

ELEKTROTEHNIČKI FAKULTET OSIJEK		Električni strojevi
Vježba br. 6:	Programiranje u MATLAB-u	
Student:		
Grupa:		

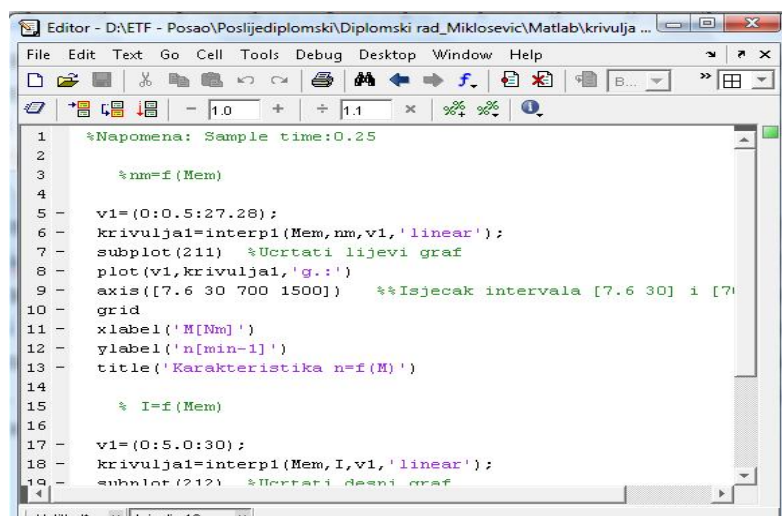
6.1. CILJ VJEŽBE

Ova vježba je namijenjena studentima koji se prvi puta susreću sa MATLAB - ovim programskim jezikom. U tekstu vježbe biti će uključena dva primjera koja bi trebala pomoći da se brzo nauče osnove programskog jezika MATLAB. Korištenjem MATLAB - ovih naredbi moguće je kreirati *m - funkcije* koje simuliraju električni stroj u različitim pogonskim uvjetima, a pomoću *m – skripte* je isto tako moguće analizirati rezultate simulacije kako bi se dobilo željeno rješenje.

6.2. MATLAB - ove m – datoteke

MATLAB je potpuni programski jezik u kojem je moguće napisati vlastite programske dijelove. Pomoću ugrađenih funkcija i programskih paketa moguće je graditi nove programe. Svaki skup MATLAB naredbi napisan korištenjem tekst editora (*m – editora*) koji je pohranjen u datoteci *.m* predstavlja jedan MATLAB program.

Postoje dvije vrste matlabovih *m - datoteka*: skriptna datoteka (*script files*) i funkcijska datoteka (*function files*). Glavna razlika je u dosegu varijabli. MATLAB - ove funkcije se sastoje od kombinacije MATLAB - ovih različitih funkcija, povezanih tako da kao cjelina obavljaju predviđenu zadaću. Funkcijske datoteke mogu prihvatiti ulazne argumente i vratiti izlazne. Skriptne datoteke ne uzimaju ulazne argumente ali vraćaju izlazne. Skripte sadrže slijed naredbi koji bi uobičajeno unosili u komadni redak. Jednostavno zamjenjuju unos iz komadne linije. Na taj način eliminira se ponavljanje unosa naredbi pri složenijim proračunima.



```
1 %Napomena: Sample time:0.25
2
3 %nm=f(Mem)
4
5 v1=(0:0.5:27.28);
6 krivulja1=interp1(Mem,nm,v1,'linear');
7 subplot(211) %Uvertati lijevi graf
8 plot(v1,krivulja1,'g:');
9 axis([7.6 30 700 1500]) %Isjecak intervala [7.6 30] i [700 1500]
10 grid
11 xlabel('M[Nm]')
12 ylabel('n[nmin-1]')
13 title('Karakteristika n=f(M)')
14
15 % I=f(Mem)
16
17 v1=(0:5.0:30);
18 krivulja1=interp1(Mem,I,v1,'linear');
19 subplot(212) %Uvertati desni graf
```

Slika 6.1. M - editor (Primjer skriptne datoteke)

Za stvaranje m – datoteke otvara se New iz izbornika file i odabire se M – file. Na taj način otvara se MATLAB - ov m – editor u koji nam služi za upisivanje naredbi. Takvu datoteku jednostavno pokrećemo utipkavanjem njegovom imena ispred znaka *prompt* unutar komadne linije MATLAB - a, ali pri tome aktivna datoteka mora biti ona u kojoj se m - file pohranjuje. Na taj način sve varijable MATLAB - a postaju dio radnog prozora.

