# Preface, Contents

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basics of Sequential Control Systems</td>
<td>1</td>
</tr>
<tr>
<td>Working with the SFC Editor</td>
<td>2</td>
</tr>
<tr>
<td>Sequential Control System on the AS</td>
<td>3</td>
</tr>
<tr>
<td>Test and Startup</td>
<td>4</td>
</tr>
<tr>
<td>Documentation</td>
<td>5</td>
</tr>
</tbody>
</table>

## Appendix

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technical Specifications</td>
<td>A</td>
</tr>
<tr>
<td>Abbreviations</td>
<td>B</td>
</tr>
<tr>
<td>Glossary</td>
<td></td>
</tr>
<tr>
<td>Index</td>
<td></td>
</tr>
</tbody>
</table>

---

Edition 01/2003

ASE00177375-01
Safety Guidelines

This manual contains notices intended to ensure personal safety, as well as to protect the products and connected equipment against damage. These notices are highlighted by the symbols shown below and graded according to severity by the following texts:

---

**Danger**

indicates that death, severe personal injury or substantial property damage will result if proper precautions are not taken.

---

**Warning**

indicates that death, severe personal injury or substantial property damage can result if proper precautions are not taken.

---

**Caution**

indicates that minor personal injury can result if proper precautions are not taken.

---

**Notice**

draws your attention to particularly important information on the product, handling the product, or to a particular part of the documentation.

---

Qualified Personnel

Only qualified personnel should be allowed to install and work on this equipment. Qualified persons are defined as persons who are authorized to commission, to ground and to tag circuits, equipment, and systems in accordance with established safety practices and standards.

Correct Usage

Note the following:

---

**Warning**

This device and its components may only be used for the applications described in the catalog or the technical description, and only in connection with devices or components from other manufacturers which have been approved or recommended by Siemens.

This product can only function correctly and safely if it is transported, stored, set up, and installed correctly, and operated and maintained as recommended.

---

Trademarks

SIMATIC®, SIMATIC HMI® and SIMATIC NET® are registered trademarks of SIEMENS AG.

Third parties using for their own purposes any other names in this document which refer to trademarks might infringe upon the rights of the trademark owners.

---

Copyright © Siemens AG 2003 All rights reserved

The reproduction, transmission or use of this document or its contents is not permitted without express written authority. Offenders will be liable for damages. All rights, including rights created by patent grant or registration of a utility model or design, are reserved.

---

Disclaimer of Liability

We have checked the contents of this manual for agreement with the hardware and software described. Since deviations cannot be precluded entirely, we cannot guarantee full agreement. However, the data in this manual are reviewed regularly and any necessary corrections included in subsequent editions. Suggestions for improvement are welcomed.
Preface

Purpose of the Manual
This manual supports you when you create sequential control systems and parameter controls. It provides you with an overview of the following:

• The basics of sequential control systems
• Working with the SFC Editor
• Sequential control systems on the programmable controller
• Putting sequential control systems into operation and monitoring and testing them
• Documenting SFC charts

You will find a detailed description of the software and procedures in the SFC online help.

This manual "SFC for S7" provides you with the information you require to use the SFC configuration tool in conjunction with CPUs in SIMATIC S7 programmable controllers (PLCs). If you use other target systems (for example, SIMADYN D), please read the additional documentation for this target system.

How Sections for Specific Systems are Indicated
If sections, paragraphs or even individual sentences in this S7 manual relate solely to S7 users, this is indicated by [S7]. This means that the information is relevant only to S7 or is different in other systems. In this case, if you use a different PLC, you will find the information you require in the manual for your specific system.

If the [S7] label is in a title, the entire section applies only to S7; if the label is at the start of a paragraph, the paragraph is solely relevant to S7. In lists, the [S7] label applies only to the particular list.

Audience
This manual is intended for personnel involved in configuring, commissioning, and service.
Basic experience of working with PCs and Windows is assumed.

Validity
This manual is valid for the SFC software version V6.0 or higher.
You will find the latest information that could no longer be included in this manual along with instructions on installation in the README.TXT file accompanying the product.
Standard
The SFC software is based on the international standard DIN EN 61131-3 (IEC 1131-3) for programming languages for programmable logic controllers.

