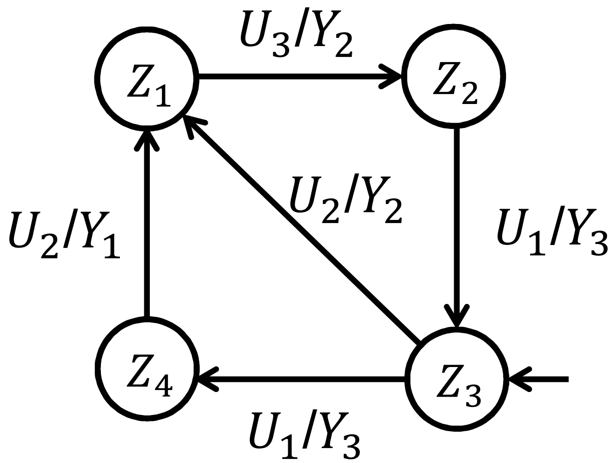
Logic Control

Prof. Dr. Ping Zhang

WS 2017/2018

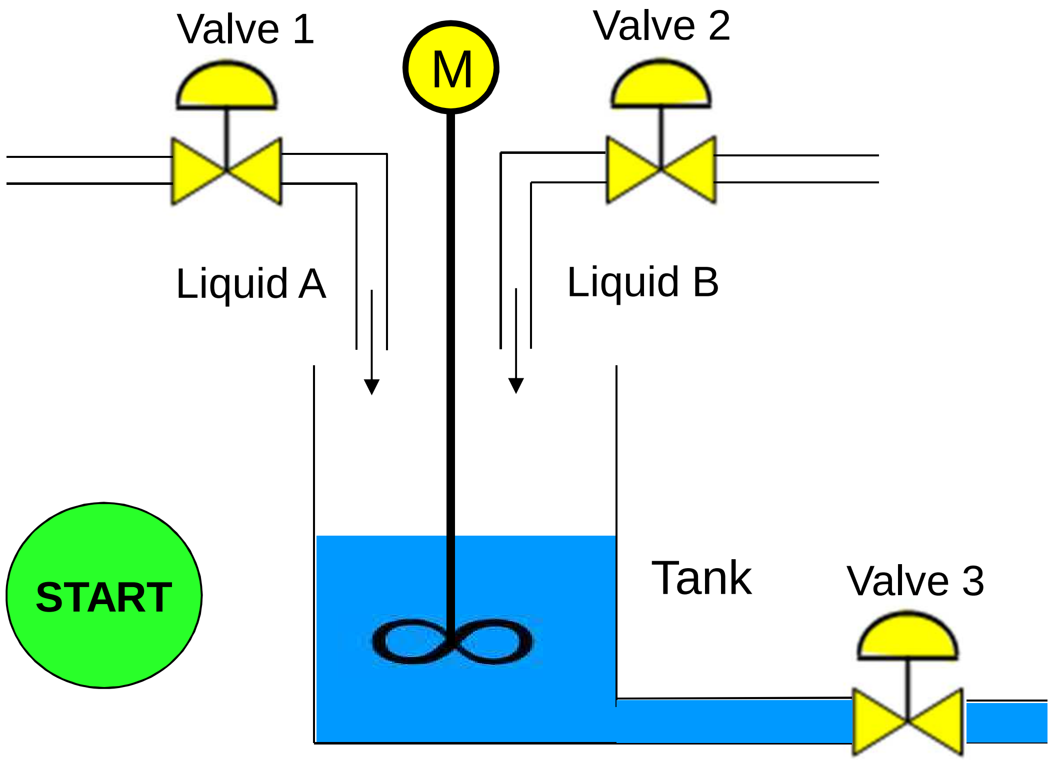
Overview of the course

- 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)



Modelling by SIPN: Mixing tank

Example 1: Mixing tank



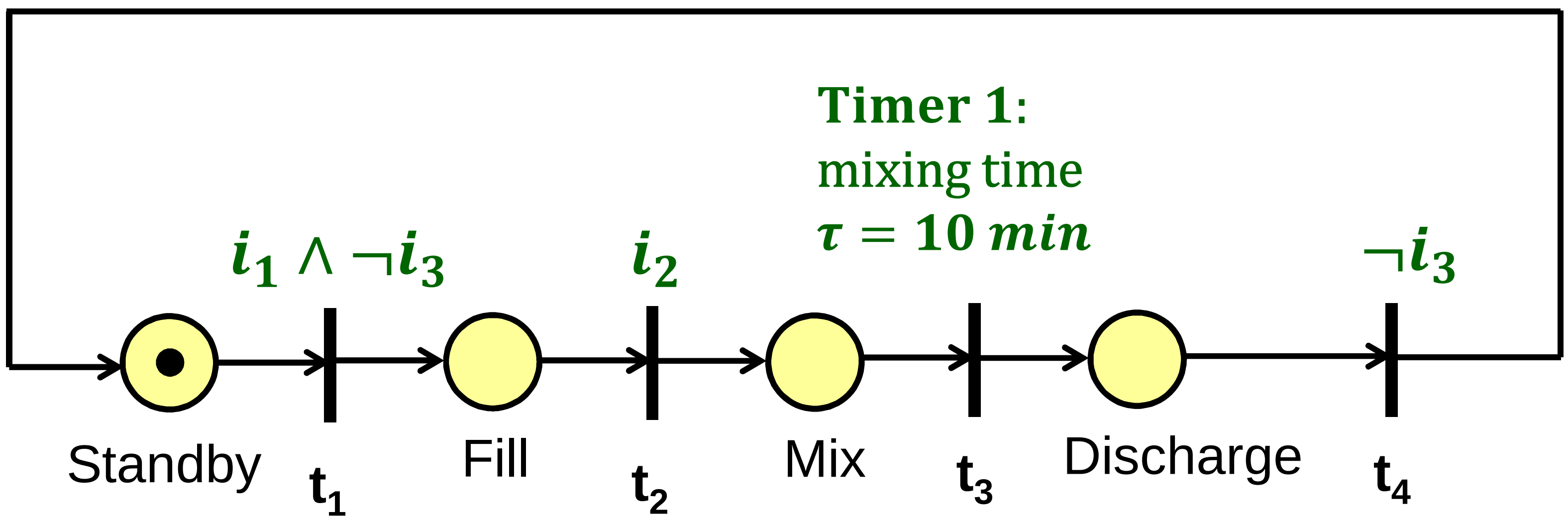
Specifications:

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

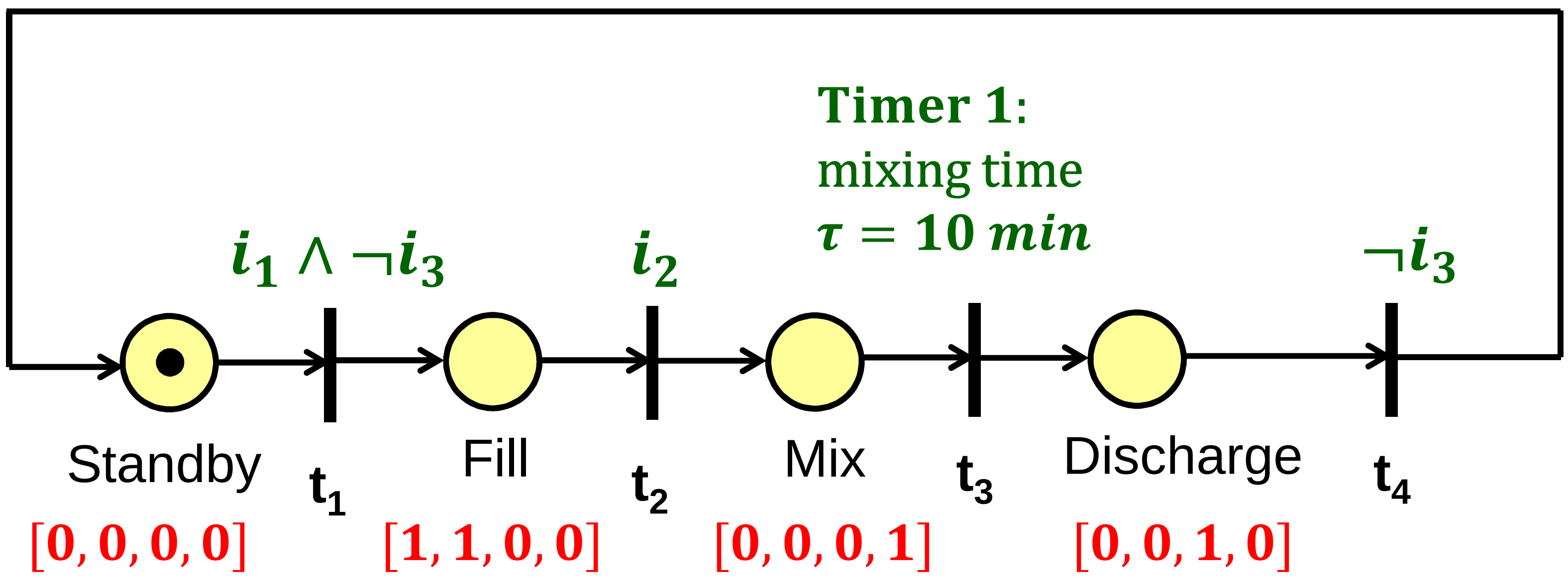
Modelling by SIPN: Mixing tank

Signals	I/O	Symbol	Logic assignment
START	I	i1	START button is pressed i1=1
Level switch 1	I	i2	Tank is full i2=1
Level switch 2	I	i3	Tank is empty i3=0
Valve 1	O	o1	Open Valve 1, o1=1
Valve 2	O	o2	Open Valve 2, o2=1
Valve 3	O	o3	Open Valve 3, o3=1
Motor	O	o4	Motor on, o4=1

Modelling by SIPN: Mixing tank

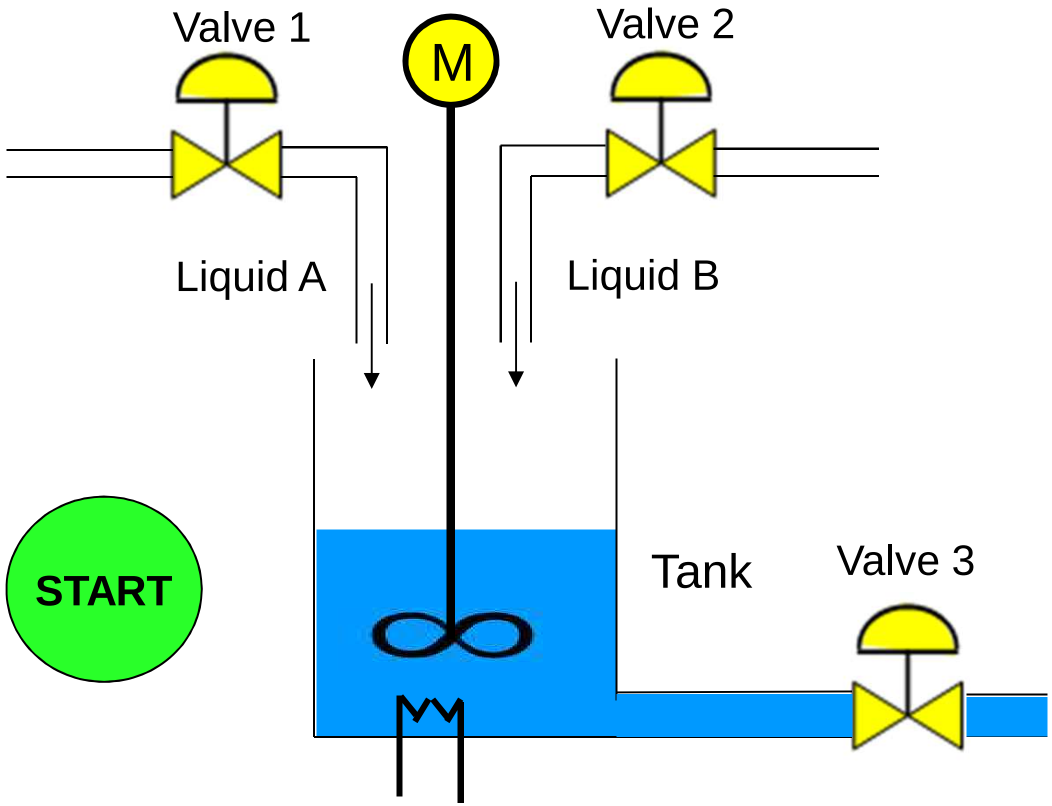


Modelling by SIPN: Mixing tank



Modelling by SIPN: Heated mixing tank

Example 2: Heated mixing tank



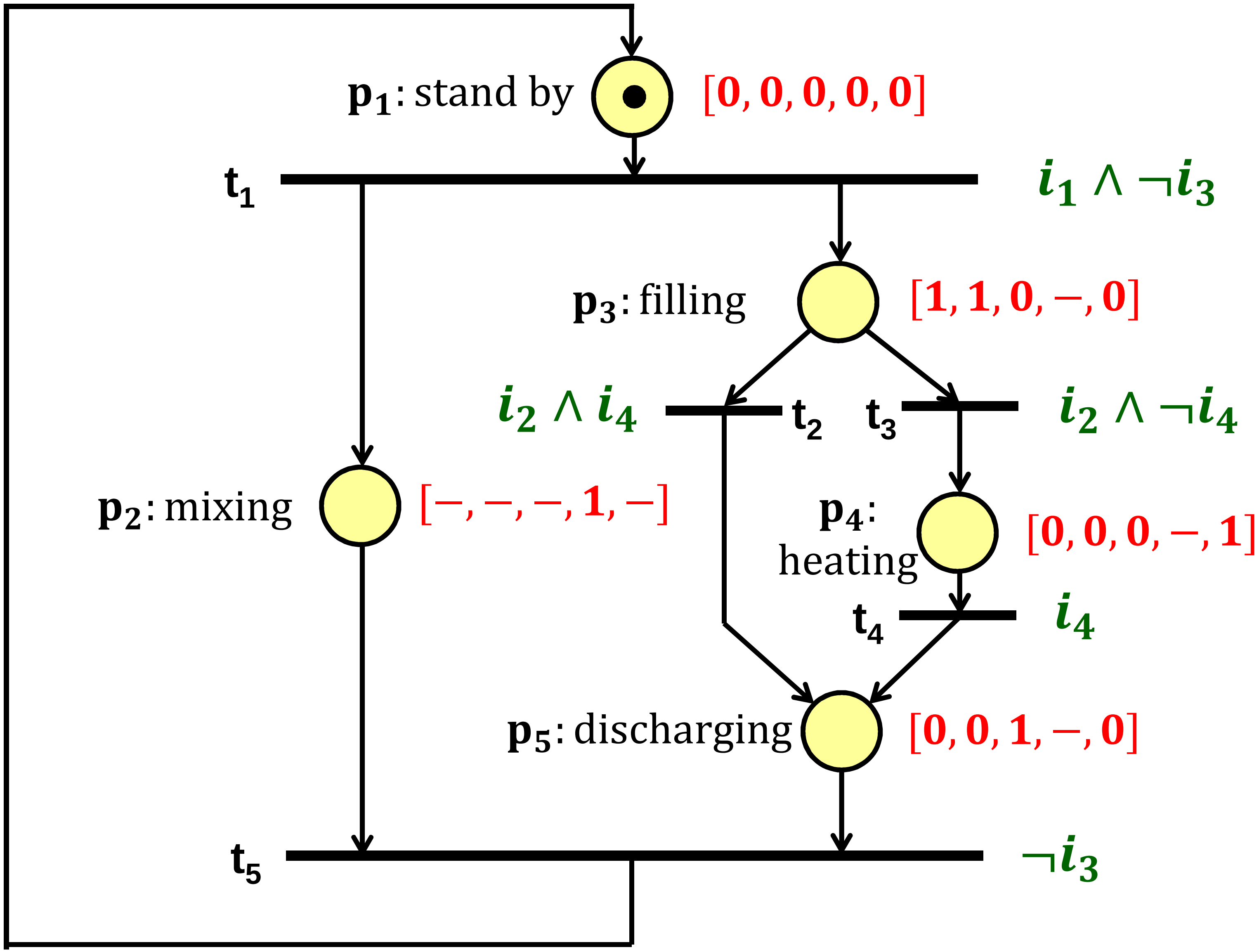
Specifications:

