02619 Model Predictive Control
Autumn 2016

Model Predictive Control

Lecture #1

Cyber-Physical Systems

MPC & Computer Controlled Systems

Simulation

Quadruple Tank Process

Learning Objectives

- Lecture #1 will enable you to
  - Describe the components in a computer controlled system.
  - Identify, describe and analyze a control structure in terms of CVs, MVs and DVs.
  - Model and simulate a process system consisting of differential equations
  - Simulate a stochastic system
  - Simulate a deterministic/stochastic systems with digital PI-controllers in the loop.

Literature / Reading List

- Rawlings (2000): ”Tutorial Overview of Model Predictive Control”
- Qin & Badgwell (2003): ”A survey of industrial model predictive control technology”
- Bauer & Craig (2008): ”Economic Assessment of Advanced Process Control – A Survey and Framework”

- Maciejowski (chap 1 – skip technical details)

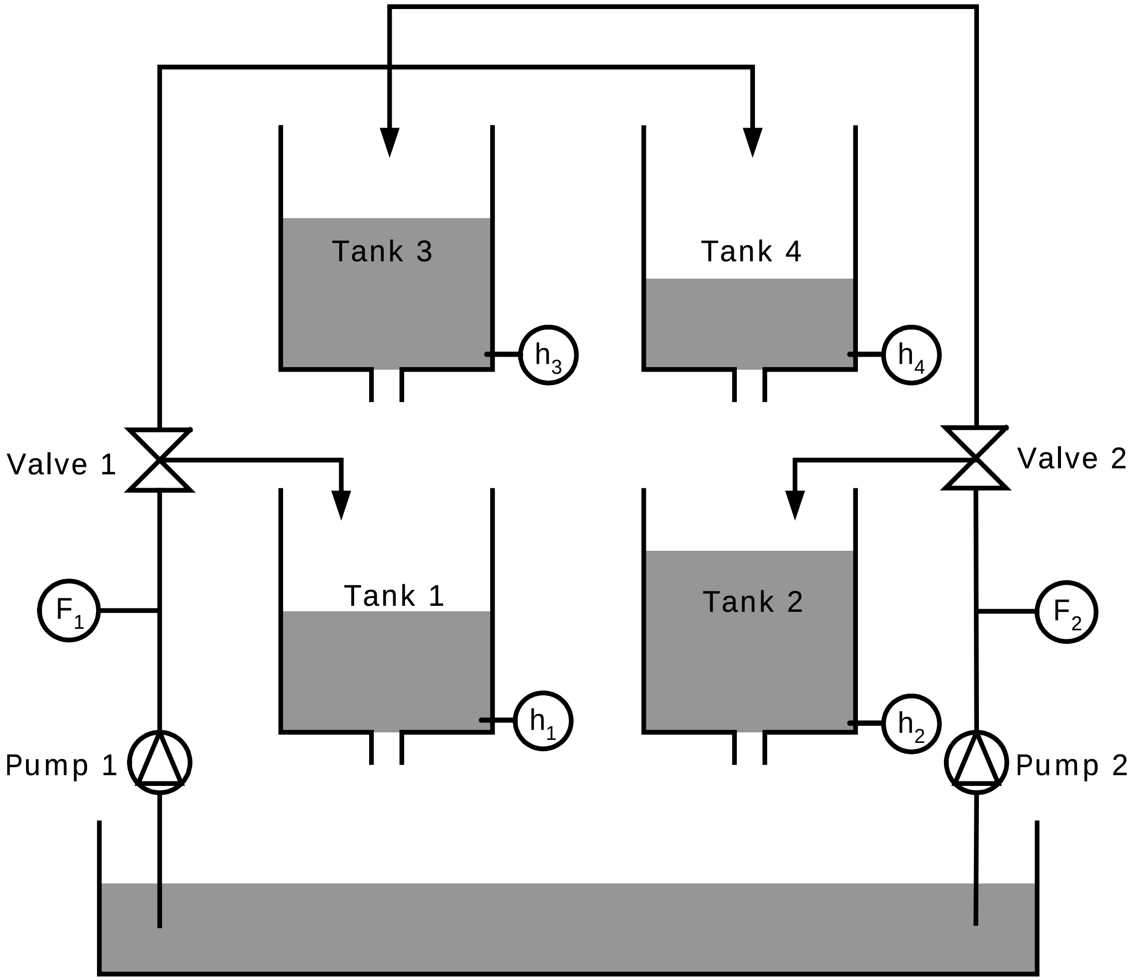
- Jørgensen – Chapter: The Quadruple Tank Process
  – we will continue with this material in Lecture #2
- Henriksson (2000): ”The quadruple tank process ...”

Computer Controlled Systems

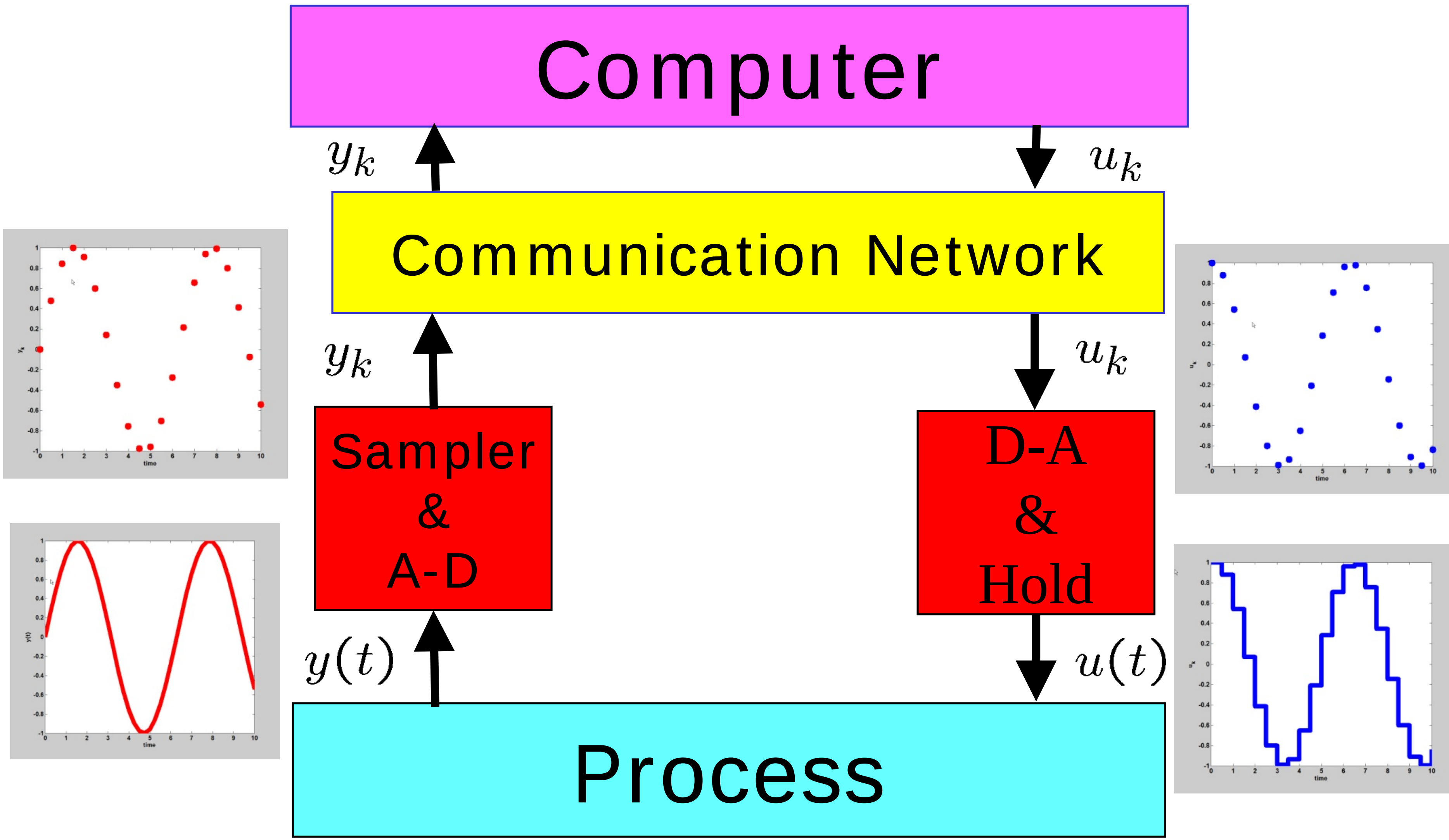
&

MPC

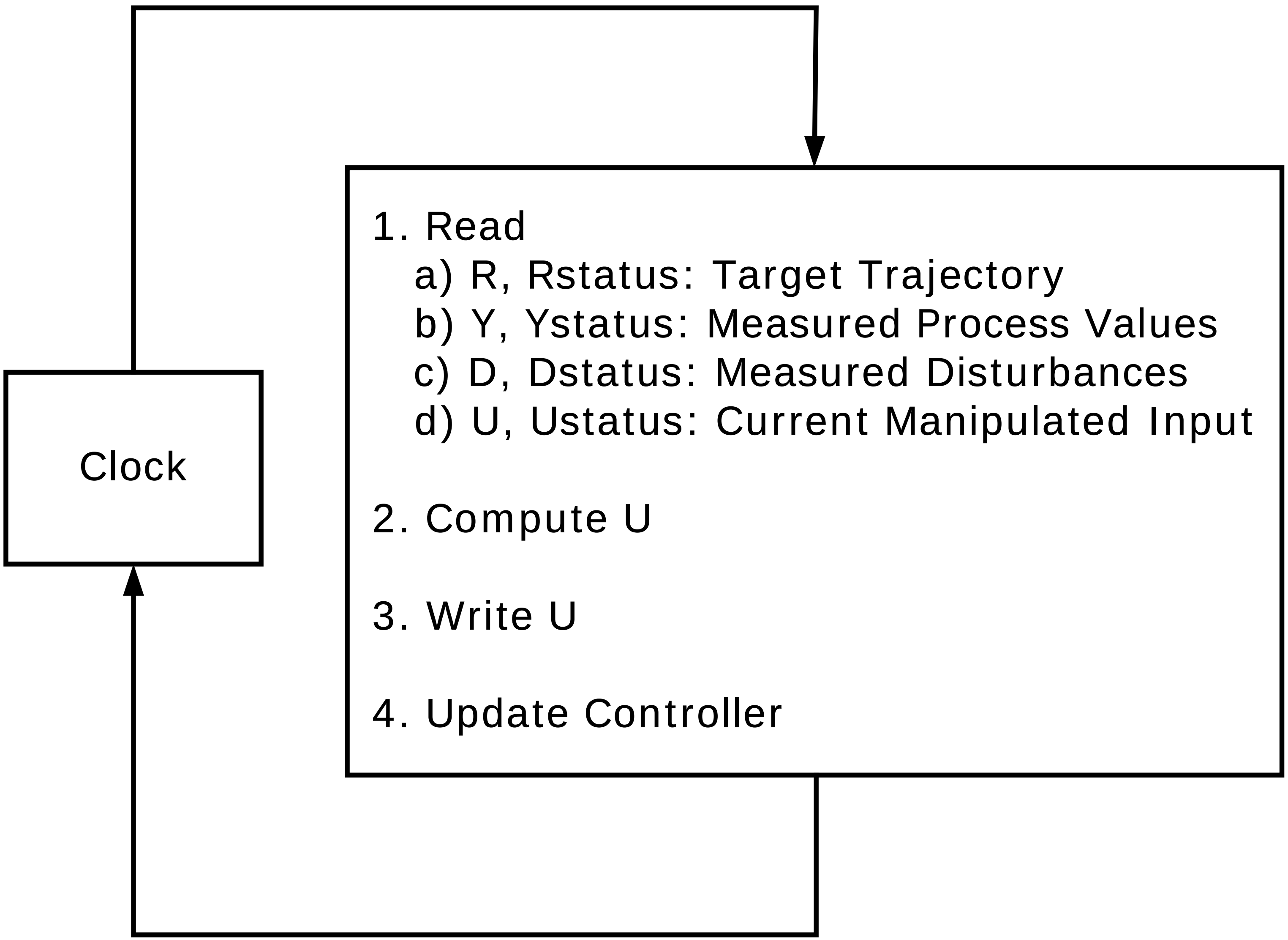
Quadruple Tank Process



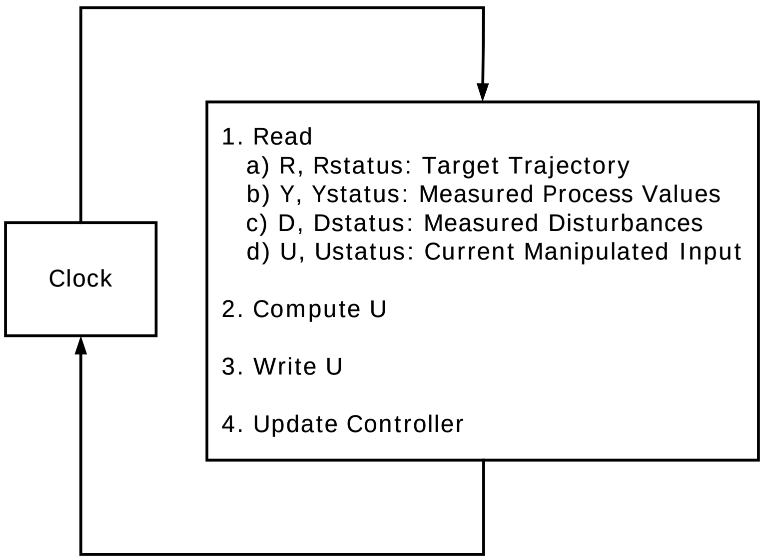
Computer Controlled Systems



Tasks in Computer Controlled Systems



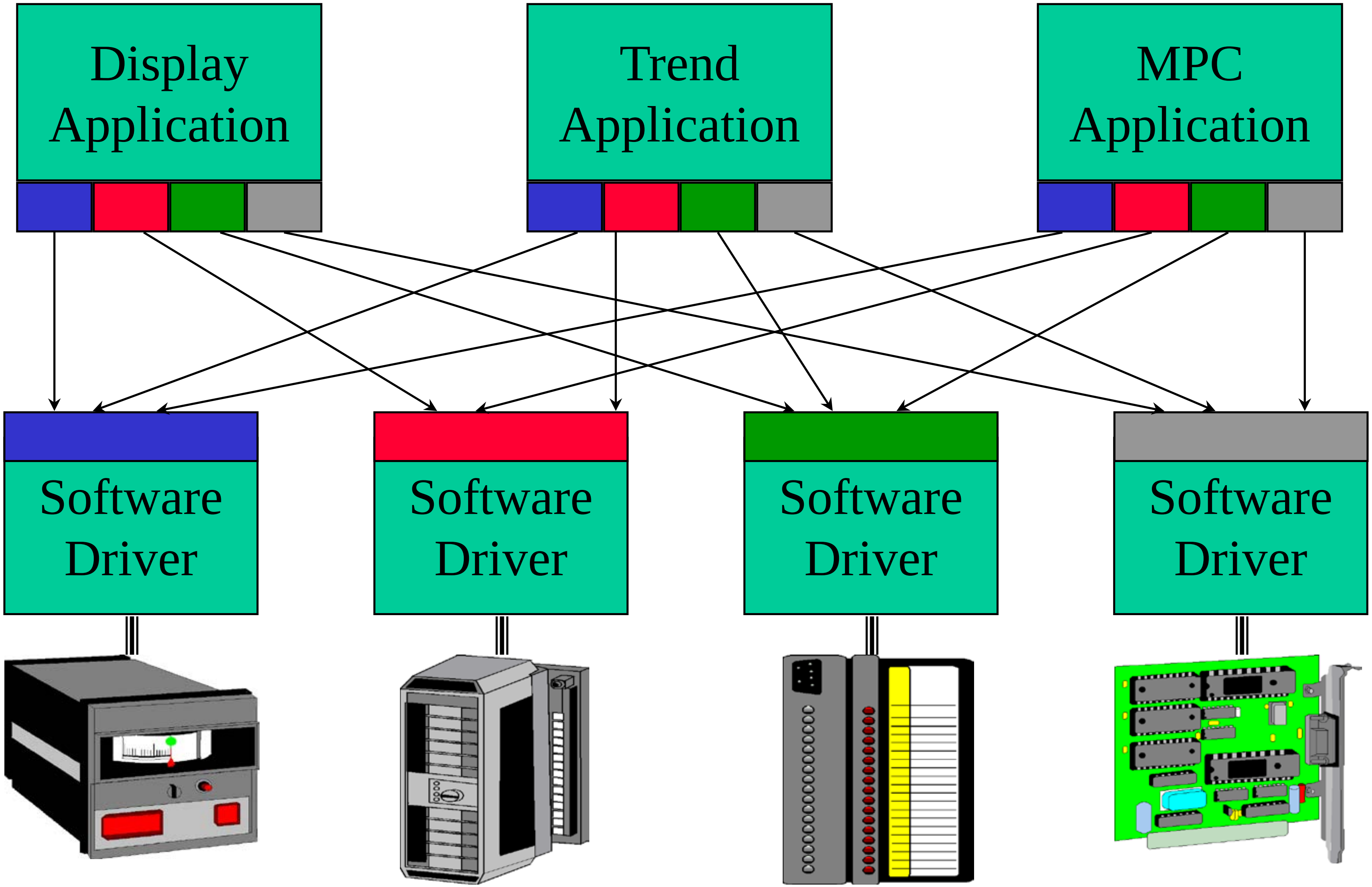
Tasks in Computer Controlled Systems



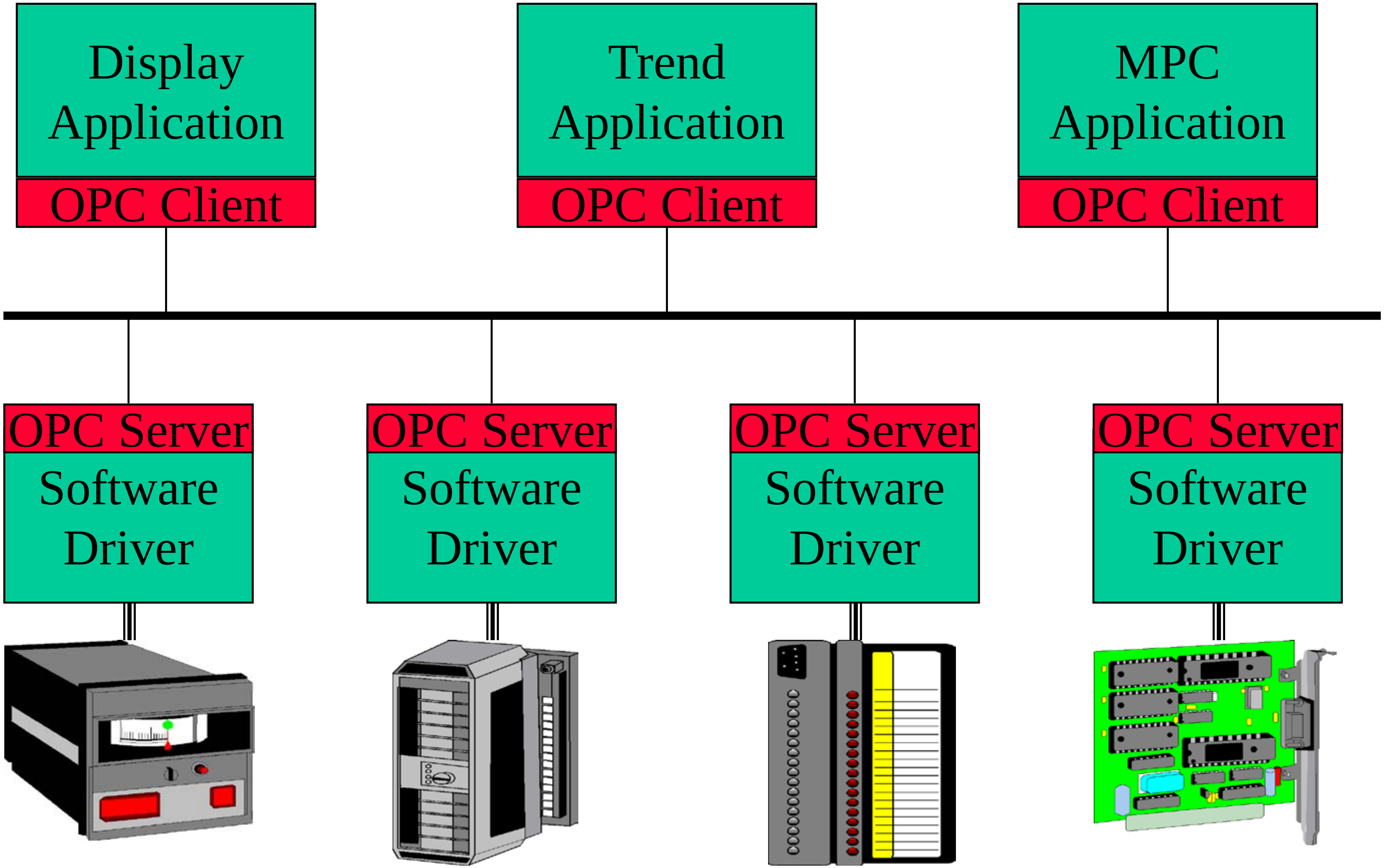

```
DigitalMPCtimer = ...
timer(...
 ’TimerFcn’, @MPCfun,...
 ’ExecutionMode’, ’fixedRate’,...
 ’Period’, 10.0);

start(DigtitalMPCtimer);
```

```
function MPCfun(obj,event)

% 1. Read
[R,Rstatus] = OPCRead(Rtag);
[Y,Ystatus] = OPCRead(Ytag);
[D,DStatus] = OPCRead(Dtag);
[U,Ustatus] = OPCRead(Utag);

% 2. Compute
Unew = MPCcompute(R,Y,D,U);

% 3. Write
OPCWrite(Unew);

% 4. Update controller
MPCupdate();
```