Further Support
If you have questions about using the products described in the manual and you cannot find the answers here, please contact your local Siemens representative.
http://www.siemens.com/automation/partner

Training Center
To help you to become familiar with working with S7 PLCs, we offer a range of courses. Please contact your regional training center or the central training center in D 90327 Nuremberg.
Phone: +49 (911) 895-3200.
Internet: http://www.sitrain.com
### A&D Technical Support

Open round the clock, worldwide:

![Map with support locations](image)

<table>
<thead>
<tr>
<th>Worldwide (Nuremberg) Technical Support</th>
<th>United States (Johnson City) Technical Support and Authorization</th>
<th>Asia / Australia (Peking) Technical Support and Authorization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local time: 0:00 to 24:00 / 365 days</td>
<td>Local time: Mo.-Fr. 8:00 to 17:00</td>
<td>Local time: Mo.-Fr. 8:30 to 17:30</td>
</tr>
<tr>
<td>Phone: +49 (0) 180 5050-222</td>
<td>Phone: +1 (0) 423 262 2522</td>
<td>Phone: +86 10 64 75 75 75</td>
</tr>
<tr>
<td>Fax: +49 (0) 180 5050-223</td>
<td>Fax: +1 (0) 423 262 2289</td>
<td>Fax: +86 10 64 74 74 74</td>
</tr>
<tr>
<td>E-mail: <a href="mailto:adsupport@siemens.com">adsupport@siemens.com</a></td>
<td>E-mail: <a href="mailto:simatic.hotline@sea.siemens.com">simatic.hotline@sea.siemens.com</a></td>
<td>E-mail: <a href="mailto:adsupport.asia@siemens.com">adsupport.asia@siemens.com</a></td>
</tr>
<tr>
<td>GMT: +1:00</td>
<td>GMT: -5:00</td>
<td>GMT: +8:00</td>
</tr>
</tbody>
</table>

Technical Support and Authorization speak German and English.
Service & Support on the Internet

In addition to our documentation services, you can also make use of all our knowledge on the Internet.

http://www.siemens.com/automation/service&support

Here, you will find:

- The Newsletter that keeps you constantly up to date with the latest information on the products you use.
- The documents you need using the search functions in Service & Support.
- A Forum in which users and specialists exchange information and experience.
- Your local contact for Automation & Drives in our contacts database.

Information on local service, repairs, and spare parts. Much more under the heading “Services”.

# Contents

1 Basics of Sequential Control Systems ................................................................. 1-1
   1.1 General Information on Sequential Control Systems........................................ 1-1
   1.2 SFC in the STEP 7 Environment .................................................................. 1-2
   1.2.1 SFC and the Plant Hierarchy [S7] ................................................................. 1-3
   1.3 Steps in Configuration ................................................................................ 1-3
   1.4 Creating the Project Structure .................................................................. 1-4
   1.5 Creating Sequential Control Systems ...................................................... 1-5
   1.6 SFC Type, SFC Instance, and External View of the Chart .................... 1-7
   1.7 SFC Elements ......................................................................................... 1-9
      1.7.1 What is a Sequencer? ............................................................................. 1-9
      1.7.2 What are Sequence Path Elements? ..................................................... 1-10
      1.7.3 What is a Step? ...................................................................................... 1-10
      1.7.4 What is a Transition? .............................................................................. 1-11
      1.7.5 What is a Text? ...................................................................................... 1-11
      1.7.6 What is a Sequence and What is a Sequencer? .................................. 1-12
      1.7.7 What is a Simultaneous Sequence? ...................................................... 1-12
      1.7.8 What is an Alternative Sequence? ......................................................... 1-13
      1.7.9 What is a Loop? ...................................................................................... 1-13
      1.7.10 What is a Jump? ................................................................................... 1-14