Naredbe MATLAB - a za rad sa datotekama prikazuje sljedeća tablica:

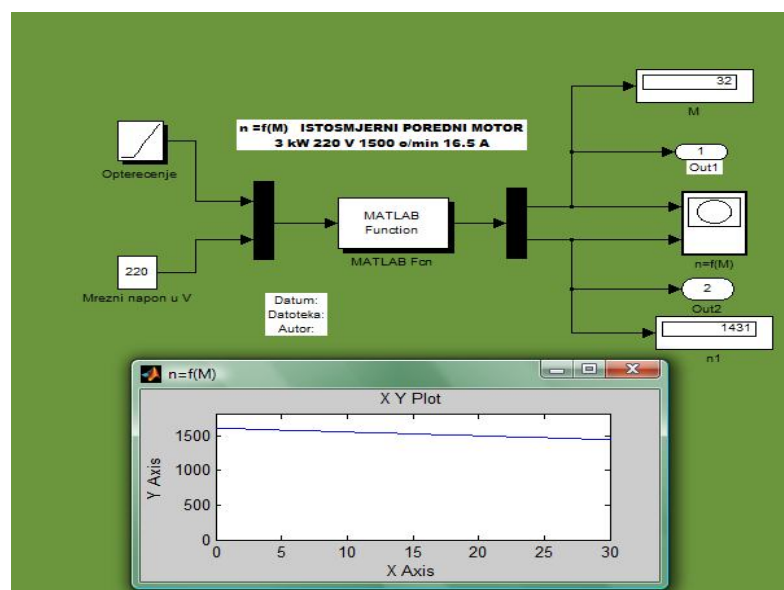
Tablica 6.1.Naredbe za rad sa datotekama

NAREDBA	OPIS NAREDBE
what	Popis m – file -ova u direktoriju
dir	Popis svih datoteka u direktoriju
delete primjer	Briše .m file primjer.m
cd path	Prelazi u zadani direktorij
cd	Prikazuje trenutne direktorije
which primjer	Prikazuje put do datoteke primjer.m
edit primjer	Otvora m - file

6.3. Zadaci za vježbu

6.3.1. Simuliranje kvazistatičkog ponašanja istosmjernog porednog motora korištenjem m – datoteke

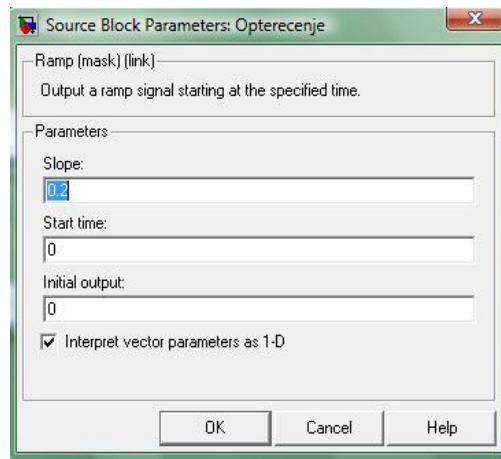
Pojednostavnite blok dijagram iz predhodne vježbe tako da zamijenite čitav dinamički sustav koji proizvodi skokovitu promjenu opterećenja s jednim blokom Ramp iz biblioteke Sources, a sustav motora s jednim blokom MATLAB Fcn iz biblioteke User - Defined Functions. Ostalo izvedite tako dobiveni prikaz odgovara slici 6.2