- After pressing the START button, the tank is filled by opening Valve 1 and Valve 2 and the motor of the mixer is started.
- Valve 1 and Valve 2 are closed if the tank is full.
- Then the mixture in the tank is heated, until a certain temperature is exceeded.
- After that Valve 3 is opened. If the tank is empty, close Valve 3 and stop the motor of the mixer.

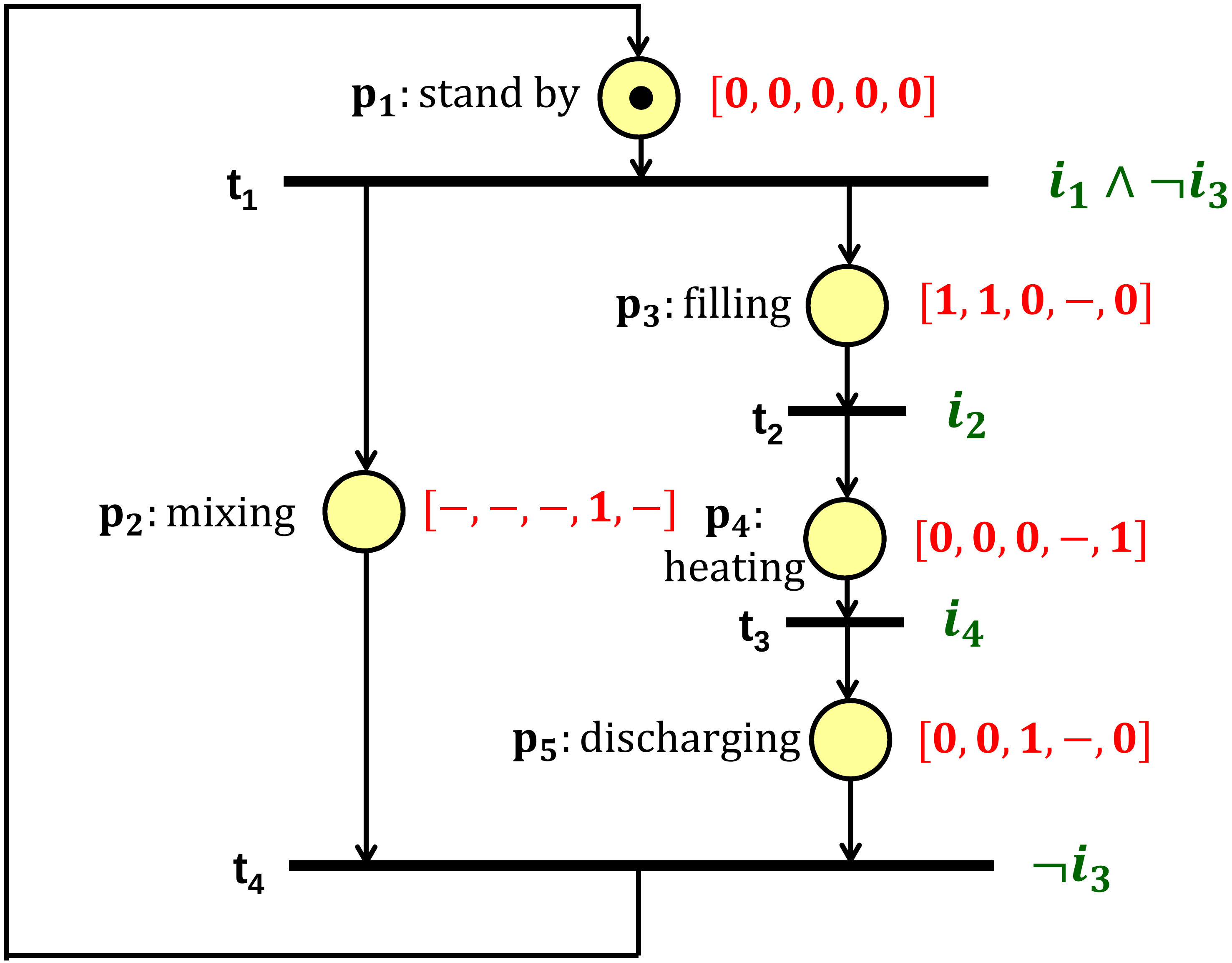
Modelling by SIPN: Heated mixing tank

Signals	I/O	Symbol	Logic assignment
START	I	i1	START button is pressed i1=1
Level switch 1	I	i2	Tank is full i2=1
Level switch 2	I	i3	Tank is empty i3=0
Temperature switch	I	i4	Temperature above limit i4=1
Valve 1	O	o1	Open Valve 1, o1=1
Valve 2	O	o2	Open Valve 2, o2=1
Valve 3	O	o3	Open Valve 3, o3=1
Motor	O	o4	Motor on, o4=1
Heater	O	o5	Heater on, o5=1

Modelling by SIPN: Heated mixing tank



Modelling by SIPN: Heated mixing tank

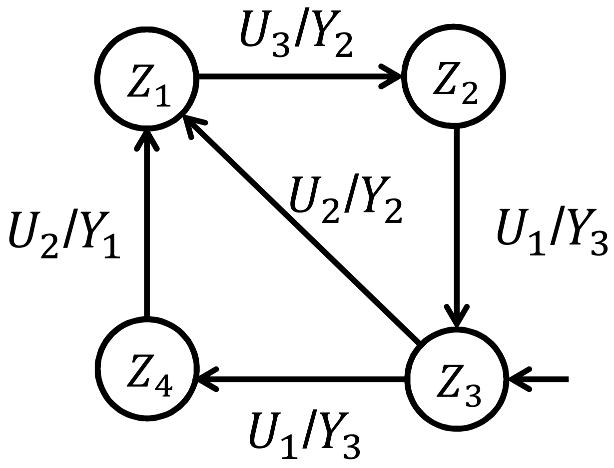


Summary of signal interpreted Petri nets

- The SIPN has been developed to take into account the **input and output signals** in logic control systems.
- The **input signals** incluence the **firing** of the transitions in the SIPN. The **places** in the SIPN decides the **output signals**.
- For complex systems, the SIPN can be built up in a **hierarchical** way (for instance, the place is indeed again an SIPN)

