რედაქტირების ფანჯრის გადატანას სქემის ამ ნაწილში. ამ მდგომარეობაში თავის თავისუფალი ტარება თვალიერების არეში იწვევს ფანჯრის გადაადგილებას სქემის ნებისმიერ არეში. მარცხენათი მეორეჯერ დაწკაპუნება დააფოქსირებს მიმდინარე მდგომარეობას რედაქტირების ფანჯარში. ეს განსაკუთრებით მოსახერხებელია დიდი სქემების დამუშავებისას, როდესაც მთელი სქემა ვერ თავსდება რედაქტირების ველში. ეს მსგავსია ლუპის გადაადგილებისა ობიექტზე.
- ეკრანის გადაადგილება რედაქტირების არეში შეიძლება განხორციელდეს პიქტოგრამა  გამოყენებით ინსტრუმენტების პანელიდან. მარცხენა კლავიშით დაწკაპუნება აფიქსირებს ეკრანის მდგომარეობას. ეკრანის კიდესათან პიქტოგრამა მოქმედებს SHIFT პანორამირების მსგავსად.

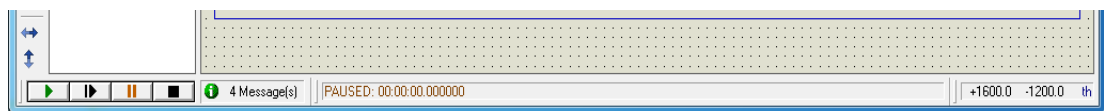
ნახაზების აგების მოხერხებულობისთვის რედაქტირების ფანჯარაში შესაძლებელია საკოორდინატო ბადის გამოყენება მომხმარებლისთვის სასურველი ბიჯებით და ფორმებით. რეჟიმების მართვა შესაძლებელია მენიუდან **View => Grid**, ან პიქტოგრამა  გამოყენებით ინსტრუმენტების პანელიდან, ან კლავიატურიდან კლავიშით **'G'**. ამავე საშუალებებით შესაძლებელია ბადის მოხსნა. შესაძლებელია, აგრეთვე, ბადის კომპონენტების ფერის შეცვლა- **Template => Set Design Defaults**.

თვალიერების ფანჯრის ქვევით განთავსებულია ობიექტების ამორჩევის ფანჯარა- **Object Selector**, რომელსაც ობიექტების კონტინერსაც უწოდებენ. ბიბლიოთეკიდან შერჩეული ყოველი ობიექტის იდენტიფიკატორი თავსდება ამ ფანჯარაში, ხოლო აქ შერჩეული(მონიშნული) ობიექტის სახე (ნახაზი) ავტომატურად თავსდება თვალიერების ფანჯარაში, სადაც მასზე შეიძლება გარკვეული მანიპულაციების ჩატარება. ამის შემდეგ ობიექტი შეიძლება გადატანილიქნეს რედაქტირების ფანჯარაში. სელექტორის გამოყენებას გავეცნობით ქვემოთ.

სელექტორის ფანჯარის თავზე განთავსებულია ორი დილაკი ასოებით-**P** და **L**.

P(Pick Devices) დილაკის გააქტიურება იწვევს ბიბლიოთეკაში კომპონენტების საძიებო ფორმის გამოძახებას. **L**(Library) დილაკი იძახებს ბიბლიოთეკის მენეჯერს, რომლის დახმარებით შესაძლებელია კომპონენტების ახალი ბიბლიოთეკის ჩართვა სისტემაში.

თავის მდებარეობის კოორდინატები ავტომატურად აისახება ინდიკატორში ეკრანის ქვედა მარჯვენა კუთხეში-სურ. 1.4.



სურ. 1.4.

რედაქტირების, თვალიერების და სელექტორის ფანჯრების ზომების შეცვლა შესაძლებელია ჩვეულებრივი ხერხით - საზღვარზე ორთავა ისრის გამოყენებით.

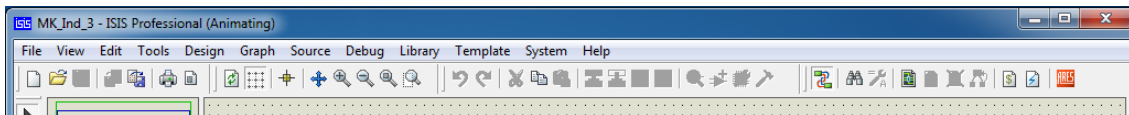
შესაძლებელია ინსტრუმენტების პანელის ან მისი ნაწილების სხვა კიდეებზე გადატანა, როგორც ნაჩვენებია სურ. 1.4 ქვედა მარცხენა კუთხეში, მაგრამ რეკომენდებულია სტანდარტულის გამოყენება.

1.2. სამუშაო ინსტრუმენტების ორგანიზაცია

1.2.1. მენიუს პანელი

ISIS-ს გაშვების შემდეგ გაიხსნება ეკრანი, როლის პირველ სტრიქონზე განთავსებულია სისტემის დასახელება და გახსნილი ფაილის სახელი.

მის ქვევით განლაგებულია მენიუს სტრიქონი, რომელიც, ანალოგიურად ცნობილი გარაფიკული ინტერფეისის სისტემებისა, შეიცავს ბრძანებების ერთობლიობას ყველა იმ ოპერაციების განსახორციელებლად, რომელიც დასაშვებია მოცემულ სისტემაში. ამ ბრძანებების ძირითადი ნაწილი დუბლირებულია მენიუს ქვევით განლაგებულ ინსტრუმენტების პანელზე პიქტოგრამები სახით.



სურ. 1.5.

შენიშვნა: ჩვენ განვიხილავთ მხოლოდ აუცილებელ ოპციებს.

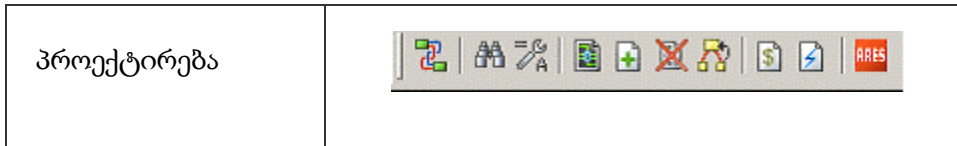
პანელი შეიძლება გადატანილიქნეს სისტემის ფანჯრის ნებისმიერ კიდეზე.

უფრო დაწვრილებით ოფციების შესახებ იხილეთ წყაროებში.

1.2.2. ბრძანებათა პანელები

ინსტრუმენტების პანელი პირობითად შედგება ოთხი დამოუკიდებელი, განუყოფელი ქვესტრუქტურისგან.

პანელის დანიშნულება	ინსტრუმენტები
ფაილი/ბეჭდვა	
გამოსახვა	
რედაქტირება	



შესაძლებელია ცალკეული ან ყველა პანელის დამალვა მენიუდან **View =>Toolbars** (ინსტრუმენტების პანელი).


ეკრანის შევიწროებისას ბოლო პანელები ავტომატურად ინაცვლებენ მომდევნო სტრიქონზე.


ყოველ პიქტოგრამაზე კურსორის მიტანისას გამოჩნდება ფანჯარა, რომელშიც მოცემულია პიქტოგრამის დანიშნულება.


შენიშვნა: რამდენადაც ბრძანებების(ოპერაციების) რეალიზაცია შესაძლებელია როგორც მენიუდან, ასევე ინსტრუმენტების პანელიდან და კლავიატურიდან ე. წ. "ცხელი კლავიშებით", ბრძანების გარჩევისას, სადაც ეს შესაძლებელია, მივუთითებთ სამივე ვარიანტს. ამასთან, ბრძანება მენიუს სახელთან ერთად მიეთითება მხოლოდ ერთხელ-დასაწყისში.

მენიუ **File** (ფაილი):





File => New Design... (ახალი პროექტი), ბრძანება ქმნის ახალ პროექტს სახელით UNTITLED.DSN. იგივე სრულდება მოკლე გზით ინსტრუმენტების პანელიდან პიქტოგრამა  გამოყენებით. ფაილის სახელის შეცვლა შეიძლება შენახვისას ან სისტემიდან გამოსვლისას.

Open Design... (პროექტის გახსნა)-ხსნის არსებული პროექტების დირექტორიას, საიდანაც ან სახელით და **Open** ღილაკით, ან ფაილის სახელზე მარცხენა კლავიშის ორჯერ დაწკაპუნებით ხსნით პროექტს. ამის შესრულება შეიძლება უფრო სწრაფად ინსტრუმენტების პანელიდან პიქტოგრამა  გამოყენებით.

Save Design (პროექტის შენახვა)-ინახავს პროექტის ფაილს შერჩეულ დირექტორიაში სასურველი სახელით. ამის შესრულება შეიძლება უფრო სწრაფად ინსტრუმენტით .

Save As... (შეინახოთ როგორც)-პროექტის შენახვა სხვა სახელით.

Export Section... (სექციის ექსპორტი)-სექციის ან სქემის მონიშნული ნაწილის განთავსება ფაილში, საიდანაც ის შეიძლება გადაეცეს სხვა პროექტებს. ფაილების პანელზე მარცხნიდან მეოთხე პიქტოგრამა- .

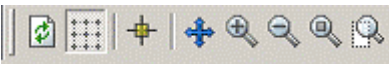
Import Section...(*სექციის იმპორტი*) -სექციის ფაილიდან გაიტანს სექციას მითითებულ პროექტში, სადაც ის შეიძლება განათავსოთ ნებისმიერ ადგილზე. ფაილების პანელზე მარცხნიდან მეხუთე პიქტოგრამაა - .


Import Bitmap ... (*რასტრული გამოსახულების იმპორტი ...*) - საშუალებას იძლევა განვათავსოთ პროექტში სურათი, შესრულებული 256 ფერში და მოცემული BMP (Bitmap(ბიტური რუქა) Picture) ფორმატში. როგორც წესი, სურათი თავსდება რედაქტირების ფანჯარაში შეკუმშული სახით, ხოლო თავისუფალ ადგილზე სურათის მიხედვით იგება სქემა ISIS-ის ელემენტებისაგან.


Export Graphics (*გრაფიკის ექსპორტირება*) - საშუალებას იძლევა გადავიტანოთ ISIS-ის პროექტი სხვა სისტემებში როგორც გრაფიკული გამოსახულება, მაგალითად AutoCAD-ში, DXF ფორმატში.

ფაილის პანელის მეექვსე და მეშვიდე პიქტოგრამა ემსახურება პროექტის ან მისი ნაწილების ამობეჭდვას, რასაც განვიხილავთ გამოყენების დროს.

მენიუ **View** (*ხედი*):

ინსტრუმენტები: 

View => Redraw Display (*ფანჯრის განახლება*)- ამ ფანჯარაში ჩანს სქემა ან სქემის ის ნაწილი, რომლის რედაქტირება ახლა მიმდინარეობს. ასახვის პანელზე პირველი პიქტოგრამაა - , შეესაბამება კლავიშა **R**. ბრძანება ანახლებს სისტემის ეკრანს - ნახაზი გაიწმინდება ზედმეტი ელემენტებისგან, "ჭუჭყისაგან".

Toggle Grid (*ბადის გადამრთველი*)- ბადის სახეს აქვს რამდენიმე ვარიანტი, რომლის შეცვლა ან საერთოდ მოხსნა შესაძლებელია მენიუდან ბრძანებით: **View => Grid**, ან ინსტრუმენტების პანელიდან პიქტოგრამით , ან კლავიშით **G**. წერტილებს შორის მანძილი დამოკიდებულია გაწყობაზე ბრძანებით **მიზმა(Snap)** :

View => Snap, ან ცხელი კლავიშებით: F4, F3, F2 და Ctrl+F1.

ხაზვის კოორდინაციისათვის, როგორც ISIS-ში ისე ARES-ში გამოიყენება საკოორდინატო ბადე, რომლის ბიჯი ფიქსირდება შემდეგ სისტემაში:

1 დიუმი = 10 ხაზს = 25,4 მმ;

1 ხაზი = 10 წერტილს = 2,54 მმ;

1 წერტილი = 1/100 დიუმს = 1/10 ხაზს = 0,254 მმ.


სტანდარტულ ბიჯად გამოიყენება 1 წერტილი ანუ 0,254 მმ.

საკოორდინატო ბადის საწყისი წერტილი ფიქსირებულია სამუშაო არის ცენტრში, რის გამოც კოორდინატები ღებულობენ როგორც დადებით ისე უარყოფით მნიშვნელობებს.


კურსორის კოორდინატები გამოისახება ინდიკატორზე ეკრანის ქვედა მარჯვენა კუთხეში.


თუ საჭიროა წერტილის კოორდინატის ზუსტი განსაზღვრა, უნდა გამოვიყენოთ ბრძანება **X-კურსორი(X-Cursor)**:


View => X-Cursor; რედაქტირების ფანჯარაში კურსორის წვერზე გაჩნდება ჯვარი(x). ამ დროს **x** კლავიშზე დარტყმა გამოაჩენს კოორდინატთა ორთოგონალურ ღერძებს ცენტრით **x** წერტილში, რომელიც თან მიჰყვება კურსორის გადაადგილებას. ცენტრის კოორდინატები მკაფიოდ აისახება კოორდინატთა ინდიკატორზე.


Toggle False Origin (რთავს/გამორთავს კოორდინატების დასაწყისის წარმოსახვით წერტილს) - ასახვის პანელზე მესამე პიკტოგრამაა - , კლავიშა **O**. ბრძანება საჭიროა გრაფიკული მოდელების შექმნის დროს.


საკოორდინატო ბადის საწყისი წერტილი შეიძლება გადატანილიქნეს სხვა ადგილას, თუ ეს ქმნის მოხერხებულობას პროექტირებისთვის-განსაკუთრებით ტრასირების დროს. ამ დროს კოორდინატების ინდიკატორი იცვლის ფერს შავიდან ვარდისფერზე. **Origin**-ის ხელახალი გამოყენება ხსნის რეჟიმს.

Center At Cursor (ცენტრირება კურსორით) - ასახვის პანელზე მეოთხე პიკტოგრამაა- , კლავიშა **F5**. აფიქსირებს გამოსახულების ცენტრს კურსორით მითითებულ ადგილზე მარცხენა კლავიშის დაწკაპუნებით;

Zoom In (მაშტაბის გადიდება) - ასახვის პანელზე მეხუთე პიკტოგრამა - , კლავიშა **F6**. ადიდებას გამოსახულებას უძრავი ცენტრის მიმართ.

Zoom Out (მაშტაბის შემცირება) - ასახვის პანელზე მეექვსე პიკტოგრამაა- , კლავიშა **F7**. ამცირებს გამოსახულებას უძრავი ცენტრის მიმართ.

Zoom To View Entire Sheet(Zoom All) (გადიდება მთელი ფურცლის დასაწყისად) - ასახვის პანელზე მეშვიდე პიკტოგრამაა- , კლავიშა **F8**. სქემის მაშტაბები შეიცვლება ისე, რომ მთელი სქემა მოთავსდეს ეკრანზე. ეს დასახელება გამოჩნდება პიკტოგრამაზე კურსორის მიყვანისას ჩამოშლილ ფანჯარაში; მენიუში ის არის სახელით: **View => Zoom All...** - გადიდდეს ყველა.


Zoom To Area (გადიდდეს ზონა) - ასახვის პანელზე მერვე პიკტოგრამაა - , კლავიშა არა აქვს. ბრძანება ქმნის კურსორს, რომლითაც უნდა შექმნათ მართკუთხედი ამორჩეულ ზონაზე წესით: დაწკაპუნება მარცხენა კლავიშით ზედა მარცხენა კუთხეზე, მოძრაობა დიაგონალზე და დაწკაპუნება მარცხენა კლავიშით ქვედა მარჯვენა კუთხეზე. ზომების შემდგომი ცვლილებები შესაძლებელია მაშტაბირების ოპერაციებით.


ფუნქცია **Toolbars...** უკვე განმარტებული იყო.


მენიუ **Edit** (რედაქტირება):


ინსტრუმენტები:





Edit => Undo (დაბრუნება წინა მოქმედებაზე) - რედაქტირების პანელზე პირველი პიქტოგრამაა-, კლავიშები **Ctrl+z**. სტანდარტულად შეიძლება 20 ოპერაციით უკან დაბრუნება. ხელოვნურად დაყენება შეიძლება ბრძანებით **System => Set Environment...**(გარემოს დაყენება).


Redo (დაბრუნება მომდევნო მოქმედებაზე) - აქტიურია **Undo** შემდეგ. რედაქტირების პანელზე მეორე პიქტოგრამაა-, კლავიშები **Ctrl+y**.


Cut (ამოჭრა) - რედაქტირების პანელზე მესამე პიქტოგრამაა-, კლავიშები არ გააჩნია. აქტიურია ობიექტის მონიშვნის შემდეგ. ამოღებული ობიექტი გადააქვს ბუფერში.

Copy (ასლირება, კოპირება) - რედაქტირების პანელზე მეოთხე პიქტოგრამაა-, კლავიშები არ გააჩნია. აქტიურია ობიექტის მონიშვნის შემდეგ. ობიექტის ასლი გადააქვს ბუფერში.


Paste (ჩასმა) - რედაქტირების პანელზე მეხუთე პიქტოგრამაა-, კლავიშები არ გააჩნია. სამუშაო ველის მითითებულ ადგილზე ჩასვამს ობიექტს ბუფერიდან.


Block Copy (ბლოკის კოპირება) - რედაქტირების პანელზე მეექვსე პიქტოგრამაა-, კლავიშები არ გააჩნია. მონიშნული ბლოკის ასლის გადატანა და შემდგომი ფიქსაცია მარცხენა კლავიშის დაწკაპუნებით.


Block Move (ბლოკის გადატანა) - რედაქტირების პანელზე მეშვიდე პიქტოგრამაა-, კლავიშები არ გააჩნია. მონიშნული ბლოკის გადატანა და შემდგომი ფიქსაცია მარცხენა კლავიშის დაწკაპუნებით.


Block Rotate (ბლოკის მობრუნება) - რედაქტირების პანელზე მერვე პიქტოგრამაა-, კლავიშები არ გააჩნია. საშუალებას იძლევა 2D ელემენტებისგან შემდგარი მონიშნული ბლოკის მობრუნების ან სარკული ანარეკლის შექმნისა. პარამეტრები მიეთითებება ამ დროს გამოჩენილი დიალოგური ფანჯრიდან. კავშირები არ წყდება.


უნდა აღინიშნოს, რომ ამ ოპერაციების ჩატარება უფრო ჩქარია მარჯვენა კლავიშით გამოძახებული მენიუდან. ამასთან, მენიუ ელემენტისთვის და ბლოკისთვის განსხვავებულია.

Block Delete (ბლოკის გაგდება(გაუქმება)) - რედაქტირების პანელზე მეცხრე პიქტოგრამაა-, კლავიშები არ გააჩნია. შლის მონიშნულ ბლოკს/ელემენტს.

Pick Parts From Library (ნაწილების ამორჩევა ბიბლიოთეკიდან) - რედაქტირების პანელზე მეათე პიქტოგრამაა-, კლავიშები არ გააჩნია. განსხვავებით სელექტორის **P** ლილაკისა, ეს ინსტრუმენტი იძახებს მხოლოდ ელექტრო-რადიოკომპონენტების ბიბლიოთეკას. **P** ლილაკის მოქმედება იცვლება სელექტორის რეჟიმის მიხედვით. განიხილება ქვევით.

Make Device (მოწყობილობის შექმნა) - რედაქტირების პანელზე მეთერთმეტე პიქტოგრამა-, კლავიშები არ გააჩნია. გამოიყენება ახალი მოწყობილობის(კომპონენტის) შესაქმნელად მომხმარებლის მიერ. აქ არ განვიხილება.

Packaging Tool (ინსტრუმენტები კორპუსირებისთვის) - რედაქტირების პანელზე მეთორმეტე პიქტოგრამა-, კლავიშები არ გააჩნია. გამოიყენება ახალი კომპონენტის კორპუსების შესაქმნელად მომხმარებლის მიერ. აქ არ განვიხილება.

Decompose (დაშლა) - რედაქტირების პანელზე მეცამეტე პიქტოგრამა-, კლავიშები არ გააჩნია. გამოიყენება მოწყობილობის კომპონენტებად დასაშლელად. აქ არ განვიხილება.


ინსტრუმენტები არ გააჩნია **Edit** მენიუს შემდეგ ოპციებს:

Find and Edit Component... (კომპონენტის პოვნა და რედაქტირება...) - გამოიძახებს დიალოგურ ფანჯარას(კლავიშა **E**) სადაც მიეთითება კომპონენტის იდენტიფიკატორი და შეიტანება **OK** კლავიშით. ის გამოიძახებს რედაქტირების ფანჯარას და მოქმედებები გაგრძელდება ჩვეულებრივი წესით.

Tidy (გასუფთავება) - ოფცია ობიექტების სელექტორის ფანჯრიდან აძევებს ყველა იმ ელემენტს, რომელიც არ არის გამოყენებაში რედაქტირების ფანჯარაში. იგივე შეიძლება გაკეთდეს მარჯვენა კლავიშით და ოფცია **Delete**.


Edit - ის დანარჩენ ოფციებს აქ არ შევხებით, რადგანაც სწავლის დასაწყისში არააქტუალურია.

მენიუ **Tools** (ინსტრუმენტების, პროექტირების პანელი):

ინსტრუმენტები: 


ეს ინსტრუმენტები გადანაწილებულია **Tools** და **Design** მენიუებს შორის

Tools => Real Time Annotation (ანოტირება რეალურ დროში) - ჩართავს/გამორთავს ელემენტების ავტონუმერაციას მისი შეტანისას რედაქტირების ფანჯარაში. ის ავტომატურად ჩართულია, რაზეც მიუთითებს ღილაკი **UI** ჩაწეული მდგომარეობა. მორიგი დაწკაპუნება მას გადაიყვანს საპირისპირო მდგომარეობაში. იგივე სრულდება კლავიშებით **Ctrl+N**. პიქტოგრამა არ გააჩნია.


Wire Auto Router (გამტარების ავტომატური გაყვანა). პროექტირების პანელზე შეესაბამება პირველი პიქტოგრამა-, კლავიატურაზე კლავიშა **W**. ოფცია ავტომატურად ჩართულია და უზრუნველყოფს ტრასის გატარებას ბადის ორდინატების გაყოლებაზე მხოლოდ მართი კუთხით მოხვევით. გამორთულ ოფციაზე გამტარი გაივლება

ნებისმიერი მიმართულებით. ყველა რეჟიმში, მოხვევის დასაფიქსირებლად დააწკაპუნეთ მარცხენა კლავიშით.

Search and Tag Components (New) (კომპონენტის ძიება და გამოყოფა).

პიქტოგრამების პანელზე მეორე ელემენტი-, კლავიატურაზე კლავიშა **T**. გამოაქვს დიალოგური ფანჯარა სადაც მიეთითება საძიებო ობიექტის თვისება ანუ თვისობრივი კუთვნილება რაიმე ჯგუფთან, მათი ურთიერთდამოკიდებულებები და ძიების რეჟიმები. ფანჯარაში მოიცემა ძიების შედეგი: ობიექტის სახელი, სიდიდე და გვერდის ნომერი. სახელზე მარცხენა კლავიშით ორმაგი დაწკაპუნება იწვევს ობიექტის მონიშვნას.


Property Assignment Tools(PAT) (თვისების მინიჭების ინსტრუმენტი). პიქტოგრამების

პანელზე შეესაბამება მესამე ელემენტი-, კლავიატურაზე კლავიშა **A**. შესანიშნავი საშუალებაა ერთგვაროვანი ობიექტებისთვის იდენტიფიკატორების, მნიშვნელობების და მახასიათებლების ავტომატიზირებული მინიჭებისათვის. დაწვრილებით-გამოყენებისას.


Global Anotator (გლობალური ანოტატორი(კომენტატორი)). პიქტოგრამა და

კლავიშა არ გააჩნია. იძახებს დიალოგურ ფანჯარას, სადაც მიეთითება მოქმედების არეალი - გვერდი თუ მთელი დოკუმენტი, და მუშაობის რეჟიმი.


Bill of Materials(მასალების უწყისი). შეიცავს მონაცემებს სქემაზე და გამოყენებული

მასალების, კომპონენტების, ბლოკების და სხვა შემადგენელი ნაწილების ჩამონათვალს ტიპების და პარამეტრების ჩვენებით. პიქტოგრამაა .

Electrical Rule Check(ელექტრული წესების გასინჯვა). ასახავს გასინჯვის უწყისს.

პიქტოგრამაა .

Netlist to ARES(წრედების(კავშირების) სია ARES-თვის). ელექტრული კავშირების

სია საჭიროა ტრასირების სქემის შესაქმნელად საბეჭდი ფირისთვის(პლატისთვის), რომელიც სრულდება ARES-ის საშუალებით. პიქტოგრამაა .

Tools - ის დანარჩენ ოფციებს განვიხილავთ გამოყენებისას.

მენიუ **Design** (დიზაინი, პროექტი)

ინსტრუმენტები:




Design => Edit Design Properties ... (პროექტის თვისებების რედაქტირება). აქ საყურადღებოა მოსანიშნი პუნქტები **Global Power Nets?** და **Cache Model Files?**. პირველით მიეთითება საერთოა თუ არა კვება მთელი პროექტისათვის, ხოლო მეორეთი- შეინახება თუ არა მოდელის ფაილები პროექტის შიგნით, რაც საჭიროა პროექტის გადატანისათვის. დასაწყისში ვტოვებთ ჩართულს.


Edit Sheet Properties ... (გვერდის თვისებების რედაქტირება). რედაქტირდება სახელი და სხვა მონაცემები, რომლებიც დასაწყისში არააქტუალურია.

Edit Design Notes ... (პროექტზე ჩანაწერების რედაქტირება). შეიტანება ინფორმაცია, რომელიც საჭიროა მომხმარებლისთვის მაგრამ, არ აისახება ფორმალურ სტრუქტურებში. პიქტოგრამა და კლავიშა არ გააჩნიათ. უფრო დაწვრილებით განვიხილება გამოყენებისას.

Configure Power Rails (კვების წყაროს სალტების კონფიგურირება) - იძახებს დიალოგურ ფანჯარას, სადაც განისაზღვრება არჩეული სალტის თვისებები და პარამეტრები, შესაძლებელია ძველი სალტის გაგდება, ახლის დამატება და სხვა. პიქტოგრამა და კლავიშა არ გააჩნია. უფრო დაწვრილებით - გამოყენებისას.

Design Explorer(დიზაინის გამყოლი). პიქტოგრამაა . ხსნის ფანჯარას **Physical Partlist View**, სადაც მოცემულია რადიოელემენტების ჩამონათვალი ფურცლების მიხედვით. ფანჯარა შედგება ორი ნაწილისგან. პირველში მოცემულია მენიუ და მის ქვევით ფურცლის სახელი, რომელიც საწყისად არის **Root10**(პროტეუსი ფურცლებს ნომრავს 10-ის ჯერადობით. შეცვლა-რედაქტირებით). მარჯვენა ნაწილში მოცემულია ელემენტების ჩამონათვალი და მათი მახასიათებლები. სახელზე მარცხენა წკაპს გადმოაქვს ჩამონათვალი მარცხენა ფანჯარაში, ხოლო აქ ამორჩეული კონკრეტული ელემენტის მახასიათებლები მარჯვენაში. რეჟიმი საშუალებას იძლევა გაისინჯოს ყველა ელემენტი და მასთან დაკავშირებული წრედი.

New Sheet(ახალი ფურცელი). ხსნის პროექტის ახალ ფურცელს. პიქტოგრამაა .

Remove Sheet(ფურცლის გაგდება). პიქტოგრამაა .

Exit to Parent Sheet(გადასვლა ამოსავალ გვერდზე). პიქტოგრამაა .

მენიუში მოცემულია აგრეთვე ოპციები წინა, მომდევნო და მითითებულ გვერდებზე გადასვლისათვის. პიქტოგრამა არ გააჩნიათ.

მენიუებს **Graph, Source, Debug** და **Library** დაწვრილებით განვიხილავთ გამოყენების დროს. აქ შევჩერდებით მენიუების **Template** და **System** აუცილებლად საჭირო პუნქტებზე.

Template მენიუში ამჯერად საყურადღებოა ოპცია **Set Design Default**, რომელიც ხსნის რედაქტირების ფანჯარას-**Edit Design Defaults**, სადაც შეირჩევა ფერები და ფონტები პროექტისათვის და ჩვენების ალამები დასამალი ობიექტებისათვის: **Show Hidden Text** და **Show Hidden Pins**. პირველი ნებას რთავს ელემენტის ქვევით ნაჩვენები იყოს ტექსტი, რომელიც აკრებილი იყო რედაქტირების ფანჯრის **Other Properties** ველში. თუ ეს ველი ცარიელია აქ გამოდის ბაცი ფერის(შეიძლება შეიცვალოს) ტექსტი <TEXT>. აკრძალვისთვის ეს ალამი უნდა მოიხსნას. **Show Hidden Pins** ალამი ნებას რთავს ნაჩვენები იქნეს ელემენტის კვების წყაროების კონტაქტები, მაგრამ მათი ცხადი გამოყენება სქემაში არ შეიძლება. ის განიხილება ცალკე, კვების წყაროების კონტაქტების ვიზუალიზაციის დროს.

System მენიუში თავიდან საყურადღებოა ოპცია **Set Paths**(დააყენეთ გზა), სადაც ველში **Initial Folder For Design** გადამრთველით მიეთითება დირექტორია სადაც განთავსებულია პროექტის ფაილები. თუ თქვენ პროექტებს ინახავთ ერთ დირექტორიაში, რეკომენდებულია მესამე ალამი და ამ დროს გააქტიურებულ ფანჯარაში უნდა მიუთითოთ გზა.

საჭიროა ოპცია **Set Sheet Size...** (ფურცლის ზომა) გამოყენების ცოდნა. საწყისში ის დგება A4 ფორმატზე, მაგრამ შეიძლება მისი შეცვლა.

ასევე საინტერესოა ოპცია **Set Animation Option...** (ანიმაციის ოპციის დაყენება), რომელიც იძახებს რედაქტირების ფანჯარას სქემების ანიმაციური კოფიგურირებისათვის- **Animated Circuits Configuration**. აქ შეიძლება შეირჩეს ქმედებები კომპონენტების

კომპონენტის ორიენტირების პიქტოგრამები განხილულია ქვევით, ელემენტთა სქემაზე განთავსების აღწერისას.

