Week 2: Control Basics & PLCs

Dr. Philipp Sommer

Spring 2025

The material of this course has been initially created by Prof. Dr. H. Kirrmann and adapted by Dr. Y-A. Pignolet, J-C. Tournier & Philipp Sommer.

Agenda

Today:

- **Basics of control**
- **Programmable Logic Controllers (PLCs)**

EPFL

Basics of Control

Motivation for this chapter

- This is an intuitive introduction to control as a preparation for the PLC programming workshop at Siemens, intended for students who have not attended control courses previously.
- For a correct engineering approach, dedicated courses are recommended.
- Content of this chapter:
  - modeling of plants
  - two-point controller
  - PID controller
  - nested controllers

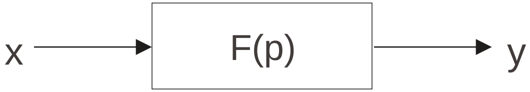
EPFL

Modeling

1. Analysis of control systems
2. Define a controller that meets physical and economical requirements

- The first step is to get to know the plant, i.e., express the plant's behavior in a mathematical way, generally as a system of differential equations:
  - White box approach: analyzing physical principles (requires that all elements are known)
  - Black box approach: identifying the plant's parameters by analyzing its behavior (output) in response to an input change.

Continuous plants



Examples: drives, ovens, chemical reactors

- Continuous (analog) variables (temperature, voltage, speed,...).
- Input/output relation: transfer function, described by differential equations
- Conditions necessary for control:
  - Reversible: output can be brought back to previous value by acting on input
  - Monotone: increasing input causes output to react monotonically
- Linear system: Laplace Transformation from time to frequency domain (simpler notation and computation)

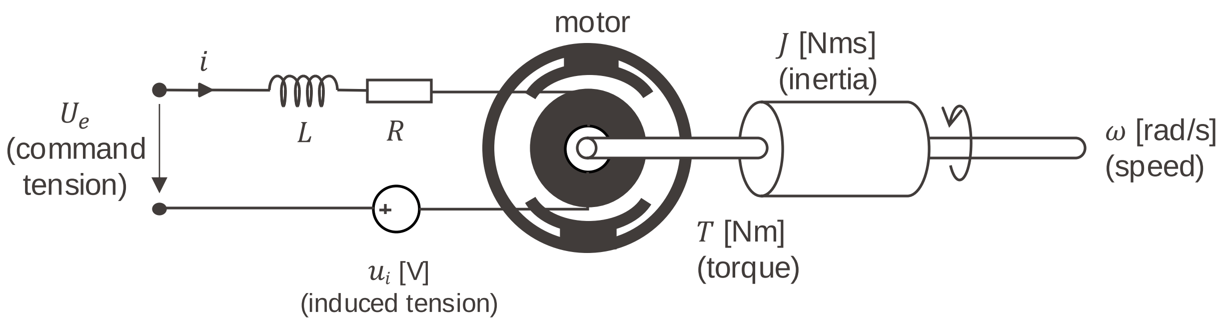
Laplace transformation: $L[f(t)] := \int_{-0}^{\infty} f(t)e^{-st}dt$

$$f(t) \Rightarrow G(s), \text{ where } s = \sigma + j\omega.$$

Main goal: maintain the state on given level or trajectory

Example of a linear model: electrical motor with permanent magnet

Not for exam, illustration only



$$U_e = Ri + L\frac{di}{dt} + u_i$$

$$u_i = K\,\omega$$

$$T = K\,i$$

$$\frac{d\omega}{dt} = \frac{T}{J}$$

$$\frac{di}{dt} = \frac{1}{L}(U_e - K\omega - Ri)$$

$$\frac{d\omega}{dt} = \frac{K}{J}i$$

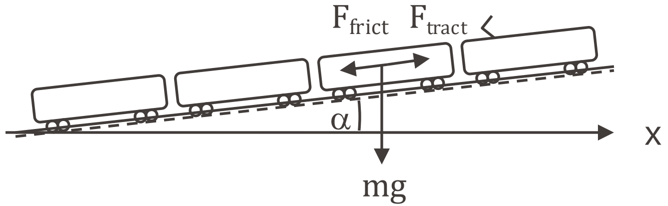
$$\frac{\omega}{U_e} = \frac{K}{s^2(LJ) + s(RJ) + K^2}$$

Laplace transform (since the plant is linear)

Example of non-linear model: train

Not for exam, illustration only

Obtain the relation between applied motor force (current) and the position of a train.



$$\begin{cases} \dfrac{dx}{dt} = v \\ \dfrac{dv}{dt} = \dfrac{1}{m\rho}\left(F_{tract} - mg\sin(\alpha) - m\dfrac{K_c}{radius} - C_x v^2 - C_f v\right) \end{cases}$$

motor force

slope

mass of the train plus contribution of rotating parts (wheels and rotors)

curve friction

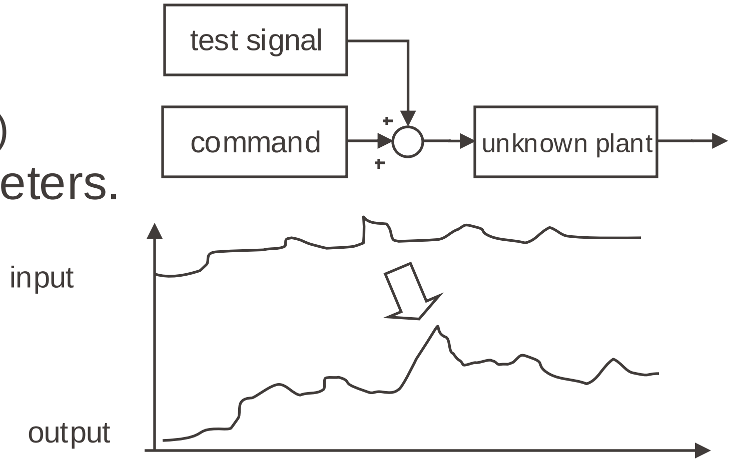
air friction

mechanical friction

Plant Identification

- Once model is approximately known, parameters must be determined by measurements.

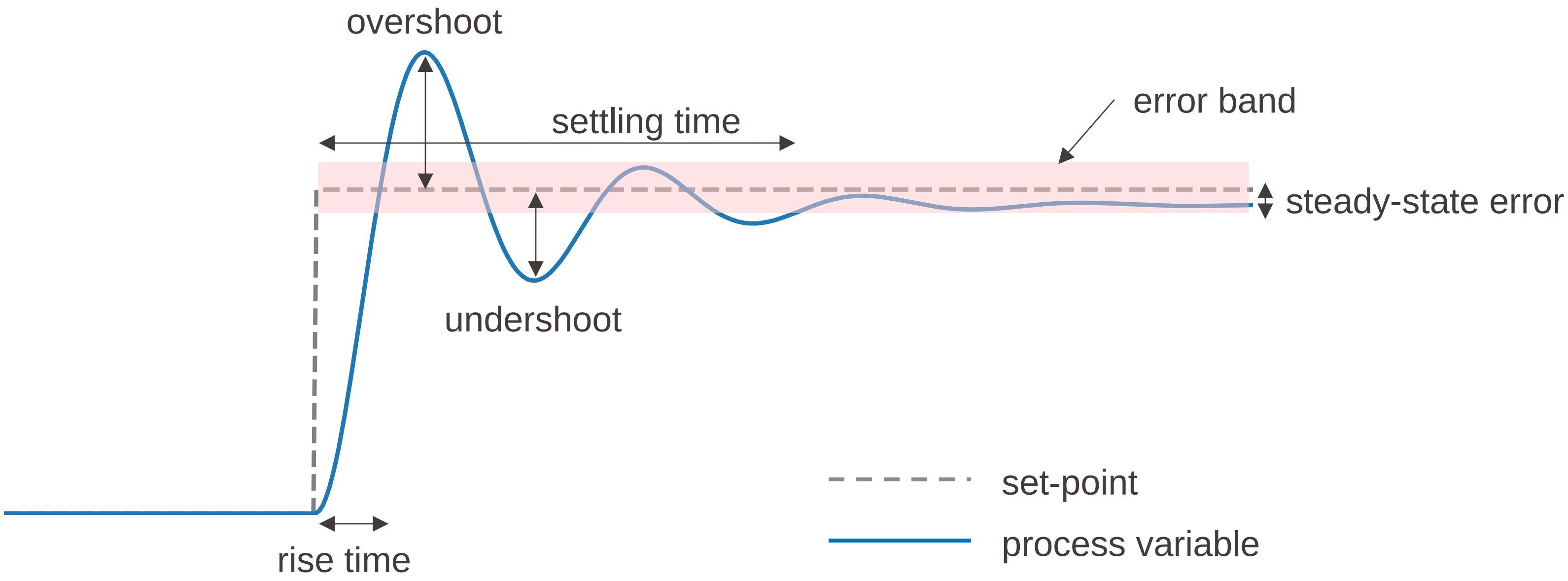
- Classical methods:
  - Response to a pulse at input
  - Response to calibrated noise at input (in case the command signal varies little)
- Signal correlation then yields the parameters.



Control Mechanism

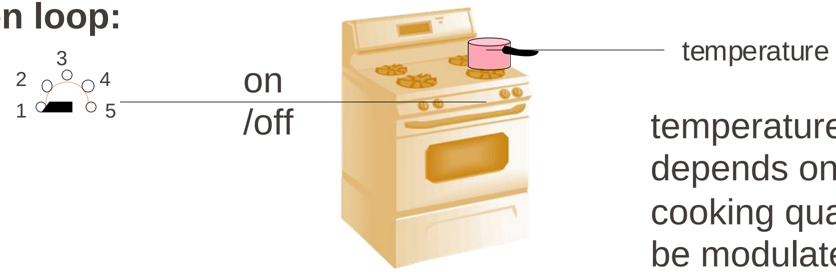
- When plant is known, a controller can be designed.
- In practice, plant parameters vary (e.g., # passengers in train), and the plant is subject to disturbances (wind, slope).
- Controller
  - needs to measure through sensors the state of the plant, and if possible, the disturbances.
  - follows certain quality laws to stabilize the output within useful time, not overshoot, minimize energy consumption, etc.

Response to a change in the set-point (step-response)



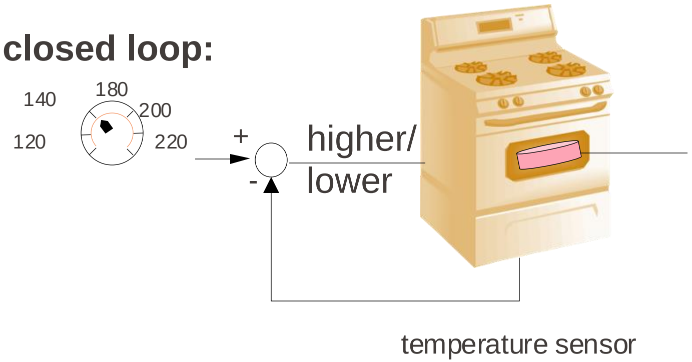
Control: Open loop and closed loop

open loop:



temperature is imprecise, depends on ambient temperature and cooking quantity, but time of heating can be modulated.

