```
%-- 07/03/2017 21:08 --%
power_PVarray_grid_det
%-- 08/03/2017 23:02 --%
open_system(docpath(fullfile(docroot, 'toolbox', 'simulink', 'examples',↙
'ex_data_import_test_case_from_workspace.slx')));
var.time = [ 1 ];
var.signals.values = [7];
var.signals.dimensions = 1;
var.time = [ 1 2 ];
var.signals.values = [ 7 3 ];
var.time = [0 1 1 5 5 8 8 10];
var.signals.values = [0 0 2 2 2 3 3 3]';
var.signals.dimensions = 1;
var.time =[];
var.signals.values = [0; 0; 2; 2; 2; 3; 3; 3]';
%-- 09/03/2017 18:19 --%
open_system(docpath(fullfile(docroot, 'toolbox', 'simulink', 'examples',↙
'ex_data_import_test_case_from_workspace.slx')));
\
var.time =[];
var.signals.dimensions = 1;
var.signals.values = [0 0 2 2 2 3 3 3]';
for
for count in 1:20;
for count in 1:20
for count
for count = 1:20
disp(count);
end
for count = 1:20
x = input('Next value: ');
aux = var.signal.values;
var.signal.values = [aux x];
end
3
signal_concat
4
A = [1 2 3 4]
B = [A 5]
signal_concat
1
signal_concat
1
2
3
signal_concat
1
2
3
B = [[] 3.4]
C = [B 4.3]
B = [C 777]
signal_concat
1
signal_concat
1
2
```

```
signal_concat
1
signal_concat
1
2
3
4
open_system(docpath(fullfile(docroot, 'toolbox', 'simulink', 'examples',↙
'ex_data_import_continuous_11b.slx')));
%-- 23/03/2017 14:55 --%
importdata('temperature_matrix.json')
txt = importdata('temperature_matrix.json')
txt(1)
txt(1:3)
strcat(txt(1:3))
txt = importdata('temperature_matrix.json', ',')
txt = importdata('temperature_matrix.json', ' ')
power_PVarray_grid_det
open_system(docpath(fullfile(docroot, 'toolbox', 'simulink', 'examples',↙
'ex_data_import_continuous_11b.slx')));
open_system(docpath(fullfile(docroot, 'toolbox', 'simulink', 'examples',↙
'ex_data_import_test_case_from_workspace.slx')));
signal_concat
;
A = []
signal_concat
var.time = [0 1 1 5 5 8 8 10];
var.signals.values = [0 0 2 2 2 3 3 3]';
importdata('temperature_matrix.txt')
importdata('temperature_matrix.json')
ETHZ
EWMA
build_signal
Temp(1,1:size(Temp,1))
build_signal
var.time = [0 1 1 5 5 8 8 10];
var.signals.values = [0 0 2 2 2 3 3 3]';
build_signal
var.time = [0 1 1 5 5 8 8 10];
var.signals.values = [0 0 2 2 2 3 3 3]';
build_signal
Temp(1,1:size(Temp,1))
Temp(1,1:size(Temp,1))'
build_signal
power_PVarray_grid_det
var
build_signal
%-- 24/03/2017 12:13 --%
power_PVarray_grid_det
%-- 26/03/2017 15:09 --%
importdata('raw_data/irradiation_matrix.json');
ir = importdata('raw_data/irradiation_matrix.json');
temp = importdata('raw_data/temperature_matrix.json');
temp_pred = importdata('raw_data/temperature_pred_matrix.json');
ir_pred = importdata('raw_data/irradiation_pred_matrix.json');
power_PVarray_grid_det
sldemo_varsize_basic
```

```
x = (1:5, 1:5)
x = (1:5;1:5)
x = (1:5)
x = (1:5),1:5
x = (1:5:10)
x = (1:5,3)
x = (1:5;3)
x = [1:5;3]
x = [1:5]
x = [1:5,3,4,5]
x = [1:5;3;4;5]
x = [1:5;:]
x = [1:5;1:5;1:5;1:5;1:5]
y = x*10
