

Course unit title:
Basics of
Information
Systems
Course unit code:
NIRIA1SEND



Simulink

Manipulating images

Dr. habil. Levente Kovács
associate professor

Obuda University,
John von Neumann Faculty of Informatics,
Department of Information Systems,
Physiological Controls Group

01.10.2013.



Simulink

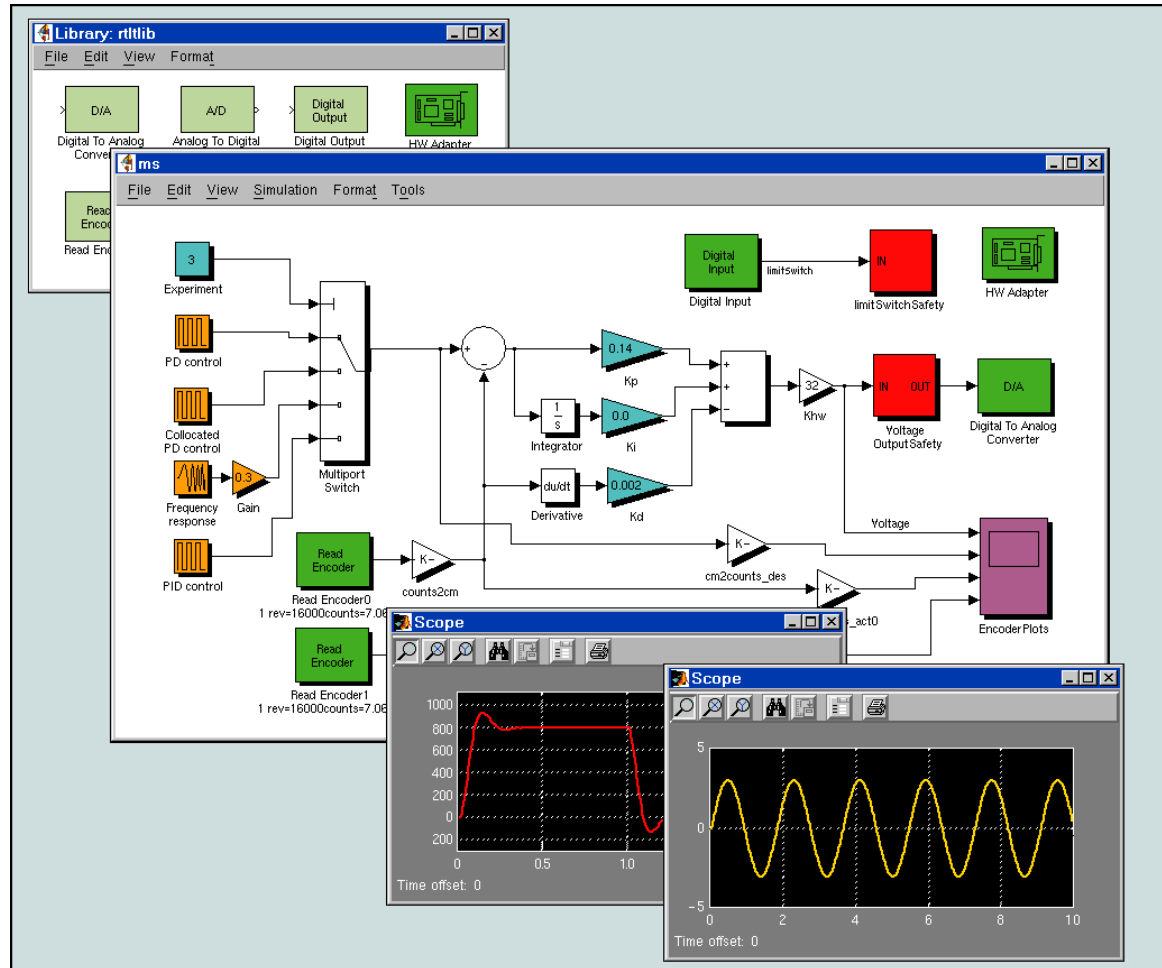


Simulink



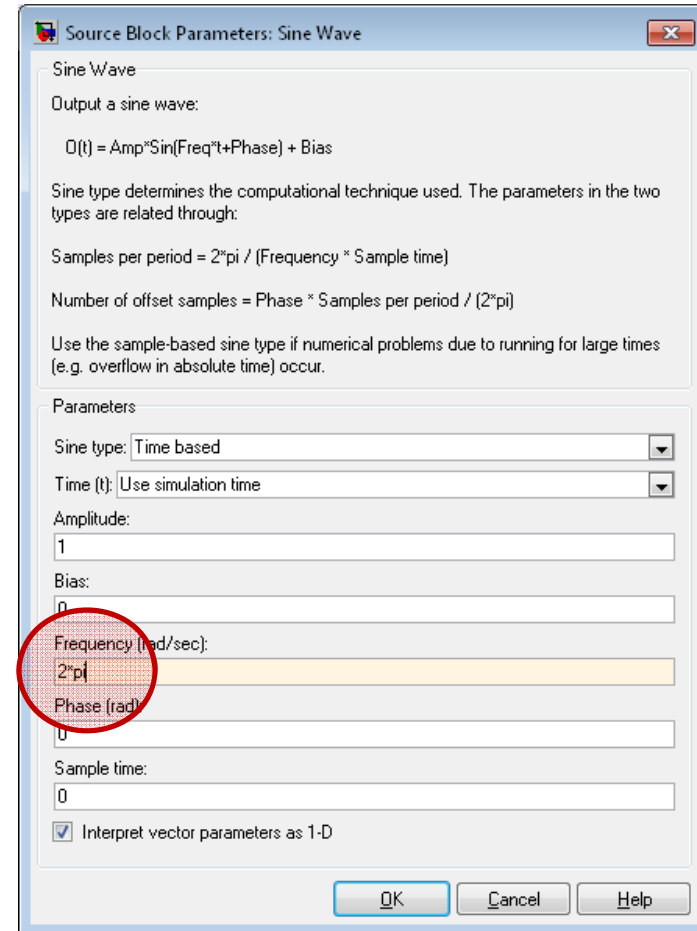
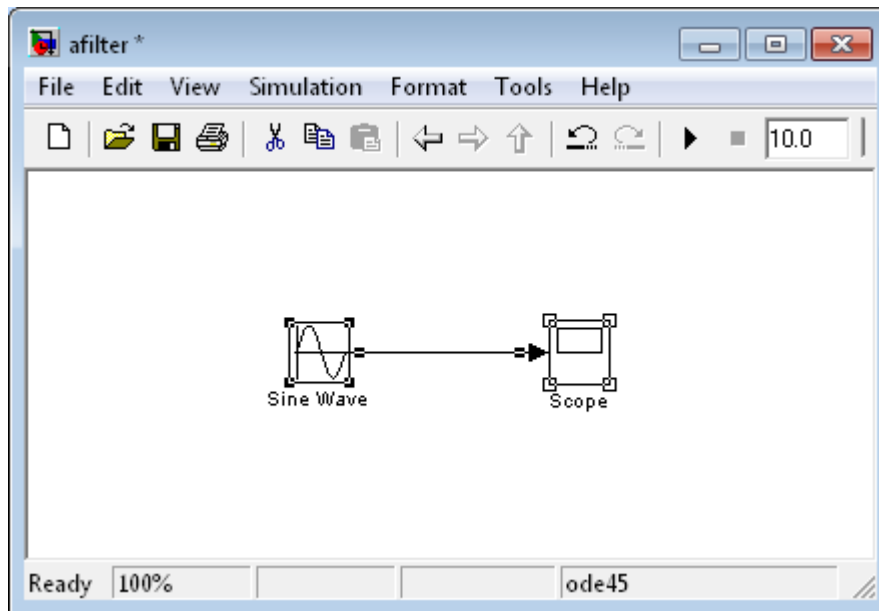
Current directory >
new > model