Slika 6.2. Pojednostavljeni blok dijagram

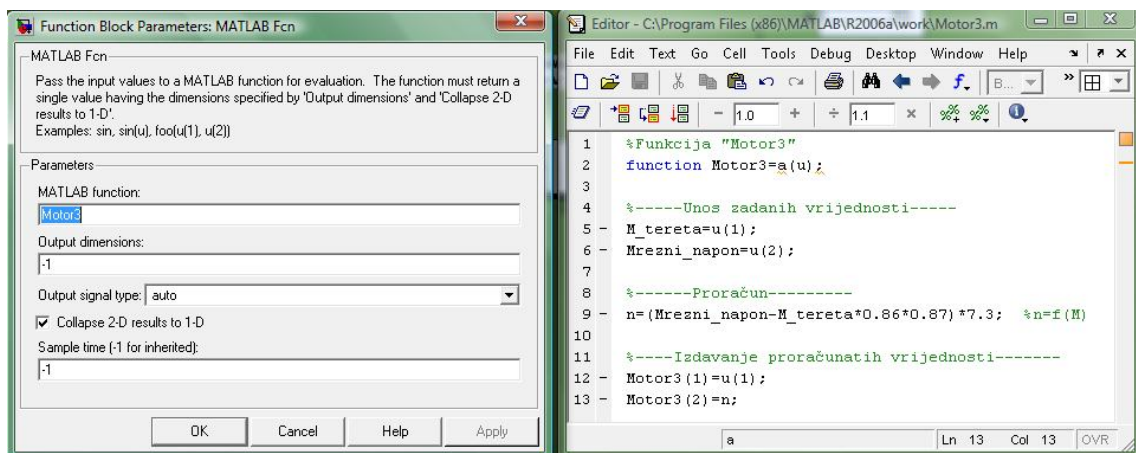
U slici je vidljiv i graf kao rezultat simulacije u Simulinku. No pored njega potrebno je prikazati i rezultat simulacije u MATLAB-u. Iz tog su razloga u blok dijagramu dodana dva bloka Out.

Blok Ramp parametriramo kao u slici 6.3. Aktiviranjem simulacije blok Ramp će pokrenuti opterećivanje i ono će se u skokovima od 2 Nm povećavati sve do dostignuća najvećeg iznosa od 30 Nm. Taj će podatak pokazati displej označen s *M*. Upoznajte se s blokom Ramp preko njegove Help funkcije.



Slika 6.3. Parametriranje bloka Ramp

Sadržaj MATLAB Fcn bloka programirajte u jednoj *m* - datoteci, kako to odgovara programiranju u MATLAB - u. Otvorite ju MATLAB-ovim nalogom *File/New/M-file*, pohranite ju pod jednim nazivom i parametrirajte MATLAB Fcn blok tako da on funkcionira po nalogima te datoteke. U prikazanoj *m* - datoteci nalozi su pohranjeni pod nazivom *Motor3.m*. Naziv funkcije je određen imenom *m* – datoteke, a ne nazivom navedenim u samom kodu funkcije. U slučaju da neznamo kako se zove datoteka u koju smo snimili neku funkciju, korištenjem naredbe *lookfor* ('Naziv funkcije') MATLAB pretražuje sve datoteke do koje ima definirane putanje (*Path*). Pronađe li traženu funkciju isti trenutak navodi i njoj pridruženu *m* – datoteku.

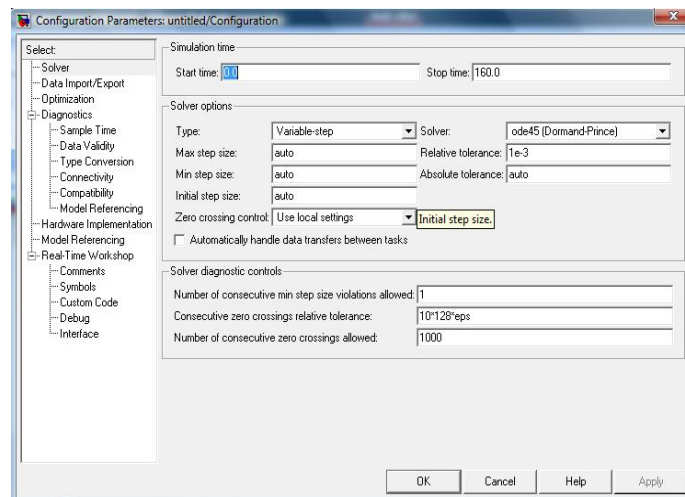


Slika 6.4. Parametriranje bloka MATLAB Fcn Slika 6.5. MATLAB - ova *m* – funkcija

Analizirajmo sadržaj slike 6.5. Prvi red započinje s znakom %. To je *comment* (komentar). MATLAB ignorira sve komentare. Umeću se da bi se poboljšala čitljivost koda. Kada se zadraži pomoć *help naziv funkcije* MATLAB će ispisati komentar koji slijedi iza početnog redka u funkciji. U daljnjem dijelu koda prvo je definiran naziv funkcije. Zatim se defniraju ulazni argumenti, te se preko poznate funkcijske povezanosti brzine vrtnje motora sa momentom tereta prikazuju simulirane (željene) vrijednosti (izlazni argumenti).

