Logic Control

Prof. Dr. Ping Zhang

WS 2017/2018

TECHNISCHE UNIVERSITÄT KAISERSLAUTERN

Organizational issues

- **Lecture:** Prof. Dr. Ping Zhang
  Email: pzhang@eit.uni-kl.de
  Office: 12/474
  Appointment through Ms. Monika Kunz
  Email: mkunz@eit.uni-kl.de
  Office: 12/476
- **Exercise:** Dipl.-Ing. Anna Nehring
  Email: nehring@eit.uni-kl.de
- **Language of the course:** English
- **Scope:** 3 SWS (2V 1Ü)
- **Script:** available on the OLAT (password: Automatisierungstechnik)
- **Examination:** written examination, 90 min.

The lab “**Control of a production line with PLC**” is supervised by Ms. Anna Nehring.

Literature

- L. Litz. **Grundlagen der Automatisierungstechnik**, Chapter 3, Oldenbourg Verlag, 2013.
- J. Lunze. **Automatisierungstechnik**, Teil 3, Oldenbourg Verlag, 2013.
- C. G. Cassandras and S. Lafortune. **Introduction to Discrete Event Systems**, Springer, 2009.
- Frey, G.: **Design and formal Analysis of Petri Net based Logic Control Algorithms**. Shaker Verlag, Aachen, 2002.
- John, K.-H. and M. Tiegelkamp. **IEC 61131-3: Programming Industrial Automation Systems**. Springer, 2001.

Overview of the course

- Introduction
- **Modeling** of logic control systems
  - Boolean algebra
  - Finite state automata
  - Petri nets, SIPN
- **Analysis** of logic control systems
- **Design** of logic control systems
- **Verification and validation**
- **Online diagnosis** of logic control systems
- **Implementation** of logic control systems
  - PLC
  - Programming languages (IEC 61131-3)
  - Automatic code generation
- Distributed control (optional)

Study goal

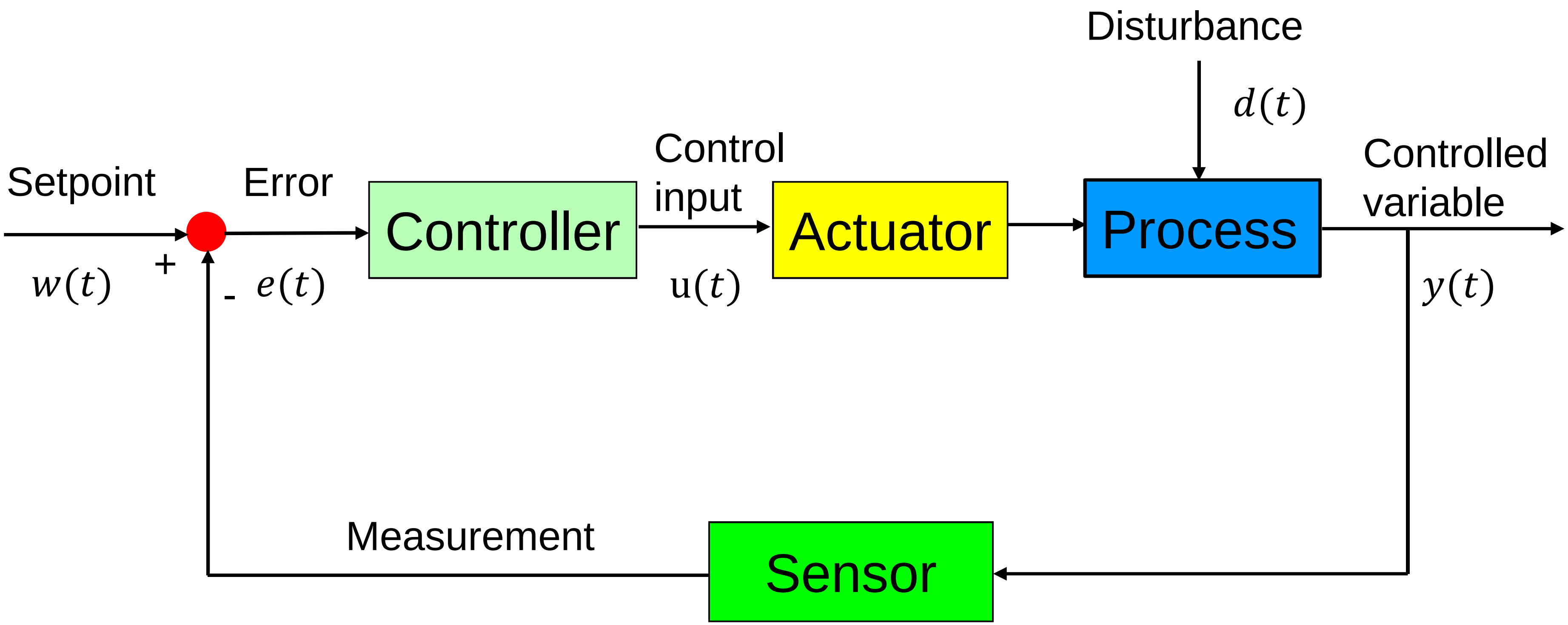
- Know typical logic control systems
- Be able to model logic control systems
- Be able to analyze and evaluate the behavior of logic control systems
- Be able to design logic control systems
- Be able to implement simple logic control systems on PLC
- Be able to verify/validate logic control algorithms
- Know how to organize industrial projects related to logic control systems

