





مدل سازی شناختی

Cognitive Modeling

Presented by: Dr. Maleki,

Semnan University,

Spring 2024,

<http://maleki.semnan.ac.ir>



مبحث چهارم: میان پرده: محاسبه کردن با حلقه ها

ای نامی اسرار الهی که تویی
ای نامی اسرار الهی که تویی
دوی آیه سی حال شاهی که تویی
سیرون ز تونیت آخی در عالم هست
از خود بطلب هر آخی خواهی که تویی

Presented by: Dr. Maleki,

Semnan University,

Spring 2024,

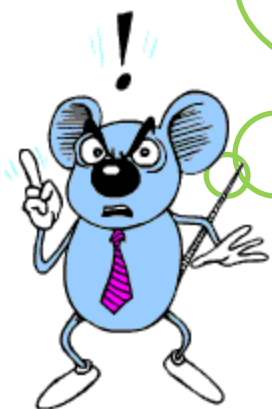
<http://maleki.semnan.ac.ir>

فهرست مطالب

- ← بیش از یک نوع زبان برنامه نویسی کامپیوتر وجود دارد!
- برنامه نویسی فرمائی
- تمرین هارمونیک فرد پاستون توسط حلقه ها
- بیشتر بدانیم: برنامه نویسی مثال حرکت فرد متلب
- بیشتر بدانیم: برنامه نویسی مثال حرکت فرد سیمولینک

تمام زبان‌های کامپیوتری یکسان نیستند. علاوه بر تفاوت‌های بین زبان‌های مجزا، تفاوت‌هایی در نوع زبان‌ها نیز وجود دارد.

شبیه تفاوت حروف الفبا! حروف الفبای انگلیسی و فارسی ظاهری کاملاً متفاوت دارند ولی در هر دو، واژه‌ها از پشت سر هم قرار گرفتن حروف ایجاد می‌شوند. پس، از یک نوع هستند. در حالی که تصویرنگاری (هیروگلیف) مصری یا کاراکترهای چینی از نوعی دیگر است که در آنها، هر پیکوگراف نشانگر یک واژه یا مفهوم است.



دو نوع اصلی زبان‌های کامپیوتری:

❖ زبان‌های فرمانی (imperative language)

❖ زبان‌های تابعی (functional language)

فهرست مطالب

بیش از یک نوع زبان برنامه نویسی کامپیوتر وجود دارد!

برنامه نویسی فرمائی ←

تمرین هارمونی در پیاپیون توسط حلقه ها

بیشتر بدانیم: برنامه نویسی مثال حرکت فرد مبتد

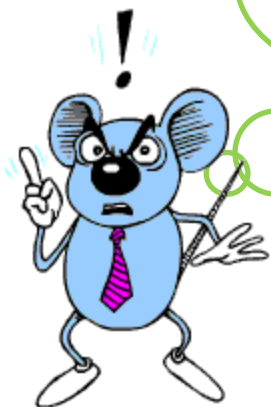
بیشتر بدانیم: برنامه نویسی مثال حرکت فرد سمولینک

برنامه نویسی فرمائی

هر زبان برنامه نویسی که به صورت دنباله‌ای از دستورالعمل‌ها طراحی شده باشد.

در این فصل به زبان برنامه نویسی پایتون (Python) خواهیم پرداخت.

- مثال‌های فراوانی از آن می‌توان در اینترنت یافت.
- رایگان است.
- ابزارهای مناسبی برای آن وجود دارد.
- کتابخانه‌های بسیار سودمندی برای آن وجود دارد.



نمونه‌هایی از کتابخانه‌های پایتون:

- ❖ کتابخانه `scipy`: شامل توابعی برای روال‌های رایج علمی نظیر تبدیل فوریه
- ❖ کتابخانه `psychopy`: جعبه‌ابزاری برای انجام آزمون‌های روان فیزیک

شبکه حلقه for در پایتون

اگر لیستی از اعداد داشته باشیم و بخواهیم مربع هر کدام را محاسبه کنیم:

Listing 4.1 Pseudocode for loop

A simple { for } loop example

```
myListNum = [ 1 , 2 , 3 , 4 ]
```

```
myNewListNum = [ ]
```

```
for i in myListNum :
```

```
    myNewListNum . append ( i * i )
```

```
end
```

❖ شبه کد

❖ عبارت توضیحی

❖ پیوست کردن

حلقه‌های for برای تکرار مجموعه‌ای از دستورات به دفعات مشخص کاربرد دارد.

شبهه حلقه while در پایتون

Listing 4.2 Pseudocode while loop

```
myListNum = [ 1 , 2 , 3 , 4 ]
```

```
myNewListNum = [ ]
```

```
i = 0
```

```
while i < length (myListNum) :
```

```
    myListNum . append (myListNum[ i ] * myListNum [ i ] )
```

```
    i = i + 1
```

```
end
```

❖ حلقه‌ی بی‌نهایت

حلقه‌های **while** برای تکرار مجموعه‌ای از دستورات تا هنگام برآورده شدن شرطی خاص کاربرد دارد. این شرط، مسند (**predicate**) نامیده می‌شود.

فهرست مطالب

بیش از یک نوع زبان برنامه نویسی کامپیوتر وجود دارد!