Simulink library



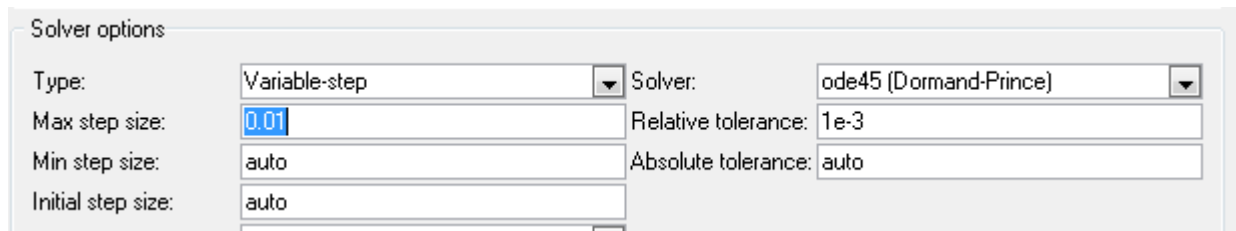
Simulink

Simulink > Sources > Sine wave
Simulink > Sinks > Scope

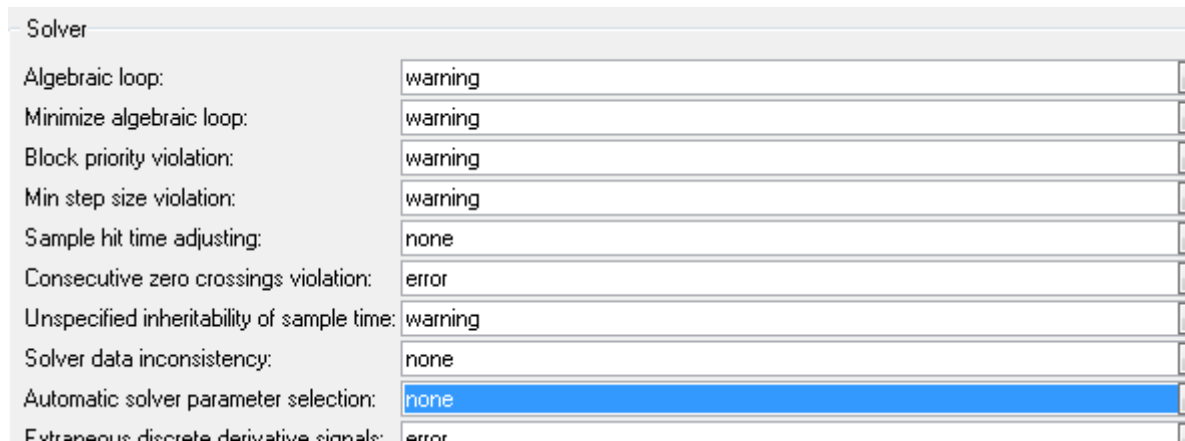


Simulation > Configuration parameters >

- Solver > Max step size: 0.01

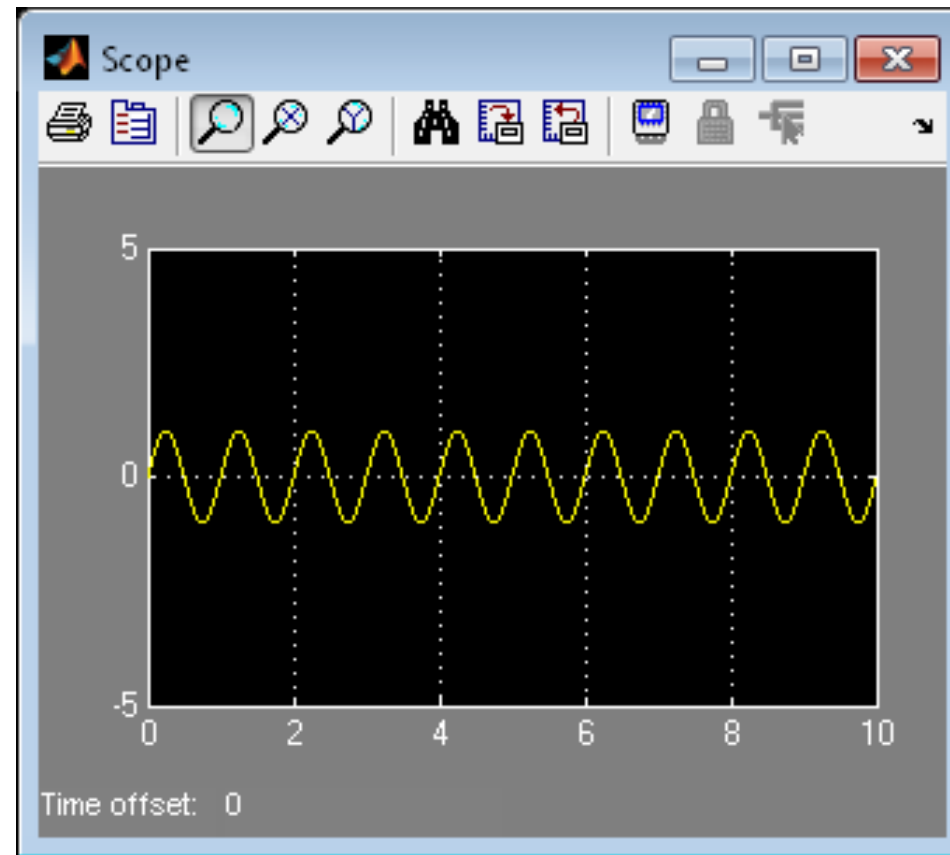


- Diagnostics > Automatic solver parameter selection: none



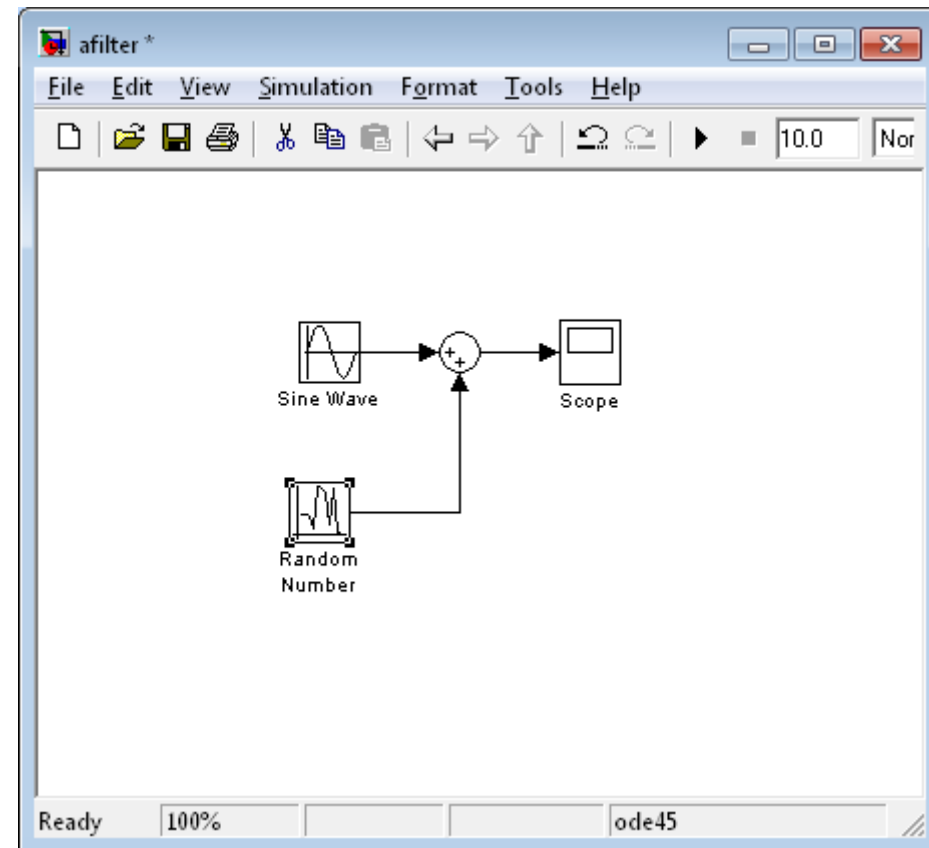
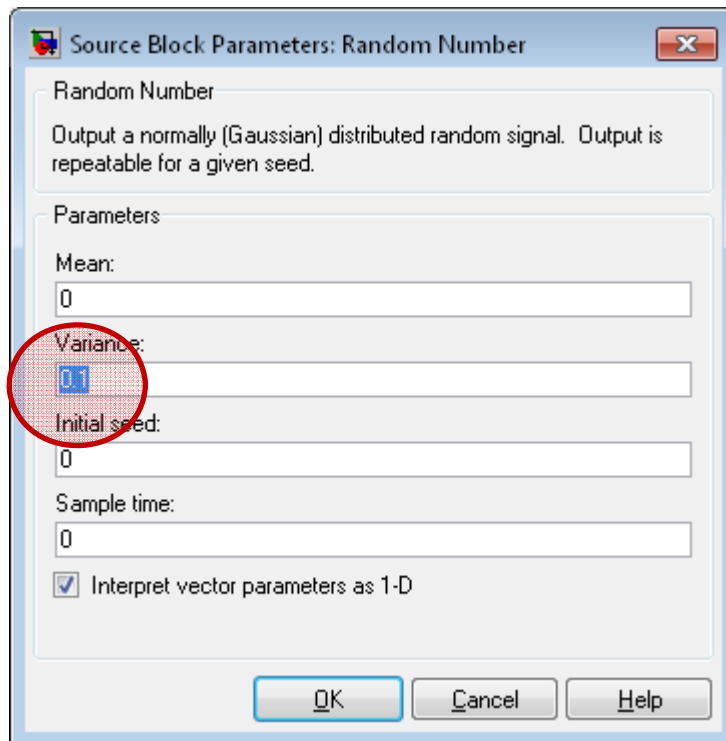
Simulink