გავაკეთოთ მოკლე განმარტებები დანარჩენ პიქტოგრამებზე. მოკლე განმარტებას-დასახელებას მიიღებთ მაშინაც, როდესაც კურსორს მიიტანთ პიქტოგრამასთან.

1.2.5. ძირითადი რეჟიმების პიქტოგრამები



- **Selection Mode** (*ამორჩევის რეჟიმი*). რედაქტირების ფანჯარაში ნებისმიერ ობიექტთან ისარის მიტანა და მარცხენა კლავიშის დაწკაპუნება იწვევს მის ამორჩევას-გაწითლებას. თუ დაჭერილ კლავიშას გადავატარებთ დიაგონალზე-მოწინაშე კომპონენტების ჯგუფს, ისინი მოყვებიან მართკუთხა კონტეინერში.



- **Component Mode** (*კომპონენტების რეჟიმი*). ამ რეჟიმში კომპონენტები სელექტორიდან გადაიტანება რედაქტირების ფანჯარაში. ამისთვის, მარცხენა კლავიშით სელექტორში ვირჩევთ კომპონენტს და წკაპით საფიქსირებთ რედაქტირების ფანჯარაში. უფრო დაწვრილებით - ქვევით.



- **Junction Dot Mode** (*შერთების წერტილების რეჟიმი*). გამტართა გადაკვეთის ადგილზე წერტილის დასმა მარცხენა კლავიშით ორჯერ დაწკაპებით.



- **Wire Label mode** (*გამტარების და სალტეების მონიშვნა*). კურსორის ობიექტთან მიტანისას ფანქრის წვერთან გამოჩნდება ირიბი ჯვარი, ამ დროს მარცხენა კლავიშით დაწკაპება გამოიძახებს რედაქტირების ფანჯარას (**Edit Wire Label**), სადაც მითითებულ ველებში უნდა ჩაიწეროს სახელი და სხვა პარამეტრები.



- **Text Script Mode** (*ტექსტის ჩაწერის რეჟიმი*) სქემის ნებისმიერ ადგილას. მარცხენა წკაპუნის სქემის თავისუფალ ადგილზე იძახებს დიალოგურ ფანჯარას, სადაც აიკრიბება ტექსტი, რომელიც OK-თი განთავსდება კურსორის ადგილზე. სქემაზე რედაქტირების რეჟიმში შესაძლებელია ტექსტის ექსპორტ/იმპორტი სხვა ფაილიდან/ში, აგრეთვე ტექსტისორიენტაციის შეცვლა.



- **Buses Mode** (*სალტეების ხაზვის რეჟიმი*). საწყის წერტილში მარცხენა წკაპი და ფანქრის გადაადგილება საჭირო მიმართულებით, მართკუთხა გაღუნვის წერტილში ისევ წკაპი, თუ **ctrl**-ს დავაჭერთ გაღუნვა ნებისმიერი მიმართულებით, ორჯერ მარცხენა წკაპით დამთავრება.



- **Subcircuit Mode** (*ქვესქემების რეჟიმი*). საჭირო ელემენტები მონიშნება ჩვეულებრივ და კონტეინერს დაეწერება სახელი **SUB?**, ხოლო შემდეგ რედაქტირების რეჟიმში დაზუსტდება სახელი, გამომყვანები და სხვა.

1.2.6. ხელსაწყოების პიქტოგრამები:



- **Terminals Mode** (ტერმინალების განთავსების რეჟიმი). ტერმინალები საშუალებას იძლევიან დავაკავშიროთ სქემის ორი ან რამდენიმე წერტილი, როგორც ერთ ისე რამდენიმე ფურცელზე ისე, რომ არ ჩანდეს მათ შორის დამაკავშირებელი ხაზი. ამისთვის, თვისებების ფანჯარაში(**Properties**) მათ არქმევენ ერთდამიმავე სახელს. ფანჯარა გამოიძახება დადგენილი წესით-ორი მარცხენა წკაპი ან მარჯვენა წკაპი და ოპცია **Edit Properties**. სელექტორში ამორჩეული ტერმინალი შეიძლება დავამუშაოთ წინათვალისწიანების ფანჯარაში-**Preview**. დასახელებები **Default**, **Output**, **Input** და **Bidir** მიუთითებენ მხოლოდ ფუნქციონალურ დანიშნულებაზე-სიგნალს ისინი ატარებენ ნებისმიერი მიმართულებით. **Power** და **Ground** ტერმინალები თუ არ არიან სახელდებული განსაკუთრებულად, ითვლებიან ჩართულად გლობალურ კვებასთან და მიწასთან. რაც შეეხება ტერმინალ **BUS**-ის და სალტების მარკირებას(**LBL** რეჟიმი), მათი სახელი ფორმირდება სახით: **სახელი[საწყისი_თანრიგი..ბოლო_თანრიგი]**,

სადაც, სახელი-სალტის აღნიშვნაა, ლათინური სიტყვა შუალედების და სპეცსიმბოლოების გარეშე; საწყისი_თანრიგი და ბოლო_თანრიგი ორი რიცხვია უწყვეტი, ზრდადი მიმდევრობიდან, რომელიც განსაზღვრავს ტერმინალის თანრიგის რაოდენობას. მათ შორის წერტილები უნდა იყოს ორი, ხოლო ფრჩხილები აუცილებლად მართკუთხა. მაგალითად, 16-თანრიგა ტერმინალი სახელით A აღნიშნება- **A[0..15]**, რომელიც შედგება A0, A1, A2 A15 თანრიგებისგან. აქიდან შეიძლება გაკეთდეს საჭირო თანრიგების განტოტება ტერმინალის იმავე სახელით-**A[8..11]**, სადაც გაიცემა A8, A9...A11 თანრიგები.



- **Device Pins Mode** (მოწყობილობის გამომყვანების განთავსების რეჟიმი). ახალ მოწყობილობაზე გამომყვანების განლაგების განსაზღვრა. დაწვრილებით-გამოყენებისას.



- **Graph Mode** (გრაფიკის რეჟიმი). გამოიყენება სქემის და მისი კომპონენტების მუშაობის დროითი დიაგრამების ასახვისათვის. დაწვრილებით განვიხილავთ ქვევით.



- **Tape Recorder Mode** (მაგნიტოფონზე ჩაწერის რეჟიმი). სიმულაციის პროცესში გენერირებული სიგნალების ჩაწერა ვირტუალურ მაგნიტოფონზე.



- **Generator Mode** (გენერატორის რეჟიმი). ამ რეჟიმში შესაძლებელია სელექტორიდან შეირჩეს სხვადასხვა ტიპის სიგნალების გენერატორი, როგორც ციფრული ისე ანალოგური სიგნალებისათვის. აღსანიშნავია პროგრამული გენერატორი **SCRIPTABLE**, რომლისთვის პროგრამა იწერება პროტეუსის ენაზე **Easy HDL**. მსგავსია გენერატორები **FILE** და **AUDIO**, რომლებიც იყენებენ ფაილში ჩაწერილ ინფორმაციას. თვალისწიანების ფანჯარაში-**Preview**, შესაძლებელია გენერატორის გრაფიკული სიმბოლოს ორიენტაციის შეცვლა. გენერატორის განთავსებისას სახელის ადგილზე იწერება ნიშანი "?", ხოლო კონტაქტზე მიერთებისას მისი სახელი. შემდგომში შესაძლებელია შეცვლა რედაქტირების რეჟიმში.



- **Voltage Probe Mode** (ძაბვის სასინჯის(ზონდის) რეჟიმი). სასინჯის ორიენტირება, დაყენება და რედაქტირება ხდება ანალოგიურად გენერატორებისა იმ განსხვავებით, რომ სასინჯი უერთდება გამტარს და არა გამომყვანს.

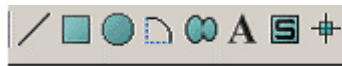


- **Current Probe Mode** (დენის სასინჯის რეჟიმი). გამოიყენება ანალოგიურად ძაბვის სასინჯისა იმ განსხვავებით, რომ ისარი წრეში მიმართული იყოს გამტარის პარალელურად. როდესაც დენის მიმართულება ემთხვევა ისრის მიმართულებას-მნიშვნელობა დადებითია, არდამთხვევისას უარყოფითი.



- **Virtual Instruments Mode** (ვირტუალური ინსტრუმენტების რეჟიმი). პროტეუსში, ისე როგორც ვირტუალური მოდელირების ყველა სისტემაში, არსებობს ვირტუალური ხელსაწყოების ნაკრები: ვოლტმეტრები, ამპერმეტრები, მრავალარხიანი ოსცილოგრაფები, სიგნალების ანალიზატორები, მთვლელები, სიხშირის მზომები, სიგნალ-გენერატორები და სხვა. უნდა გავითვალისწინოთ, რომ მათ უმრავლესობას გააჩნია ერთპოლუსიანი ჩართვა-გაზომვა წარმოებს მხოლოდ მიწის მიმართ. და კიდევ-ეს ხელსაწყოები მოითხოვენ კომპიუტერის გაზრდილ რესურსებს, რომლის გათვალისწინება რთული სქემების დამუშავებისას მნიშვნელოვანია.

1.2.7. 2D გრაფიკის პიქტოგრამები:



ამ პიქტოგრამების ძირითადი ნაწილის დანიშნულება ინტუიტიურად გასაგებია, მითუმეტეს პირთათვის ვისაც უმუშავია გრაფიკულ რედაქტორებთან. მაგრამ, გარკვეულობისათვის მაინც გავაკეთებთ მოკლე განმარტებებს პიქტოგრამების რიგის მიხედვით.

წრფივი ხაზი; მართკუთხედი, რომლის ფერით შევსება შეიძლება; წრე, რომლის ფერით შევსება შეიძლება; რკალი; შეკრული ოვალური ფიგურა(რეალურად მრავალკუთხედი) ფერით შევსებით; სიმბოლო **A** - ტექსტის განთავსება კურსორით მითითებულ ადგილზე; სიმბოლო **S** მართკუთხედში-რეჟიმი, როდესაც სელექტორის **P** იმახებს სპეციალურ სიმბოლოთა ბიბლიოთეკას; და ბოლოს, წრფეთა გადაკვეთა კვადრატში-გრაფიკული მარკერის დასმა. უფრო დაწვრილებით-გამოყენებისას. ყველა ობიექტის რედაქტირება შესაძლებელია ცნობილი წესით გარდა მარკერისა.

ეკრანის ქვედა სტრიქონზე განლაგებულია სიმულაციის მართვის ღილაკები. მარჯვნიდან მარცხნივ:











გაშვება(**Start**), ნაბიჯი(**Step**), პაუზა(**Pause**), გაჩერება(**Stop**). გამოყენების თავისებურებებს გავეცნობით პროგრამების გამართვის აღწერისას.

თავი 2. სქემის პროექტის შექმნა

2.1. სქემის ელემენტების შერჩევა

ანალოგიურად სხვა გარაფიკული დიალოგური სისტემებისა, ISIS-შიც მანიპულაციის ძირითად საშუალებას წარმოადგენს თავისი კურსორი. იმისათვის, რომ რაიმე ობიექტზე ჩატარდეს რაიმე ოპერაცია საჭიროა მისი (გ)ამორჩევა. გამორჩევა ის ობიექტი რომელზეც დგას კურსორი და რომელიც ამ დროს შემოისაზღვრა პუნქტირის ხაზით. იმისდა მიხედვით, თუ რა ოპერაცია ხორციელდება გამორჩეულ ობიექტზე, იცვლება კურსორის ფორმა და ფერი, როგორც მოცემულია ქვედა ცხრილში.

-  სტანდარტული კურსორი-გამოიყენება ობიექტის გამორჩევისთვის
-  ობიექტის განთავსების თეთრი ფერის კურსორი - ობიექტის განთავსება იწყება თავისი მარცხენა კლავიშის დაწკაპუნებით და მთავრდება მეორე დაწკაპუნებით
-  გამტარების განთავსების აქტიური, მწვანე ფერის კურსორი - გამოჩნდება გამტარების განთავსების თავსა და ბოლოში თავისი მარცხენა კლავიშზე დაჭერისას
-  სალტის განთავსების აქტიური, ლურჯი ფერის კურსორი - გამოჩნდება სალტის განთავსების თავსა და ბოლოში თავისი მარცხენა კლავიშზე დაჭერისას
-  ობიექტი კურსორის ქვეშ ამორჩევა თუ დავაწკაპუნებთ მარცხენა კლავიშას
-  კურსორის ქვევით ობიექტი გადაადგილდება კურსორის მიმყოლად დაჭერილი მარცხენა კლავიშის პირობებში
-  გამტარის სეგმენტი გადაადგილდება კურსორის მიმყოლად დაჭერილი მარცხენა კლავიშის პირობებში
-  მარცხენა კლავიშის დაწკაპუნება ობიექტს მინაჭებს მითითებულ თვისებას ინსტრუმენტ Property Assignment Tool გამოყენებისას

სქემის შექმნა იწყება ელემენტის ამორჩევით ბიბლიოთეკიდან. ამორჩევა კი ხორციელდება ბიბლიოთეკის ბრაუზერის (**Browse Library**) დიალოგური ფანჯრის **Pick Devices** საშუალებით, რომლის გამოძახება შეიძლება სამი გზით:

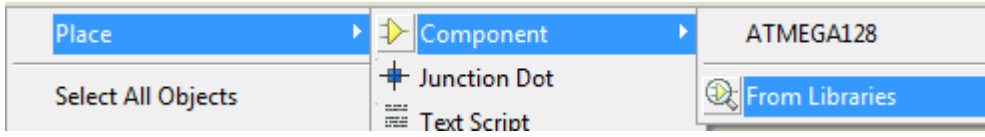
1. სელექტორის ველში ან კლავიატურაზე **P** დილაკზე დაჭერით - სურ. 2.2.



სურ. 2.2.

2. მარცხენა კლავიშის ორჯერ დაწკაპუნებით სელექტორის ველზე-**Object Selector**.
3. სქემის თავისუფალ ადგილზე მარჯვენა კლავიშის დაწკაპუნებით ეკრანზე გამოვა

კონტერქსტური მენიუ, სადაც ვირჩევთ ოპციებს: **Place => Component => From Libraries** - სურ. 6:



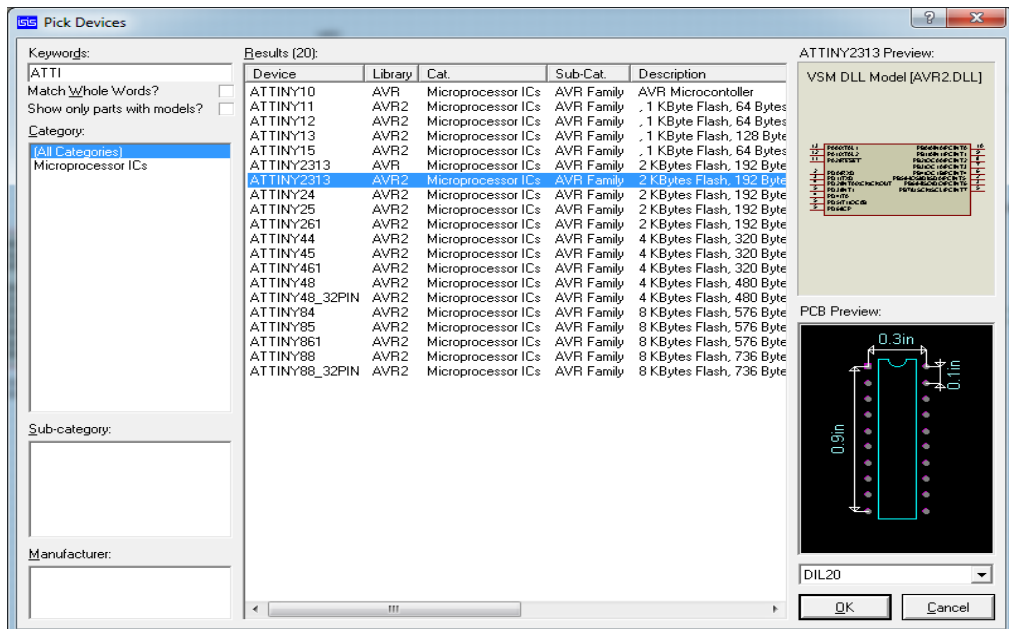
სურ. 2.3.

Pick Devices ფანჯრას გააჩნია რამდენიმე ველი, რომლებიც ერთობლიობაში საშუალებას იძლევიან დადგინდეს საჭირო ელემენტი, მისი სქემური გამოსახულება და კონსტრუქციული პარამეტრები, რომლებიც აუცილებელია ბეჭდური ფირის ნახაზის შესაქმნელად.

დიალოგურ ფანჯრაში საჭირო ელემენტის ძიება შესაძლებელია რამდენიმე გზით. ყველაზე მარტივია ელემენტის სრული იდენტიფიკატორის გამოყენება, ან იდენტიფიკატორის ფრაგმენტის გამოყენება შემდგომი დაზუსტებით ან ამორჩევით სიმრავლიდან-შეტანილი ფრაგმენტი შეიძლება აღმოაჩნდეს მრავალ ელემენტს.

იდენტიფიკატორი ან მისი ფრაგმენტი შეიტანება ველში **Keywords**(საკვანძო სიტყვები) და თუ საჭირო გახდა, დაზუსტება შეიძლება ოპციებში: **Category**(კატეგორიები), **Sub-Category**(ქვეკატეგორიები), **Manufacturer**(მწარმოებელი) და **Library**(ბილიოთეკა).

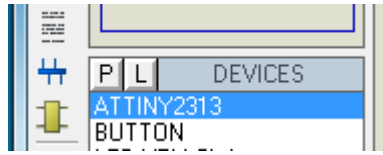
აღსანიშნავია, რომ საკვანძო სიტყვების და კატეგორიების ველების გამოყენების მიმდევრობა შეიძლება იყოს ნებისმიერი - დამოკიდებულია მოხერხებულობაზე, ორივეჯერ ხდება სიის შემცირება-ფილტერაცია. მაგალითები მოვიყვანილია სურ. 2.4, 2.5:



სურ. 2.4.

მას შემდეგ, რაც ელემენტი ნაპოვნია, მის სტრიქონზე მარცხენა კლავიშით ერთჯერ დაწკაპუნებით, ელემენტის ელექტრონული სქემა აღმოჩნდება **Pick Devices** წინასწარი თვალიერების ველში(**Schematic Preview**)-ზედა მარჯვენა კუთხე, ხოლო კორპუსის ნახაზი და ტიპი-მარჯვენა ქვედა კუთხეში(**PCB Preview (PCB- printed circuit board)**) . თუ

ყველაფერი მისაღებია ვაჭერთ ღილაკზე **OK**(კორპუსის ტიპის ქვევით) და სქემა აღმოჩნდება ძირითადი ფანჯრის თვალთვლების ველში, ხოლო დასახელება სელექტორის ველში.




სურ. 2.5.


უფრო მარტივად კი, ელემენტი აღმოჩნდება ძირითადი ფანჯრის თვალთვლების ველში, თუ მის სტრიქონზე დავაწკაპუნებთ ორჯერ მარცხენა კლავიშით.

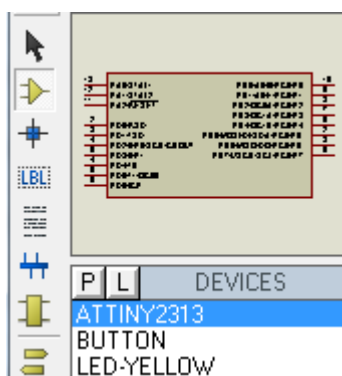
მაგალითში, საკვანძო სიტყვების ველში შეტანილია მკ-ს არასრული დასახელება **ATTI**, კატეგორიების ველში მონიშნული რჩება **All Categories**.

ამ ხერხით ამოიკრიბება სქემისთვის საჭირო ყველა ელემენტი და განთავსდება სელექტორის ველში სიის სახით.

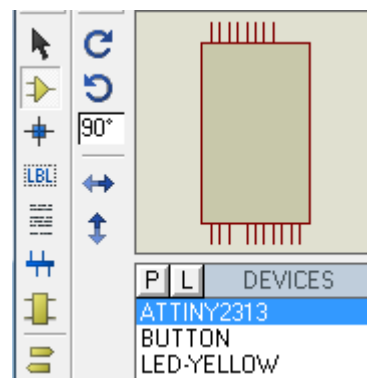
2.2. ელემენტების განთავსება სქემაზე

ინსტრუმენტების მარცხენა პანელზე ვირჩევთ კომპონენტების რეჟიმს (**Component mode**) პიქტოგრამით . შემდეგ ვირჩევთ კომპონენტს(ელემენტს) სელექტორში მარცხენა კლავიშით: ის მონიშნება და ერთდროულად მისი სქემა გამოჩნდება თვალთვლების ფანჯარაში. უნდა შევნიშნოთ, რომ ეს ოპერაცია შეიძლება განხორციელდეს კომპონენტების სელექტორში ელემენტის პირდაპირ ამორჩევით. ჩვენს მაგალითში მკ **Attiny2313**-სურ. 9.

თვალთვლების ფანჯარაში მოთავსებულ ელემენტს შეიძლება ვუცვალოთ ორიენტაცია: თარაზულ სიბრტყეში წრიულად პიქტოგრამებით . მოზრუნების კუთხე შეირჩევა ფანჯარაში ამ პიქტოგრამების ქვევით.




სურ. 2.6.




სურ. 2.7.

ეს ოპერაციები შეიძლება შესრულდეს ოპერატულად რედაქტირების ძირითად ფანჯარაში: სქემაზე ამოირჩევა ელემენტი და კლავიშებით "+" და "-" ვატრიალებთ მას.

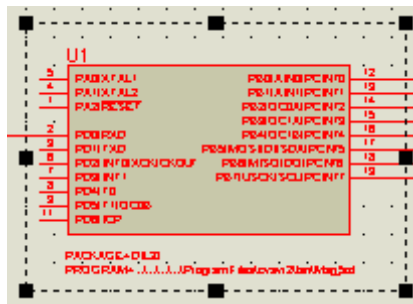
თვალთვლების ფანჯარაში ასევე შესაძლებელია სარკული ანასახის მიღება ჰორიზონტალურ ან ვერტიკალურ სიბრტყეში პიქტოგრამებით .

როდესაც მიღებულია ელემენტის საჭირო გამოსახულება შეიძლება მისი გადატანა რედაქტირების ძირითად ფანჯარაში. ამისთვის კი კურსორს ვათავსებთ არის შერჩეულ ადგილას. გამოჩნდება ელემენტის გამოსახულება მკრთალ ფერებში(შეცვლა შეიძლება), რომელიც კურსორის მოძრაობას თან მიყვება. კურსორის მეორეჯერ დაწკაპუნება აფიქსირებს ელემენტის მდგომარეობას.

ISIS საშუალებას იძლევა მოვახდინოთ განთავსებული სქემის ელემენტების რედაქტირება, მაგრამ ამისთვის ელემენტი უნდა იყოს გამოყოფილი(მონიშნული). გამოყოფა შეიძლება რამდენიმე ხერხით:

- სტანდარტული ხერხი, როდესაც ვრთავთ გამოყოფის პიქტოგრამას, ვირჩევთ ობიექტს და დაწკაპუნებთ თავის მარცხენა კლავიშით. მომდევნო ოპერაციების ჩასატარებლად უნდა გადავიდეთ შესაბამის პიქტოგრამაზე.
- თავის მარჯვენა კლავიშით დაწკაპუნება გამოყოფს ობიექტს და თან გამოიტანს კონტექსტურ მენიუს, რომელშიც მოცემულია ობიექტზე დასაშვებ მოქმედებათა ნუსხა.
- მარცხენა კლავიშით დაწკაპუნება ობიექტზე, როდესაც კურსორს აქვს სახე  .

დაჭერილი მარცხენა კლავიშის გამოყენებით, ცნობილი წესით, ჩავსვათ ობიექტი გამოყოფის ჩარჩოში(კონტეინერში)- სურ. 2.8.

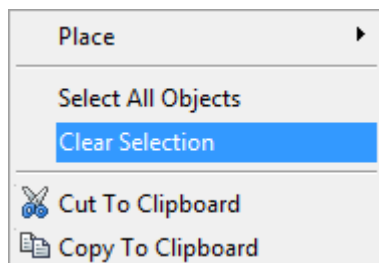


სურ. 2.8.

• რამდენიმე ობიექტის გამოსაყოფად საჭიროა გამოვიყენოთ დაჭერილი კლავიშა **Ctrl** და თავის მარცხენა კლავიშით დაწკაპუნება.

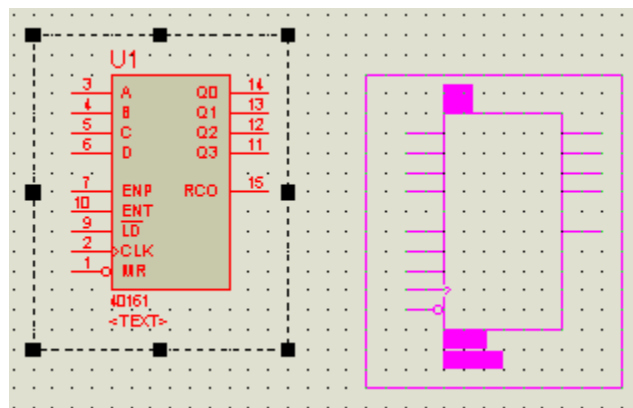
ერთი ან რამდენიმე გამოყოფის მოსახსნელად საჭიროა თავისუფალ ადგილზე თავის მარცხენა კლავიშით დაწკაპუნება, ან მარჯვენა კლავიშით დაწკაპუნება და კონტექსტურ მენიუში **Clear Selection** პუნქტის ამორჩევა - სურ. 12.

შესაძლებელია სქემის ელემენტის გადაადგილება მოქმედებათა შემდეგი მიმდევრობით: ელემენტის მონიშვნა მასზე მარცხენა კლავიშის დაწკაპუნებით, ისრებიანი ჯვრის გამოჩენა მიუთითებს გადაადგილების ნებართვაზე, დაჭერილი მარცხენა



სურ. 2.9.

კლავიშის მდგომარეობაში თავის მოძრაობა, რომელსაც თან მიყვება ელემენტი, საჭირო ადგილზე კლავიშის აშვება, ელემენტი რჩება მონიშნული, მარცხენა კლავიშის თავისუფალ ადგილზე დაწკაპუნება ხსნის მონიშნვას - სურ. 2.10.



სურ. 2.10.

შესაძლებელია ელემენტის მონიშნვა მარჯვენა კლავიშით და ბრძანება **Drag Object**-ის გამოყენება კონტექსტური მენიუდან.

შესაძლებელია ელემენტის ბრუნვა გადაადგილების პროცესში.

აღსანიშნავია, რომ ელემენტებს შორის არსებული კავშირები ნარჩუნდება.

ერთგვაროვანი ელემენტების გამრავლება შესაძლებელია კოპირებით.

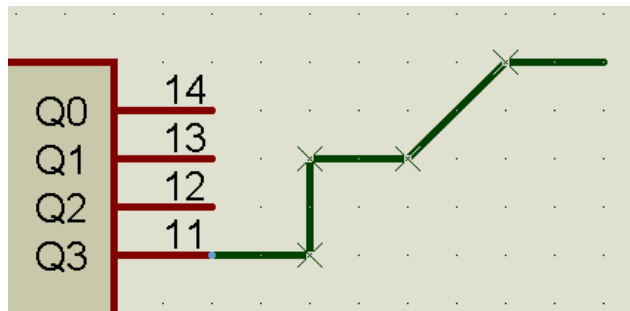
2.3. სქემის ელემენტების შეერთება

ელემენტების შეერთება ერთერთი საყურადღებო და შრომატევადი პროცესია ელექტრონული სქემების დაუშვებში. ISIS არ მოითხოვს რაიმე განსაკუთრებულ რეჟიმს ამ ოპერაციების ჩატარებისათვის, მაგრამ არსებობს მცირე გამონაკლისებიც.

გამტარები შეიძლება განთავსდეს ან რედაქტირებულიყნეს ნებისმიერ დროს გარდა ელემენტების მონიშვნის და ამორჩევის რეჟიმებისა.

რაიმე ელემენტის გამომყვანთან კურსორის მიტანისას ის გამწვანდება, რითაც მიუთითებს, რომ გამტარის გაყვანა შეიძლება. ამ დროს დააწკაპუნეთ მარცხენა კლავიშით და დაიწყეთ თავის გადაადგილება საჭირო მიმართულებით-კურსორი გათეთრდება, რაც მიანიშნებს, რომ მიმდინარეობს გამტარის გაყვანა, რომელიც ვიზუალურადაც ჩანს. ამ პროცესს **მიმყოლი** გაყვანა(განთავსება) ეწოდება - სურ. 14. ამ დროს გამტარის გზა მიყვება საკოორდინაციო ბადის ხაზს, მართკუთხა გადასვლებით მათ შორის. ამ რეჟიმის შეცვლა შეიძლება თუ გაყვანისას დავაჭერთ **Ctrl**-ს - გამტარი გაივლის უმოკლესი გზით.

გამტარების გაყვანის პროცესის უკეთ მართვისათვის შემოტანილია ცნება **მიერთების ღუზა** ანუ შეერთებათა ფორმირება. გაყვანილობის ფორმირების პროცესში მარცხენა კლავიშით ყოველი დაწკაპუნება გააფორმებს ამ ადგილზე მიერთების წერტილს-პატარა ჯვარი გამტარზე-სურ. 2.11. ასე მიღებული მონაკვეთები ითვლება გამტარის დამოუკიდებელ სექციად. ამის შემდეგ მარჯვენა კლავიშით დაწკაპუნება გააუქმებს ბოლო შეერთებას, ყოველი შემდგომი დაწკაპუნება გააუქმებს მომდევნო შეერთებას და ა. შ.



სურ. 11.

თუ საჭიროა გაყვანილობების რედაქტირება - უნდა მოინიშნოს შერჩეული გამტარი მარჯვენა კლავიშით, კონტექსტურ მენიუში აირჩიოთ **Drag Wire** და პარალელურად გადაადგილოთ ის მონაკვეთი რომლიდანაც დაიწყო მონიშნვა. მარცხენა კლავიშით დაწკაპუნების შემდეგ გამოჩნდება ორისრიანი კურსორი და დაჭერილი მარცხენა კლავიშით შეგიძლიათ გააგრძელოთ პარალელური გადატანა და დააფიქსიროთ მარცხენა კლავიშით თავისუფალ ადგილზე დაწკაპუნებით.


უფრო მარტივად კი - მარცხენა კლავიშით დაწკაპუნებით გამტარის ამორჩევა, გამოჩნდება ორისრიანი კურსორი და იმოქმედებთ აღწერილი წესით.