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 3

Analysis of logic control systems

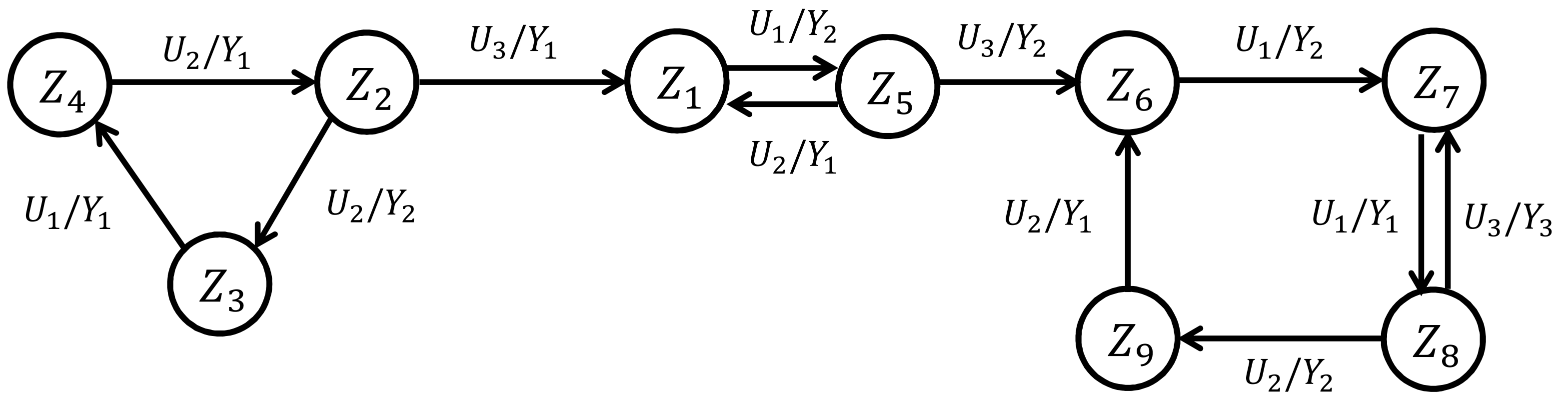
Analysis of logic control systems

- Based on the model of the logic control systems (Boolean algebra, FSA, PN, SIPN), the system behaviour can be analyzed **systematically**.
- Some often encountered questions regarding the characteristics of logic control systems:
  - Certain states should not be reached (for instance, due to safety considerations)
  - Some states should be reached
  - No deadlocks in the system
  - ......

Analysis based on Boolean Algebra

- There is no dynamics in the standard Boolean algebra based system description
  - analysis straightforward
- **Some recent research in logic control networks**, in which the state transitions in logic control systems are expressed by boolean expressions and can be handled by a recently proposed mathematical tool (the socalled **semi-tensor product**).

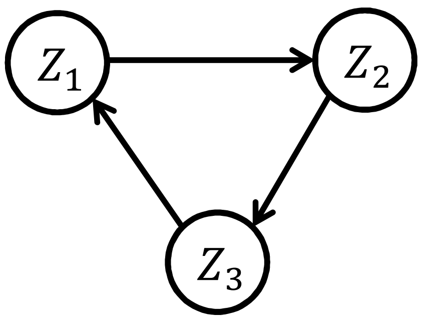
Analysis based on FSA



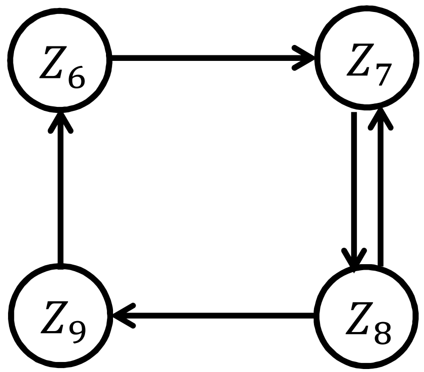
- **Reachability analysis** in the FSA
  - If a state z_j is on a path that begins at the state z_i, then z_j is **reachable** from z_i.
  - Correspondingly, the sequence of the inputs labeled on the arcs in the path are the **input sequence** that can control the state from z_i to z_j.

Analysis based on FSA

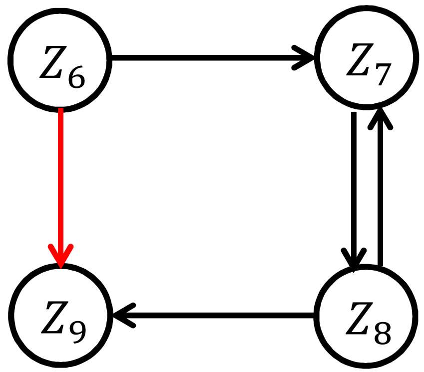
- **Structural analysis** in the FSA
  - Two states z_i and z_j are called **strongly connected**, if there is a path in the state transition diagram that goes from z_i to z_j and also a path from z_j to z_i .
  - Example:



$\{Z_1, Z_2, Z_3\}$
strongly connected



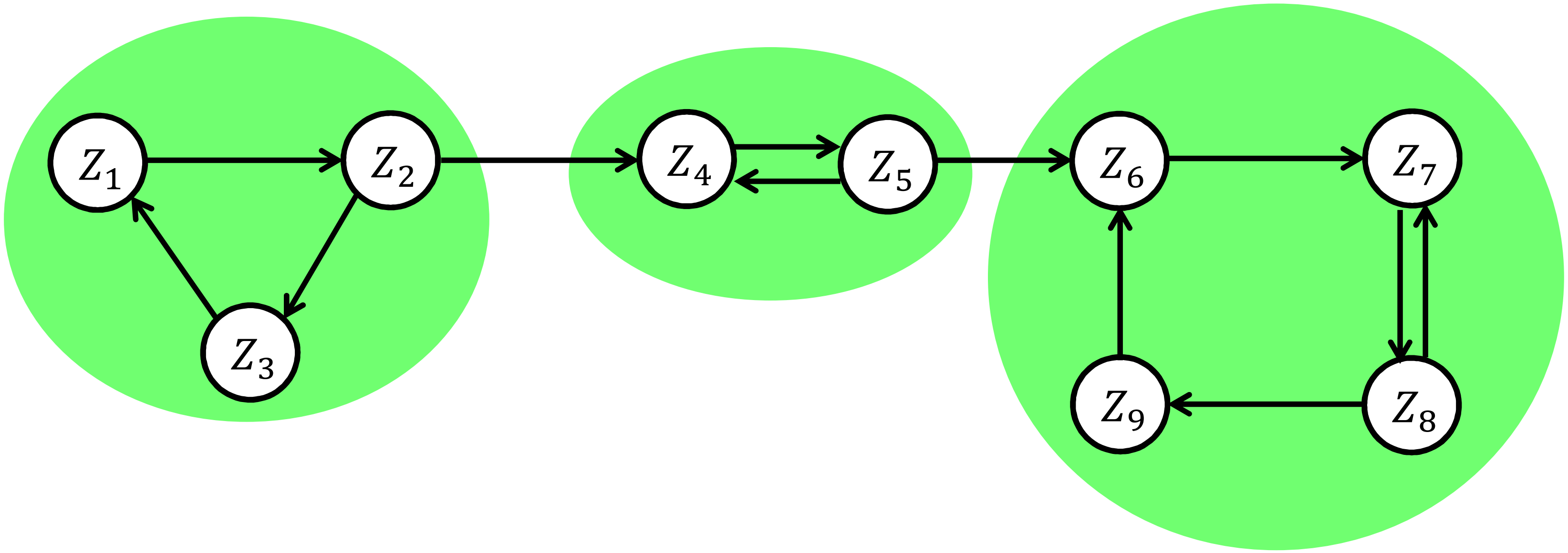
$\{Z_6, Z_7, Z_8, Z_9\}$
strongly connected



$\{Z_6, Z_7, Z_8, Z_9\}$
not strongly connected;
$\{Z_7, Z_8\}$ s.c.