Chapter 1

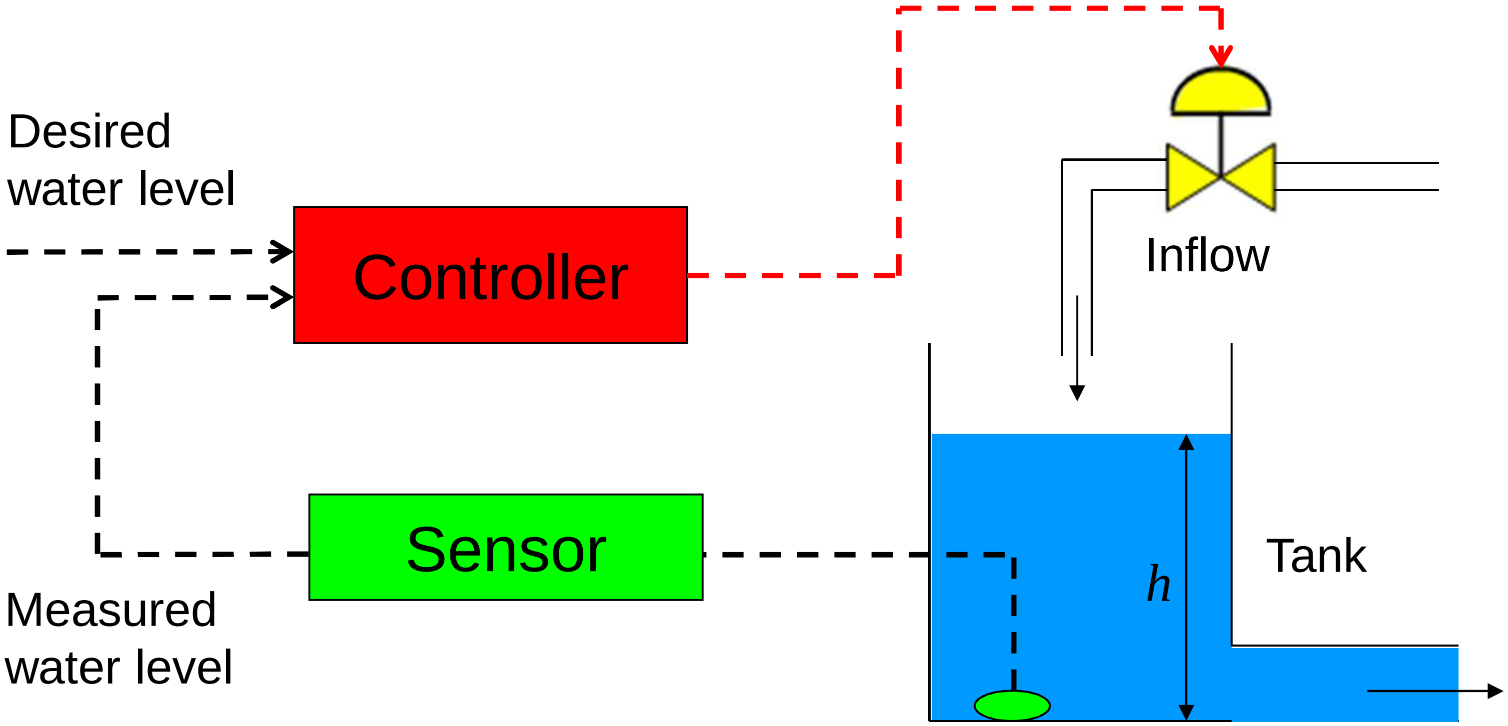
Introduction

Introduction

Review: General structure of **feedback control systems**

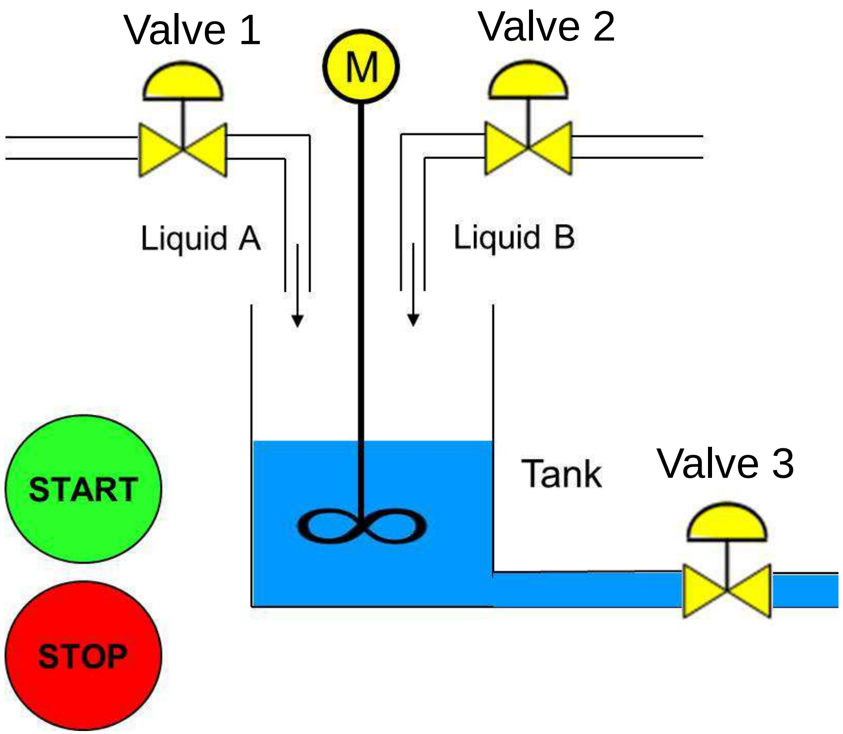


Review: **Feedback control** of water level in a tank



Typical specification: The water level in the tank should track the given setpoint signal.

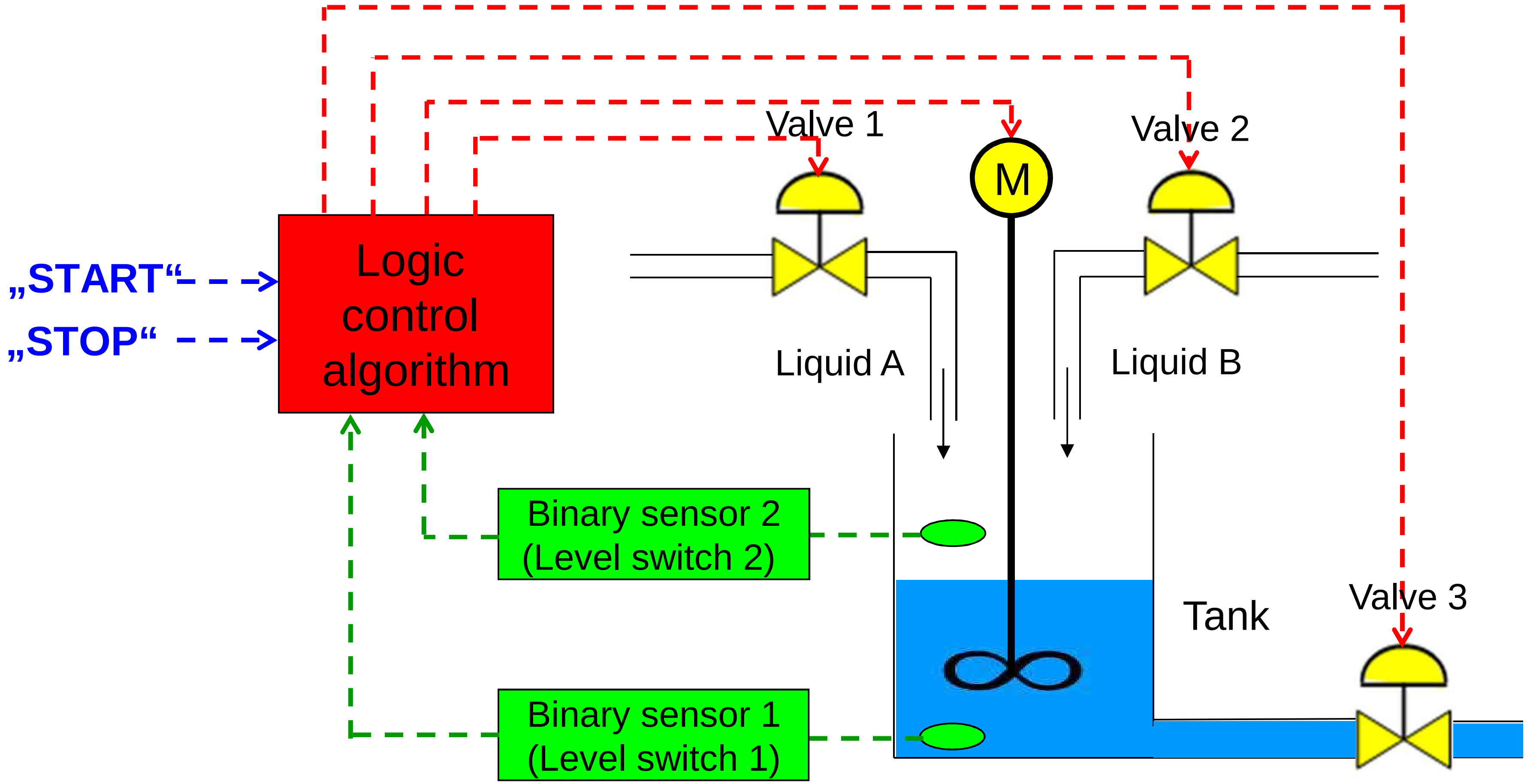
Problem formulation: Control of a mixed tank system



Specifications:

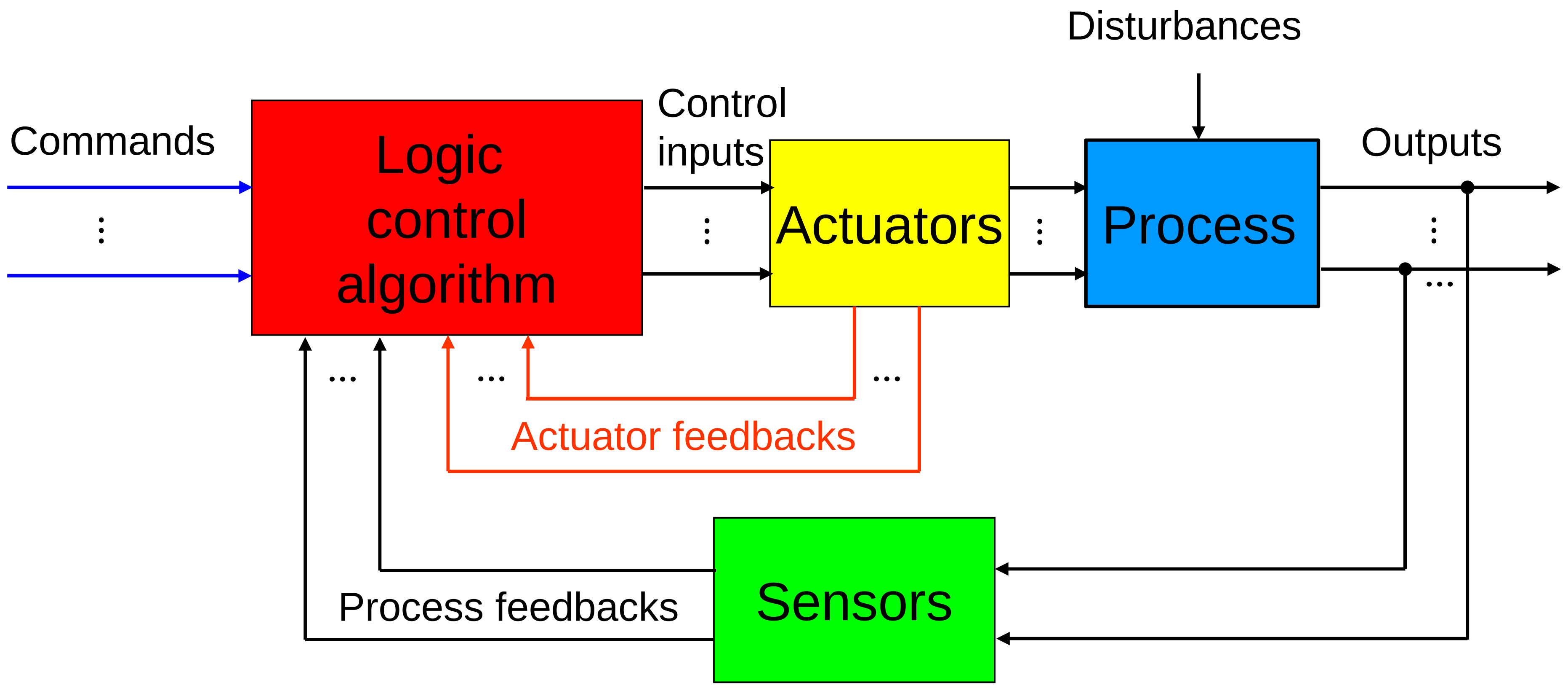
- If the **button START** is pressed, **open Valve 1** and **Valve 2** to fill in, respectively, the liquid A and B.
- When **the tank level reaches Level 2**, **close** both **Valve 1** and **Valve 2** and **start** the **motor** of the mixer.
- After 10 minutes, **turn off** the **motor** of the mixer and **open Valve 3**.
- When **the tank is empty**, **close Valve 3**.
- If the **button STOP** is pressed at any time, stop the process immediately.

Problem solution: Control of a mixed tank system



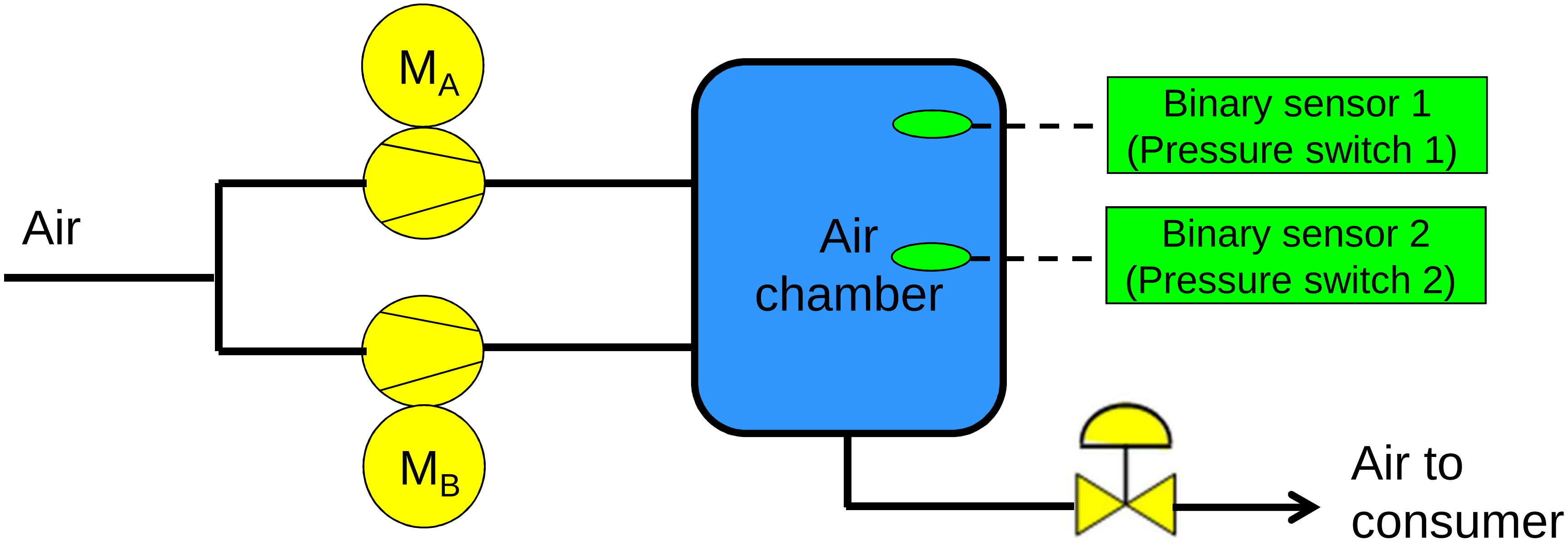
Introduction

General structure of **logic control systems**



Examples of logic control

Example 1: **Air chamber** (Litz, 2012)