2 Working with the SFC Editor ........................................................................... 2-1
   2.1 Working with Charts, Types, and Instances ............................................. 2-1
      2.1.1 How to Create a Chart .......................................................................... 2-1
      2.1.2 [S7] Creating an SFC Type .................................................................. 2-2
      2.1.3 [S7] Creating an SFC Instance ............................................................... 2-2
      2.1.4 Opening a Chart or Type .................................................................... 2-3
      2.1.5 [S7] Opening an SFC Instance ............................................................... 2-3
      2.1.6 Copying Charts .................................................................................... 2-3
      2.1.7 [S7] Copying and Moving SFC Types ................................................ 2-3
      2.1.8 [S7] Copying and Moving SFC Instances .......................................... 2-4
      2.1.9 Deleting Charts and Types ................................................................. 2-4
      2.1.10 [S7] Deleting SFC Instances ............................................................... 2-4
      2.1.11 Representation and Interconnection of the External View .............. 2-5
   2.2 Properties of Charts and Types ................................................................ 2-6
      2.2.1 Adapting Chart Properties ................................................................. 2-6
      2.2.2 [S7] Adapting Type Properties .............................................................. 2-6
      2.2.3 [S7] Adapting Instance Properties ...................................................... 2-7
   2.3 The Run-Time Properties ........................................................................ 2-8
      2.3.1 Run Sequence ...................................................................................... 2-8
      2.3.2 Run-Time Groups ................................................................................ 2-9
   2.4 Configuring Sequential Control Systems ............................................... 2-11
      2.4.1 Creating the Sequencer Topology ...................................................... 2-12
      2.4.2 Configuring Several Sequencers ....................................................... 2-13
2.5 Creating SFC Elements .......................................................................................... 2-15
2.5.1 Creating a Sequence ......................................................................................... 2-15
2.5.2 Creating and Extending a Simultaneous Sequence ........................................ 2-16
2.5.3 Creating and Extending an Alternative Sequence ........................................... 2-17
2.5.4 Creating a Loop ............................................................................................... 2-18
2.5.5 Creating a Jump .............................................................................................. 2-19
2.5.6 Creating and Editing Text Objects ................................................................... 2-20
2.6 Editing SFC Elements .......................................................................................... 2-21
2.7 Editing in the Object Properties Dialog .............................................................. 2-22
2.7.1 Editing Object Properties: Step ......................................................................... 2-22
2.7.2 Edit Object Properties: Transition .................................................................... 2-25
2.8 [S7] Creating an SFC Type .................................................................................. 2-28
2.9 [S7] Configuring in the Characteristics Dialog .................................................... 2-30
2.10 [S7] The I/O Groups .......................................................................................... 2-32
2.11 [S7] Configuring Messages ................................................................................ 2-33
2.12 Compiling and Downloading .............................................................................. 2-34
2.12.1 Compile .......................................................................................................... 2-35
2.12.2 [S7] Downloading ......................................................................................... 2-37
2.13 [S7] Parameter-Controlled Operation ............................................................... 2-39

3 Sequential Control System on the AS ................................................................. 3-1
3.1 General................................................................................................................. 3-1
3.2 How the SFC Runs ............................................................................................. 3-2
3.2.1 [S7] Mode ................................................................................................ ...... 3-2
3.2.2 [S7] Step Control Mode .................................................................................. 3-2
3.2.3 Chart Execution Options .............................................................................. 3-3
3.3 How the Sequential Control System Behaves during Operation ....................... 3-4
3.3.1 [S7] The Operating States .............................................................................. 3-5
3.3.2 Operating State Logic for SFC (SFC-OSL) ...................................................... 3-6
3.3.3 Operating State Logic for Sequencers (sequencer OSL) .................................. 3-8
3.3.4 Commands ..................................................................................................... 3-10
3.3.5 Sequencer Execution ..................................................................................... 3-11
3.3.6 Starting an SFC (chart or instance) ............................................................... 3-14
3.3.7 Execution of a Sequential Control System ................................................... 3-15

4 Test and Startup ...................................................................................................... 4-1
4.1 Activating the Test ............................................................................................. 4-1
4.2 Display in the Test Mode .................................................................................... 4-3
4.3 Operator Control and Monitoring of the Sequential Control System ............... 4-6
4.3.1 The Object Properties of a Step during Testing ............................................ 4-6
4.3.2 The Object Properties of a Transition during Testing .................................... 4-9

5 Documentation ....................................................................................................... 5-1
5.1 Printing an SFC .................................................................................................. 5-1
5.2 Defining Footers ............................................................................................... 5-2
5.3 Chart Reference Data ....................................................................................... 5-3
5.4 Logs .................................................................................................................... 5-3
Contents

A  Technical Specifications                      A-1
   A.1  Technical Specifications  A-1

B  Abbreviations                                B-1

Glossary

Index
1 Basics of Sequential Control Systems

Introduction
This chapter explains the basics of sequential control systems. Here, you will learn what a sequential control system is and what it is used for. You will get to know the terminology and elements of SFC and the rules governing the topology of sequencers.

1.1 General Information on Sequential Control Systems

What is SFC?
An SFC chart, an SFC type, or an SFC instance is a sequential control system. The SFC editor is a tool with which you can create a sequential control system integrating several sequencers that can be started separately.
In these descriptions, the term “SFC” is used to mean a chart, a type, an instance, or the SFC editor depending on the context.
An SFC chart is assigned uniquely to a CPU and is also executed completely on this CPU. Instances of a type can exist on several CPUs.