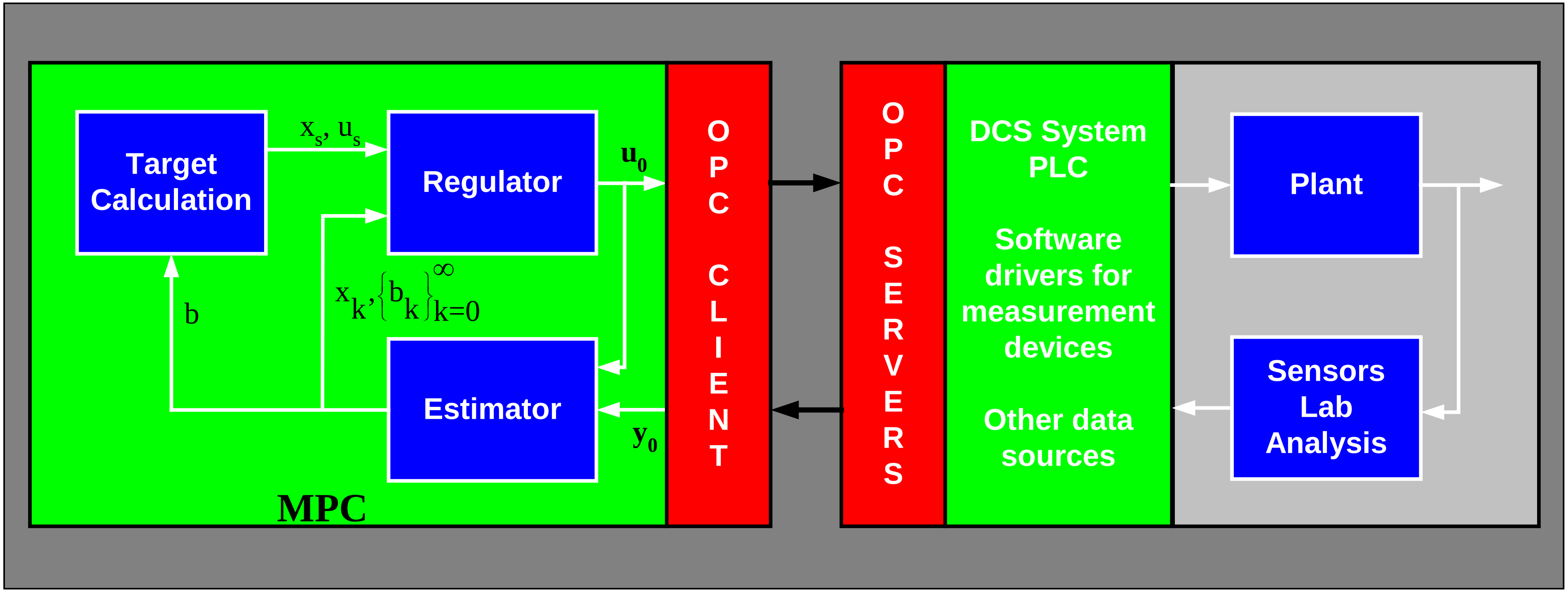
Communication: Read & Write



Read & Write using OPC

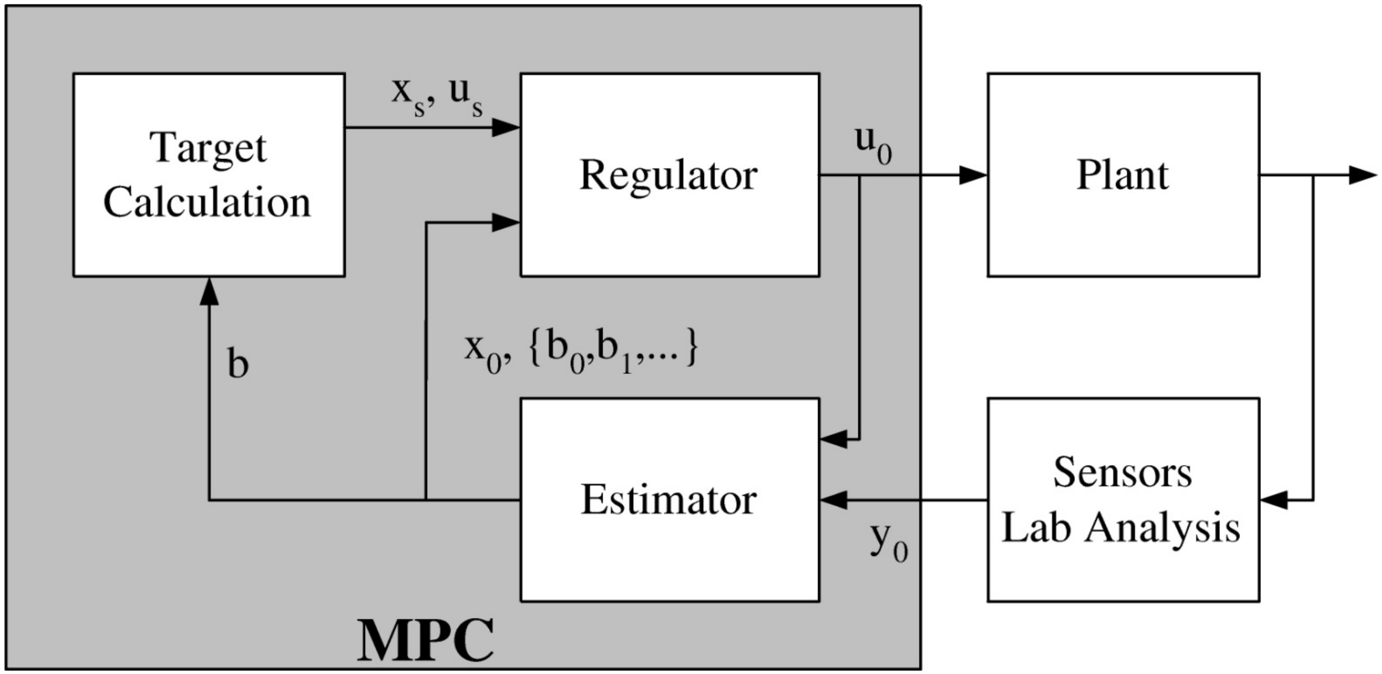


Connection of MPC App to Plant

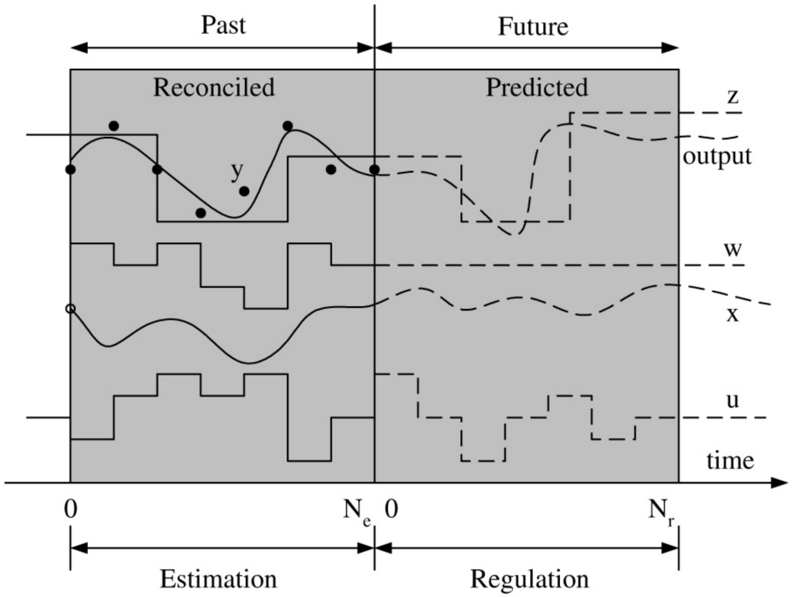


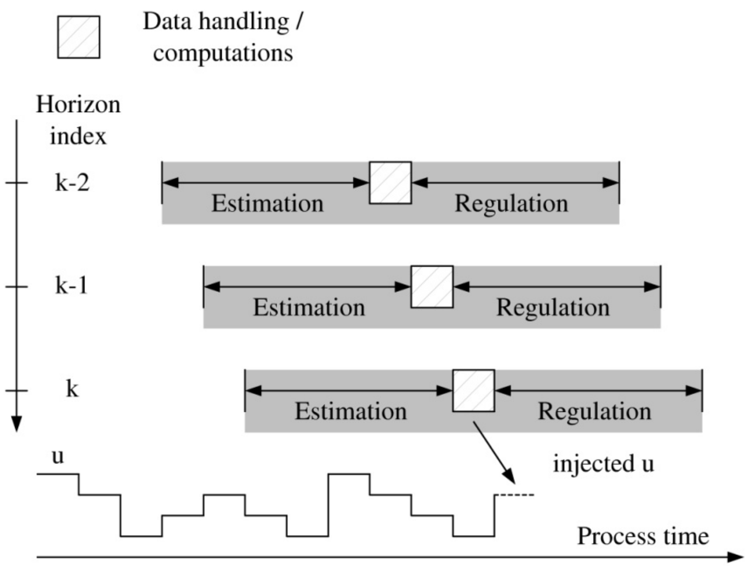
Industrial IT is accessing, monitoring and controlling physical plant hardware

Model Predictive Controller



MPC – Basic Idea





Estimation and regulation problem

Moving horizon implementation

State Estimation

$$\begin{aligned}
\min_{\{x_0,w,v\}} \quad & \phi = \frac{1}{2}\|x_0 - \bar{x}_0\|_X^2 + \frac{1}{2}\sum_{k=0}^{N_e} \|v_k\|_V^2 + \|w_k\|_W^2 & \\
s.t. \quad & x_{k+1} = Ax_k + Bu_k + Ed_k + Gw_k & k = 0, 1, \ldots, N_e - 1 \\
& y_k = Cx_k + v_k & k = 0, 1, \ldots, N_e
\end{aligned}$$

$$\widehat{x}_{N_e} = \mu_e(\bar{x}_0, \{u_k\}_{k=0}^{N_e-1}, \{d_k\}_{k=0}^{N_e-1}, \{y_k\}_{k=0}^{N_e})$$

The Kalman Filter
is the solution to this problem

Regulation

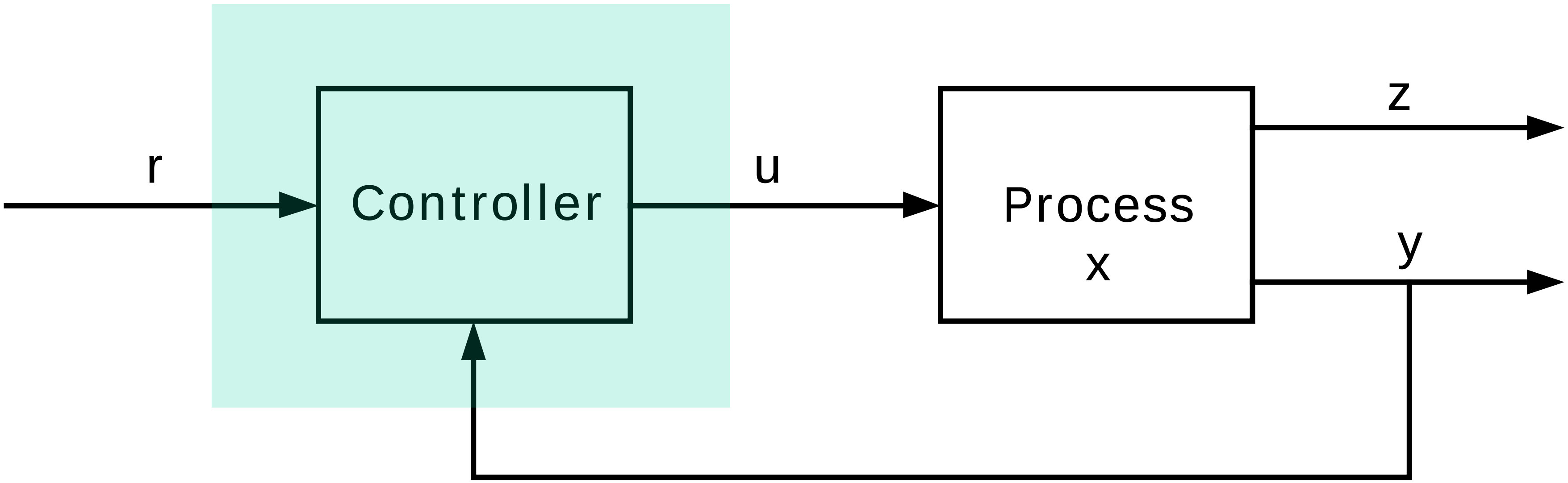
$$\min_{\{x,u,z\}} \phi = \frac{1}{2}\left(\sum_{k=0}^{N-1} \|z_k - r_k\|_{Q_z}^2 + \|\Delta u_k\|_S^2\right) + \frac{1}{2}\|z_N - r_N\|_{Q_z}^2$$

$$\begin{aligned}
s.t. \quad & x_{k+1} = Ax_k + Bu_k + Ed_k && k = 0, 1, \dots, N-1 \\
& z_k = C_z x_k && k = 0, 1, \dots, N \\
& u_{\min} \leq u_k \leq u_{\max} && k = 0, 1, \dots N-1 \\
& \Delta u_{\min} \leq \Delta u_k \leq \Delta u_{\max} && k = 0, 1, \dots, N-1 \\
& z_{\min} \leq z_k \leq z_{\max} && k = 0, 1, \dots, N
\end{aligned}$$

$$\Delta u_k = u_k - u_{k-1} \qquad x_0 = \hat{x}_{N_e}$$

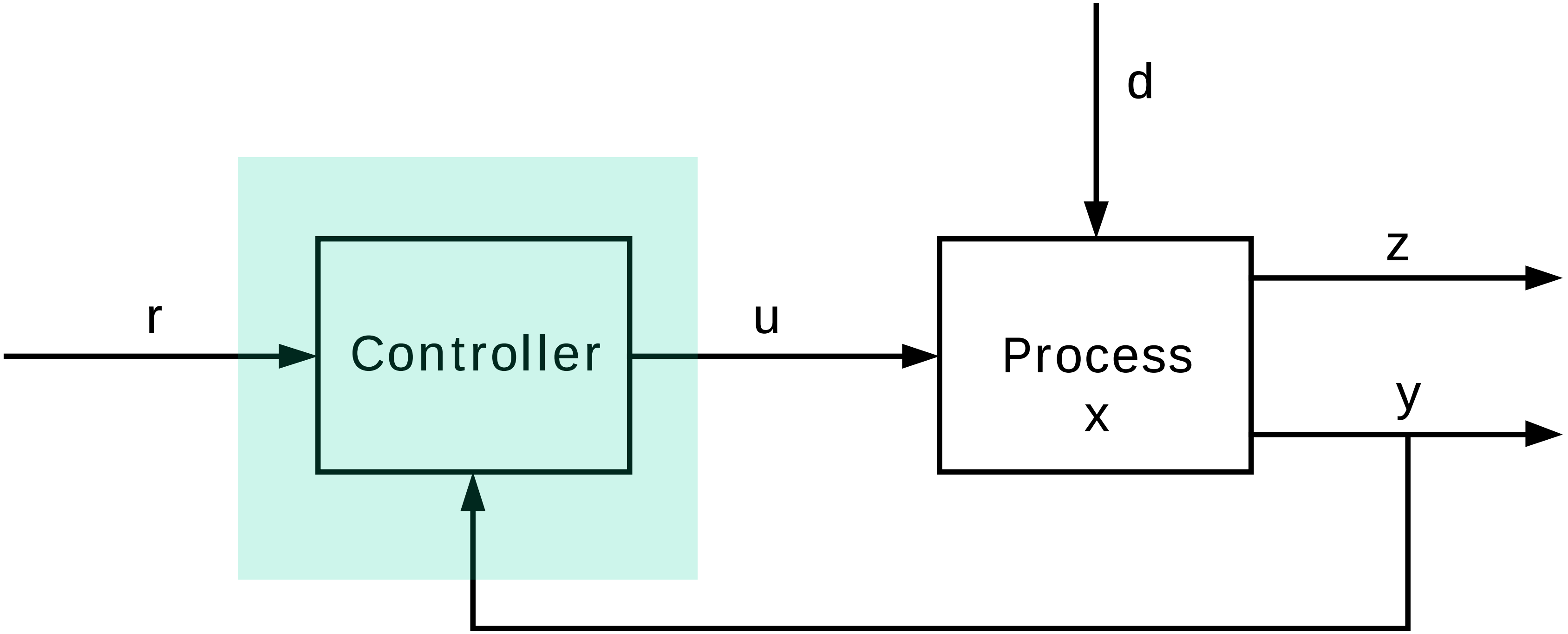
$$\{u_k^*\}_{k=0}^{N-1} = \mu_r(x_0, u_{-1}, \{r_k\}_{k=0}^{N}, \{d_k\}_{k=0}^{N-1})$$

