

# Sequential Function Chart SFC

# Why using graphical function charts ?

- ☑ Written language is :
  - Not always clear
  - Inefficient, clumsy
    - if ... then ... else ... but only when ...
  - ambiguous → misunderstanding
  - Only suitable for linear relations

# Why Sequential Function Charts:

- ☑ Easy to understand
- ☑ Well suited for parallel processes
- ☑ Large theoretical background (all theory of Petri-Nets applicable)
- ☑ Used in IEC61131-3 engineering tools

# Application areas of SFC and Petri-Nets

## Information Sciences:

- ☑ Representation of program evolution
- ☑ Modeling, analysis and programming of concurrent processes, distributed applications
- ☑ Modeling access to shared resources
- ☑ Data base access
- ☑ Real-time systems
- ☑ Modeling and simulation of queues

# Application areas of SFC and Petri-Nets

## Automation:

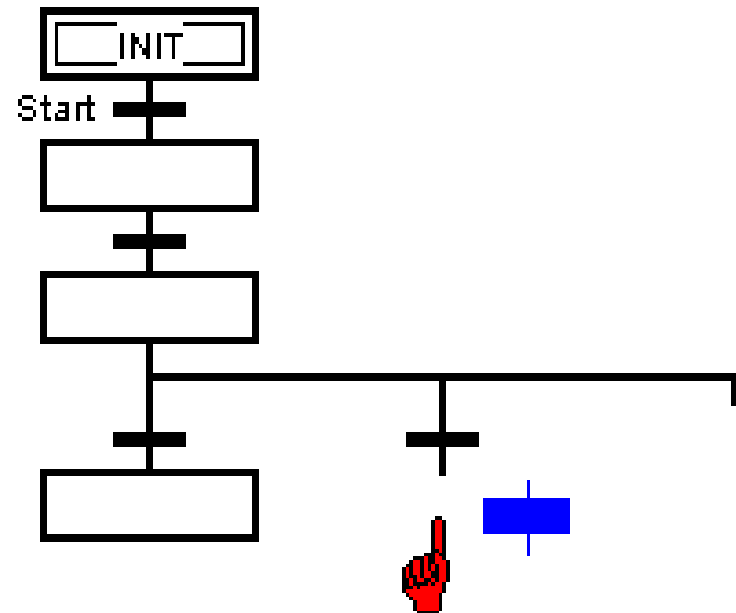
- ☑ Modeling, analysis and programming of control software
- ☑ Analysis of parallel processes

## Operations engineering:

- ☑ Workflow
- ☑ Modeling and analysis of production lines
- ☑ Representation of processes

# Topics

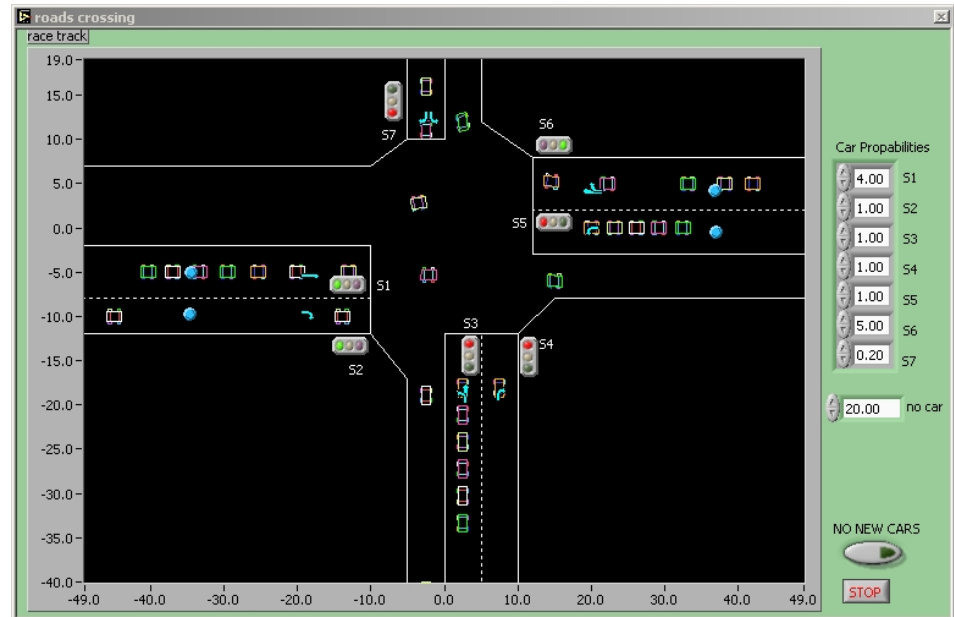
- ☑ How to draw a sequential function chart?