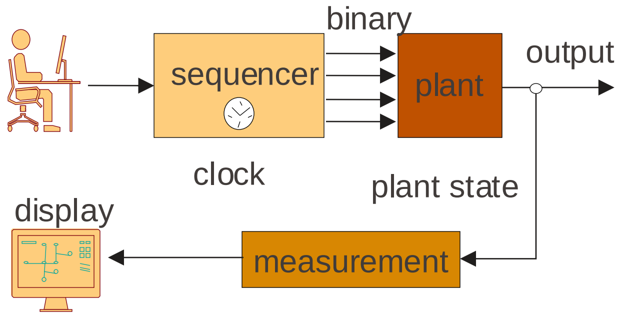
closed loop:



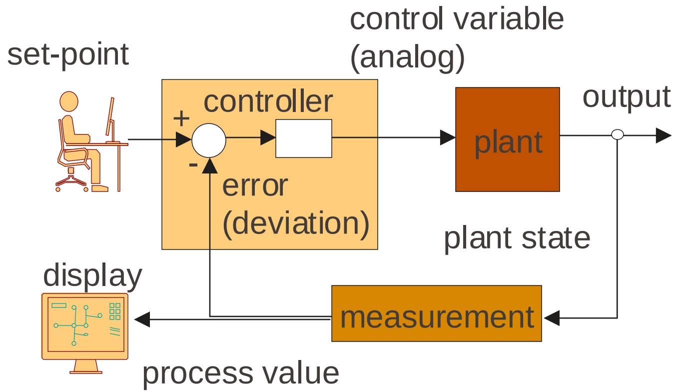
temperature closely controlled, requires measurement of the output variable (temperature)

Control: Open loop and closed loop

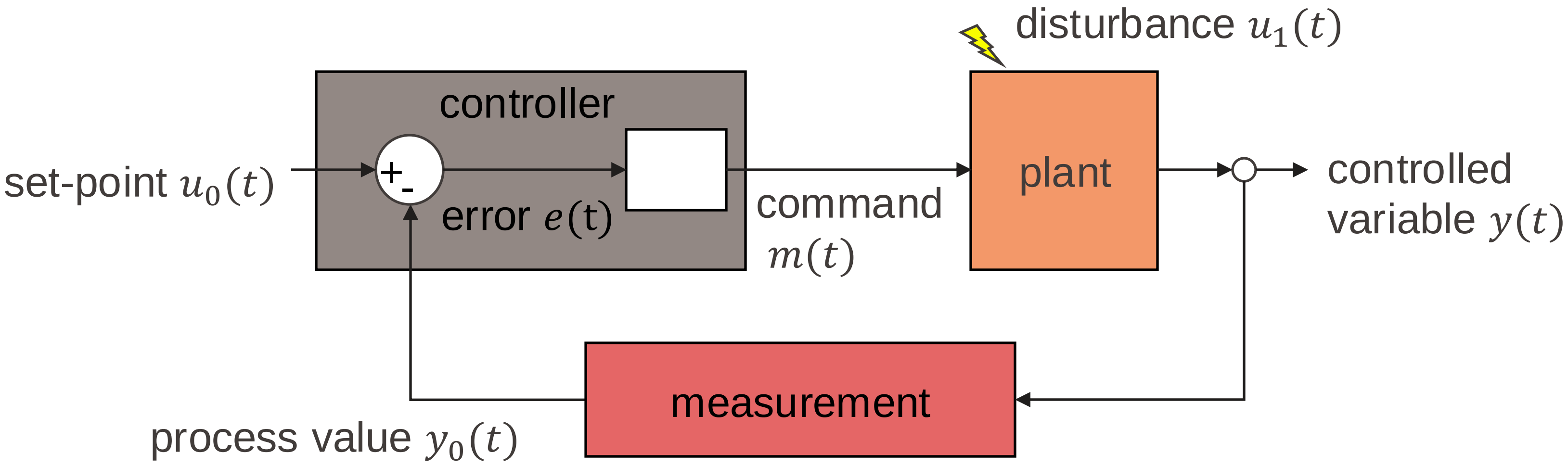
- open-loop control / command
  - sequential / combinatorial, binary variables, discrete processes, "batch control", "manufacturing"



- closed-loop control / regulation
  - feedback, analog variables, continuous processes, "process control"



Controller loop



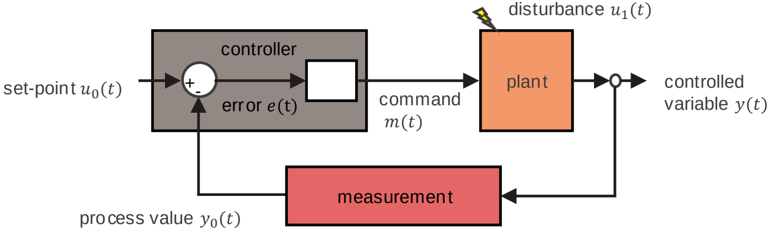
Implemented by mechanical or electrical elements, computers, ...
Controlled variable can not always be measured directly.

EPFL

Example: Cruise Control

Source: OCAL, clker.com

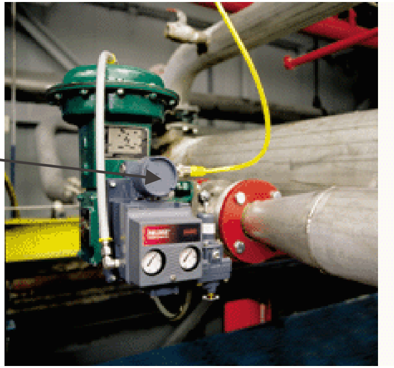
- Control Objective: maintain car velocity
- Measured Process Variable: car velocity
- Manipulated Variable:
  pedal angle, flow of gas to engine
- Controller Output:
  signal to actuator that adjusts gas flow
- Set point: desired car velocity
- Disturbances:
  hills, wind, curves, passing trucks….



Source: http://apmonitor.com/che436/uploads/Main/Lecture3_notes.pdf

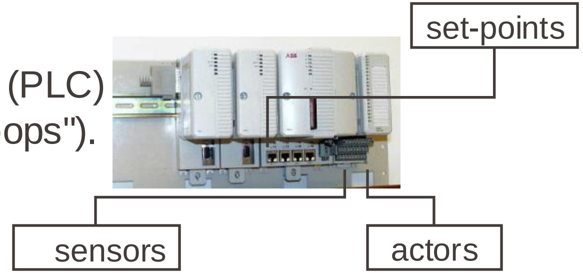
Where is the controller located?

directly in the sensor
or in the actuator



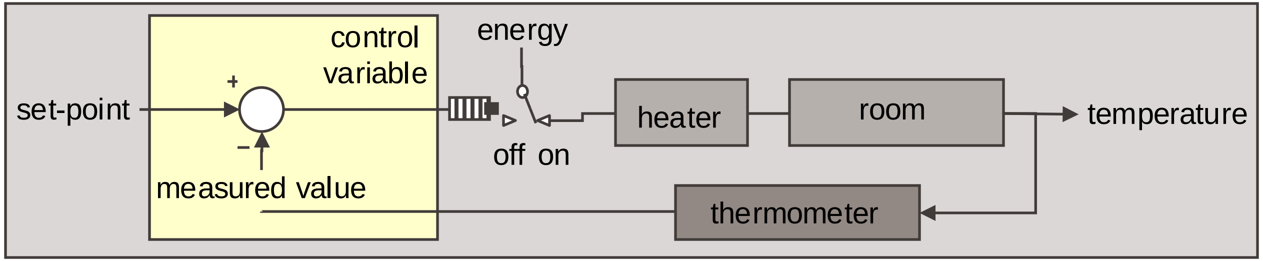
high-end: in a set of possibly redundant controllers (PLCs) (here: turbine control)

as an algorithm in a computer (PLC)
(that can handle numerous "loops").



Two-point controller: principle

The two-point controller (or bang-bang controller, regulator) has a binary output: on or off (example: heating)



Honeywell T-86 “Round” Thermostat (1953)

Nest 2nd Gen Learning Thermostat (2014)

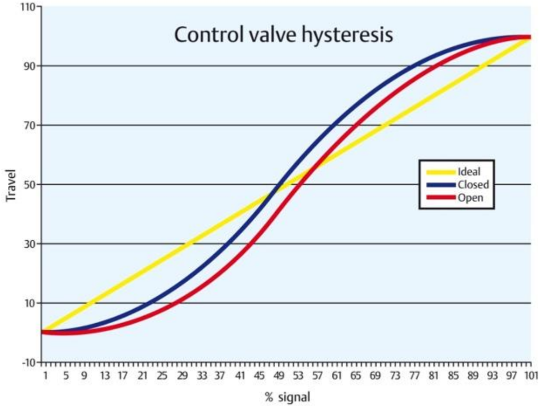
Hysteresis and Deadband of a Valve

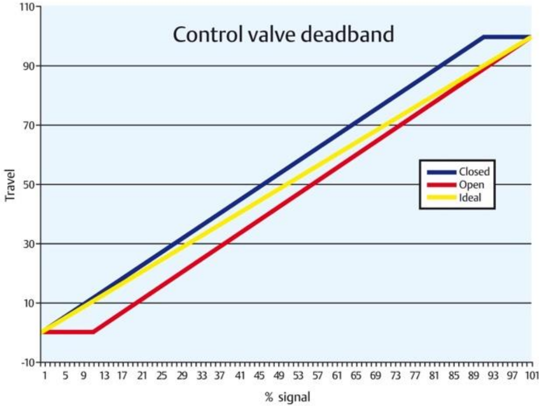
- **Hysteresis**

  difference between the valve position on the up-stroke and its position on the down-stroke at any given input signal (static friction)

- **Deadband**

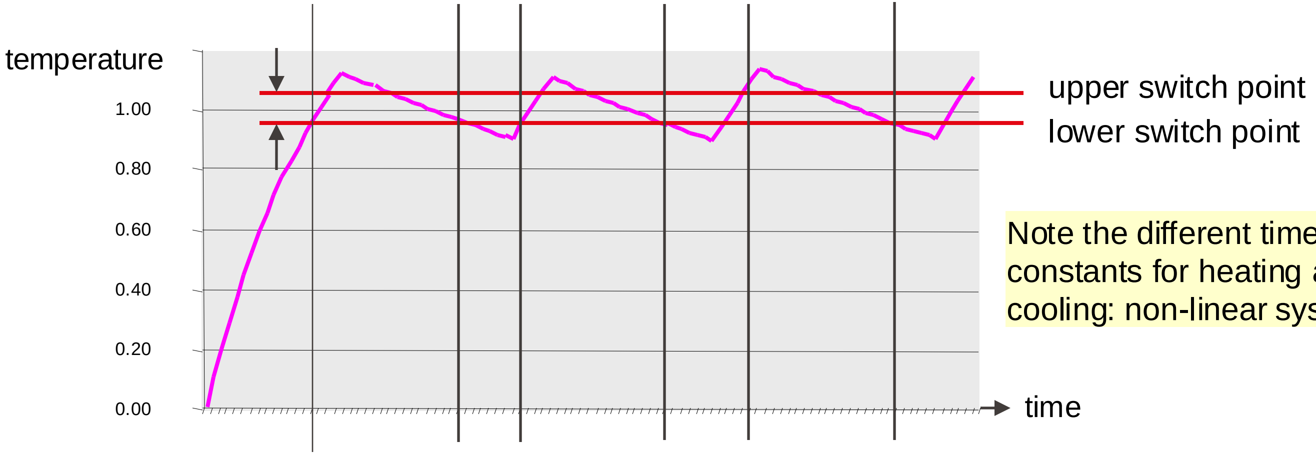
  no movement, generally occurs when the valve changes direction.