Analysis based on FSA

- **Structural analysis** in the FSA
  - The state transition diagram can be **decomposed** into q sub-diagrams. The states in each sub-diagram are strongly connected with each other. The sub-diagrams are connected by one-directional arc.
  - If all the states in the FSA are strongly connected (i.e. $q = 1$), then the FSA is said to be **irreducible**. Otherwise, it is called reducible.
  - The states in an irreducible FSA are said to be **recurrent**.

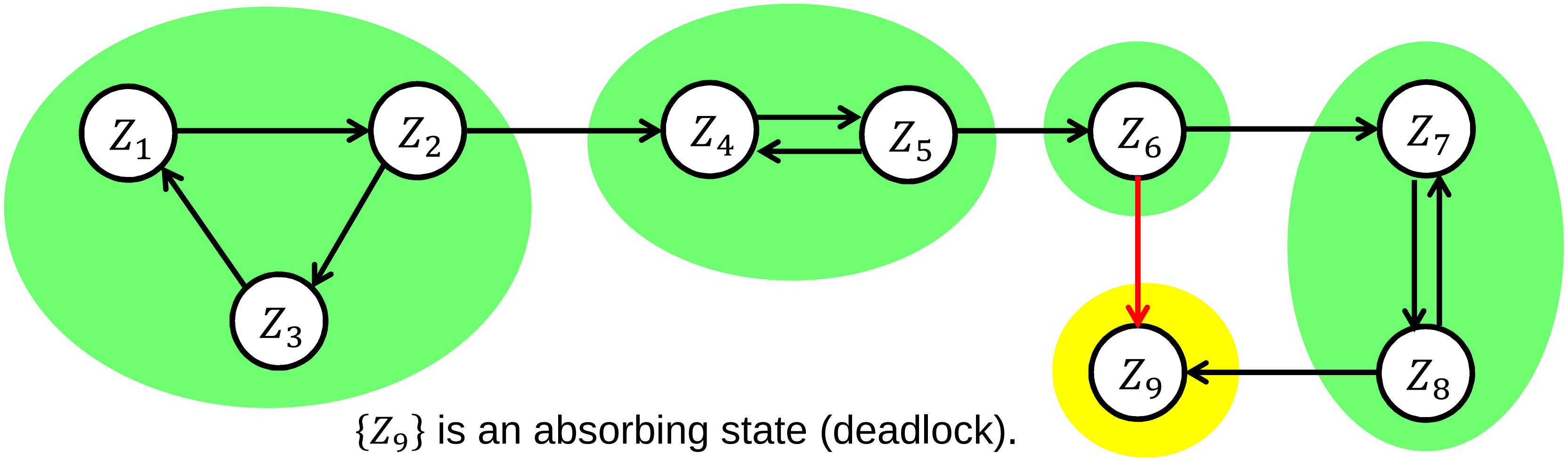


Analysis based on FSA

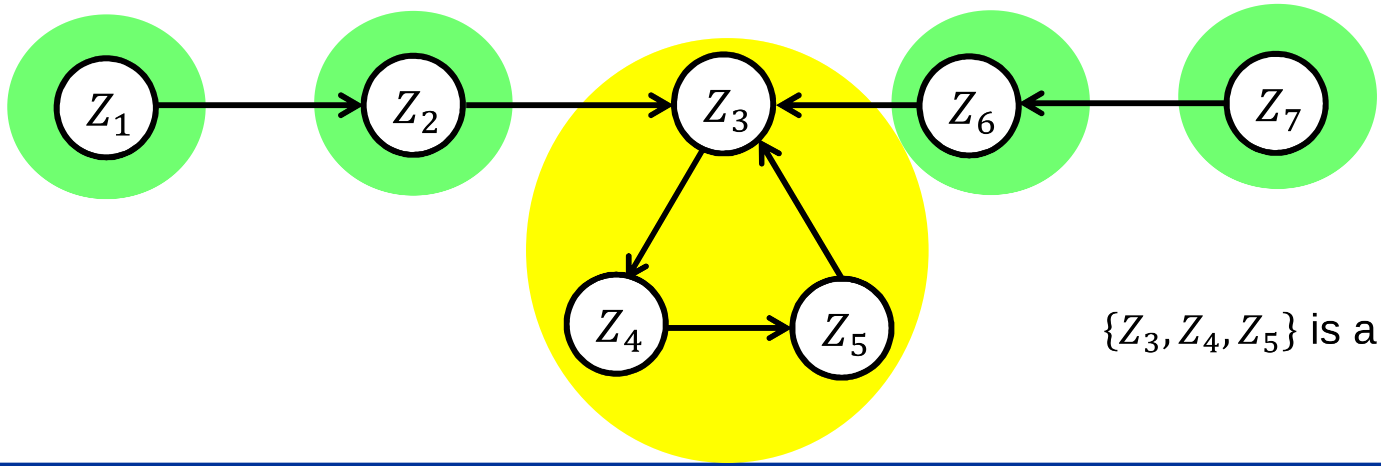
- **Structural analysis** in the FSA
  - The sub-diagram that has only incoming arcs (i.e. no arcs come out of this sub-diagram) is called **ergodic**.
  - If an ergodic sub-diagram contains only one state, then this state is said to be an **absorbing state (deadlock)**.
  - If an ergodic sub-diagram contains more than one state, then this sub-diagram is said to be an **livelock**.
  - The states in the state transition diagram that can only be reached ***once*** are called **transient states**.
  - An FSA is called **live**, if it can generate state transition sequence of arbitrary length.

Analysis based on FSA

Example:



$\{Z_9\}$ is an absorbing state (deadlock).