# Topics

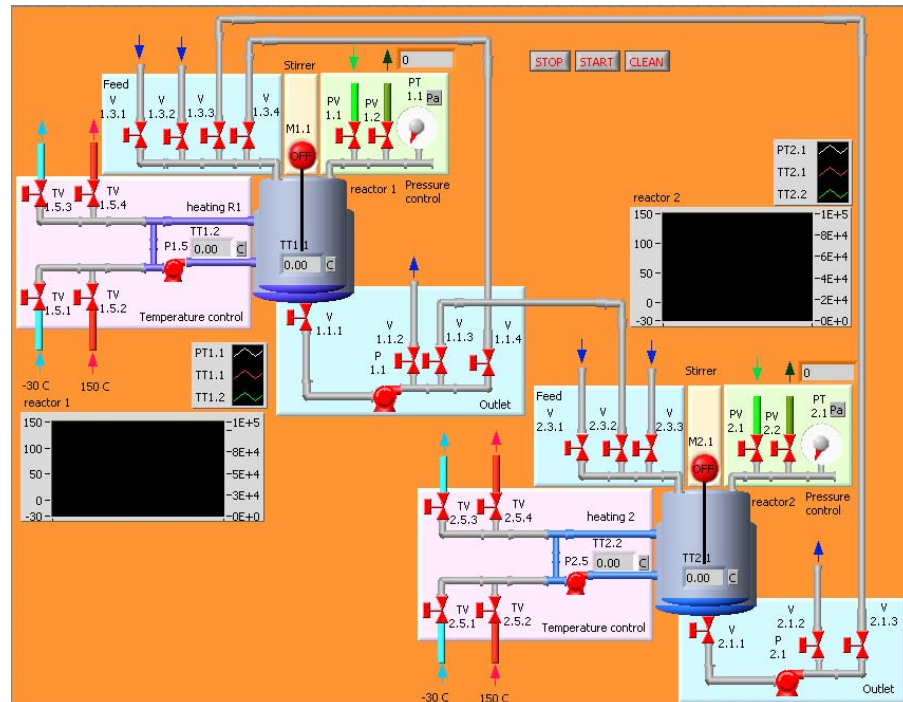
☑ Problems to solve:

Control of traffic  
lights



# Topics

- ☑ Problems to solve:  
control of a  
pharmaceutical batch  
plant



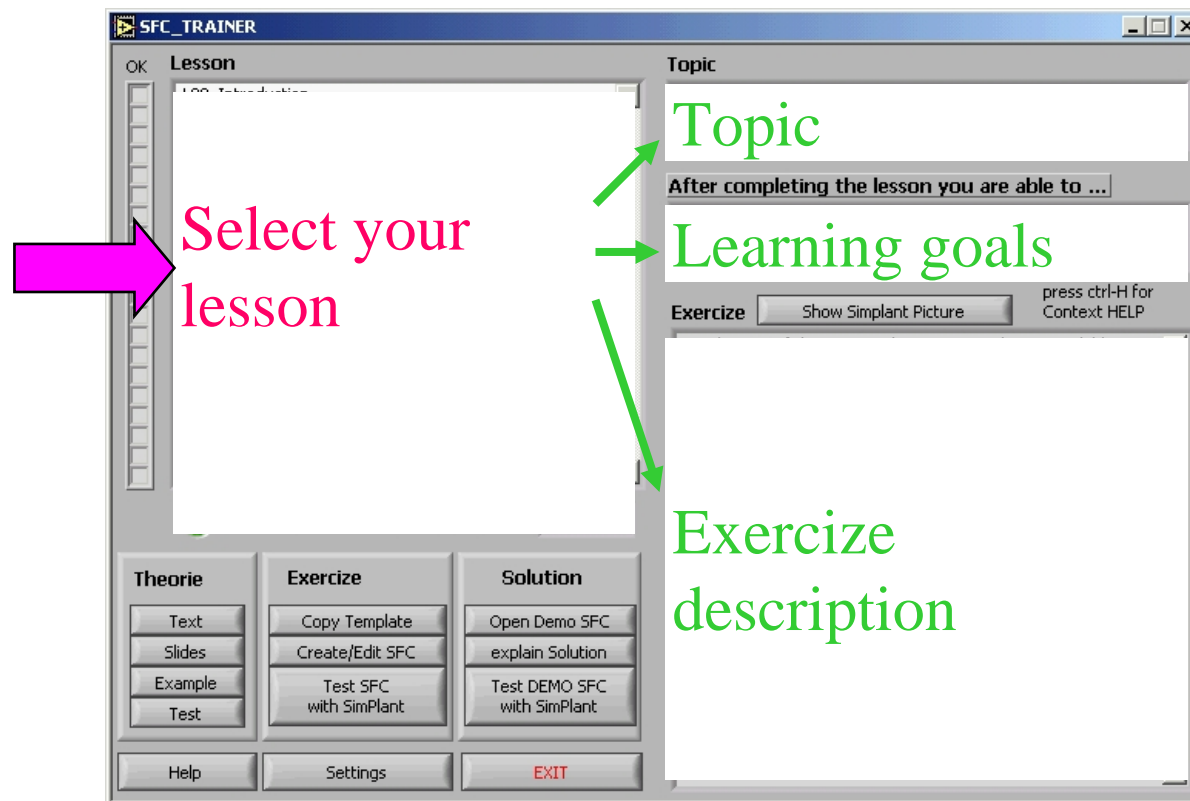


# SFC-Trainer

## a interactive e-learning-Tool

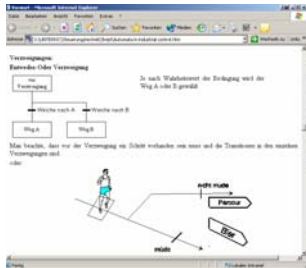
- ☑ Lesson: learning by doing
- ☑ Interactiv:
  - draw and simulate your own SFC
  - test your SFC with a real-life simulation plant
  - compare your solution with a proposed solution

# How to use the SFC-Trainer

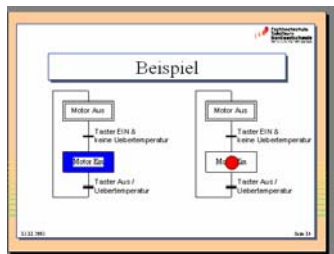


# 1. Theory

## Lecture notes



## Animated slides



**Example**

**SFC\_TRAINER**

**Lesson**

- L00\_Introduction
- L01\_Step\_Actions\_and\_Transitions
- L02\_Switching\_Rules**
- L03\_Easy\_Plant
- L04\_Either\_OR\_Fork\_and\_Join
- L05\_Modelling\_User\_Interaction
- L06\_Plant\_with\_External\_Interactions
- L07\_Simple\_Synchronisation
- L08\_Parallel\_Processes
- L09\_Large\_Plant
- L10\_Traffic\_Light\_Control
- L11\_Matrix\_Representation\_Easy
- L12\_Matrix\_Representation\_Adv
- L13\_Reachability\_Easy
- L14\_Reachability\_Adv
- L15\_Safe\_SFC\_Easy
- L16\_recipe
- L17\_recipe\_control\_Plant

**Topic**

Simple SFC-switching rules

**After completing the lesson you are able to ...**

- explain the major difference between an SFC and a Flowchart
- explain, under which circumstances the actions in a step are active or executed

**Exercise**  press ctrl-H for Context HELP

Use the SFC of the previous lesson. A template is available. Use 'Copy template' to create a copy in your working directory. Add an additional step and a transition at the end of the SFC, just before the final jump. In the additional step increase the counter once when the new step gets marked. Use the 'Go On 3'-Transition to leave the step.

The control which sets the 'Go On 3' condition to TRUE has a different switching action: when the button is pressed, it remains in this state, until it is released.

Test the behaviour of the SFC and observe the counter value in the simulation user interface. Does it behave as expected?