What is a sequential control system?
A sequential control system is a controller partitioned to ensure step-by-step execution with control passing from one state to the next state dependent on conditions.
Sequential control systems can be used, for example, to describe the manufacture of products as event-controlled processes (recipes).
With a sequential control system, functions from basic control (typically created with CFC) are controlled by mode and state changes and executed selectively.
Where are sequential control systems used?

The typical applications of sequential control systems involve processes and plants with discontinuous characteristics. Sequential control systems can, nevertheless, also be used for continuous processes and plants, for example for approach and withdrawal movements, operating point changes, and state changes due to faults etc.

[S7] Such systems can be used at various levels of a process or plant:

- Device control level (open valve, start motor .....)
- Group control level (proportioning, stirring, heating, filling .....)
- Unit level (tank, mixer, scales, reactor .....)
- Plant level (synchronization of units and common resources, for example routing)

1.2 SFC in the STEP 7 Environment

The SIMATIC Manager is used for all ASs as the central database to coordinate the tools and objects. It manages the tools and data and is used to create and modify a project structure (CPU, CFC/SFC charts) and to start the SFC editor.

Figure 1-1 shows how SFC fits into the STEP 7 and PCS 7 environment:

Legend: PH (Plant Hierarchy), IEA (Import/Export Assistant) and PO (Process Object view) are components of the Process Control System (PCS 7) and extend the SIMATIC Manager. WinCC is the operator control and monitoring system in PCS 7.
1.2.1 SFC and the Plant Hierarchy [S7]

The plant hierarchy (PH) allows charts to be arranged and managed not only from the point of view of running them on a CPU, but also according to technological or plant criteria (for example an SFC chart for device control, group control, or unit control).

If the SFC chart was assigned to a plant hierarchy folder, the path of the plant hierarchy is added to the chart name. You can use the naming scheme of your plant as the criteria for arranging charts in the project.

For more detailed information on the plant hierarchy, refer to the online help of PH.

1.3 Steps in Configuration

How the Editor Works

Using the SFC editor, you create your sequential control system using graphic tools. The elements of the SFC are positioned according to fixed rules in the sequencer. You do not need to be aware of details such as algorithms or the assignment of machine resources but can concentrate solely on the technological aspects of your configuration.

After creating the sequencer topology, you move on to configure the object properties where you formulate the properties of the sequencers, the individual steps and transitions; in other words, you configure the actions and conditions.

After configuration, you compile the executable machine code with SFC, download it to the AS, and test it with the SFC test functions.
1.4 Creating the Project Structure

Three alternative methods of creating a project structure are shown below:

1. Start the SIMATIC Manager and then the PCS 7 Wizard (if this is not already started automatically) by selecting “File > ‘New Project’ Wizard...”. Work through the dialogs displayed by the Wizard. In the dialog “Which objects will be used in the project?”, the option “SFC chart” may be set as the default in the “AS objects” box.

The wizard creates a single project or multiproject in the plant view and in the component view. Apart from the actual project, a multiproject also includes a master data library. During the creation of the project, certain defaults are set that you can modify later if necessary.

2. The STEP 7 Wizard was started in the SIMATIC Manager. (The setting PCS 7 or STEP 7 is made with "Options > Customize..." in the "Wizards" tab of the dialog.)

Using this wizard, you create a STEP 7 project; in other words the SIMATIC Station, the CPU, the S7 program and the block folder.

- Select the menu command “Insert > S7 Software > Chart Folder”. Insert a chart in the chart folder.
- Select the chart folder and then the menu command “Insert > S7 Software > SFC”.

3. Create a new project in the SIMATIC Manager using menu commands.

- Select “File > New...”, enter the name of the project in the dialog box and confirm with "OK".
- Insert an S7 program in the project (Insert > Program > S7 Program).
- Insert a chart folder in the S7 program (Insert > S7 Software > Chart Folder).
- Insert a chart in the chart folder (Insert > S7 Software > SFC).

The recommended method is to use the PCS 7 Wizard.
1.5 Creating Sequential Control Systems

The steps outlined below represent the most efficient procedure for configuring sequential control systems (SFC charts) for your AS: The steps listed below, relate to the configuration of an SFC chart. The procedure for configuring an SFC type is basically the same (see also Section 2.8, Creating an SFC Type):

1. Create the project structure
   Create a project structure in the SIMATIC Manager in which you can configure CFC/SFC charts (see Section 1.4).