Feedback Controller



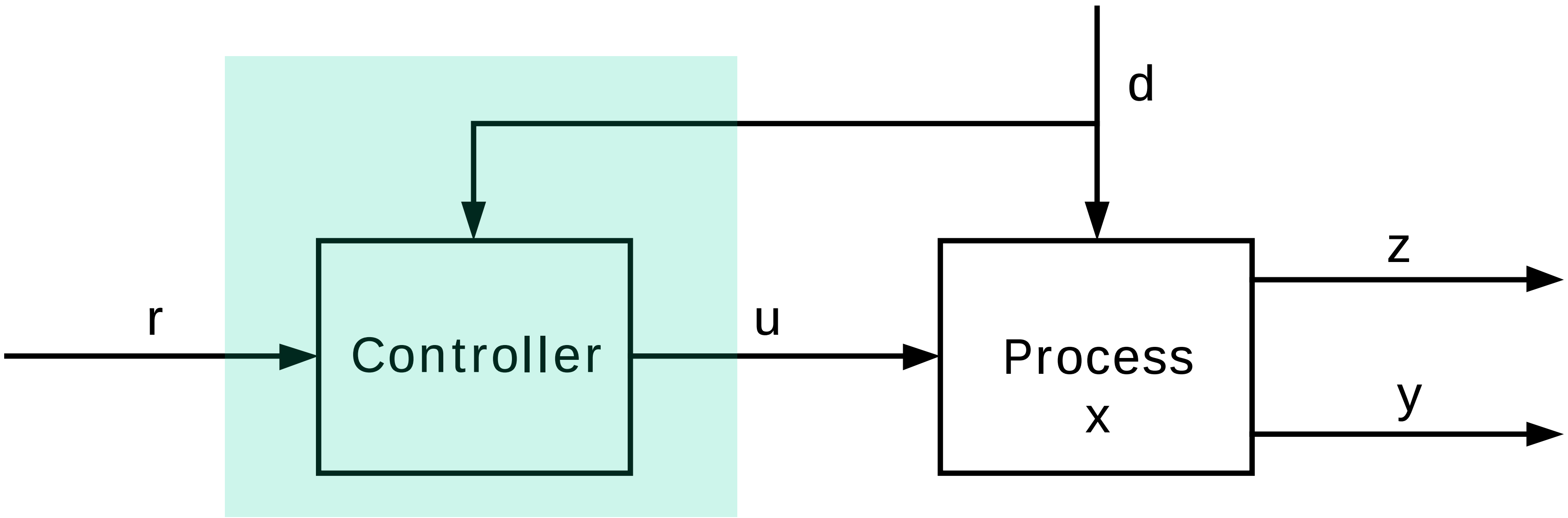
$$u(t) = \mu(r(t), y(t))$$

Feedback Controller



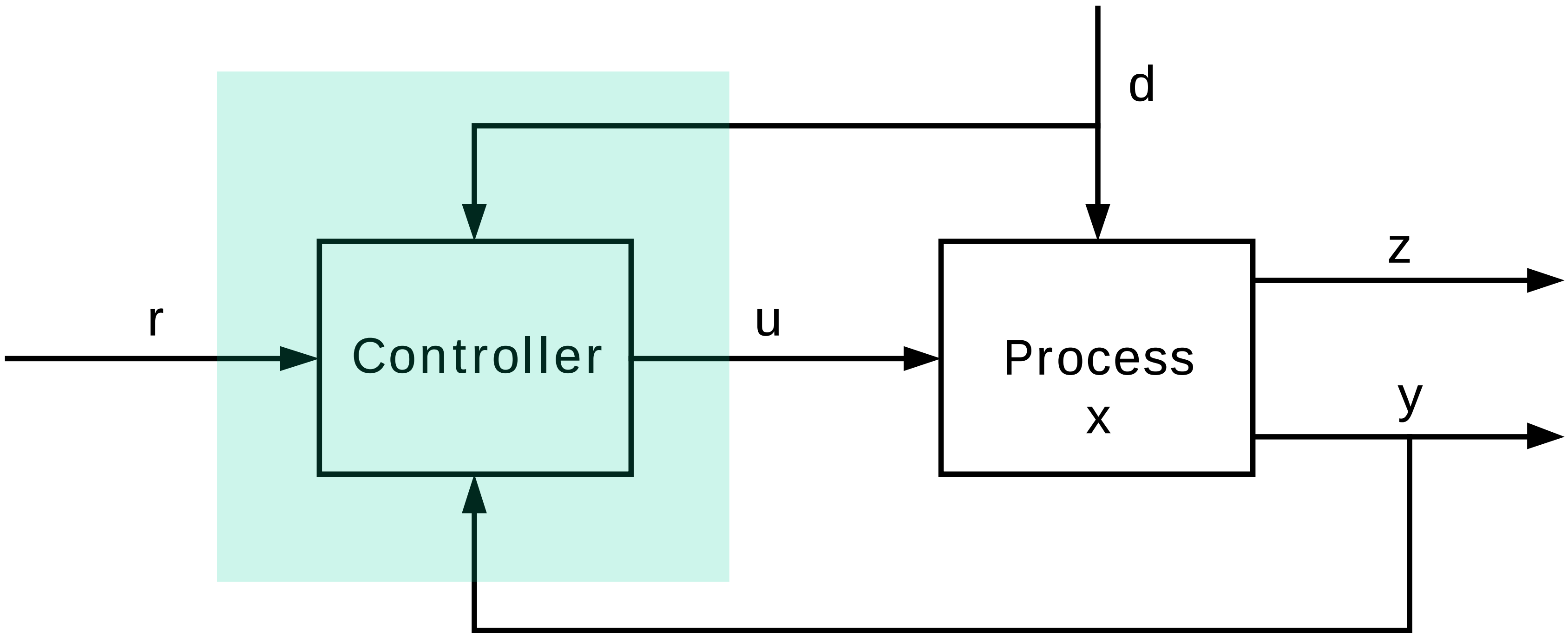
$$u(t) = \mu(r(t), y(t))$$

Feedforward Controller



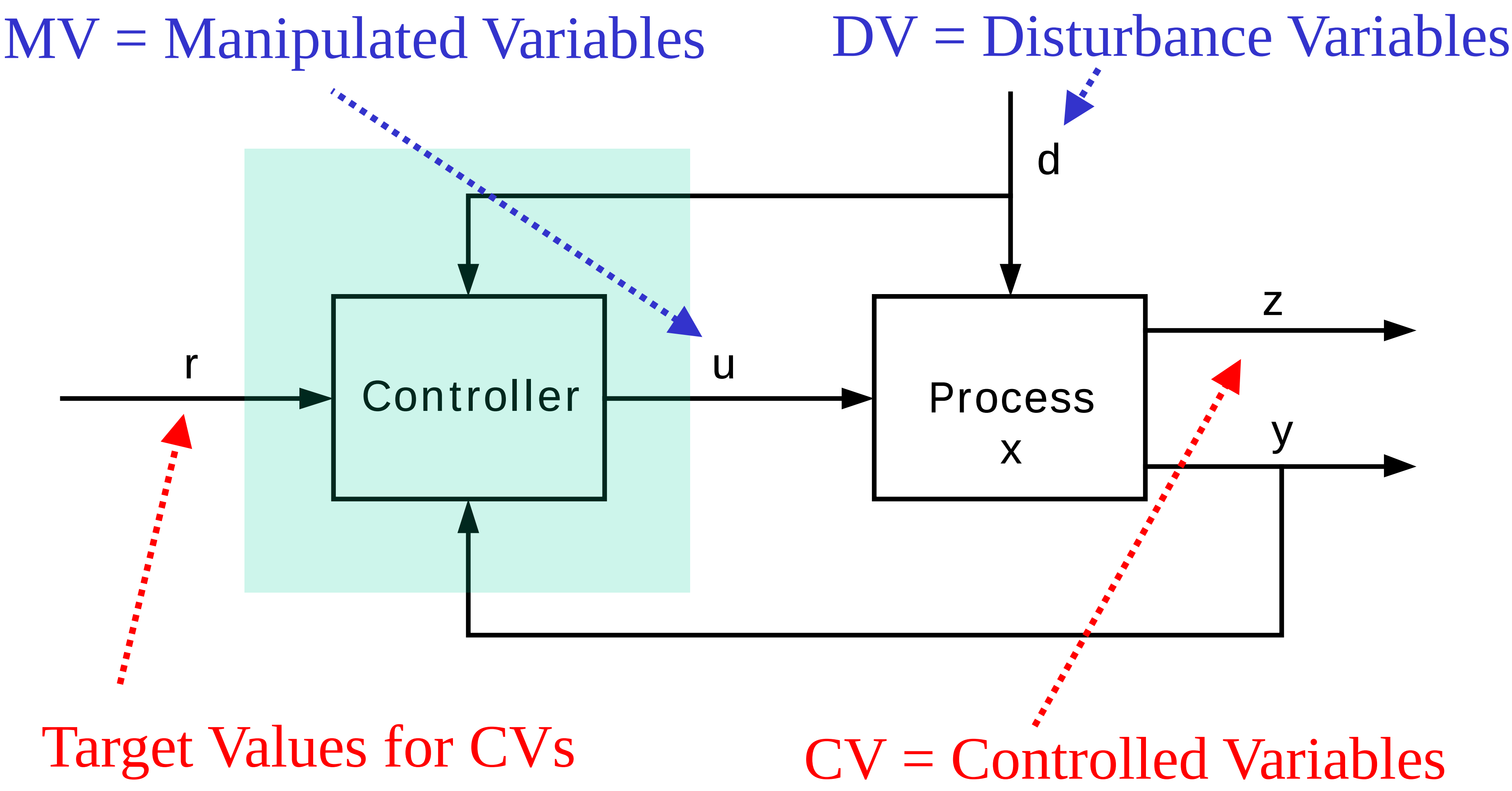
$$u(t) = \mu(r(t), d(t))$$

Feedforward-Feedback Controller



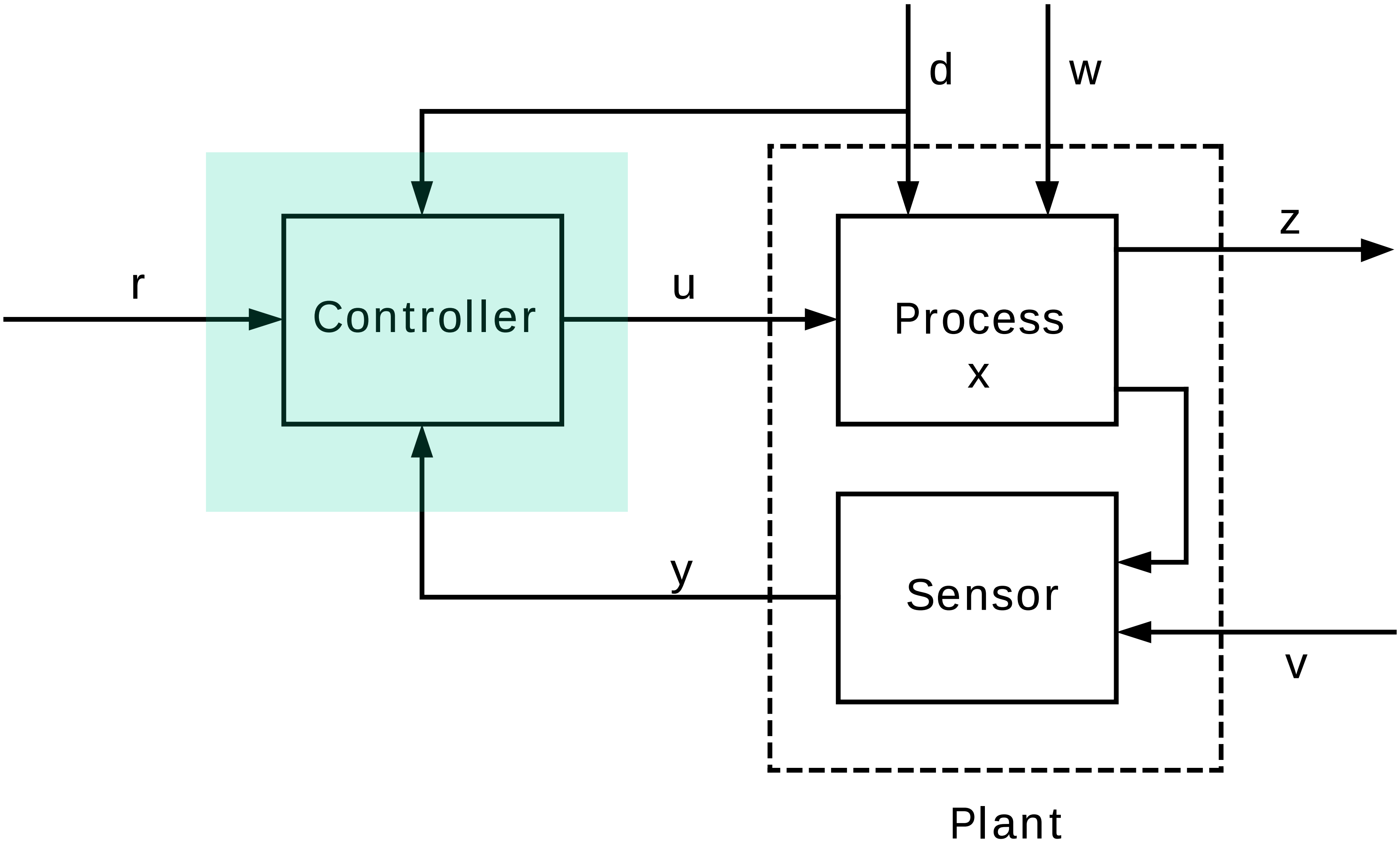
$$u(t) = \mu(r(t), y(t), d(t))$$

MVs, DVs, CVs

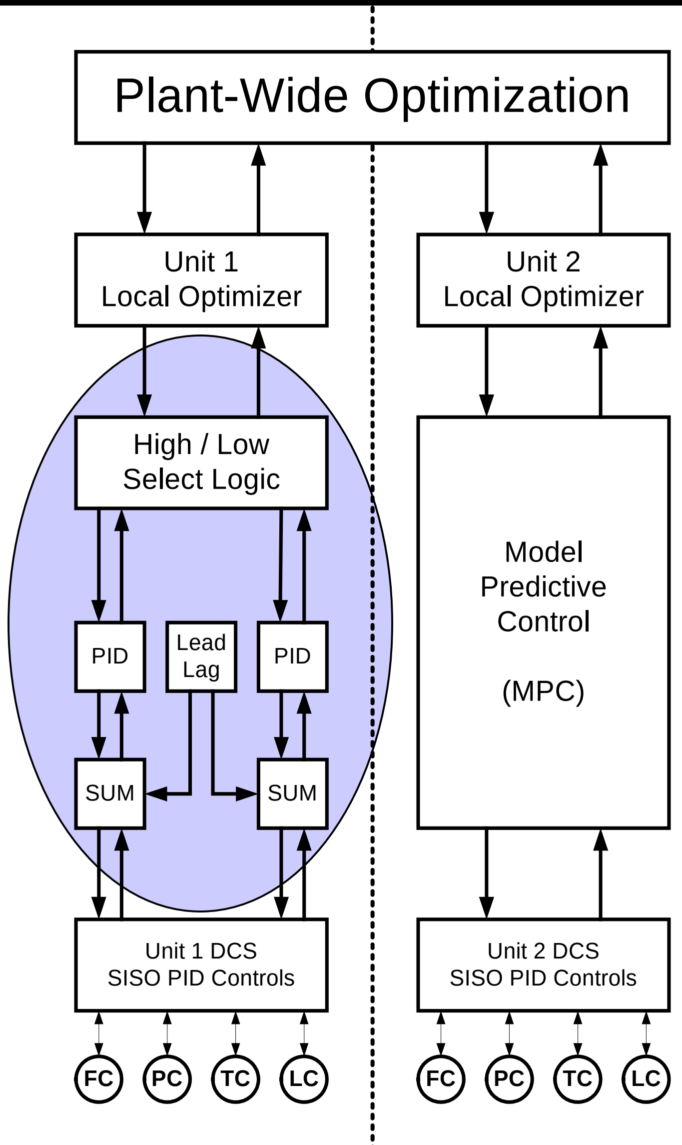


MV = Manipulated Variables
DV = Disturbance Variables
d
z
r
u
Controller
Process
x
y
Target Values for CVs
CV = Controlled Variables

MPC Block Diagram



Role of MPC in the Operational Hierarchy



Global steady state optimization
(every day)

Local steady state optimization
(every hour)

Make fine adjustments for operating conditions of local units

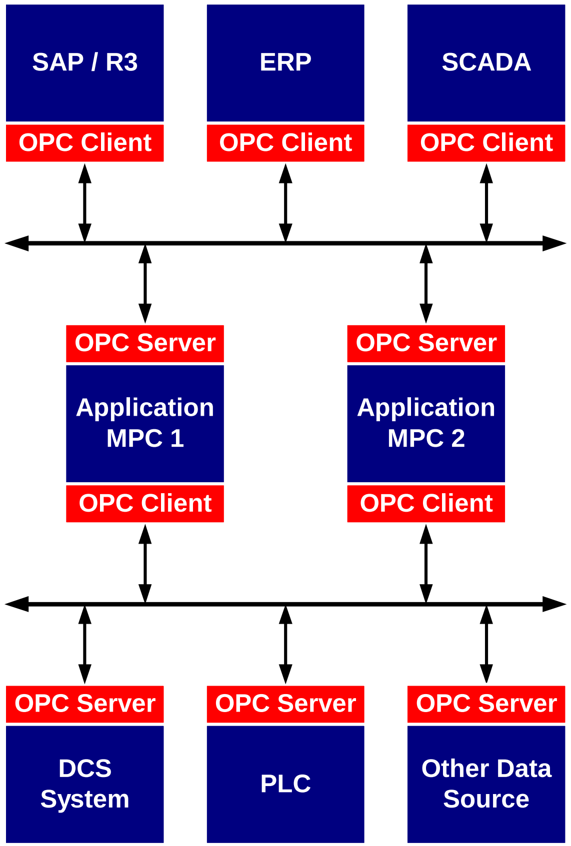
Dynamic constraint control
(every minute)

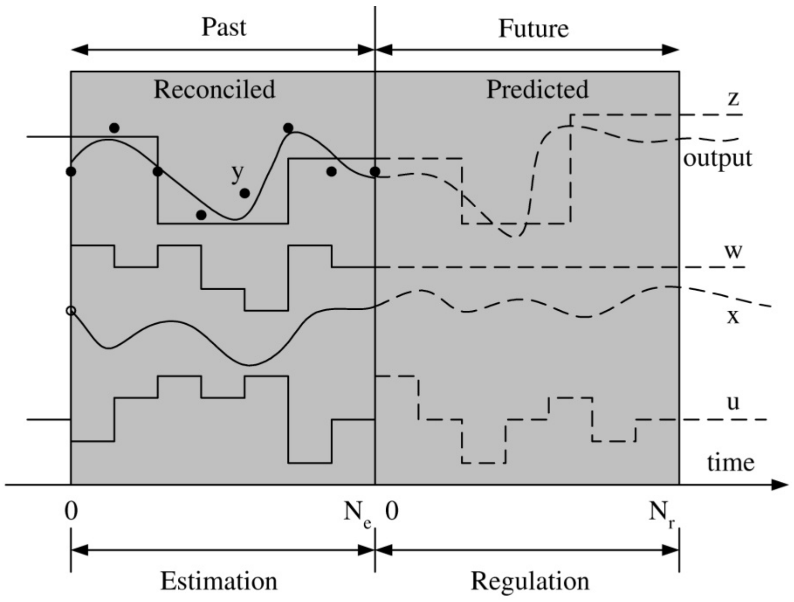
Take each local unit to the optimal condition.

Reject Disturbances.

Basic dynamic control
(every second)

Information Technology Infrastructure



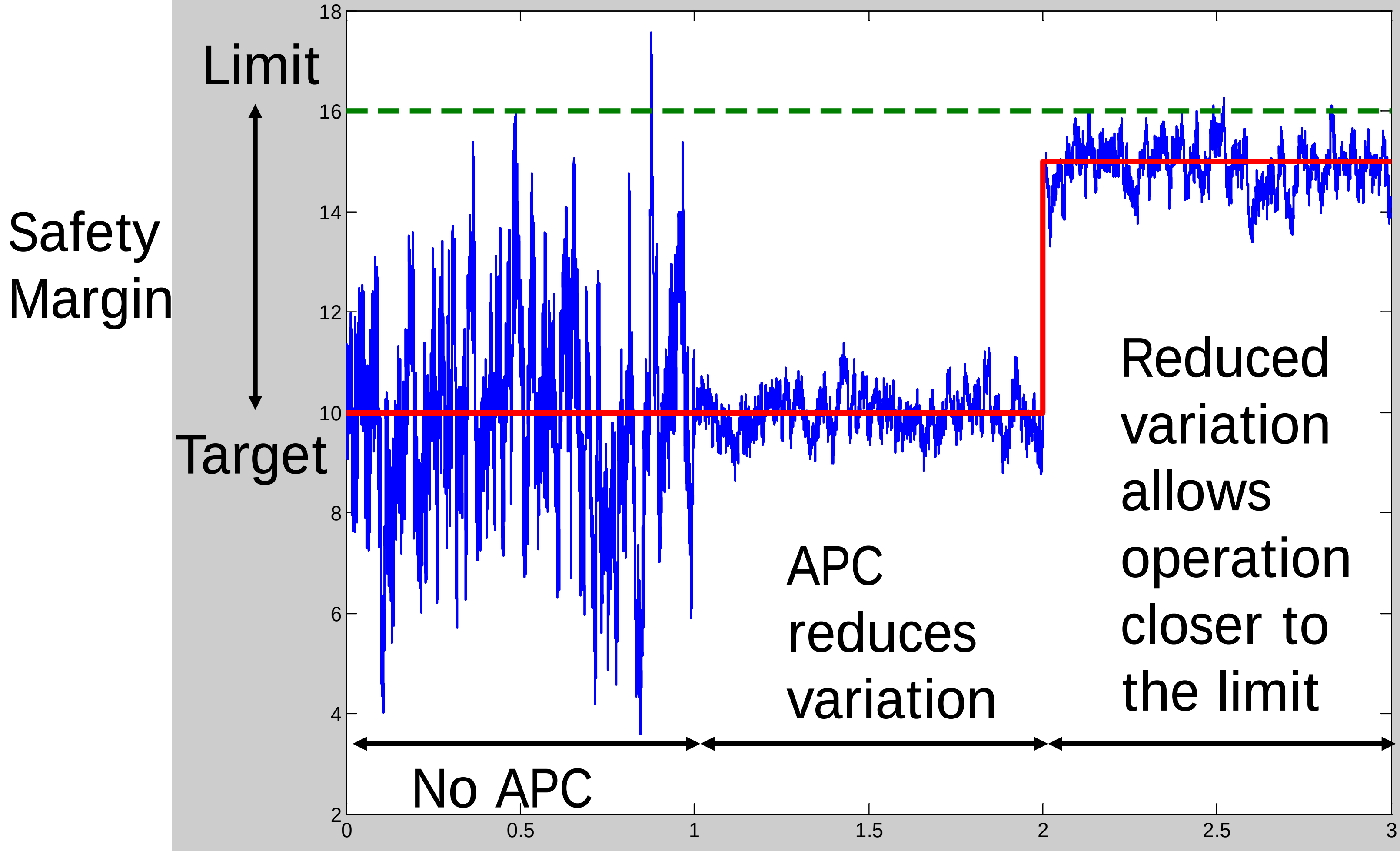


Read about the OPC toolbox in Matlab
– only for Windows

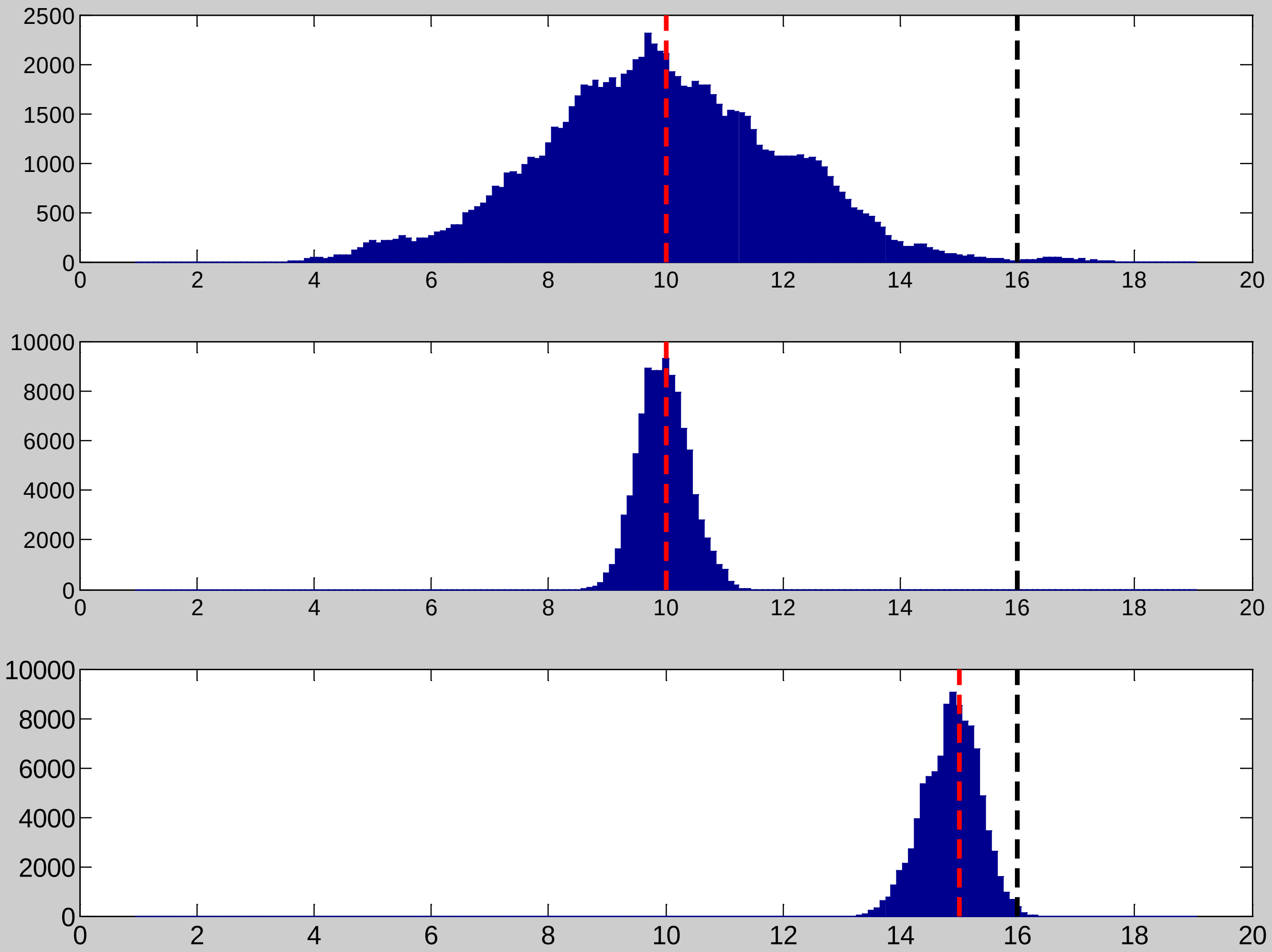
Technical Advantages of MPC

- **Explicit process models allow control of difficult dynamics**
  - Dead-time (time delay)
  - Inverse response
  - Interactions (multivariate)
  - Nonlinearity
- **Optimization of future plant behavior handles**
  - Feedforward from measured or estimated disturbances
  - Feedforward from setpoint changes and desired future trajectory
  - Feedback
- **Input and output constraints are handled by the controller**
- **Infrequent and irregular laboratory measurements**