Run



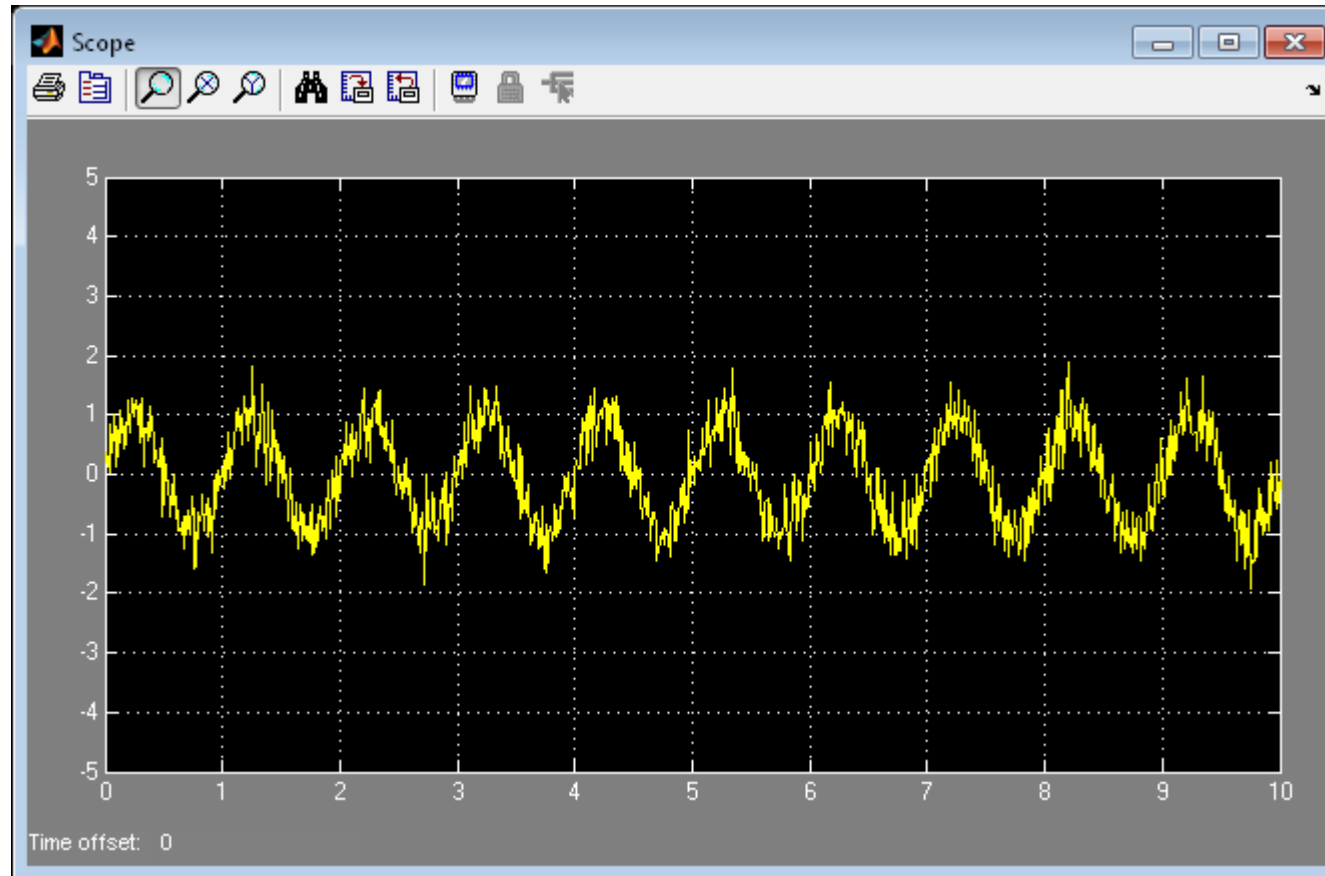
Simulink

Commonly used blocks > sum
Sources > random number



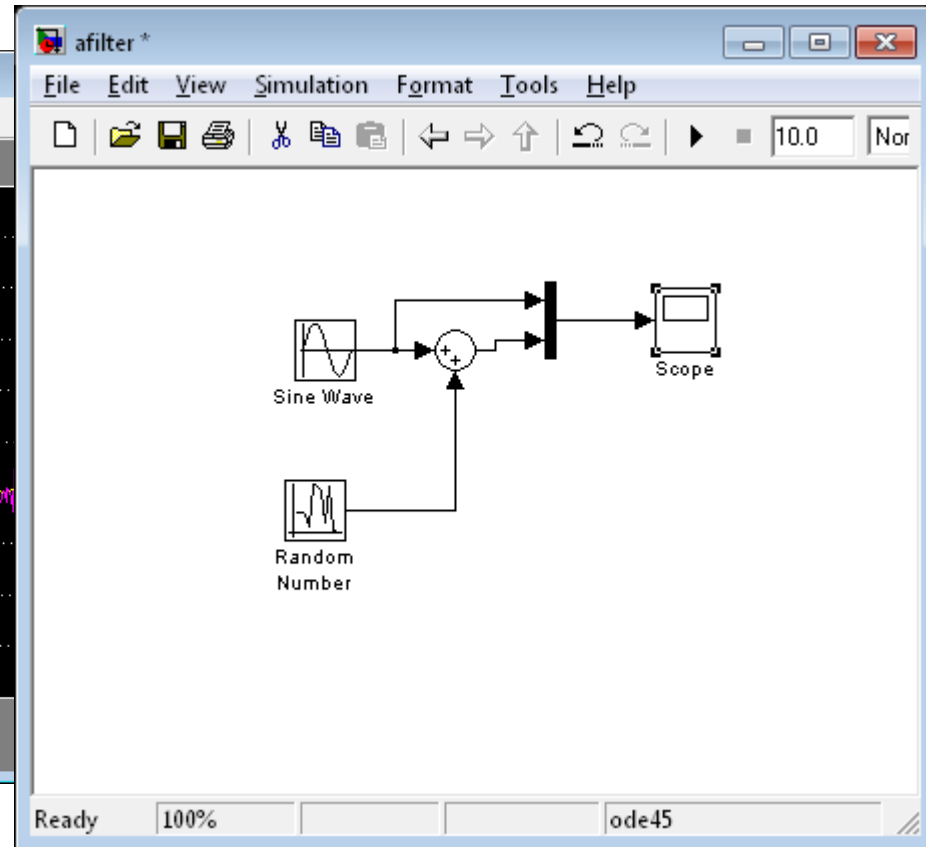