Source: https://www.processindustryforum.com/article/valve-terminology-basic-understanding-key-concepts

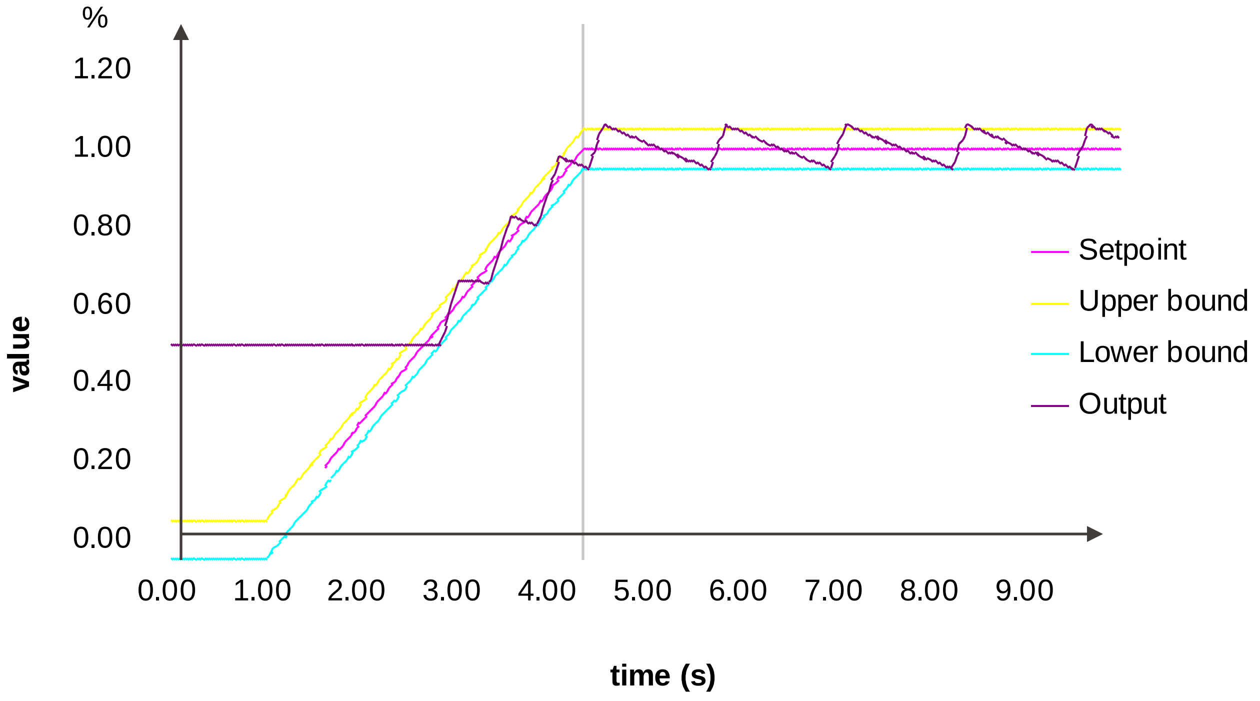
Two-point controller: Hysteresis / Deadband



Note the different time constants for heating and cooling: non-linear system

- If the process is not slow enough, hysteresis and deadband are included in switch point calculation to limit switching frequency and avoid wearing off the contactor (thermal processes are normally so inertial that this is usually not needed).

Two-point controller: Input variable as ramp

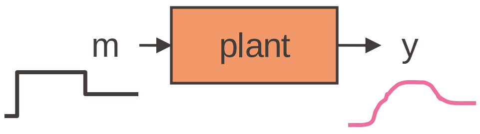


Plant model example

The following examples use a plant modeled by a 2nd order differential equation:

$$y + y'T_1 + y''T_2 = m$$

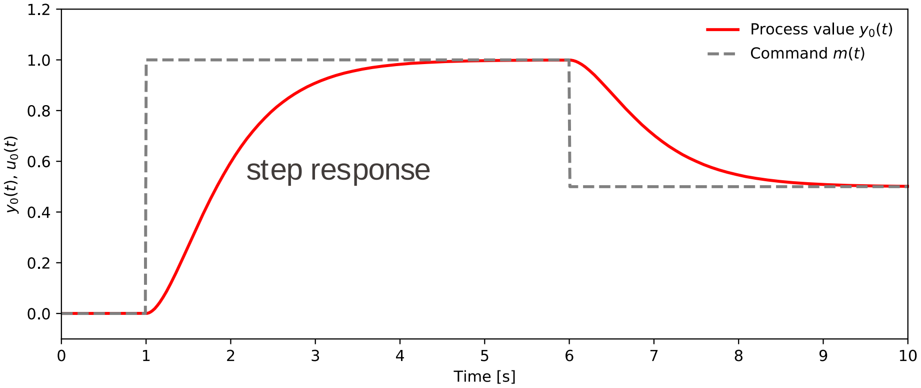
differential equation



$$\frac{y}{m} = \frac{1}{1 + sT_1 + s^2T_2}$$

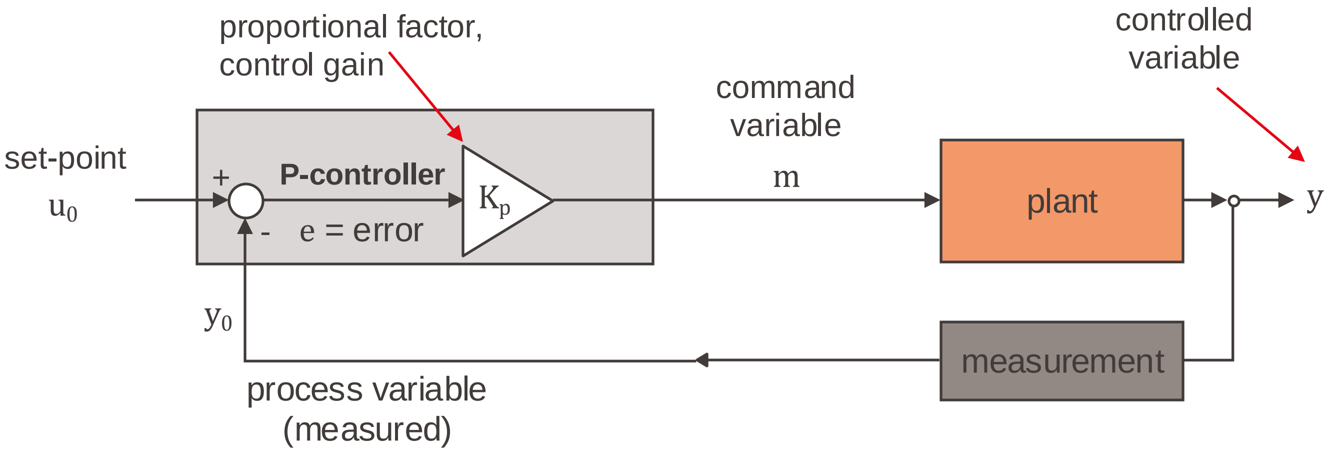
Laplace transfer function (since system is linear)

Typical transfer function of a plant with slow response.



In the examples:
$T_1 = 1.0$
$T_2 = 0.25$

P-Controller: simplest continuous regulator



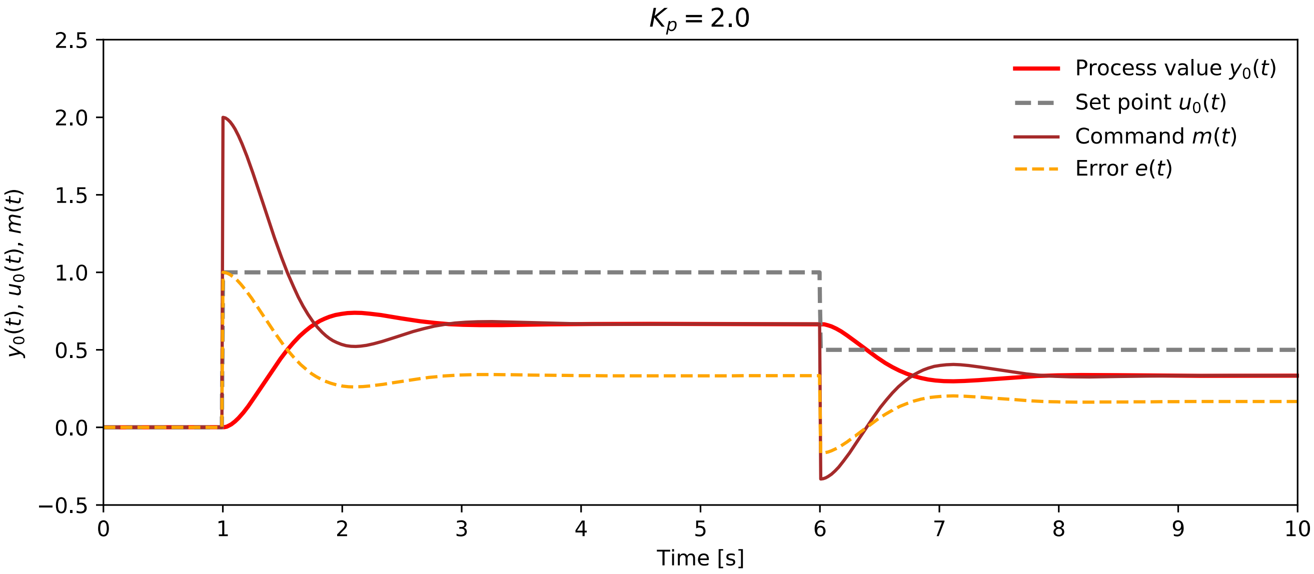
The P-controller simply amplifies the error to obtain the command variable:

$$m(t) = K_p\, e(t) = K_p\, (u_0(t) - y_0(t))$$

This works, but if plant has a proportional behavior, an error always remains.

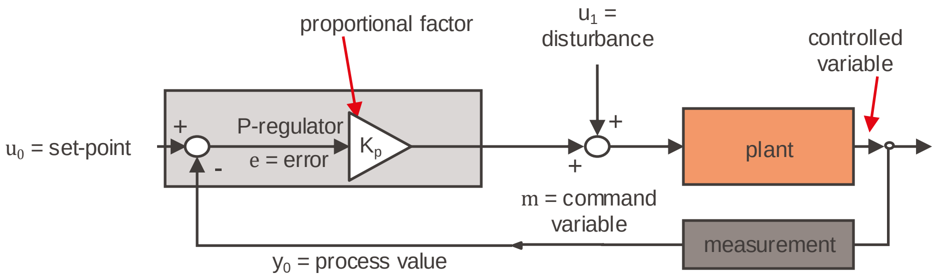
P-Controller: Step response

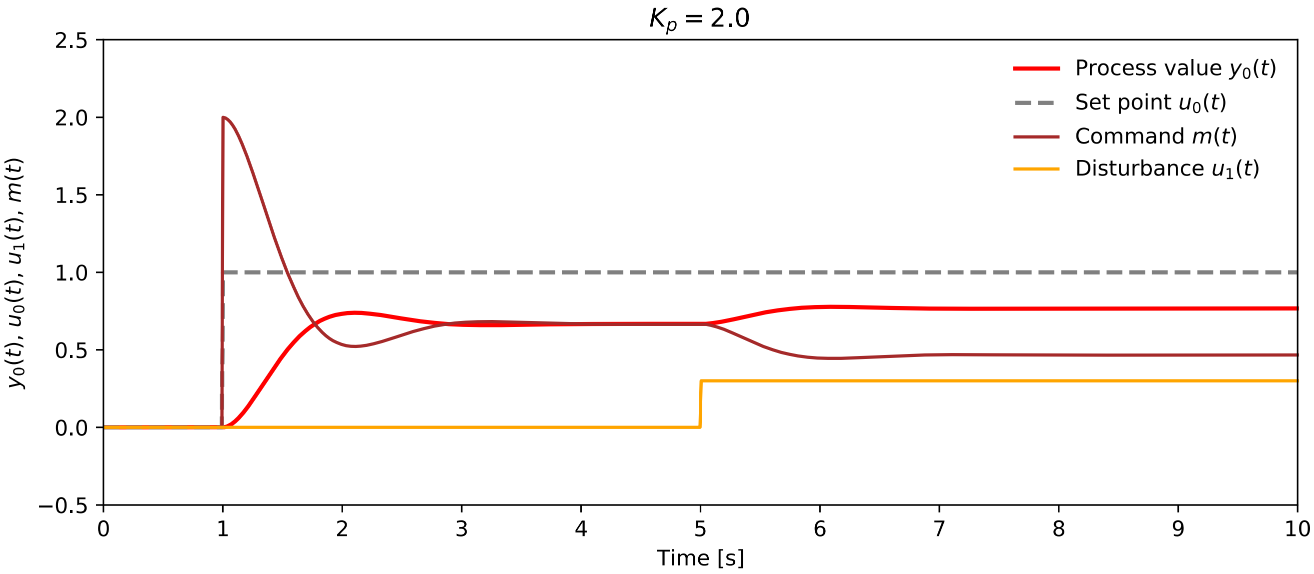
$$m(t) = K_p\, e(t) = K_p\, (u_0(t) - y_0(t))$$



The larger the set-point, the greater the error.