Now, let the 'Go On 3' condition be true, i.e. let the corresponding button be pressed and test the SFC once again. Back on the initial step, why doesn't the counter have the same value as before?

Theory	Exercise	Solution
<input type="button" value="Text"/>	<input type="button" value="Copy Template"/>	<input type="button" value="Open Demo SFC"/>
<input type="button" value="Slides"/>	<input type="button" value="Create/Edit SFC"/>	<input type="button" value="explain Solution"/>
<input type="button" value="Example"/>	<input type="button" value="Test SFC with SimPlant"/>	<input type="button" value="Test DEMO SFC with SimPlant"/>
<input type="button" value="Test"/>		
<input type="button" value="Help"/>	<input type="button" value="Settings"/>	<input type="button" value="EXIT"/>

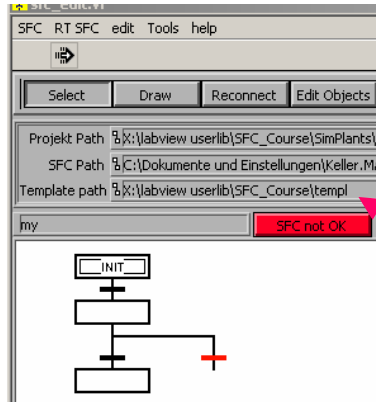
**Self check**

# 2. Exercise

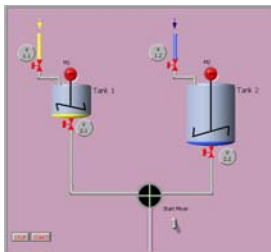
Preview of the simulation plant

Copy template

Draw and configure SFC



Test SFC



Topic

Simple SFC-switching rules

After completing the lesson you are able to ...

- explain the major difference between an SFC and a Flowchart
- explain under which circumstances the actions in a step are active or executed

Exercise

Show Simplant Picture

press ctrl-H for Context HELP

Use the SFC of the previous lesson. A template is available. Use 'Copy template' to create a copy in your working directory. Add an additional step and a transition at the end of the SFC, just before the final jump. In the additional step increase the counter once when the new step gets marked. Use the 'Go On 3'-Transition to leave the step.

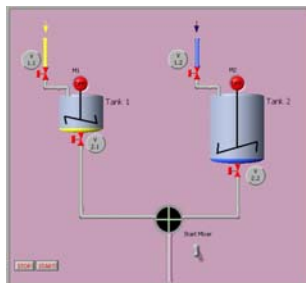
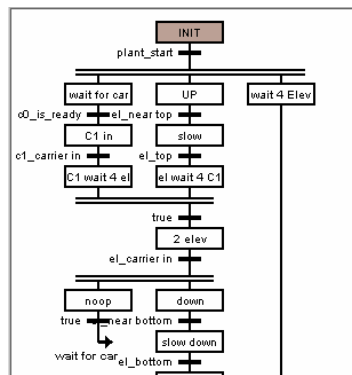
The control which sets the 'Go On 3' condition to TRUE has a different switching action: when the button is pressed, it remains in this state, until it is released.

Test the behaviour of the SFC and observe the counter value in the simulation user interface. Does it behave as expected?

Now, let the 'Go On 3' condition be true, i.e. let the corresponding button be pressed and test the SFC once again. Back on the initial step, why doesn't the counter have the same value as before?

Theorie	Exercise	Solution
Text	Copy Template	Open Demo SFC
Slides	Create/Edit SFC	explain Solution
Example	Test SFC with SimPlant	Test DEMO SFC with SimPlant
Test		
Help	Settings	EXIT

# 3. Proposed Solution



OK

Lesson

L00\_Introduction

L01\_Step\_Actions\_and\_Transitions

L02\_Switching\_Rules

L03\_Easy\_Plant

L04\_Either\_OR\_Fork\_and\_Join

L05\_Modelling\_User\_Interaction

L06\_Plant\_with\_External\_Interactions

L07\_Simple\_Synchronisation

L08\_Parallel\_Processes

L09\_Large\_Plant

L10\_Traffic\_Light\_Control

L11\_Matrix\_Representation\_Easy

L12\_Matrix\_Representation\_Adv

L13\_Reachability\_Easy

L14\_Reachability\_Adv

L15\_Safe\_SFC\_Easy

L16\_recipe

L17\_recipe\_control\_Plant

mandatory

Time required [min] 30

Theorie

Exercise

Solution

Text

Slides

Example

Test

Copy Template

Create/Edit SFC

Test SFC with SimPlant

Open Demo SFC

explain Solution

Test DEMO SFC with SimPlant

Help

Settings

EXIT

Topic

Simple SFC-switching rules

After completing the lesson you are able to ...