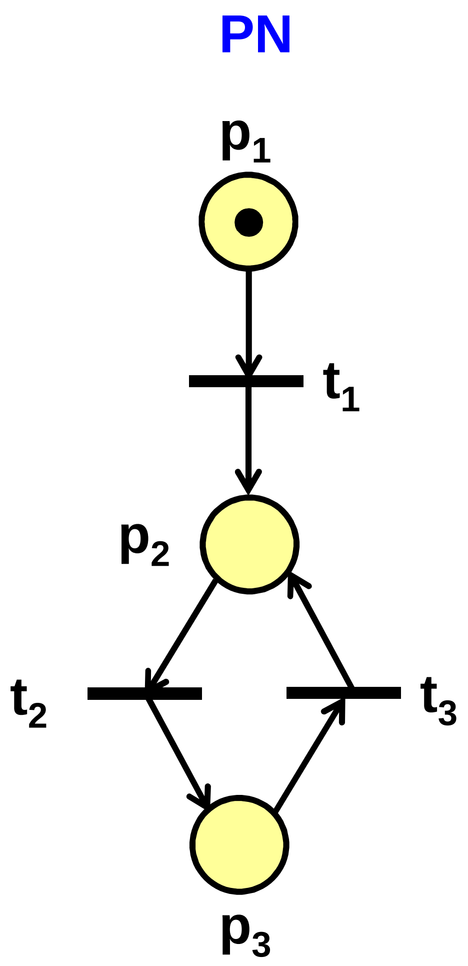


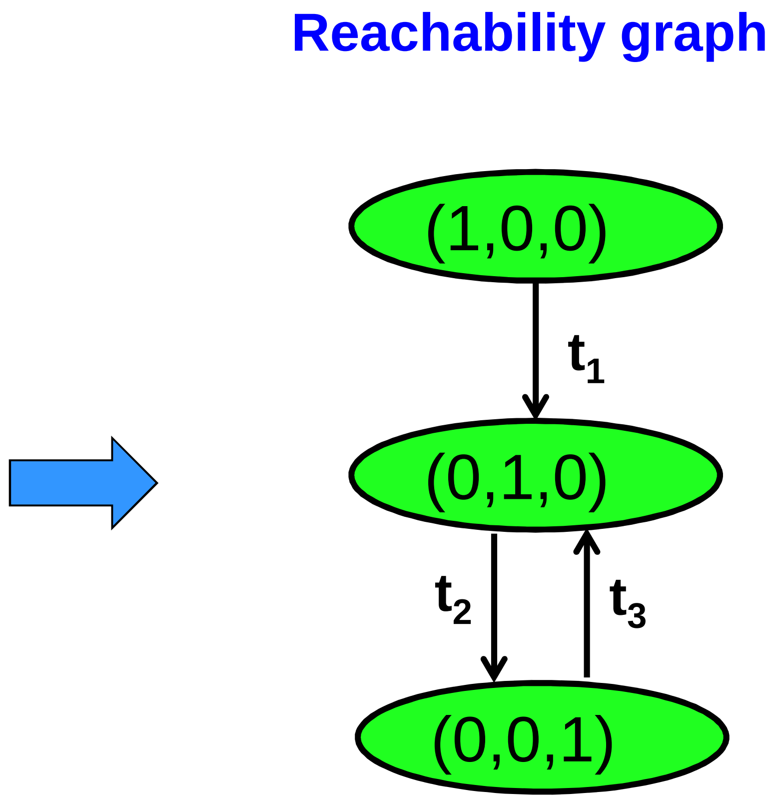
$\{Z_3, Z_4, Z_5\}$ is a livelock.

Analysis based on PN

- The main tool used in the analysis of PN is **reachability graph.**
- If a marking m_j can be produced from the marking m_i by firing of transitions, then m_j is **reachable** from m_i.
- The **reachable set** of a Petri net consists of the initial marking m_0 and all the markings that are reachable from m_0.
- The **reachability graph** is obtained as follows. The initial marking m_0 and all the markings that are reachable from m_0 are the nodes. If m_j is **reached** from m_i by the firing of the transition t_k, then there is a directed arc from m_i to m_j and this arc is labeled by t_k.

