# Kalman Filtering Example

```
clear all

% Set default text interpreter for plots
set(0, 'DefaultTextInterpreter', 'none')
set(0, 'DefaultLegendInterpreter', 'none')
set(0, 'DefaultAxesTickLabelInterpreter', 'none')
```

## Define a process model

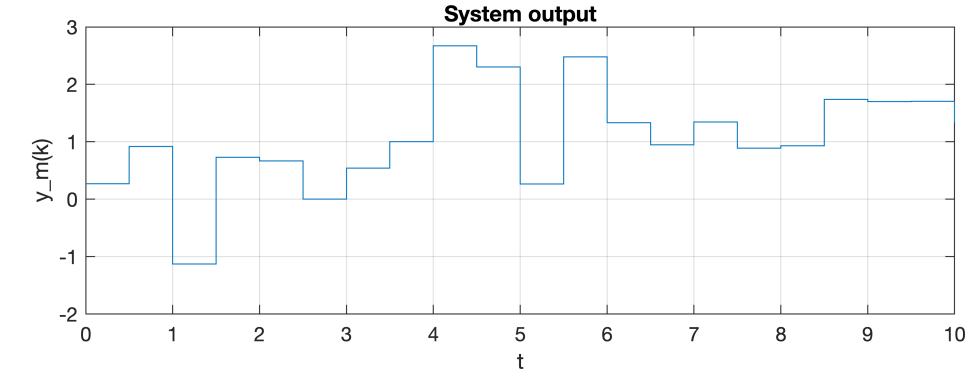
```
% Sampling period
Ts = 0.5;

% Discrete-time transfer function
Gpd = tf(0.3, [1 -0.7], Ts);
```

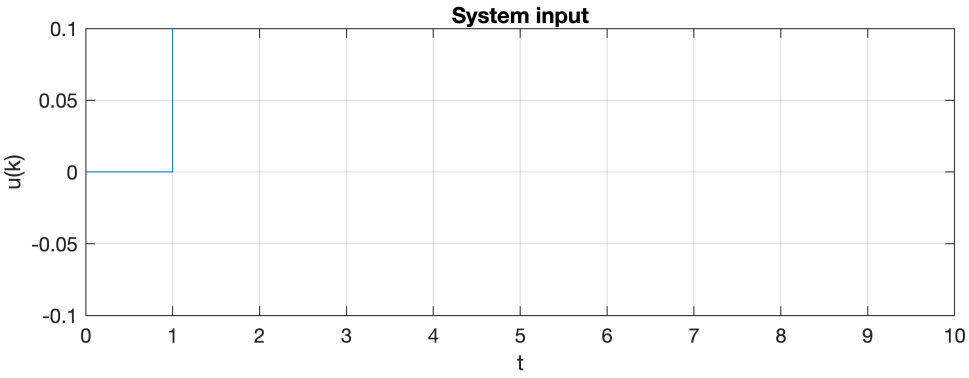
## Generate some data by simulating the system

```
% Number of sample periods
nT = 20;

% Time vector
t = Ts*(0:nT)';

% Input signal
u = zeros(nT+1,1);
u(t>=1) = 1;

[y, t] = lsim(Gpd,u,t);

% Add measurement noise
rng(0);
v = 0.5*randn(nT+1,1);
y_m = y + v;
```

## Plot simulation results

```
figure(1); clf
subplot(2,1,1)
stairs(t,y_m)
xlabel('t')
ylabel('y_m(k)')
title('System output')
grid on
subplot(2,1,2)
stairs(t,u)
ylim([-0.1 0.1])
xlabel('t')
ylabel('u(k)')
title('System input')
grid on
```




## Define observer

```
% Process model for use by observer
A = 0.7;
B = 1;
C = 0.3;
D = 0;

% Kalman filter parameters
P0 = 1;  % initial variance of the estimate
Q = 0.01;  % estimated process noise variance
R = 0.5^2;  % measurement noise variance
KF1 = KalmanFilter(A,B,C,Ts,P0,Q,R,'KF1');
```