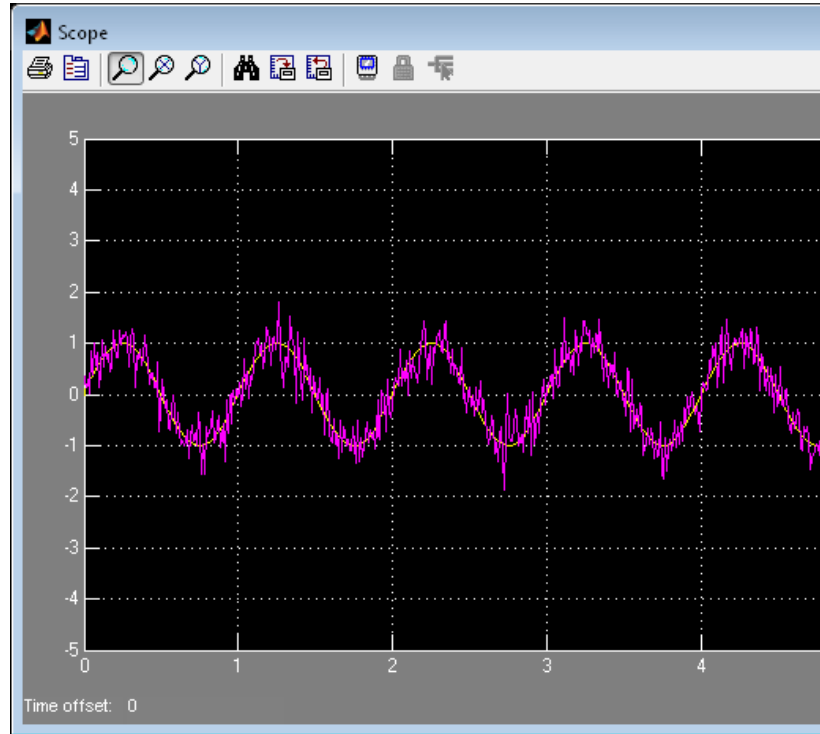
Simulink

Run



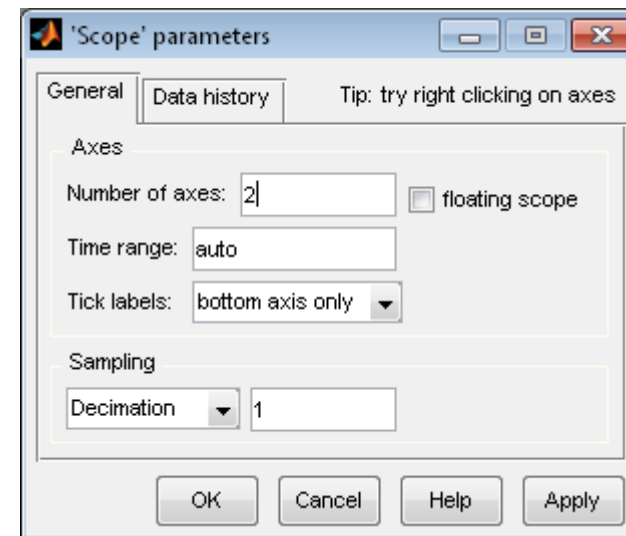
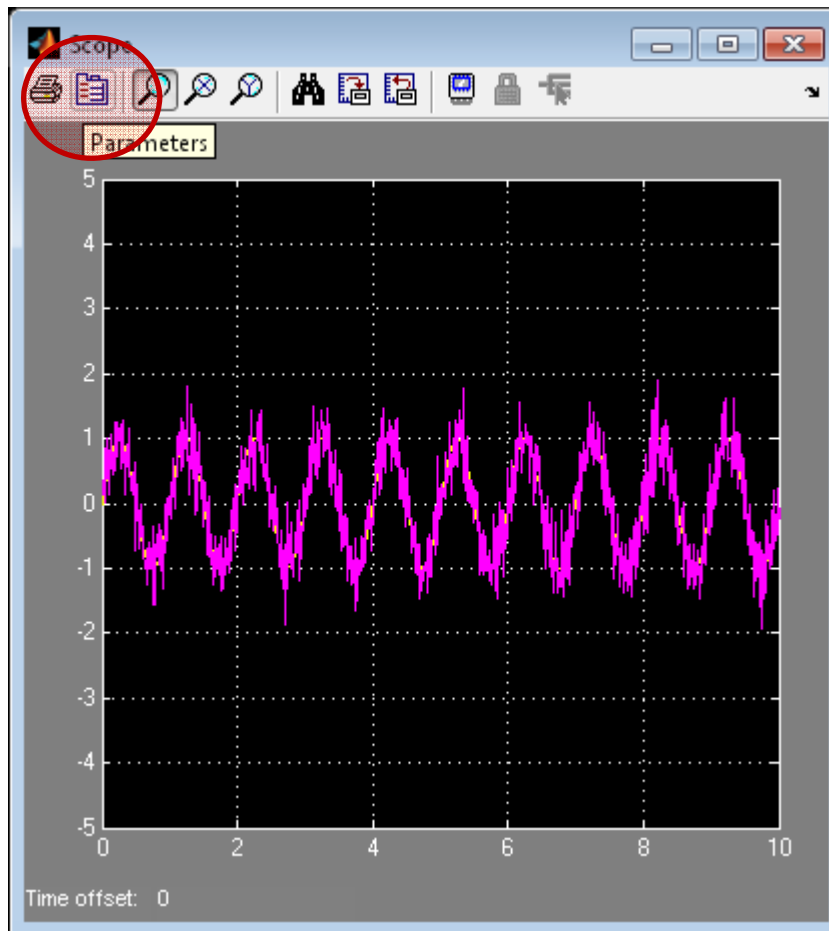
Simulink