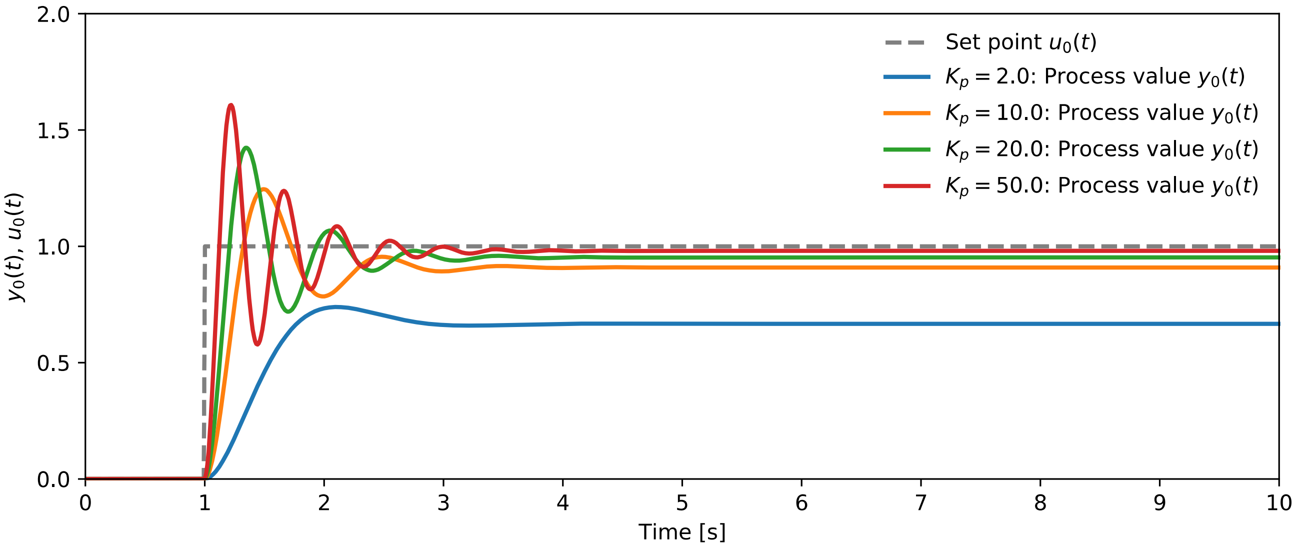
P-Controller: Effect of Load change





Not only a set-point change, but a load change causes the error to increase or decrease.
(A load change, modeled by disturbance u_1, is equivalent to a set-point change)

P-Controller: Increasing the proportional factor

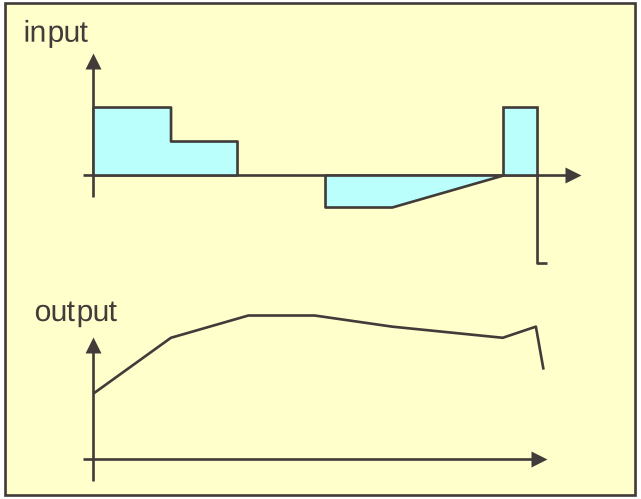


Increasing the proportional factor reduces the error, but the system tends to oscillate.

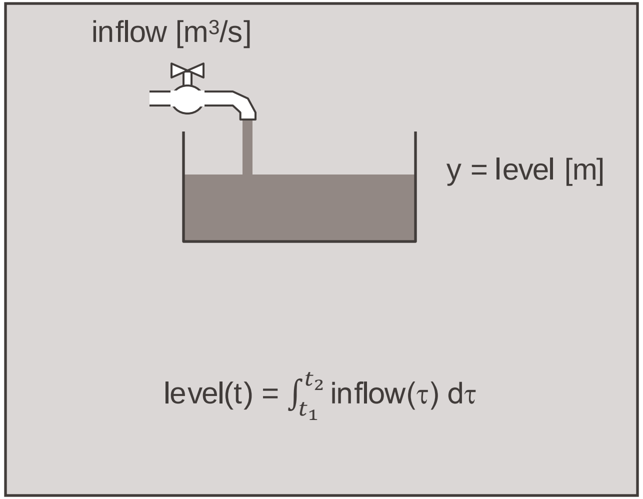
PI-Controller (Proportional Integrator)

Time domain $$m(t) = K_p\left(e(t) + \frac{1}{T_i}\int_{t_0}^{t} e(t)\,d\tau\right)$$

Laplace domain $$\tilde{m}(s) = K_p\left(1 + \frac{1}{sT_i}\right)\tilde{e}(s)$$

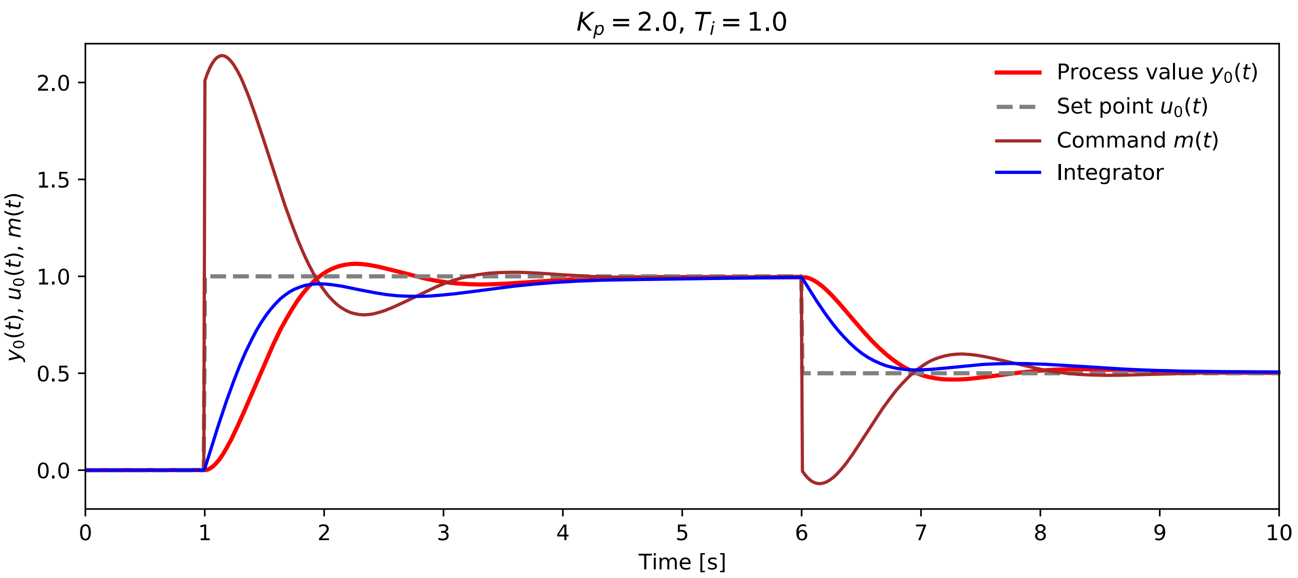


Time response of an integrator



Example of an integration process

PI-Controller: response to set-point change

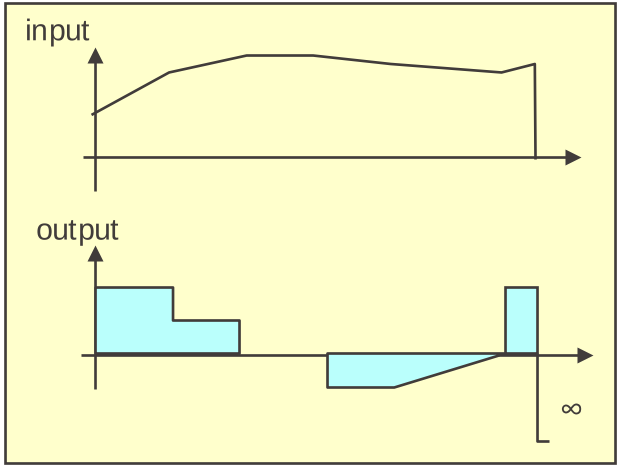


The integral factor reduced the asymptotical error to zero, but is slowing down the response. If K_p is increased to make it faster, the system becomes unstable.

PD-Controller: Proportional Differentiator

Time domain $$m(t) = K_p\left(e(t) + T_d\frac{de(t)}{dt}\right)$$

Laplace domain $$\tilde{m}(s) = K_p(1 + T_d s)\tilde{e}(s)$$

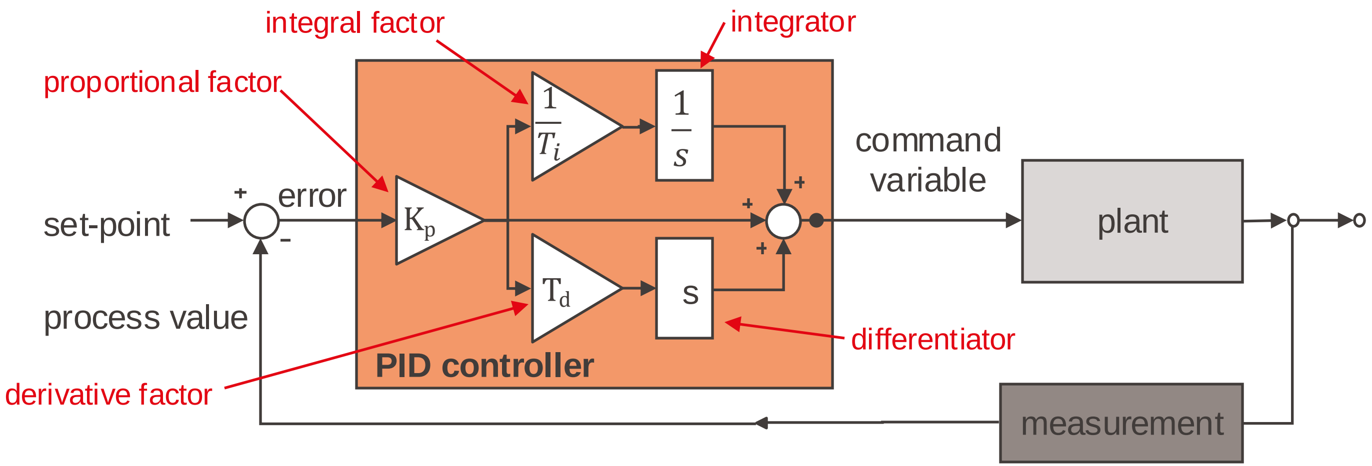


temporal response

A perfect differentiator does not exist.
Differentiators increase noise.
Differentiators are approximated by feed-back integrators (filtered differentiator):

Instead of differentiating, one can use an already available variable:
e.g. the speed for position control

PID-Controller



- K_p generates output proportional to error, requires non-zero error
- Increasing K_p decreases the error, but may lead to instability
- Increasing T_i can make system slower
- T_d speeds up response by reacting to error change proportionally to slope of change.