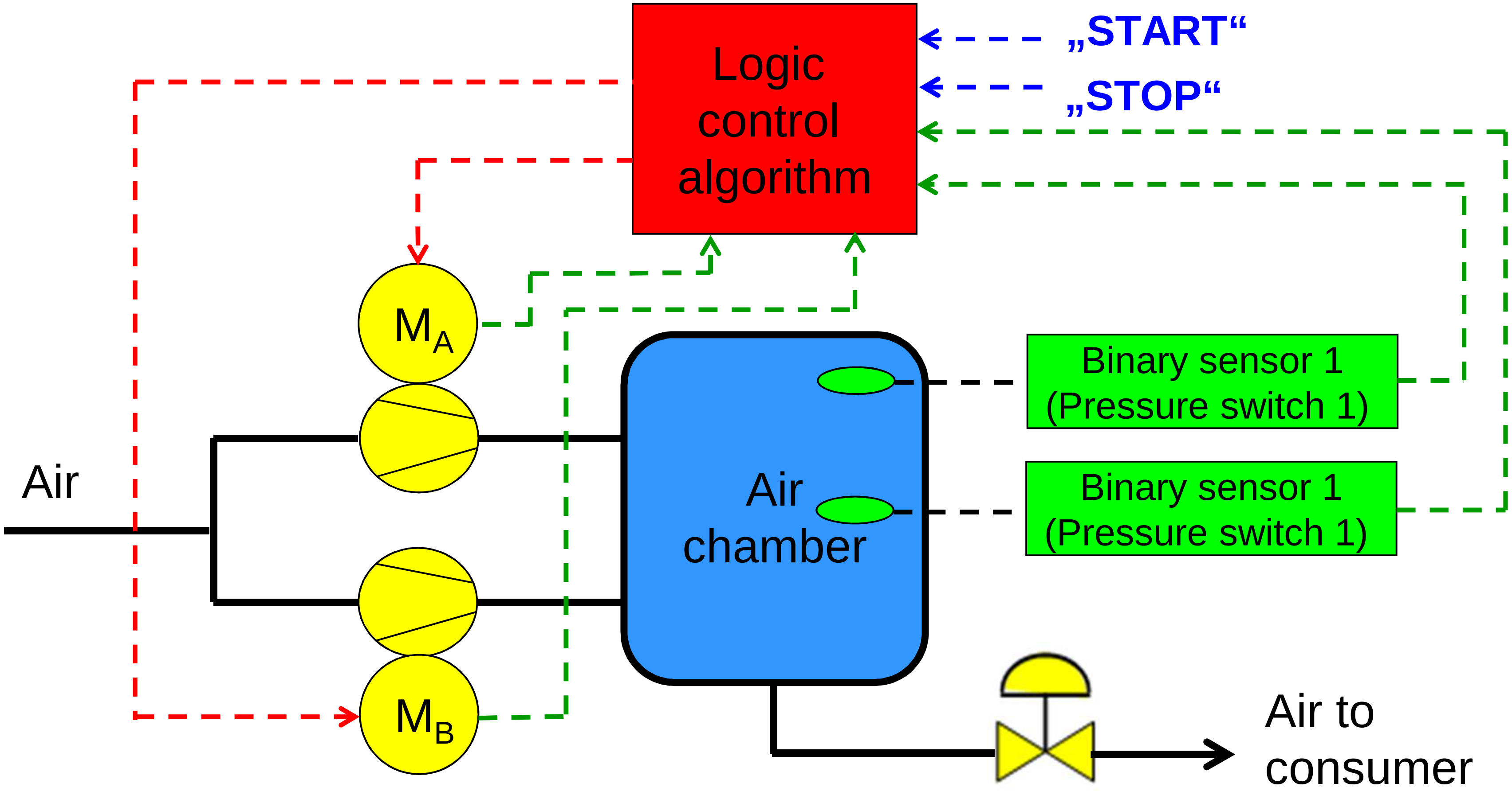


Specifications:

- If Pressure > 6.1 bar, no motor should run.
- If Pressure < 6.1 bar, one motor should run.
- If Pressure < 5.9 bar, both motor should run.
- The two motors should run alternatively.
- If one motor is defect, the other one should take over its work.

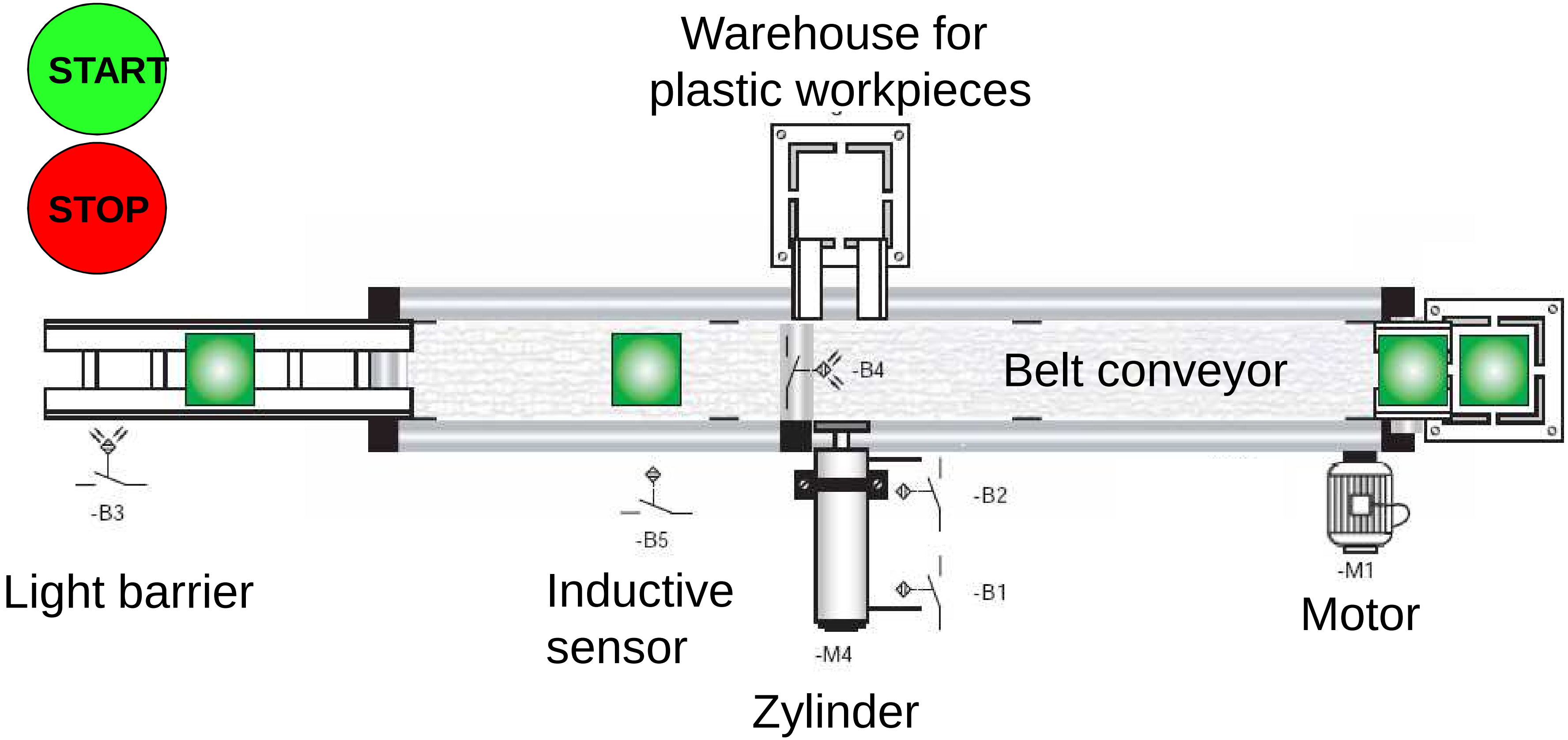
Examples of logic control

Example 1: **Air chamber** (Litz, 2012)