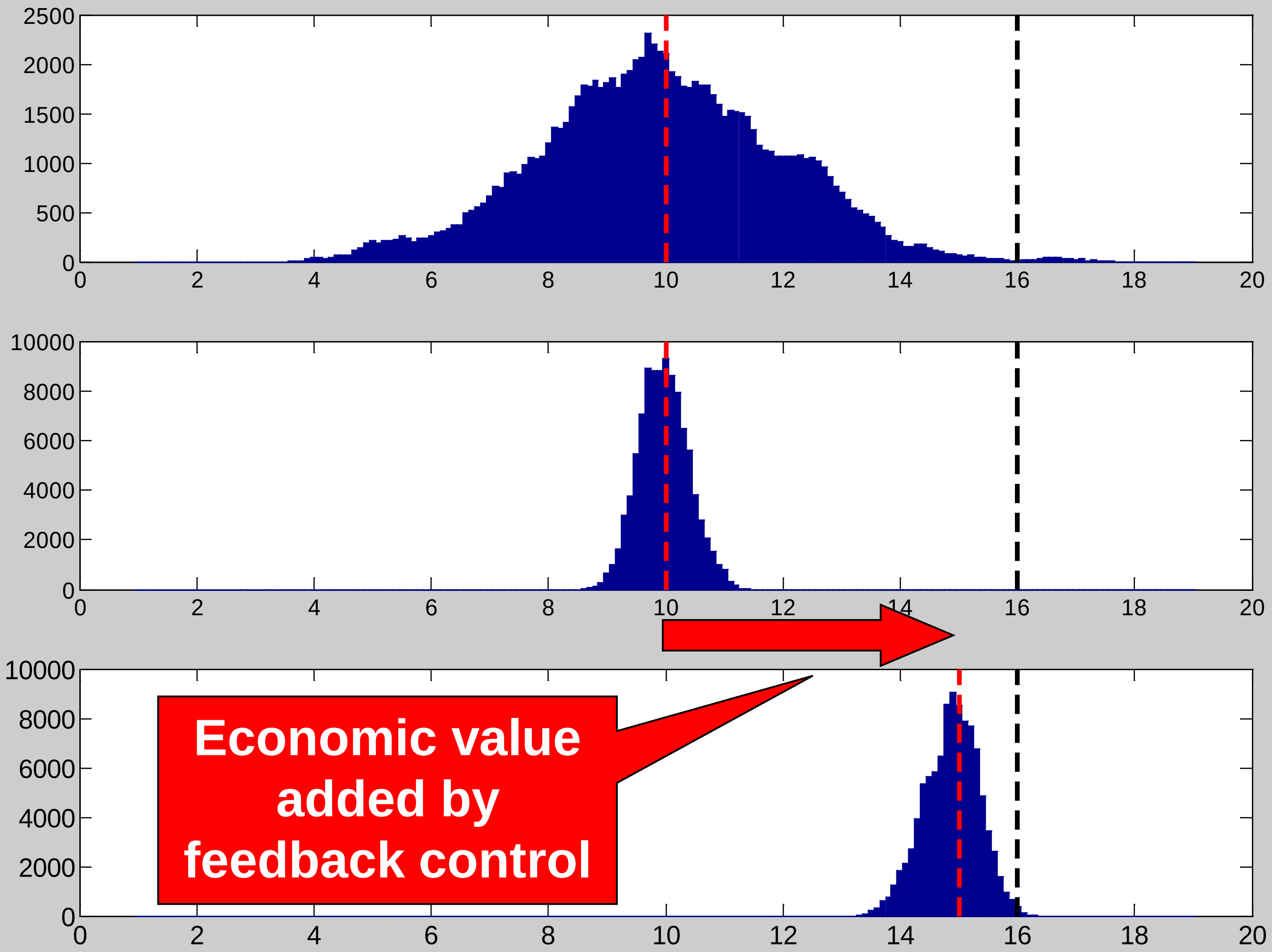
Economic Benefit of Process Control



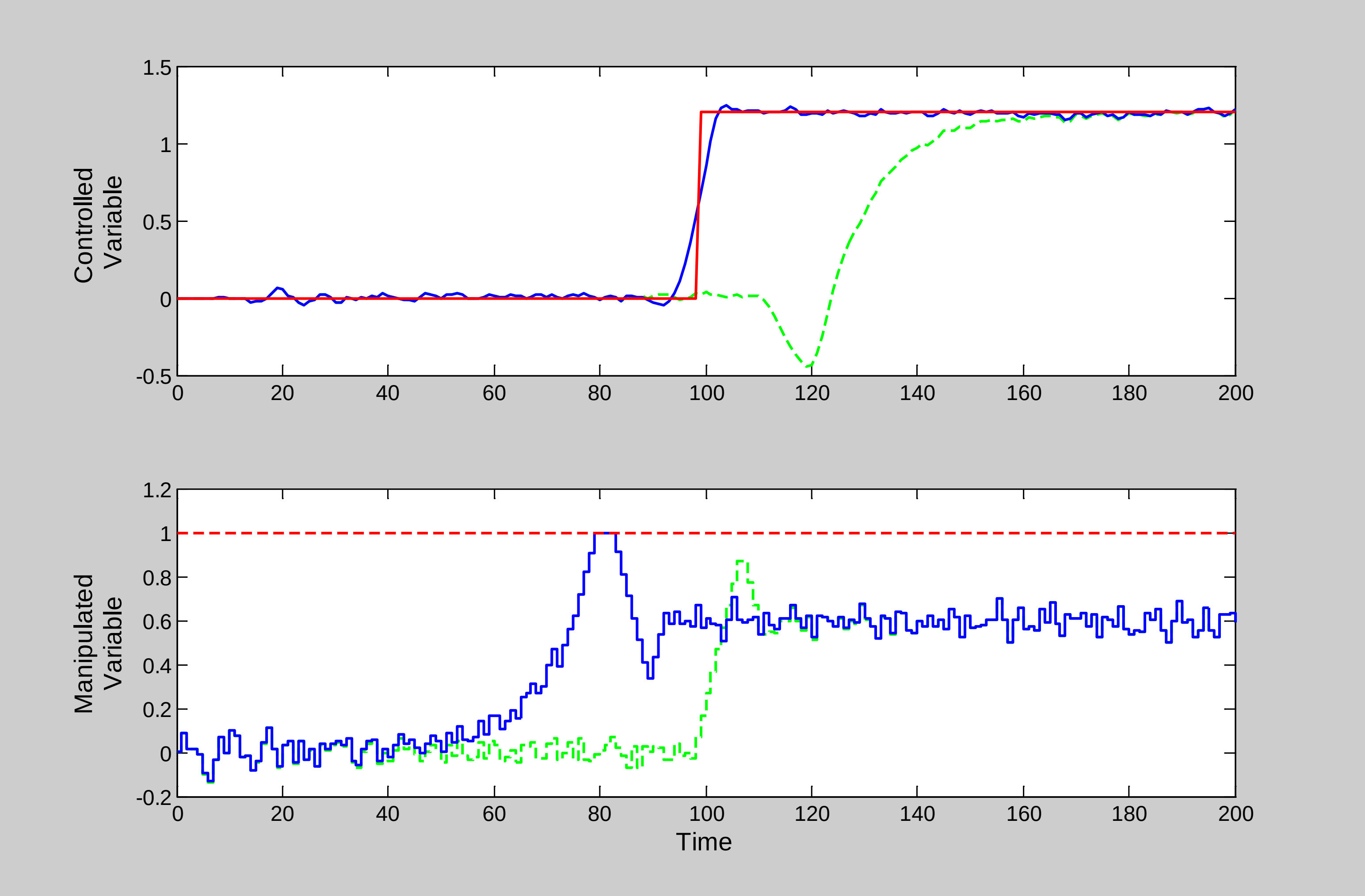
Economic Benefit of Process Control



Economic Benefit of Process Control

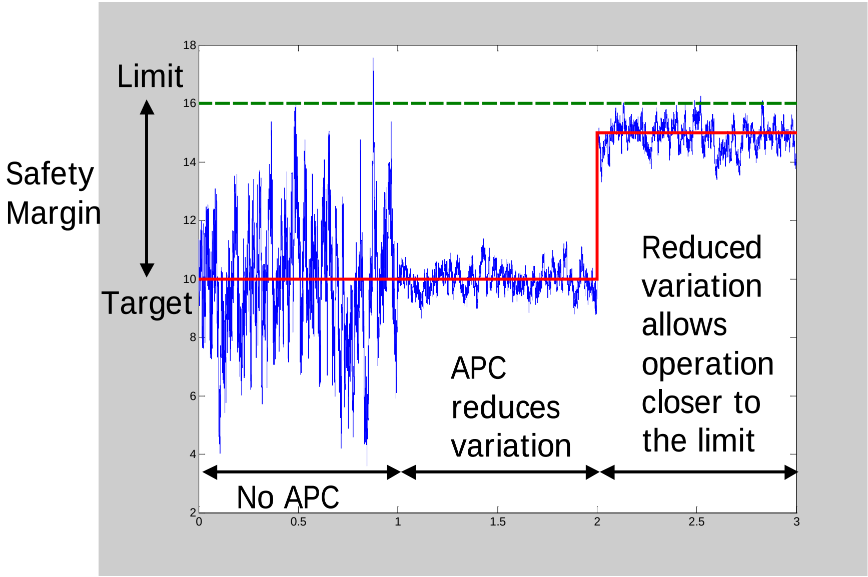


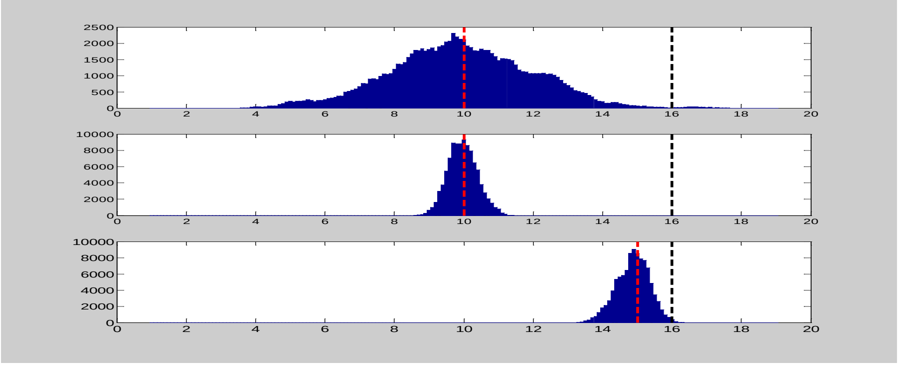
Rapid Product Change



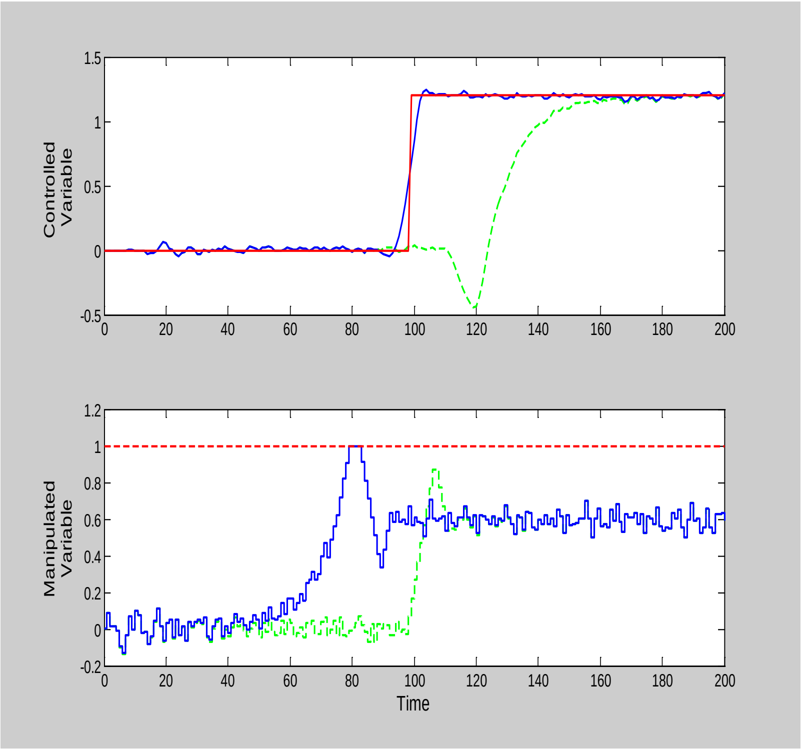
Economic Benefits of Process Control

Disturbance Rejection





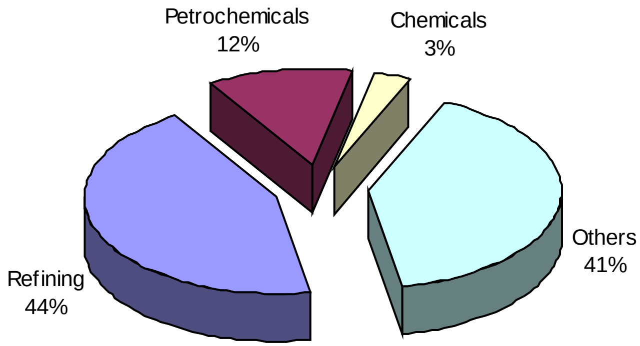
Reference Tracking



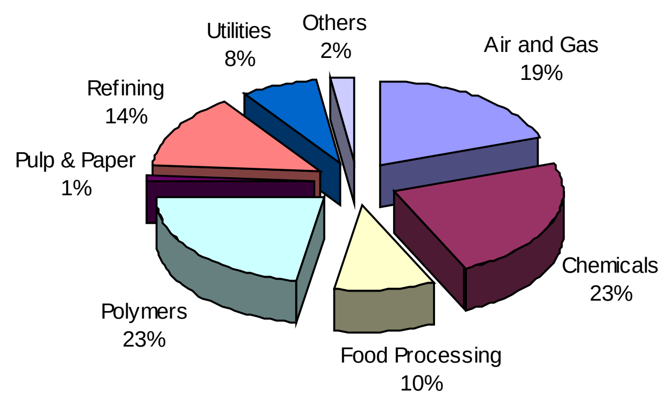
Applications in the Process Industries

- More than 4500 linear MPC applications
- Approx. 100 Nonlinear MPC
- Only 5 involved real first-principles models
- Several academic NMPC implementations (~50)
- 1000s of simulation (theoretical) NMPC papers

Linear MPC



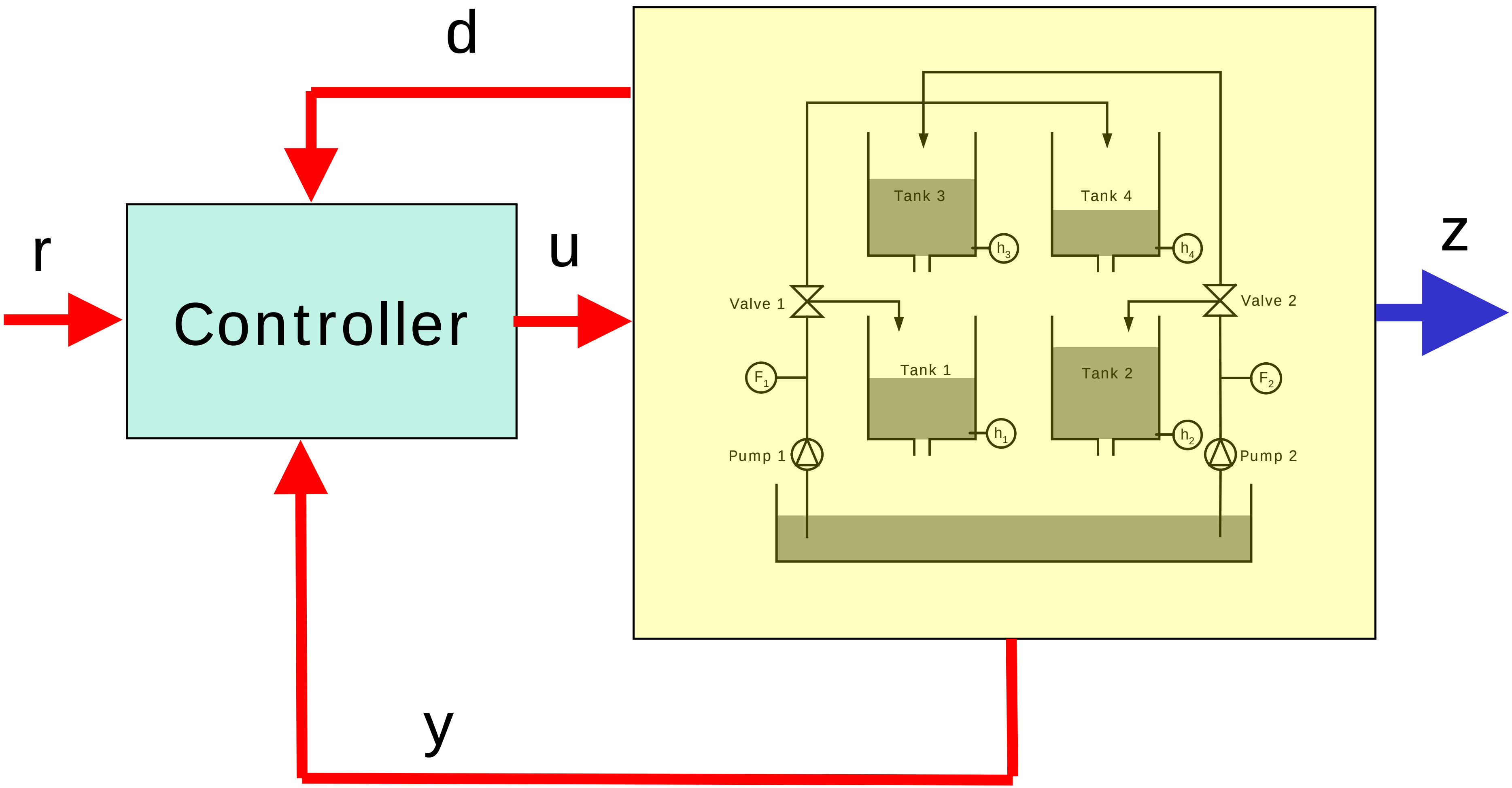
Nonlinear MPC



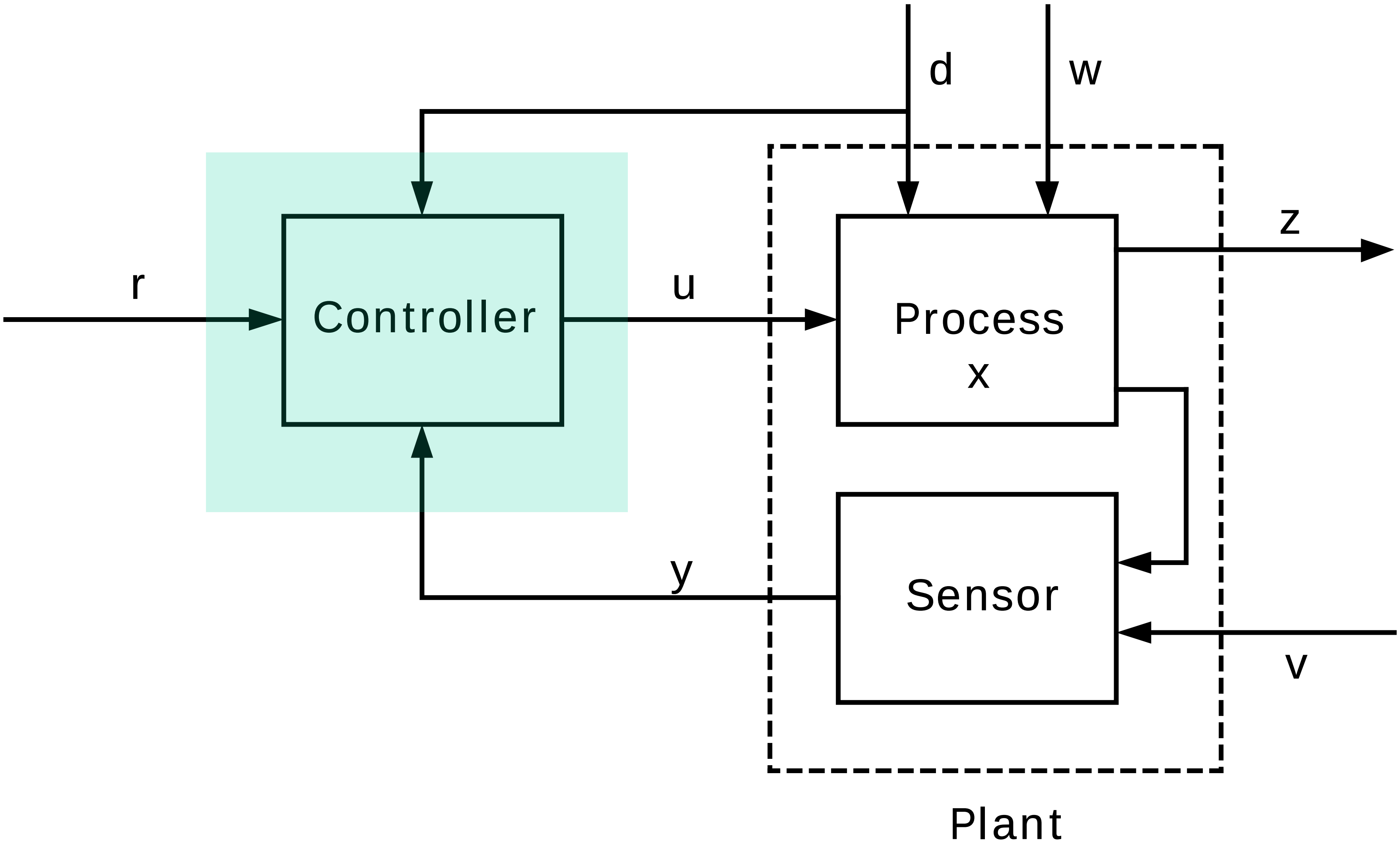
Qin and Badgwell (1996,2000,2001)

Creating a Virtual Plant using Modelling and Simulation

Motivation for Virtual Plant



Motivation for Virtual Plant



Modeling & Simulation

- Model a process as a system of ordinary differential equations (ODE)

$$\dot{x}(t) = f(x(t), u(t))$$
$$x(t_0) = x_0$$

- Simulate the system as the solution to this system of ordinary differential equations