Commonly used blocks > Mux



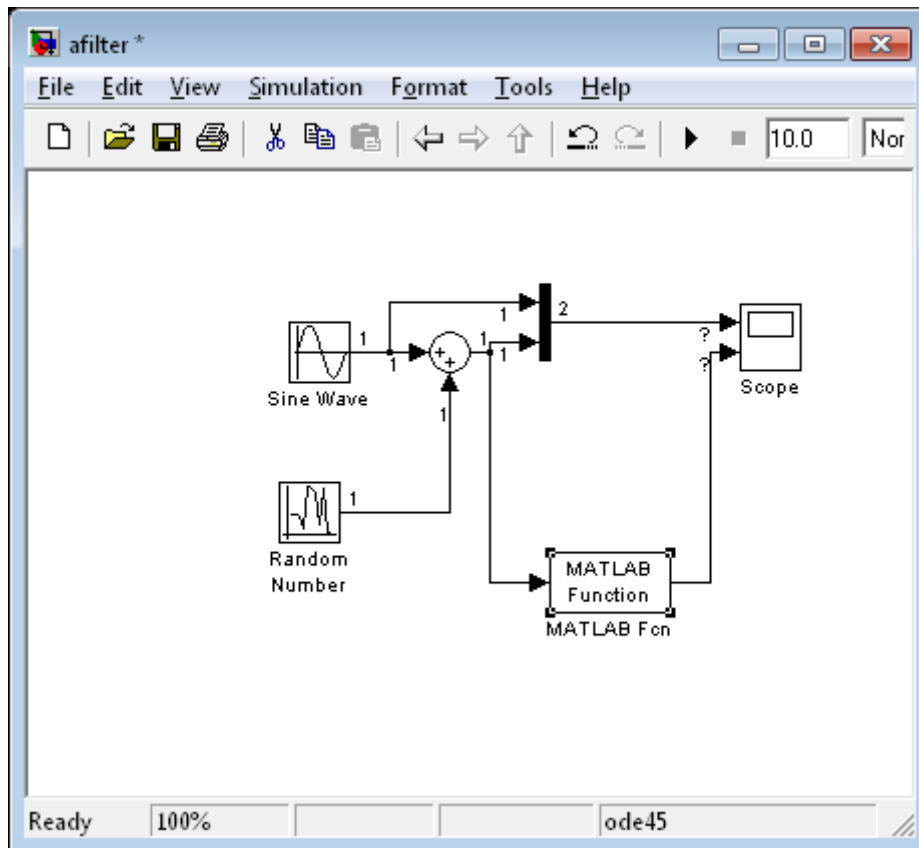
Simulink

Parameters



Simulink

User-defined functions > MATLAB fcn



Function Block Parameters: MATLAB Fcn

MATLAB Fcn

Pass the input values to a MATLAB function for evaluation. The function must return a single value having the dimensions specified by 'Output dimensions' and 'Collapse 2-D results to 1-D'.

Examples: `sin(u)`, `foo(u(1), u(2))`

Parameters

MATLAB function:

Output dimensions:

Output signal type:

Collapse 2-D results to 1-D

Sample time (-1 for inherited):



Simulink

Command window:

```
>> global sw
```

```
>> sw=[0,0,0,0,0]
```

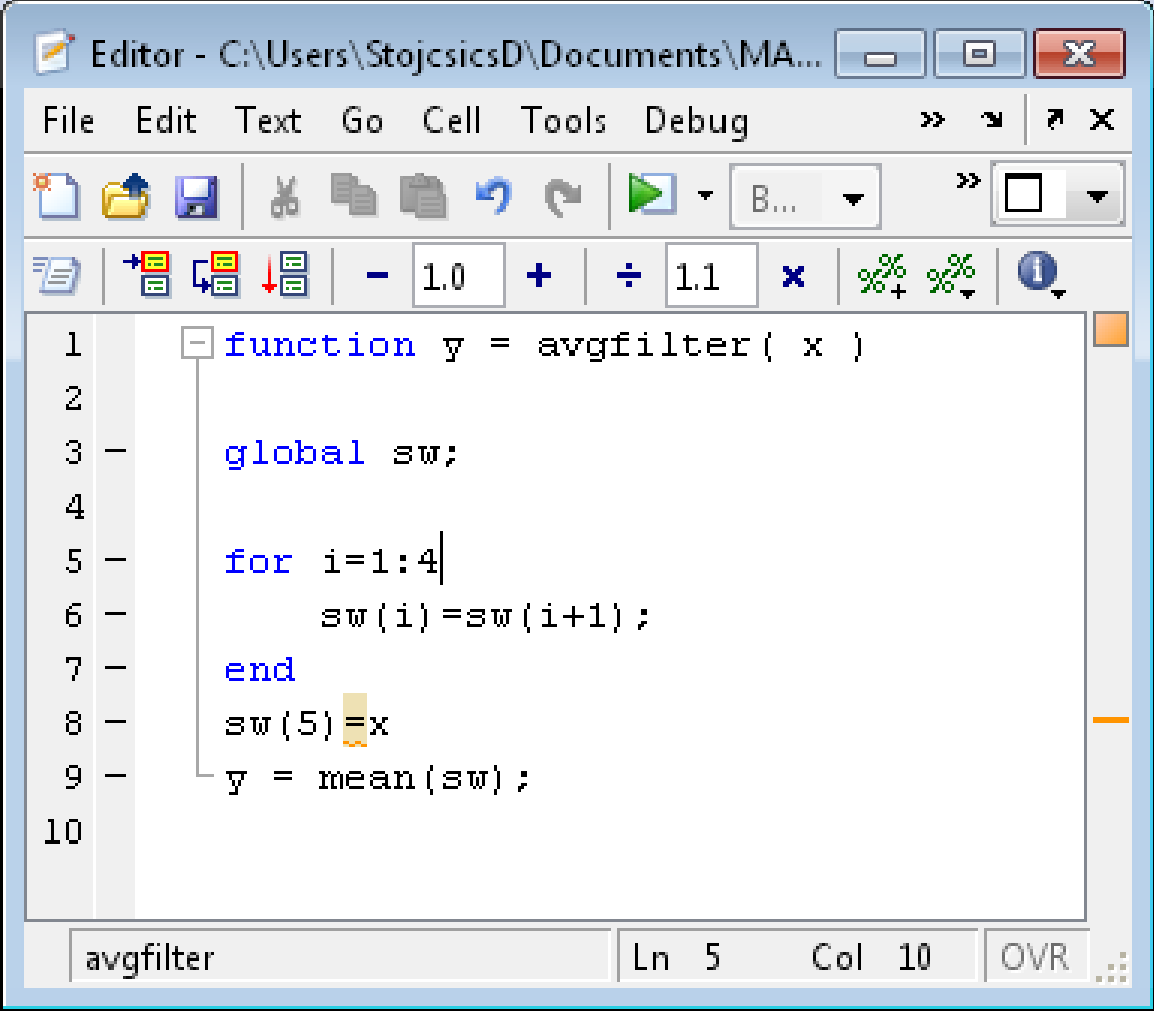
```
sw =
```

```
0 0 0 0 0
```



Simulink

Creat new m-file



```
1 function y = avgfilter( x )
2
3     global sw;
4
5     for i=1:4
6         sw(i)=sw(i+1);
7     end
8     sw(5)=x
9     y = mean(sw);
10
```