y = y -1
y(:,1) = 0
y(:,1:2) = 0
y(:,4:5) = 0
ir(:,1:5) = 1
ir(:,6) = ir(:,6)*2
ir(:,21:24) = 1
ir(:,20) = ir(:,20)*2
ir = (1 - ir)*1000
Ir = ir
Temp = temp
Ir.signals.values = Ir
Ir.signals.values = ir
Ir.signals.dimensions = 1
Ir.time = size(ir,1)
block = signalbuilder([], 'create', [0 5], {[2 2];[0 2]});
channels(1).yData = [1000 750 250 250 1000 1000];
channels(1).yData = [100 750 250 250 1000 1000];
open_system(docpath(fullfile(docroot,'toolbox', 'simulink', 'examples',↙
'ex_data_import_continuous_11b.slx')));
open_system(docpath(fullfile(docroot,'toolbox', 'simulink', 'examples',↙
'ex_data_import_test_case_from_workspace.slx')));
var.time=[1;2;3;4]
var.signals.values=[2;2;3;4]
var.signals.dimensions=1
var.time=[1;2;3;4;10]
var.signals.values=[2;2;3;4;5]
Ir.signals.values = ir(1,:)'
Ir.time = (1:24)'
Temp.signals.time = (1:24)'
Temp.signals.dimensions = 1
Temp.signals.values = temp(1,:)'
Temp.time = (1:24)'
Ir.time = [0:2.5]'
Ir.time = ([0:25]*0.1)'
Ir.time = ([0:25]*2.5/24)'
Ir.time = ([0:23]*2.5/24)'
Temp.time = ([0:23]*2.5/24)'
Temp.signals.values =  Temp.signals.values+20
Temp = Ir
Temp.signals.values = temp(1,:)'
Ppv(1:1000:24000)
Ppv(1:1000:24000,1)
```

```
Ppv()
Ppv(){1}
Ppv.signals.values(1:1000:24000)
Es=Ppv.signals.values(1:1000:24000)
ETHZ
temp(:)
Ir.signal.values = ir(:)
Ir.signals.values = ir(:)
ir_tras = ir'
Ir.signals.values = ir(:)
Ir.signals.values = ir_tras(:)
open_system(docpath(fullfile(docroot, 'toolbox', 'simulink', 'examples', ↙
'ex_data_import_test_case_from_workspace.slx')));
var
var.signals.dimensions = [3,3]
var.signals.values = [1,2,3;4,5,6;7,8,9]
var.time = 9
var.signals.dimensions = 1
var.signals.dimensions = 3
var.time = 3
var.time = [1,2,3;4,5,6;7,8,9]
magic(10)
t1 = 0.2 * [0:49]';
m = magic(10);
M = repmat(m,[1 1 length(t1)]);
var.time=t1;
var.signals.values = M;
var.signals.dimensions=[10 10];
m(1,:) = 1:10
ir_tras
Ir.time = size(Ir.signals.values, 1)
Ir.time = 1:(2.5/size(Ir.signals.values, 1)):2.5
Ir.time = Ir.time'
time2w = 1:336
2.5/336
time2w = time2w/0.0074
time2w = time2w/2
Ir.time = (1:336)*(2.5/336)
Ir.time = Ir.time'
Ir.time = ((1:336)*(2.5/336))'
Ir.time = ((1:336)*(14/336))'
Ir.time = ((1:336)*(15/336))'
Temp.signals.values = temp(:)'
temp_tras = temp'
Temp.signals.values = temp(:)'
Temp.signals.values = temp(:)
Temp.time = ((1:336)*(15/336))'
ir_file = importdata('raw_data/irradiation_matrix.json');
ir_file = ir_file'
ir_file = (1-ir_file)*1000
ir_file(1:5,:) = 0
ir_file(21:24,:) = 0
ir_file(6,:) = ir_file/2
ir_file(6,:) = ir_file(6,:)/2
ir_file(20,:) = ir_file(20,:)/2
Ir.signals.values = ir_file(:)
Ir.signals.dimensions = 1
```

```
Ir.time = 1:336 * (15/336)
Ir.time = (1:336)' * (15/336)
temp_file = importdata('raw_data/temperature_matrix.json');
temp_file=temp_file'
Temp.signals.values = temp_file(:)
Temp.values.dimensions = 1