Nonlinear State Space Model

Differential Equation

$$\dot{x}(t) = f(x(t), u(t))$$

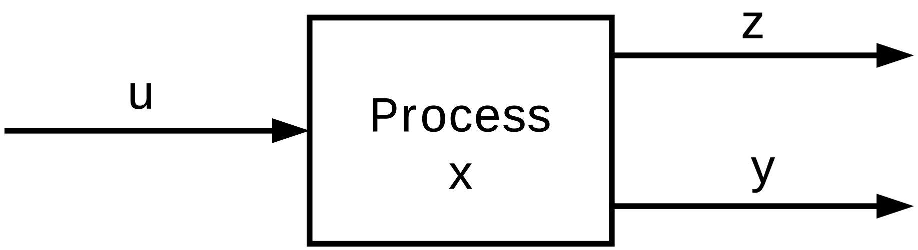
$$x(t_0) = x_0$$

Measurement Equation

$$y(t) = g(x(t))$$

Output Equation

$$z(t) = h(x(t))$$



Continuous and Discrete Time

Continuous Time

Differential Equation

$$\dot{x}(t) = f(x(t), u(t))$$

$$x(t_0) = x_0$$

Discrete Time

$$t_k = t_0 + kT_s$$

Difference Equation

$$x(t_k) = x_k$$

$$u(t) = u_k \qquad t_k \leq t < t_{k+1}$$

$$\dot{x}(t) = f(x(t), u(t))$$

$$x_{k+1} = x(t_{k+1})$$

$$\Downarrow$$

$$x_{k+1} = F(x_k, u_k)$$

Nonlinear State Space Model

Differential Equation

$$\dot{x}(t) = f(x(t), u(t), d(t))$$

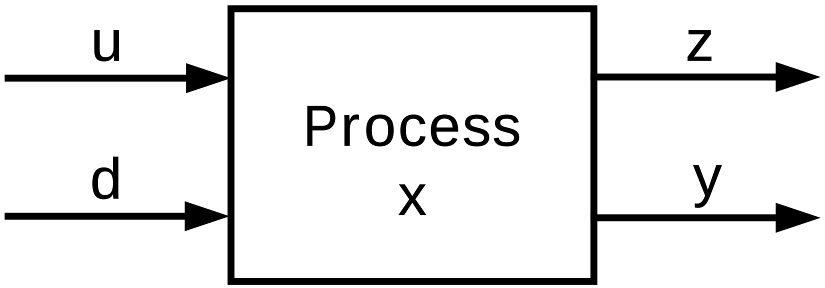
$$x(t_0) = x_0$$

Measurement Equation

$$y(t) = g(x(t))$$

Output Equation

$$z(t) = h(x(t))$$



Continuous-Time vs Discrete-Time

Continuous Time

$$\dot{x}(t) = f(x(t), u(t), d(t))$$

$$x(t_0) = x_0$$

$$y(t) = g(x(t))$$

$$z(t) = h(x(t))$$

Discrete Time

$$t_k = t_0 + kT_s$$

$$x_{k+1} = F(x_k, u_k, d_k)$$

$$x(t_k) = x_k$$

$$u(t) = u_k \quad t_k \leq t < t_{k+1}$$

$$d(t) = d_k \quad t_k \leq t < t_{k+1}$$

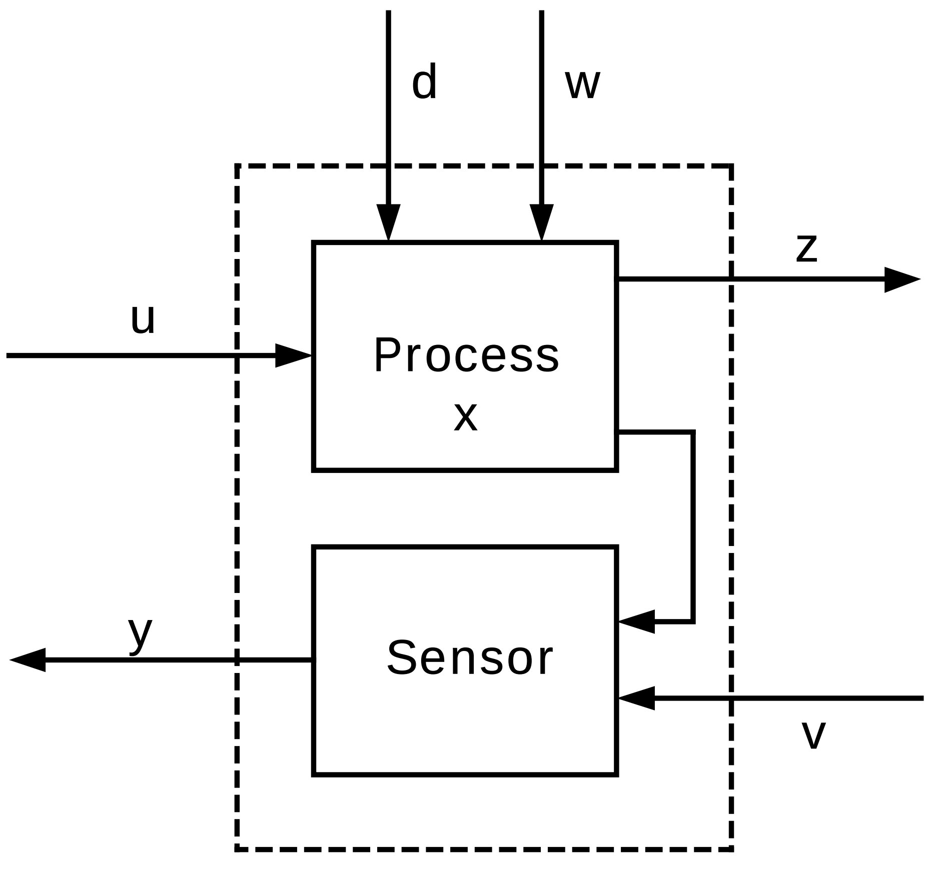
$$\dot{x}(t) = f(x(t), u(t), d(t))$$

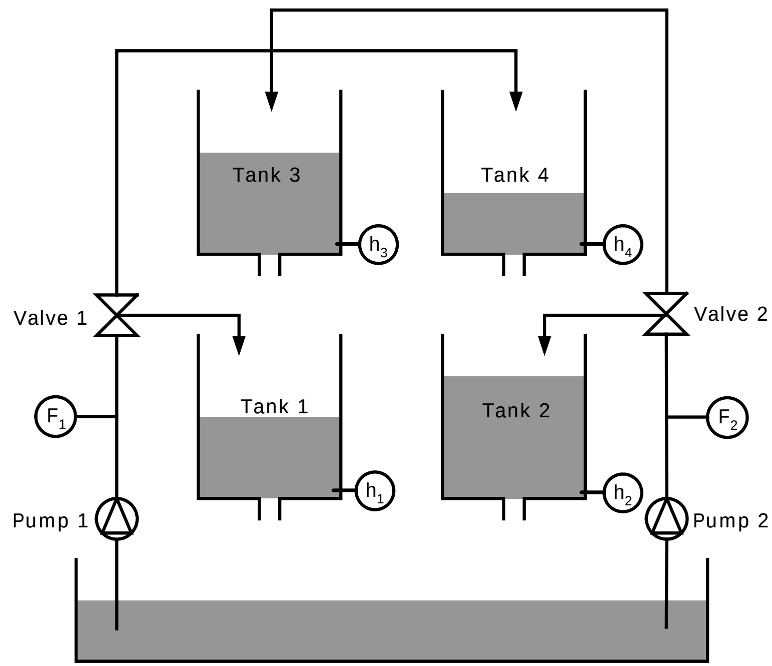
$$x_{k+1} = x(t_{k+1})$$

$$y_k = g(x_k)$$

$$z_k = h(x_k)$$

Process Model with All Signals





Discrete-Time State Space Model

$$\boldsymbol{x}_{k+1} = F(\boldsymbol{x}_k, u_k, d_k, \boldsymbol{w}_k) \qquad \boldsymbol{w}_k \sim N_{iid}(0, Q)$$

$$\boldsymbol{y}_k = g(\boldsymbol{x}_k) + \boldsymbol{v}_k \qquad \boldsymbol{v}_k \sim N_{iid}(0, R)$$

$$\boldsymbol{z}_k = h(\boldsymbol{x}_k)$$

The difference operator F is defined as

$$\begin{aligned}
x(t_k) &= x_k \\
u(t) &= u_k & t_k \leq t < t_{k+1} \\
d(t) &= d_k & t_k \leq t < t_{k+1} \\
w(t) &= w_k & t_k \leq t < t_{k+1} \\
\dot{x}(t) &= f(x(t), u(t), d(t), w(t)) & t_k \leq t < t_{k+1} \\
x_{k+1} &= x(t_{k+1})
\end{aligned}$$

Matlab Implementation

$$x(t_k) = x_k$$

$$u(t) = u_k \qquad t_k \le t < t_{k+1}$$

$$d(t) = d_k \qquad t_k \le t < t_{k+1}$$

$$w(t) = w_k \qquad t_k \le t < t_{k+1}$$

$$\dot{x}(t) = f(x(t), u(t), d(t), w(t)) \qquad t_k \le t < t_{k+1}$$

$$x_{k+1} = x(t_{k+1})$$

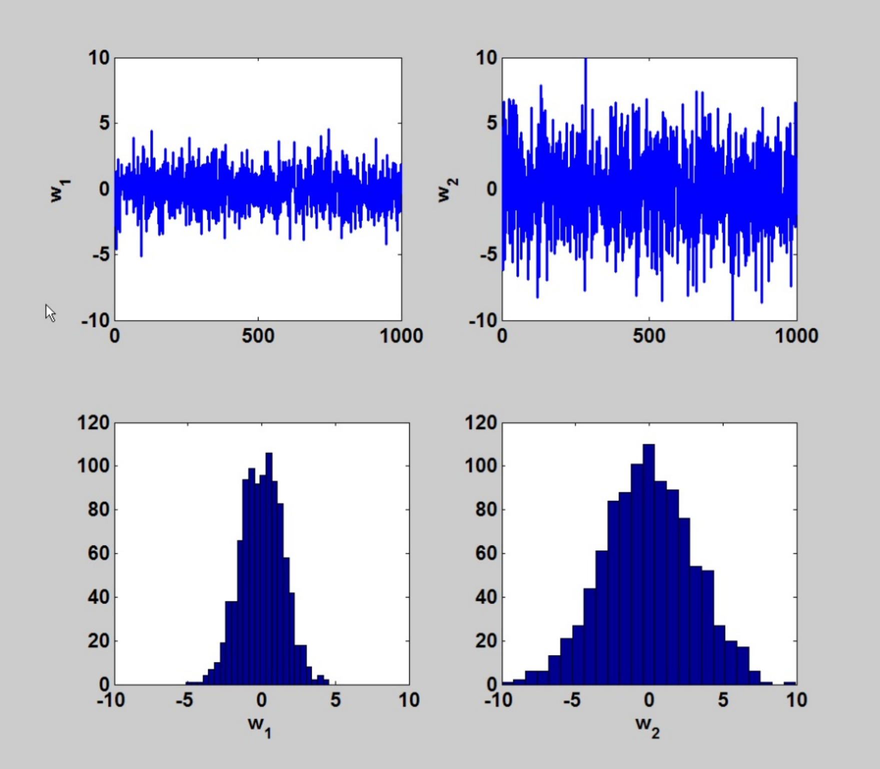
```
[T,X]=ode15s(@f, [t(k) t(k+1)], x(:,k), ...
                odeOptions, ...
                u(:,k), d(:,k), w(:,k) );

x(:,k+1) = X(end,:)';
```

Matlab Implemenation

$$w_k \sim N_{iid}(0, Q) \qquad Q > 0$$

$$w_k = L e_k \quad e_k \sim N_{iid}(0, I) \qquad Q = LL'$$

```
Q = [2 1; 1 10];
L = chol(Q)';

MySeed = 100;
randn('state', MySeed);

w = L*randn(2,1000);

subplot(221); plot(w(1,:));
subplot(222); plot(w(2,:));
subplot(223); hist(w(1,:),25);
subplot(224); hist(w(2,:),25);
```



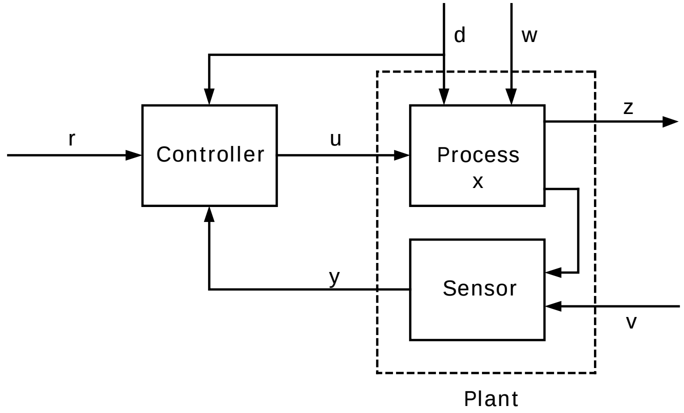
Matlab Implementation

$$\boldsymbol{y}_k = g(\boldsymbol{x}_k) + \boldsymbol{v}_k \qquad \boldsymbol{v}_k \sim N_{iid}(0, R)$$

$$\boldsymbol{z}_k = h(\boldsymbol{x}_k)$$

```
y(:,k) = g(x(:,k)) + v(:,k);
z(:,k) = h(x(:,k));
```

Matlab Implementation




```
% Measurement and Output Function
y(:,k) = g(x(:,k)) + v(:,k);
z(:,k) = h(x(:,k));

% Controller
u(:,k) = Controller(r(:,k),y(:,k),d(:,k));

% Simulate process from t(k) to t(k+1)
[T,X]=ode15s(@f, [t(k) t(k+1)], x(:,k), ...
             odeOptions, ...
            u(:,k), d(:,k), w(:,k) );
x(:,k+1) = X(end,:)';
```

SISO Discrete-Time Controllers

Proportional Controller

$$e_k = r_k - y_k$$
$$u_k = u_s + K_c e_k$$

```
function u = PControl(r,y,us,Kc)
e = r-y;
u = us + Kc*e;
```

Proportional-Integral Controller

$$e_k = r_k - y_k$$
$$i_{k+1} = i_k + \frac{K_c}{T_i} T_s e_k$$
$$u_k = u_s + K_c e_k + i_k$$

```
function [u,i] =...
          PIControl(i,r,y,us,Kc,Ti,Ts)
e = r-y;
u = us + Kc*e + i;
i = i+(Kc*Ts/Ti)*e;
```

SISO Discrete-Time Controllers with Clipping

Proportional Controller

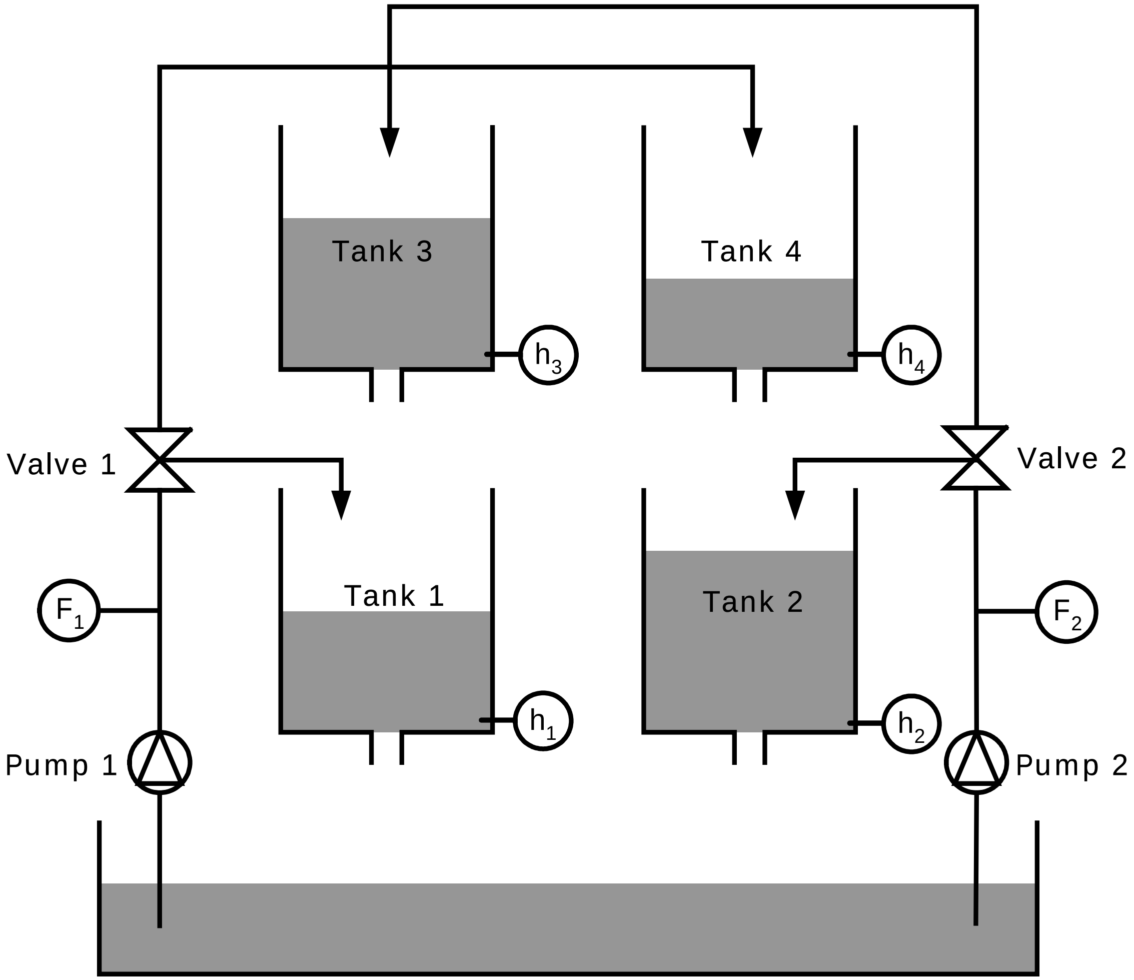
```
function u = PControl(r,y,us,Kc,umin,umax)
e = r-y;
v = us + Kc*e;
u = max(umin,min(umax,v));
```

Proportional-Integral Controller

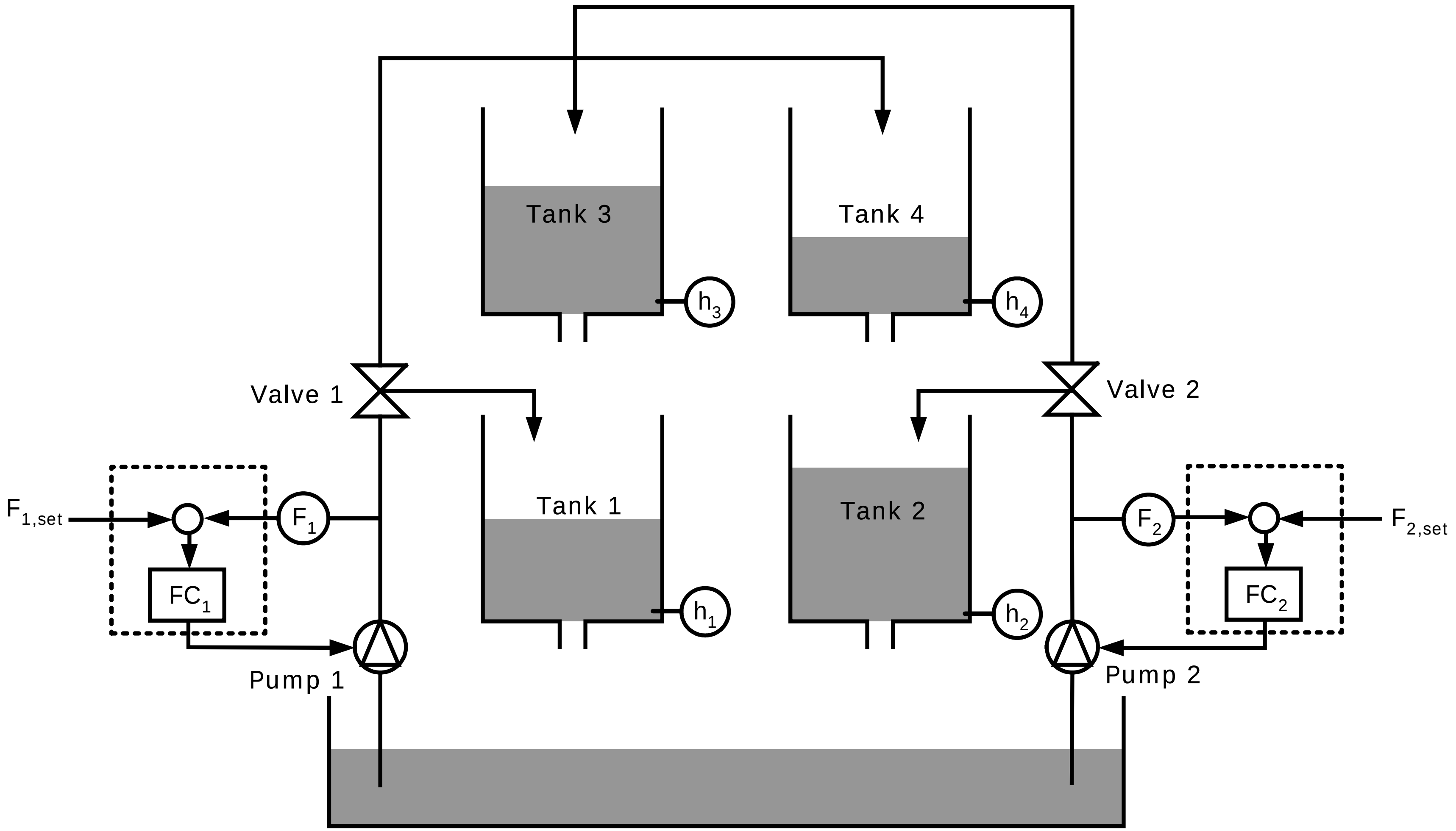
```
function [u,i] = PIControl(i,r,y,us,Kc,Ti,Ts,umin,umax)
e = r-y;
v = us + Kc*e + i;
i = i+(Kc*Ts/Ti)*e;
u = max(umin,min(umax,v));
```