უნდა გვახსოვდეს, რომ ობიექტზე ზედიზედ მარჯვენა კლავიშით დაწკაპუნება აუქმებს ამ ობიექტს.

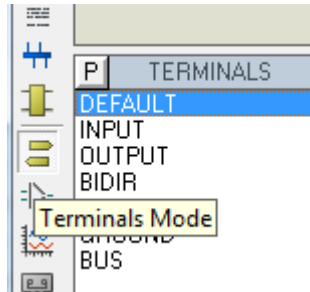
ასევე, ობიექტზე ზედიზედ მარცხენა კლავიშით დაწკაპუნებას გამოაქვს ამ ობიექტის რედაქტირების ფანჯარა.

თუ საჭიროა გაყვანილობის ხელით შესრულება - დააწკაპუნეთ მარცხენა კლავიშით საწყის წერტილზე და აწკაპუნეთ მარცხენა კლავიშით მოხვევის ყოველ წერტილზე, ბოლოს კი დააწკაპუნეთ მიერთების წერტილზე.

2.4. ტერმინალური წერტილები

ტერმინალური წერტილებიდან აუცილებელია საერთო წერტილი (მიწა, კორპუსი) და კვების წყაროები. მათი სქემაში შემოტანისთვის ინსტრუმენტების მარცხენა პანელიდან ვირჩევთ პიქტოგრამას  - **Terminals Mode**, რაზეც სისტემა რეაგირებს სელექტორის ველში ხელმისაწვდომი კლიპების დასახელების გამოტანით- სურ. 2.12


რომ არ გადაიტვიტოს ნახაზი, როგორც წესი კვების მისაერთებელი კონტაქტები არ ჩანს(თუმცა მათი გამოჩენა შეიძლება) და შესაბამისად მათი გაყვანილობებიც არ ჩანს. ეს იმიტომ, რომ რედაქტირების ველში შემოტანილი ტერმინალი ავტომატურად უერთდება შესაბამისი ელემენტის შესაბამის კონტაქტებს.



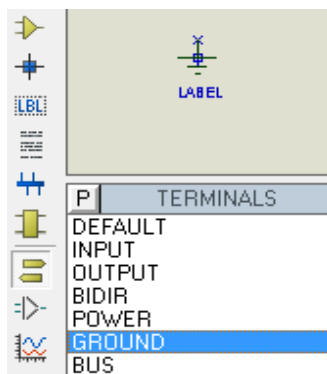
სურ. 2.12.

სელექტორის ველში ამორჩეული ტერმინალის პირობითი გამოსახულება მყისეულად ჯდება თვალსაჩინოების ველში, სადაც შეიძლება მისი ორიენტაციის შერჩევა, ხოლო შემდეგ მისი გადატანა ჩვეულებრივად რედაქტირების ველის სასურველ უბანში. მიწის(**GROUND**) შემთხვევაში ეს მარტივია- სურ. 2.13., კვების(**POWER**) შემთხვევაში კი ის მოითხოვს რედაქტირებას ნომინალის დასადგენად მაინც- სურ. 2.14.

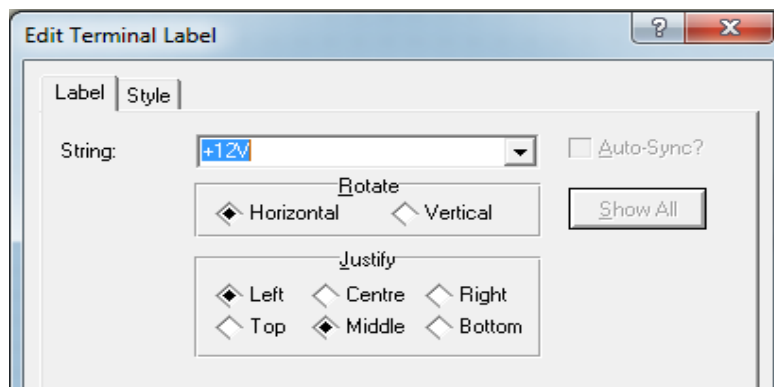
როგორც წესი, კვების წყარო გამოსახება მოკლე ხაზი ისრით. როგორც სხვა ობიექტებზე, მონიშნვა და რედაქტირების ფანჯრის გამოძახება შეიძლება ხერხებით:

- მარცხენა კლავიშით ორჯერ დაწკაპუნებით;
- მარჯვენა კლავიშით დაწკაპუნებით და კონტექსტური მენიუდან **Edit Properties**-ის ამორჩევა;
- ამორჩევის რეჟიმში შესვლა - , მარცხენა კლავიშით დაწკაპუნებით მონიშნვა, შემდეგ მარჯვენა კლავიშით **Edit Properties**-ის გააქტიურება და გამოსვლა რეჟიმიდან; ცხადია პრაქტიკულად ყველაზე მოხერხებულია პირველი ვარიანტი.

რედაქტირების ფანჯრის **String** ველში უნდა ჩაიწეროს ძაბვის ნომინალი. ეს პროცესი მეორდება ძაბვის ყველა ნომინალისთვის.



სურ. 2.13.



სურ. 2.14.

გარდა კვების წყაროების სქემაში შეიძლება არსებობდეს სასიგნალო წრედები, რომელთაც ჭირდებათ სტანდარტული დაბოლოებები. ამ შემთხვევაში სელექტორის ველში ვირჩევთ სტანდარტულ კლემას(DEFAULT), რომელიც თვალყურის ფანჯარაში დამუშავების შემდეგ გადაიტანება რედაქტირების ველის სასურველ უბანზე - სურ. 2.15.



სურ. 2.15.

კლემებს გააჩნიათ გარკვეული ზოგადი თვისებები:

თვისება	აღწერა
NET	კლემის ქსელის ნიღაბი
SYMBOL	კლემის აღნიშვნა. ის შეიძლება იყოს როგორც სტანდარტული, ისე შემოტანილი მომხმარებელი მიერ
TYPE	კლემის ტიპი. ის შეიძლება იყოს პასიური(PASSIVE), შესასვლელი(INPUT), გამოსასვლელი(OUTPUT), ორმომართულებითი(BIDIR) და კვების(POWER).

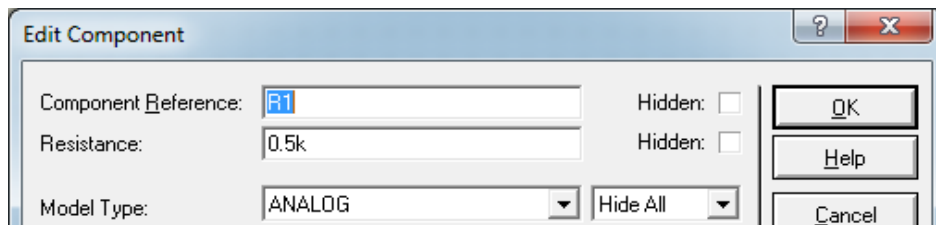
ყველა ობიექტს გააჩნია კონტექსტური მენიუ, სადაც მოცემულია მოქმედებები, რომელიც შეიძლება განხორციელდეს ამ ობიექტზე. ამიტომ, გავრცელებულია გამოთქმა: "თუ გეეჭვება-დააჭირე თავის მარჯვენა კლავიშას".

2.5. ელემენტების აღნიშვნა სქემაზე

ელექტრონულ-პრინციპიალურ სქემაზე ყოველ ელემენტს უნდა ახლდეს უნიკალური დასახელება (იდენტიფიკატორი) და თუ საჭიროა პარამეტრების

მაჩვენებლები. ერთი ტიპის ელემენტების სახელები შეიძლება ჩაისვას ავტომატურად რიგითი ნომრის ჩვენებით, თუ ჩართულია ოპცია **ანოტირება რეალურ დროში**(Tools => **Real Time Annotation**). ის კი ყოველთვის ჩართულია თუ არ გამორთავთ.

ელემენტის დასახელების რედაქტირება ისევე შეიძლება როგორც სხვა ობიექტებისა-შეცვლა, სიდიდის კორექცია, გადაადგილება. ამისთვის საჭიროა სტანდარტული გზით **კომპონენტის რედაქტირების** ფანჯრის გამოძახება (**ობიექტის ამორჩევა => მარჯვენა კლავიშა => Edit Properties => Edit Component**) - სურ. 2.16.

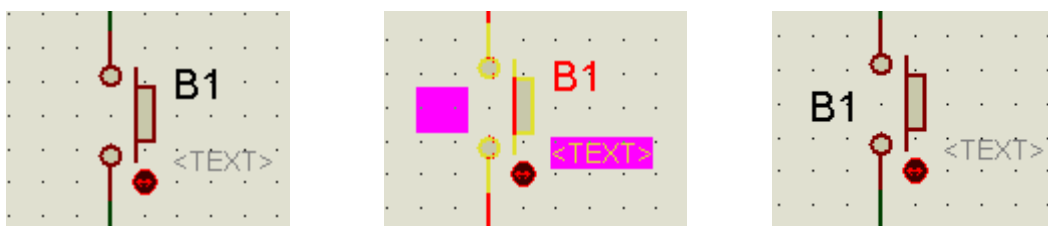


სურ. 2.16.

ნახაზზე შეტანილია წინააღმდეგობის სახელი R1 და სიდიდე 1k. გვერდზე ფანჯრები მიუთითებს, რომ შეიძლება მათი დამალვა-ეს გამოიყენება სქემების განტვირთვისათვის, უკეთ აღქმისათვის.

სახელის შეცვლა შეიძლება, მაგრამ საფრთხილოა-შეიძლება აღმოჩნდეს ერთნაირი სახელები, რაც გამოიწვევს გაურკვეველობას, ამიტომ, ასეთი ოპერაცია რთული სქემებისთვის უკეთესია შესრულდეს **გლობალური ანოტატორის** მეშვეობით.

წარწერის გადაადგილება საჭიროა იმ შემთხვევაში, თუ რაიმე მიზეზით ამ ადგილზე უნდა განთავსდეს სხვა ობიექტი, მაგალითად, გამტარი. ამისათვის, უნდა მოვნიშნოთ ელემენტი ან რომელიმე მისი ატრიბუტი(სახელი, სიდიდე, TEXT) რომლებიც ავტომატურად იწვევენ ელემენტის მონიშნვას და მარცხენა კლავიშით გადავათრიოთ შერჩეული წარწერა-სურ. 2.17.



სურ. 2.17.

მოხერხებულობის და სიზუსტის მიზნით შესაძლებელია ბადის ბიჯის შემცირება 100th (2,54 მმ)-დან 50th (1,27 მმ)-მდე კლავიშით F2 ან ოპციით **ფიქსირება**(View => **Snap 50th**). ოპერაციის შემდეგ დავუბრუნდეთ 100th-ს კლავიშით F3. შევნიშნავთ, რომ ეს ოპერაციები სავალდებულო არ არის - ISIS რეჟიმი ოპციით **ფიქსაცია რეალურ დროში** (**Real Time Snap**), ისედაც ახერხებს ელემენტთა განთავსებას.

ISIS გააჩნია ოთხი ხერხი ელემენტებზე დასახელების მინიჭების და რედაქტირებისათვის, რომლებიც ხელმისაწვდომია მენიუდან **Tools**.

- **ხელით ანოტირება**(**Manual Annotation**), როდესაც დასახელება ხელით ეწერება. მისი რედაქტირება შეიძლება იმავე წესით, როგორც დანარჩენი

ელემენტებისა დიალოგური ფანჯრის საშუალებით, სადაც შეიტანება აღნიშვნა(Reference), სიდიდე (Value) და სხვა.

- თვისებათა მიმნიჭებელი ინსტრუმენტი (Property Assignment Tool- PAT)- მისი საშუალებით შესაძლებელია ფიქსირებული ან ინკრემენტირებული სახელების მინიჭება ან ყველა ობიექტზე ან ყველა მონიშნულ ობიექტზე, (როგორც მიმდინარე, ისე დანარჩენ ფურცლებზე) და ყველა მომდევნო ობიექტზე. ის უფრო ჩქარია ვიდრე ხელით ანოტირება, მაგრამ უფრო ნელია ვიდრე ავტომატური ანოტირება(Automatic Annotator).
- გლობალური ანოტატორი(Global Annotator)-ის უზრუნველყოფს მთელი პროექტის ანოტირებას უმოკლეს დროში, მაგრამ არაინტერაქტიურია, ამიტომ პროცესში უშუალო ჩარევა ადამიანის მიერ გამორიცხულია. დასაშვებია მხოლოდ შემდგომი კოექცია.
- ანოტირება რეალურ დროში(Real Time Annotation)-ის უზრუნველყოფს ავტომატურ ანოტირებას ელემენტის სქემაში შეტანის დროს. ის არაინტერაქტიურია - შეიძლება მხოლოდ შემდგომი კოექცია.

როგორც წესი, პრაქტიკაში იყენებენ ამ უკანასკნელს, ხოლო შემდეგ თუ საჭიროა, PAT ან ავტომატური ანოტირებას.

განვიხილოთ უფრო დაწვრილებით PAT ფუნქცია.

დავუშვათ, ჩვენ გვინდა გადავანოტიროთ წინააღმდეგობის სიმრავლე დაწყებული R5-დან: R5, R6, R7 და ა. შ. შევდივართ ოპციაში Tools => Property Assignment Tool. გაიხსნება დიალოგური ფანჯარა-სურ.2.18:

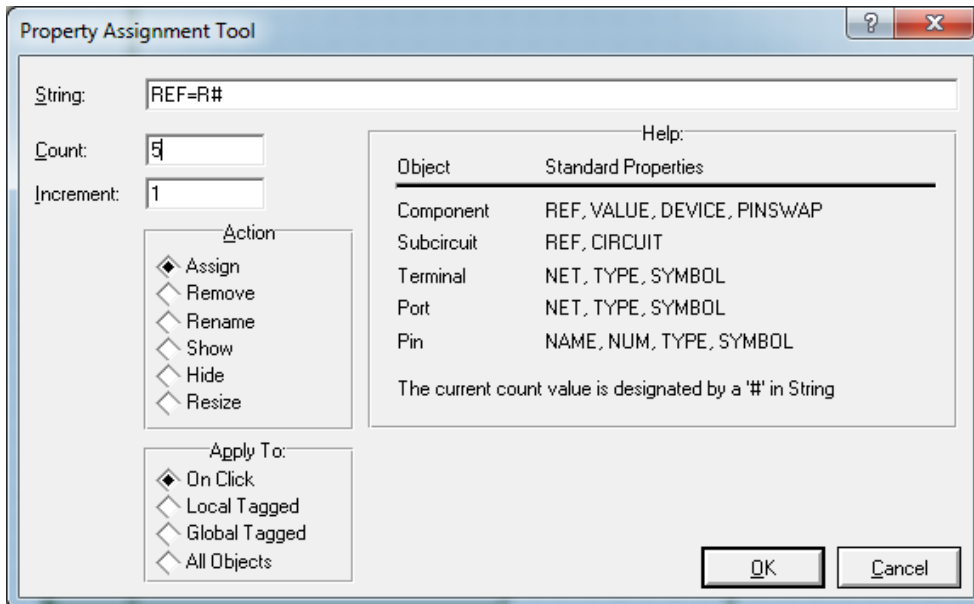
ველში სტრიქონი(String) შევიტანოთ REF=R#, ველში თვლა(Count) - 5, ველში ნამატი(increment) - 1; ფანჯარაში გამოყენება(Apply To) უნდა იყოს ჩართული ოპცია დაჭერაზე (On Click), და OK. ამის შემდეგ ვაწკაპუნებთ მარცხენა კლავიშით რეზისტორებზე მიმდევრობით დაწყებული R5-დან. ყოველ ჯერზე სიმბოლო # შეიცვლება მნიშვნელობით თვლის ველიდან და მიენიჭება დაწკაპუნებულ რეზისტორს, ხოლო # სიმბოლოს მნიშვნელობა გაიზრდება სიდიდით ნამატის ველიდან.

ეს ოპცია არის ერთერთი მძლავრი ინსტრუმენტი პროექტირებისთვის, მაგრამ თხოულობს ყურადღებას, რათა არ წარმოიშვას ერთსახელა კომპონენტები. ანალოგიურად, PAT ფუნქციით შესაძლებელია ელემენტის სხვა მონაცემების კორექტირება.

ასევე მნიშვნელოვანია გლობალური ანოტატორი. მას შეუძლია ყველა ელემენტის ანოტირება, მათ შორის იმათიც, რომელთაც აღნიშვნაში უწერია ნიშანი "?".

მას გააჩნია ორი რეჟიმი:

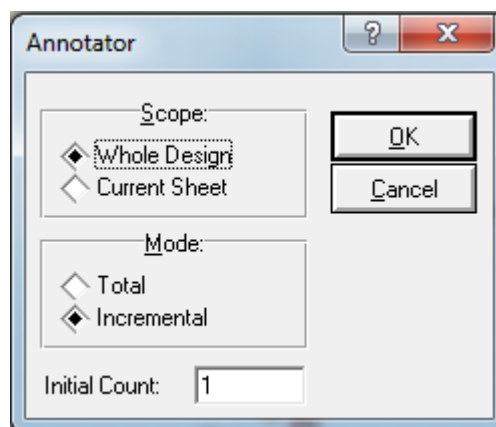
- ანოტაცია ნამატით(Incremental Annotation), ის მოქმედებს მითითებულ საზღვრებში(მიმდინარე გვერდი ან მთელი პროექტი) და მხოლოდ არაანოტირებულ ელემენტებზე.



სურ.2.18.

სრული ანოტირება(Total Annotation). მოქმედებს მიმდინარე გვერდის ან მთელი პროექტის საზღვრებში ყველა ელემენტზე.

რეჟიმების დაყენება შეიძლება დიალოგური ფანჯრიდან, რომელიც გამოიძახება **Tools => Global Annotator**, ხოლო შემდეგ ოპციები შეირჩევა ველებიდან: **Scope** და **Mode**.

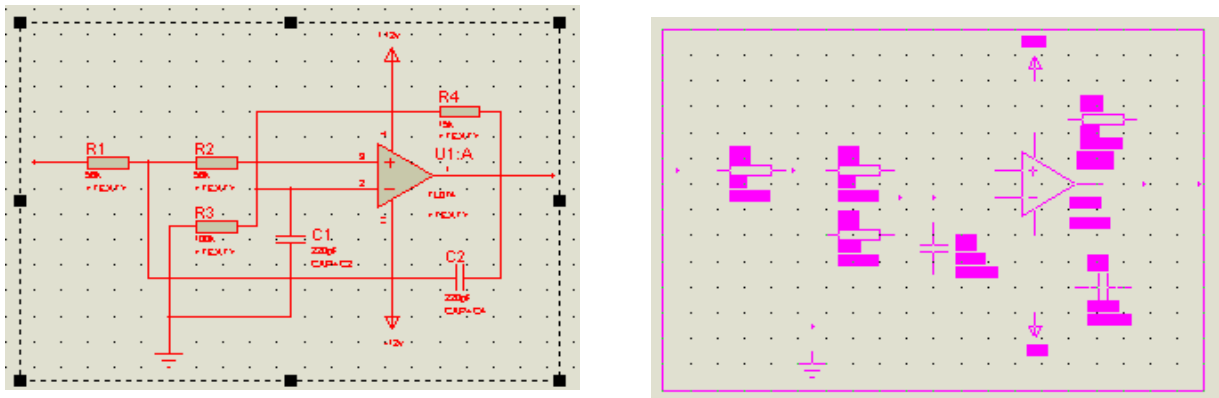


სურ.2.19.

2.6. სქემის ბლოკების რედაქტირება

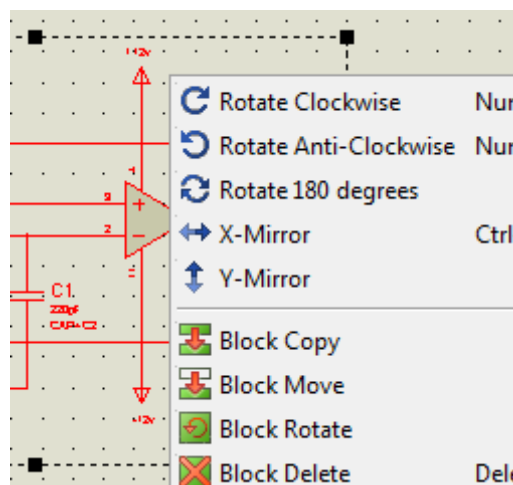
ISIS საშუალებას იძლევა იოლად მოვახდინოთ სქემის ნაწილების გადაადგილება, რომელიც მსგავსია ადრე განხილული ოპერაციებისა: ჩავსვათ სქემის შერჩეული ნაწილი კონტეინერში(მოვნიშნოთ)-კურსორის გადაადგილება დაჭერილი მარჯვენა ან მარცხენა კლავიშით დიაგონალზე, კლავიშის გაშვება, კურსორი კონტეინერის საზღვარზე ან მის შიგნით ღებულობს ჯვარედინი ისრის ფორმას, რაც მიუთითებს ობიექტის

გადაადგილების შესაძლებლობაზე, კურსორის გადაადგილება საჭირო მიმართულებით, მარცხენა კლავიშის აწვევით ობიექტის დაფიქსირება ადგილზე და კონტინერის გარეთ მარცხენა კლავიშის დაწკაპუნებით მონიშვნის მოხსნა - სურ.2.20.



სურ.2.20.

ასევე შესაძლებელია კონტინერით მონიშნულ სქემაზე ასლირების, გაგდების და ბრუნვის ოპერაციების ჩატარება - სურ. 2.21. მაგრამ, ბრუნვის ოპერაციების ჩატარება არ შეიძლება ფიქსირებული ორიენტაციის სქემაზე, მაგ. გრაფიკებზე.



სურ.2.21.

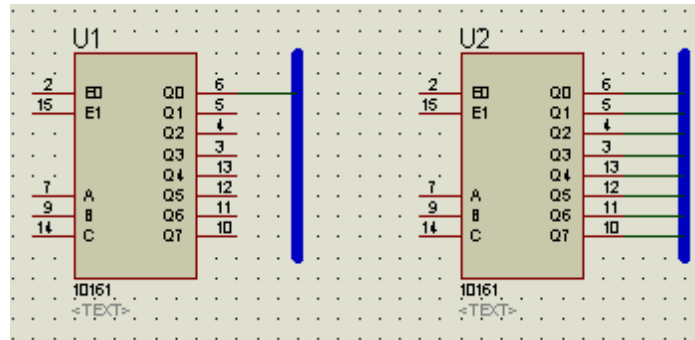
ანოტირება უნდა დამთავრდეს მანამ, სანამ შეერთებათა ცხრილი გადაეცემა ARES-ს. შემდგომი კორექცია უფრო რთულია.

2.7. გამტარების და სალტეების გატარება

ელექტრული სქემების დამუშავებაში ერთერთი შრომტევადი სამუშაოა ელექტრული კავშირების დამყარება და მათი ასახვა გამტარების და სალტეების სახით.

როგორც აღნიშნული იყო, ISIS-ს გააჩნია შესანიშნავი საშუალება გამტარების ეკონომიური გატარებისა სქემაზე სალტეების(BUS) სახით. ის წარმოადგენს ერთეულოვანი გამტარების ერთობლიობას, რომელიც უერთდება მას სქემის ერთი ან რამდენიმე კომპონენტიდან, ხშირად პარალელური ხაზების სახით. არსებობს ხერხი,

რომლის დროსაც საკმარისია ერთი ხაზის გატარება, რომ ელემენტის მორიგ გამომყვანზე ორჯერ მარცხენა წკაპით ხაზი ავტომატურად, პარალელურად გაივლის სალტემდე. ხაზი აბსოლუტურად იმეორებს პროტოტიპის გაყვანის მიმართულებას, ფორმას და სიგრძეს. ამ დროს მიზანშეწონილია გამორთული იყოს გამტარის მხოლოდ კოორდინატებზე გატარების ოპცია-**Wire Autorouter**. ეს ხერხი გამოსადეგია კოპონენტებს შორის კავშირის დასამყარებლად, როდესაც კავშირები ერთგვაროვანია. მაგალითი-სურ. 2.22-ზე.



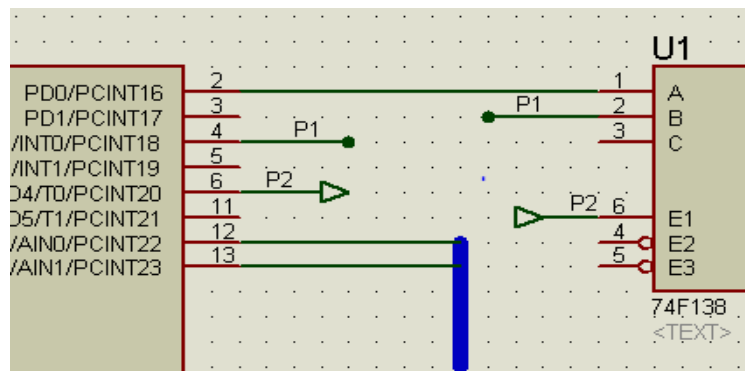
სურ. 2.22.

თუ გამტარის ბოლო არ მიუერთდა რაიმე კონტაქტს, ის მთავრდება ტერმინალის წერტილით, რომელიც შემდეგ შეიძლება გაგრძელდეს. ამასთან დასაწყისი უნდა იყოს დასაშვები ობიექტიდან: კოპონენტების გამომყვანი, სხვა გამტარი ან სალტე.

2.8. გამტარების და სალტეების მარკირება

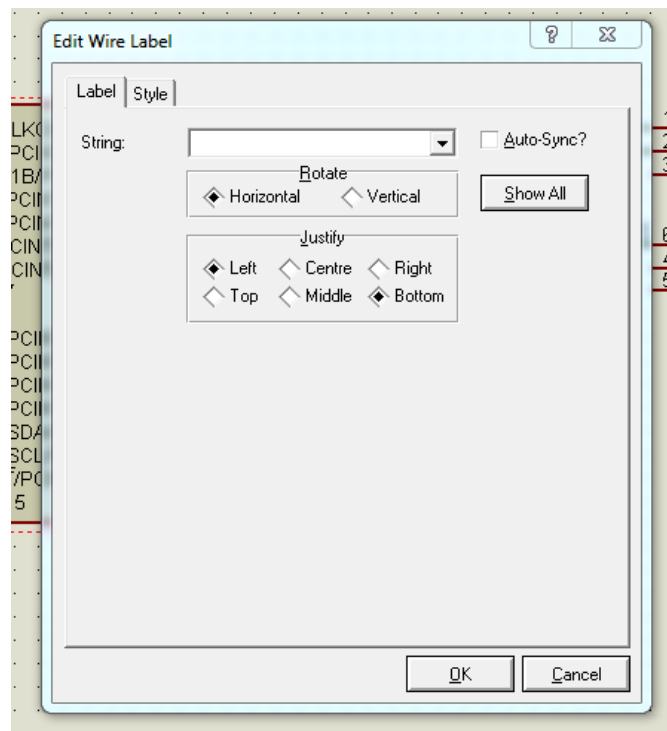
როგორც აღნიშნული იყო, გამტარების და სალტეების მარკირება გამოიყენება კავშირის არხების დასამალად სქემის განტვირთვის მიზნით. პროტეუსი თვლის, რომ ერთსახელა არხები გაერთიანებულია და კავშირების სიაში ის წარმოდგება ერთ ერთეულად.

არსებობს ელექტრული კავშირის ასახვის სამი ვარიანტი: მაკავშირებელი უწყვეტი ხაზი, გაწყვეტილი ხაზი გადასავლელი წერტილებით ორივე ნაწილზე(Junction Dot) და ერთი სახელით და გაწყვეტილი ხაზი ტერმინალებით. მაგ. სურ. 2.23-ზე.



სურ. 2.23.

ქდე(მარკერი, **Label**) P1 აერთიანებს გაწყვეტილ ხაზს გადასავლელი წერტილებით, P2 აერთიანებს გაწყვეტილ ხაზს ტერმინალებით, ხოლო მათ ზევითაა უწყვეტი ხაზი. ერთეული გამტარების ხელით მარკირებისთვის პიქტოგრამა **LBL**-ით ვიძახებთ რეჟიმს **Wire Label Mode**, მიგვყავს კურსორი გამტართან იმ ადგილას სადაც უნდა დაისვას სახელი, გამოჩნდება თეთრი ირიბი ჯვარი, მარცხენა წკაპით ვიძახებთ რედაქტირების ფანჯარას **Edit Wire Label** (სურ. 2.24).

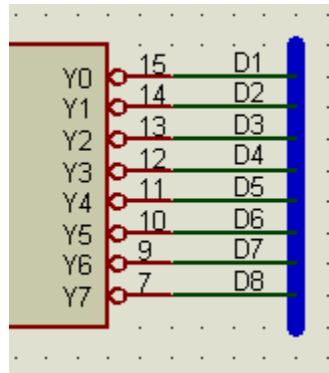


სურ. 2.24.

Label ჩანართის **String** ველში ან ხელით ან ჩამოშლადი სიიდან შეგვყავს ახალი სახელი. სხვა ოპციების დანიშნულება ცხადია სახელებით. **Style** ჩანართში ვაყენებთ ტექსტის პარამეტრებს.

Property Assignment Tools (PAT) ოპციის გამოყენება ჩვენ უკვე განმარტებული გვაქვს წინაღობების მარკირებისთვის. იგივე ოპციის გამოყენება შესაძლებელია გამტარებისთვის. აქ მოტანილია მაგალითი დეშიფრატორის და სალტის კავშირებისთვის. ვიძახებთ **PAT** ოპციას (პიქტოგრამა ან ასო **A**). დეშიფრატორის გამოსასვლელი გამტარები გვინდა დაინომროს რიგით: **D1**, **D2**, ...და ა. შ. გამტარებისა და ტერმინალებისათვის გამოიყენება თვისება **NET(Network-აქ წრედი)**, ამიტომ რედაქტორის ფანჯრის **String** ველში შეგვყავს **NET=D#**, სადაც # ადგილია ცვლადი სიმბოლოსთვის-ნომრისთვის(ეს სიმბოლო შეიძლება იყოს თავში). **Count** ველში შეგვყავს საწყისი მნიშვნელობა 1, **Increment** -ში 1, **Action**-ში მოვნიშნავთ **Assign**, **Apply To**-ში **On Click**-ს და ვაჭერთ **OK**.