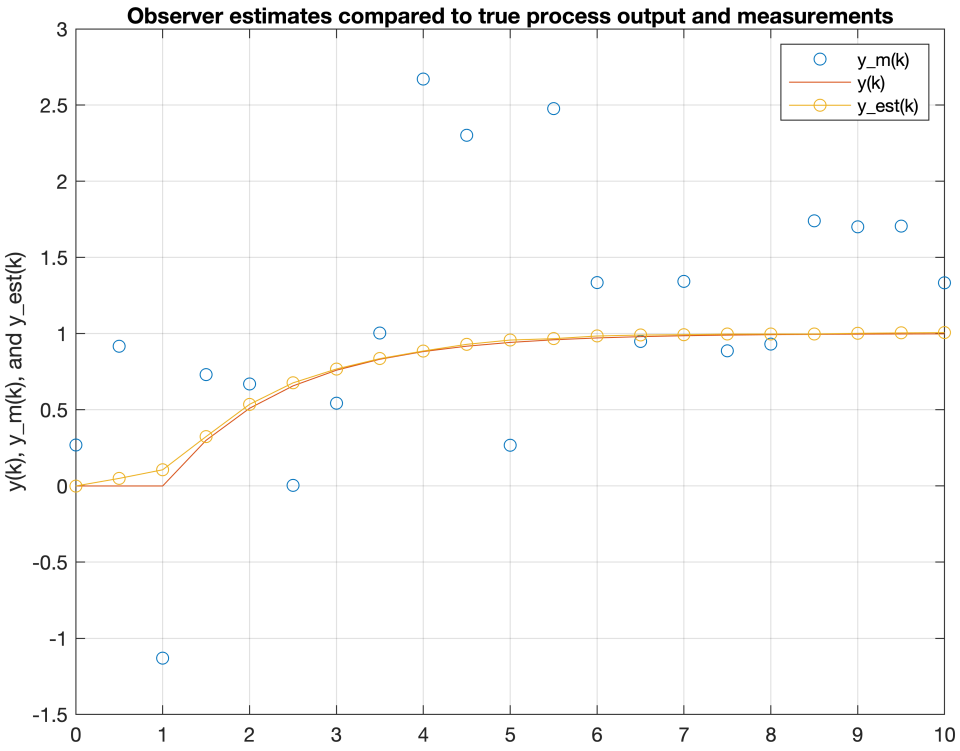
### Simulate observer

```
% Array to store simulation results
y_est = nan(nT+1,1);
obs = KF1;
% Initial estimate (at t=0)
y_est(1,:) = obs.ykp1_est;
for i = 1:nT

    % update observer
    obs.update(y_m(i), u(i));

    % get estimate of output at next sample time
    y_est(i+1,:) = obs.ykp1_est;
```

```
end
```

## Compare observer estimates to plant data

```
figure(2); clf
plot(t,y_m,'o',t,y,'-',t,y_est,'o-')
grid on
ylabel('y(k), y_m(k), and y_est(k)')
legend('y_m(k)','y(k)','y_est(k)')
title("Observer estimates compared to true process output and measurements")
```




```
% Display simulation results
table(t,u,y,v,y_m,y_est)
```

ans = 21×6 table

| | t | u | y | v | y_m | y_est |
|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0.2688 | 0.2688 | 0 |
| 2 | 0.5000 | 0 | 0 | 0.9169 | 0.9169 | 0.0498 |
| 3 | 1 | 1 | 0 | -1.1294 | -1.1294 | 0.1063 |
| 4 | 1.5000 | 1 | 0.3000 | 0.4311 | 0.7311 | 0.3245 |
| 5 | 2 | 1 | 0.5100 | 0.1594 | 0.6694 | 0.5359 |
| 6 | 2.5000 | 1 | 0.6570 | -0.6538 | 0.0032 | 0.6769 |
| 7 | 3 | 1 | 0.7599 | -0.2168 | 0.5431 | 0.7679 |

| | t | u | y | v | y_m | y_est |
|---|---|---|---|---|---|---|
| 8 | 3.5000 | 1 | 0.8319 | 0.1713 | 1.0032 | 0.8360 |
| 9 | 4 | 1 | 0.8824 | 1.7892 | 2.6715 | 0.8862 |
| 10 | 4.5000 | 1 | 0.9176 | 1.3847 | 2.3024 | 0.9298 |
| 11 | 5 | 1 | 0.9424 | -0.6749 | 0.2674 | 0.9578 |
| 12 | 5.5000 | 1 | 0.9596 | 1.5175 | 2.4771 | 0.9671 |
| 13 | 6 | 1 | 0.9718 | 0.3627 | 1.3345 | 0.9844 |
| 14 | 6.5000 | 1 | 0.9802 | -0.0315 | 0.9487 | 0.9908 |
| 15 | 7 | 1 | 0.9862 | 0.3574 | 1.3435 | 0.9933 |
| 16 | 7.5000 | 1 | 0.9903 | -0.1025 | 0.8878 | 0.9970 |
| 17 | 8 | 1 | 0.9932 | -0.0621 | 0.9311 | 0.9974 |
| 18 | 8.5000 | 1 | 0.9953 | 0.7448 | 1.7401 | 0.9979 |
| 19 | 9 | 1 | 0.9967 | 0.7045 | 1.7012 | 1.0021 |
| 20 | 9.5000 | 1 | 0.9977 | 0.7086 | 1.7063 | 1.0049 |
| 21 | 10 | 1 | 0.9984 | 0.3357 | 1.3341 | 1.0068 |