Upoznajte se s načinom parametriranja bloka XY graf preko njegove Help funkcije.

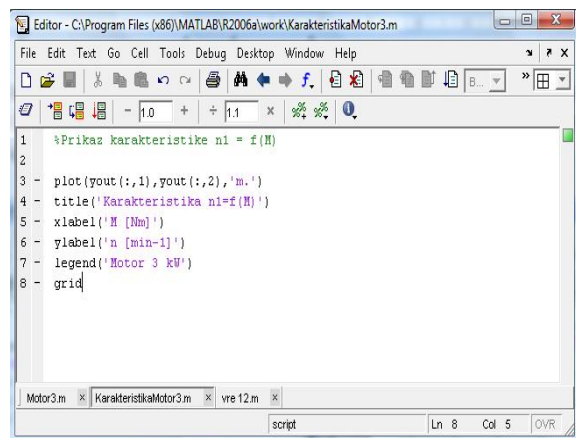
Nakon kreiranja blok modela provedite postupak parametriranja simulacije prema slici 6.6.



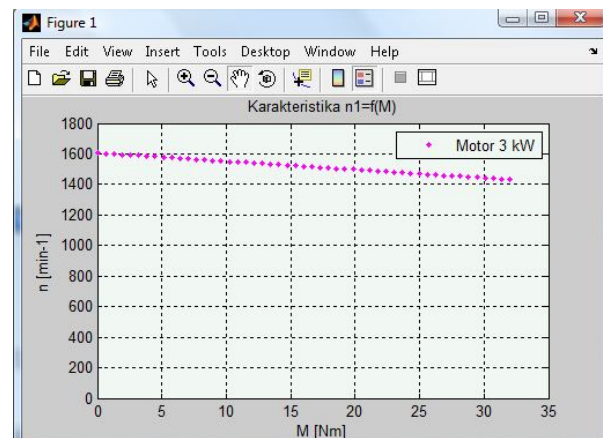
Slika 6.6. Parametriranje simulacije

Na kraju je još potrebno dijagram kao rezultat simulacije (slika 6.2) usporediti s dijagramom koji je programiran u jednoj *m - skripti*.


U prikazanom primjeru je to datoteka *KarakteristikaMotor3.m*. U slici je prikazan i pripadajući *Workspace*. Izgled *m - skripte* treba izgledati kako to pokazuje slika 6.7.



Slika 6.7. M – skripta

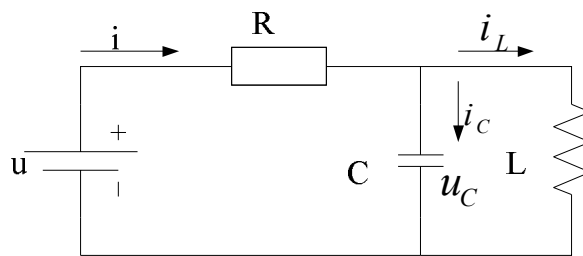


Skripta 6.8. Rezultat simulacije

Klikom na simbol  zadani se nalozi kompiliraju i otvara se dijagram na slici 6.7. Malo doradite dijagram pomoću dijalognog prozora. Doradite krivulju tako da bude točkasta u ljubičastoj boji. Doradite uz to i pripadne osi tako da počinje s vrijednošću nula.

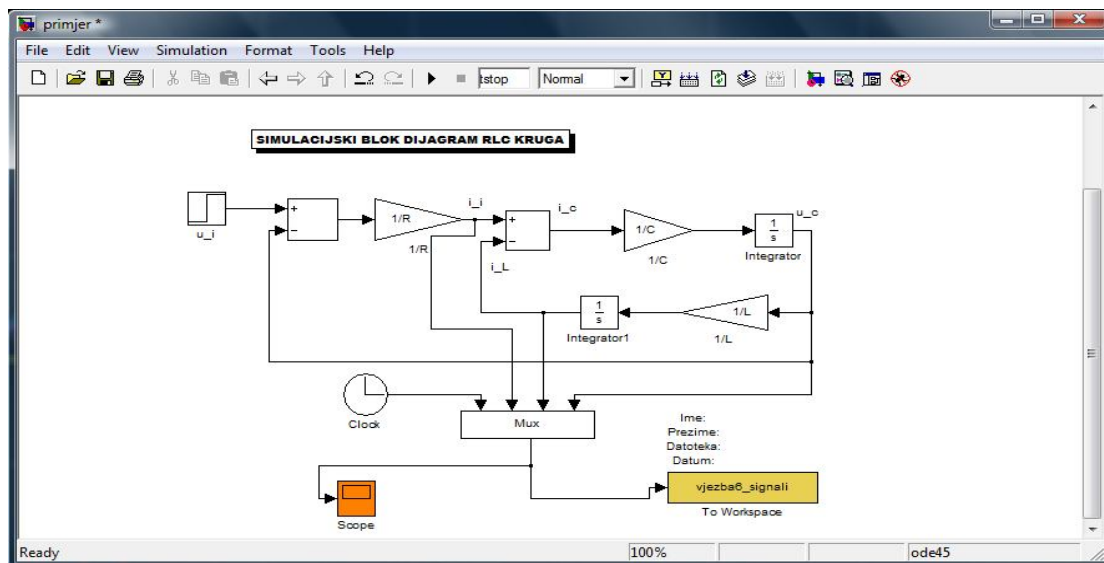
6.3.2. Simuliranje RLC kruga korištenjem m- skripte