Examples of logic control

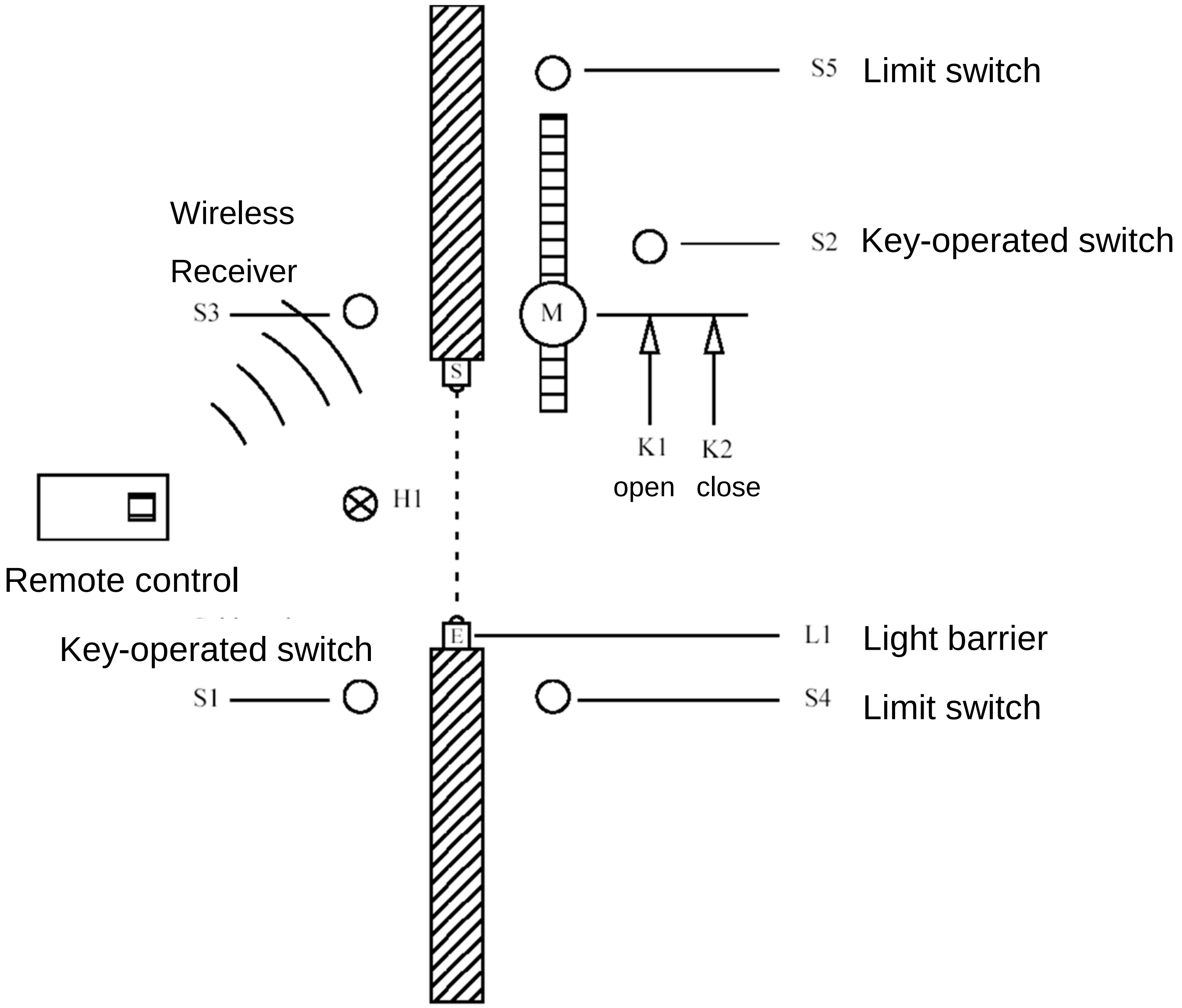
Example 2: **Sorting system**



* See, for instance, the following website for a demo of a similar sorting system:
http://www.kahlert.com/web/steuerungsortieranlage.php

Examples of logic control

Example 3: **Gate control**



Examples of logic control

Specifications:

- The gate can be opened and closed by controlling the electric motor. The rotary direction of the electric motor (**clockwise rotation** or **anti-clockwise rotation**) is controlled by two contactors.
- The gate can be opened from outside or inside by the **key-operated switches**, or by a **remote control unit**.
- For safety reasons, a **light barrier** is mounted. If the light barrier senses an object, the gate should not be closed.
- **Two limit switches** notify the states of the gate (i.e. „gate is opened“ or „gate is closed“).
- The completely opened gate should be automatically closed after 20 seconds waiting time.
- A **red flashing light** should notify the opening and closing of the rolling gate on both sides.

TECHNISCHE UNIVERSITÄT KAISERSLAUTERN

Logic Control vs. Feedback Control

	Logic control	Feedback control
Comparison unit	No	Yes
Stability	not relevant	important
Value range of input/ output signals	discrete (often 0/1)	continuous
Specifications	No standard formulation	Standard formulation: 1. guarantee stability 2. track setpoints 3. suppress disturbances
Number of input / output signals	quite a lot	limited
Robustness to unknown disturbances	only known disturbances can be handled	considered
Models	Boolean algebra, Automata, Petri nets	Differential / Difference equation