Temp.time = (1:336)' * (15/336)
Ts
Ts;
Ts = 0
Ts_Control = 00
Ts_Power = 0
Sample_Time = 0
power_PVarray_grid_det
%-- 26/03/2017 19:55 --%
signal_builder
power_PVarray_grid_det
simulink
signal_builder
ETHZ
Untitled3
export
Untitled3
ETHZ
Ppv.signals.values(1:1000:)
Ppv.signals.values(1:1000:end)
size(Ppv.signals.values(1:1000:end),1)
Ppv.signals.values(day_ini:day_ini+24)
day_ini = 24*(days-1);
days=1
day_ini = 24*(days-1);
day_ini = 24*(days-1)
Ppv.signals.values(day_ini:day_ini+24)
Ppv.signals.values(day_ini+1:day_ini+24)
power = Ppv.signals.values(1:1000:end)
power = Ppv.signals.values(1:100:end)
dayPower = power(1:4:99)
dayPower = power(1:4:96)
dayPower = power(1:4:99)
dayPower = power(100:4:199)
plot(power, 1:1500)
plot(power, 1:size(power))
plot(power', 1:size(power))
plot(1:size(power), power)
day_ini = 24*(days-1)*10;
day_ini = 24*(days-1)*10
days=2
day_ini = 24*(days-1)*10
day_ini = 24*(days-1)*100
days=14
day_ini = 24*(days-1)*100
day_ini = (days-1)*100
day_ini = (days-1)*1000
day_ini = days*10000
power = Ppv.signals.value(day_ini:400:day_ini+10000)
power = Ppv.signals.values(day_ini:400:day_ini+10000)
plot(1:size(power), power)
```

```
power = Ppv.signals.values(day_ini:416:day_ini+10000)
power = Ppv.signals.values(day_ini:417:day_ini+10000)
plot(1:size(power), power)
for days = 1:14
day_ini = days*10000;
Tamb(days) = Energy.signals.values(day_ini:417:day_ini+10000);
end
for days = 1:14
day_ini = days*10000;
Tamb(days) = Ppv.signals.values(day_ini:417:day_ini+10000);
end
Tamb(1) = 1:5
Tamb(1,:) = 1:5
Tamb(2,:) = 5:10
Tamb(2,:) = 6:10
ETHZ
Pred = ETHZ
size(Pred)
plot(1:size(Pred,1)*size(Pred,2), Pred(:))
X = 1:size(Pred,1)*size(Pred,2)
signal_builder
Signal_compressor
plot(X, Pred(:))
plot(X, Pred(:),X,TheRealpred(:))
error = Pred - TheRealpred;
plot(X, error(:))
%-- 27/03/2017 12:12 --%
WCMA
EWMA
Ppv = importdata('data_formatted/Ppv.txt');
plot(1:size(Ppv,1),Ppv)
plot(1:size(Ppv,1),Ppv(1:100:end))
plot(1:size(Ppv,1)/100,Ppv(1:100:end))
Ppv_short = Ppv(1:100:end);
plot(1:size(Ppv_short,1), Ppv_short)
Ppv_short = Ppv(1:1000:end);
plot(1:size(Ppv_short,1), Ppv_short)
Ppv_short
Ppv
Ppv = importdata('data_formatted/Ppv.txt');
Ppv_short = Ppv(1:1000:end);
plot(1:size(Ppv_short,1), Ppv_short)
EWMA
error = Eewma - Tamb.data
1000/24
ceil(1000/24)
ceil(10000/24)
ETHZ
ethz_pred = ETHZ
plot(1:336, ethz_pred(:))
load_power_to_workspace
signal_builder
power_PVarray_grid
power_PVarray_grid_det
ETHZ
pred_ethz = ETHZ;
load_power_to_workspace
```

```
E = load_power_to_workspace
load_power_to_workspace
E = load_power_to_workspace;
plot(1:336, E(:));
pred_ethz = ETHZ;
error = E - pred_ethz;
plot(1:336, error(:));
%-- 27/03/2017 18:39 --%
signal_builder
%-- 28/03/2017 06:47 --%
pred_ethz = ETHZ();
ETHZ
load_power_to_workspace
E = load_power_to_workspace();
ETHZ
pred_ethz = ETHZ();
plot(1:336, pred_ethz(:))
```