Na primjeru sustava drugog reda (slika 6.9.) programirajte *m – skriptu* na način da se umjesto konkretnih numeričkih vrijednosti parametara simulink modela koriste imena varijabli kojima se vrijednost pridjeljuju iz *m – skripte* koja se izvodi prije izvođenja simulacije. Skripta nakon izvođenja simulacije mora generirati dobivene vrijednosti. Usporedite dobivene rezultate sa rezultatima dobivenim u 5. vježbi.



Slika 6.9. RLC krug

Simulacijski blok dijagram prikazuje slika 6.10.

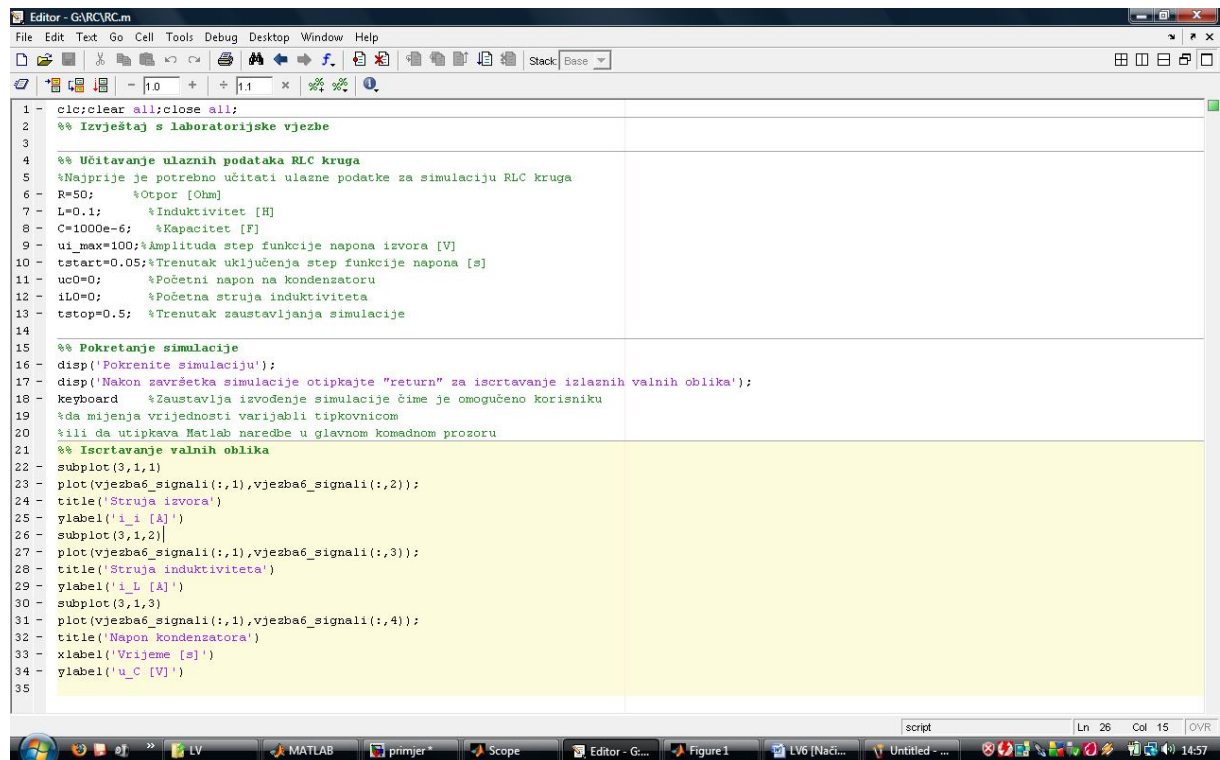


Slika 6.10. Blok dijagram RLC kruga

Vrijednosti parametara RLC kruga postavite na $R=50\Omega$, $L = 0,1 \text{ H}$ i $C=1000\mu\text{F}$. Amplitudu napona izvora podesite na 100 V. Početne vrijednosti struje induktiviteta i napona na kondenzatoru su jednake nuli. Napon izvora narinuti u trenutku $t=0,05\text{s}$, a vrijeme trajanja simulacije je 0,5 s.

U parametrima simulacije odabire se metoda integracije *ode45*, za početno vrijeme (*Start time*) postavite 0 s, a za konačno vrijeme (*Stop time*) utipkajte *tstop*. Vrijednost varijable *tstop* će biti preuzeta iz radnog prozora nakon izvođenja *m – skripte*. Za *min step size* odabire se 0,1ms, *max step size* 1ms, *relativna tolerancija* je $1e-5$.

Slijed unosa naloga unutar *m – skripte* prikazuje slika 6.11. Skriptu sačuvajte pod imenom *RC.m* i spremite je u istu datoteku gdje se nalazi i model *primjer.mdl*



```

1 - c1c;clear all;close all;
2   %% Izvještaj s laboratorijske vježbe
3
4   %% Učitavanje ulaznih podataka RLC kruga
5   %Najprije je potrebno učitati ulazne podatke za simulaciju RLC kruga