برنامه نویسی فرمائی

تمرین هارمونی در پیاپیون توسط حلقه ها ←

بیشتر بدانیم: برنامه نویسی مثال حرکت فرد متلب

بیشتر بدانیم: برنامه نویسی مثال حرکت فرد سیمولینک

کد مثال حرکت نردبایتون

Listing 4.3 Python and our spring

```
import pylab as pyl
dt = 0.05
p = -5.0
sp = 5.0
acc = [ p*sp ]
vel = [ 0.0 ]
s = [ sp ]
t = [ 0.0 ]
for i in range ( 1 , 100 ) :
    acc.append ( s[-1] * p )
    vel.append ( vel[-1] + acc[-1] * dt )
    s.append ( s[-1] + vel[-1] * dt )
    t.append ( dt * i )
dp = pyl.plot ( t , s )
pyl.show ( )
```

❖ فصل ۱۶: روند نصب و استفاده از پایتون

❖ تایپ و ذخیره متن در فایلی مثلا به نام `de.py`

❖ باز کردن یک خط فرمان در کامپیوتر

❖ مسیریابی به پوشه فایل

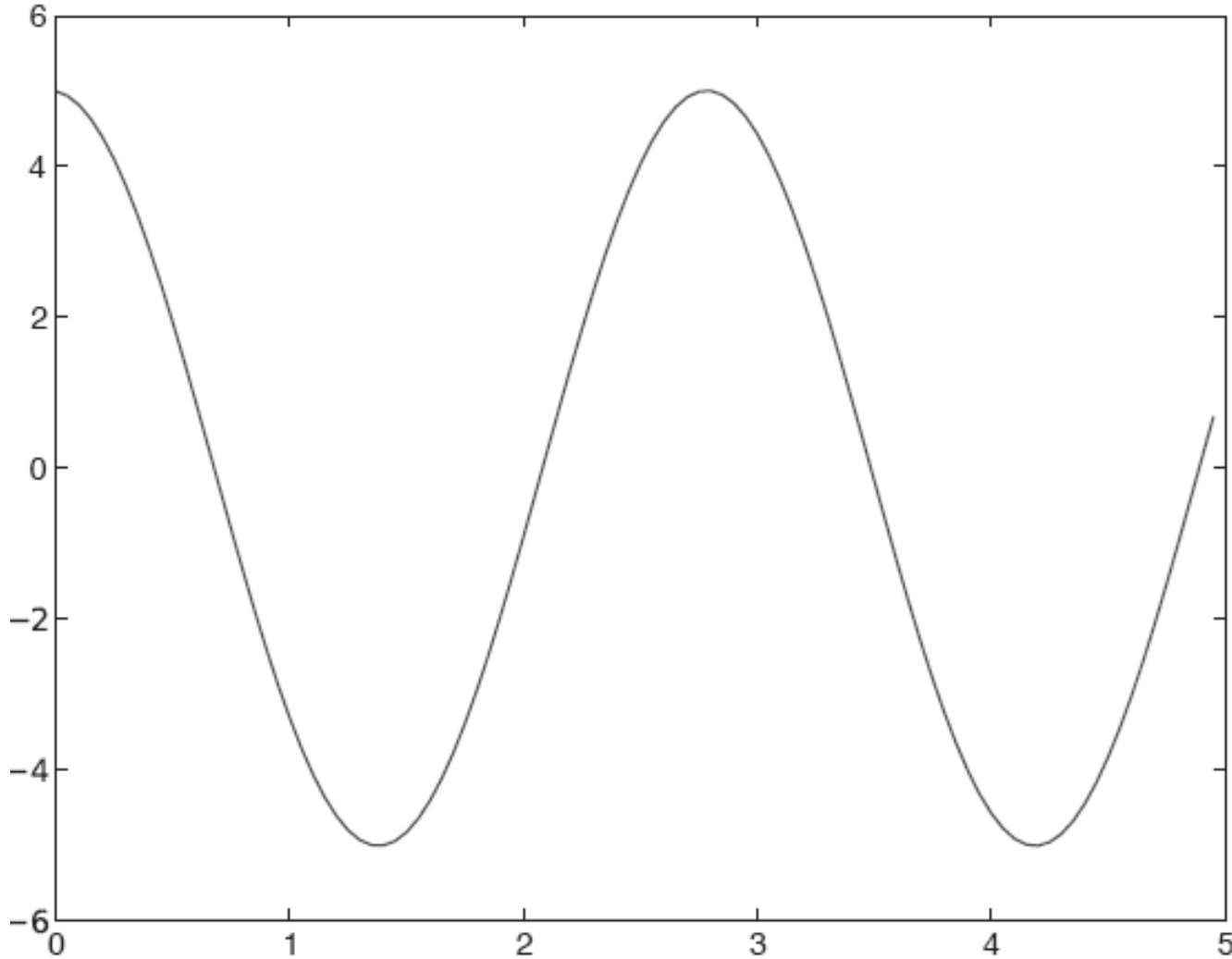
❖ Python2 `de.py`

$$a(t) = -P s(t)$$

$$\Delta v(t) = a(t) \Delta t$$

$$\Delta s(t) = v(t) \Delta t$$

نتیجہ اجرائی برنامہ



فهرست مطالب

بیش از یک نوع زبان برنامه نویسی کامپیوتر وجود دارد!