Applications of logic control

Home

Building

Manufacturing industry

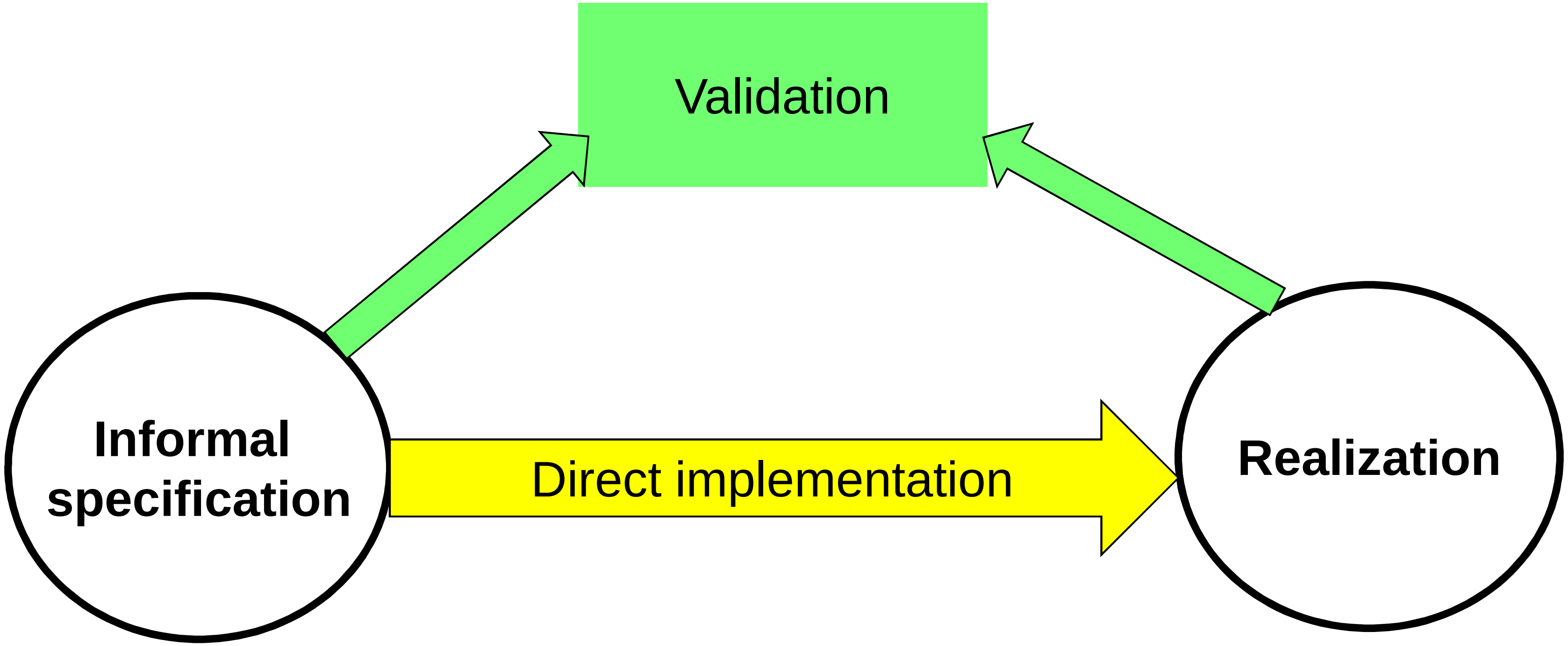
Logistics

Transport

Process industry

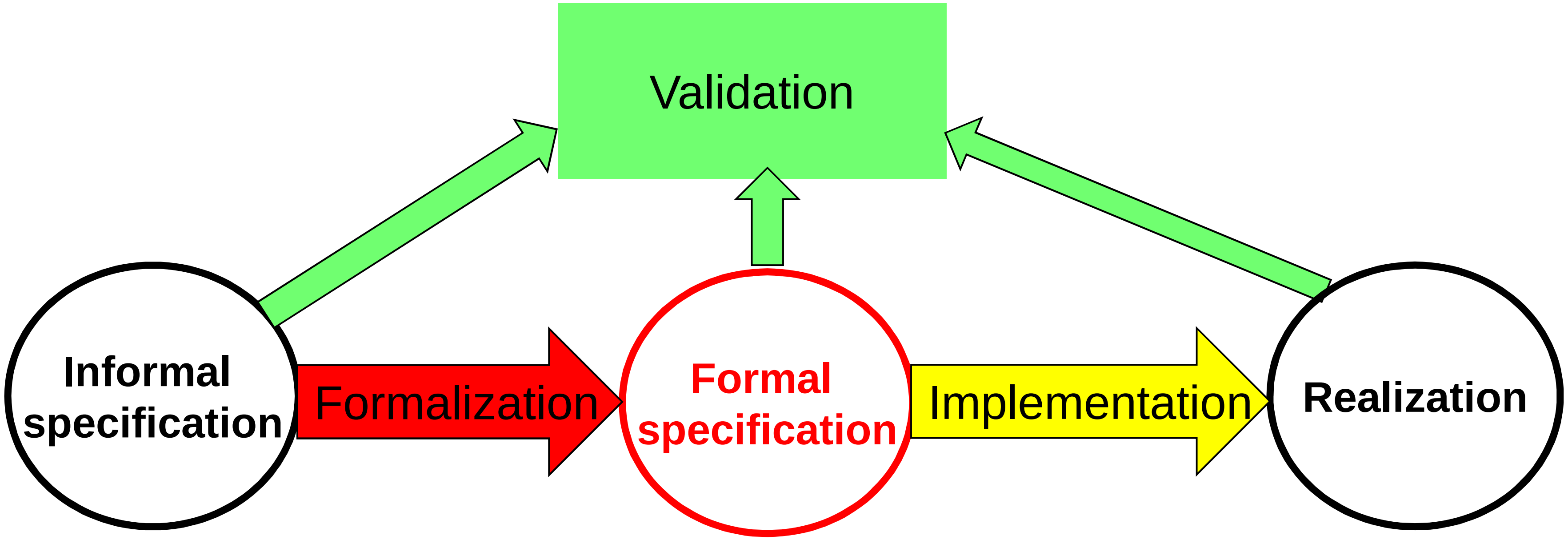
Development procedure of logic control systems

Conventional development procedure



Development procedure of logic control systems

Formal methods



Advantages:

- Clear (unambiguous) specification
- Systematic solution
- Find out problems earlier
- Reduce implementation efforts

Professional communities

- VDI/VDE GMA (Gesellschaft der Mess- und Automatisierungstechnik) **Technical Committee 1.50 „Methoden der Steuerungstechnik“**
- NAMUR (International User Association of Automation Technology in Process Industries) **Working Group 2.1 „PLC and DCS“**
- IFAC (International Federation of Control) TC (Technical Committee) **1.3 “Discrete Event and Hybrid Systems”**
- IEC (International Electrotechnical Commission) TC (Technical Committee) **65 “Industrial-process measurement, control and automation”**
- **PLCopen** - International Not-For-Profit organization that promotes the propogation of IEC 61131 programming standard.
- …

TECHNISCHE UNIVERSITÄT KAISERSLAUTERN

Overview of the course

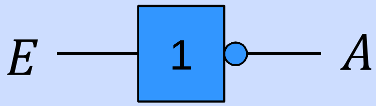
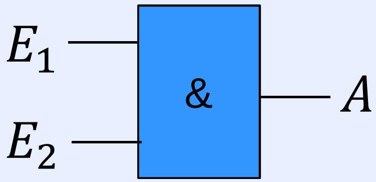
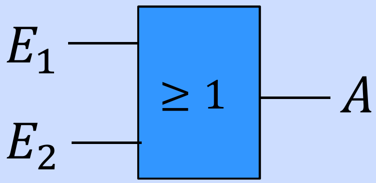
- **Introduction**
- **Modeling of logic control systems**
  - **Boolean algebra**
  - Finite state automata
  - Petri nets, SIPN
- Analysis of logic control systems
- Design of logic control systems
- Verification and validation
- Online diagnosis of logic control systems
- Implementation of logic control systems
  - PLC
  - Programming languages (IEC 61131-3)
  - Automatic code generation
- Distributed control (optional)

Chapter 2

Modelling of logic control systems

Boolean Algebra

- Defined for binary variables
- Basic operations: **AND**, **OR**, **NOT**

Operation	Logic diagram	Symbol	Truth table
NOT		$A = \bar{E} = \neg E$	E A 0 1 1 0
AND (conjunction)		$A = E_1 E_2 = E_1 \wedge E_2$	E2 E1 A 0 0 0 0 1 0 1 0 0 1 1 1
OR (disjunction)		$A = E_1 + E_2 = E_1 \vee E_2$	E2 E1 A 0 0 0 0 1 1 1 0 1 1 1 1

Laws of Boolean Algebra

Complements: $a \wedge \bar{a} = 0,\qquad a \vee \bar{a} = 1$

Identity: $a \wedge 1 = a,\qquad a \vee 0 = a$

Commutativity: $a \wedge b = b \wedge a,\qquad a \vee b = b \vee a$

Associativity: $a \wedge b \wedge c = (a \wedge b) \wedge c = \mathrm{a} \wedge (b \wedge c)$

$a \vee b \vee c = (a \vee b) \vee c = \mathrm{a} \vee (b \vee c)$

Distributivity: $\mathrm{a} \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$