ამ მოსამზადებელი სამუშაოების შემდეგ, რიგრიგობით ჩამოვუვლით დეშიფრატორის წრედებს და საჭირო ადგილზე ვაკეთებთ მარცხენა წკაპს მაშინ, როდესაც მიმთითებელი ხელის მარჯვნივ გამოჩნდება მწვანე კვადრეტი-სურ. 2.25.



სურ. 2.25.

ამის შემდეგ, კიდევ ვიძახებთ **PAT** რედაქტორს და ვაჭერთ **Cancel**-ს. ამით მთავრდება რედაქტორის მუშაობა. **PAT**-ის სხვა ოპციებზე აქ არ შევჩერდებით.

თავი 3. ციფრული გრაფიკების გამოყენება

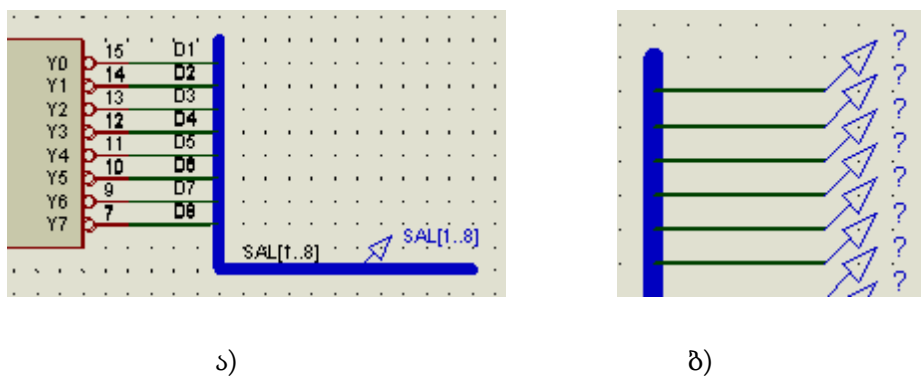
3.1. სასინჯი ზონდების გამოყენება მოდელირებისას

სქემების მრავალფეროვნებიდან გამომდინარე, პროტეუსში გამოიყენება ერთი და მრავალარხიანი ვირტუალური საზომი ხელსაწყოების ნაირსახეობები: ვოლტმეტრები, ამპერმეტრები, ოსცილოგრაფები, ლოგიკური ანალიზატორები და სხვა დამხმარე ხელსაწყოები. ყველა შემთხვევაში ობიექტიდან ინფორმაციის მოხსნისათვის მოიხმარება ზონდი-სასინჯები.

როგორც უკვე აღნიშნული იყო, ერთეულ წერტილში ძაბვის გასაზომად გამოიყენება ვოლტმეტრი, რისთვისაც საჭიროა: ეკრანის მარცხენა მხარეს, რეჟიმების პანელზე მოინიშნოს ვოლტმეტრის პიქტოგრამა, კურსორი მიყვანილიქნეს გამტართან, გამოჩნდება თეთრი X, მარცხენა წკაპი და გამტარის გვერდით აისახება ისარი წვერთან იდენტიფიკატორით, რომელიც შეიძლება შეიცვალოს რედაქტირებით. იდენტიფიკატორი შეიძლება იყოს გამტარის სახელი(თუ მინიჭებულია), ან მიერთებული რადიოელემენტის და მისი კონტაქტის სახელი, ხოლო თუ კონკრეტული სახელები არ არსებობს- კითხვის ნიშანი.

ანალოგიურად განთავდება ზონდები მრავალარხიან სალტეზე იმ განსხვავებით, რომ კურსორი უნდა იყოს ახლოს როგორც სალტესთან ისე სახელთან. იდენტიფიკატორი იქნება სალტის სახელი ან კითხვის ნიშანი, რომელიც რედაქტირდება.

ასევე დაყენდება სასინჯები სალტის განშტოებად გამტარებზე. უსახელი გამტარებზე დაყენებული ვოლტმეტრების სახელებში კითხვის ნიშანები შეიცვლება კონკრეტულით, როგორც კი ის მათ მათ მიენიჭება, ან მიენიჭება და სიმულატორი გაიშვება - სურ. 3.1.



სურ. 3.1.

როგორც აღნიშნული იყო, დენის სასინჯის ისარი მიმართული უნდა იყოს გამტარის გასწვრივ. თუ ის ემთხვევა დენის მიმართულებას სიდიდე აისახება დადებით რიცხვებში, თუ არ ემთხვევა-უარყოფითებში.

ისევ გავიმეორებთ: რადგანაც ხელსაწყოები ძაბვას ზომავენ მიწის მიმართ, მიწის მიმართ განმხლოებულ გამტარზე განაზომი იქნება შეცდომითი.

შეგვრდეთ სასინჯების თვისებების რედაქტირებაზე, განსაკუთრებით ძაბვისაზე, როგორც ყველაზე გამოყენებადზე.

აქ და საერთოდ უნდა გავითვალისწინოთ ელექტრონული ელემენტების პარამეტრების მოცემის ფორმები პროტეუსში:

ძირითადი სიდიდეებისათვის:

მაზვისთვის $+5=5$ ვოლტს, წინაღობისთვის $510=510$ ომს, ტევადობისთვის $1=1$ ფარადს, დროისთვის $10=10$ წამს, სიხშირისთვის $1000=1000$ ჰერცს. მისაღებია შემდეგი ვარიანტებიც:

A -ამპერი, **V**- ვოლტი, **Ohm**(ან **R**)-ომი, **F**ფარადი, **H**-ჰენრი, **S**-წამი, **Hz**-ჰერცი.

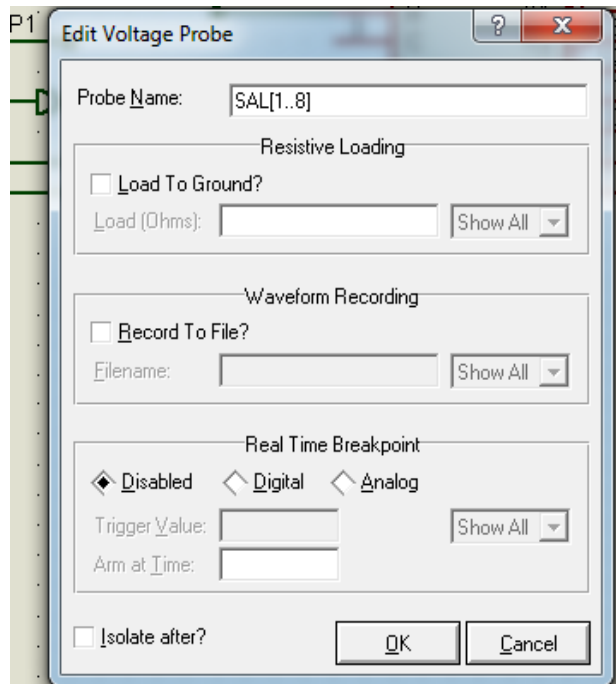
ნაწარმოები სიდიდეებისათვის გამოიყენება მამრავლები:

პიკო - $p=10^{-12}$, ნანო- $n=10^{-9}$, მიკრო- $\mu=10^{-6}$, მილი- $m=10^{-3}$, კილო- $k=10^3$, მეგა- $M=10^6$, გიგა- $G=10^9$, ტერა- $T=10^{12}$. გადასვლა ძირითად და ნაწარმოებ სიდიდეებს შორის ადვილია. მაგ.: 1S შეიძლება ჩაიწეროს როგორც 1000mS.

მაზვის სასინჯის თვისებების რედაქტირების ფანჯარას აქვს სახე(სურ. 3.2);

ალამი **Load To Ground**(დატვირთვა მიწაზე) მოითხოვს შიგა წინაღობის იმიტაციის ჩართვას. მაშინ, გააქტიურებულ ველში **Load(Ohm)**, უნდა ჩაიწეროს წინაღობის მნიშვნელობა. ალამი **Record To File** იძლევა შედეგების მაგნიტოფონზე შენახვის უფლებას-ამჟამად არააქტუალურია.

განვიხილოთ უფრო საჭირო ოპცია-წყვეტის გამომუშავება სასინჯიდან სიგნალის მიხედვით. ამ დროს სიმულაცია გადაგის პაუზის რეჟიმში. ეს რეჟიმი მისახერხებელია მოვლენებს შორის ინტერვალის გასზომად, მაგ., რაღაც წერტილში ციფრულ სკემაში



სურ. 3.2.

ლოგიკური ერთების ან ნოლების მიმდევრობით გამოჩენა. ამისთვის, **Real Time Breakpoint** ველში ვაყენებთ ალამს **Digital**, ხოლო გააქტიურებულ ველში **Trigger Value** ვწერთ ერთს და ვუშვებთ სიმულაციას. გაჩერებისას ვიწერთ ჩვენებას და ისევ ვუშვებთ სიმულაციას. მეორე გაჩერების შემდეგ მიღებულ ჩვენებას ვაკლებთ პირველს და ვადგენთ დროის

ინტერვალს. ეს რეჟიმი გამოსადეგია ანალოგური ცქემისთვისაც: ვრთავთ ალამ **Analog, Trigger Value**-ში შეგვაქვს ძაბვის მნიშვნელობა და ვიმეორებთ აღწერილ მოქმედებებს.

ველი **Am at Time** -ს დანიშნულებაა დროის დაყენება, საიდანაც უნდა დაიწყოს სიმულაცია, მაგ., 0,5 წამი, 1 წამი. ზონდი ანალოგურ სიგნალს ყოველთვის ანახებს რიცხვითი სახით, მაგრამ ლოგიკური სიგნალი შეიძლება გამოსახოს ქვევით მოყვანილი სახით(სურ. 3.3).

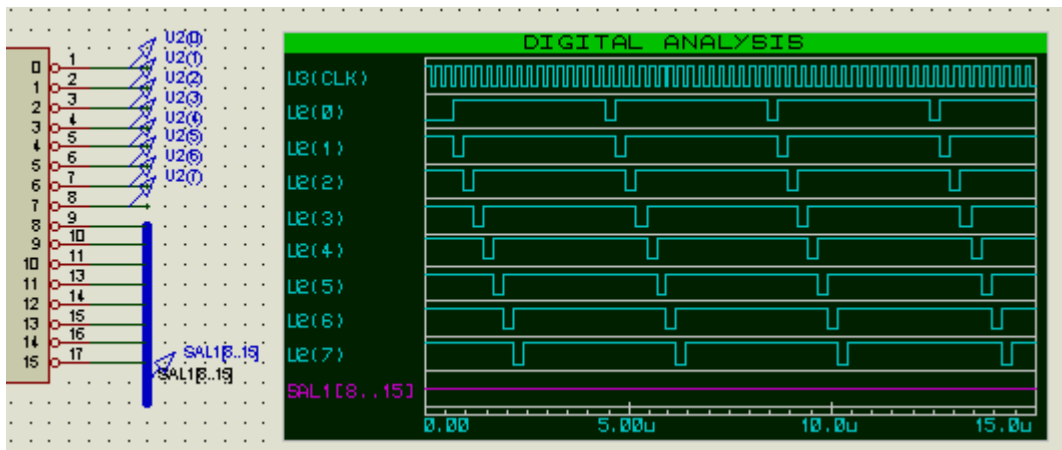
State Type	Keyword	Description
Power High	PHI	Логическая 1 шина питания
Strong High	SHI	Логическая 1 активный выход
Weak High	WHI	Логическая 1 пассивный выход
Floating	FLT	Высокоимпедансное состояние
Undefined	WUD	Неопределенное между 1 и 0
Contention	CON	Неопред. конфликт сигналов
Weak Low	WLO	Логический 0 пассивный выход
Strong Low	SLO	Логический 0 активный выход
Power Low	PLO	Логический 0 шина питания

სურ. 3.3.

3.1. ციფრული გრაფიკების გამოყენება

ციფრული სქემების მუშაობის გრაფიკული ასახვა ერთერთი მძლავრი საშუალებაა მათი მუშაობის ანალიზისათვის. ISIS-ში ამისთვის არსებობს მრავალარხიანი ოსცილოგრაფი. ეკრანის მარცხენა მხარეს, რეჟიმების პანელზე ვირჩევთ ღილაკს **Graph Mode**, სელექტორში **Digital** და სამუშაო ველის თავისუფალ ადგილზე დაჭერილ მარცხენა კლავიშს ვამოძრავებთ წარმოსახვითი ეკრანის დიაგონალზე. გამოჩნდება ეკრანის ჩარჩო, რომელიც მარცხენა კლავიშის დაწკაპუნებით გარდაიქმნება შავ მართკუთხედად. მონიშნით და ცნობილი მანიპულაციებით შეიძლება მათკუთხედის ზომების და მდებარეობის შეცვლა. სურ. 31-ზე მოცემულია დემიფრატორის ცალკეული გამოსასვლელების ასევე, სალტის დროითი დიაგრამა.

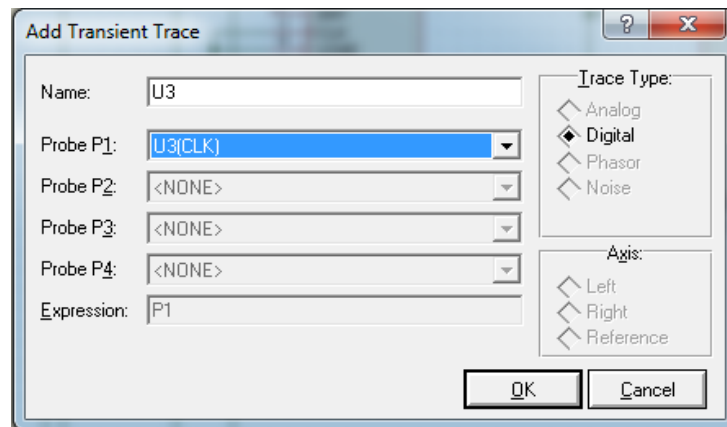
ეკრანზე გრაფიკის განსათავსებლად არსებობს ორი ხერხი:



სურ. 3.4.

მარცხენა წკაპით ვირჩევთ ზონდს და დაჭერილი მარცხენა კლავიშით გადაგვაქვს შავ ეკრანზე. კლავიშის გაშვების შემდეგ, ვერტიკალური საკოორდინატო ღერძის მარცხნივ გამოჩნდება ზონდის დასახელება, ხოლო მარჯვნივ ხაზი ნოლის მდგომარეობის ჩვენებით.

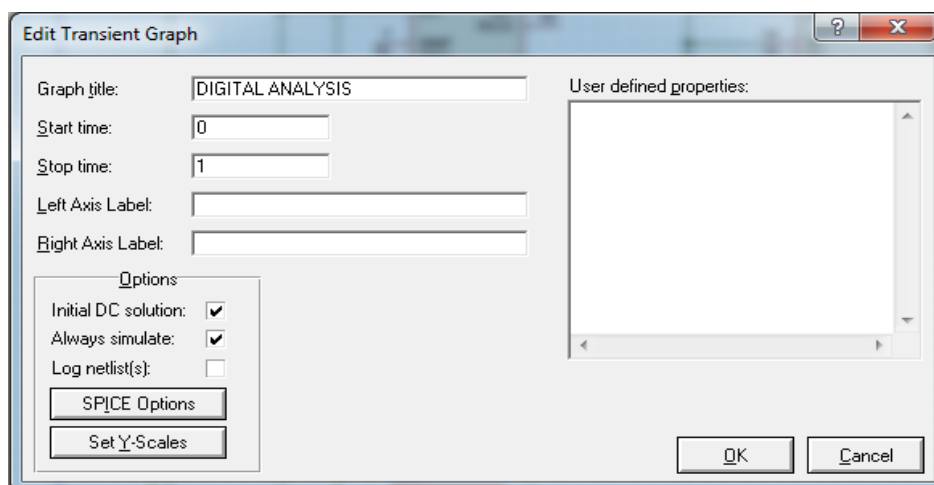
მეორე ხერხით: მარჯვენა წკაპით შავ ეკრანზე ვიძახებთ დიალოგურ ფანჯარას და მასში ვირჩევთ ოპციას **Add Traces...** . ის ხსნის ფანჯარას, სადაც ზონდის სახელის მიხედვით ჩამოვშლით მენიუს და ვირჩევთ სიგნალს სახელის მიხედვით(სურ. 3.5).



სურ. 3.5.

ამგვარად ვათავსებთ ყველა გრაფიკს ეკრანზე. გრაფიკი შესრულებაზე შეიძლება გავუშვათ მარჯვენა წკაპით გამოძახებულ მენიუმში ავირჩიოთ **Simulate Graph** ან დავარტყათ შუალედის კლავიშას.

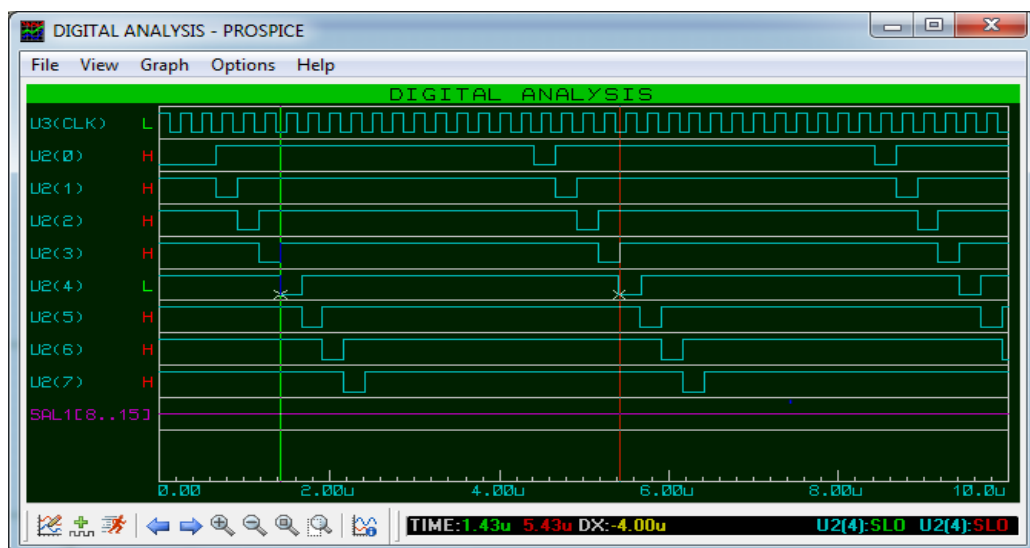
როგორც ყველა ობიექტზე, ეკრანის პარამეტრებზეც შეიძლება გარკვეული მანიპულაციების ჩატარება. ცნობილი გზებით შევდივართ ოპციაში **Edit Properties** და ვაყენებთ საჭირო პარამეტრებს(სურ. 3.6).



სურ. 3.6.

აქ შეიძლება სახელის შეცვლა, მარცხენა და მარჯვენა Y ღერძებზე სახელის დარქმევა, მაგრამ მრავალი არხების დროს ის არ მოქმედებს, **Inicial DC Solution** არაქტუალურია, **Always simulate** აღმის მოხსნით შესაძლებელია გრაფიკის შენახვა დაუზიანებლად სხვა მანიპულაციების დროს. უფრო საინტერესოა ფანჯრები **Start Time**(საწყისი დრო) და **Stop Time**(დამთავრების დრო)-განსაზღვრავენ გრაფიკის აღების ინტერვალს. მისი ზომები უნდა იყოს შეთანხმებული სქემის მუშაობის სიხშირესთან თვალსაჩინო გრაფიკის მისაღებად.

სრულყოფილი ანალიზი ყველაზე მოხერხებულია ჩატარდეს მაქსიმალური ეკრანის დროს, რომელიც ყენდება კონტექსტური მენიუდან ოპციით **Maximize (Show Window)**(სურ. 3.7).



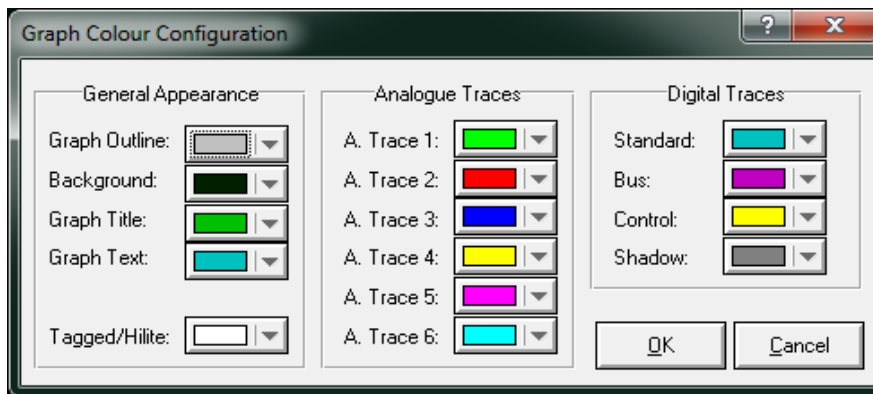
სურ. 3.7.

ნახაზზე ორი მარკერია-მწვანე და წითელი. მისი მდებარეობა X ღერძის მიმართ მოცემულია რიცხობრივად ბოლო სტრიქონზე მწვანე ფერით. მისი გადაწევა შეიძლება ნებისმიერ ადგილზე დაჭერილი მარცხენა კლავიშით. რომელიმე გრაფიკზე მარცხენა წვაპი იწვევს ამ წერტილზე მწვანე მარკერის გადმოსვლას და გრაფიკის მონიშვნას ჯვარედინი ხაზებით. იგივე შეიძლება გაკეთდეს წითელი მარკერისთვის მხოლოდ, უნდა იყოს დაჭერილი კლავიშა **CTRL**. ბოლო სტრიქონზე მინაწერი **DX:-1.68m** უჩვენებს სხვაობის სიდიდეს მარკერებს შორის. მწვანეს აკლდება წითელი ამიტომ, სხვაობა უარყოფითია.

მაქსიმუმ ეკრანს გააჩნია თავისი ძირითადი მენიუ ზევდა კიდეზე და ინსტრუმენტების პანელი ქვედა კიდეზე, რითაც შეიძლება მეტი მანიპულაციების ჩატარება სხვადასხვა პარამეტრებზე.

File -ს საშუალებით გრაფიკი შეიძლება ამოიბეჭდოს შავ-თეთრ ან ფერად ფერებში, ან შენახულიქნეს ფაილი სხვადასხვა გრაფიკულ ფორმატებში-**bitmap (BMP)**, **AutoCAD (DXF)**. **View** და **Graph**-ის ოპციები ფაქტიურად მეორდება ინსტრუმენტებში-განვიხილავთ იქ. მენიუ **Options** იმეორებს ოპციებს ISIS-ის ძირითადი მენიუდან **Template** და **System**, მაგრამ აქ ადვილად ხელმისაწვდომია. ამ ოპციებიდან აჟამად საინტერესოა **Set Graph**

Colours..., რომელიც საშუალებას იძლევა შევარჩიოთ ფერები(სურ. 3.8). შეგახსენებთ, რომ ამის გაკეთება შეიძლება **Template**-დანაც, აქ მოხერხებულობისთვისაა.



სურ. 3.8.

მარცხენა სვეტზე მოცემულია ძირითადი ფერები- საერთო ყველა გრაფიკისთვის. **Outline** –ზადის და კოორდინატების ხაზები, **Background** – ფონი, **Title** – სათაურები, **Text** – ტექსტი, **Tagged** –საკონტროლო წერტილები. შუა სვეტში მოცემულია ექვსი ფერი ანალოგური გრაფიკებისთვის, ხოლო ბოლოში ციფრული გრაფიკებისთვის. ყველა ამ ფერის შეცვლა შეიძლება ჩამოშლადი მენიუდან.

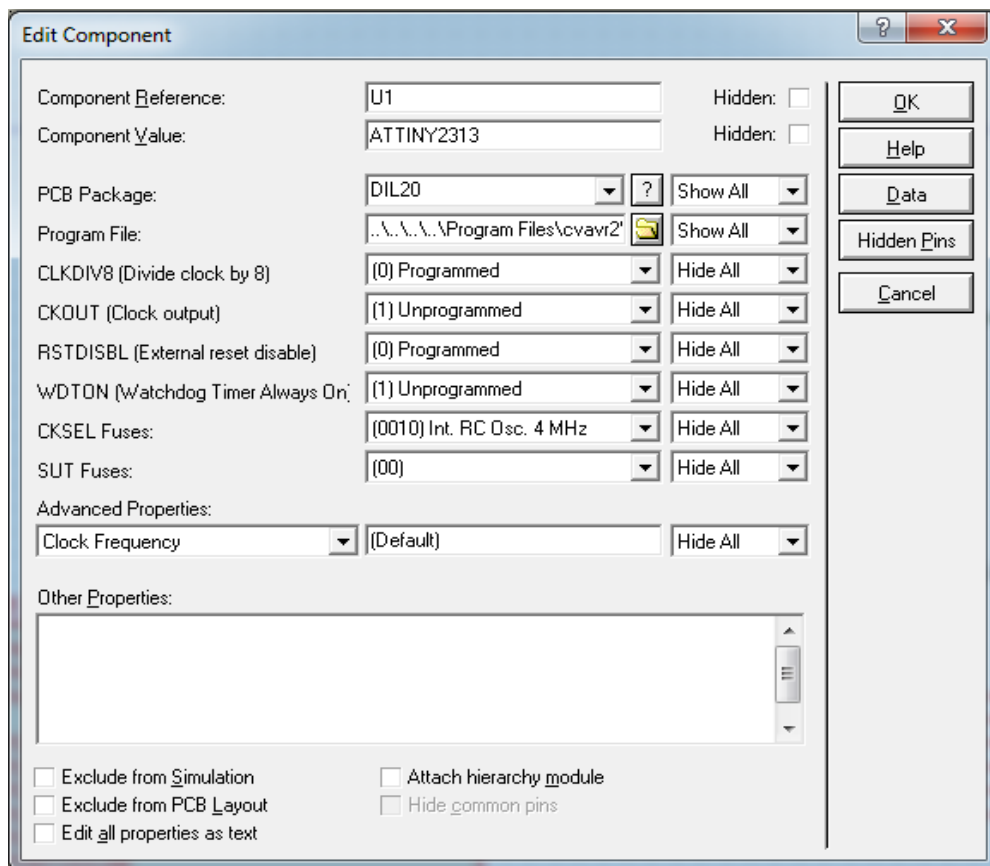
რაც შეეხება ინსტრუმენტებს, მიმდევრობით მოცემულია ნაცნობი ოპციები: **Edit Graph**, **Add Traces** და **Simulate Graph**, ამიტომ მათზე არ შევჩერდებით. ისრებიანი ღილაკები იწვევენ ეკრანის გადაადგილებას შესაბამისი მიმართულებით. შემდეგ მოთავსებულია ოთხი ღილაკი ლუპის სახით, რომელთა ფუნქციები იგივეა რაც ძირითად მენიუში, მხოლოდ უნდა გავითვალისწინოთ, რომ მასშტაბირება აქ ხდება X კოორდინატის მიხედვით. ოპცია **Zoom To View Entire Sheet**-ის თავისებურება აქ ის არის, რომ მთელი ფურცლის ქვეშ იგულისხმება დრო **Start Time** და **Stop Time** შორის. ერთიც, ეკრანის დახურვის შემდეგ-ღილაკი X ეკრანის ზედა ბოლოში ან ოპციით **Restore (Close Window)** კონტექსტური მენიუდან, შეტანილი ცვლილებები ნარჩუნდება.

თავი 4. პროგრამების გამართვა

4.1. პროგრამის შეტანა მიკროკონტროლერში

როგორც ნებისმიერ კომპონენტს, მკ-საც გააჩნია რედაქტირების ფანჯარა სადაც მოწმდება და კორექტირდება მისი თვისებები.

ამ თვისებებიდან ერთერთი პირველი უნდა დაკორექტირდეს კვების წყაროების სტანდარტული, ზოგადი სახელები **VCC** და **CND**, თუ მოსალოდნელია, რომ სქემაში კვება იქნება სხვადასხვა ტიპის სხვადასხვა წყაროებიდან. ამისთვის, საჭიროა ჩვეულებრივი წესით გამოვიძახოთ მკ-ს რედაქტირების ფანჯარა და შემდეგ ღილაკით **Hidden Pins** გავხსნათ წყაროების რედაქტირების ფანჯარა **Edit Hidden Power Pins**(სურ. 4.1).



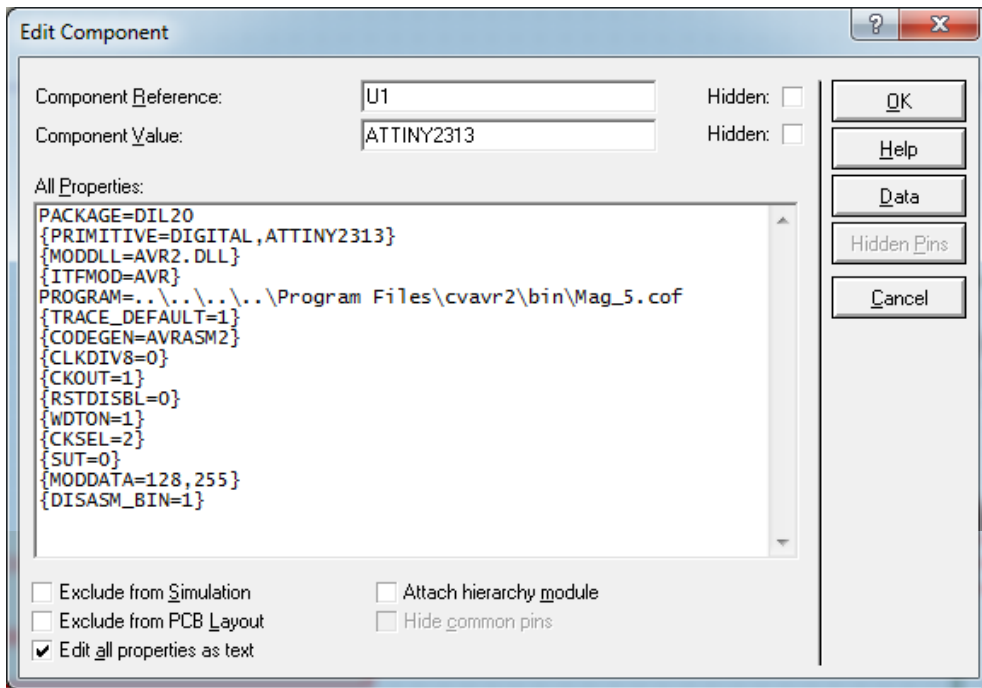
სურ. 4.1.