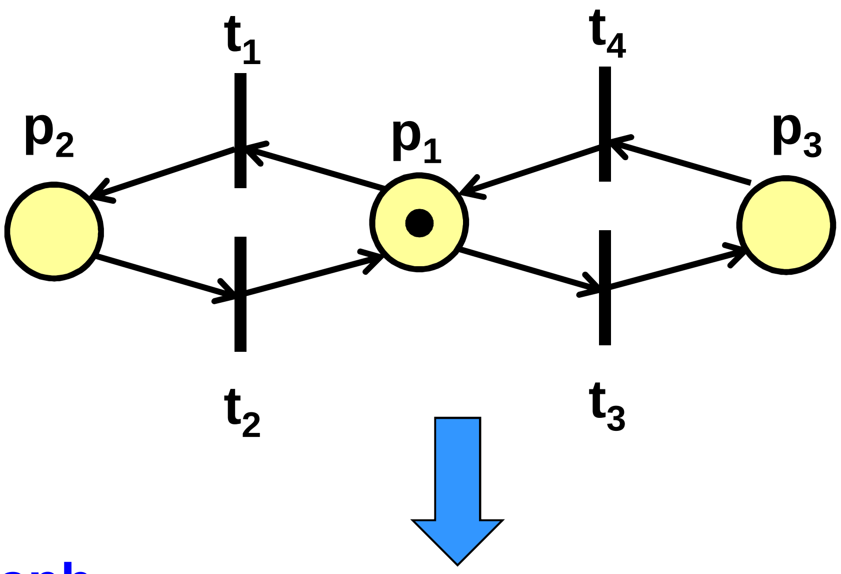
Analysis based on PN



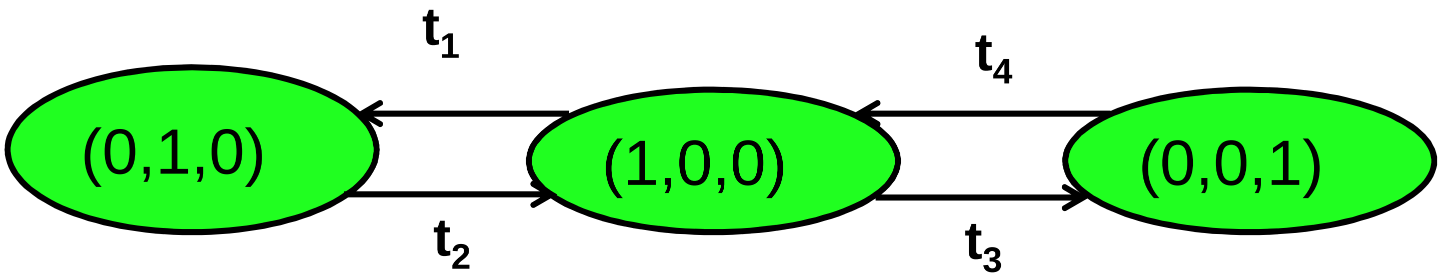


Analysis based on PN

PN

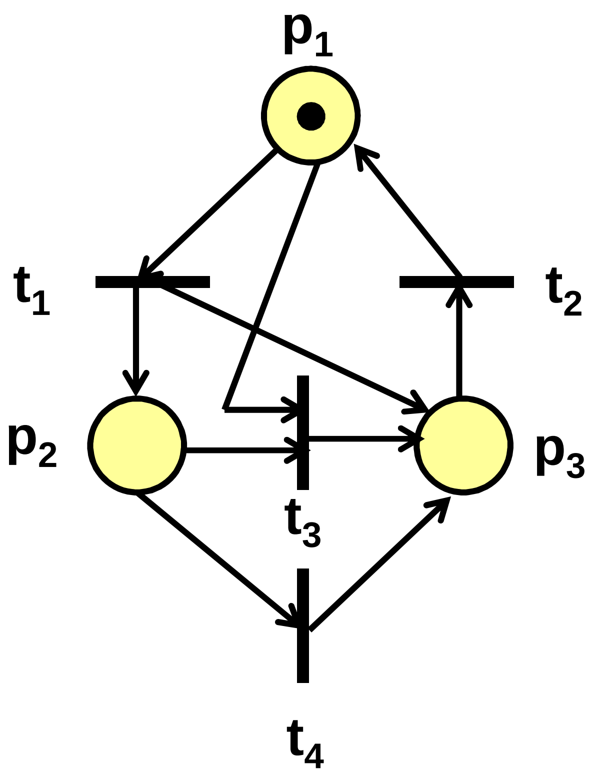


Reachability graph

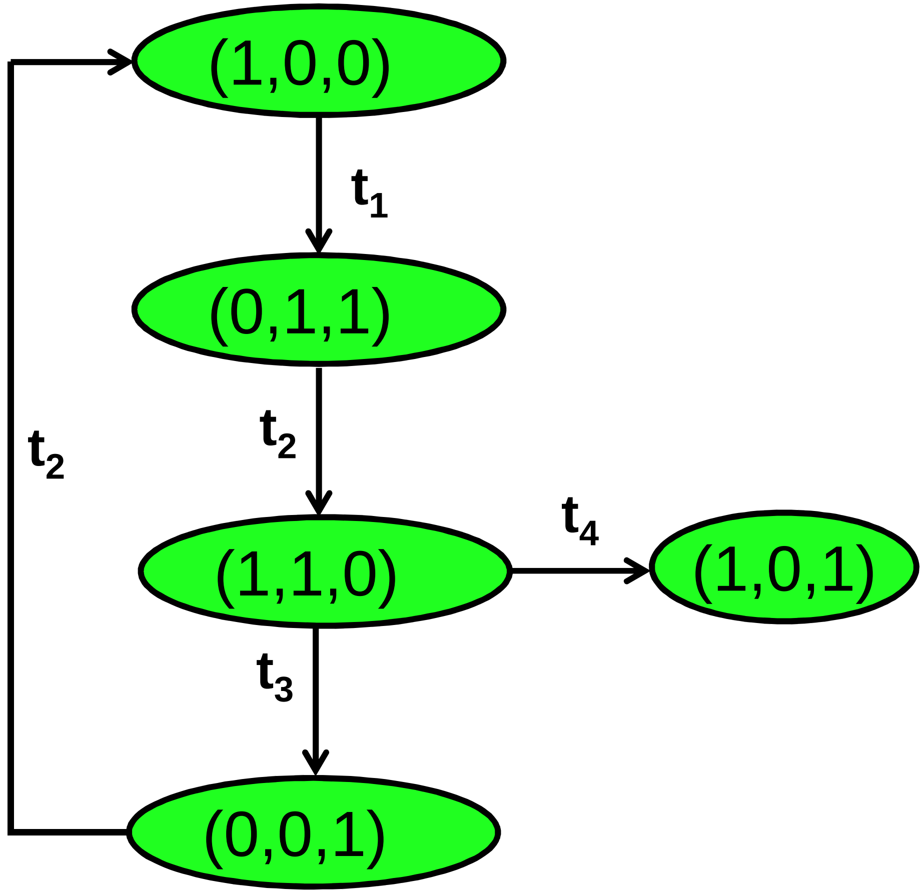


Analysis based on PN

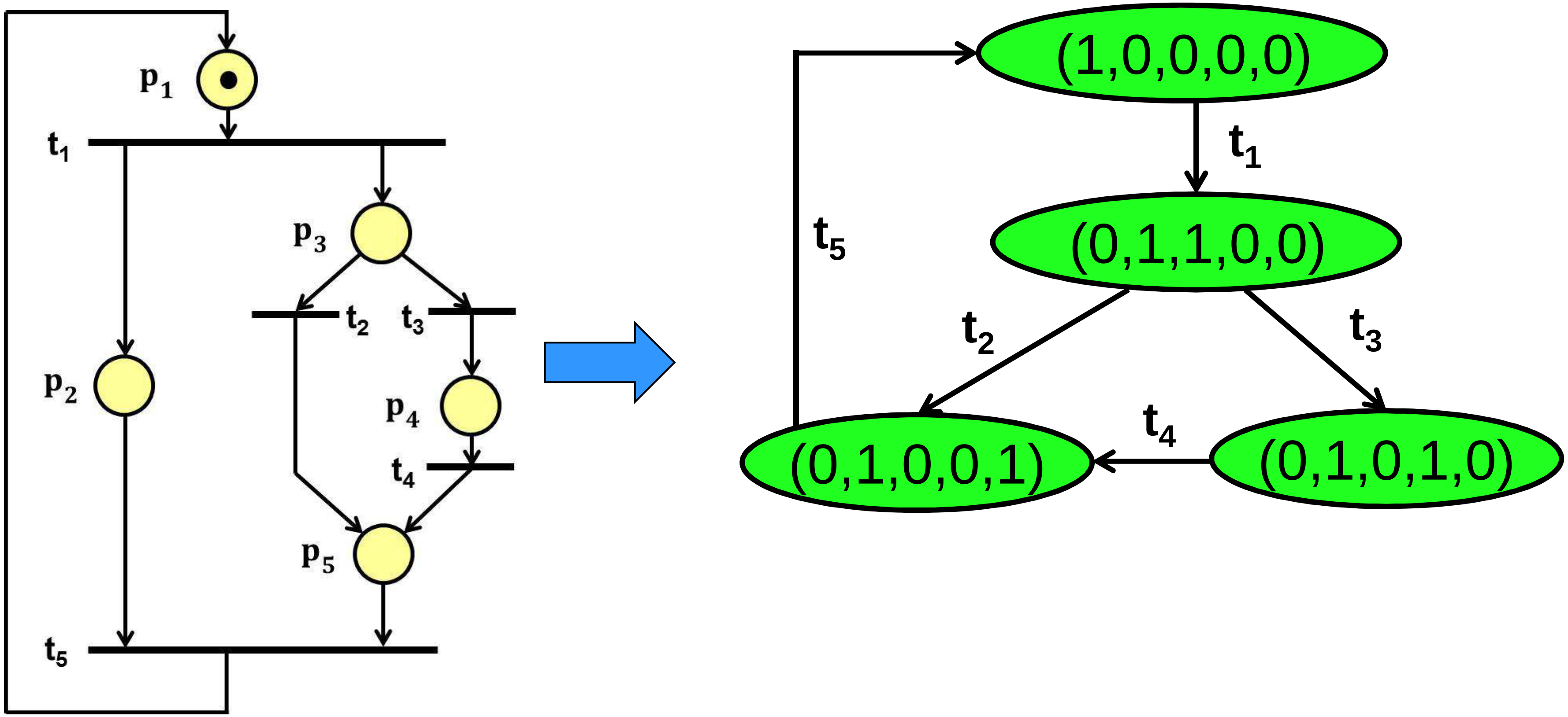
PN



Reachability graph

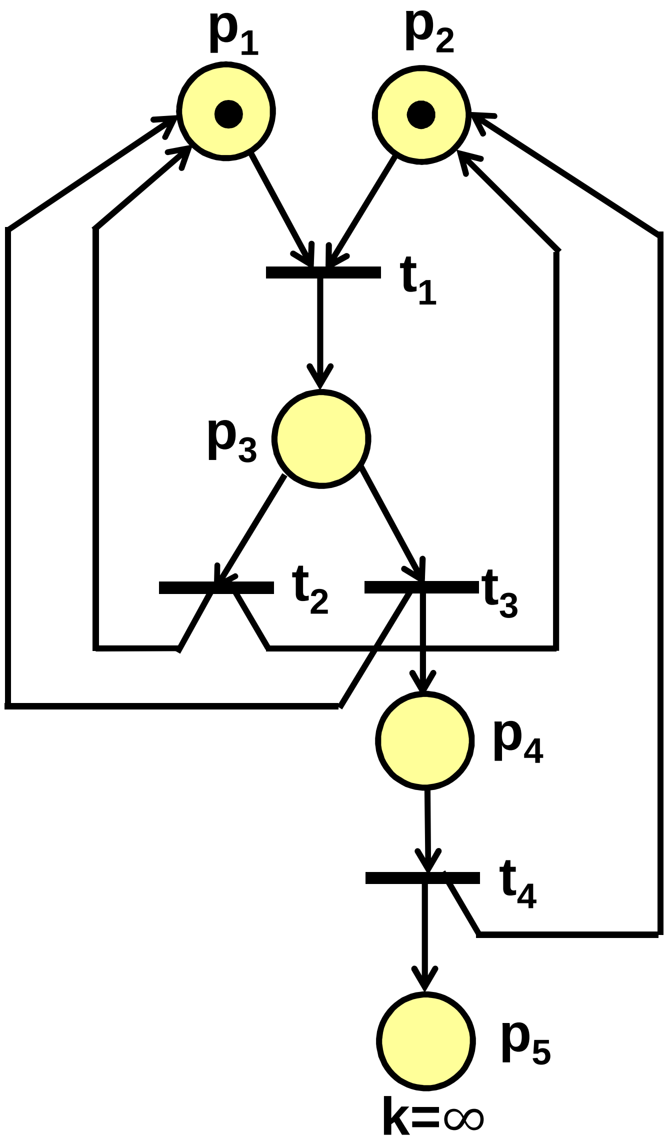


Analysis based on PN

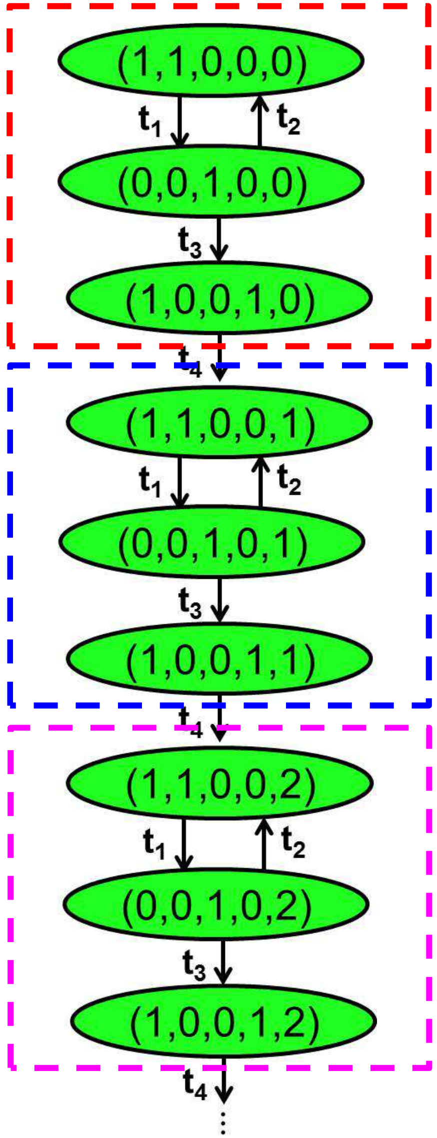