$\mathrm{a} \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$

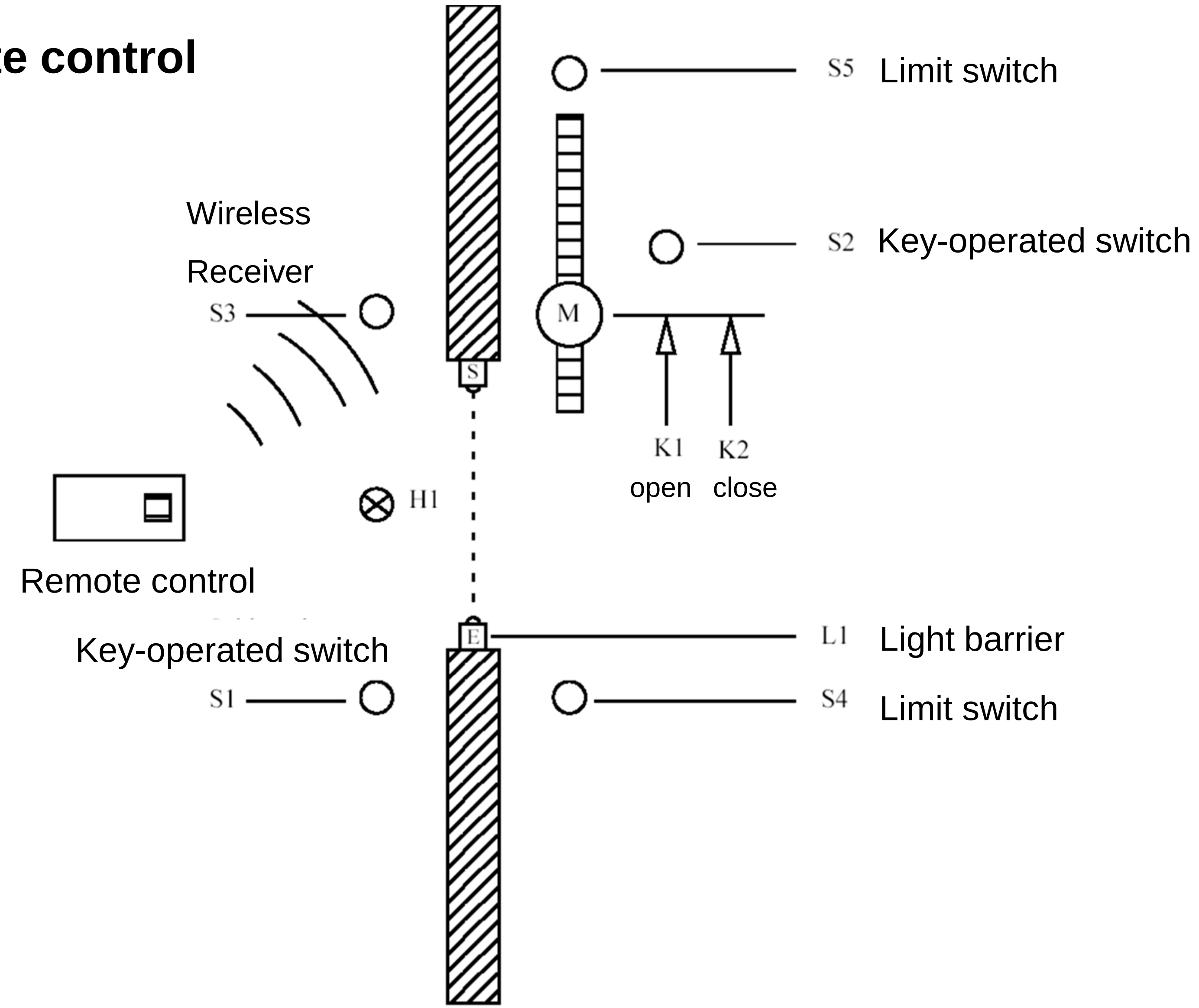
Idempotence: $a \wedge a = a,\qquad a \vee a = a$

Absorption: $\mathrm{a} \vee (a \wedge b) = a,\qquad \mathrm{a} \wedge (a \vee b) = a$

De Morgan's Law: $\overline{a \wedge b} = \bar{a} \vee \bar{b},\qquad \overline{a \vee b} = \bar{a} \wedge \bar{b}$

Modeling by Boolean Algebra

Example 1: Gate control



Modeling by Boolean Algebra

Specifications:

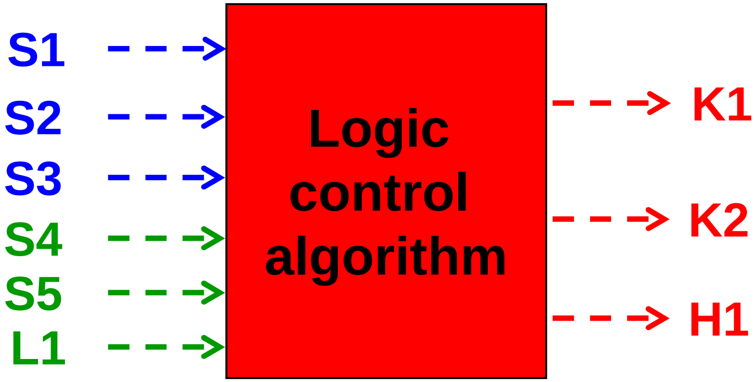
- The gate can be opened and closed by controlling the electric motor. The rotary direction of the electric motor (**clockwise rotation** or **anti-clockwise rotation**) is controlled by two contactors.
- The gate can be opened from outside or inside by the **key-operated switches**, or by a **remote control unit**.
- For safety reasons, a **light barrier** is mounted. If the light barrier senses an object, the gate should not be closed.
- **Two limit switches** notify the states of the gate (i.e. „gate is opened“ or „gate is closed“).
- ~~The completely opened gate should be automatically closed after 20 seconds waiting time.~~
- A **red flashing light** should notify the opening and closing of the rolling gate on both sides.

TECHNISCHE UNIVERSITÄT KAISERSLAUTERN

Modeling by Boolean Algebra

Signals	I/O	Symbol	Logic assignment
Key-operated switch (outside)	I	S1	Operated S1=1
Key-operated switch (inside)	I	S2	Operated S2=1
Wireless receiver	I	S3	Code received S3=1
Limit switch (gate opened)	I	S4	Gate is opened S4=1
Limit switch (gate closed)	I	S5	Gate is closed S5=1
Light barrier	I	L1	Interrupted L1=0
Flash light	O	H1	Light on H1=1
Contactor (opening gate)	O	K1	Contactor activated K1=1
Contactor (closing gate)	O	K2	Contactor activated K2=1

Modeling by Boolean Algebra

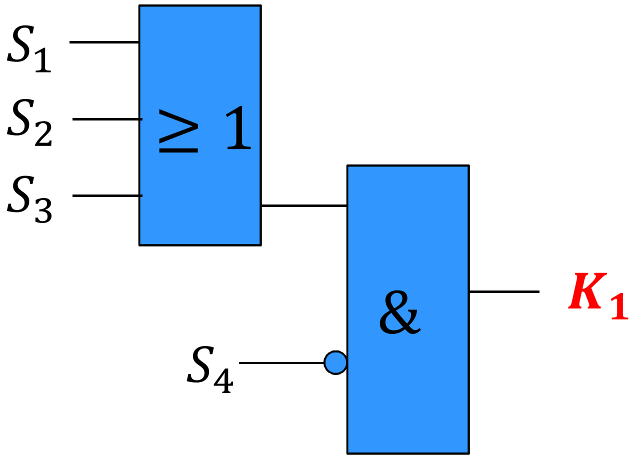


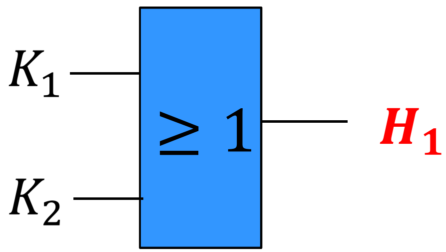
$$K_1 = (S_1 \vee S_2 \vee S_3) \wedge \bar{S}_4$$