მკ-ს რედაქტირების ფანჯარას პროტეუს 7.4 ვერსიიდან და ზევით აქვს სურ. 4.1-ზე მოცემული სახე: აქ, მიმდევრობით, ზევითან ქვევით ჩამოთვლილია მკ-ს თვისებები: სახელი სქემაზე, ტიპი, კორპუსის ტიპი, გზა პროგრამული ფაილისკენ, რომელიც უნდა ჩაიტვირთოს მკ-ში, სიხშირის დამყოფის ოპციები, ნოლზე დაყენების წესი, **ფუზეს მუშაობი სიხშირე** და სხვა. ღილაკების ფუნქციები გასაგებია დასახელებიდან.

Advanced Properties ველების ოპციებით შეირჩევა მკ-ს მუშაობის სიხშირე, რომელიც დამოკიდებულია ფუზეს რეჟიმზე და სხვა პარამეტრები.

მარცხნივ, ქვევით მოცემული ალმებით მიეთითება, ზევიდან: გამოირიცხოს კომპონენტი სიმულაციიდან, გამოირიცხოს პლატის ტრასირებიდან, თვისებები რედაქტირდეს ტექსტურ ფორმაში.

ბოლო ალმის დაყენებისას რედაქტორის ფანჯრის **Other Properties** ველში გამოჩნდება პარამეტრების ჩამონათვალი, რომელთა კორექტირება შეიძლება(სურ. 4.2).



სურ. 4.2.

პროგრამის გზის ჩვენებისას განსაკუთრებული ყურადღება უნდა მიექცეს ფაილის ტიპს- გაფართოებას. **Program File** ველში უნდა მიეთითოს ფალამდე გზა და ფაილის სახელი გაფართოებით:

.cof - თუ თქვენ აპირებთ აწარმოთ გამართვა **C** ენაზე დაწერილი ამოსავალი პროგრამის კონტროლით, რომელიც შედგენილია **CodeVisionAVR** - **CVAVR** კომპილატორის გამოყენებით.

.elf - თუ პროგრამა დამუშავებულია **C** ენაზე **WinAVR** კომპილატორის გამოყენებით.

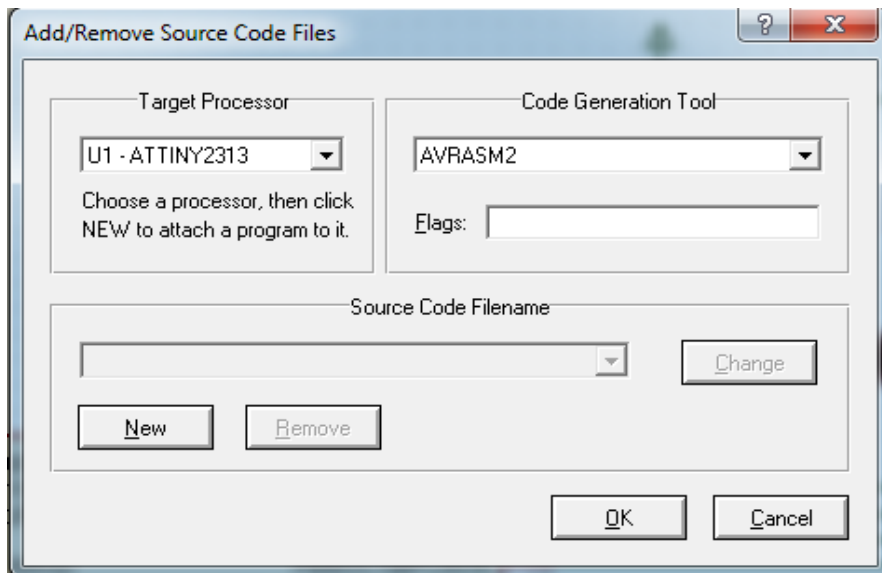
UBROF - თუ პროგრამა დამუშავებულია **C** ენაზე **IAR** კომპილატორის გამოყენებით.

.obj - თუ პროგრამა დამუშავებულია **PROTEUS**-ში **BASCOM-AVR** კომპილატორის გამოყენებით.

.hex - ეს რეალურ მკ-ში ჩასატვირთი ფაილია და გამართვისას მისი ამოსავალი კოდის ნახვა შეუძლებელია. ამიტომ, მიმართავენ თექვსმეტობითი(ორობითი) კოდის ასემბლერის კოდში გადაყვანას-**Disassemble Binary Code** (ველი **Advanced Properties**).

ჩვენ გამოვიყენებთ **.cof** გაფართოებას, რომელიც რეკომენდებულია დამწყებთათვის და გამოიმუშავებს **CVAVR** კომპილატორი.

ცხადია, სიმულაციის წინ მკ-ში უნდა განთავსდეს პროგრამის ფაილი, რაც ხორციელდება წესით: შესვლა ძირითად მენიუში- **Source** და ოპცია **Add/Remove Source Files...** გამოძახება, რომელიც გახსნის დიალოგურ ფანჯარას-სურ. 4.3. მოკლედ ჩამოვთვალოთ მოქმედებათა მიმდევრობა: მკ-ს შერჩევა, თუ მკ რამდენიმეა - ის რომლის პროგრამა მოწმდება; შემდეგ კომპილატორის შერჩევა-**AVR-მკ-თვის AVRASM2**; როცა ფაილი ახალია ვაჭერთ ღილაკს **New**, ზედა გააქტიურებულ ველში შეგვაქვს ფაილის მთლიანი სახელი გზით; ვაჭერთ **OK**.

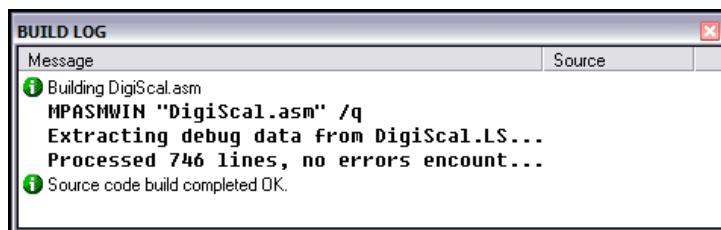


სურ. 4.3.

სიმულაციის გაშვების წინ მიზანშეწონილია პროგრამის კომპილაცია პრეტეუსში ოპციით **Source => Build All...** . კომპილაციის შემდეგ გამოვა ფანჯარა **BUILD.LOG** შედეგებით. თუ შიგ წითელი სტრიქონები არ არის, შედეგი სწორია, თუ არა - საძებნია შეცდომები პროგრამაში.

4.2. პროგრამის გამართვის ბიჯური რეჟიმი

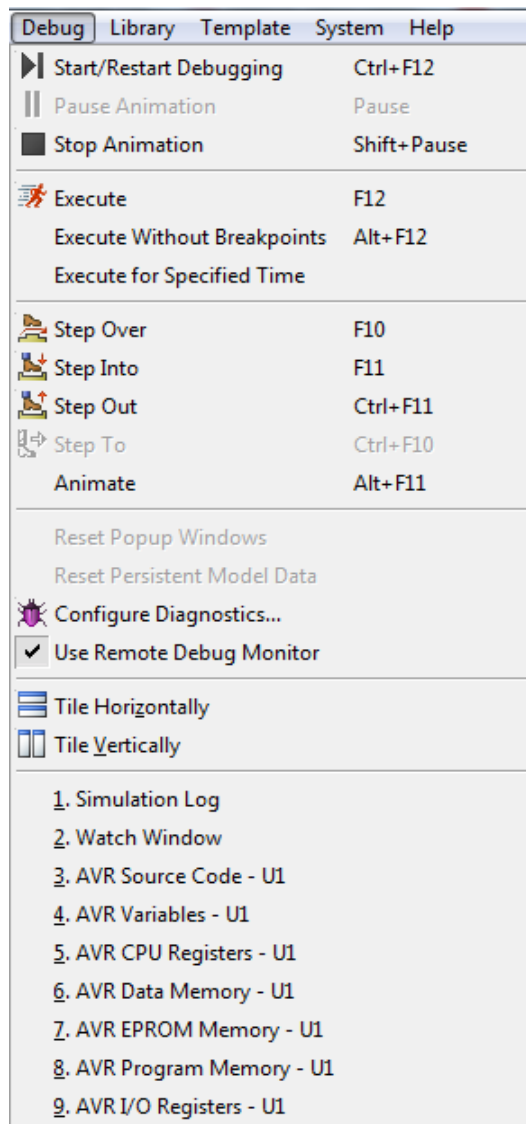
წარმატებული **Build All...**-ის შემდეგ ფანჯარა **BUILD.LOG**-ს აქვს სახე:



სურ. 4.4.

პროგრამების გამართვა, როგორც წესი, ხორციელდება ნაბიჯურ(ბიჯურ) რეჟიმში, რომელიც გაიშვება **Debug** მენიუს ფანჯრიდან - **Start/Restart Debugging** ოპციით, ან **Step** ან

Pause ღილაკებით. ამ დროს გააქტიურდება, განათდება ღილაკები **Play** და **Pause. Debug** მენიუზე მარცხენა წკაპი გამოიძახებს ფანჯარას ოპციების მენიუთი-სურ. 4.5.



სურ. 4.5.

შევნიშნავთ, რომ პროტეუსის ვერსიაზე დამოკიდებულებით ოპციების შემადგენლობა და თავისებურებები შეიძლება იყოს უმნიშვნელოდ განსხვავებული.

აქ მსგავსი ბრძანებები დაჯგუფებულია სექციებში. ზედა სექციაში დუბლირებულია სიმულაციის ღილაკები, ამიტომ მასზე აქ არ შევჩერდებით.

მეორე სექციაში ოპცია მორბენელი ფიგურის გამოსახულებით-**Execute**, უშვებს სიმულაციას უწყვეტ რეჟიმში(**Run simulation**).

შემდეგი ოპცია-**Execute Without Breakpoints**, ასრულებს სიმულაციას გაჩერების წერტილების გაუთვალისწინებლად. ოპცია **Execute For Specified Time** იძახებს ფანჯარას სადაც მოიცემა სიმულაციის შესრულების დრო-ჩაურევლად 1 წამია.

მესამე სექციაში მოთავსებულია ბიჯურ რეჟიმში სამუშაო ინსტრუმენტები, რომლებიც ჩანან პროგრამის შესრულების სათვალთვალ ფანჯარაში(დაწვრილებით ქვევით):

Step Over (Step Over Source Line) - ბრძანებებს ასრულებს ბიჯურ რეჟიმში, მაგრამ ფუნქციების შიგნით ბრძანებებს ანუ ფუნქციებს ასრულებს ავტომატურ რეჟიმში;

Step Into(Step Into Source Line) - ყველა ბრძანებებს ასრულებს ბიჯურ რეჟიმში;

Step Out(Step Out from Source Line) - ფუნქციის შიგნით წყვეტს ბიჯურ რეჟიმს, დარჩენილ ბრძანებებს ასრულებს ავტომატურ რეჟიმში და გამოდის ფუნქციიდან;

Step to(Run To Source Line არააქტიური პიქტოგრამა(აქტიურდება სტრიქონის მონიშვნისას) იწვევს ბრძანებების ავტომატურ შესრულებას ბრძანების კურსორიდან (სამკუთხედი სტრიქონის თავში), სტრიქონის მონიშვნის კურსორამდე(ფერადი ზოლი სტრიქონზე);

Animate - ბრძანებები სრულდება ავტომატურ რეჟიმში, მაგრამ ნელა, ადამიანისათვის აღქმად ტემპში.

მეოთხე სექციაში მოცემულია ორი არააქტიური ოპცია(აქტიურებია სქემის გაშვებამდე): **Reset Popup Windows** და **Reset Persistent Model Data**. მათ მოყავთ საწყის მდგომარეობაში შესაბამისად, დიალოგური ფანჯრები და გარკვეული მონაცემები მუდმივ მეხსიერებებში. ფანჯრების ჩამონათვალი მოცემულია მენიუს ბოლოში. მარცხენა წკაპი რომელიმე სტრიქონზე იწვევს სტრიქონის მონიშვნას და ფანჯრის გამოტანს ეკრანზე.

შევნიშნავთ, რომ ფანჯრების ჩამონათვალში მოცემული მკ-ს სახელი ავტომატურად იცვლება სქემაში გამოყენებული მკ-ს სახელით.

მათ ქვევით მოცემულია ოპცია ხოჭოს გამოსახულებით-**Configure Diagnostics**, რომელიც ალებს ფანჯარას საიდანაც შესაძლებელია დიაგნოსტიკის რეჟიმების და შეტყობინებების ფორმების დაყენება.

Use Remote Debug Monitor ოპციას არ განვიხილავთ, როგორც ჯერჯერობით არააქტუალუს.

მეუხთე სექციის ოპციები **Tile Horizontally** და **Tile Vertically** მიუთითებენ დიალოგური ფანჯრების განლაგებაზე-ჰორიზონტალური თუ ვერტიკალური, რომლებიც გამოიძახება ქვედა, მეექვსე სექციის ოპციებით. მათ შესახებ-ქვევით.

4.3. ბიჯური რეჟიმის გამოყენების თავისებურებები

რეჟმი რეალიზდება სათვალთვალო ფანჯრის (**CPU Source Code**) გამოყენებით, სადაც ჩანს ამოსავალი პროგრამა ასემბლერის ან C ენაზე და აქვს სახე-სურ. 4.6:

ეკრანის ზედა მარჯვენა კუთხეში განლაგებულია პიქტოგრამები, რომელთა ფუნქცია უკვე განმარტებულია. მათი მომდევნო ორი წერტილი გამოიყენება გაჩერების წერტილის დასაყენებლად და მოსახსნელად.

ტექსტი შედგება სვეტებისაგან: ბრძანების მისამართი თექვსმეტობით კოდში (ბაცი მწვანე ფერი), ბრძანებები ტექსტურ ფორმატში(ლურჯი ან შავი ფერი), კომენტარები(მწვანე ფერი). სტრიქონებს, რომლებიც მეხსიერებაში ადგილს არ იკავებენ მისამართები არ გააჩნიათ, რადგანაც ისინი შესრულებად ფაილში არ ხვდებიან. რომელიმე სტრიქონის თავში წითელი სამკუთხედი- მიმდინარე შესასრულებელი ბრძანებაა; წითლად შეღებილი წრე სტრიქონის წინ-გაჩერების წერტილია; წითელი

წერილი- დროებით გაუქმებული გაჩერების წერტილია; ფერადი ზოლი სტრიქონზე- სტრიქონის მონიშვნა- სრულდება მარცხენა წკაპით სტრიქონზე. როგორც წესი,

```

AVR Source Code - U1
..\\..\\..\\Program Files\\cvavr2\\bin\\Mag_5.c

---- // Analog Comparator Input Capture by Timer/Counter 1: off
009E ACSR=0x80;
00A2 DIDR=0x00;
----
---- while (1)
---- {
00A6     {if (PIND.0 ==1)
----         {rab = 0b10000000;
00AA             while (rab != 0)
00AE                 {
00B2                     PORTB = rab ^ 0xFF;
00B8                     rab=rab >> 1 ;
00BA                         delay_ms(400);
00C0                 }
00C2             }
----         else
00C4             {rab = 0b00000001;
00C8                 while (rab != 0)
00CC                     {
00D2                         PORTB = rab ^ 0xFF;
00D4                         rab=rab << 1 ;
00D6                             delay_ms(400);
00D8                     }
00DA                 }
----         }
----     }
---- }

```

სურ. 4.6.

ბრძანებების შესრულების დროს ზოლი ემთხვევა და თან მიჰყვება წითელ სამკუთხედს.

CPU Source Code ფანჯარაზე მარჯვენა წკაპი იძახებს დიალოგურ ფანჯარას, რომელიც მოცემულია-სურ. 4.7-ზე.

ის შედგება ხუთი სექციისგან:

პირველი სექციაში ოპციას **Dissassembly** გადაყავს პროგრამა მაღალი დონის ენიდან- მაგ. სი, ასემბლერის ფორმაში. ეს საჭირო ხდება პროგრამის დაწვრილებითი განხილვის დროს.

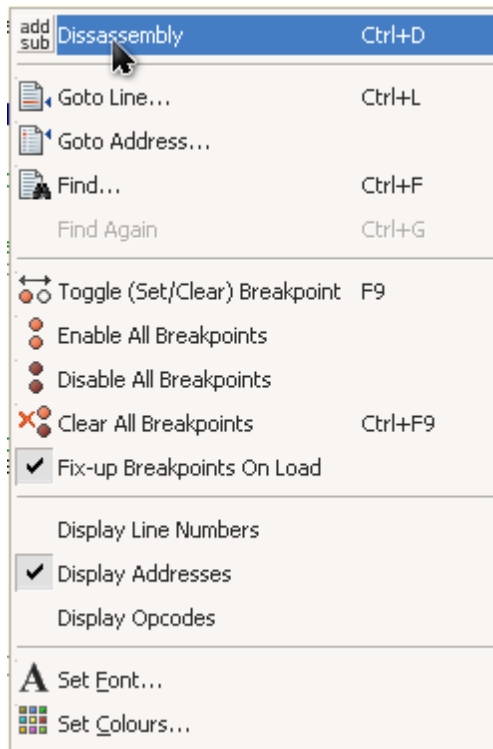
მეორე სექციაში ოპციებს **Goto Line...** და **Goto Addresss...** გადააქვს მონიშვნის ზოლი მითითებულ სტრიქონზე ან მისამართზე,თუ ისინი ნაჩვენებია. შემდეგი ორი ოპცია **Find...** (ძებნა) и **Find Again...** (განმეორებითი ძებნა) მითითებული მონაცემების მიხედვით, ცნობილი წესით.

მესამე სექციის ოპციებს იყენებენ გაჩერების წერტილების დასაყენებლად და მოსახსნელად. პირველი-**Toogle ...** უკვე განმარტებული იყო. შემდეგი სამი ოპცია-**Enable** (აქტივირება), **Disable**

(დეაქტივირება) და **Clear**(გაწმენდა) ერთდროულად მოქმედებს გაჩერების ყველა წერტილზე. ოპცია **Fix-Up Breakpoints On Load** ააქტიურებს ყველა ადრე დაყენებულ წერტილს სიმულაციის ახალი გაშვების დროს.

მეოთხე სექციის ოპციებით **CPU Source Code** ფანჯარაში ხორციელდება: **Display Line Numbers**-სტრიქონის ათობითი ნომრის და კომენტარების ჩვენება, **Display Addresses** - თექვსმეტობითი მისამართის ჩვენება ასემბლერის ბრძანებების დონეზე, **Display Opcodes**-ასემბლერის ბრძანებების თექვსმეტობითი კოდის ჩვენება მისამართის გვერდზე.

მეხუთე სექციის ოპციებით-**Set Font...** და **Set Colours...** ხორციელდება თქვენთვის სასურველი შრიფტებისა და ფერების დაყენება.

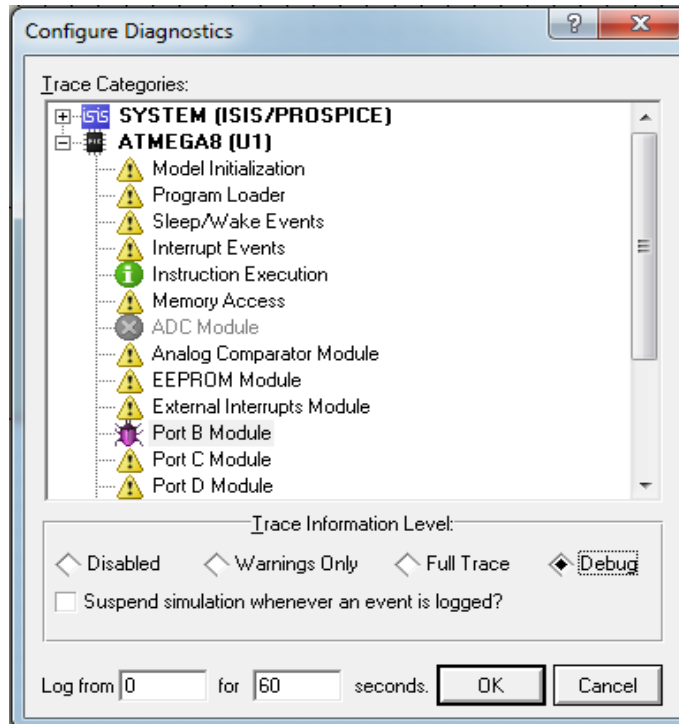


სურ. 4.7.

4.4. დიაგნოსტიკურ შეტყობინებათა ფორმირება Debug-ში

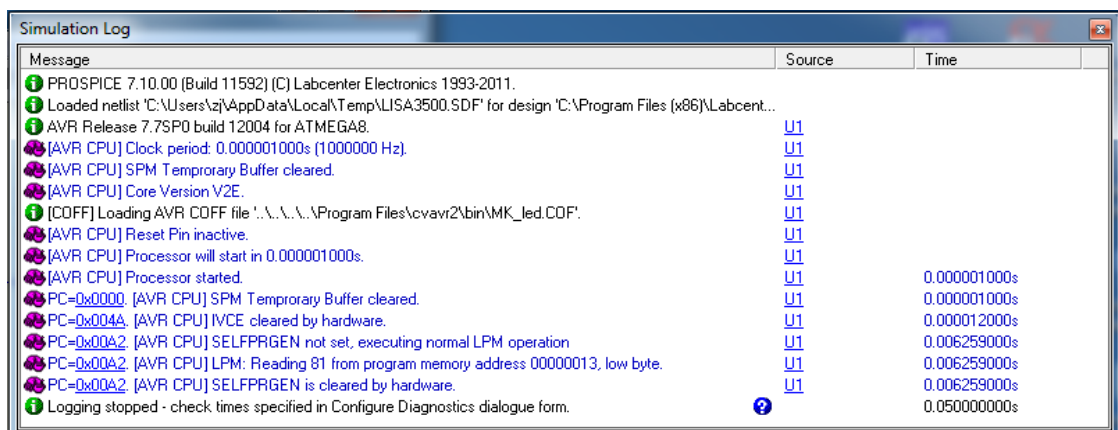
Configure Diagnostics ფანჯარა(ხოჭოს ლოგო) ხელმისაწვდომია როგორც გაშვებული, ისე გაჩერებული სიმულაციის დროს. აქ განისაზღვრება **Simulation Log** ფანჯრაში შეტყობინებათა ფორმა და შინაარსი-სურ. 4.8.

სურათზე დამალულია სიმულატორ **PROSPICE**-ის შეტყობინებები და გაშლილია მკ-ს შეტყობინებები: შესაბამისად + და - სახელების წინ. ყვითელი სამკუთხედები ძახილის ნიშნით მიუთითებს, რომ ამ კომპონენტებისთვის დასაშვებია მხოლოდ გაფრთხილებების გამოტანა-**Warnings Only**, რაც აისახება გადამრთველზე **Trace Information Level** ველში. იმისათვის, რომ რომელიმე კომპონენტს მიენიჭოს ტრასირების რომელიმე რეჟიმი სჭიროა, მონიშნოს კომპონენტი მარცხენა წკაპით და ჩაირთოს სასურველი გადამრთველი. ყვითელი სამკუთხედები შეიცვლებიან შესაბამისი ლოგოთი. სურათზე გამორთულია(**Disabled**) შეტყობინებები კომპონენტიდან, რომლის წინ ზის რუხი წრე **X**-ით. ცხადია უფრო ინფორმაციულია რეჟიმები **Full Trace** (სრული ტრასირება), ის მონიშნება მწვანე ფერის წრეში ჩასმული ასო **i**, და **Debug**(გამართვა)-ხოჭოს ლოგო.



სურ. 4.8.

ქვევით სურათზე(სურ. 4.9), მოცემულია მაგალითი, სადაც სრულ ტრასირებას(Full Trace) ექვემდებარება მკ-ს ბრძანებების შესრულება(Instruction Execution). ტრასირება გაკეთებულია ასემბლერის ბრძანებების მიხედვით და ნაჩვენებია ყველა ბრძანების შესრულების დრო. ამდენად, შესაძლებელია პროგრამის ნებისმიერი მონაკვეთის



სურ. 4.9.

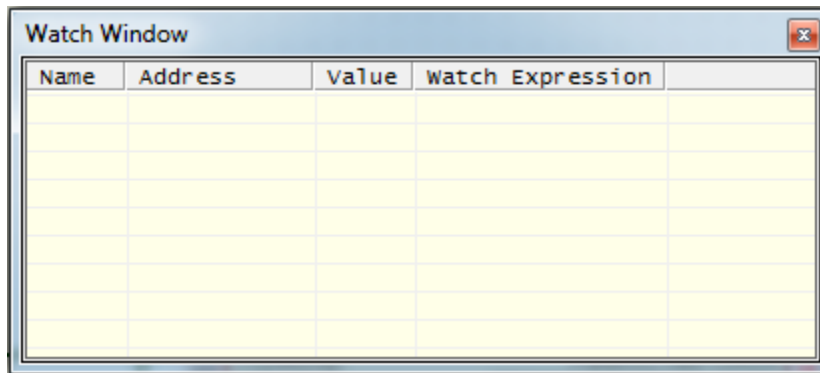
ხანგრძლივობის გამოთვლა. პაუზის რეჟიმში თუ დავაწკაპებთ პროგრამული მთვლელის სიტყვაზე(PC=0x0080), გამოჩნდება ამოსავალი კოდის ფანჯარა(CPU Source Code) და მონიშნული იქნება შესაბამისი ბრძანება.

Configure Diagnostics ფანჯარას გააჩნია ველები სადაც შესაძლებელია დროის ხანგრძლივობის მოცემა, რომლი განმავლობაში ხდება მონაცემების აღება- **Log from ... for... seconds**. საწყისად ჩადებულია ერთი წუთი.

Debug -ის მეექვსე სექციის ოპციების წინ დასმული ალაში იწვევს შესაბამისი

ფანჯრის გამოჩენას ეკრანზე. ოპციების რაოდენობა და შემადგენლობა დამოკიდებულია პროგრამირებადი კომპონენტების ტიპზე. ოპციებიდან ორი უკვე განხილულია: **CPU Source Code** და **Simulation Log**. ოპციების დანიშნულება გასაგებია მათი დასახელებიდან, მაგრამ უფრო დაწვრილებითი განხილვა საჭიროა სასარგებლო ოპციისთვის **Watch Window**. შევნიშნავთ, რომ **Simulation Log** და **Watch Window** -ის ფანჯრები ხელმისაწვდომია რეალურ დროში სიმულაციის დროსაც, დანარჩენი კი, მხოლოდ ბიჯურ რეჟიმში. ისინი ატარებენ ძირითადად ინფორმაციულ ხასიათს.

განვიხილოთ **Watch Window**(თვალყურების ფანჯრის) გამოყენების ხერხები. გადავდივართ პაუზის რეჟიმში, შევდივართ **Debug**-ის მენიუში, ვსვამთ ალამს **Watch Window** -ის წინ და გამოდის ფანჯარა-სურ. 4.10.

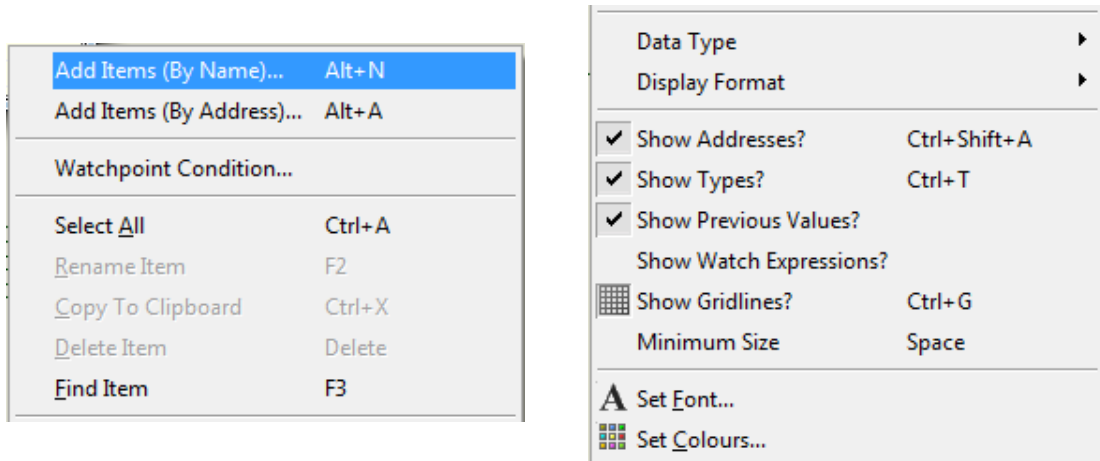


სურ. 4.10.

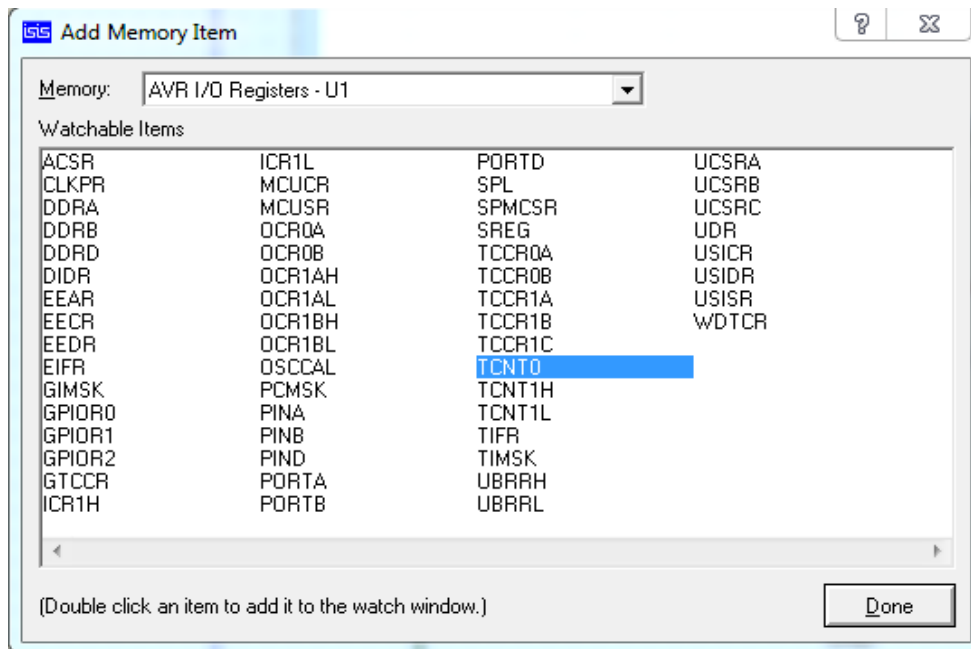
ფანჯარაზე მარჯვენა წკაპით ვიძახებთ კონტექსტურ მენიუს- სურ. 4.11. (გაყოფილია და წარმოგენილია ორ ნაწილად).