Analysis based on PN

PN

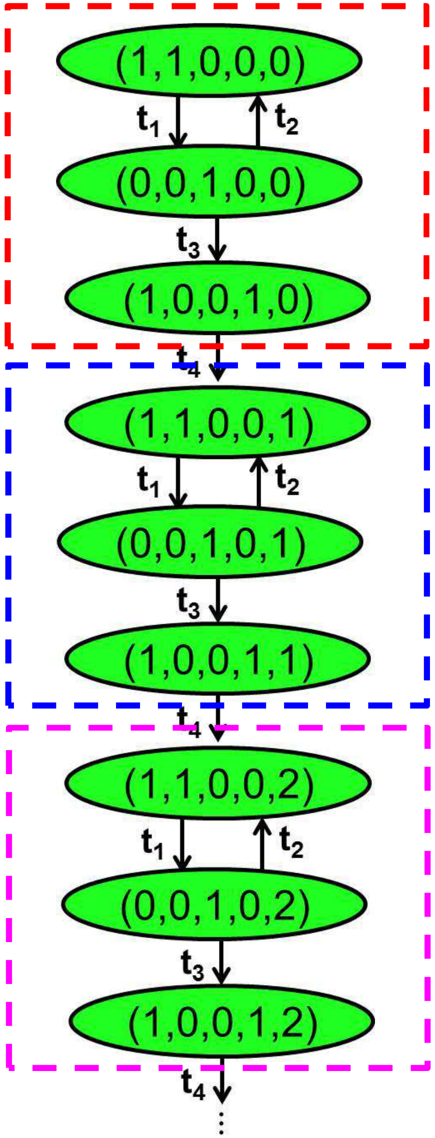


Reachability graph

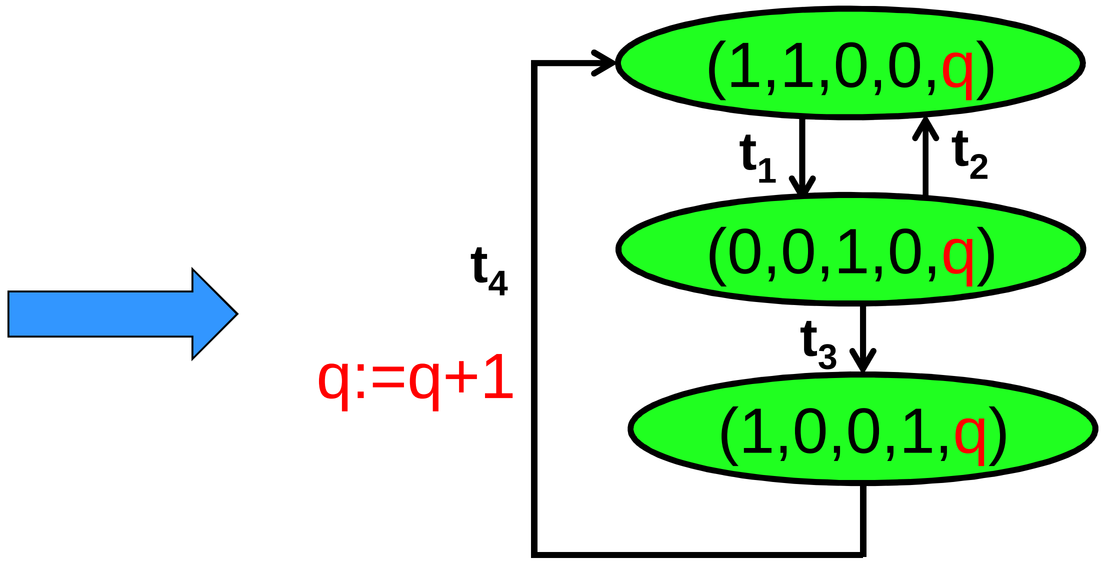


Analysis based on PN

Reachability graph



Coverability graph



An infinite number of reachable marking

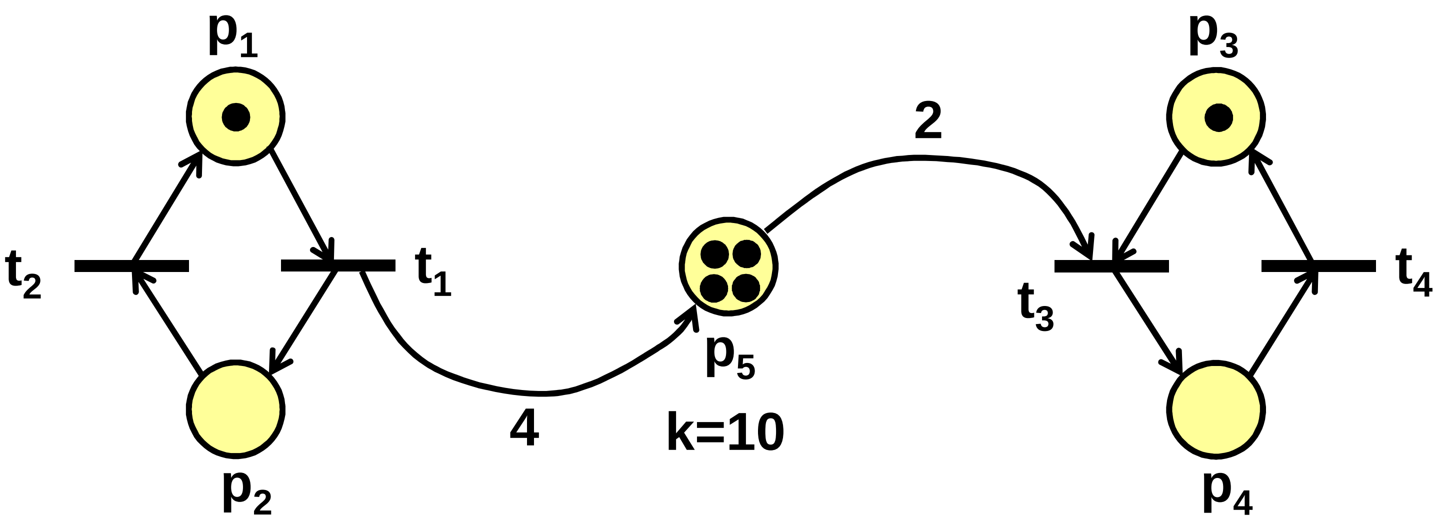
→ Infinite reachability graph

→ An additional variable q is introduced

→ coverability graph

Analysis based on PN

PN



Coverability graph