برنامه نویسی فرمائی

تمرین هارمونی در پیاپیون توسط حلقه ها

بیشتر بدانیم: برنامه نویسی مثال حرکت فرد متلب ←

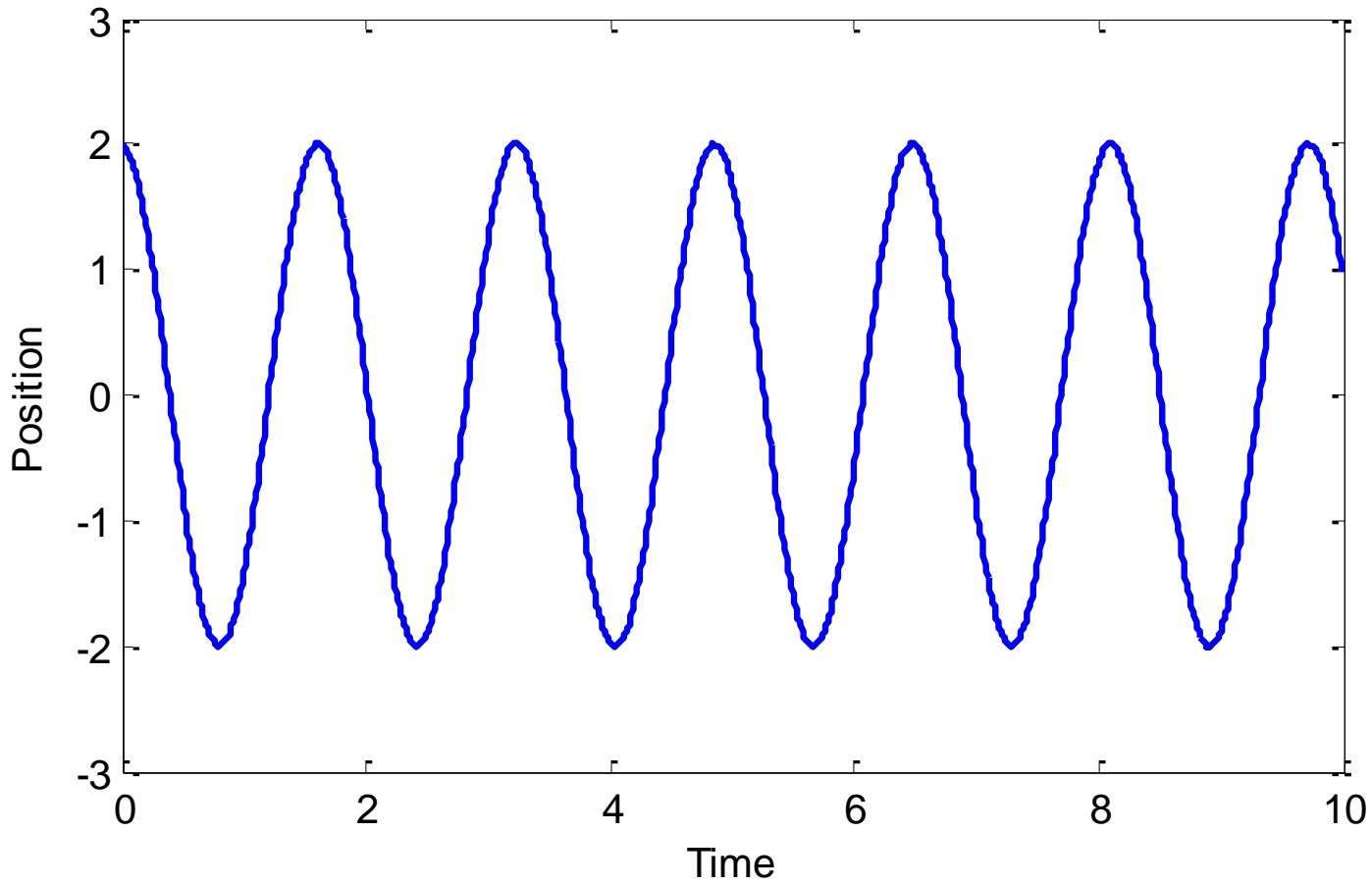
بیشتر بدانیم: برنامه نویسی مثال حرکت فرد سیمولینک

شیه سازی در متلب

```
clc, close all, % Preparation
P = 15; % Constants and Initial Conditions
Dt = 0.01;
V = 0;
S = 2;
a(1) = -P*S;
Dv(1) = a(1)*Dt;
v(1) = V + Dv(1);
Ds(1) = v(1) * Dt;
s(1) = S + Ds(1);
t(1) = 0;

for i = 2:1000,
    a(i) = -P*s(i-1);
    Dv(i) = a(i)*Dt;
    v(i) = v(i-1) + Dv(i);
    Ds(i) = v(i) * Dt;
    s(i) = s(i-1) + Ds(i);
    t(i) = t(i-1) + Dt;
end
plot(t,s), xlabel('Time'), ylabel('Position'),
```

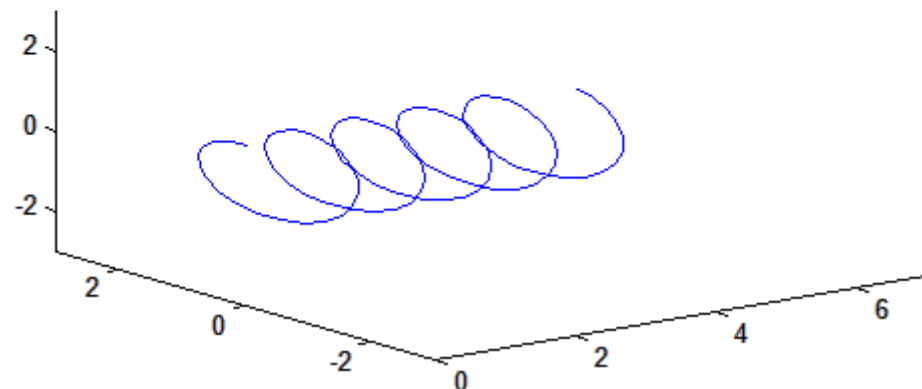
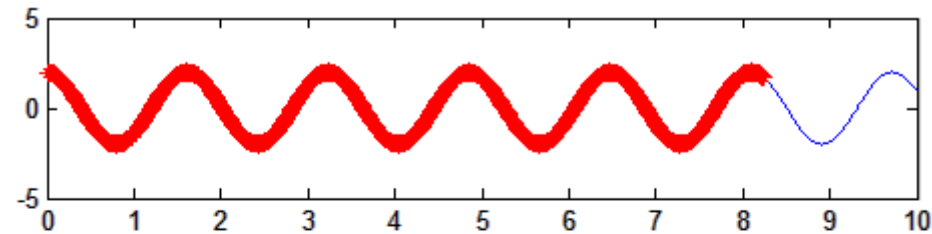
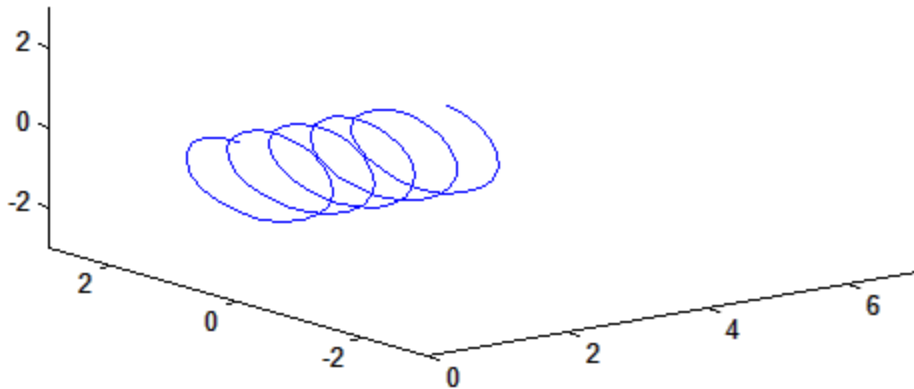
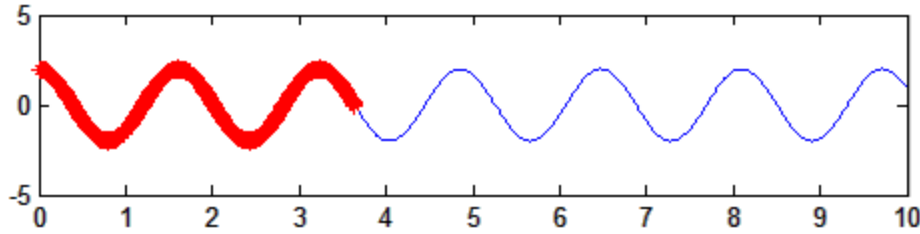