პირველი ორი სტრიქონით შესაძლებელია ობიექტების შეტანა ფანჯარაში სახელის (**By Name**) ან მესხიერების უჯრედის მისამართის (**By Address**) მიხედვით. **By Name** -ს დროს გამოდის ფანჯარა რეგისტრების ჩამონათვალით-სურ. 4.12.

ორჯერ მარცხენა წკაპით სახელზე **PORTB**, ის აღმოჩნდება ფანჯარაში. შეგვიძლია შევასრულოთ იგივე მოქმედებები სახელისთვის **TCNO**. თუ სახელის წინ დაჯდა პლიუს ნიშანი-ეს ნიშნავს, რომ ბიტებს გააჩნიათ სხვადასხვა დანიშნულება. მარცხენა წკაპი პლიუსზე გახსნის შემადგენლობას.

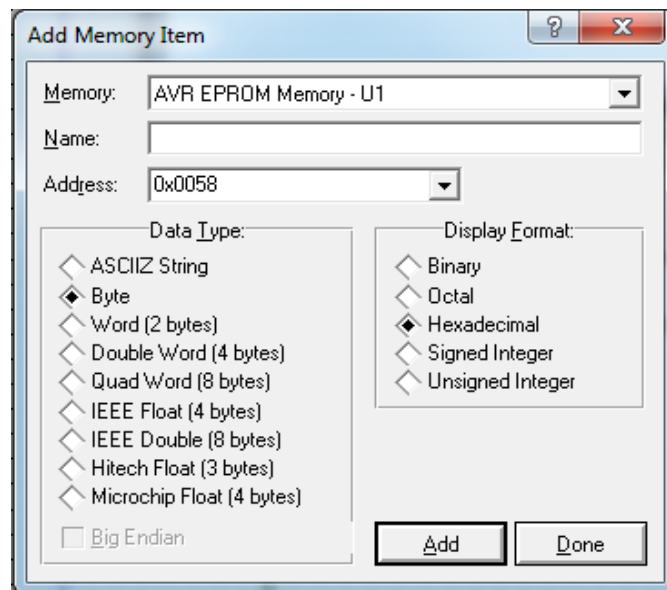


სურ. 4.11.



სურ. 4.12.

By Address -ით გამოძახების შემთხვევაში გამოვა ფანჯარა სახით-სურ. 4.13:



სურ. 4.13.

აქ, სახელი და მისამართი შეიტანება კლავიატურიდან **ADD** ღილაკის გამოყენებით. თუ სახელს არ მიუთითებთ, სისტემა თვალსაზრისით ფანჯარაში სახელის ადგილზე ჩასვამს ისევ მისამართს. ამჟამად შეტანილია მისამართი 0x0058. ფანჯარაში შესაძლებელია მონაცემთა ტიპის (**Data Type**) და ფორმატის (**Display Format**) მითითება. ორივე ეს ოპცია მოცემულია "დებაგის" დიალოგურ ფანჯარაშიც-სურ.46, მაგრამ გააქტიურებისთვის საჭიროა დავდგეთ **Watch Window** - ის რომელიმე ცვლად პუნქტზე. **Watch Window** -ის საბოლოო სახე მოცემულია სურ. 4.14-ზე.

Name	Address	Value	Watch Expression
PORTB	0x0038	0xF7	
PORTB0	0x0018	1	
PORTB1	0x0018	1	
PORTB2	0x0018	1	
PORTB3	0x0018	0	
PORTB4	0x0018	1	
PORTB5	0x0018	1	
PORTB6	0x0018	1	
PORTB7	0x0018	1	
TCNT0	0x0052	0x00	
0x0058	0x0058	0xFF	

სურ. 4.14.

საყურადღებოა, რომ თუ რაიმე თანრიგები გაერთიანებულია ერთ კოდურ მნიშვნელობაში, ისინი მოიცემა ერთ სტრიქონში.

მას შემდეგ, რაც **Watch Window**-ში ობიექტები შეტანილია, მის მენიუში ოპციების გასააქტიურებლად საჭიროა **Show...** აღმების დაყენება.

Watch Window - ის მენიუში განსაკუთრებით საყურადღებოა ოპცია **Watchpoint Condition...**(წყვეტის პირობები შეტანილი ობიექტებისათვის), რომელიც იძახებს სურ. 4.15-ზე ნაჩვენებ ფანჯარას.

სურ. 4.15.

ის საშუალებას იძლევა განვახორციელოთ წყვეტა, სიმულაციის შეჩერება, ნებისმიერი რეგისტრის შიგთავსის ცვლილების მიხედვით, თუნდაც ერთი თანრიგის. ეს გვანთავისუფლებს აპარატურული "ბრეკპოინტების" გამოყენებისგან. სამპოზიციანი გადამრთველი აყენებს წყვეტის გლობალურ პირობას(**Global Break Condition**): **Turn Off...** - თიშავს წყვეტით გაჩერებას; **Suspend the simulation if ANY expression is true** - შეჩერდეს სიმულაცია თუ რომელიმე გამოსახულება ჭეშმარიტია; **... only when ALL... true** - შეჩერდეს სიმულაცია თუ ყველა გამოსახულება ჭეშმარიტია. **Watch Window**-ის ყველა

პოზიციისთვის (სტრიქონისათვის) შეიძლება შეირჩეს თავისი პირობა. მესამე გადამრთველის პირობა იმუშავებს მაშინ, როდესაც ყველა პირობა შესრულდება. მაგალითში, სახელის პოზიციაში-Item, აღებულია AVR მკ-ს რეგისტრი PORTB; ნიღბის(Mask) ველებში, ლოგიკური ოპერაცია AND და ნიღბის მნიშვნელობა 0x0F; პირობის ველში - Condition , Equals(ტოლობა), ხოლო მნიშვნელობის (Value) ველში-0x02.

ახლა კი, Watch Window - ის მენიუს დანარჩენ პუნქტებზე: ოპციების, დაწყებული Select All(ავირჩიოთ ყველა) და დამთავრებული Find Item(იპოვე სახელი) დანიშნულება ცნობილია გავრცელებული სისტემებიდან და აქ არ განვმარტავთ.

ოპციები Data Type და Display Format უკვე განმარტებულია.

მომდევნო რამდენიმე ოპცია განკუთვნილია ფანჯარაში დამატებით სვეტების შესაქმნელად და შესაბამისი მონაცემების საჩვენებლად(Show...), სტანდარტული მინიმუმი ორია.


ოპცია Show Gridlines –რთავს და გამორთავს ბადეს ეკრანზე. Minimum Size - ამცირებს ეკრანს მასში არსებული მონაცემების დონემდე.

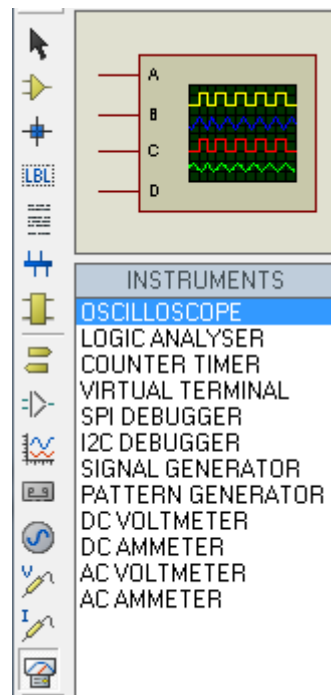
დანარჩენი ორი ოპციის დანიშნულება განმარტებულია ადრე.

თავი 5. ვირტუალური გამოზომი ხელსაწყოების გამოყენება

შესავალი

როგორც აღნიშნული იყო, რეჟიმი **Virtual Instruments Mode** (ვირტუალური ინსტრუმენტების რეჟიმი) საშუალებას იძლევა ავირჩიოთ ვირტუალური ხელსაწყოები სიიდან: ვოლტმეტრები, ამპერმეტრები, მრავალარხიანი ოსცილოგრაფი, სიგნალების ანალიზატორი, მთვლელები, სიხშირის მზომი, სიგნალ-გენერატორი და სხვა. უნდა გავითვალისწინოთ, რომ მათ უმრავლესობას გააჩნია ერთპოლუსიანი ჩართვა-გაზომვა წარმოებს მხოლოდ მიწის მიმართ. ყველა ამ ხელსაწყოს გააჩნია საკუთარი სქემური აღნიშვნა (ლოგო), შესასვლელ/გამოსასვლელი კონტაქტების მითითებით და მართვის და ინდიკაციის საშუალებებით.

ხელსაწყოების სიის გამოძახება ხდება პიქტოგრამით , რეჟიმების პანელიდან (მარცხენა სვეტი) და აქვს სახე-სურ. 5.1:



სურ. 5.1.

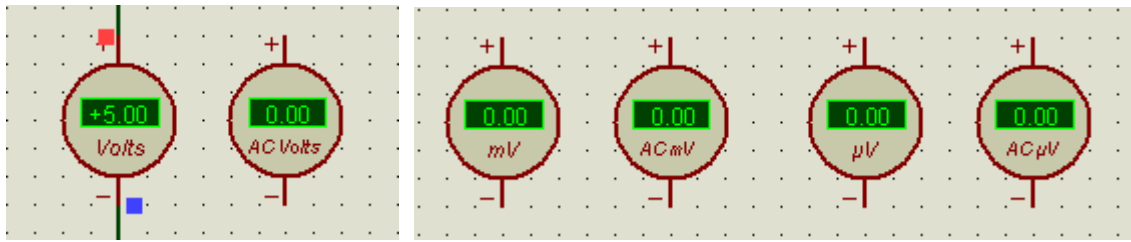
სიაზე მოინიშნება ხელსაწყო სახელი, ხელსაწყო გამოჩნდება თვალიერების ფანჯარაში და შესაძლებელია მისი ორიენტაციის ცვლა ცნობილი ხერხებით, ხოლო შემდეგ ავსახავთ რედაქტირების ფანჯარაში. რედაქტირების მანიპულაციები ხორციელდება ისევე, როგორც ყველა დანარჩენი ობიექტებისთვის.

განხილვა დავიწყოთ ყველაზე მარტივი და გავრცელებული ხელსაწყოებით - ვოლტმეტრებით და ამპერმეტრებით. ისინი ორი ტიპისაა: მუდმივი-DC (direct current), და ცვლადი-AC (alternating current) ძაბვების და დენების გაზომვისათვის.

5.1. ვოლტმეტრი

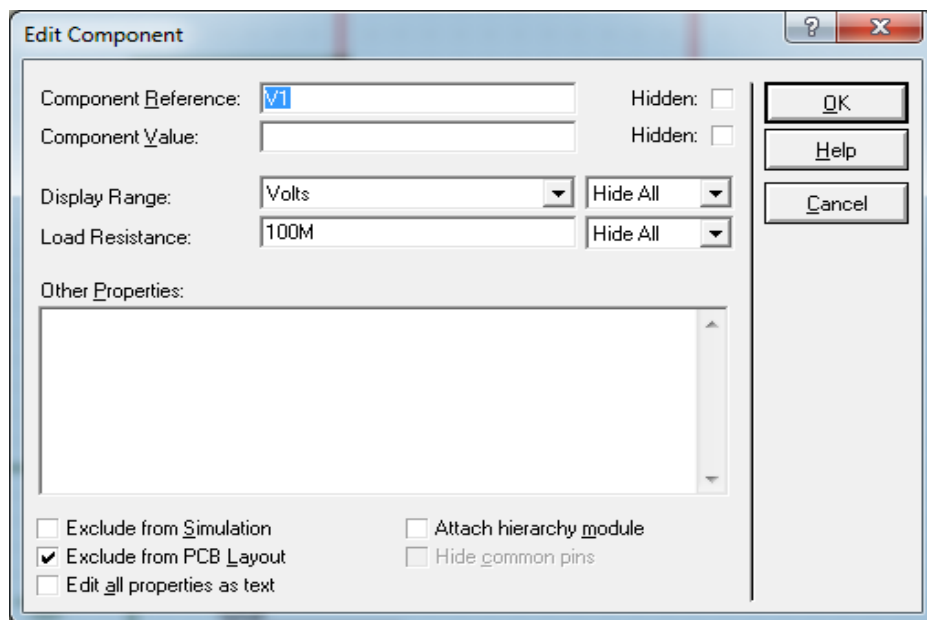
მისი სქემური გამოსახულება ძაბვის სხვადასხვა დიაპაზონისთვის მოცემულია

სურ. 5.2-ზე:



სურ. 5.2.

რედაქტირების ფანჯარაში(სურ. 5.3) შირჩევა ხელსაწყოს სახელი, გაზომვის ერთეული **Display Ranges** ველის ჩამომლადი სიიდან-**Volts**, Millivolts(**mV**), Microvolts(**uV**) და ვოლტმეტრის შიგა წინალობა-აქ 100M და სხვა, თუ ისინი საჭიროა.



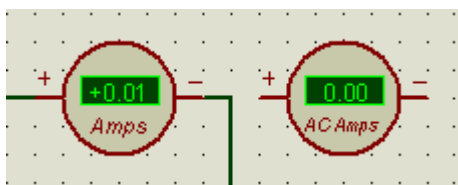
სურ. 5.3.

სურ. 53-ზე პირველი ვოლტმეტრი(DC) გადაღებულია მუშა მდგომარეობაში - ჩართულია კვების ძაბვის წრედში, ხოლო დანარჩენი განთავსებულია ჩართულ სქემაზე, მაგრამ წრედში არ არის ჩართული. მიუხედავად ამისა მათი ინდიკატორი გააქტიურებულია. რედაქტირებით, ყოველ ვოლტმეტრზე მითითებულია გაზომვის ერთეული: ვოლტი, მილივოლტი და მიკროვოლტი. შეგახსენებთ, რომ ვოლტმეტრი სქემაში ირთვება დატვირთვის პარალელურად.

5.2. ამპერმეტრი

მისი სქემური გამოსახულება მოცემულია სურ. 5.4-ზე.

როგორც ვოლტმეტრების შემთხვევაში, აქაც ამპერმეტრი შეირჩევა წრედის ტიპის მიხედვით-მუდმივი თუ ცვლადი. სურათზე პირველი ამპერმეტრი(DC) ნაჩვენებია მუშა მდგომარეობაში, ხოლო მეორე- არამუშა მდგომარეობაში. რედაქტირებით, აქაც შეიძლება

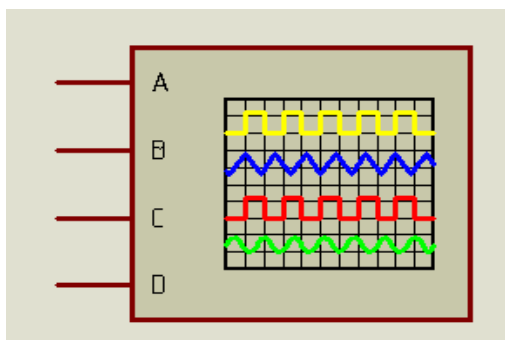


სურ. 5.4.

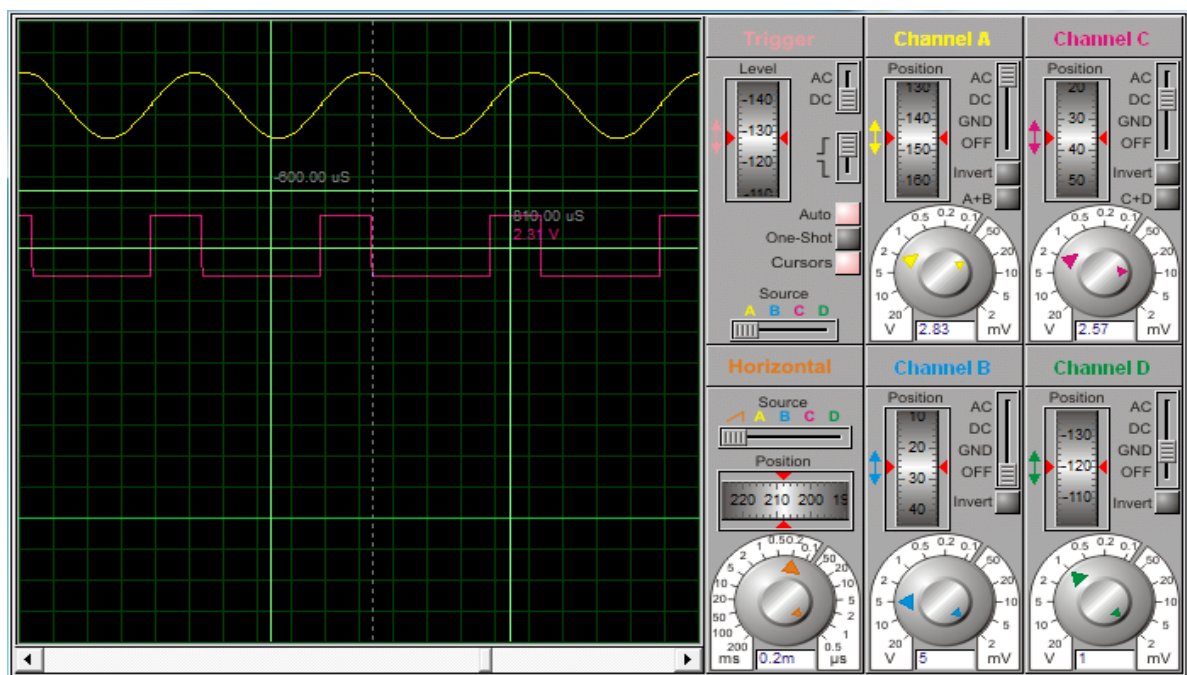
გაზომვის ერთეულის მითითება: ამპერი, მილიამპერი, მიკროამპერი. ისევ შეგახსენებთ, რომ ამპერმეტრი სქემაში ირთვება დატვირთვასთან მიმდევრობით.

5.3. ციფრული ოსცილოგრაფი (ოსცილოსკოპი)

მისი სქემური გამოსახულება მოცემულია სურ. 5.5. ა-ბე. მას გააჩნია ოთხი საინფორმაციო შესასვლელი(ოთხარხიანია)-A, B, C, D. ისინი, როგორც რეალურ სქემებში, უერთდებიან გასასინჯ წერტილებს.



ა.



ბ.

სურ. 5.5.

მართვის და ინდიკაციის პანელის მეშვეობით(სურ. 5.5. ბ) შესაძლებელია შემდეგი მანიპულაციების ჩატარება:

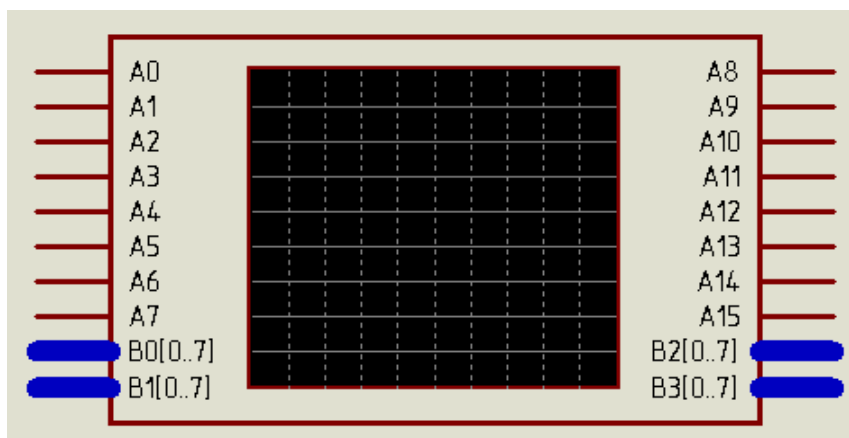
ყოველ არხს გააჩნია გადამრთველები, რომლითაც დაყენდება არხში სიგნალის მიწოდების რეჟიმი: **AC**-ატარებს ცვლად მდგენელს, არ ატარებს მუდმივ მდგენელს, **DC**-ატარებს ყველა მდგენელს, **GND**-მიეწოდება ნოლი, **OFF** -არხი გათიშულია. **Invert** დილაკით ეწოდება სიგნალი ინვერტირებული ფორმით. **A** და **C** არხს აქვს დილაკი **A+B** და **C+D** ჯამის ასახვისათვის. ვერტიკალური გორგოლაჭებიანი სკალით შესაძლებელია ჰორიზონტალური (**X**) ღერძის გადაადგილება ვერტიკალურ კოორდინატაზე შესაბამის გრაფიკთან ერთად. წრიული სქალებით რეგულირდება გამოსახულების ამპლიტუდა არხში -გარე სკალით მკვეთრი რეგულირება, შიგა სკალით რბილი რეგულირება.

Trigger-ის ველში მოთავსებულია გადამრთველები: **AC, DC**- გამშვები სიგნალის ტიპის განსაზღვრისათვის, მის ქვევით ამ სიგნალის ფრონტის განსაზღვრისათვის, **Source**-გამშვები სიგნალის წყაროს განსაზღვრისათვის; დილაკები: **Auto**-სიგნალების ავტომატური მიღება; **One-Shot** -ერთჯერადი მიღება; **Cursors-X, Y**კურსორების წყვილის ჩვენება/არჩვენება ეკრანზე. ჩვენებისას შესაძლებელია წყვილის მოძრაობა, დაფიქსირება მარცხენა წკაპით საჭირო ადგილებზე და ანათვლების აღება ძირითად **X, Y** ღერძების მიმართ. ვერტიკალური ბორბლი სქალით- **Level**-ის დანიშნულებაა **X** საკოორდინატო ღერძის გადატანა ვერტიკალურ არეში.

Horizontal ველში მოთავსებულია სიგნალების გაშლის სიხშირის მარეგულირებელი სქალა წრიული გადამრთველით-გარე სკალით მკვეთრი რეგულირება, შიგა სკალით რბილი რეგულირება. ჰორიზონტალური სქალიანი ბორბლის-**Position**, დანიშნულებაა **Y** საკოორდინატო ღერძის გადატანა ჰორიზონტალურ არეში. **Source** გადამრთველის დანიშნულებაა გამშლელი სიგნალის წყაროს დაყენება. როგორც წესი, ძირითადი რეჟიმისთვის აყენია ხერხისებური ძაბვის წყარო.

5.4. ლოგიკური ანალიზატორი

მისი სქემური სახეა:

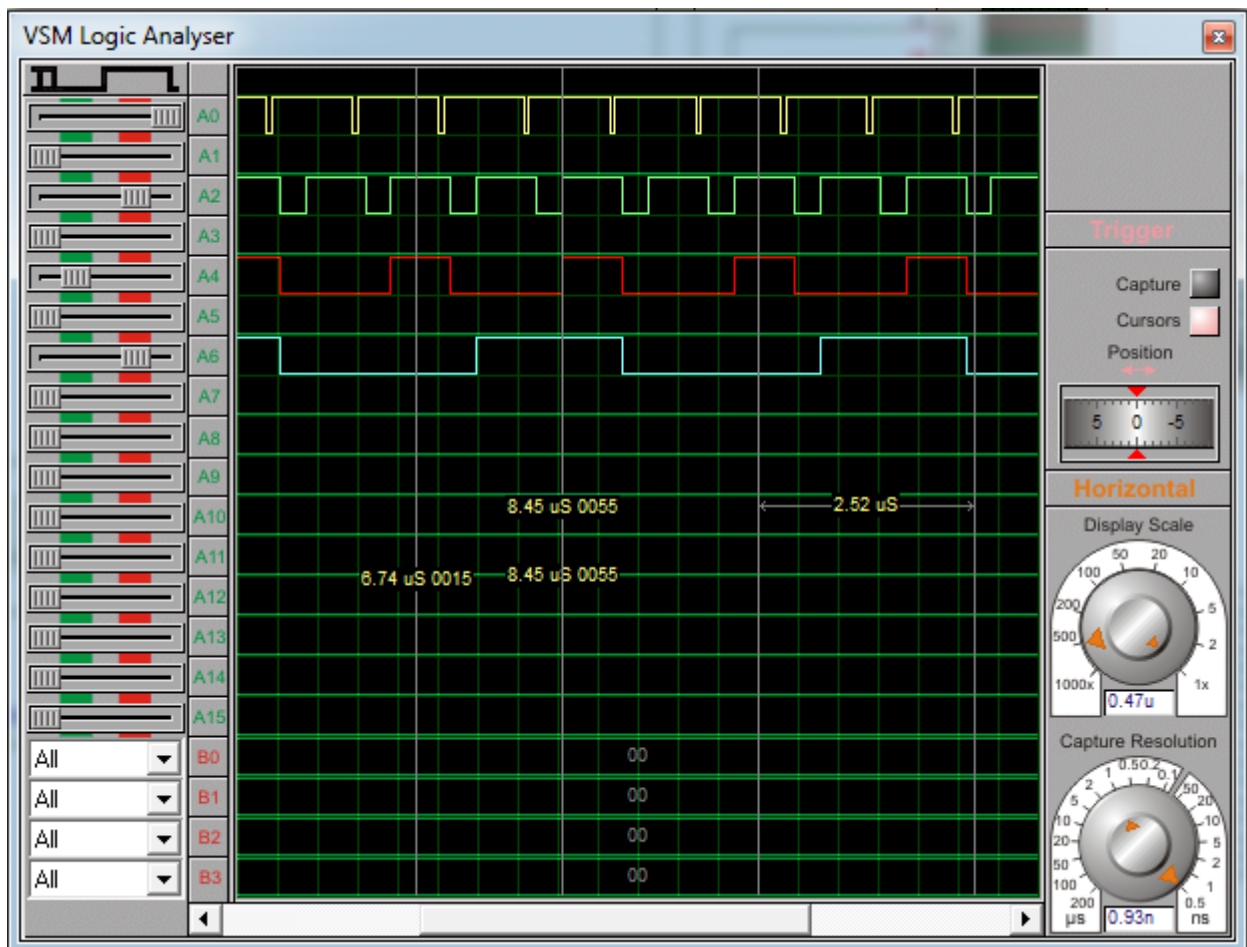


სურ. 5.6.

A0...A15 შესასვლელები განკუთვნილია ანალიზატორის გაშვების(მონაცემთა შეგროვების დაწყების) მომენტის განსაზღვრისათვის **AND** ლოგიკური ფუნქციის გამოყენებით. **B0...B3** სალტეები საინფორმაციო არხებია-უერთდებიან საკონტროლო წერტილებს.

მონაცემთა ადების ინტერვალი იცვლება დიაპაზონში 0,5 μ -დან 200 μ -დე. ადების დრო 10n-4s დიაპაზონშია. შენახვის ბუფერის ზომაა 40000 ბაიტი.

ანიმაციის(სიმულაციის) გაშვების შემდეგ Debug-ის მენიუში უნდა ჩაირთოს ალამი VSM Logic Analyser-გამოჩნდება ანალიზატორის მართვის და ინდიკაციის პანელი-სურ. 57.



სურ. 5.7.

ფანჯრის მარცხენა მხარეს განთავსებულია გაშვების სიტყვის 16 არხის გადამრთველები, რომელთაც გააჩნიათ ხუთი პოზიცია(ლოგო სვეტის თავზე): გამორთული, ნულის დონე, დადებითი ფრონტი, ერთის დონე, უარყოფითი ფრონტი.

ქვევით განლაგებულია სალტეების არხები.

მარჯვენა მხარეს **Trigger** ველში განლაგებულია დილაკი **Capture**, რომელიც რთავს მონაცემთა მიღება/ინდიკაციის რეჟიმს: წითელი ფერი-გაშვების სიტყვის ლოდინია, მწვანე ფერი-მონაცემთა ბუფერში მიღების დროა, გათიშვა-ინდიკაციაზე გადასვლა-

დიაგრამები გამოჩნდება ეკრანზე, რომელიც დარეგულირდება **Display Scale** გადამრთველით.

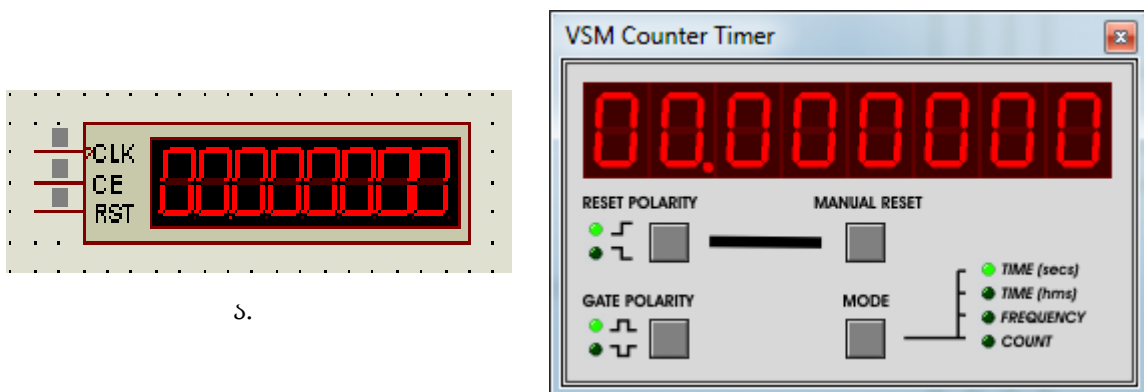
ლილაკი **Cursors** ნებას რთავს კურსორების გამოყენებას: თავის კურსორის შეტანა ეკრანზე წარმოქმნის ვერტიკალურ კურსორს; ყოველი მარცხენა წკაპი ეკრანზე აფიქსირებს მიმდინარე კურსორს და წარმოქმნის ახალს; თავის კურსორის გადატანა დაჭერილი მარცხენა კლავიშის პირობებში იწვევს ბოლო ორ კურსორს შორის დროის მონაკვეთის რიცხვით ფორმაში ჩვენებას.

Position მბრუნავი სქალით შეიძლება ეკრანის გადააგილება მეხსიერების ბუფერში შენახულ მასაზე, რამდენადაც ეკრანი იტევს გაცილებით ნაკლებ ინფორმაციას ვიდრე ბუფერშია. გადადგილების არეალია ბუფერის -50%-დან +50%-მდე.

სქალა **Capture Resolution** უზრუნველყოფს მონაცემთა აღების სიხშირის შერჩევას.

5.5. ტაიმერ-მთვლელი

ის წარმოადგენს მრავალფუნქციურ ხელსაწყოს და განკუთვნილია დროითი და მასთან დაკავშირებული პარამეტრების გაზომვისათვის. სურ. 5.8-ზე მოცემულია როგორც მისი სქემური გამოსახულება ისე მისი მართვის და ინდიკაციის პულტი.



სურ. 5.8.