6   R=50;      %Otpor [Ohm]
7   L=0.1;     %Induktivitet [H]
8   C=1000e-6; %Kapacitet [F]
9   ui_max=100;%Amplituda step funkcije napona izvora [V]
10  tstart=0.05;%Trenutak uključjenja step funkcije napona [s]
11  uc0=0;     %Početni napon na kondenzatoru
12  iL0=0;     %Početna struja induktiviteta
13  tstop=0.5; %Trenutak zaustavljanja simulacije
14
15  %% Pokretanje simulacije
16  disp('Pokrenite simulaciju');
17  disp('Nakon završetka simulacije otipkajte "return" za iscrtavanje izlaznih valnih oblika');
18  keyboard %Zaustavlja izvođenje simulacije čime je omogućeno korisniku
19  %da mijenja vrijednosti varijabli tipkovnicom
20  %ili da utipkava Matlab naredbe u glavnom komadnom prozoru
21  %% Iscrtavanje valnih oblika
22  subplot(3,1,1)
23  plot(vjezba6_signali(:,1),vjezba6_signali(:,2));
24  title('Struja izvora')
25  ylabel('i_1 [A]')
26  subplot(3,1,2)
27  plot(vjezba6_signali(:,1),vjezba6_signali(:,3));
28  title('Struja induktiviteta')
29  ylabel('i_L [A]')
30  subplot(3,1,3)
31  plot(vjezba6_signali(:,1),vjezba6_signali(:,4));
32  title('Napon kondenzatora')
33  xlabel('Vrijeme [s]')
34  ylabel('u_C [V]')
35

```

Slika 6.11. M – skripta

Objasnenje naloga m – skripte:

Naredbom *keyboard* u *m – skripti* se zaustavlja izvođenje datoteke čime je omogućeno korisniku da mijenja vrijednosti parametara tipkovnicom ili da utipkava MATLAB naredbe u glavnom komadnom prozoru.

Naredbom *return* nastavlja se izvođenje *m – skripte* počevši od naredbe koja slijedi iz programske linije u kojoj se nalazi naredba *keyboard*.