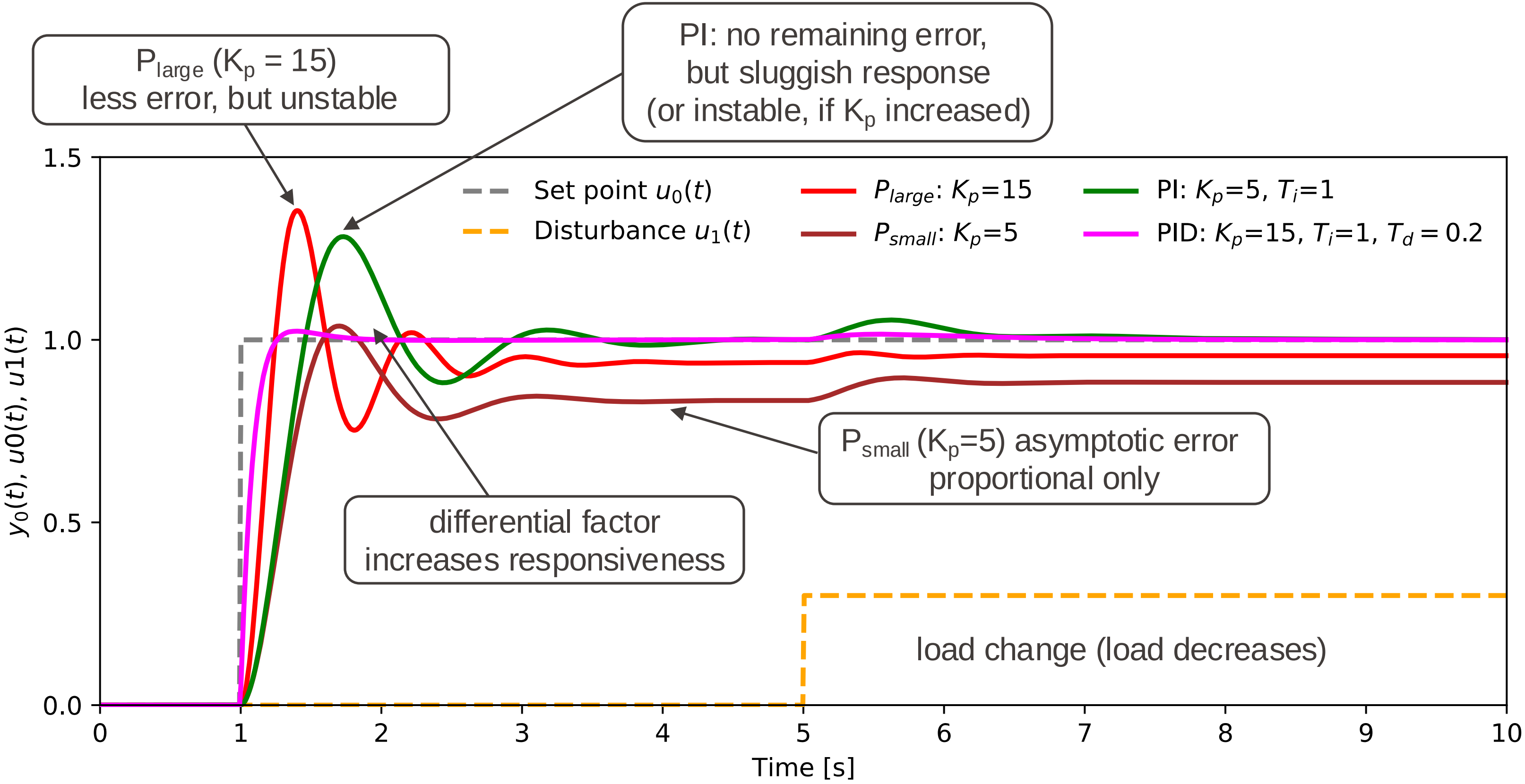
Proportional-Integral-Derivative (PID) Controller

- Generic and widely used control loop feedback mechanism
- Mode of operation:
  1. calculate error *e(t)*, the difference between measured process variable and the desired setpoint.
  2. try to minimize error by adjusting the process control output *m*.

$$m(t) = K_p \left(e(t) + \frac{1}{T_i} \int_{t_0}^{t} e(\tau)\, d\tau + T_d \frac{de(t)}{dt} \right)$$

K_p, T_i, T_d tuning parameters

PID response summary



Play with Matlab: http://ctms.engin.umich.edu/CTMS/index.php?example=Introduction&section=ControlPID

EPFL

PID-Controller: Manual Tuning

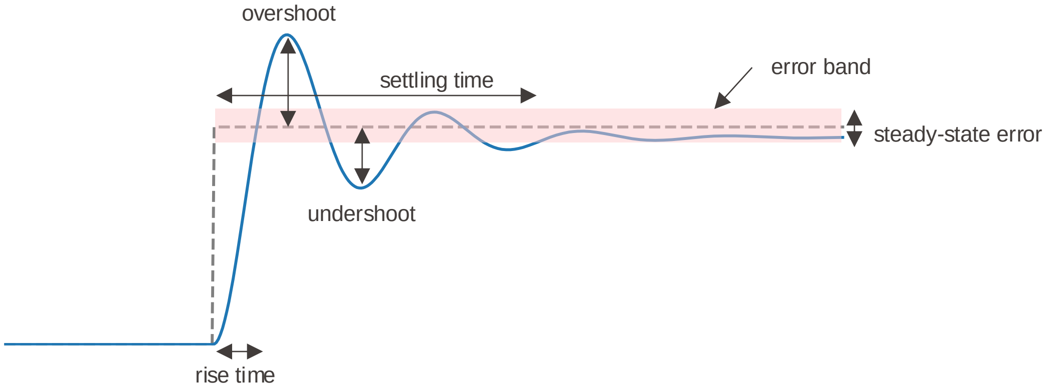


Table: Effect of *increasing* a parameter independently

Parameter	Rise time	Overshoot	Settling time	Steady-state error	Stability
K_p	Decrease	Increase	Small change	Decrease	Degrade
K_p/T_i	Decrease	Increase	Increase	Eliminate	Degrade
K_pT_d	Minor change	Decrease	Decrease	No effect in theory	Improve if K_pT_d small

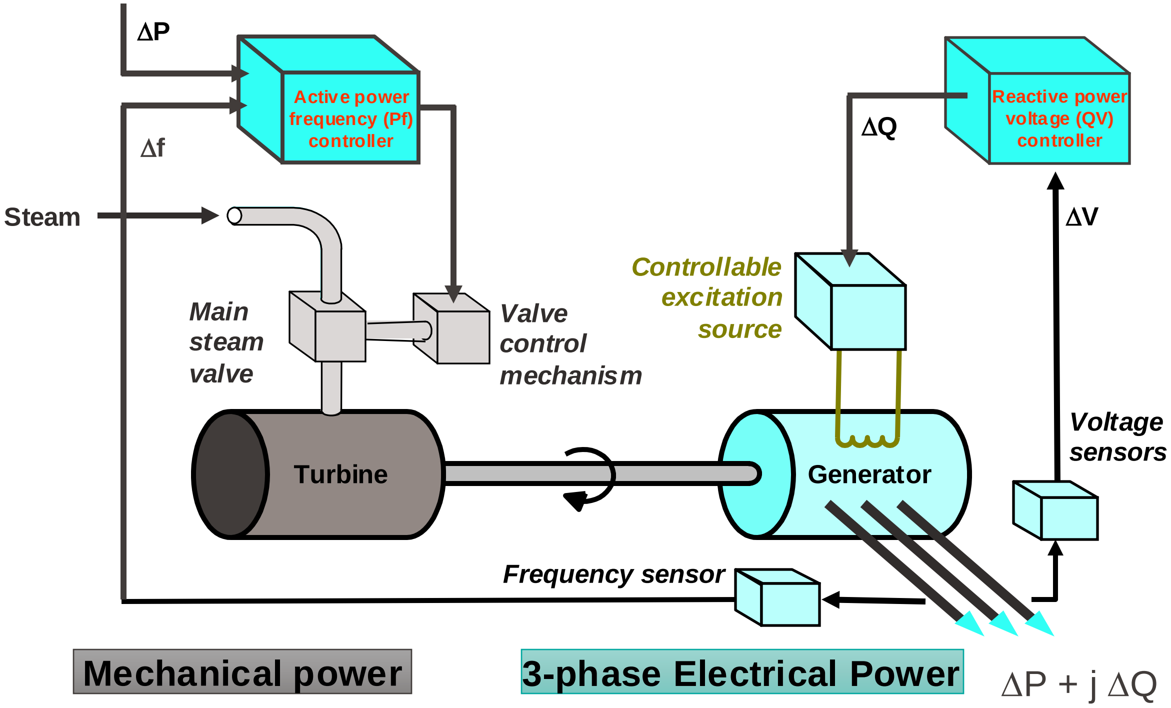
→ Ziegler-Nichols tuning: heuristic method to tune a PID controller.

Note: Modern PLCs might provide auto-tuning mechanisms

Source: https://en.wikipedia.org/wiki/PID_controller

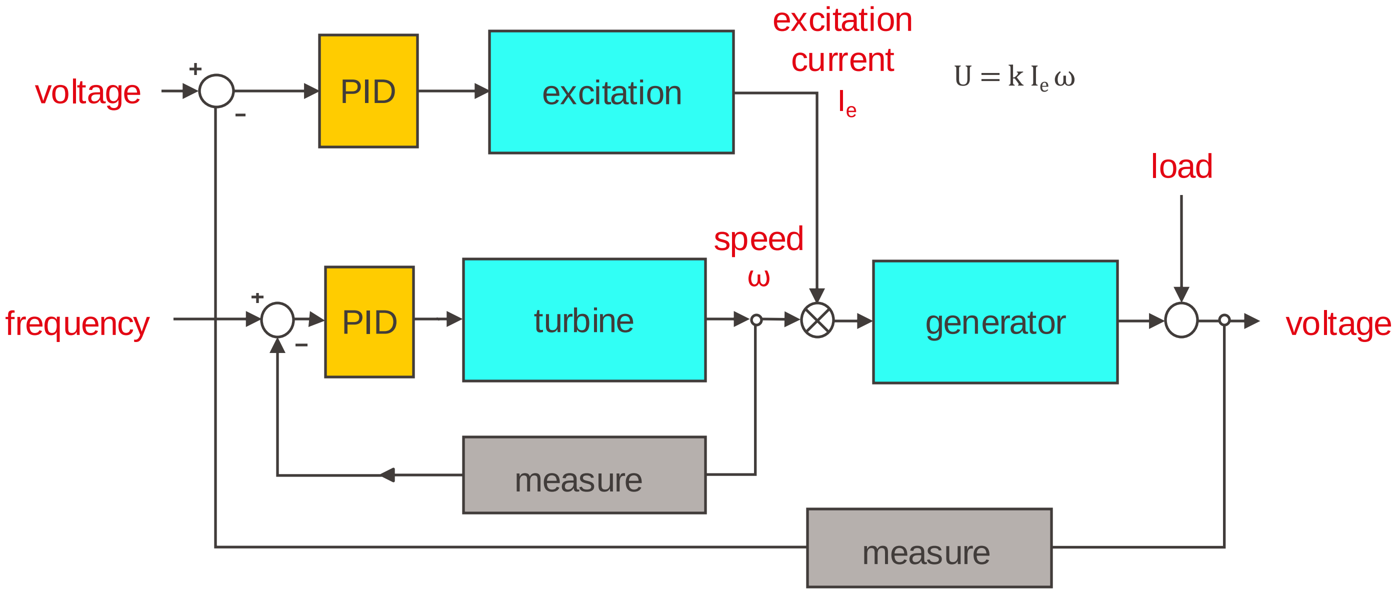
Several controllers act together: Electricity Generator