მას გააჩნია შემდეგი რეჟიმები:

TIME(secs)-ტაიმერის რეჟიმი(წამები), გარჩევადობა(სიზუსტე) 1 μ S;

TIME(hms)- ტაიმერის რეჟიმი(საათები წუთები, წმები), გარჩევადობა(სიზუსტე) 1mS;

FREQUENCY-სიხშირის გაზომვის რეჟიმი, გარჩევადობა 1Hz;

COUNT-თვლის რეჟიმი, მაქსიმალური რაოდენობა 99 999 999.

ხელსაწყოს აქვს შესასვლელი: CLK-თვლის შესასვლელი, CE-თვლის ნებართვის შესასვლელი, RST-ნულზე დაყენების შესასვლელი.

როგორც ხელსაწყოს დისპლეიდან ჩანს- სურ. 5.8 ბ, ის განკუთვნილია ანათვლებისათვის რვათანრიგა ათობითი რიცხვის ფარგლებში.

პულტის მარცხენა მხარეს განთავსებულია ნულზე დაყენების ლილაკი სიგნალის პოლარობის ჩვენებით-**Reset Polarity**. მის ქვევით განთავსებულია **თვლის შეჩერების**

ლილაკი- Gate Polarity. მარჯვენა მხარეს განლაგებულია ხელით ჩამოგდების ლილაკი-**Manual Reset** და რეჟიმების ამორჩევის ლილაკი-**Mode**.

5.5.1. დროითი ინტერვალების გაზომვა

ხელსაწყოს სქემაზე განთავსების შემდეგ მისი შესასვლელში უნდა მიუერთდეს შესაბამისი წერტილებს. ამასთან, როგორც ყველა დანარჩენის ხელსაწყოები, სქემაზე შეძლება შემოტანილიქნეს რამდენიმე ასეთი ხელსაწყო. ჩვეულებრივი წესით მოხდება მათი რედაქტირება: სახელი, რეჟიმები, პარამეტრები. მართვის პანელის გამოძახება უნდა მოხდეს მენიუდან, რომელიც გაიხსნება ან **Debug**-დან ან მარჯვენა წკაპით ხელსაწყოზე.

სურ. 5.9 ა-ზე მოცემულია დროითი ინტერვალების გაზომვის სკალა **TIME(secs)** რეჟიმისათვის, რომელიც შეიცავს თანრიგებს მთელი ნაწილის ორი ციფრისთვის და წილადი ნაწილის ექვსი ციფრისთვის-გაზომვა წამებშია, ხოლო სიზუსტე მიკროწამებში.



ა.

ბ.

სურ. 5.9.

დროის ინტერვალის სიდიდე განისაზღვრება იმპულსების იმ რაოდენობით, რომელიც დაგროვდება ნებართვის **CE** სიგნალის არსებობის პერიოდში. იმპულსების რაოდენობა გამრავლებული იმპულსებს შორის პერიოდზე იძლევა დროის ინტერვალის სიდიდეს, რომელიც გამოისახება ინდიკატორზე. პერიოდის მოცემა სამართლიანია ყველა რეჟიმისთვის.

სურ. 5.9 ბ -ზე მოცემულია დროითი ინტერვალების გაზომვის სკალა **TIME(hms)**-რეჟიმისათვის, რომელიც შეიცავს საათებისთვის ერთ თანრიგს, წუთებისთვის ორ თანრიგს, წამებისთვის ორ თანრიგს და წამის მეათასედისთვის სამ თანრიგს-გაზომვის სიზუსტე ხომ მილიწამებშია.

5.5.2. სიხშირის გაზომვის რეჟიმი

ამ რეჟიმის ინდიკატორის ფორმატი მოცემულია სურ. 5.10 ა-ზე. წერტილი ბოლოში ნიშნავს, რომ განაზომს არა აქვს წილადი ნაწილი. ყველაფერი იზომება ჰერცებში 0-დან 99999999-დე.

5.5. თვლის რეჟიმი

რეჟიმის ინდიკატორის ფორმატი მოცემულია სურ. 5.10 ბ-ზე. სიდიდე განზომილების გარეშეა. უჩვენებს რაოდენობას მხოლოდ მთელ რიცხვებში.



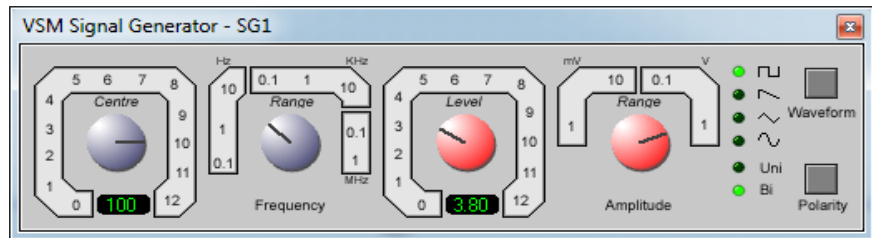
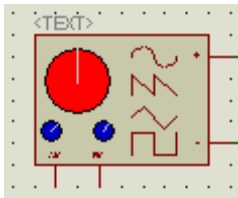
ა.

ბ.

სურ. 5.10.

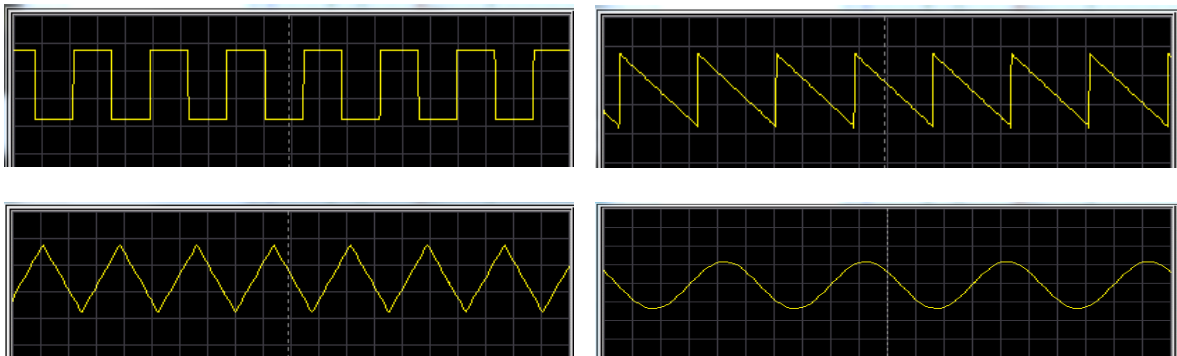
5.6. სიგანალების გენერატორი

გენერატორი შედის პროტეუსის ყველა ვერსიაში. ის ძირითადად გამოიყენება რადიოელექტრონული სქემების მოდელირებისათვის. მისი სქემური აღნიშვნა და მართვის და ინდიკაციის პულტის სახე მოცემულია სურ. 5.11-ზე.



სურ. 5.11.

ხელსაწყო ახდენს ოთხი ტიპის სიგანალების გენერაციას: მართკუთხა, ხერხისე-ბური, სამკუთხა და სინუსოიდალური. შესაბამისი ეპიურები მოცემულია სურ.5.12-ზე



სურ. 5.12.

გამომუშავებული სიგანალების სიხშირე იცვლება 0-12 MHz დიაპაზონში, ხოლო ძაბვის სიდიდე 0-12 V დიაპაზონში.

სქემურ აღნიშვნაში მარჯვნივ გამოსახულია გამომყვანები ნიშნით + და -, განკუთვნილია სიგანალების გადასაცემად მიმღებზე, ხოლო ქვედა გამომყვანებს მიეწოდება გარე სიგანალები გამომუშავებული სიგანალების ამპლიტუდური(AM) და

სიხშირული(FM) მოდულაციისათვის. შესასვლელი სიგნალების ამპლიტუდა იცვლება +/- 12V დიაპაზონში.

რაც შეეხება მართვის და ინდიკაციის პულტს, მასზე განთავსებულია შემდეგი ელემენტები:

პირველი ორი გადამრთველით რეგულირდება სიხშირე- მარჯვენათი დიაპაზონი, მარცხენათი კი დიაპაზონს შიგნით.

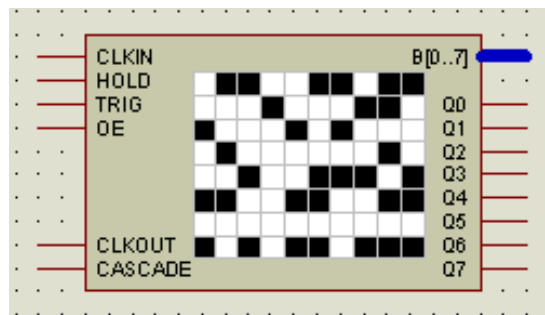
მარჯვენა ორი გადამრთველით რეგულირდება ამპლიტუდა-მარჯვენათი დიაპაზონი, მარცხენათი კი დიაპაზონს შიგნით.

მარჯვენა კიდეში განთავსებულია ინდიკატორები გამომუშავებული სიგნალების ფორმის ჩვენებით და მათი ამოსარჩეველი ლილაკი-Waveform. მის ქვევით გათავსებულია ინდიკატორები და ლილაკი გამომუშავებული სიგნალის პოლარობის ამისარჩევად-Polarity.

5.7. შაბლონების ციფრული გენერატორი

შაბლონების ციფრული ვირტუალური გენერატორი ერთერთი მძლავრი საშუალებაა ციფრული სქემების ტესტირებისათვის. ის გასცემს 8-თანრიგა სიტყვათა თანმიმდევრობას 1kB-ის მოცულობით. გაცემის სიხშირე და სხვა პარამეტრები განისაზღვრება დაპროგრამებით.

გენერატორის სქემური გამოსახულება მოცემულია სურ. 5.13:



სურ. 5.13.

გენერატორს გააჩნია შემდეგი ფუნქციონალური შესაძლებლობები:

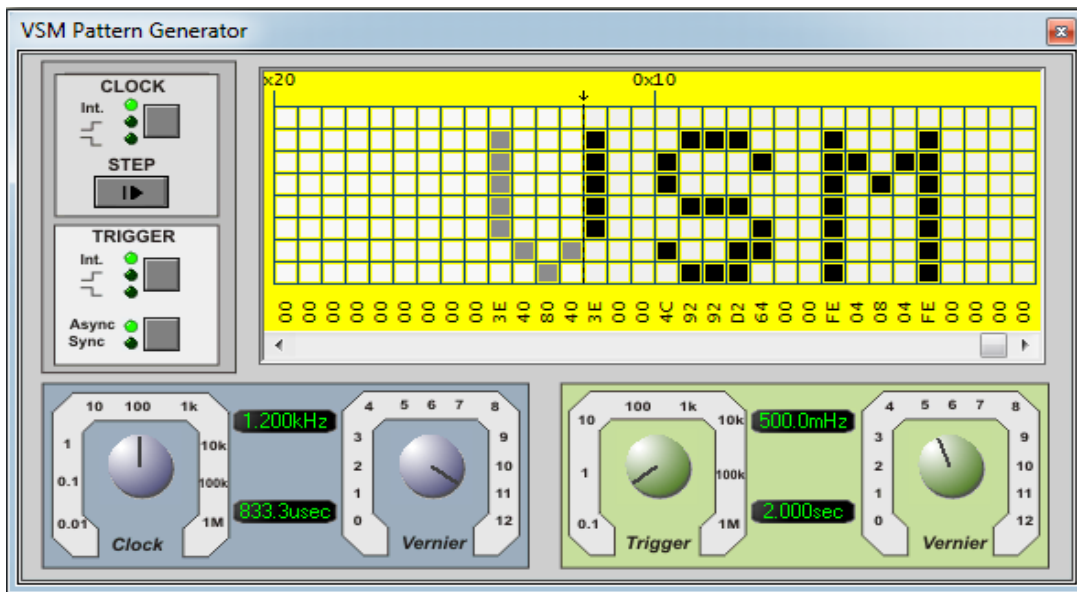
- მუშაობა როგორც ძირითად ისე ინტერაქტიურ რეჟიმში;
- ტაქტირების და სინქრონიზაციის შიგა და გარე რეჟიმები და მათი რეგულირების საშუალებები;
- თექვსმეტობითი და ათობითი ბადის რეჟიმები;
- ინფორმაციის პირდაპირ ან ფაილიდან შეტანის და შენახვის საშუალებები; მუშაობის ბიჯური რეჟიმი;
- დისპლეიზე სპეციფიკური მანიპულაციის საშუალებები.

ხელსაწყოს გააჩნია გამომყვანები:

- **CLKIN**, გარე ტაქტური გენერატორის შესასვლელი;
- **HOLD**, დადებითი სიგნალი ამ შესასვლელზე აჩერებს გენერატორს;
- **TRIG**, გარე სინქრონიზაციის სიგნალის შესასვლელი;

- **OE**, უარყოფითი სიგნალი ამ შესასვლელზე კრძალავს კოდის გაცემას გამოსასვლელზე;
- **CLKOUT**, შინაგანი ტაქტირების სიგნალის გამოსასვლელი;
- **CASCADE**, ეს გამოსასვლელი გადადის მაღალ დონეზე პირველი ბიტის გაცემისას და არის დაბალ დონეზე ყველა სხვა დროს;
- **Q0-Q7**, გაცემი კოდის თანრიგების ინდივიდუალური გამოსასვლელები;
- **B[0-7]**, გაცემი კოდის თანრიგების სალტური გამოსასვლელები;

გენერატორის სქემა განთავსდება რედაქტირების ველში ჩვეულებრივი წესით. ბიჯურ რეჟიმში, **Debug**-ის ან ჩამომლადი მენიუდან ვაქტიურებთ ოპციას **PSM Pattern Generator**. გამოჩნდება გენერატორის მართვის და ინდიკაციის პულტი- სურ. 5.14.



სურ. 5.14.

პულტის მარცხენა მხარეს, **CLOCK** ველში ინდიკატორები, რომლებიც მიუთითებენ შიგა ან გარე(ფრონტის მაჩვენებლები) ტაქტური გენერატორის გამოყენებაზე და ლილაკი რომლითაც ხდება მათ შორის გადართვა. იქვე მოთავსებულია ლილაკი **STEP**, რომლითაც სრულდება ბიჯური რეჟიმი.

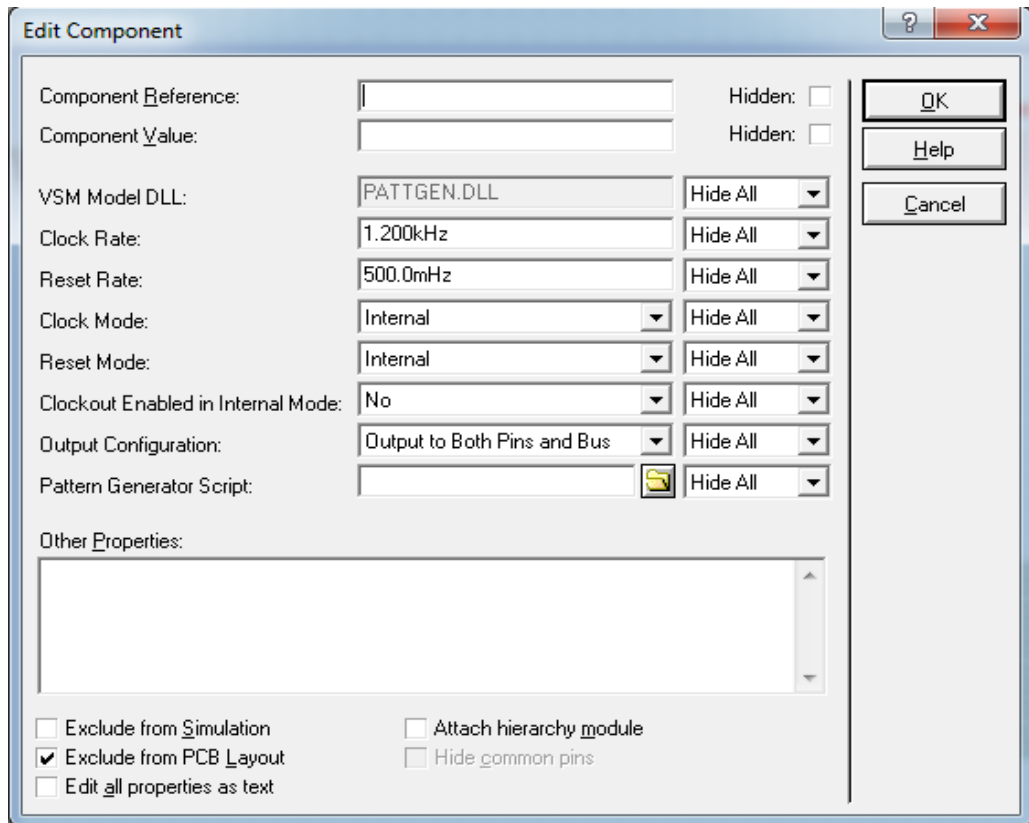
ფანჯარაში **TRIGGER** მოთავსებულია სინქრონიზაციის სიგნალების გადამრთველი ლილაკი-შიგა ან გარე და შესაბამისი ინდიკატორები.

ქვედა რიგში მოთავსებულია ტაქტური-მარცხნივ, და სინქრონიზაციის სიგნალების სიხშირის სარეგულირებელი გადამრთველები.

დისპლეიზე, სადაც გამოსახულია ტექსტი VSM, არსებობს შემდეგი ველები: ზედა ზოლზე მოცემულია ორდინატების ანათვალის 16-ით სისტემაში-0x10, 0x20 და ა. შ. იმავე ზოლშია ისარი, რომლითაც შესაძლებელია იმ ტექსტის მონიშვნა(ბაცი ფერი, ყოველთვის მარცხნიდან), რომელიც არ ექვემდებარება გაცემას. ისრის გამოძახება მოცემულ ადგილზე შესაძლებელია მარცხენა წკაპით ორდინატის თავზე, ხოლო გადადგილება მასზე წარმოშობილი ორთავა ისრით.

ქვედა ზოლზე მოცემულია კოდები, რომლებიც შეესაბამებიან შესაბამის ვერტიკალზე შეტანილ კოდს 16-ით სისტემაში-შევსებული უჯრედი ერთია, ცარიელი

ნოლი. დისპლეიზე გამოსახულების(ტექსტის) შეტანა ხდება ორი ხერხით: დისპლეის უჯრედზე მარცხენა წკაპით-ერთი წკაპი ერთიანის შეტანა, მეორე წკაპი განულება. მეორე ხერხი-ფაილით შეტანა, სადაც ტექსტი მზადდება **ASCII**-კოდებში, ხოლო ფაილს ენიჭება გაფართოება *.ptn. შაბლონთან დასაშვები მანიპულაციები მოცემულია მენიუში, რომელიც გამოიძახება შაბლონზე მარჯვენა წკაპით. მათგან ხშირად ხმარებადია: **Load Pattern...** (შაბლონის ჩატვირთვა); **Save Pattern** (შაბლონის შენახვა); **Save Pattern As...**(შაბლონის შენახვა როგორც). ფაილის მითითება კი საჭიროა შაბლონის გენერატორის გაწყობის ფანჯრიდან **Edit Component**, ველში **Pattern Generator Script** - სურ. 5.15.



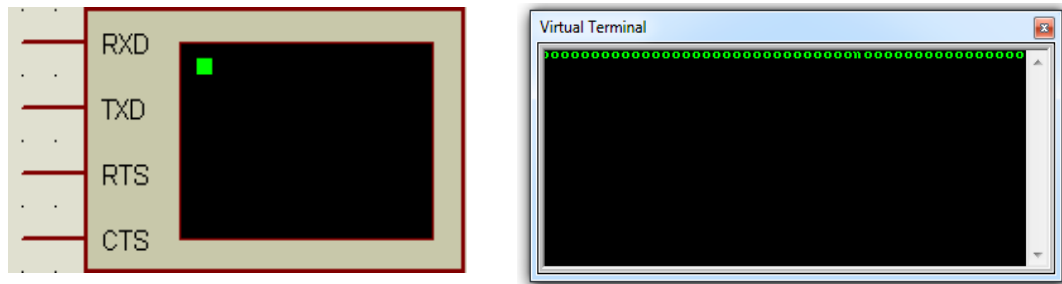
სურ. 5.15.

ვირტუალურ ტერმინალზე გატანის შემთხვევაში, კოდი უნდა გაიცეს მიმდევრობით ბიტებად, ამიტომ ის უნდა ჩაიწეროს ბოლო სტრიქონზე, რადგანაც გაიცემა **Q7** გამოსასვლელიდან.

5.8. ვირტუალურ ტერმინალი

ვირტუალურ ტერმინალი საშუალებას იძლევა პერსონალის კომპიუტერიდან (კლავიატურა, დისპლეი) განხორციელდეს დუპლექსური კავშირი მიმდევრობითი ასინქრონული ინტერფეისის **RS232**-ის საშუალებით. ეს განსაკუთრებით სასარგებლოა განაწილებული სისტემების გამართვის დროს, როდესაც საქმე ეხება ათობით და ასობით დამორეზულ ობიექტებთან კავშირს.

ხელსაწყოს სქემური გამოსახულება და სათვალთვალო ეკრანის სახე მოცემულია სურ. 5.16-ზე:



სურ. 5.16.

ვირტუალურ ტერმინალს გააჩნია შემდეგი ტექნიკური მახასიათებლები:

- სრული დუპლექსური კავშირი - ინფორმაციის მიმდევრობითი მიღება/გადაცემა **ASCII** კოდებში;
- ინფორმაციის მიმდევრობითი მიღება/გადაცემის ორი არხი: **RXD** მონაცემთა მიღებისათვის, **TXD** მონაცემთა გაცემისათვის;
- მართვის ორარხიანი აპარატურული ინტერფეისი: **RTS (ready-to-send)** - მიღების-თვის მზადყოფნა, **CTS (clear-to-send)** - გადაცემისთვის მზადყოფნა;
- გადაცემის სიჩქარე 300-57600 ბოდი;
- ინფორმაციული სიტყვის თანრიგიანობა-7 ან 8;
- ინფორმაციული სიტყვის კონტროლი - ლუწობაზე, კენტობაზე ან არ არის;
- სიტყვის დამთავრების ბიტები-0, 1 ან 2;
- კავშირის დამყარების პროგრამული პროცედურა-**XON/XOFF**, აპარატურული გარდა;
- დადებითი და უარყოფითი დონეები **RX/TX** და **RTS/CTS** სიგნალებისათვის.

ტერმინალის გამოყენებისათვის საჭიროა შესარულდეს მოქმედებები:

მისი გამოსახულების დასმა სქემაზე ცნობილი ხერხით;

RX და **TX** გამომყვანების მიერთება სამოდულო სქემასთან- **RX** შესასვლელია, **TX** გამოსასვლელი;

თუ სისტემაში გამოიყენება კავშირის მართვის აპარატურული საშუალებები მაშინ, საჭიროა სინქრონიზაციის **RTS** გამოსასვლელი მიუერთდეს სქემის იმ შესასვლელს, რომელიც ატყობინებს სისტემას, რომ ტერმინალი მზად არის მონაცემთა მიღებისათვის, ხოლო **CTS** შესასვლელი მიუერთდეს სქემის იმ გამოსასვლელს, რომელიც ატყობინებს ტერმინალს, რომ სისტემა მზად არის მონაცემთა მიღებისათვის და ტერმინალს შეუძლია დაიწყოს გადაცემა.

შესრულდეს ტერმინალის გაწყობის რედაქტირება ცნობილ ფანჯარაში-**Edit Component**: ტერმინალის სახელი, გადაცემის სიჩქარე, სიტყვის სიგრძე, პარიტეტზე კონტროლი და სხვა.

გაიშვას მოდელირება ჩვეულებრივი წესით. ტერმინალი მაშინვე გამოიტანს ეკრანზე მიღებულ მონაცემებს. როცა ტერმინალის გამოსახულების ფანჯარაში მარკერი

ანთებულია, შესაძლებელია კლავიატურაზე აკრებილი მონაცემების გადაცემა ტერმინალიდან სისტემაში.

მოდელირების პროცესში შესაძლებელია გარკვეული მანიპულაციების ჩატარება ტერმინალის მუშაობაში: ასახვის შეჩერება, კოპირება და სხვა. ეს ოპციები მითითებულია მენიუში, რომელიც გამოვა მარჯვენა წკაპით ეკრანზე.

უნდა გავითვალისწინოთ შემდეგი თავისებურებები:

- ვირტუალურ ტერმინალი რეაგირებს შემდეგ მმართველ **ASCII** კოდებზე: CR (0x0D) (კარეტის დაბრუნება), BS (0x08) (დაბრუნება ერთი სიმბოლოთი) и BEL (0x07) (ხმის სიგნალი). ყველა სხვა მმართველი კოდები LF (0x0A) ჩათვლით, იგნორირდება.
- ვირტუალურ ტერმინალი ციფრული მოდელია, ამიტომ არ საჭიროებს სპეციალურ ძაბვებს მუშაობაში.
- **RX** და **TX** გამომყვანების აქტიური მდგომარეობა ძაბვის მაღალი დონეა. იგივეა უქმი სვლის დროს. სტარტის ბიტი- დაბალი დონეა, "სტოპის" ბიტი- მაღალი დონეა. ინფორმაციულ თანრიგებში "1"-მაღალი დონეა, "0"- დაბალი დონეა. ეს თავსებადია **UART** და სხვა მრავალი ინტერფეისებისთვის. სხვა შემთხვევაში უნდა გამოვიყენოთ ინვერსიული სიგნალები.
- **RTS** და **CTS** სიგნალების აქტიური დონე მაღალია. სხვა შემთხვევაში უნდა გამოვიყენოთ ინვერტორები.
- საწყისად შეტანილი სიმბოლოების ეხობეჭდვა არ წარმოებს. თუ ეს საჭიროა უნდა მონიშნოს ოპცია **Echo Typed Characters** კონტექსტურ მენიუში.
- შესაძლებელია ტექსტის წამმღვარება გაშვებისას, რომელიც რეალიზდება ოპციით **TEXT**.

უნდა გავითვალისწინოთ, რომ ნებისმიერი დამატებითი მოდელების შემოტანა სქემაში მოითხოვს დამატებით, ხშირად არაპროპორციულ სიმძლავრეს კომპიუტერისგან.

5.9. ინტერფეისი SPI პროტოკოლის ანალიზატორი

ვირტუალური მოწყობილობა **SPI** (Serial Peripheral Interface) მოიხსენიება სახელითაც **SPI DEBUGGER** და განკუთვნილია მიმდევრობითი გადაცემის მოწყობილობებთან ურთირთქმედებისა და კონტროლისათვის. ტერმინალს აქვს მუშაობის სამი რეჟიმი:

მიმყოლი(**Slave**)- ტერმინალი მუშაობს როგორც დაქვემდებარებული **SPI** მოწყობილობა;

წამყვანი(**Master**)- ტერმინალი მუშაობს როგორც წამყვანი **SPI** მოწყობილობა;

მონიტორი(**Monitor**)-ტერმინალი მხოლოდ აფიქსირებს არხში მიმდინარე გადაცემებს.

ანალიზატორის სქემური გამოსახულება მოცემულია სურ. 5.17-ზე;



სურ. 5.17.

ანალიზატორის გამომყვანებს გააჩნია ფუნქციები:

DIN (Data IN)- მიმყოლი მოწყობილობა ღებულობს მონაცემებს წამყვანისგან;

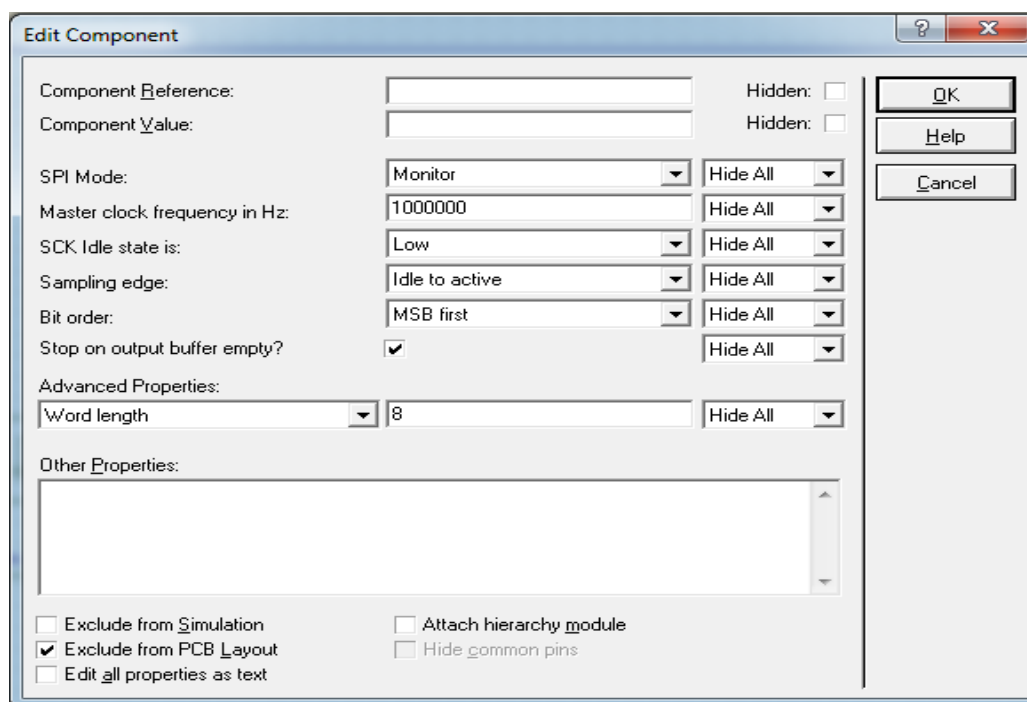
DOUT (Data OUT)- მიმყოლი მოწყობილობა გასცემს მონაცემებს წამყვანზე;

SCK (Serial Clock) - ორმომართულებიანი გამომყვანია: წამყვანის რეჟიმში გასცემს ტაქტურ იმპულსებს, მიმყოლის რეჟიმში ღებულობს ტაქტურ იმპულსებს;

SS (Slave Select) - ორმომართულებიანი გამომყვანია: მიმყოლის რეჟიმში შესასვლელია, წამყვანის რეჟიმში გამოსასვლელია;

TRIG (Trigger) - ყოველთვის შესასვლელია. წამყვანი აქ მიწოდებული სიგნალით იწვევს შენახული სტრიქონის ჩაყენებას გასაცემთა რიგში.

როგორც ყველა მრავალფუნქციური კომპონენტი **SPI** მოითხოვს თვისებათა რედაქტირებას, რომლის ფანჯარა მოცემულია სურ. 5.18-ზე. განმარტებებში, ფიგურულ ფრჩხილებში მოცემულია ტერმინები **Help**-ის მიხედვით.