Quadruple Tank Process

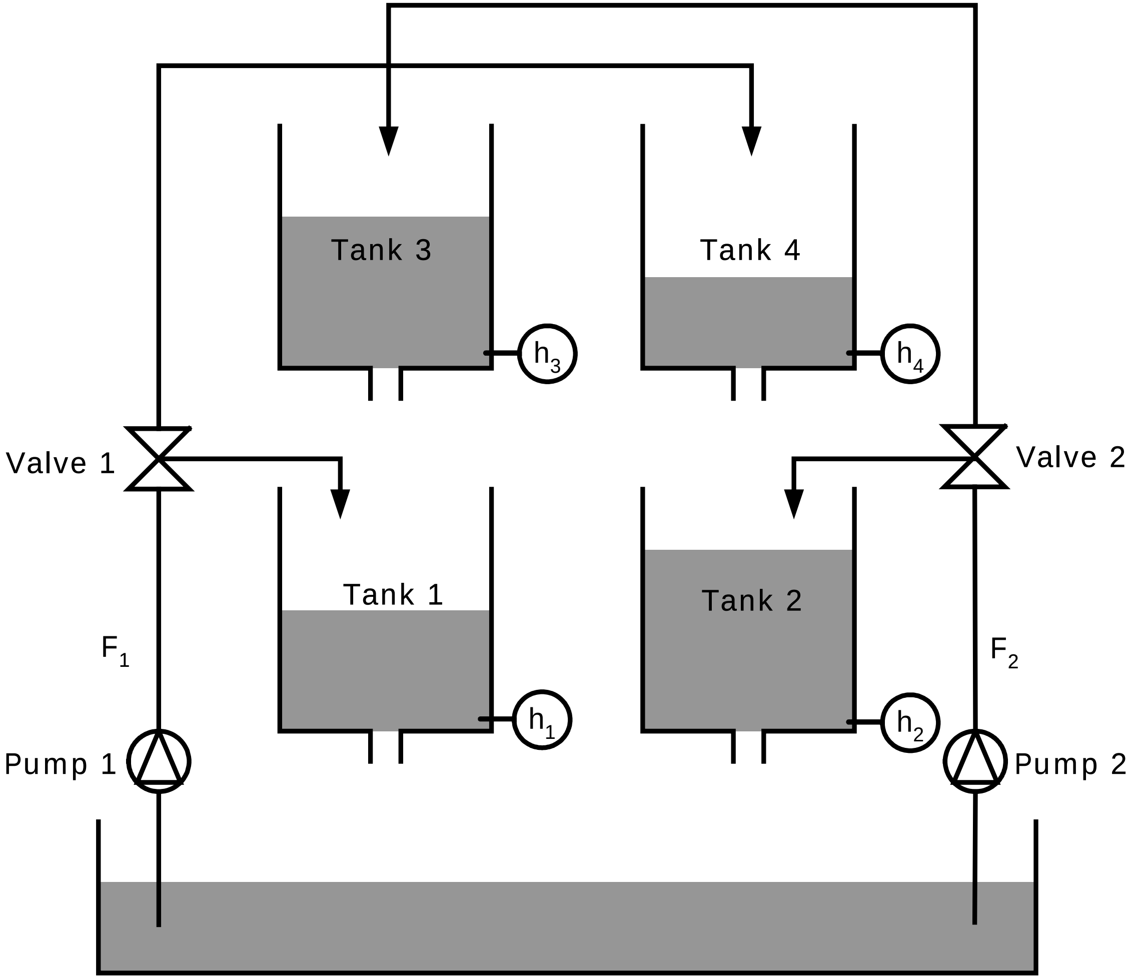
Quadruple Tank Process



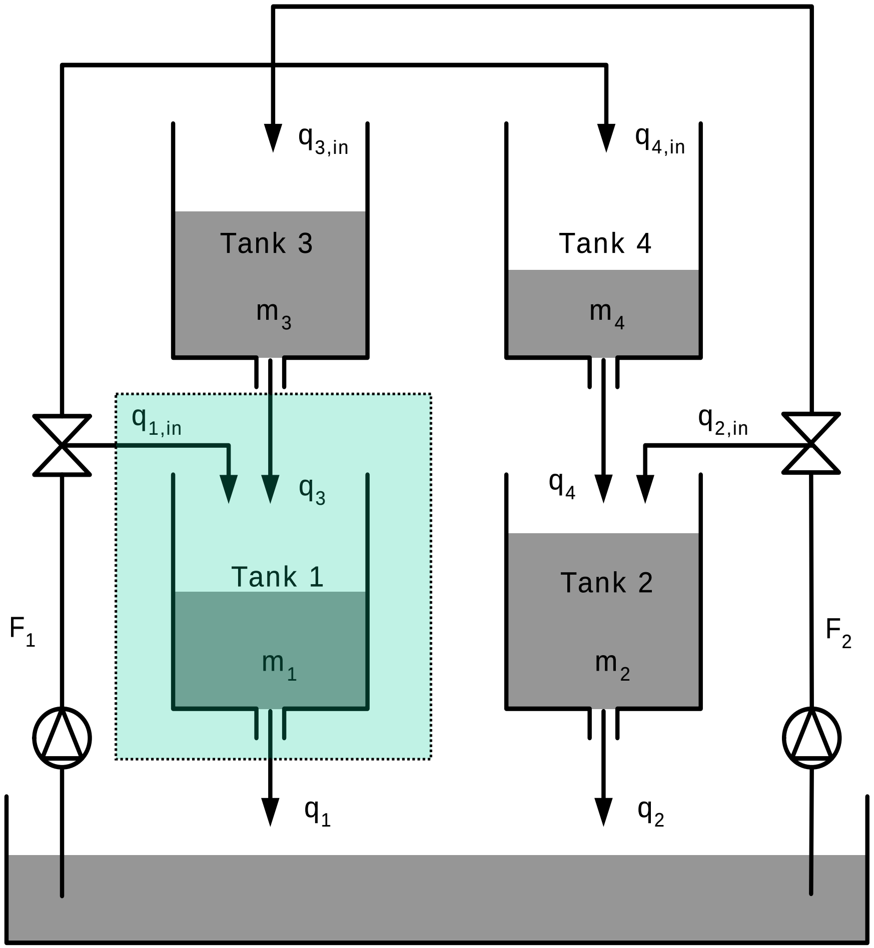
Quadruple Tank Process
Flow Controllers



Quadruple Tank Process



Process Model



m_1 : $[g]$ Mass in tank 1

ρ : $[g/cm^3]$ Density

$q_{1,in}$: $[cm^3/s]$ Flow rate

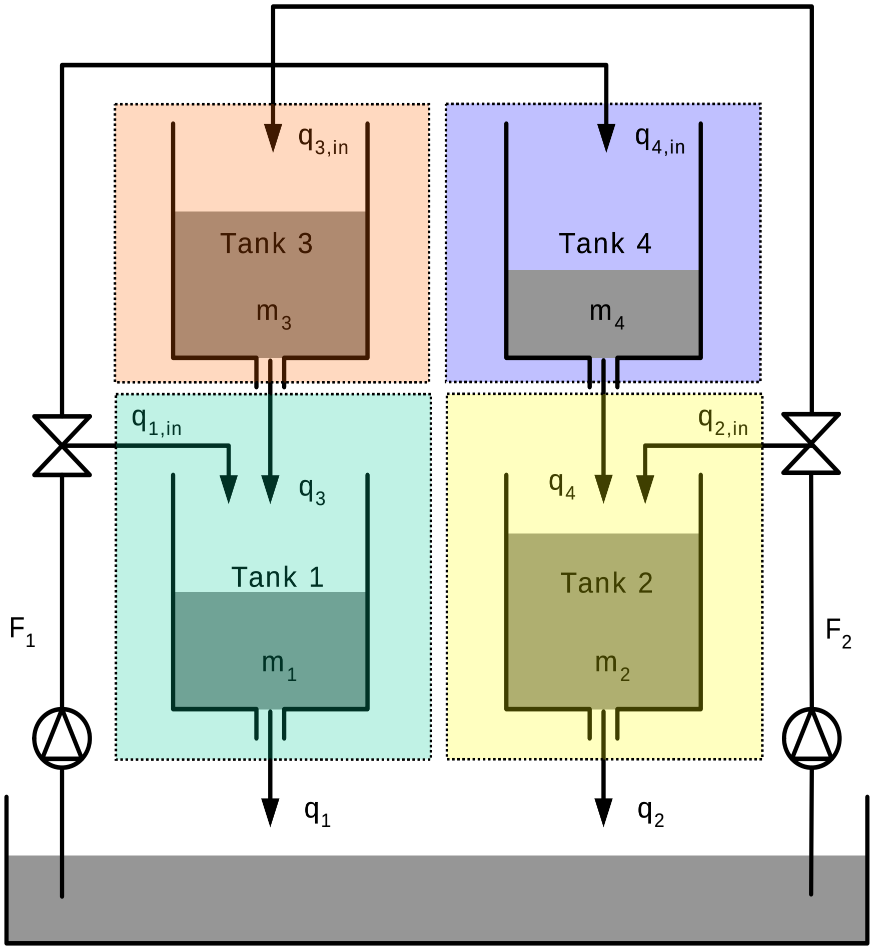
q_1 : $[cm^3/s]$ Flow rate

q_3 : $[cm^3/s]$ Flow rate

Mass Balance. Tank 1

$$\frac{dm_1}{dt} = \rho q_{1,in} + \rho q_3 - \rho q_1$$

Mass Balances



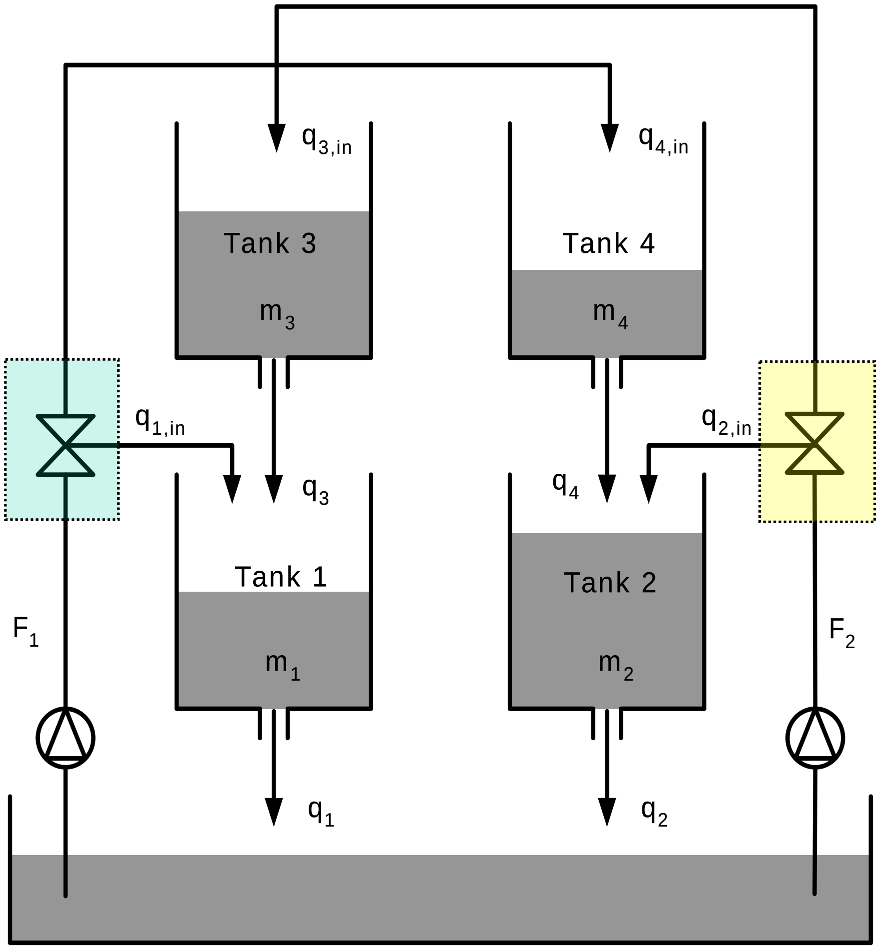
$$\frac{dm_1}{dt} = \rho q_{1,in} + \rho q_3 - \rho q_1$$

$$\frac{dm_2}{dt} = \rho q_{2,in} + \rho q_4 - \rho q_2$$

$$\frac{dm_3}{dt} = \rho q_{3,in} - \rho q_3$$

$$\frac{dm_4}{dt} = \rho q_{4,in} - \rho q_4$$

Distribution of Flows at Valves



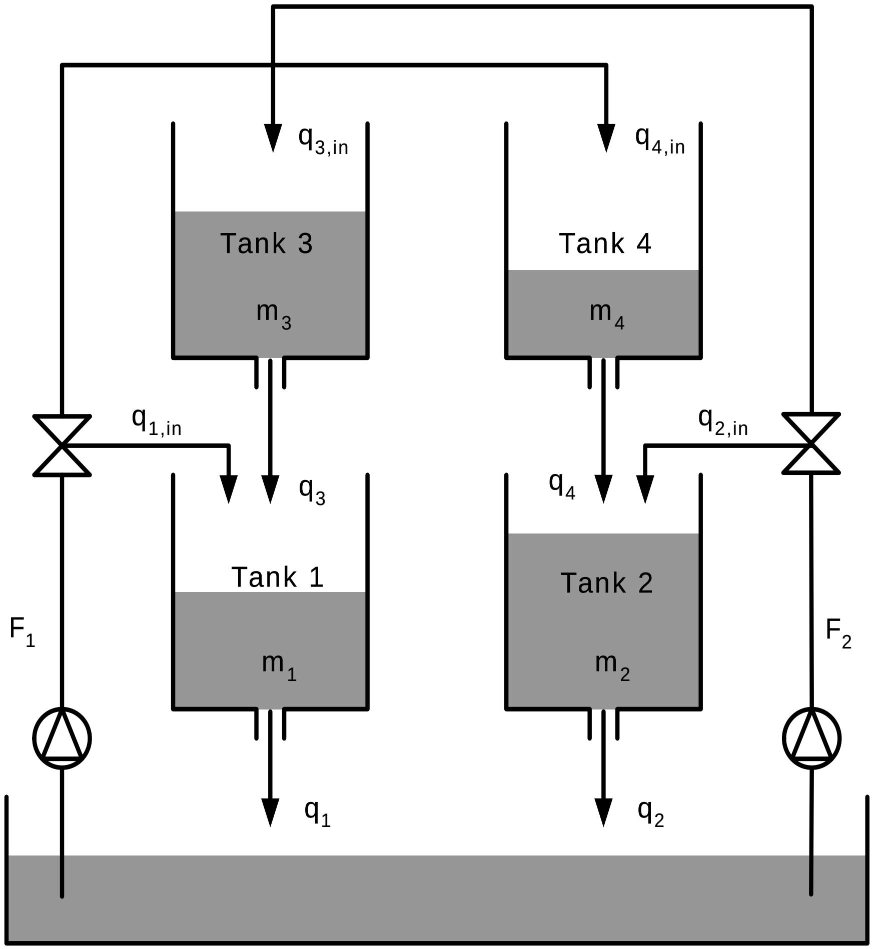
$$q_{1,in} = \gamma_1 F_1$$

$$q_{2,in} = \gamma_2 F_2$$

$$q_{3,in} = (1 - \gamma_2) F_2$$

$$q_{4,in} = (1 - \gamma_1) F_1$$

Bernoulli's Law



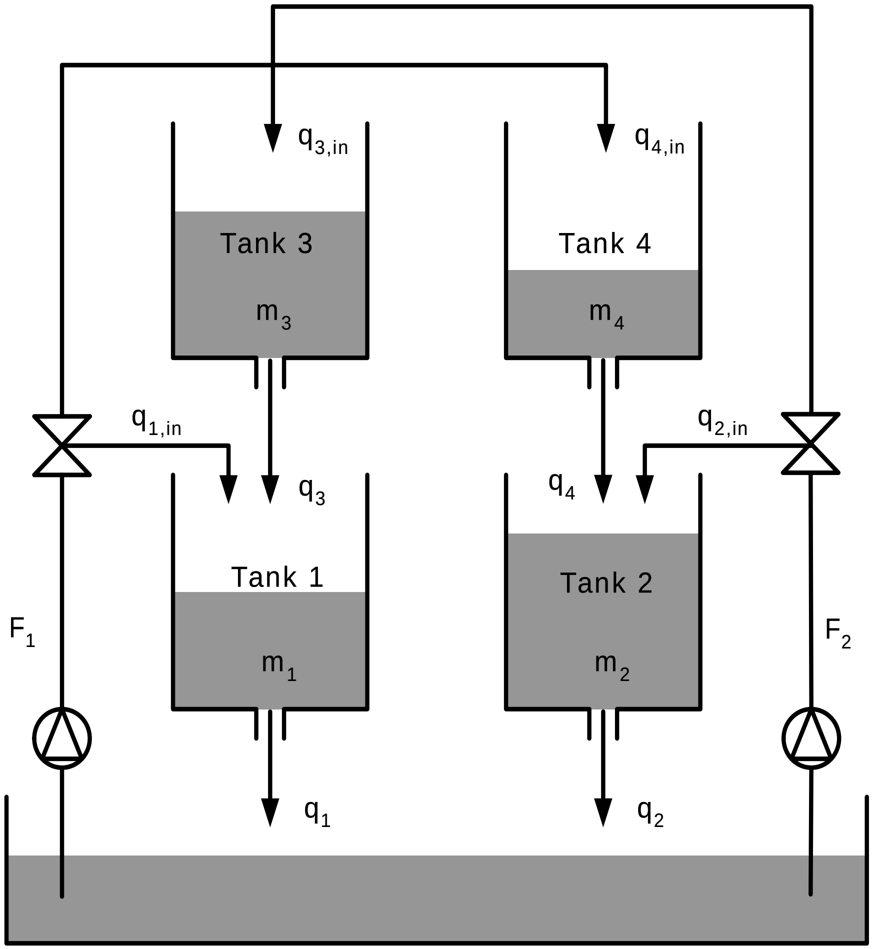
$$q_1 = a_1\sqrt{2gh_1}$$

$$q_2 = a_2\sqrt{2gh_2}$$

$$q_3 = a_3\sqrt{2gh_3}$$

$$q_4 = a_4\sqrt{2gh_4}$$

Process Model



$$m_1 = \rho V_1 = \rho A_1 h_1$$

$$m_2 = \rho V_2 = \rho A_2 h_2$$

$$m_3 = \rho V_3 = \rho A_3 h_3$$

$$m_4 = \rho V_4 = \rho A_4 h_4$$

Process Model

$$\frac{dm_1}{dt} = \rho q_{1,in} + \rho q_3 - \rho q_1$$
$$\frac{dm_2}{dt} = \rho q_{2,in} + \rho q_4 - \rho q_2$$
$$\frac{dm_3}{dt} = \rho q_{3,in} - \rho q_3$$
$$\frac{dm_4}{dt} = \rho q_{4,in} - \rho q_4$$

$$m_1 = \rho A_1 h_1$$
$$m_2 = \rho A_2 h_2$$
$$m_3 = \rho A_3 h_3$$
$$m_4 = \rho A_4 h_4$$

$$q_{1,in} = \gamma_1 F_1$$
$$q_{2,in} = \gamma_2 F_2$$
$$q_{3,in} = (1 - \gamma_2) F_2$$
$$q_{4,in} = (1 - \gamma_1) F_1$$

$$q_1 = a_1 \sqrt{2 g h_1}$$
$$q_2 = a_2 \sqrt{2 g h_2}$$
$$q_3 = a_3 \sqrt{2 g h_3}$$
$$q_4 = a_4 \sqrt{2 g h_4}$$

Process Model

$$\rho A_1 \frac{dh_1}{dt} = \rho q_{1,in} + \rho q_3 - \rho q_1$$
$$\rho A_2 \frac{dh_2}{dt} = \rho q_{2,in} + \rho q_4 - \rho q_2$$
$$\rho A_3 \frac{dh_3}{dt} = \rho q_{3,in} - \rho q_3$$
$$\rho A_4 \frac{dh_4}{dt} = \rho q_{4,in} - \rho q_4$$

$$m_1 = \rho A_1 h_1$$
$$m_2 = \rho A_2 h_2$$
$$m_3 = \rho A_3 h_3$$
$$m_4 = \rho A_4 h_4$$

$$q_{1,in} = \gamma_1 F_1$$
$$q_{2,in} = \gamma_2 F_2$$
$$q_{3,in} = (1 - \gamma_2) F_2$$
$$q_{4,in} = (1 - \gamma_1) F_1$$

$$q_1 = a_1 \sqrt{2gh_1}$$
$$q_2 = a_2 \sqrt{2gh_2}$$
$$q_3 = a_3 \sqrt{2gh_3}$$
$$q_4 = a_4 \sqrt{2gh_4}$$

Process Model

$$A_1 \frac{dh_1}{dt} = q_{1,in} + q_3 - q_1$$
$$A_2 \frac{dh_2}{dt} = q_{2,in} + q_4 - q_2$$
$$A_3 \frac{dh_3}{dt} = q_{3,in} - q_3$$
$$A_4 \frac{dh_4}{dt} = q_{4,in} - q_4$$

$$m_1 = \rho A_1 h_1$$
$$m_2 = \rho A_2 h_2$$
$$m_3 = \rho A_3 h_3$$
$$m_4 = \rho A_4 h_4$$

$$q_{1,in} = \gamma_1 F_1$$
$$q_{2,in} = \gamma_2 F_2$$
$$q_{3,in} = (1 - \gamma_2) F_2$$
$$q_{4,in} = (1 - \gamma_1) F_1$$

$$q_1 = a_1 \sqrt{2gh_1}$$
$$q_2 = a_2 \sqrt{2gh_2}$$
$$q_3 = a_3 \sqrt{2gh_3}$$
$$q_4 = a_4 \sqrt{2gh_4}$$

Process Model

$$\frac{dh_1}{dt} = \frac{q_{1,in} + q_3 - q_1}{A_1}$$
$$\frac{dh_2}{dt} = \frac{q_{2,in} + q_4 - q_2}{A_2}$$
$$\frac{dh_3}{dt} = \frac{q_{3,in} - q_3}{A_3}$$
$$\frac{dh_4}{dt} = \frac{q_{4,in} - q_4}{A_4}$$

$$m_1 = \rho A_1 h_1$$
$$m_2 = \rho A_2 h_2$$
$$m_3 = \rho A_3 h_3$$
$$m_4 = \rho A_4 h_4$$

$$q_{1,in} = \gamma_1 F_1$$
$$q_{2,in} = \gamma_2 F_2$$
$$q_{3,in} = (1 - \gamma_2) F_2$$
$$q_{4,in} = (1 - \gamma_1) F_1$$

$$q_1 = a_1 \sqrt{2 g h_1}$$
$$q_2 = a_2 \sqrt{2 g h_2}$$
$$q_3 = a_3 \sqrt{2 g h_3}$$
$$q_4 = a_4 \sqrt{2 g h_4}$$

Process Model

$$\rho A_1 \frac{dh_1}{dt} = \rho\gamma_1 F_1 + \rho a_3\sqrt{2gh_3} - \rho a_1\sqrt{2gh_1}$$

$$\rho A_2 \frac{dh_2}{dt} = \rho\gamma_2 F_2 + \rho a_4\sqrt{2gh_4} - \rho a_2\sqrt{2gh_2}$$

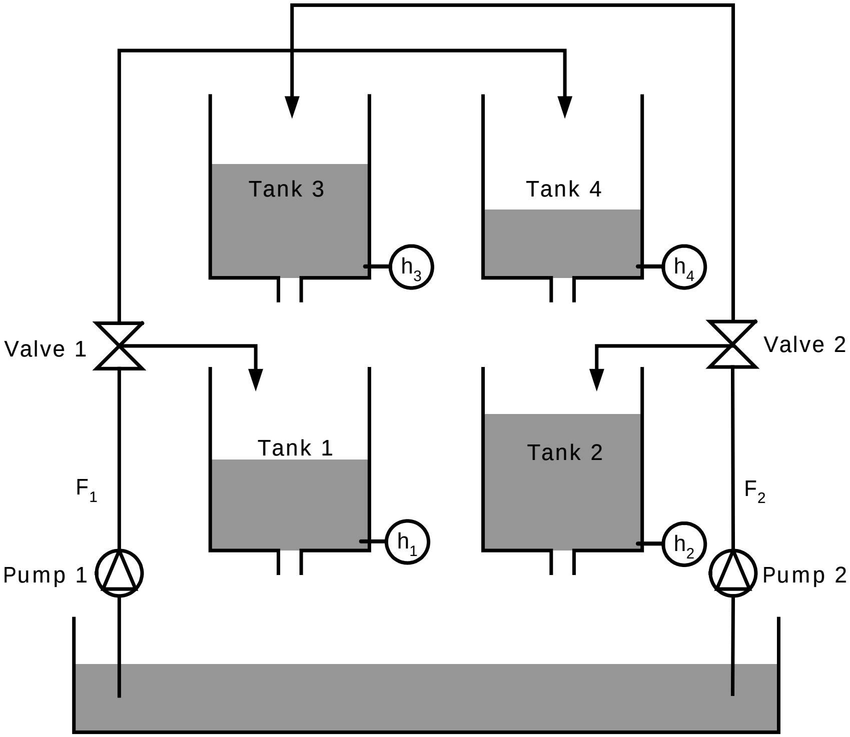
$$\rho A_3 \frac{dh_3}{dt} = \rho(1-\gamma_2) F_2 - \rho a_3\sqrt{2gh_3}$$

$$\rho A_4 \frac{dh_4}{dt} = \rho(1-\gamma_1) F_1 - \rho a_4\sqrt{2gh_4}$$

Process Model

$$\frac{dh_1}{dt} = \frac{\gamma_1}{A_1}F_1 + \frac{a_3}{A_1}\sqrt{2gh_3} - \frac{a_1}{A_1}\sqrt{2gh_1}$$
$$\frac{dh_2}{dt} = \frac{\gamma_2}{A_2}F_2 + \frac{a_4}{A_2}\sqrt{2gh_4} - \frac{a_2}{A_2}\sqrt{2gh_2}$$
$$\frac{dh_3}{dt} = \frac{1-\gamma_2}{A_3}F_2 - \frac{a_3}{A_3}\sqrt{2gh_3}$$
$$\frac{dh_4}{dt} = \frac{1-\gamma_1}{A_4}F_1 - \frac{a_4}{A_4}\sqrt{2gh_4}$$

Process Model



$$\frac{dh_1}{dt} = \frac{\gamma_1}{A_1}F_1 + \frac{a_3}{A_1}\sqrt{2gh_3} - \frac{a_1}{A_1}\sqrt{2gh_1}$$
$$\frac{dh_2}{dt} = \frac{\gamma_2}{A_2}F_2 + \frac{a_4}{A_2}\sqrt{2gh_4} - \frac{a_2}{A_2}\sqrt{2gh_2}$$
$$\frac{dh_3}{dt} = \frac{1-\gamma_2}{A_3}F_2 - \frac{a_3}{A_3}\sqrt{2gh_3}$$
$$\frac{dh_4}{dt} = \frac{1-\gamma_1}{A_4}F_1 - \frac{a_4}{A_4}\sqrt{2gh_4}$$

$$\dot{x}(t) = f(x(t), u(t))$$
$$x(t_0) = x_0$$

$$x = \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \end{bmatrix} \qquad u = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix}$$