Not for exam, illustration only



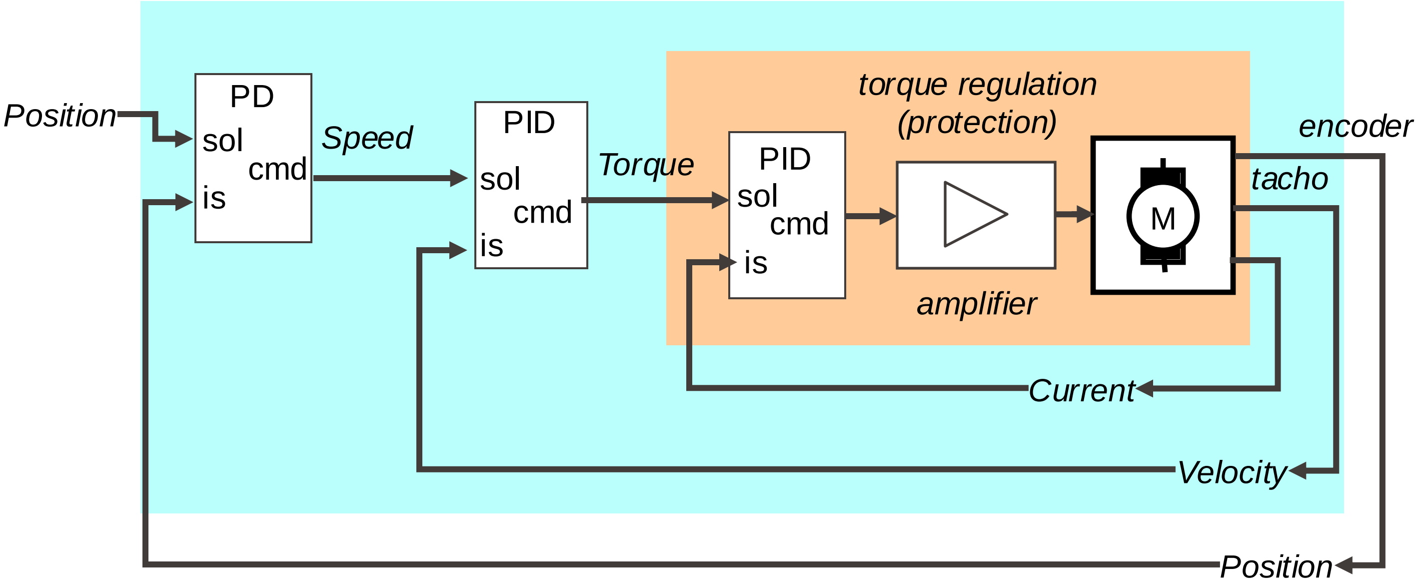
Generator Regulator structure

Not for exam, illustration only



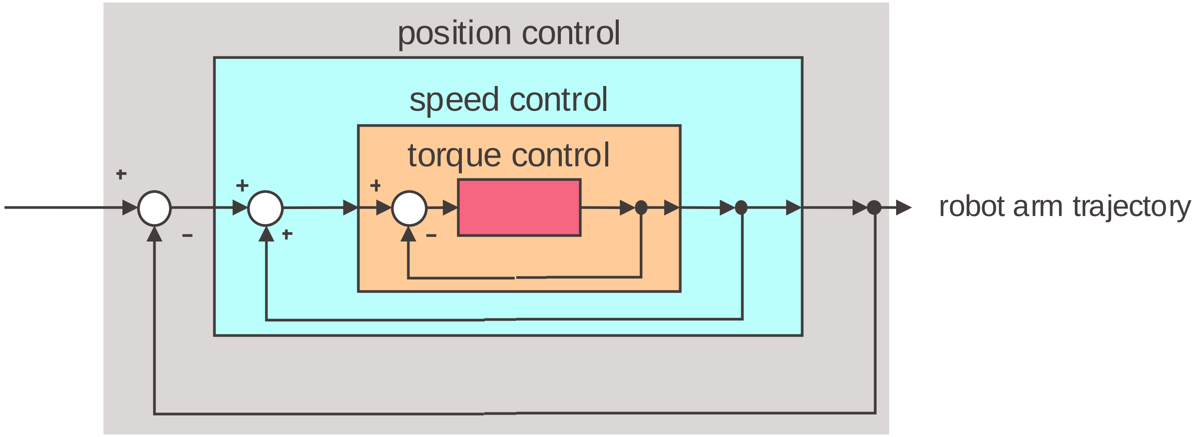
Nested control of a continuous plant - example

Example: position control of a rotating shaft



Nesting regulators allow to maintain the output variable at a determined value while not exceeding the current or speed limitations.

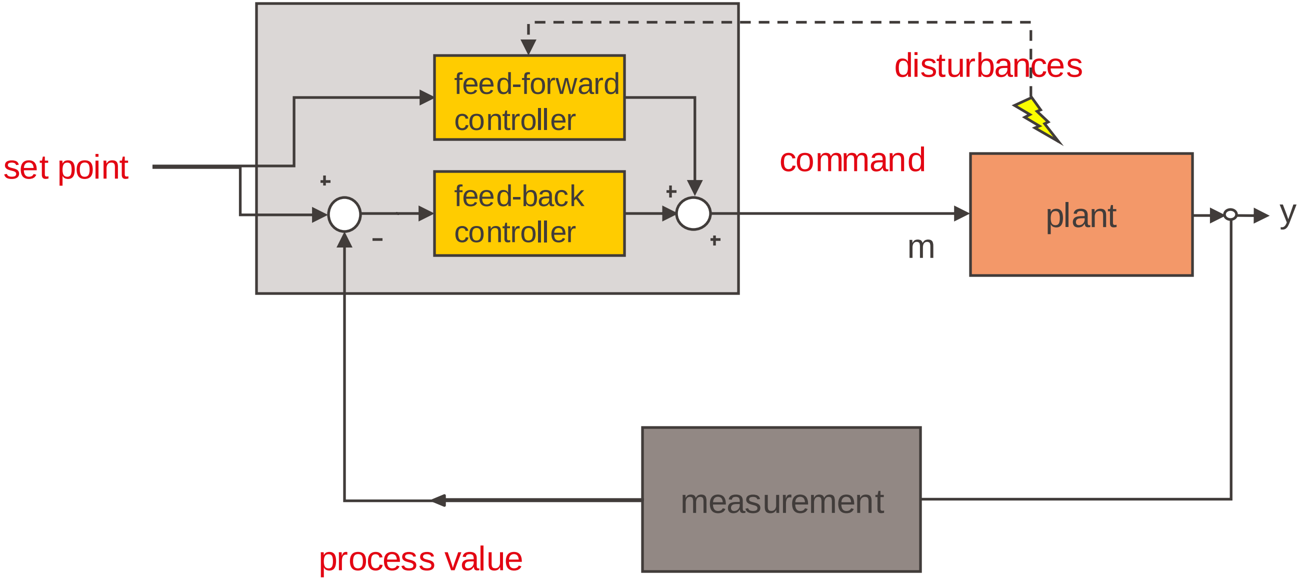
Nested loops and time response



A control system consists often of nested loops, with the fastest loop at the inner-most level

Feed Forward Control

Basic idea: bring output on good track first, let regulator correct small deviations.
Feed forward controller knows the plant, it can also consider known disturbances.



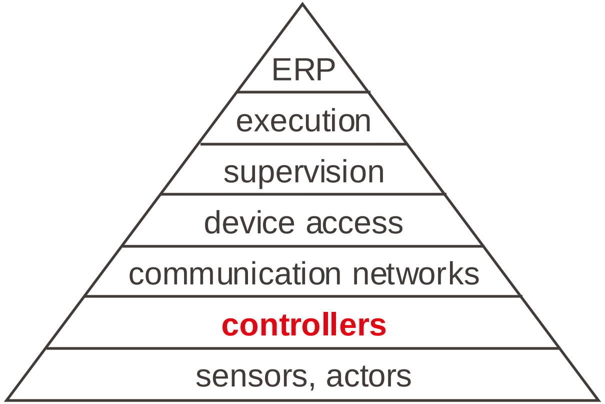
Assessment

- How does a two-point regulator works?
- How is the wear-out of the contacts prevented?
- How does a PID regulator work?
- What is the influence of the different parameters of a PID?
- Is a PID controller required for a position control system (motor moves a vehicle)?
- Explain the relation between nesting control loops and their real-time response
- What is feed-forward control?

Programmable Logic Controllers (PLCs)

PLC = Programmable Logic Controller

- Definition: small computers, dedicated to automation tasks in an industrial environment
- Real-time (embedded) computer with extensive input/output
- Functions:
  - Measure, Control, Protect + Event Logging
  - Event Logging
  - Communication
  - Human machine interface (HMI)



Programmable Logic Controller

- Components in a PLC can be categorized into:
  - Processor (CPU, etc.)
  - Input (digital, analog, etc.)
  - Output (digital, analog, etc.)
  - Communication interface

CPU

Communication interface

analog inputs / outputs

digital inputs /outputs

https://new.abb.com/plc/programmable-logic-controllers-plcs/ac500

PLC: Characteristics

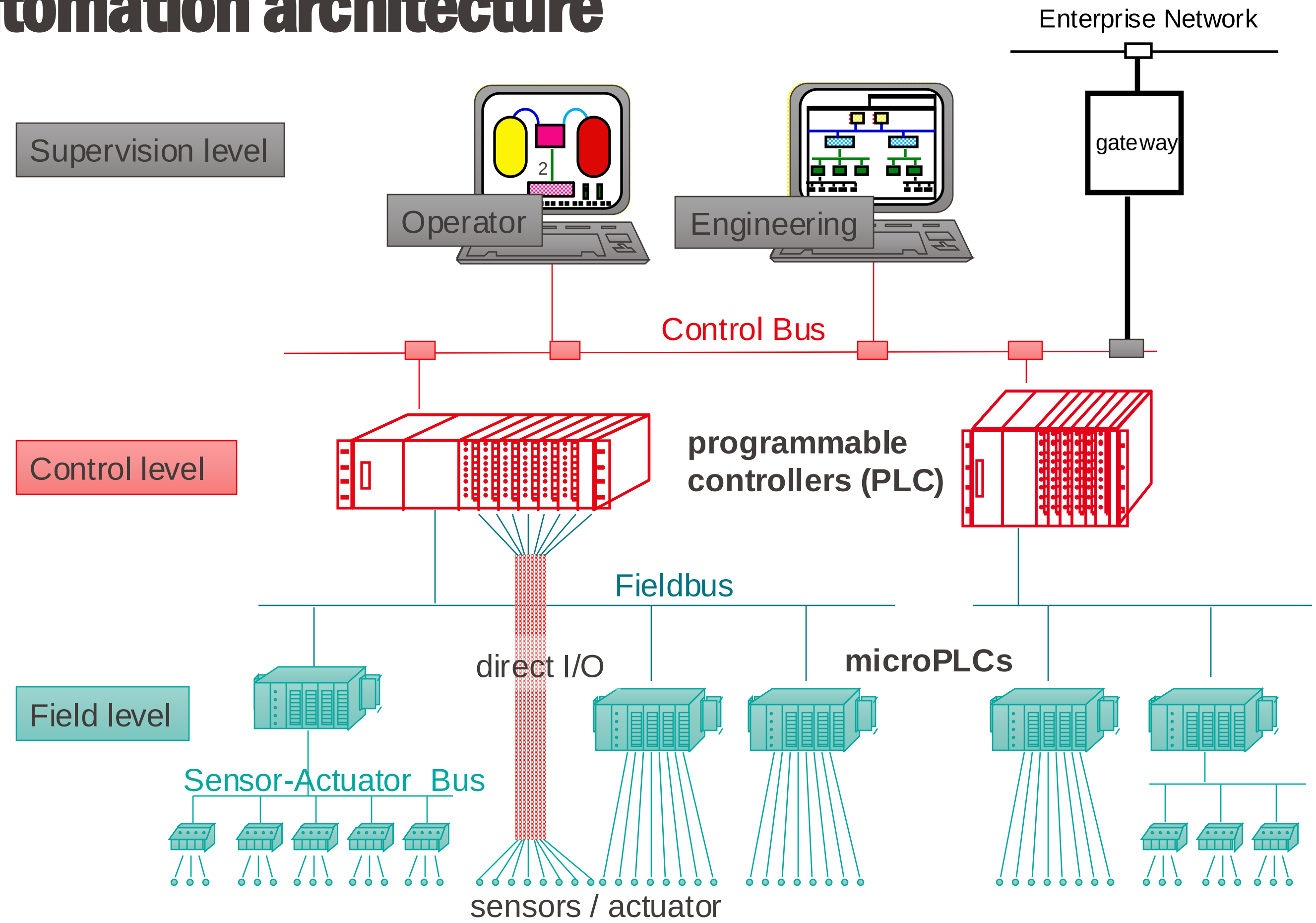
- large number of peripherals: 20..100 I/O per CPU, high density of wiring
- digital and analog input/output with standard levels
- field bus or Ethernet connection for remote I/O.
- operate under harsh conditions, require robust construction, protection against dirt, water, mechanical threats, electro-magnetic noise, vibration, extreme temperature range (-30ºC to 85ºC), sometimes directly located in the field.
- simple Human-Machine-Interface (HMI) for maintenance, either through LCD-display or external Ethernet connection (web interface).
- Economical (few 100 EUR)
- value is in the application software (license costs)

Why 24V / 48V supply ?