- explain the major difference between an SFC and a Flowchart

- explain, under which circumstances the actions in a step are active or executed

Exercise

Show Simplant Picture

press ctrl-H for Context HELP

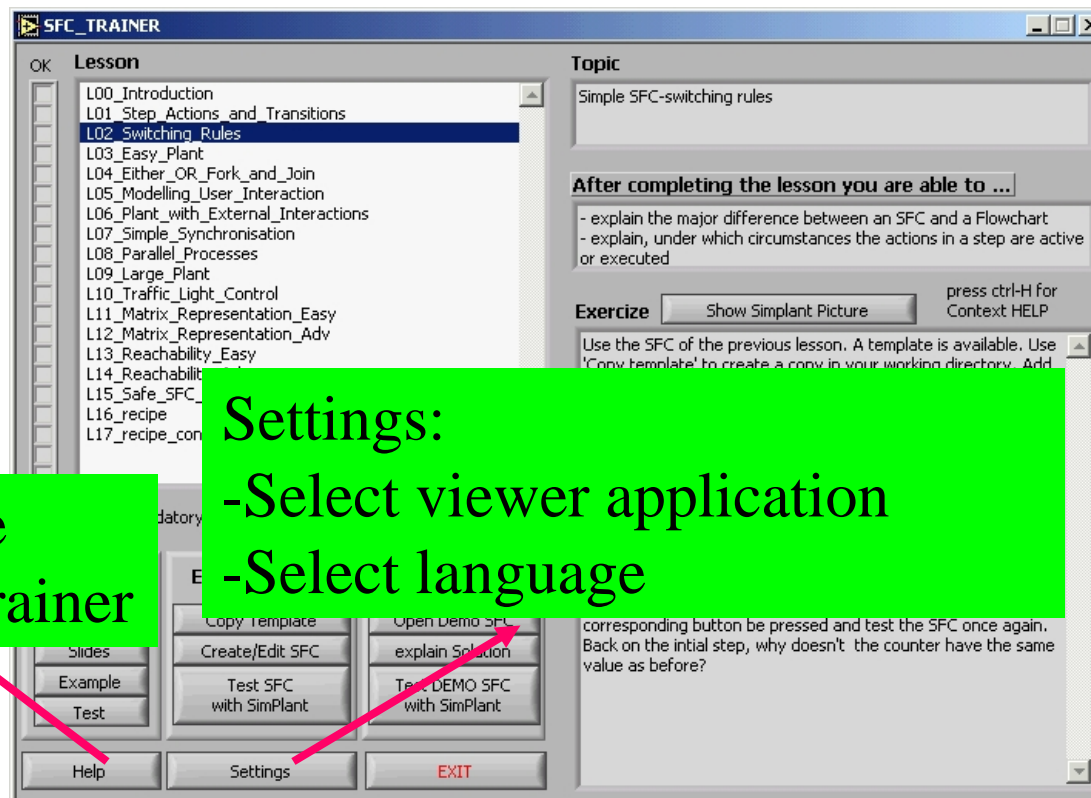
Use the SFC of the previous lesson. A template is available. Use 'Copy template' to create a copy in your working directory. Add an additional step and a transition at the end of the SFC, just before the final jump. In the additional step increase the counter once when the new step gets marked. Use the 'Go On 3'-Transition to leave the step.

The control which sets the 'Go On 3' condition to TRUE has a different switching action: when the button is pressed, it remains in this state, until it is released.

Test the behaviour of the SFC and observe the counter value in the simulation user interface. Does it behave as expected?

Now, let the 'Go On 3' condition be true, i.e. let the corresponding button be pressed and test the SFC once again. Back on the initial step, why doesn't the counter have the same value as before?

# How to use the SFC-Trainer



How to use  
the SFC-Trainer

Settings:  
-Select viewer application  
-Select language

# Let's start!