Process Simulation with Matlab

$$\frac{dh_1}{dt} = \frac{q_{1,in} + q_3 - q_1}{A_1}$$
$$\frac{dh_2}{dt} = \frac{q_{2,in} + q_4 - q_2}{A_2}$$
$$\frac{dh_3}{dt} = \frac{q_{3,in} - q_3}{A_3}$$
$$\frac{dh_4}{dt} = \frac{q_{4,in} - q_4}{A_4}$$

$$q_{1,in} = \gamma_1 F_1$$
$$q_{2,in} = \gamma_2 F_2$$
$$q_{3,in} = (1 - \gamma_2) F_2$$
$$q_{4,in} = (1 - \gamma_1) F_1$$

$$q_1 = a_1 \sqrt{2gh_1}$$
$$q_2 = a_2 \sqrt{2gh_2}$$
$$q_3 = a_3 \sqrt{2gh_3}$$
$$q_4 = a_4 \sqrt{2gh_4}$$

- **Define the model by**

```
function xdot = QuadrupleTankProcess(t,x,u,p)
```

- **Solve the differential equations using**

```
[T,X]=ode15s(@QuadrupleTankProcess,...
                  [t0 tf], x0, ODEoptions, u, p)
```

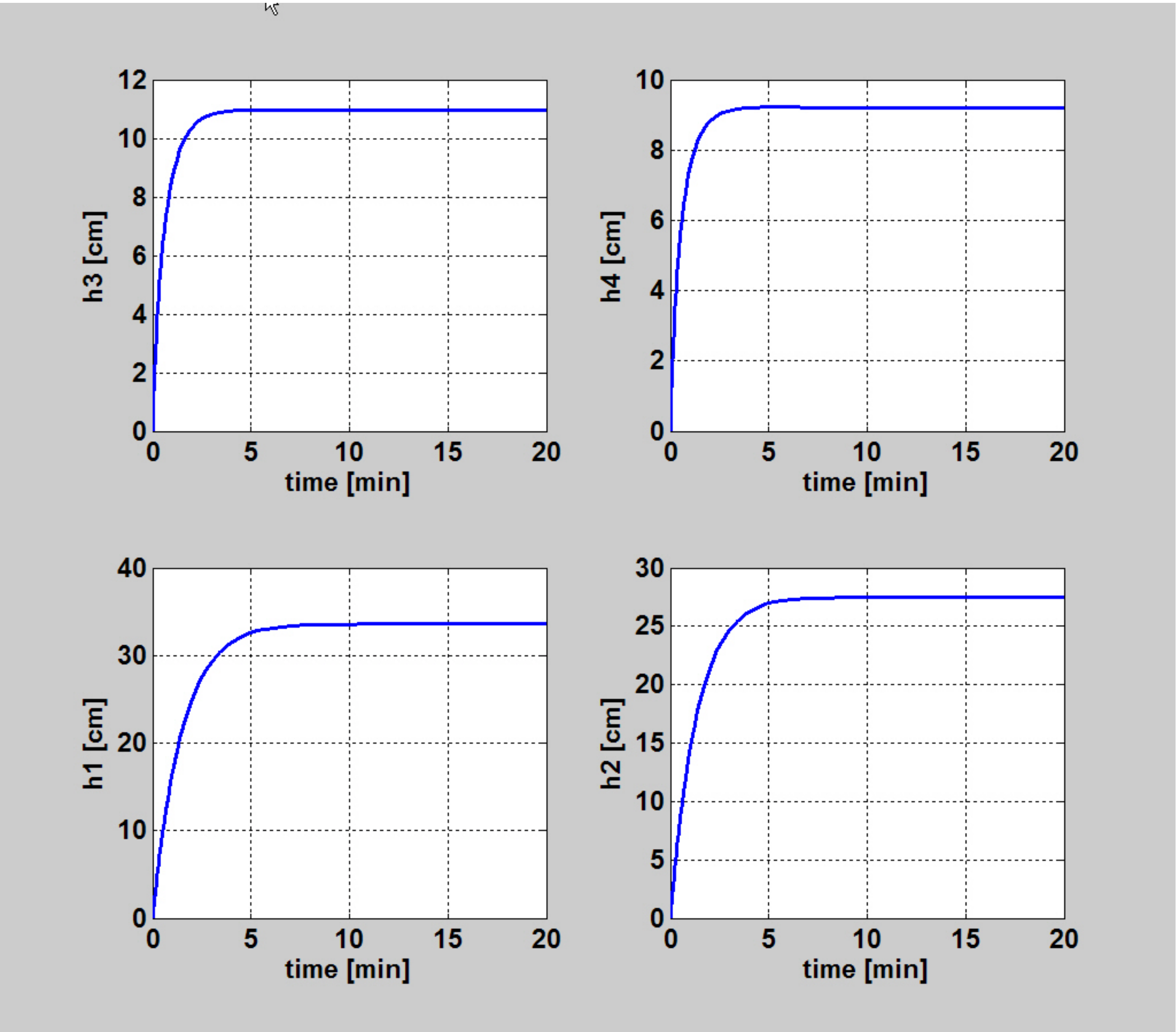
Parameters

```
% ------------------------------------------------------------
% Parameters
% ------------------------------------------------------------
a1 = 1.2272     %[cm2] Area of outlet pipe 1
a2 = 1.2272     %[cm2] Area of outlet pipe 2
a3 = 1.2272     %[cm2] Area of outlet pipe 3
a4 = 1.2272     %[cm2] Area of outlet pipe 4

A1 = 380.1327   %[cm2] Cross sectional area of tank 1
A2 = 380.1327   %[cm2] Cross sectional area of tank 2
A3 = 380.1327   %[cm2] Cross sectional area of tank 3
A4 = 380.1327   %[cm2] Cross sectional area of tank 4

g = 981         %[cm/s2] The acceleration of gravity

gamma1 = 0.45;  % Flow distribution constant. Valve 1
gamma2 = 0.40;  % Flow distribution constant. Valve 2

p = [a1; a2; a3; a4; A1; A2; A3; A4; g; gamma1; gamma2];
```

Simulation Scenario

```
% ------------------------------------------------------------
% Simulation scenario
% ------------------------------------------------------------
t0 =    0.0;          % [s] Initial time
tf = 20*60;           % [s] Final time

h10 = 0.0;            % [cm] Liquid level in tank 1 at time t0
h20 = 0.0;            % [cm] Liquid level in tank 2 at time t0
h30 = 0.0;            % [cm] Liquid level in tank 3 at time t0
h40 = 0.0;            % [cm] Liquid level in tank 4 at time t0

F1 = 300;             % [cm3/s] Flow rate from pump 1
F2 = 300;             % [cm3/s] Flow rate from pump 2

x0 = [h10; h20; h30; h40];
u = [F1; F2];
```