حاصل اجرای برنامه:



انیمیشن ساده!

```
figure, subplot(3,1,1), plot(t,s), hold on,  
subplot(3,1,[2 3])  
theta = linspace(0,10*pi,100),  
x = sin(theta);  
y = cos(theta);  
  
for j = 1:length(s),  
    subplot(3,1,1), plot(t(j),s(j), '*r')  
    z = (s(j)+3) * theta / (10*pi);  
    subplot(3,1,[2 3])  
    plot3(z,x,y);axis([ 0 7 -3 3 -3 3])  
    pause(0.01),  
end
```

حاصل اجرای برنامه در دو لحظه!



فهرست مطالب

بیش از یک نوع زبان برنامه نویسی کامپیوتر وجود دارد!

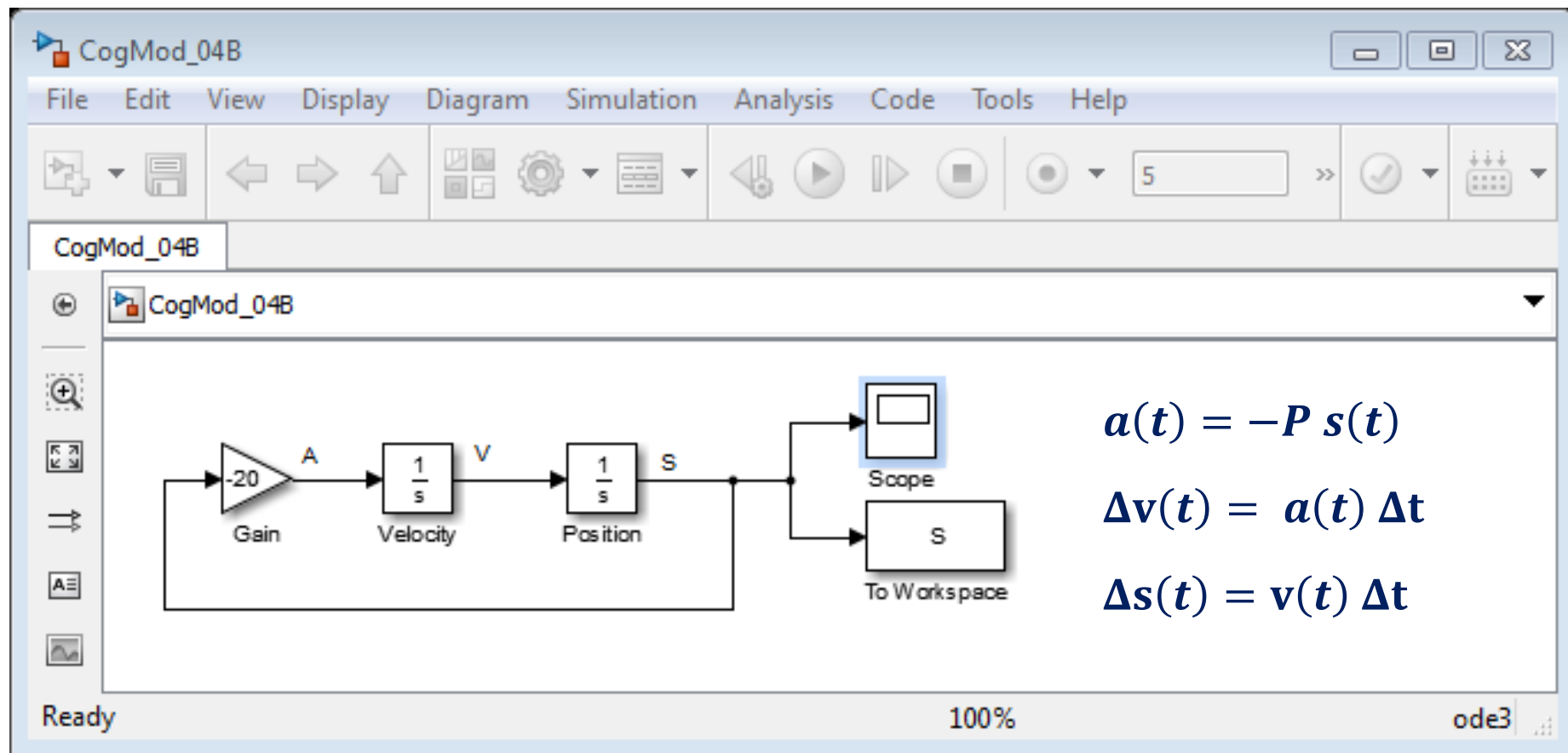
برنامه نویسی فرمائی

تمرین هارمونی در پیاپیون توسط حلقه ها

بیشتر بدانیم: برنامه نویسی مثال حرکت فرد متلب

بیشتر بدانیم: برنامه نویسی مثال حرکت فرد سیمولینک ←

نمودار بلوکی سیستم جرم - فنر در سیمولینک



$$a(t) = -P s(t)$$

$$\Delta v(t) = a(t) \Delta t$$

$$\Delta s(t) = v(t) \Delta t$$

تنظیمات بلوک بهره

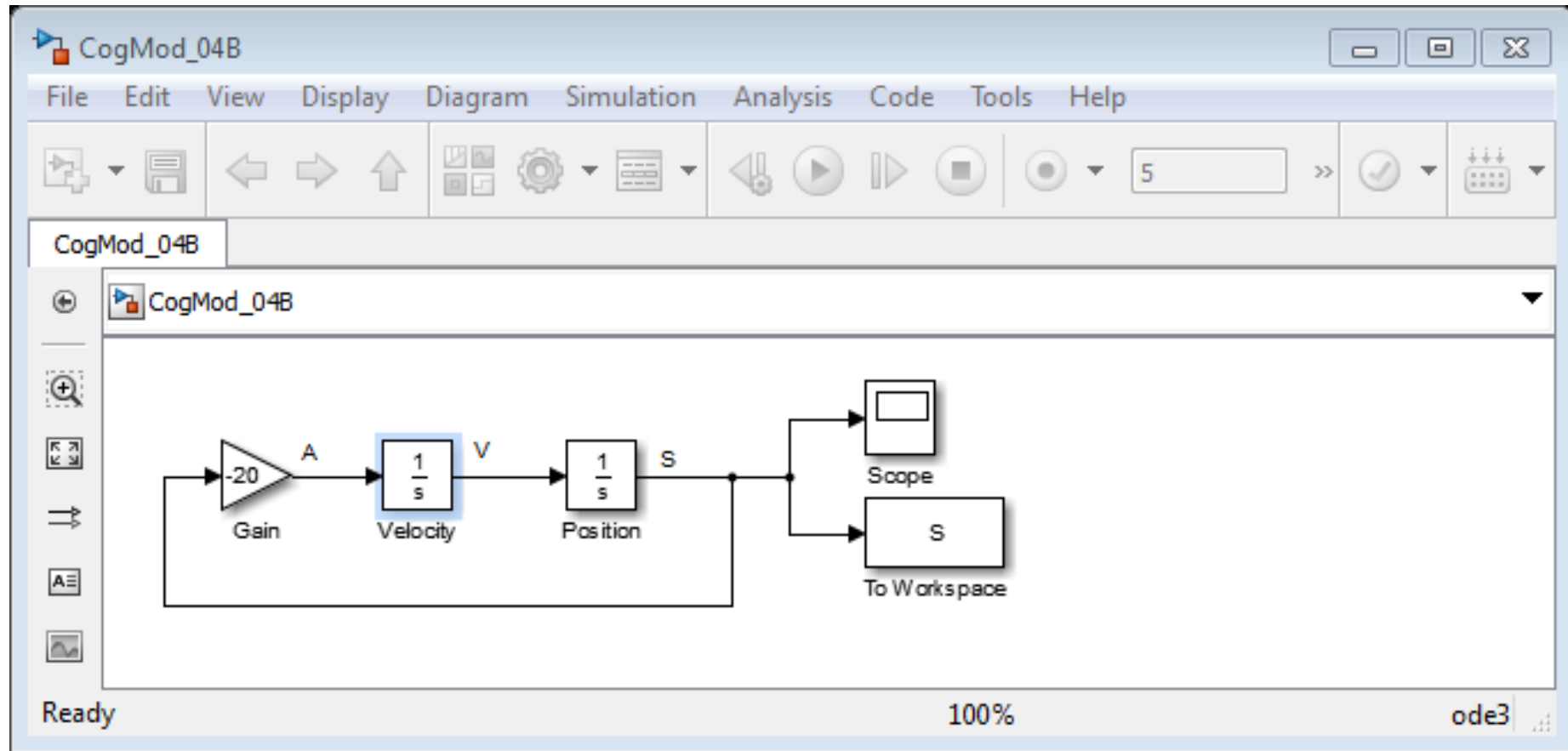
وارد کردن :?

The image shows a MATLAB Simulink interface with a 'Function Block Parameters: Gain' dialog box open. The dialog box has a title bar with a close button. It contains the following fields and options:

- Gain:** A text field containing the value -20 . Below it is the text: "Element-wise gain ($y = K.*u$) or matrix gain ($y = K*u$ or $y = u*K$).
- Main:** A tabbed interface with two tabs: "Signal Attributes" (selected) and "Parameter Attributes".
- Gain:** A label above a text field containing -20 .
- Multiplication:** A dropdown menu showing "Element-wise($K.*u$)".
- Sample time (-1 for inherited):** A text field containing -1 .
- Buttons:** "OK", "Cancel", "Help", and "Apply" buttons at the bottom.

The background shows a Simulink model with a Gain block (labeled "Gain" and "A") and a Velocity block (labeled "Velocity"). The Gain block has a value of -20 inside a triangle. The Velocity block is a rectangular block with $\frac{1}{s}$ inside.

تنظیمات بلوک انتگرال گیر سرعت



Function Block Parameters: Velocity

Integrator

Continuous-time integration of the input signal.