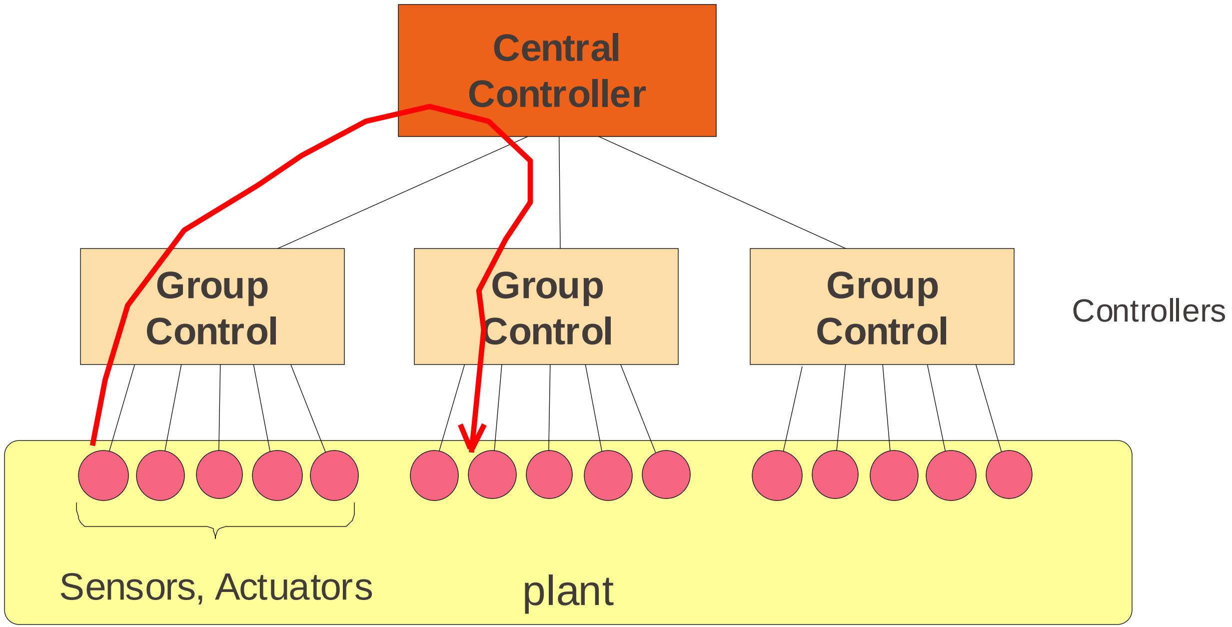
Photo: TEPCO

“ (…) After the plant lost electric power, operators could read instruments only by plugging in temporary batteries (…) [IEEE Spectrum Nov 2011 about Fukushima]

PLC: Location in the automation architecture

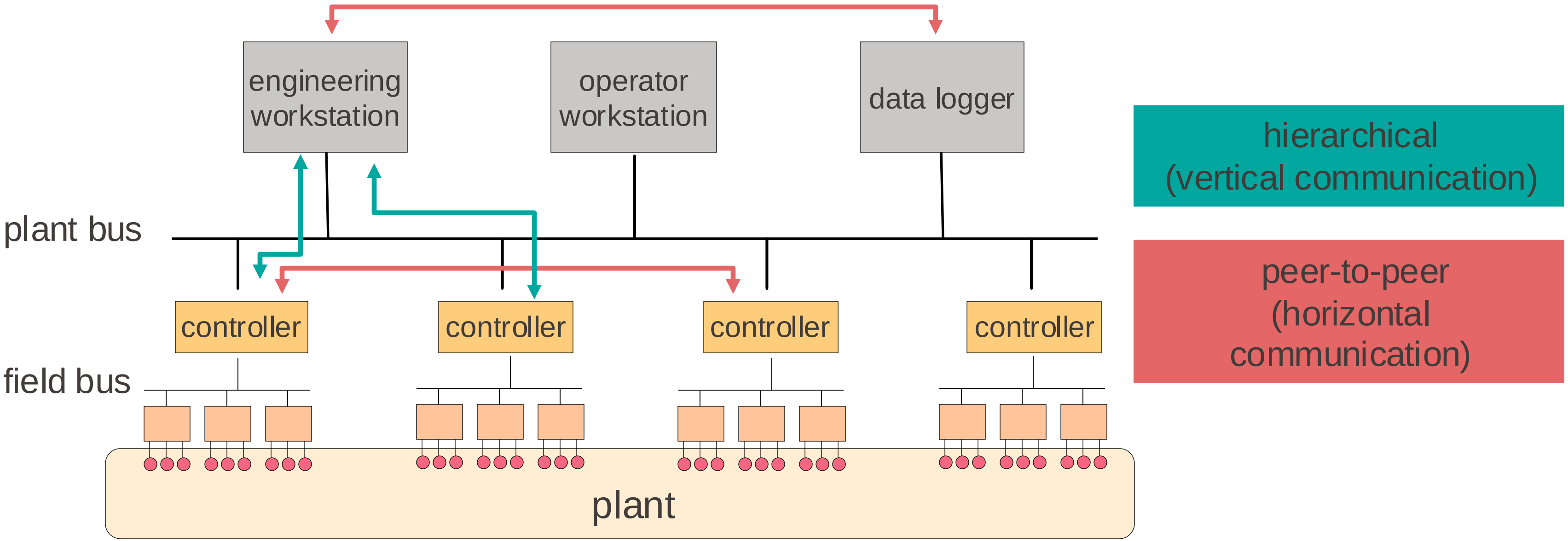


Centralized (Hierarchical) Architecture



- Classical, hierarchical, centralized architecture.
- The central controller only monitors and forwards commands to the group controller.

Decentralized Automation System



- All controllers can communicate as peers (without going through a central master), restricted only by throughput and modularity considerations.

PLC Programming and Deployment

More to follow in Week 7

- Programming:
  - Early days: very primitive with hand-held terminals on the target machine itself
  - Today: software developed on a separate computer, then compiled and uploaded on the PLC

- PLC software updates:
  - Software can be downloaded if PLC supports that feature
  - Software changes deployed using engineering workstations/laptops over control bus

Kinds of PLCs

- Compact PLCs
  - Monolithic construction
- Modular PLCs
  - Modular construction (backplane)
  - Extensible with different modules
- Soft-PLCs
  - Linux or Windows-based automation products
  - Direct use of CPU or co-processors

Compact PLC

https://new.abb.com/plc/legacy-products/ac31-and-previous-series

- Monolithic (one-piece) construction
- Fixed casing
- No additional processing capabilities
- Fixed number of I/O
- If more is needed can be extended and networked by an extension (field) bus
- Sometimes LAN connection (Ethernet)
- Typical product: Mitsubishi MELSEC F, ABB AC31, SIMATIC S7

Compact PLC: Specific Controller (example: Turbine)

tailored for a specific application, produced in large series

Programming port

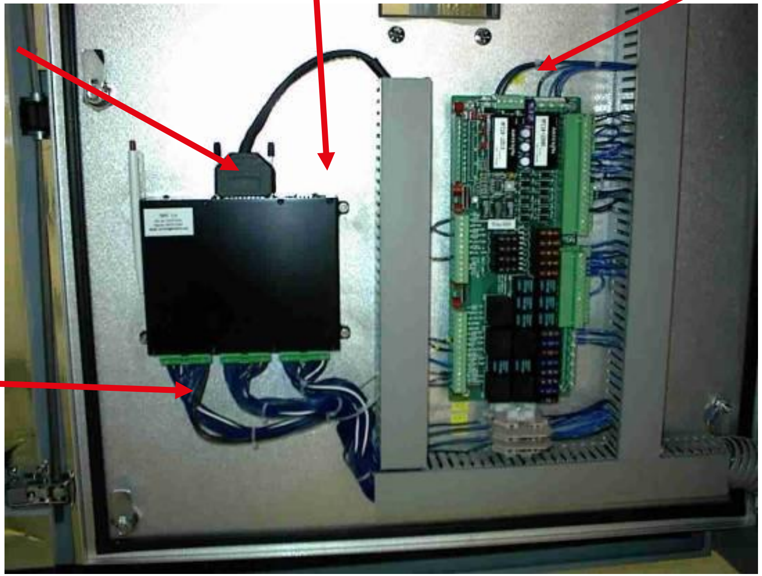
Relays and fuses

Thermocouple inputs

binary I/Os,

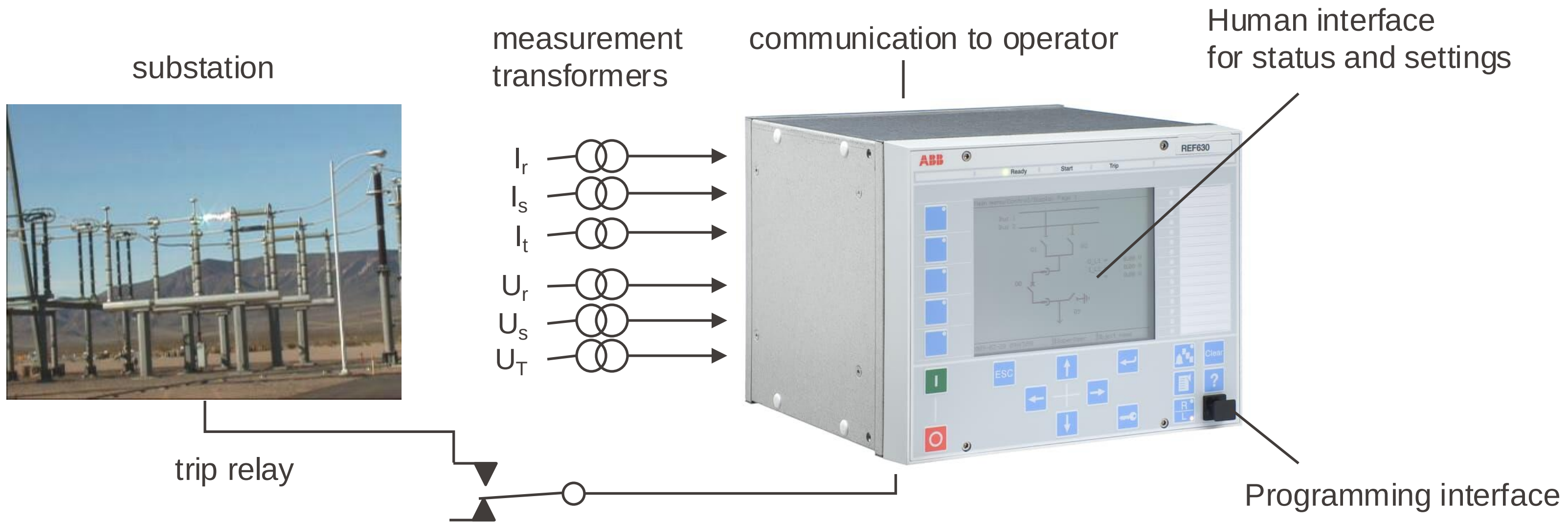
CAN field bus

RS232 to HMI



courtesy Turbec

Compact PLC: Protection devices

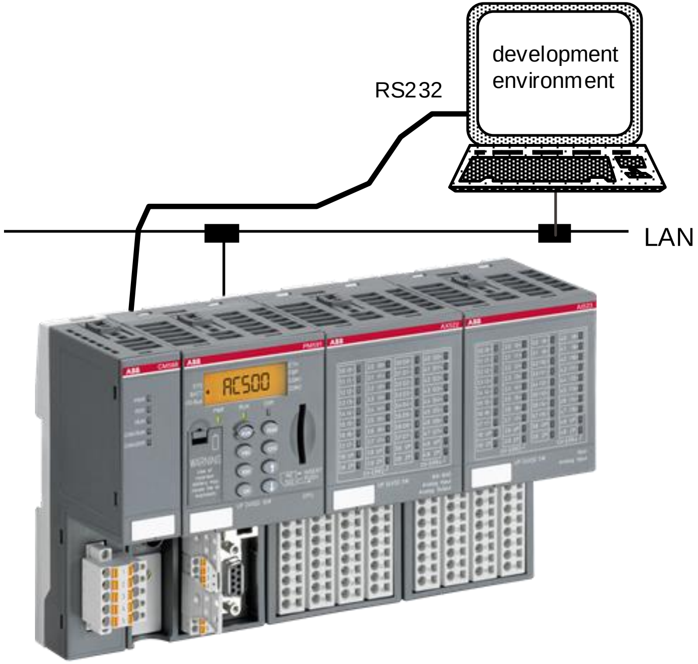


Highly specialized PLCs, measure current and voltages in electrical substations to detect dangerous situations (over-current, short circuit, overheat) and trigger the circuit breaker (“trip”) to protect the substation. In addition, they record disturbances and send reports to substation’s SCADA.