The screenshot shows a MATLAB Editor window titled "Editor - C:\Users\StojcsicsD\Documents\MA...". The window contains a function definition for "avgfilter". The code is as follows:

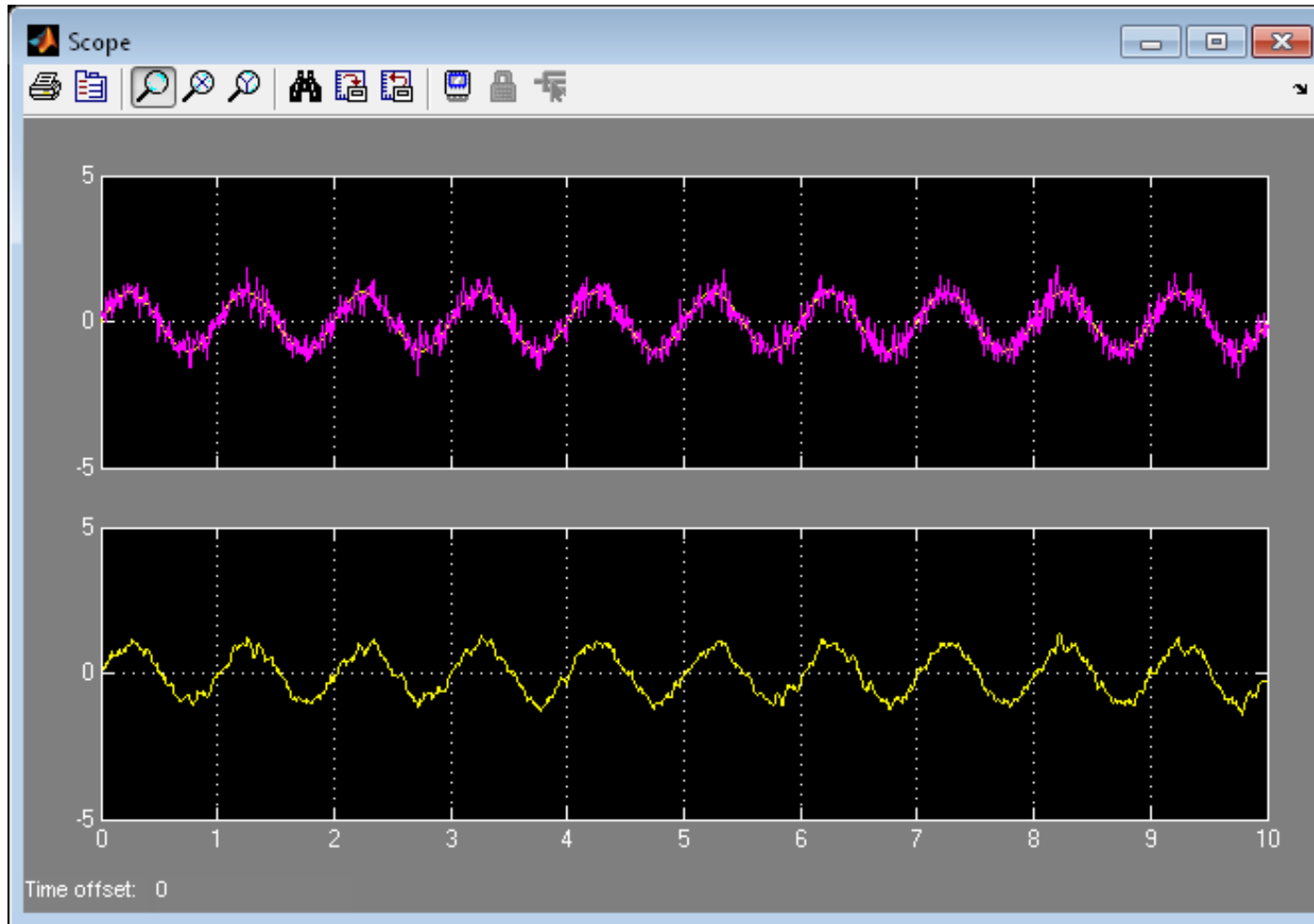
```
function y = avgfilter( x )
    global sw;
    for i=1:4
        sw(i)=sw(i+1);
    end
    sw(5)=x
    y = mean(sw);
```

The status bar at the bottom of the editor shows "avgfilter", "Ln 5", "Col 10", and "OVR".