Parameters

External reset: none

Initial condition source: internal

Initial condition:

0

Limit output

Upper saturation limit:

inf

Lower saturation limit:

-inf

Show saturation port

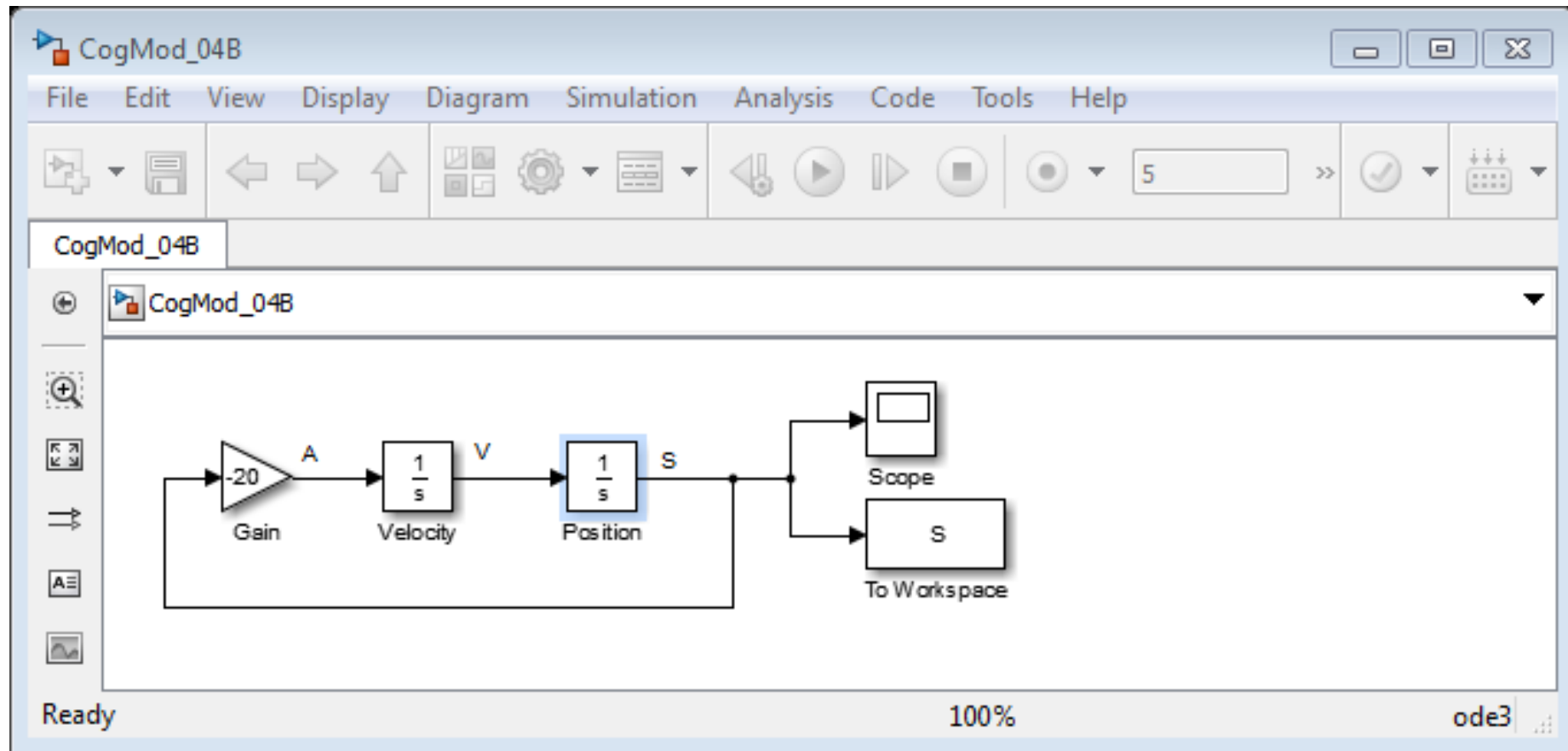
Show state port

Absolute tolerance:

auto

Ignore limit and reset when linearizing

تنظیمات بلوک انتگرال گیر موقعیت:



Function Block Parameters: Position ✕

Integrator

Continuous-time integration of the input signal.

Parameters

External reset:

Initial condition source:

Initial condition:

Limit output

Upper saturation limit:

Lower saturation limit:

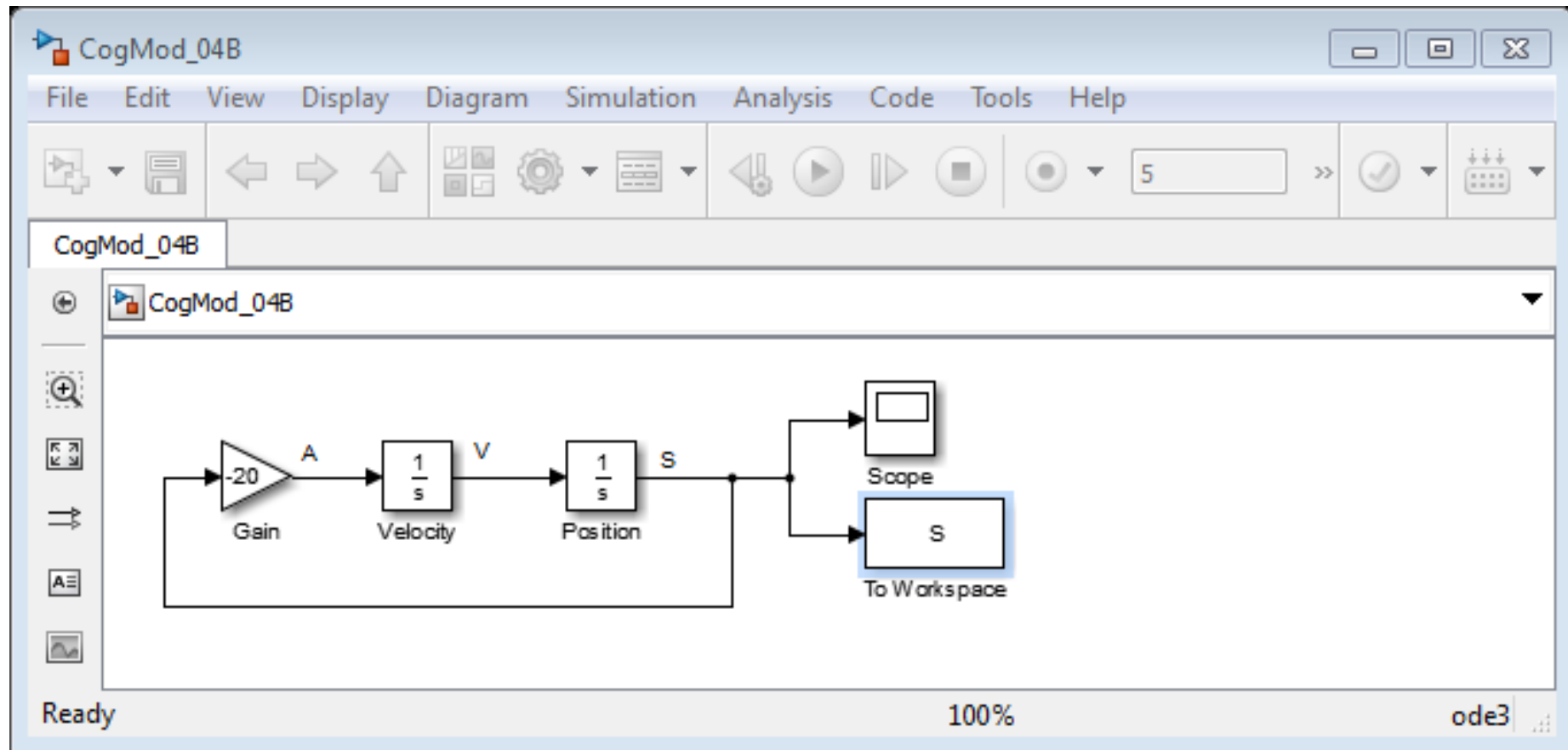
Show saturation port

Show state port

Absolute tolerance:

Ignore limit and reset when linearizing

بلوک ارسال به فضای کاری



Sink Block Parameters: To Workspace ✕

To Workspace

Write input to specified timeseries, array, or structure in a workspace. For menu-based simulation, data is written in the MATLAB base workspace. Data is not available until the simulation is stopped or paused.

To log a bus signal, use "Timeseries" save format.

Parameters

Variable name:

S

Limit data points to last:

inf

Decimation:

1

Save format: Structure With Time

Log fixed-point data as a fi object

Sample time (-1 for inherited):

-1

انجام تنظیمات شبیه سازی

The screenshot displays the MATLAB/Simulink software interface. The main window is titled "CogMod_04B". The menu bar includes File, Edit, View, Display, Diagram, Simulation, Analysis, Code, Tools, and Help. The Simulation menu is open, showing options such as Update Diagram (Ctrl+D), Model Configuration Parameters (Ctrl+E), Mode, Data Display, Step back (uninitialized), Run (Ctrl+T), Step Forward, Stop (Ctrl+Shift+T), Output, Stepping Options, and Debug. The main workspace shows a block diagram with a Gain block (value -20) and a Velocity block (transfer function 1/s). The Gain block is labeled "A" and the Velocity block is labeled "V". The status bar at the bottom left shows "Ready" and the bottom right shows "ode3".

Configuration Parameters: CogMod_04B/Configuration (Active)

Select:

- Solver
- Data Import/Export
- ▶ Optimization
- ▶ Diagnostics
- Hardware Implementation
- Model Referencing
- ▶ Simulation Target
- ▶ Code Generation
- ▶ HDL Code Generation

Simulation time

Start time: 0.0

Stop time: 5

Solver options

Type: Fixed-step

Solver: ode3 (Bogacki-Shampine)

Fixed-step size (fundamental sample time):

0.05

Tasking and sample time options

Periodic sample time constraint:

Unconstrained

Tasking mode for periodic sample times:

Auto

Automatically handle rate transition for data transfer

Higher priority value indicates higher task priority

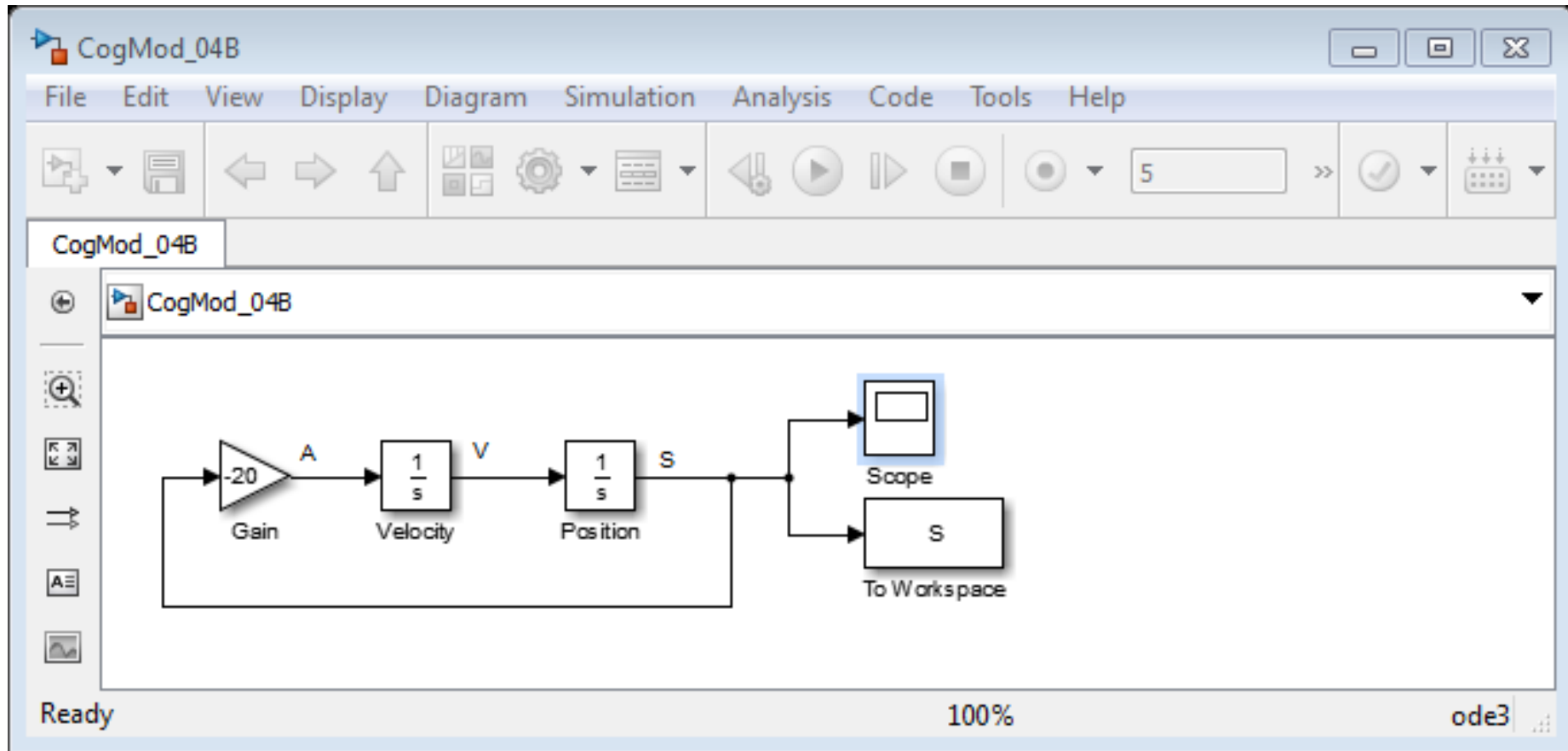
OK

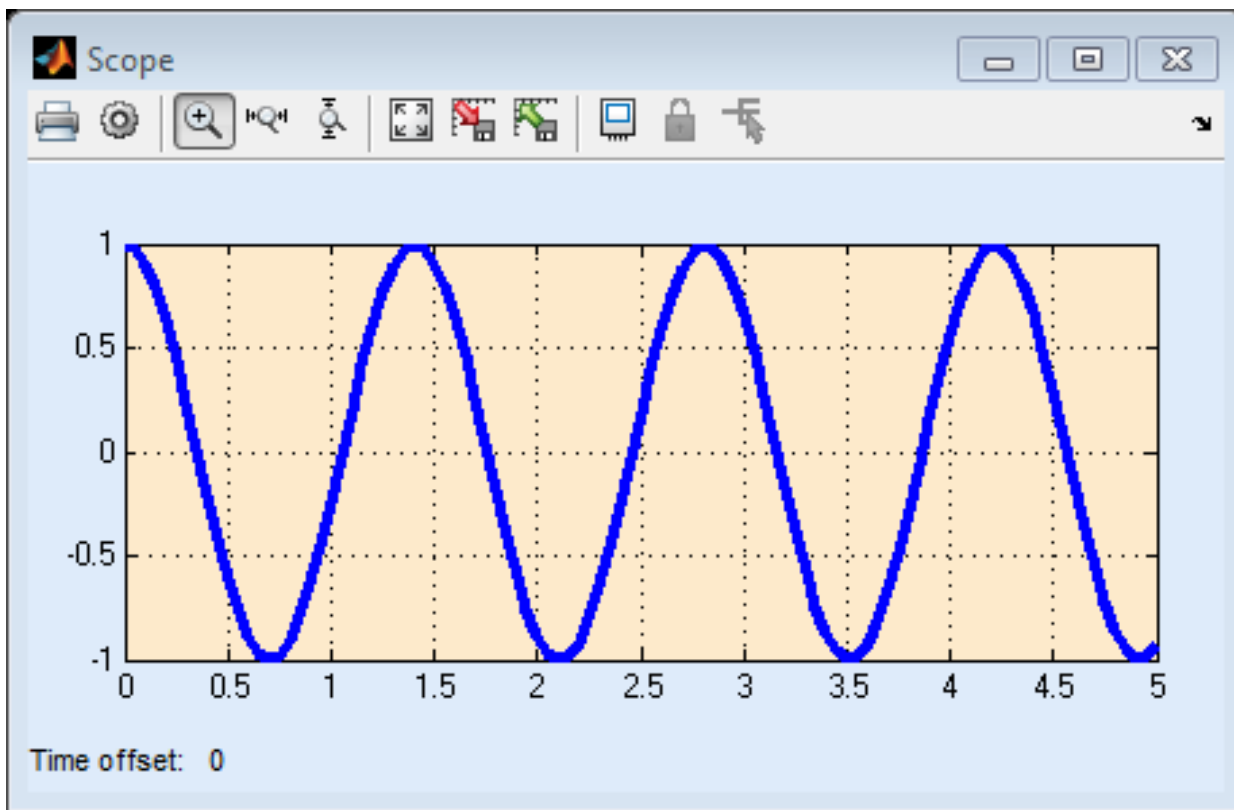
Cancel

Help

Apply

نمایش نتیجه در بلوک اسکوپ





فهرست مطالب

بیش از یک نوع زبان برنامه نویسی کامپیوتر وجود دارد! ✓

برنامه نویسی فرمائی ✓

تمرین هارمونی در پیاپیون توسط حلقه ها ✓

بیشتر بدانیم: برنامه نویسی مثال حرکت فرد متلب ✓

بیشتر بدانیم: برنامه نویسی مثال حرکت فرد سیمولینک ✓

بهترین راه به دست آوردن اعتماد به نفس، انجام کارهایی است که

از آنها می ترسیم



آموزش سخنرانی و فن بیان www.Bahrampoor.com