

MATLAB built-in functions and interfaces

By Alireza Delfarah



Open

To open another application file with MATLAB.

For example to open another m-file (MATLAB function code):

```
open test.m
```

```
% or to open a text/excel file
```

```
open ('C:\Users\delfarah\Desktop\MATLAB\test.txt')
```

```
open ('C:\Users\delfarah\Desktop\MATLAB\test.xlsx')
```

Plot and print

To save figure to specific file format.

```
% create x and y variables
```

```
x = 1:100;
```

```
y = x.^2;
```

```
% plot x vs. y and label axis
```

```
plot(x,y)
```

```
xlabel('x')
```

```
ylabel('y')
```

```
% save plot as png/jpeg/tiff
```

```
print('new plot', '-dpng')
```

```
print('new plot jpeg format', '-djpeg')
```

```
print('new plot jpeg format', '-dtiff')
```

Solve

To solve an equation. Try:

```
syms x
eqn = sin(x) == 1;
solution = solve(eqn,x)

%now try
eqn2 = x^2 - 3*x == 5;
solution2 = solve(eqn2, x)
```

```
%to solve a system of two equations and two unknowns
%2*u^2 + v^2 == 0,
%u - v == 1
```

```
syms u v
eq1 = 2*u^2 + v^2 == 0;
eq2 = u - v == 1;
[solv, solu] = solve([eq1, eq2], [v, u])
```

Fit

Fitting your data to a mathematical equation

For example, try:

```
x = 0.01:0.01:1;
y = x.^2-x.^3;

xlabel('x')
ylabel('y')

f=fit(x',y', 'poly1')
plot(f,x,y)
```

```
%now try
f=fit(x',y','poly2')
plot(f,x,y)
```

```
%now try
f=fit(x',y','poly3')
plot(f,x,y)
```

Arduino interface and package

To connect to and control Arduino inputs and outputs from MATLAB



Capabilities and Features

With MATLAB® Support Package for Arduino® Hardware, you can use MATLAB to interactively communicate with an Arduino board. The package enables you to perform tasks such as:

- Acquire analog and digital sensor data from your Arduino board
- Control other devices with digital and PWM outputs
- Communicate with an Arduino board over a USB cable or wirelessly over Wi-Fi

Because MATLAB is a high level interpreted language, you can see results immediately.

Serial Port Interface

Communication using the serial port interface



Serial communication is the most common low-level protocol for communicating between two or more devices.

Normally, one device is a computer, while the other device can be a modem, a printer, another computer, or a scientific instrument such as an oscilloscope or a function generator.

For many serial port applications, you can communicate with your instrument without detailed knowledge of how the serial port works. Communication is established through a serial port object, `serial`, which you create in the MATLAB® workspace.