$$H_1 = K_1 \vee K_2$$

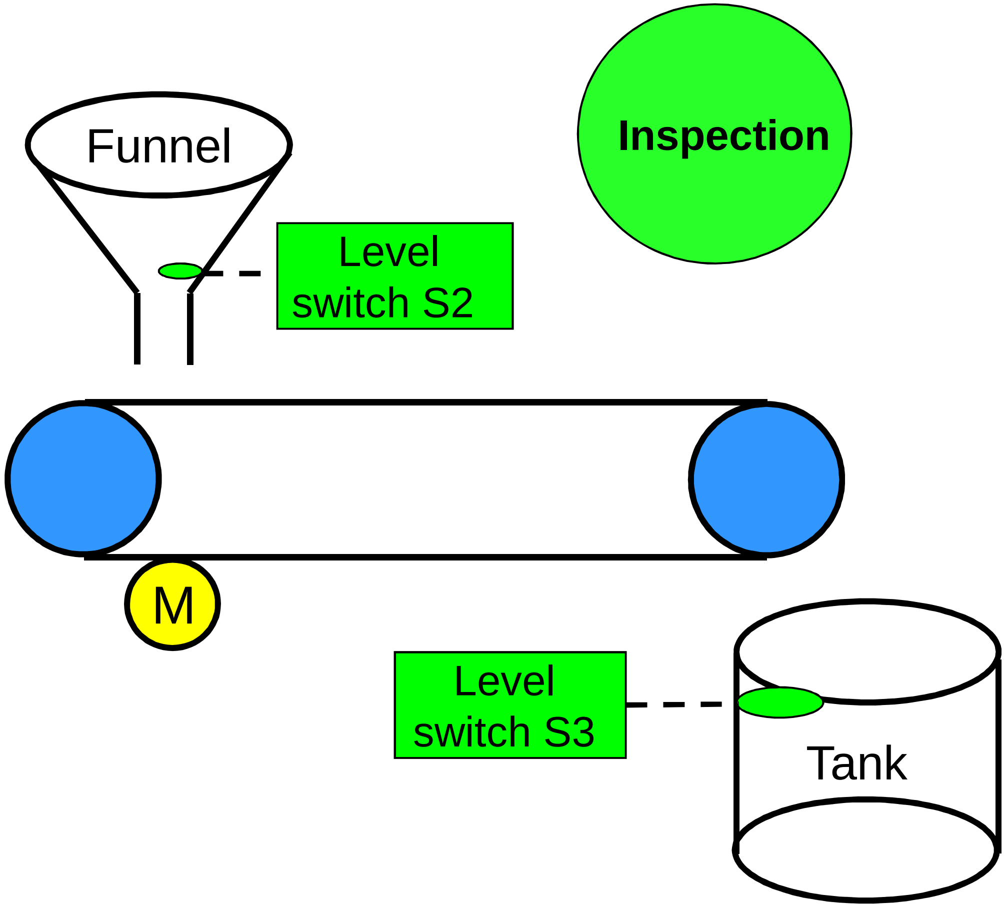
K_2 will be discussed later.

The model can also be expressed in form of logic diagrams:





Example 2: Control of Belt conveyor

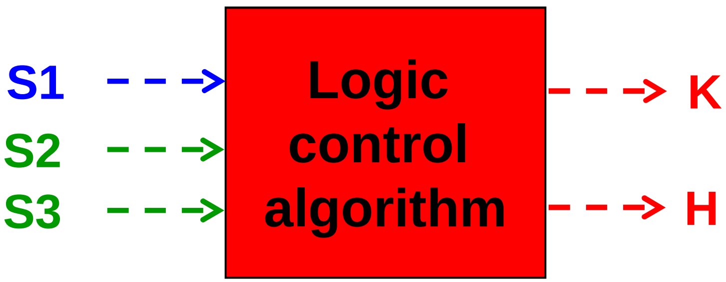


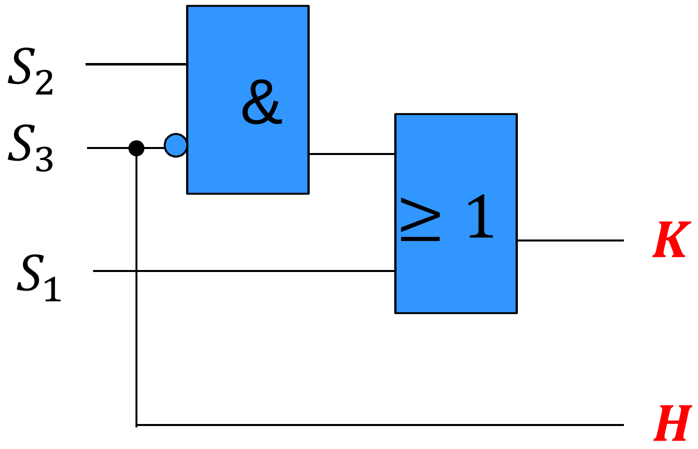
Specifications:

- The **motor** that drives the belt conveyor **runs** when **the funnel is not empty** and **the tank is not full**.
- The **motor** that drives the bet conveyor runs when the **button "INSPECTION"** is pressed.
- If the tank is full, an **indicator lamp lights up**.

Modeling by Boolean Algebra

Signals	I/O	Symbol	Logic assignment
Button “inspection”	I	S1	S1=1, if the button is pressed
Level in funnel	I	S2	S2=1, if the level is not empty
Level in tank	I	S3	S3=1, if the level is full
Indictor lamp	O	H	The lamp lights up, if H=1
Motor	O	K	Motor runs, if K=1





$$K = S_1 \vee (S_2\bar{S}_3)$$
$$H = S_3$$

Normal Forms of Boolean Algebra

- Two popular normal forms:
  - **Disjunction normal form (DNF)**: Disjunctions of conjunctions
  - **Conjunction normal form (CNF)**: Conjunctions of disjunctions

Truth table

Inputs			Output
E_1	E_2	E_3	K
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

DNF:

$$K = (\bar{E}_1 \bar{E}_2 E_3) \vee (\bar{E}_1 E_2 \bar{E}_3) \vee (\bar{E}_1 E_2 E_3)$$

Normal Forms of Boolean Algebra

- Two popular normal forms:
  - **Disjunction normal form (DNF)**: Disjunctions of conjunctions
  - **Conjunction normal form (CNF)**: Conjunctions of disjunctions

Truth table

Inputs			Output
E_1	E_2	E_3	K
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

CNF:

$$K = (E_1 \vee E_2 \vee E_3)$$
$$\wedge (\bar{E}_1 \vee E_2 \vee E_3)$$
$$\wedge (\bar{E}_1 \vee E_2 \vee \bar{E}_3)$$
$$\wedge (\bar{E}_1 \vee \bar{E}_2 \vee E_3)$$
$$\wedge (\bar{E}_1 \vee \bar{E}_2 \vee \bar{E}_3)$$

Obtain Normal Forms by Karnough Map

Simplified DNF/CNF forms can be obtained by applying **Karnough Map**.

Truth Table

Inputs			Output
E_1	E_2	E_3	K
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

Karnough Map

E_1 \ E_2E_3	00	01	11	10
0	0	1	1	1
1	0	0	0	0

DNF:

$$K = (\bar{E}_1 E_2) \vee (\bar{E}_1 E_3)$$

Obtain Normal Forms by Karnough Map

Truth Table

Inputs			Output
E_1	E_2	E_3	K
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

Karnough Map

E_1 \ E_2E_3	00	01	11	10
0	**0**	**1**	**1**	**1**
1	**0**	**0**	**0**	**0**

CNF:

$$K = \bar{E}_1 \land (E_2 \lor E_3)$$