Sampling rate: 4.8 kHz, reaction time: < 5 ms.

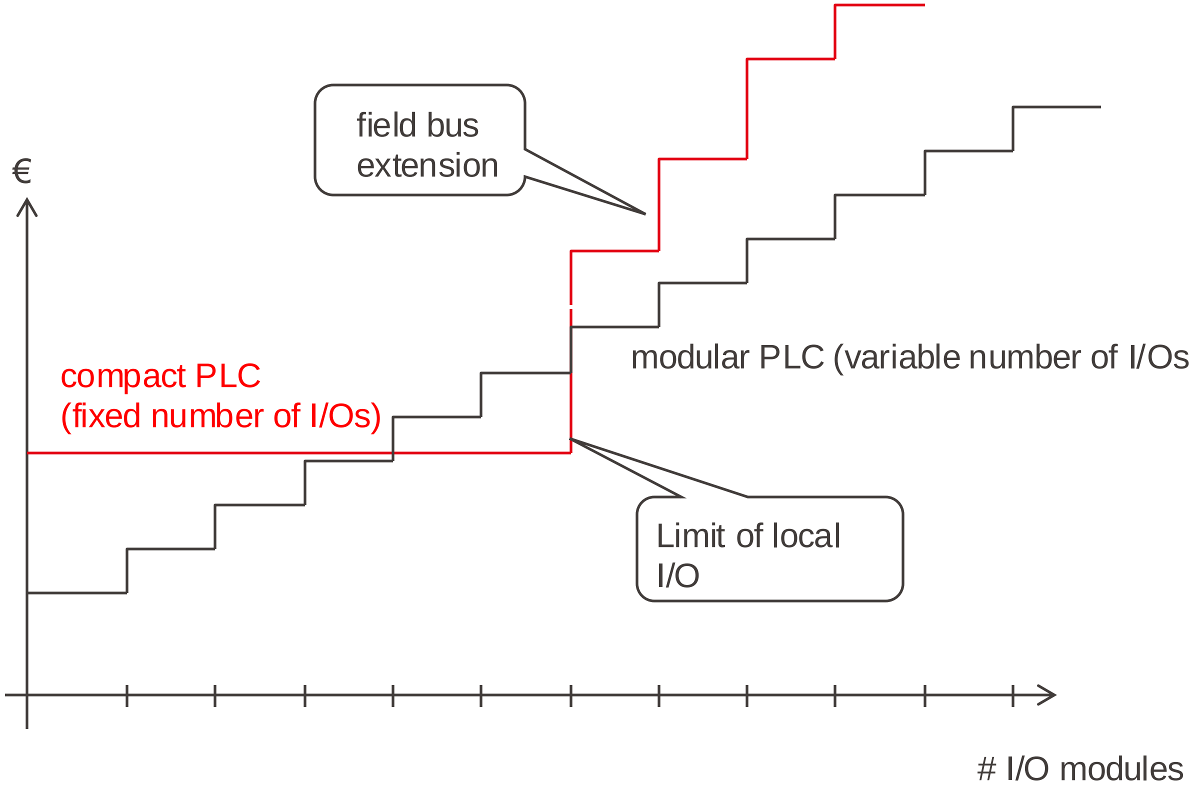
Modular PLC

- can be tailored to needs of application:
  - high processing power (several CPUs)
  - large choice of I/O boards
  - concentration of a large number of I/O
  - interface boards to field busses
- supply 115-230V, 24V or 48V (redundant)
- primitive or no HMI
- cost effective



Typical products: SIMATIC S5-115, Hitachi H-Serie, ABB AC500

Compact or modular ?

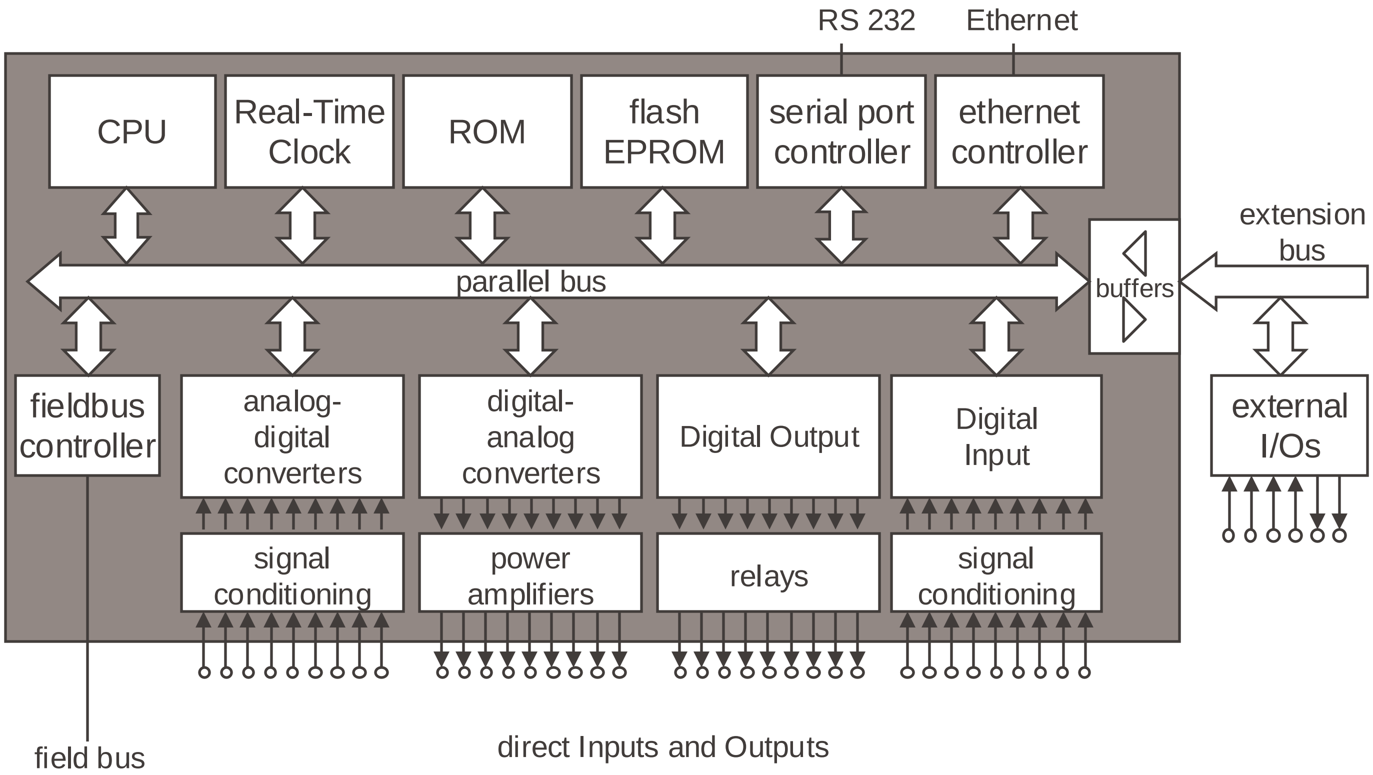


Soft-PLC: Industrial PC as PLC

- PC inside a ruggedized enclosure
- HMI (TFT, Mouse, Keyboard)
- Limited modularity through Mini PCIe cards
- Competes with modular PLC
- no local I/O, Ethernet connection instead
- costs: € 500-2000

https://www.syslogic.de/deu/industrial-pc-compact-c7-core-i-7th-generation-85598.shtml

General PLC architecture



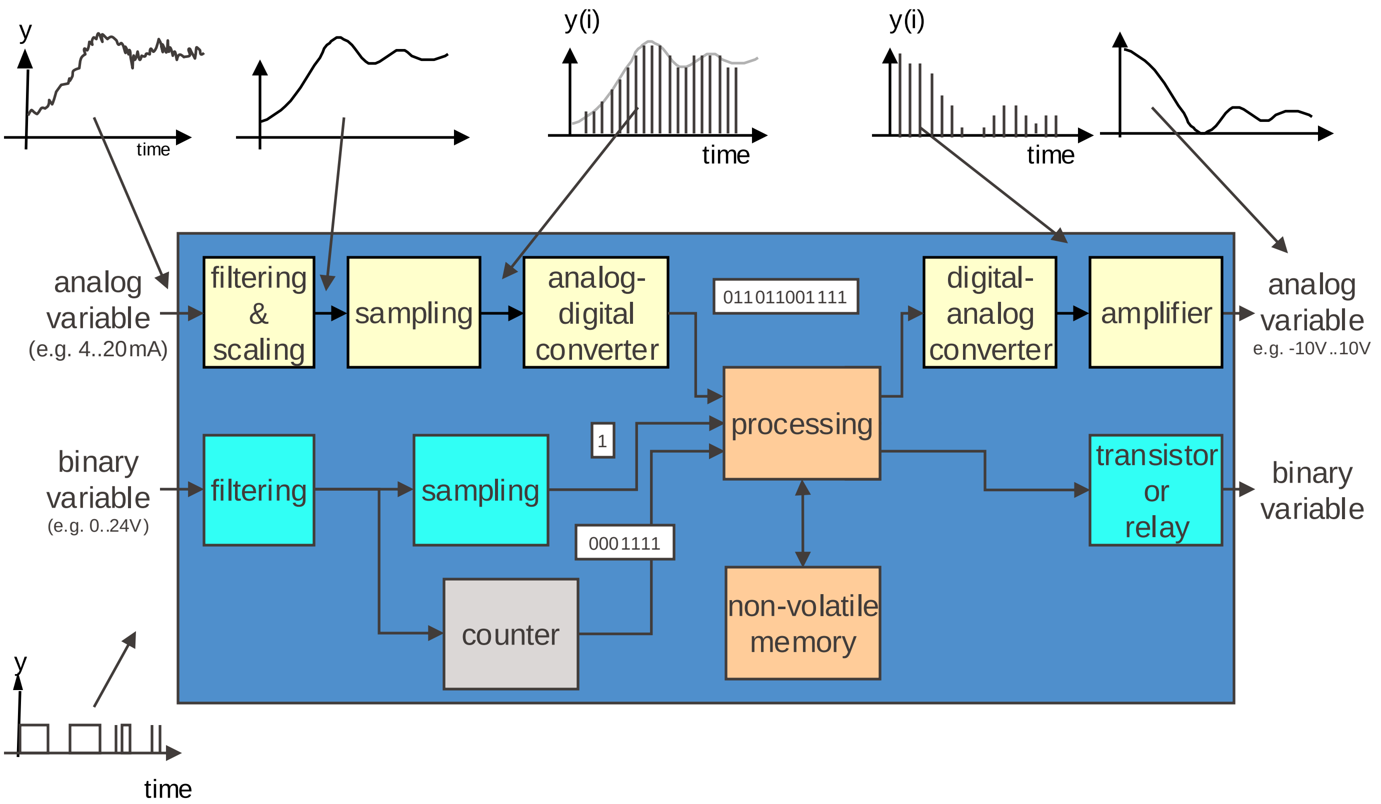
PLC Operation

- PLC operates periodically (in cycles)
- Samples signals from sensors and converts them to digital form with A/D converter
- Computes control signal and converts it to analog form for the actuators.

- A cycle consists of:
  1. Wait for clock interrupt
  2. Read input from sensor
  3. Compute control signal
  4. Send output to the actuator
  5. Update controller variables
  6. Communication
  7. Repeat

Waiwera Organic Winery, Distillation Plant

The signal chain within a PLC



Assessment

- What characterizes a PLC, which kinds exist and what is their application field?
- List selection criteria for PLCs
- Describe the chain of signals from the sensor to the actors in a PLC
- What is the difference between a centralized and a decentralized control system? What are the (dis-)advantages of the two approaches?