სურ. 5.18.

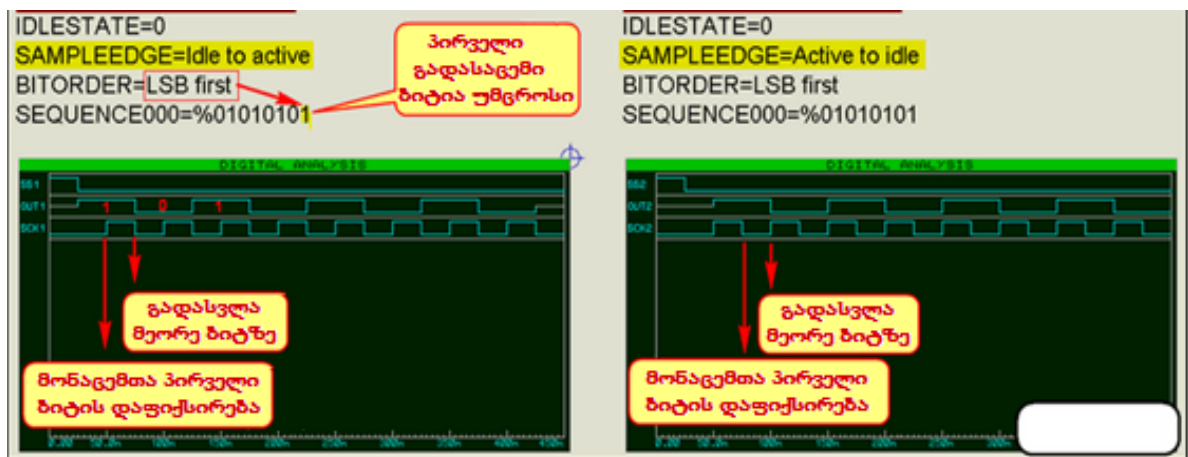
SPI Mode [MODE](SPI რეჟიმი) - ჩამისაშლელ სიაში შეიჩევა საჭირო რეჟიმი სამიდან: **Master**, **Slave**, **Monitor**.

Master clock frequency in Hz {CLOCKFREQ} - სინქრონიზაციის სიხშირე წამყვანის რეჟიმში.

SCK Idle state is {IDLESTATE} - SCK სიგნალის საწყისი დონე: **High** (ლოგ.1) ან **Low** (ლოგ. 0).

Sampling edge {SAMPLEEDGE} - იმპულსის ფრონტი, რომლის დროსაც ფიქსირდება მონაცემი: **Idle to active** - პასიურიდან აქტიურში, თუ აქტიურიდან პასიურში-**Active to idle**. სურ. 5.19 -ზე მარცხენა მხარეს ნაჩვენებია მონაცემთა დაფიქსირება დადებით ფრონტზე, მარჯვენა მხარეს-უარყოფით ფრონტზე. საზოგადოდ, აქტიური შეიძლება იყოს ნებისმიერი ლოგიკური დონე.

Bit order {BITORDER} - ბიტების მიმდევრობა გადაცემაში-უფროსი თანრიგიდან უმცროსისკენ - **MSB first** (MSB - Most Significant Bit) , თუ პირიქით- **LSB first** (Least Significant Bit).



სურ. 5.19.

Stop on output buffer empty {STOPONEMPTY} - ეს ალამი აჩერებს სიმულაციას, თუ გადაცემის ბუფერი ცარიელია. საწყისად ის დაყენებულია.

შემდეგ მოჰყვება **Advanced Properties** ოპციების ჩამოსაშლელი სია, რომელთა მნიშვნელობები დაფიქსირდება გვერდზე ველში:

Word length {WORDLENGTH} - გადასაცემი სიტყვის სიგრძე: ჩვეულებრივ რვა ბიტი, რომელიც შეიძლება შეიცვალოს 1-16-ის ფარგლებში.

Time display precision {TIMEPREC} - დროის ჩვენების სიზუსტეა, რომლითაც ფიქსირდება ბიტების (ბაიტების) მიღება/გაცემის დრო SPI-ში. ეს არის ათობითი ციფრების რაოდენობა მძიმის შემდეგ-იცვლება 1-12 -ის ფარგლებში.

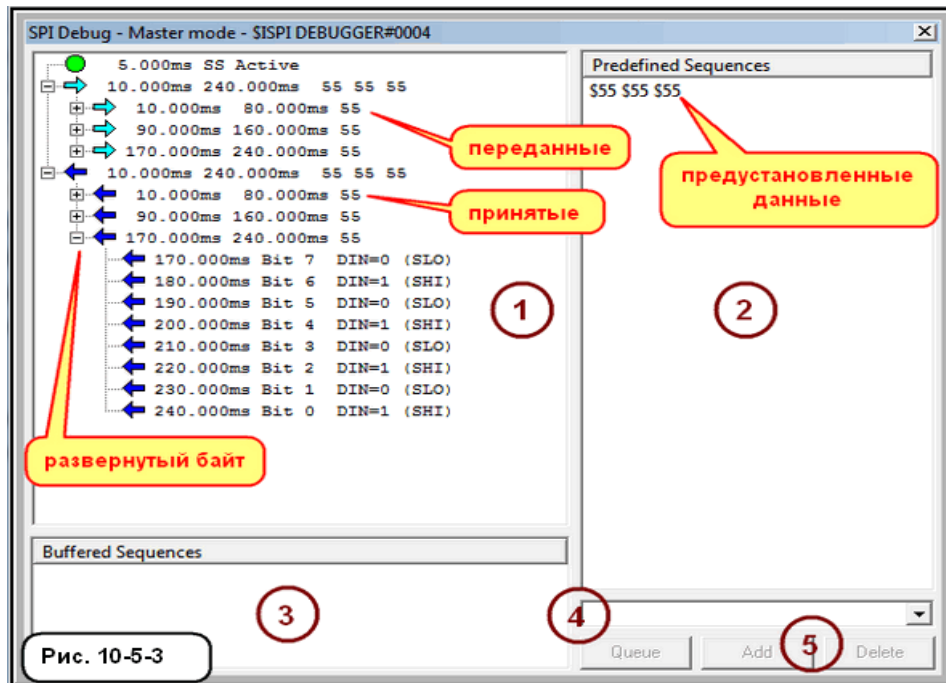
New line after {WRAPLENGTH} - ბაიტების რაოდენობა გადასაცემ სტრიქონში, რომლის შემდეგ ფორმირდება ახალი სტრიქონი. პარამეტრი საწყისად არ ყენდება-უნდა მიეთითოს და იცვლება 1-64 -ის ფარგლებში.

Queue stored sequences at startup {AUTOLOAD} - წინასწარ განსაზღვრული მიმდევრობის გაცემა სიმულაციის დასაწყისში. დაწვრილებით ქვევით.

Sequence file {SEQUENCE_FILE} - ამ სტრიქონში მიეთითება ტექსტური ფაილის სახელი, სადც განთავსებულია გადასაცემი მიმდევრობა წმყვანის რეჟიმის დროს. დაწვრილებით ქვევით.

Loopback mode_{LOOPBACK} - ციკლის რეჟიმის დაყენება, რომლის დროსაც მიღებული მიმდევრობა გაიცემა გამოსასვლელზე.

განვიხილოთ ფანჯარა, რომელიც გამოჩნდება სიმულაციის დაწყებისას, თუ ჩართულია **Debug**-ის ოპციებში ოპცია **Terminal**-სურ. 5.20.



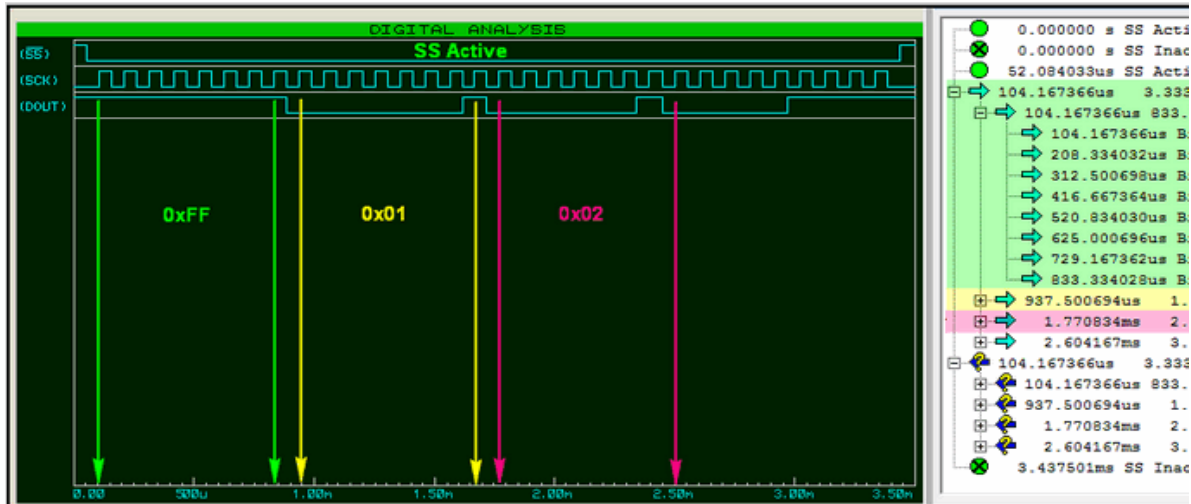
სურ. 5.20.

ფანჯარა დაყოფილია ქვეფანჯრებად, რომელიც დანომრილია ავტორის მიერ.

1.Input Data Display (შემავალი მონაცემების დისპლეი). ფაქტიურად ის აფიქსირებს გამომავალ მონაცემებსაც. ეს ნათლად ჩანს ისრებითაც: მარჯვნივ მიმართული ისრები (ფირუზის ფერი) გამომავალი მონაცემებია, მარცხნივ მიმართული(ლურჯი)-შემავალი. მწვანე ფერის წრით აღნიშნულია SS-ის გააქტიურება(დაბალი დონე). ასეთივე გადახაზული წრე კი, მიუთითებს პასიურ დონეზე-სურ. 5.21, სადაც, თვალსაჩინოებისთვის, შეკვეცილი მონაცემთა ფანჯარა და ციფრული გრაფიკი ერთადაა ნაჩვენები.

თავიდან სტრიქონები ნაჩვენებია ე. წ. დახვეული სახით, რასაც აჩვენებს მის წინ "+". მასზე მარცხენა წკაპი გაშლის ამ სტრიქონს ჯერ ბაიტების დონეზე მათ წინ დასმული პლიუსით, ხოლო მასზე წკაპი გაშლის ბაიტს ბიტებად-ნახაზზე ბოლო ბაიტი. როგორც ბაიტებს ისე ბიტებს მიწერილი აქვს მიღება/გაცემის დრო. მიღების და გაცემის დროის დამთხვევა ნიშნავს, რომ ანალიზატორი მუშაობს სინქრონულ, დუპლექსურ რეჟიმში-შემოსული მონაცემი მაშვე გაიცა გამოსასვლელზე.

თუ SPI-იმ აღმოაჩინა გაურკვეველი ლოგიკური დონე მაშინ, მონაცემებს გამოსახავს რუხ ფონზე, ხოლო ისრებზე დასმულია კითხვის ნიშანი. დროით დიაგრამაზე მოცემულია სამი ბაიტის გადაცემის პროცესი, ამასთან პირველი ბაიტი გახსნილია ბიტებად მონაცემთა ფანჯარაში. თუ SS სიგნალი გადავა ერთიანში ბაიტის გადაცემის დამთავრებამდე, მაშინ ამ ბაიტის ადგილზე დაისმევა ვარაკვლავი.



სურ. 5.21.

2. Predefined Sequences List (წინასწარ მომზადებული მიმდევრობების სია). ამ ფანჯარაში შეიძლება სტრიქონების შეტანა კლავიატურიდან, რომელიც შემდგომ, სიმულაციის დროს ორი მარცხენა წკაპით სტრიქონზე გაიცემა SPI დებაგერის DOUT გამოსასვლელზე. იგივე შეიძლება სტრიქონის მონიშვნით და **Queue** (რიგი) ღილაკზე დაჭერით. უფრო დაწვრილებით-ქვევით.

3. Buffered / Queued Sequences List (მიმდევრობის ბუფერი ანუ მონაცემთა რიგი). ამ ფანჯარაში თავსდება მონაცემები, რომლებიც აუცილებლად გაიცემა SPI დებაგერიდან ავტომატურად, განსხვავებით მეორე ფანჯრისგან, საიდანაც გაცემა ხდება ჩვენი მითითებით.

4. Sequence Entry Box (მიმდევრობის შეტანის ველი). ფანჯარა გამოიყენება მიმდევრობების მოსამზადებლად როგორც წინასწარ მომზადებული სიისთვის, ისე ბუფერისთვის. შეტანა ხდება რიცხვით ან სტრიქონულ ფორმატში: ათობითი რიცხვები 'd' სუფიქსით ან მის გარეშე-3 ან 3d; ორობითი რიცხვები პრეფიქსით '%' ან სუფიქსით 'b': %0010, 0010b; თექვსმეტობითი რიცხვები პრეფიქსებით - '0x' ან '\$' , ან სუფიქსით 'h': 0x4b, ან \$4b, ან 4bh.

5. ღილაკები Queue (რიგი) Add (დმატება) Delete (გაგდება). ღილაკით Queue შესაძლებელია მიმდევრობების მოსამზადებელი ველიდან სტრიქონის შეტანა გაცემის ბუფერში. ასევე, შესაძლებელია მომზადების ველიდან სტრიქონის დამატება მზა სტრიქონების ველში ღილაკით Add და მონიშნული სტრიქონის გაგდება მზა სტრიქონების და ბუფერის ველიდან ღილაკით Delete.

უნდა შევნიშნოთ, რომ ანალიზატორის სრულყოფილი გამოყენებისათვის მიზანშეწონილია ინტერფეისის **screenshot** - ში გარკვევა.

5.10. ინტერფეისი I2C პროტოკოლის ანალიზატორი

ვირტუალური მოწყობილობა I2C (Inter-Integrated Circuit-I²C, არსებობს სახელდებები აბრევიატურებით- IIC, TWI) განკუთვნილია მიმდევრობითი გადაცემის მოწყობილობებთან ურთირთქმედებისა და კონტროლისათვის. ტერმინალი მუშაობს ორ რეჟიმში: წამყვანის და მიმყოლის. მისი სქემური გამოსახულება მოცემულია სურ. 72-ზე:



სურ. 5.22.

ანალიზატორს აქვს კონტაქტები:

SDA (Serial Data) - ორმომართულებიანი გამომყვანია მონაცემთა მიმდევრობითი გადაცემისათვის;

SCL (Serial CLock) - ორმომართულებიანი გამომყვანია სინქროსიგნალების გადაცემისათვის;

TRIG (Trigger) - ერთმომართულებიანი შესასვლელი სიგნალია, რომელიც იწვევს წინასწარ მომზადებული მიმდევრობების გადატანას გაცემის ბუფერში(რიგში).

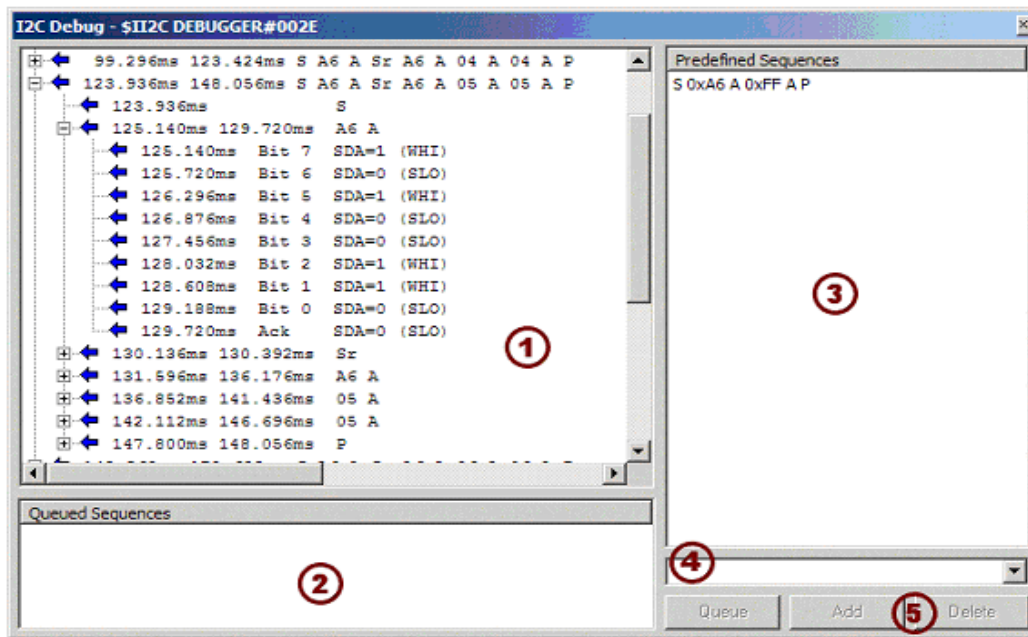
გადაცემის პროცესის კონტროლისა და ანალიზის საუკეთესო საშუალებაა გამართვის ფანჯარა, რომელიც ირთვება **Debug**-ის ოპციით **I2C Debugger**. ეს ფანჯარა დაყოფილია რამდენიმე ზონად, რომელიც საშუალებას იძლევა არა მარტო ვაკონტროლოთ I2C არხში მიმდინარე პროცესები, არამედ ჩავერიოთ მის მუშაობაში. ფანჯრის სახე მოცემულია სურ. 5.23-ზე. განმარტებების მოხერხებულობისათვის ზონები დანომრილია.

1. Input Display Data (შემავალი მონაცემების დისპლეი) - აჩვენებს გადაცემების მიმდინარეობას ბაიტების და ბიტების დონეზე შესაბამისი დროის ჩვენებით. ამასთან, ბაიტებისთვის ნაჩვენებია გადაცემის დაწყების და დამთავრების დრო, ხოლო თანრიგებისთვის მათი ნომერი და ლოგიკური მდგომარეობა დროის გვერდით. ისრებით მითითებულია გადაცემის მიმართულება - აქ შემავალი, ლურჯი ისრები მარცხნივ. თუ რომელიმე თანრიგის ლოგიკური მდგომარეობა გაურკვეველია, ისარზე გამოჩნდება კითხვის ნიშანი.

2. Buffered Sequences List (Queued Sequences) (ბუფერირებული მიმდევრობების სია) - აქ მოთავსებულია მიმდევრობები, რომლებიც ექვემდებარებიან აუცილებელ გადაცემას. მასში მონაცემები შეიტანება მეოხე- მონაცემთა აკრების ზონიდან **Queued** ღილაკზე დაჭერით.

3. Predefined Sequences List (მომზადებული მიმდევრობების სია) - მიმდევრობები, რომლებიც აიკრიბება შეტანის ველში-მეოთხე ზონა, და **Add** ღილაკით შეიტანება ამ

ველში. შემდგომ ისინი შეიძლება გატანილიქნეს გაცემის ბუფერში სტრიქონზე ორჯერ მარცხენა წკაპით, ან მონიშვნით და **Queued** ლილაკზე დაჭერით.



სურ. 5.23.

4. Sequence Entry Box (მიმდევრობის შეტანის უჯრა) - კლავიატურიდან მონაცემთა შეტანის საშუალებაა. ცნობილი წესებით გაფორმებული რიცხვები და სტრიქონები აიკრიბება აქ და გადაიგზავნება მეორე ან მესამე ზონაში. შეგახსენებთ წესებს:

0x და \$ თექვსმეტობითი რიცხვების პრეფიქსებია, h - თექვსმეტობითი რიცხვების სუფიქსია, 0b და % - ორობითი რიცხვების პრეფიქსებია, d - ათობითი რიცხვების არააუცილებელი სუფიქსია.

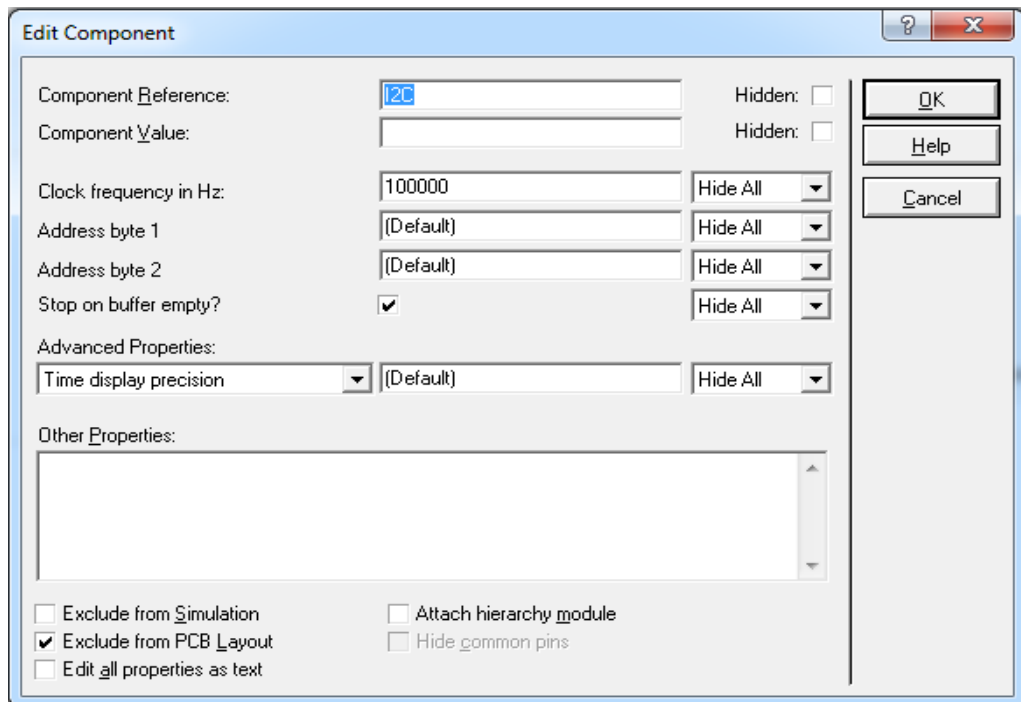
5. Queued (რიგი), **Add** (დამატება) და **Delete** (გაგდება). პირველი ორი ჩვენ უკვე განვმარტეთ. **Delete**-ით კი შეიძლება მონიშნული სტრიქონების გაგდება ნებისმიერი დიალოგური ზონიდან.

როგორც ნებისმიერ მრავალფუნქციურ კომპონენტს **I2C** მოდელსაც ჰქონდება კონფიგურირება. ჩვეულებრივად, **I2C** გამოსახულების მონიშვნით და მარჯვენა წკაპით გამოიძახება ფანჯარა **Edit Component**, სადაც განისაზღვრება პარამეტრები- სურ. 5.24:

აქ ჩამოთვლილ ოპციებში გამოყენებულია ტერმინები, როგორც რედაქტირების ფანჯრიდან, ისე **Help**-ის ტექსტიდან(ფიგურულ ფრჩხილებში).

Clock frequency in Hz {CLOCKFREQ} - სინქროსიგნალებიდან სიხშირის დაყენება, როდესაც ტერმინალი წამყვანის როლშია.

Address byte 1 {ADDRESS1} - განსაზღვრავს მიმყოლი მოწყობილობის პირველი ბაიტის მისამართს. ამ ბაიტში უმცროსი თანრიგი დამისამართებაში არ მონაწილეობს-ის მიუთითებს ოპერაციის ტიპზე: ჩაწერა თუ წაკითხვა. როცა ეს ველი შეუვსებელია - ტერმინალი გადაცემაში არ მონაწილეობს.



სურ. 5.24.

Address byte 2 {ADDRESS2} - მისამართის უფროსი ბაიტი. ეს ბაიტი გამოიყენება 10-თანრიგა დამისამართების დროს. როცა ეს ველი შეუვსებელია- იგულისხმება 7-თანრიგა მისამართი.

Stop on buffer empty {STOPONEMPTY} - ალამი მიუთითებს გადაცემის გაჩერებაზე, როდესაც გადაცემის ბუფერი ათავისუფლია - ფაჭიროა ბაიტის ჩატვირთვა.

ველში **Advanced Properties** მოცემულია ოპციების ჩამოსაშლელი სია, რომელთა მნიშვნელობები დაფიქსირდება გვერდზე ველში:

Time display precision {TIMEPREC} - უჩვენებს დროის სიზუსტეს, რომლითაც ფიქსირდება ბიტების (ბაიტების) მიღება/გაცემის დრო I2C-ში. ეს არის ათობითი ციფრების რაოდენობა მძიმის შემდეგ-იცვლება 1-12 -ის ფარგლებში.

New line after {WRAPLENGTH} - ბაიტების რაოდენობა გადასაცემ სტრიქონში, რომლის შემდეგ ფორმირდება ახალი სტრიქონი. პარამეტრი საწყისად არ ყენდება-უნდა მიეთითოს და იცვლება 1-64 -ის ფარგლებში.

Queue stored sequences at startup {AUTOLOAD} - წინასწარ განსაზღვრული მიმდევრობის გაცემა სიმულაციის დასაწყისში.

Sequence file {SEQUENCE_FILE} - ამ სტრიქონში მიეთითება ტექსტური ფაილის სახელი, სადც განთავსებულია გადასაცემი მიმდევრობა წმყვანის რეჟიმის დროს.

I2C Debugger-ში გამოიყენება სპეციალური მმართველი სიმბოლოები:

S - გამოიყენება სასტარტო პირობის მისათითებლად;

Sr - გამოიყენება რესტარტის პირობის მისათითებლად;

P - გამოიყენება გაჩერების პირობის მისათითებლად;

N - გამოიყენება უარყოფითი დადასტურების პირობის მისათითებლად;

A - გამოიყენება დადებითი დადასტურების პირობის მისათითებლად;

L - გამოიყენება მისათითებლად, რომ არბიტრაჟი დაკარგულია და მოხდება გადასვლა წამყვანის რეჟიმში;

* - მიუთითებს, რომ მონაცემები მღებულია ნაწილობრივ;

? - მიუთითებს, რომ აღმოჩენილია დაუშვებელი ლოგიკური დონე.

ისევ შევნიშნავთ, რომ **I2C** მოდელის სრულყოფილი გამოყენებისათვის უნდა ვიხელმძღვანელოთ **I2C** ინტერფეისის **Datasheet** - ის მონაცემებით.

ეს წიგნის დასასრულია. ჩვენ იმედი გავაქვს, რომ წიგნში მოცემული მასალები მნიშვნელოვნად შეუწყობს ხელს დაინტერესებულ პირებს ვირტუალური მოდელირების ისეთი მძლავრი სისტემის ათვისებაში, როგორცაა **VSM Proteus**.

ლიტერატურა

1. ISIS. Обучающая программа
2. Proteus_ISIS_russian_manual
3. ISIS Proteus. Быстрый старт
4. РАДИО-ЕЖЕГОДНИК 2013 выпуск 24
5. FAQ по PROTEUS для начинающих и не только
6. Виртуальный инструментарий – SPI DEBUGGER. Часть первая
7. Виртуальный инструментарий. Virtual Terminal
8. Краткий учебный курс PROTEUS
9. СРЕДСТВА ОТОБРАЖЕНИЯ (ВИРТУАЛЬНЫЕ ИНСТРУМЕНТЫ)
10. Установка Proteus (<http://hamlab.net/mcu/training/proteus.html>)
11. Симуляция схем в Proteus
12. Проектирование электронных устройств в Proteus 8.1. Часть 8
13. Проектирование электронных устройств в Proteus 8.1. Часть 5
14. Интерфейс SPI микроконтроллеров AVR
15. SPI Функции
16. AVR: SPI
17. Интерфейс I2C
18. Последовательный интерфейс I2C. Часть 1. Описание
19. *Intelligent Schematic Input System* User Manual Issue 6.0 -
November 2002 © Labcenter Electronics

შინაარსი

	ზოგადი საკითხები	3
თავი 1	ISIS პროგრამის ინტერფეისი	5
1.1	ეკრანის ველების ორგანიზაცია	5
1.2	სამუშაო ინსტრუმენტების ორგანიზაცია	8
1.2.1	მენიუს პანელი	8
1.2.2	ბრძანებათა პანელები	8
1.2.3	რეჟიმების პანელები	16
1.2.4	კომპონენტის ორიენტირების პანელი	16
1.2.5	ძირითადი რეჟიმების პიქტოგრამები	17
1.2.6	ხელსაწყოების პიქტოგრამები	18
1.2.7	2D გრაფიკის პიქტოგრამები	19
თავი 2	სქემის პროექტის შექმნა	
2.1	სქემის ელემენტების შერჩევა	21
2.2	ელემენტების განთავსება სქემაზე	23
2.3	სქემის ელემენტების შეერთება	25
2.4	ტერმინალური წერტილები	26
2.5	ელემენტების აღნიშვნა სქემაზე	28
2.6	სქემის ბლოკების რედაქტირება	32
2.7	გამტარების და სალტების გატარება	33
2.8	გამტარების და სალტების მარკირება	33
თავი 3	ციფრული გრაფიკების გამოყენება	
3.1	სასინჯი ზონდების გამოყენება	36
3.2	ციფრული გრაფიკების გამოყენება	38
თავი 4	პროგრამების გამართვა	
4.1	პროგრამის შეტანა მიკროკონტროლერში	42
4.2	პროგრამის გამართვის ბიჯური რეჟიმი	44
4.3	ბიჯური რეჟიმის გამოყენების თავისებურებები	46
4.4	დიაგნოსტიკურ შეტყობინებათა ფორმირება Debug-ში	48
თავი 5	ვირტუალური გამზომი ხელსაწყოების გამოყენება	
	შესავალი	53
5.1	ვოლტმეტრი	54

5.2	ამპერმეტრი	55
5.3	ციფრული ოსცილოგრაფი(ოსცილოსკოპი)	56
5.4	ლოგიკური ანალიზატორი	57
5.5	ტაიმერ-მთვლეელი	59
5.5.1	დროითი ინტერვალების გაზომვა	59
5.5.2	სიხშირის გაზომვა	60
5.5.3	თვლის რეჟიმი	60
5.6	სიგნალების გენერატორი	61
5.7	შაბლონების ციფრული გენერატორი	62
5.8	ვირტუალურ ტერმინალი	64
5.9	ინტერფეისი SPI პროტოკოლის ანალიზატორი	66
5.10	ინტერფეისი I2C პროტოკოლის ანალიზატორი	71
	ლიტერატურა	75