Start-Up Simulation



F1 = 300 cm3/s

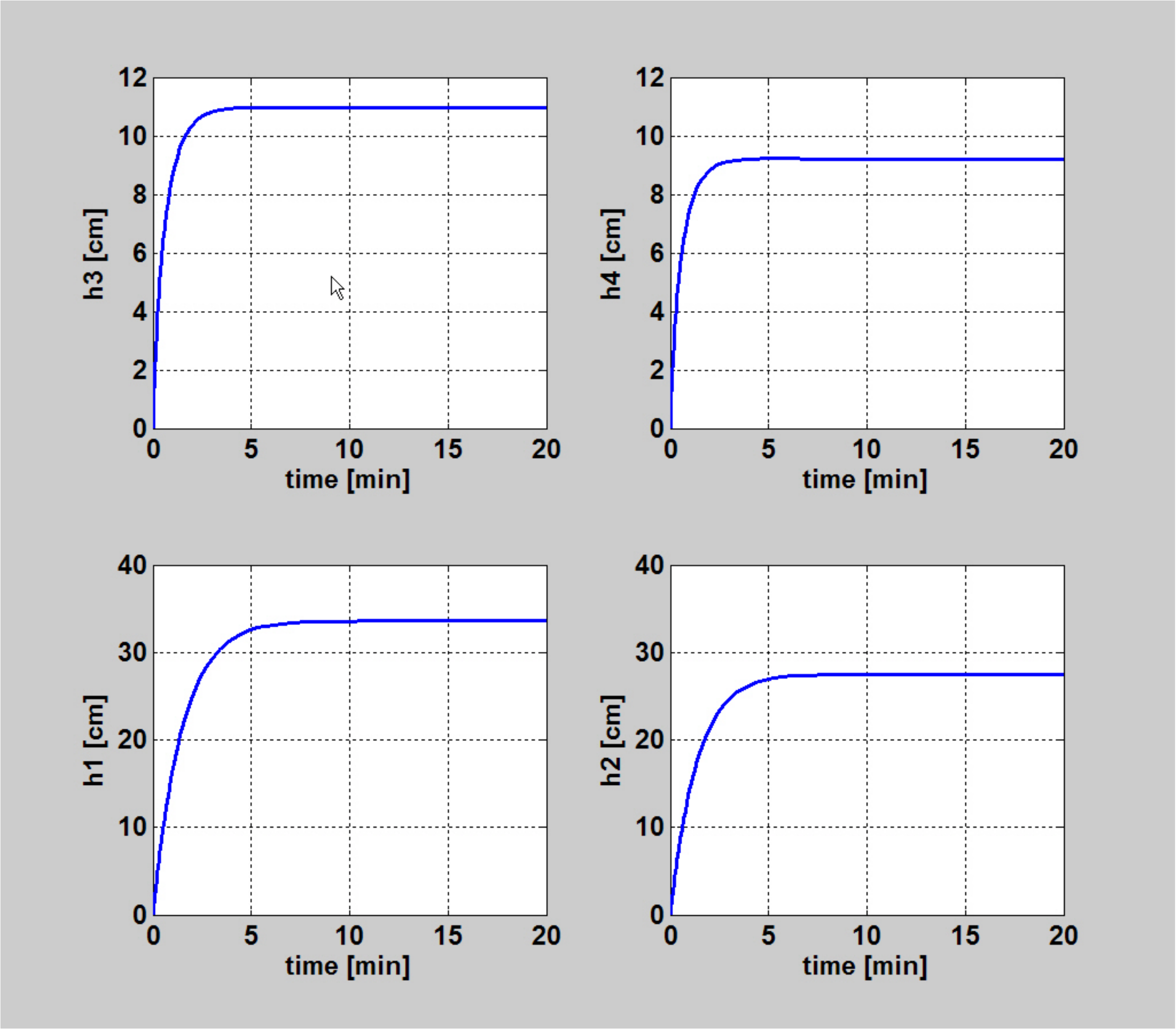
F2 = 300 cm3/s

gamma1 = 0.45

gamma2 = 0.40

Non-Minimum Phase System

Start-Up Simulation



F1 = 300 cm3/s

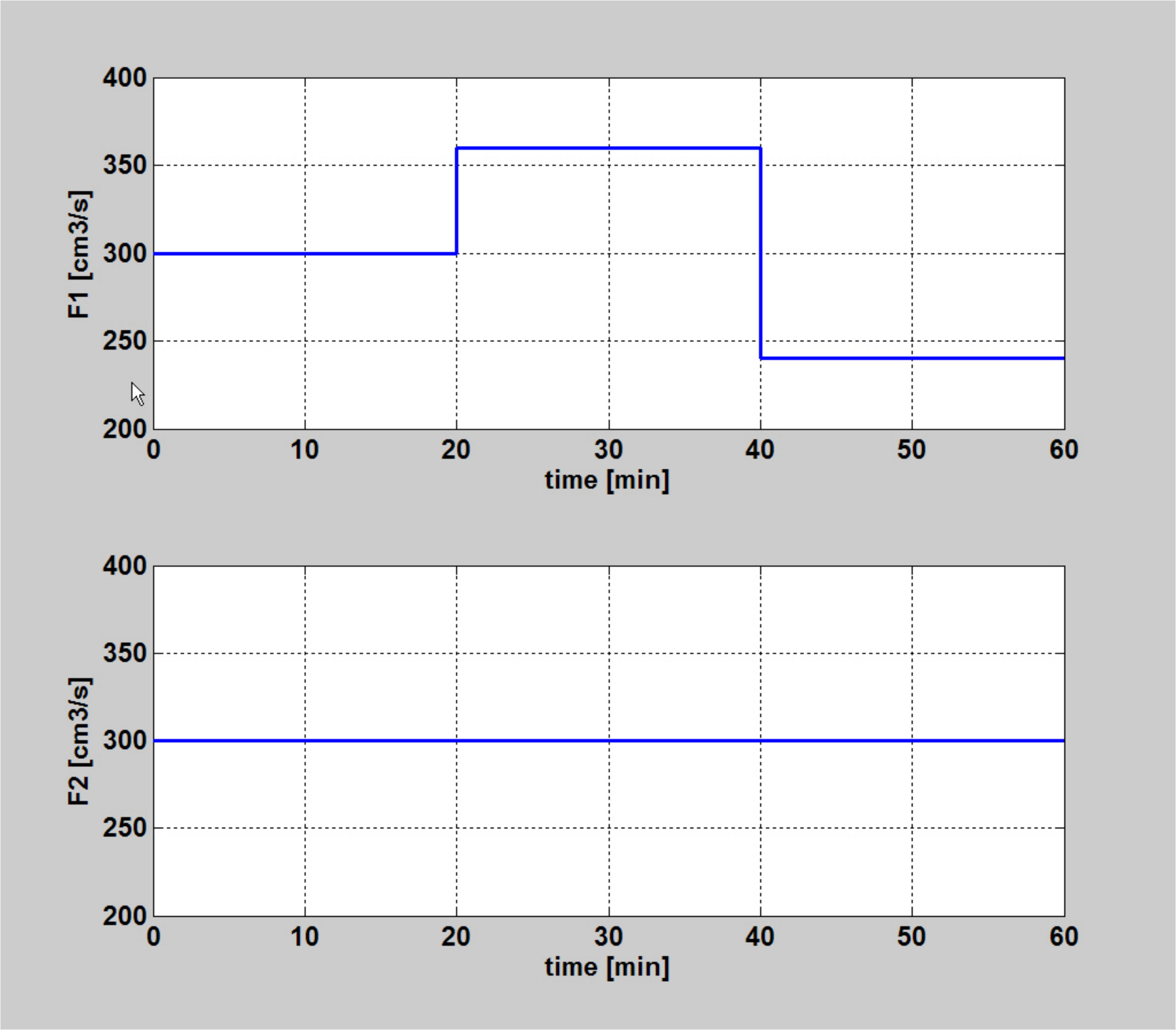
F2 = 300 cm3/s

gamma1 = 0.45

gamma2 = 0.40

Non-Minimum Phase System

Simulation. Steps in F1



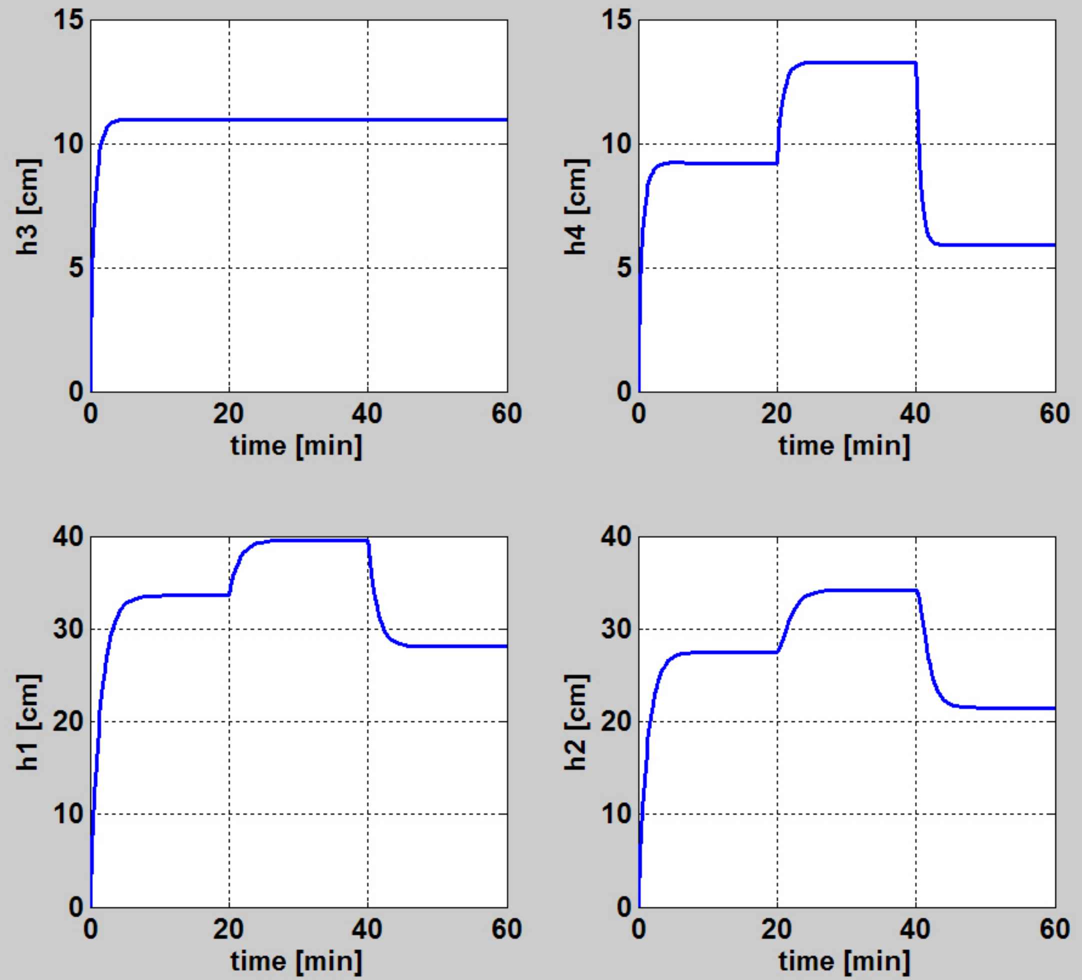
F1 = 300 cm3/s

F2 = 300 cm3/s

gamma1 = 0.45

gamma2 = 0.40

Simulation. Steps in F1



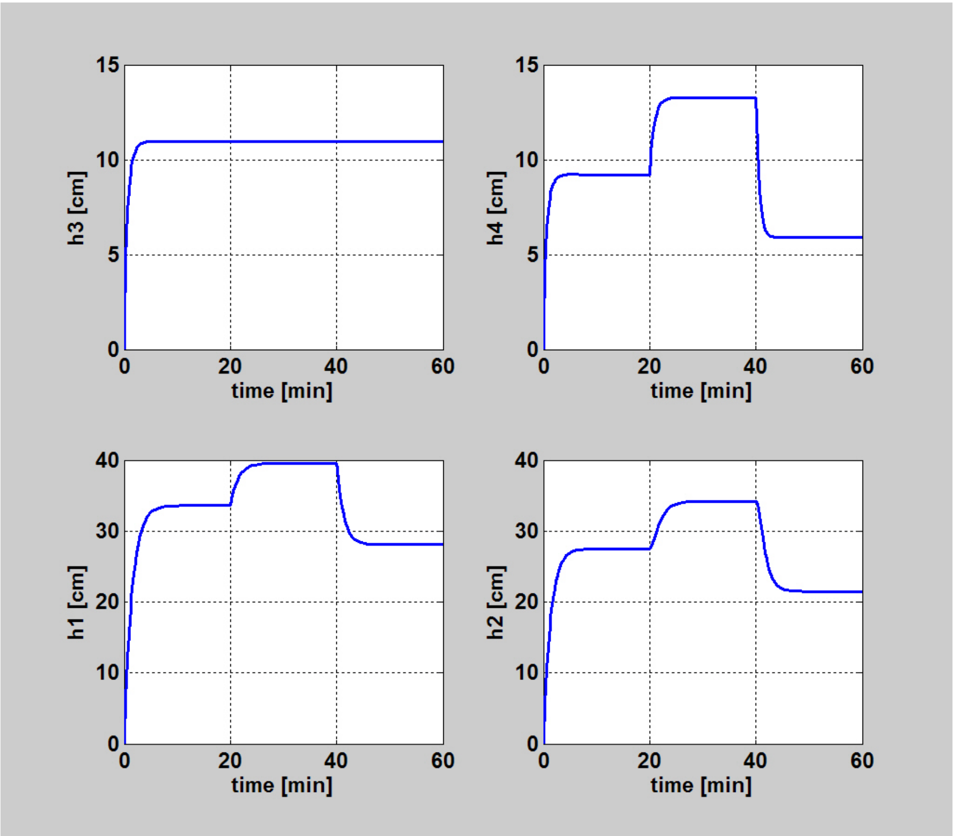
F1 = 300 cm3/s

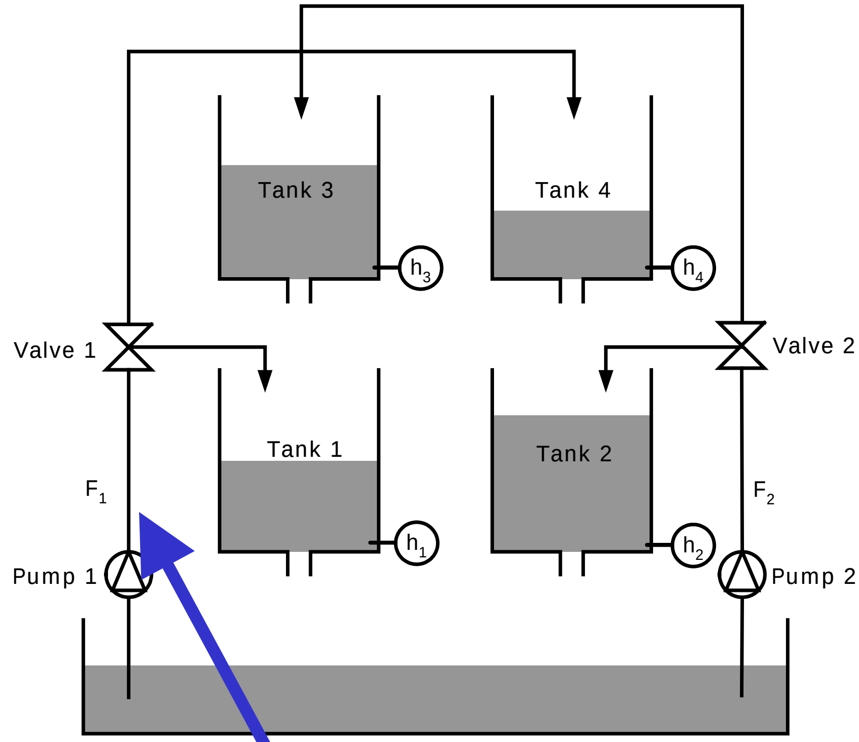
F2 = 300 cm3/s

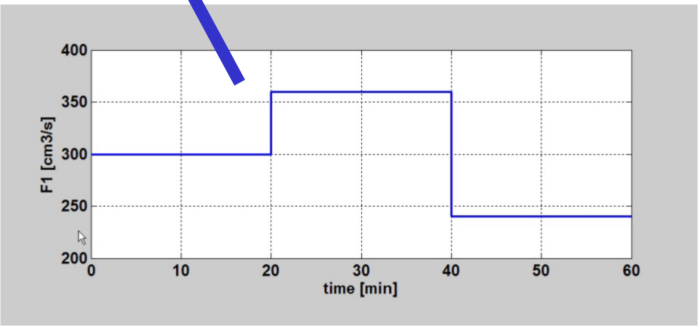
gamma1 = 0.45

gamma2 = 0.40

Simulation. Steps in F1



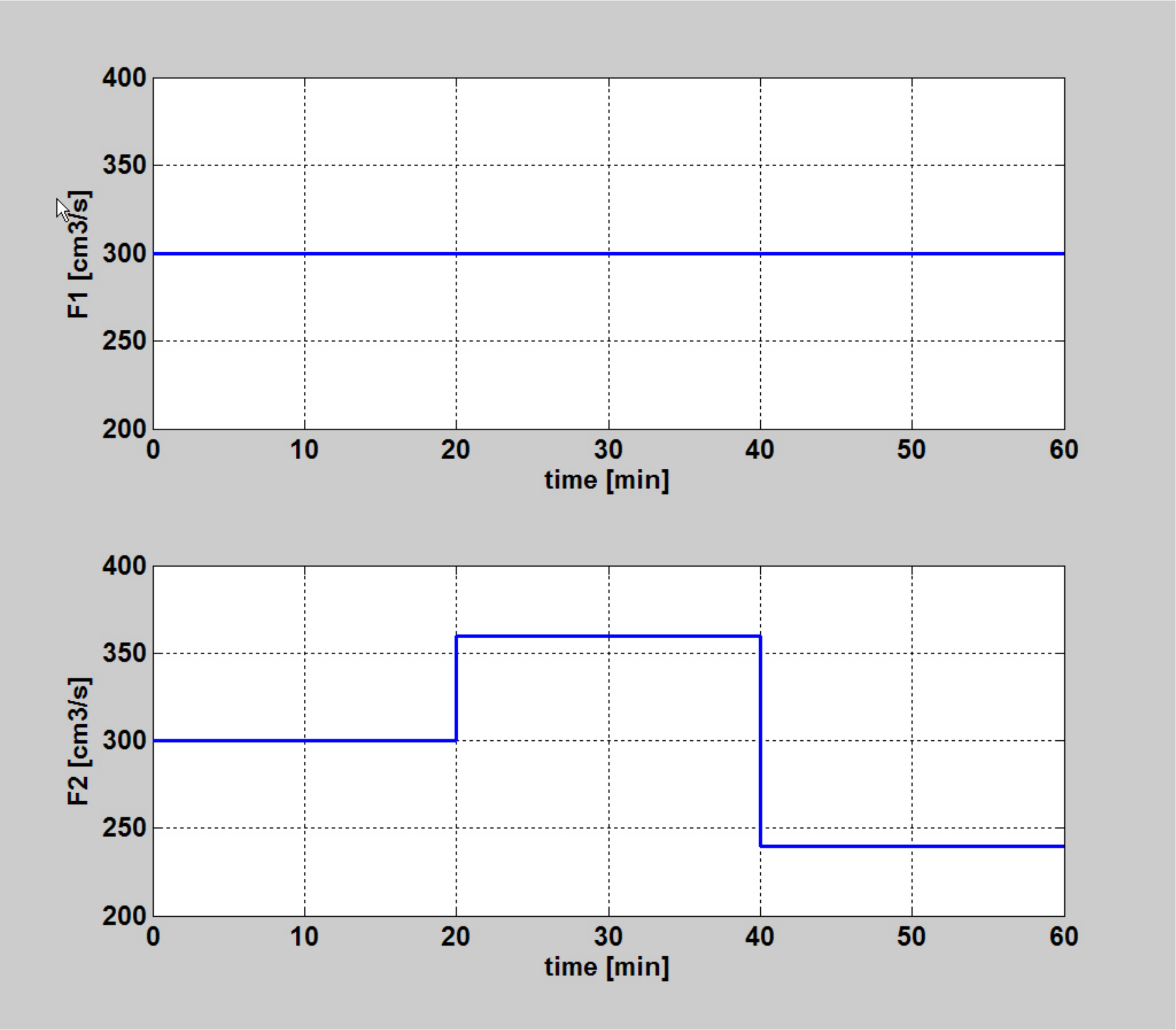




F1 = 300 cm3/s F2 = 300 cm3/s

gamma1 = 0.45 gamma2 = 0.40

Simulation. Steps in F2



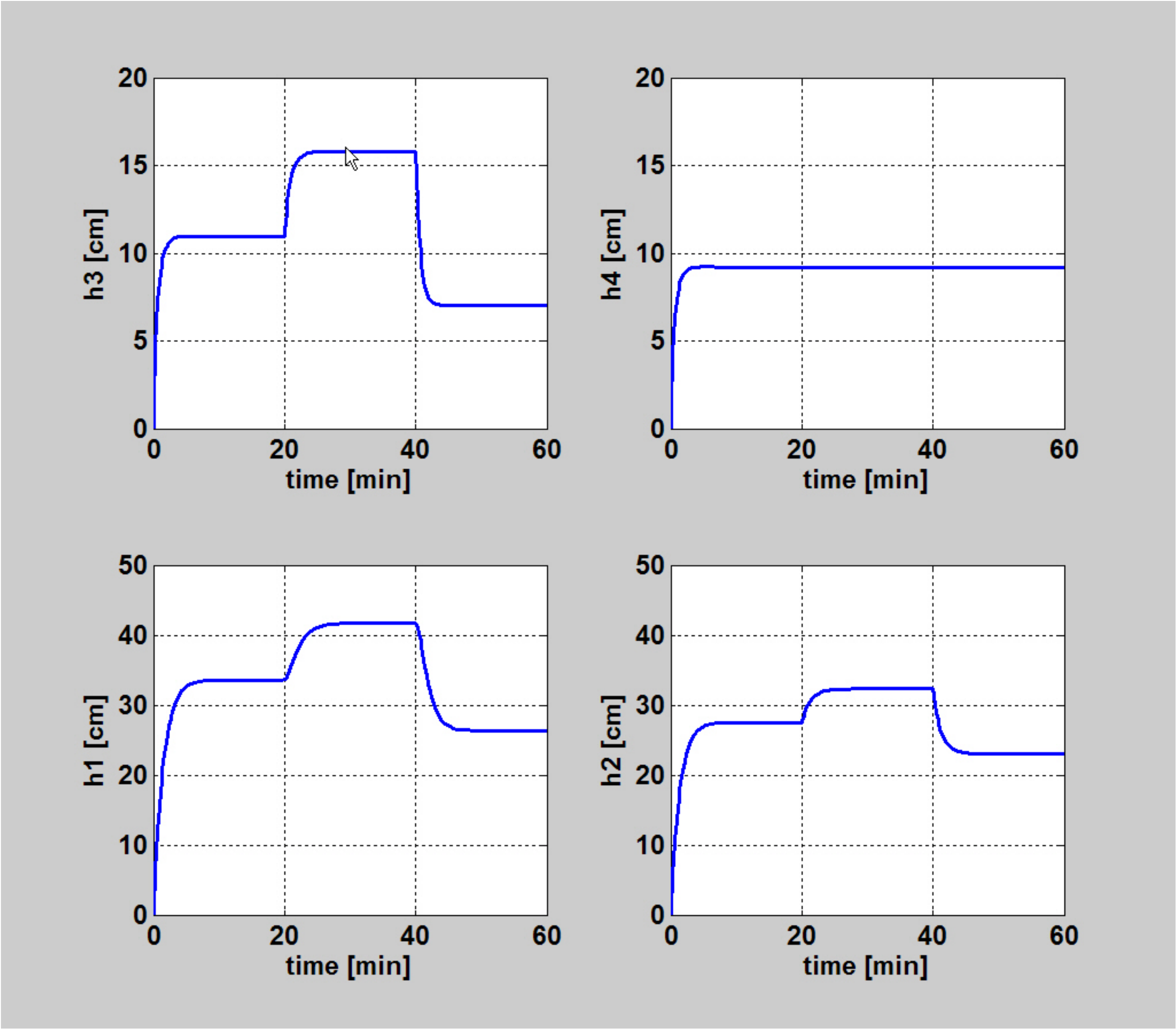
F1 = 300 cm3/s

F2 = 300 cm3/s

gamma1 = 0.45

gamma2 = 0.40

Simulation. Steps in F2



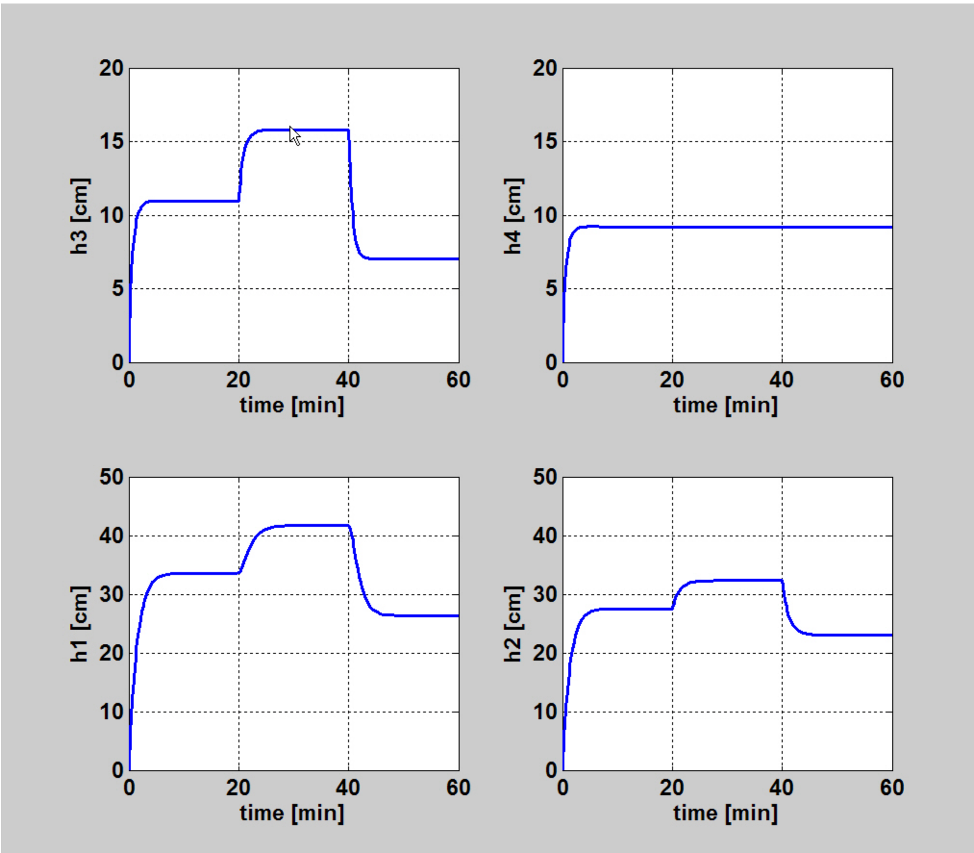
F1 = 300 cm3/s

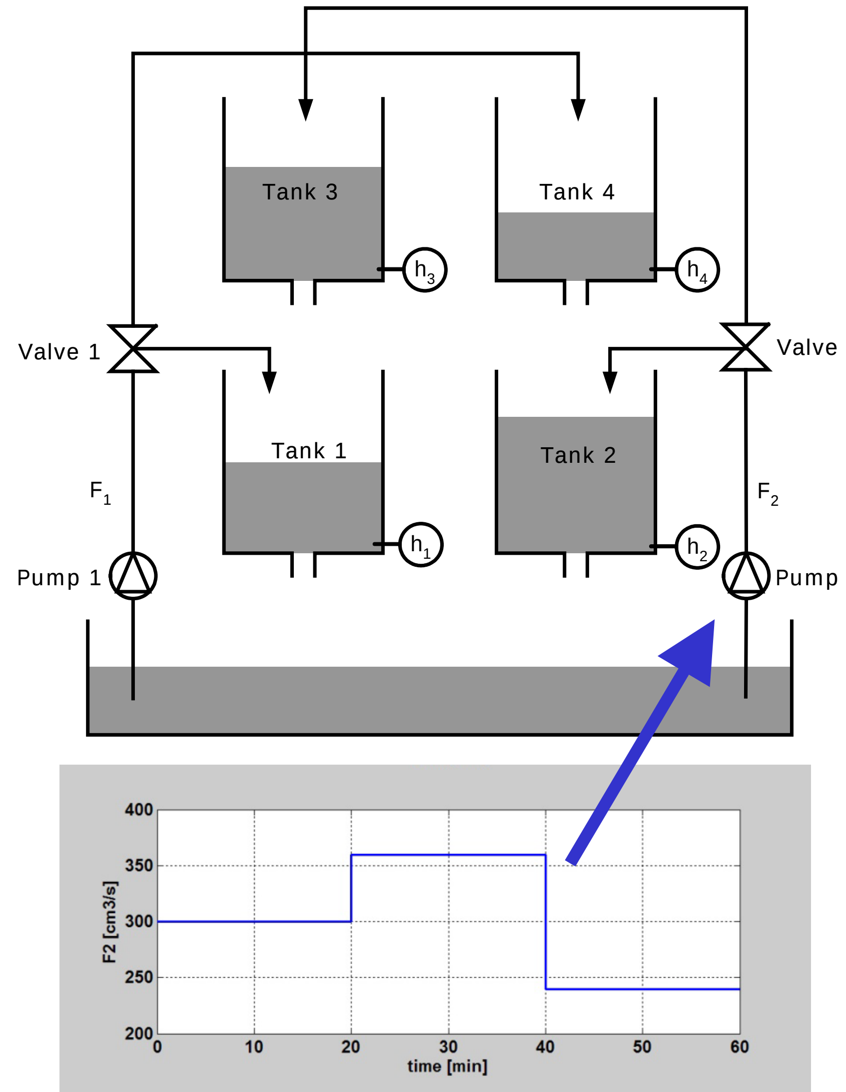
F2 = 300 cm3/s

gamma1 = 0.45

gamma2 = 0.40

Simulation. Steps in F2

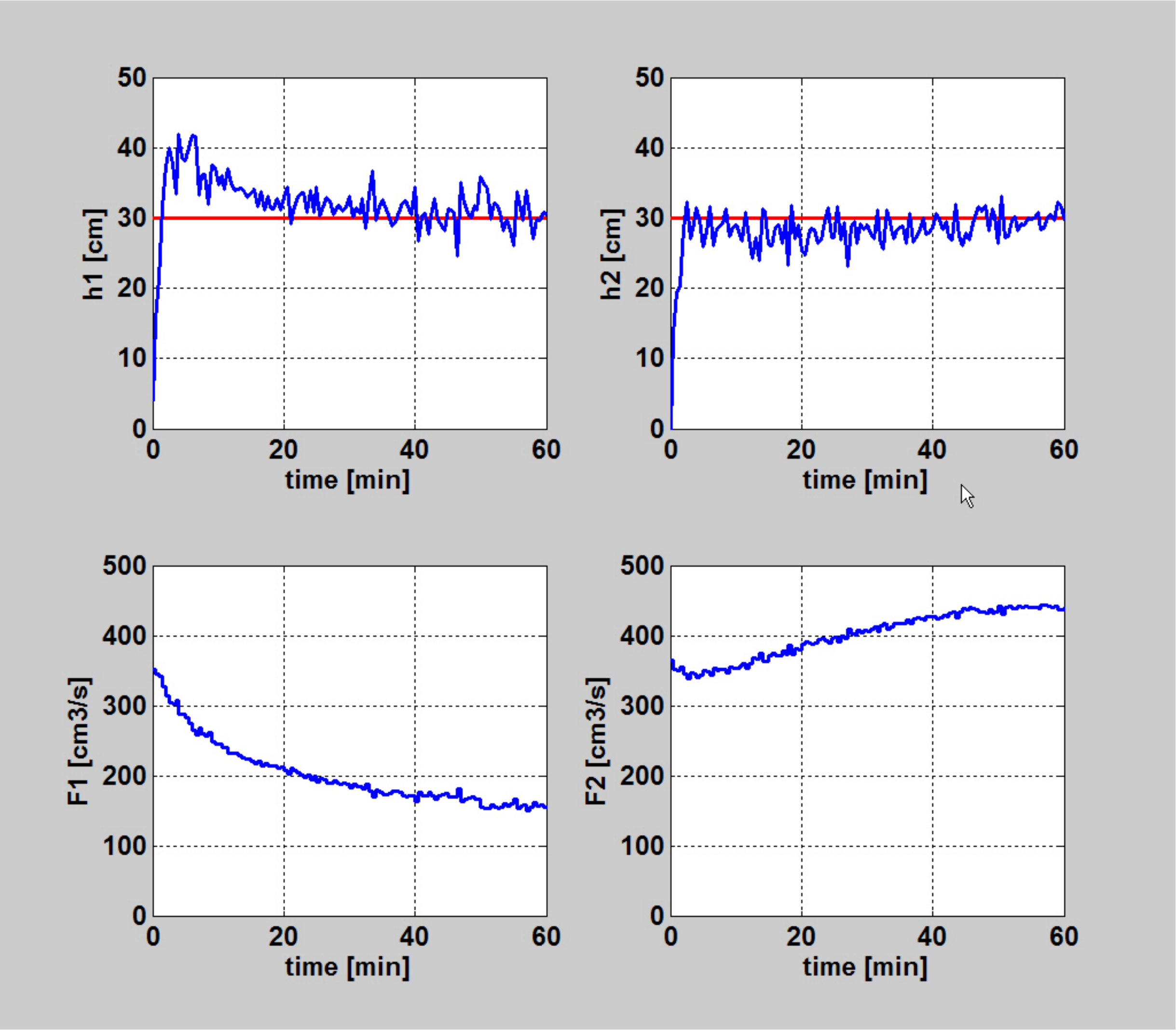




F1 = 300 cm3/s   F2 = 300 cm3/s

gamma1 = 0.45   gamma2 = 0.40

Closed Loop. PI-Controllers
Minimum Phase



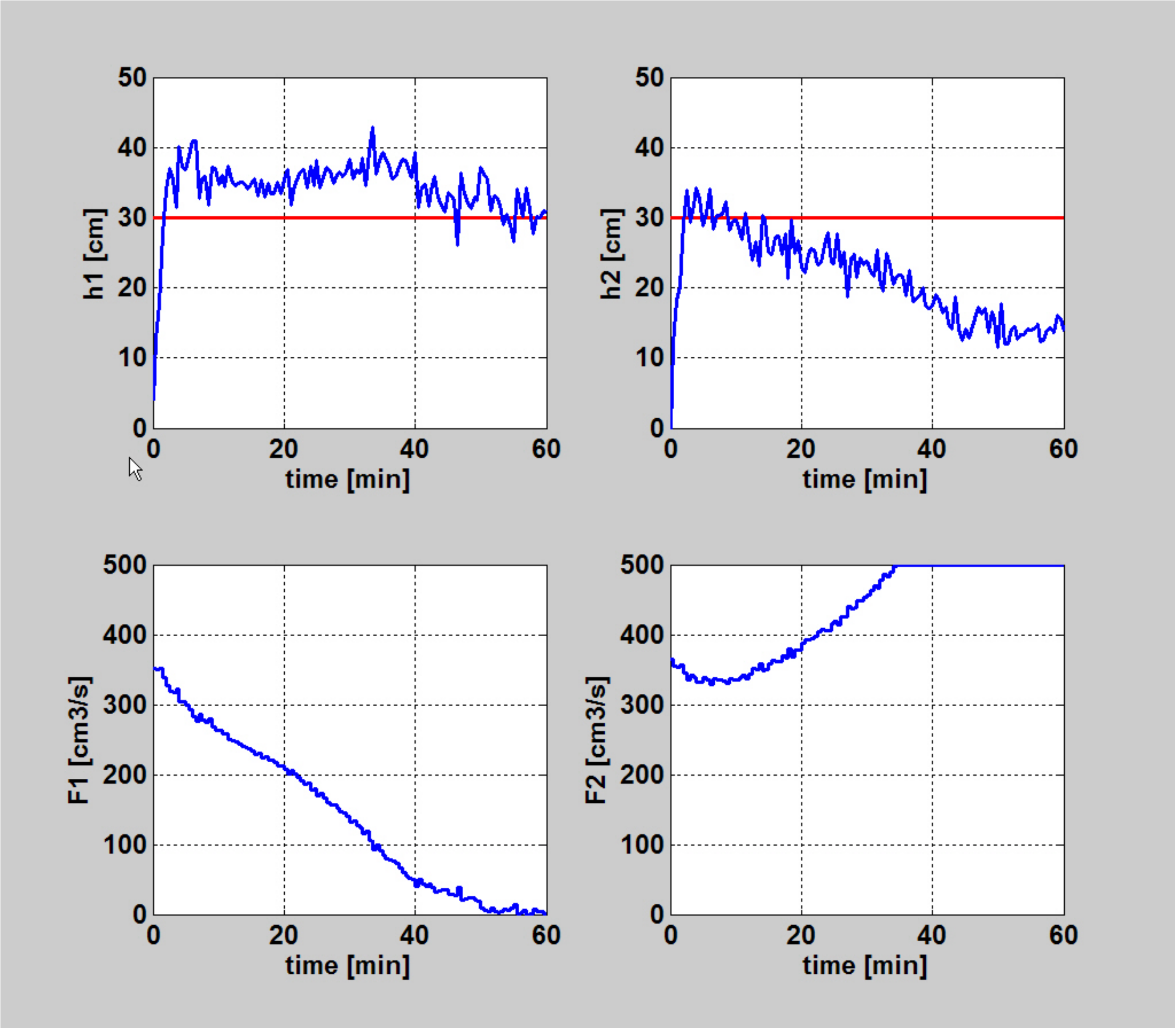
F1s = 300 cm3/s

F2s = 300 cm3/s

gamma1 = 0.65

gamma2 = 0.55

Closed Loop. PI-Controllers Non-Minimum Phase



F1s = 300 cm3/s

F2s = 300 cm3/s

gamma1 = 0.45

gamma2 = 0.40

Learning Objectives

- Lecture #1 will enable you to
  - Describe the components in a computer controlled system.
  - Identify, describe and analyze a control structure in terms of CVs, MVs and DVs.
  - Model and simulate a process system consisting of differential equations
  - Simulate a stochastic system
  - Simulate a deterministic/stochastic systems with digital PI-controllers in the loop.

Exercise Assignment #1A

- Implementation the quadruple tank process in Matlab (Make a process model, and a function for simulation of discrete time systems)
- Simulate the quadruple tank process using Matlab
- Try to design a PI-controller for the non-minimum phase system that can start and stabilize the 4-tank system. Is that possible? Can you explain why it is difficult to control the non-minimum phase system with decentralized PI-controllers?