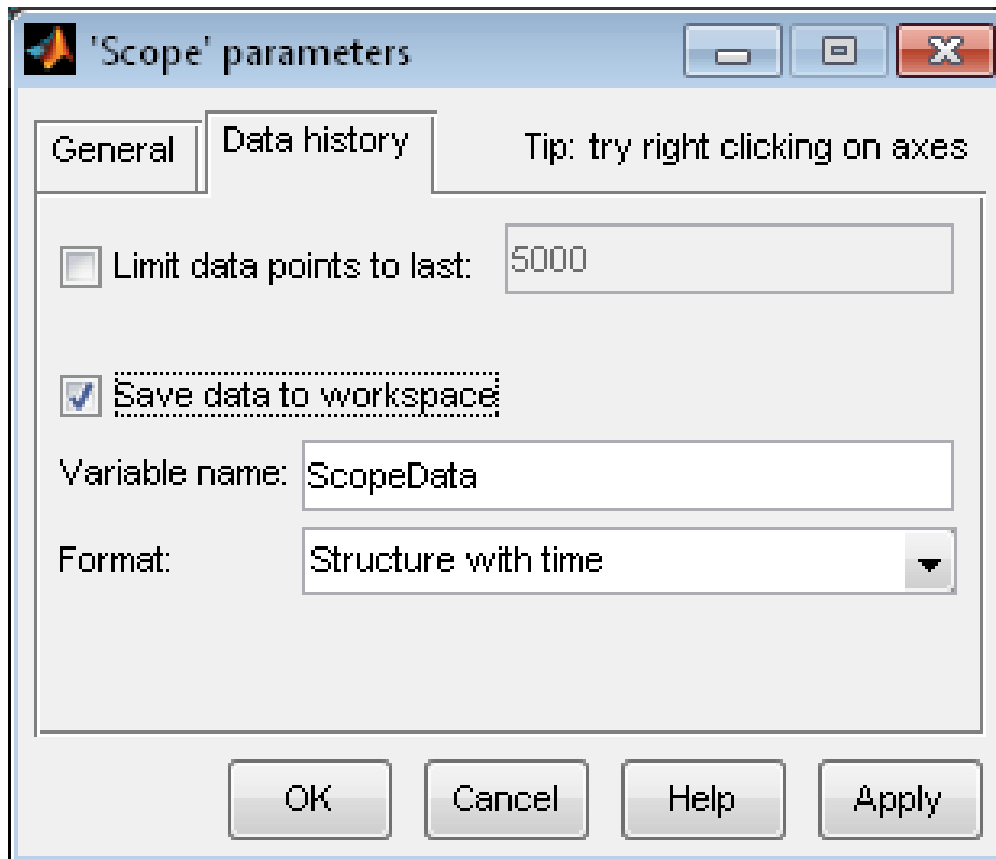


Simulink

Run



Scope parameters



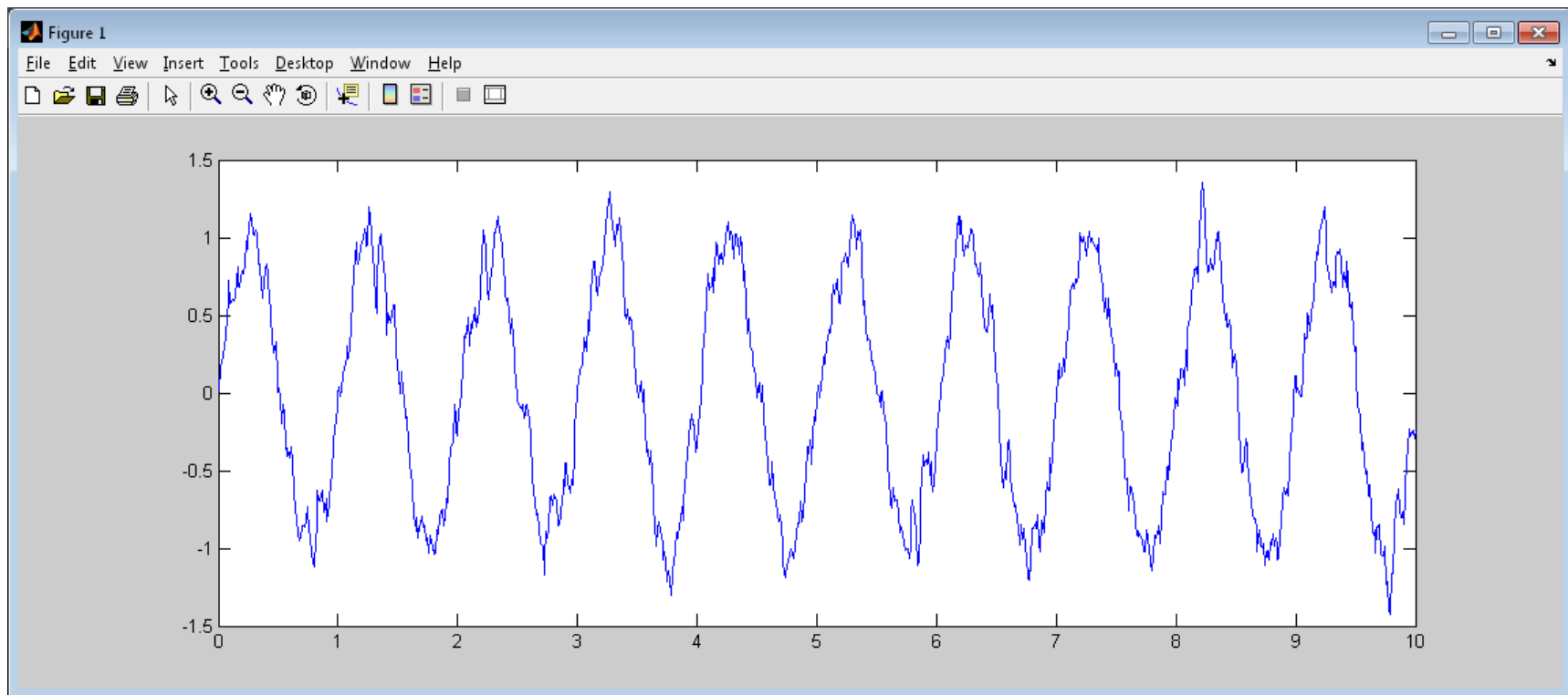
ScopeData.signals(1).values
ScopeData.signals(1).values(:,1)
ScopeData.signals(1).values(:,2)
ScopeData.signals(2).values
ScopeData.time



Simulink

Scope parameters

```
plot(ScopeData.time, ScopeData.signals(2).values)
```



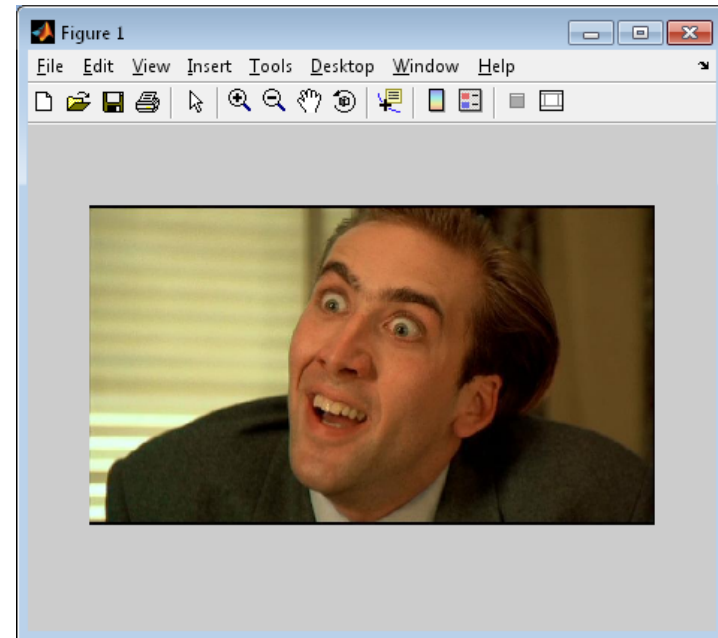
Manipulating images



Reading image from graphics file

```
img=imread( 'clouds.jpg' );  
imshow( img );
```

- the return value *img* is an array containing the image data
 - ✓ if the file contains a grayscale image, *A* is an M-by-N array
 - ✓ if the file contains a truecolor image, *A* is an M-by-N-by-3 array



imfinfo

```
info=imfinfo('YouDontSay.jpg')
info =
    Filename: 'YouDontSay.jpg'
  FileModDate: '20-nov.-2012 13:49:12'
   FileSize: 27702
    Format: 'jpg'
 FormatVersion: ''
     Width: 758
    Height: 429
   BitDepth: 24
   ColorType: 'truecolor'
FormatSignature: ''
NumberOfSamples: 3
   CodingMethod: 'Huffman'
 CodingProcess: 'Sequential'
     Comment: {}
```



Compression ratio

```
>> maxsize=(info.Width*info.Height*info.BitDepth)/8
```

```
maxsize =
```

```
975546
```

```
>> info.FileSize/maxsize
```

```
ans =
```