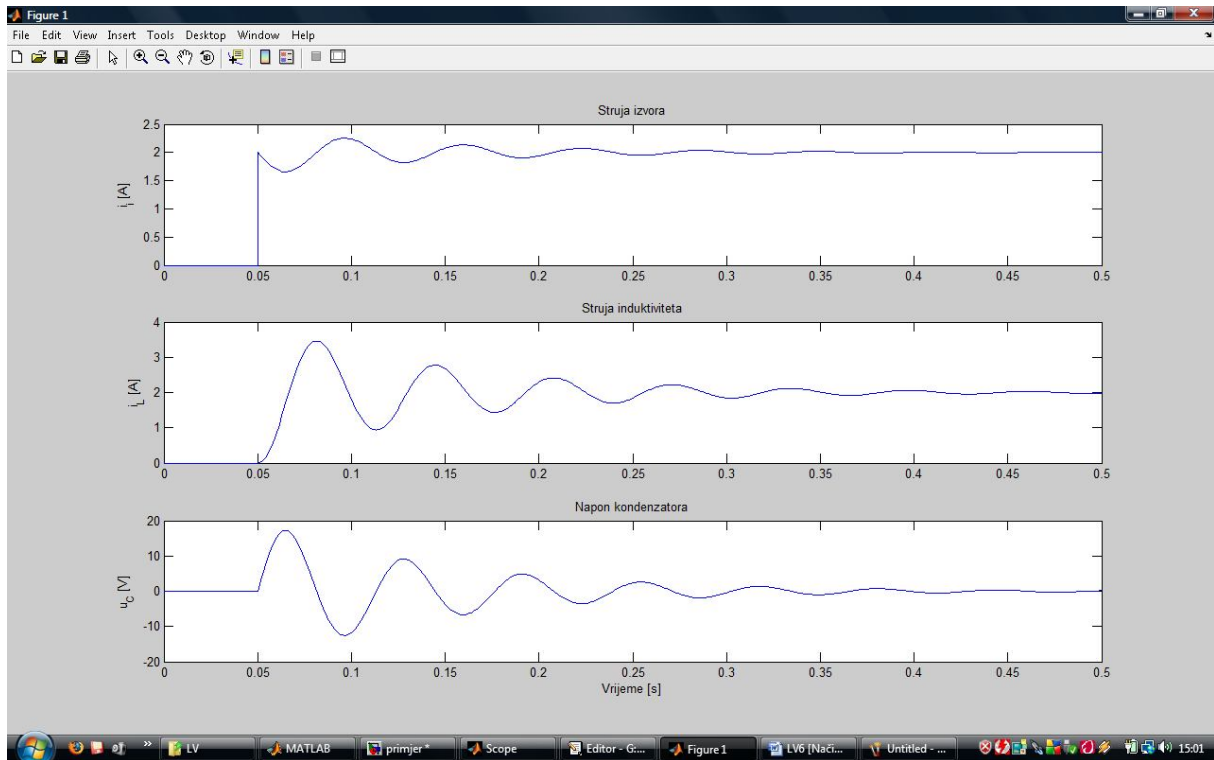
Naredbom *disp('pokrenite simulaciju')* ispisuje se unutar radnog prozora obavijest o mogućnosti pokretanja simulacije. Nakon toga slijedi mogućnost pokretanja simulacije.

Od inačice 7 MATLAB podržava novi oblik komentara unutar koda. Obični komentar počinje s znakom *%*, a osim njega uveden je i znak *%%*. Takav komentar određuje početak bloka koda sve do slijedećeg retka s *%%*. Osjenčani dio je trenutno aktivni blok. Ugrađeni *cell editor* podržava jednostavnu manipulaciju s takvim dijelovima koda što uključuje i izvođenje koda koji se nalazi između takva dva komentara.

Napomena:

Nalogom *sim('naziv modela')* automatski se pokreće simulacija iz Simulinka (predhodno podešena u izborniku *Configuration Parameters*) te se pripadne definirane izlazne varijable spremaju unutar radnog prostora MATLAB - a. Nalog *sim('naziv modela')* nije unesen u *m – skriptu*. Želi li se automatsko generiranje izlaznih varijabli bez zaustavljanja izvođenja *m - datoteke* tada je potrebno umjesto 17. i 18. linije koda unijeti nalog *sim('naziv modela')*.

Rezultati simulacije pohranjeni su u varijabli *vjezba6_signali*. Generiraju se na način da se prvo kompilira jedan dio *m – skripte* sve do naloga *keyboard*. Nakon toga potrebno je pokrenuti simulaciju modela iz čega se dobivaju izlazne varijable koje se direktno učitavaju u radni prostor MATLAB - a. Otipkavanjem naloga *return* u radni prostor MATLAB - a slijedi generiranje valnih oblika dobivenih unutar istog prozora. Sliku pod nazivom Figure 1 spremite u datoteku sa ostalim matlabovim fajlovima.



Slika 6.12. Valni oblici odziva RLC kruga na step napon izvora