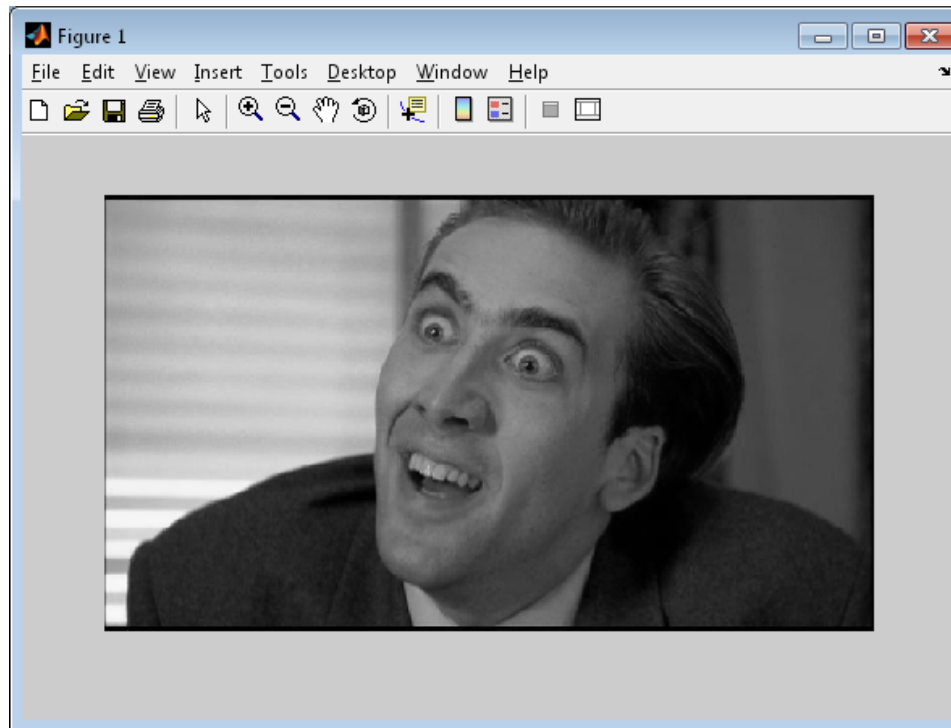
```
0.028396405705113
```



Grayscale image

```
imshow( rgb2gray( img ) ) ;
```

Convert RGB image or colormap to grayscale

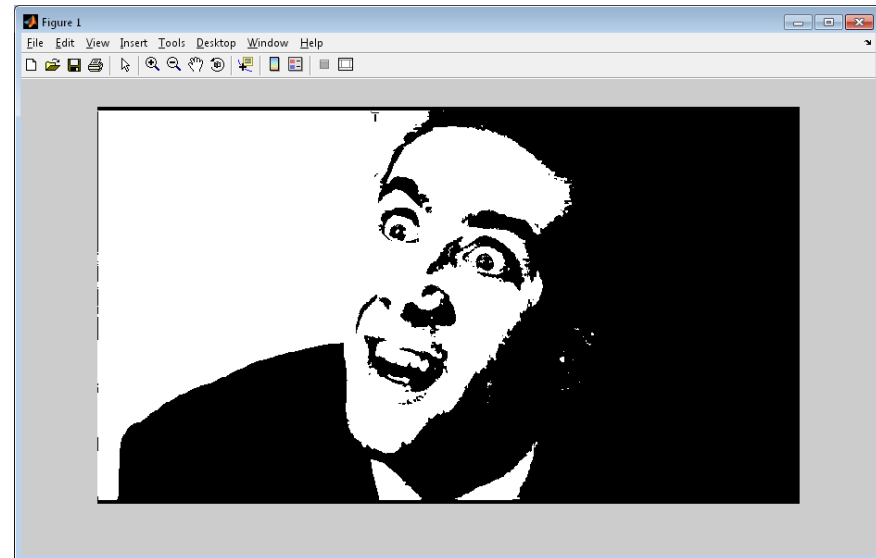


Black and white image

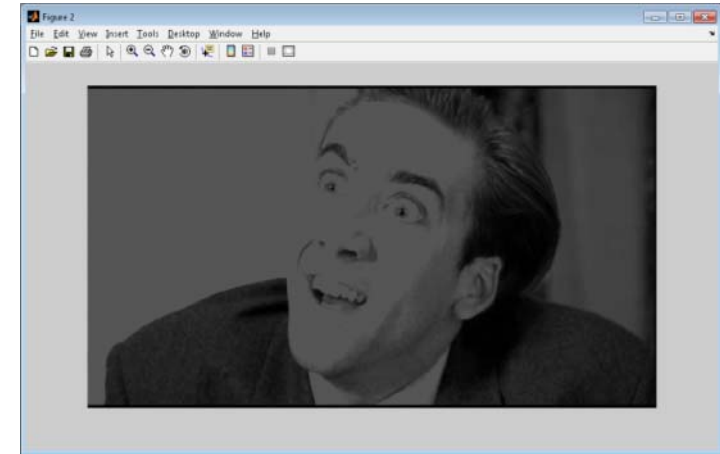
```
imshow(im2bw(img,  
0.35));
```

`BW = IM2BW(RGB,LEVEL)`
converts the RGB
image RGB to black
and white

Note that you specify
LEVEL in the range
[0,1],
regardless of the
class of the input
image.



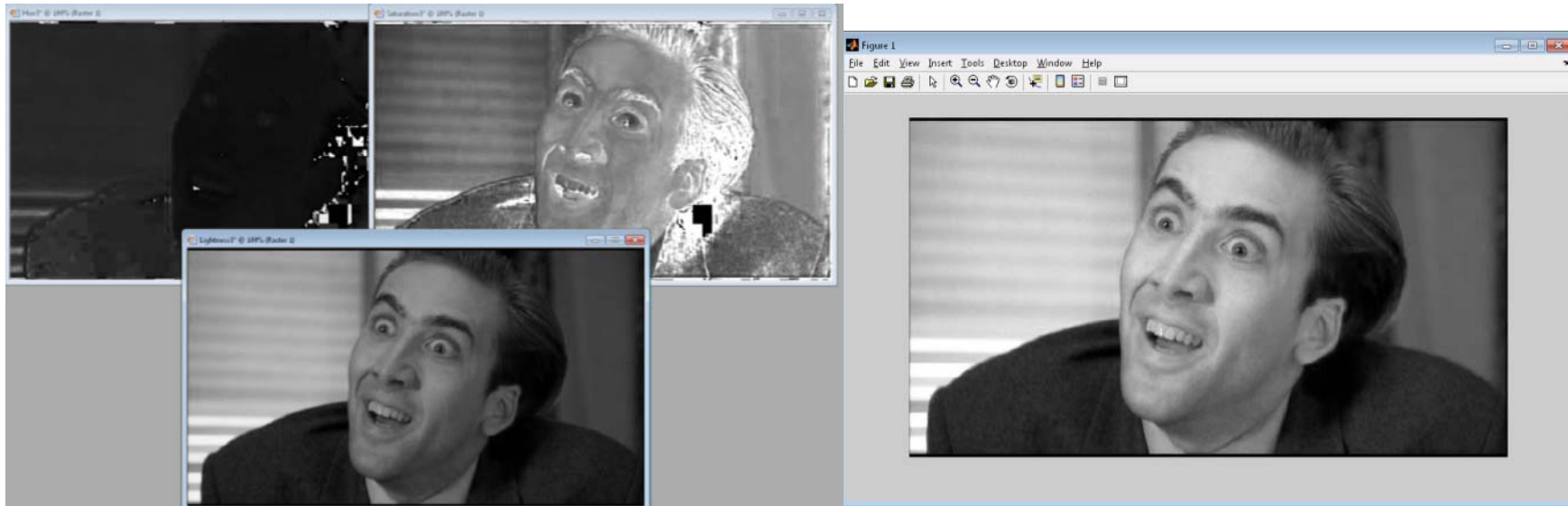
Grayscale image II.



```
img_gray=(img(:,:,1)+img(:,:,2)+img(:,:,3))/3;  
figure  
imshow(img_gray);
```



Grayscale image III.



```
hsv=rgb2hsv(img);           Convert RGB colormap to HSV colormap  
figure()  
imshow(hsv);  
img_gray=hsv(:, :, 3);  
figure()  
imshow(img_gray);
```



Black and white image II.

```
img_bw=img_gray;
threshold_level=70; % in some cases 0.7
for i=1:info.Height
    for j=1:info.Width
        if(img_gray(i,j)>threshold_level)
            img_bw(i,j)=255;
        else
            img_bw(i,j)=0;
        end
    end
end
figure
imshow(img_bw);
```



Course unit title:
Basics of
Information
Systems
Course unit code:
NIRIA1SEND



01.10.2013.

Thank you for your attention!

kovacs.levente@nik.uni-obuda.hu