2. Specify the chart properties
   When you specify the chart properties, you can change the chart name and add a comment. (For example, the technological function.) The plant properties also include the operating parameters with default values that you can modify (see Section 3.3).

3. Adapt operating parameters and run-time properties:
   By setting the operating parameters, you specify the behavior of the sequential control system, such as the mode (manual, auto), step control mode (T, C, T and C...) and other chart execution options (cyclic operation, time monitoring, autostart, ...).
   The run-time properties of an SFC chart determine how the SFC chart is included in the execution of the entire structure on the AS (in the window of the CFC run-time editor).

4. Configure the sequencer properties
   For each sequencer, you configure the start condition and, as an option, the action for preprocessing and postprocessing.

5. Create the topology of the sequential control system:
   You configure sequential control systems with the SFC charts by inserting the steps and transitions, and when necessary other structure elements, for one or more sequencers (see Section 2.5).

6. Configure steps (in the Object Properties dialog):
   In the steps, you formulate actions. The actions contain statements with which the values of block inputs and of shared addresses can be changed or run-time groups or other SFC charts can be activated and deactivated (see Section 2.7.1).

7. Configure transitions (in the Object Properties dialog):
   In the transitions, you formulate conditions. The conditions read the values of block I/Os, of shared addresses or the state (active/inactive) of run-time groups or other SFC charts. When the conditions of a logic operation are met, the following step becomes active and its actions are executed (see Section 2.7.2).

8. Compile and download:
   During compilation, the CFC and SFC charts of the active chart folder are converted to an executable user program (compile entire program).
   After compilation, you can download the program to the AS (CPU) (see Section 2.12).
9. Test:
   After compiling and downloading, you can test the program in the process mode or in the laboratory mode. Using the SFC test functions, you can run the sequential control system in various operating modes and step control modes and monitor and modify the values of addresses on the CPU. You can also influence the most important operating modes (STOP, clear/reset, RUN, ...) on the CPU.
1.6 SFC Type, SFC Instance, and External View of the Chart

[S7] The Type/Instance Concept

With SFC V6.0, the concept of type and instance has been introduced. This makes it possible to create sequential control system types that create SFC instances when placed in a CFC chart.

[S7] What is an SFC Type?

In SFC, there is not only the object type "SFC chart" but also the object type "SFC type". The SFC type allows the definition of sequential control systems including an interface. The sequential logic of the SFC type is based solely on the interface I/Os of the SFC type; in other words in contrast to an SFC chart, an SFC type cannot access every process signal.

The SFC type cannot run alone. Like a function block type, an SFC type must be placed in a CFC chart to obtain a runnable object, in this case, an SFC instance.

The SFC type and the SFC instances are compiled when you compile the program. To run an SFC instance, both the SFC type and the SFC instance are loaded on the AS.

For an SFC type, you can configure seven messages that must be acknowledged and five that do not require acknowledgment. The SFC type itself requires the remaining available messages (one per message type and 10 notify messages for BATCH).

[S7] What is an SFC Instance?

An SFC instance is derived from an SFC type. The SFC type is first inserted into a CFC chart in the same way as a function block type in CFC. The SFC instances are therefore always assigned to a CFC chart and are addressed using the chart. SFC instances are displayed like CFC instances; in other words, their interface is visible in the CFC chart.

The I/Os of SFC instances can have parameter values assigned to them and they can be interconnected.

SFC instances are not displayed in the SIMATIC Manager since they can only be addressed here via the CFC chart. With the assignment of the CFC chart to the plant hierarchy, the SFC instances contained are also indirectly assigned to the plant hierarchy.
What is an External View?

The SFC chart has a standard interface (derived from the interface of the run-time system). This interface is represented as the graphic "external view" of the chart.

With the menu command "View > External View", you open CFC with a window of the external view of the SFC chart.

The external view shows the SFC chart like a block. Using CFC interconnections, the chart can be controlled via the I/Os. To distinguish it from CFC blocks and hierarchical charts, the external view has the "SFC chart" icon in the header. The block name is the same as the SFC chart name and cannot be modified.

You cannot place any other objects in the external view (for example, blocks). The external view is interconnected via the sheet bar as usual for CFC blocks.

Figure 1-2: External View of the SFC Chart
1.7 SFC Elements

1.7.1 What is a Sequencer?

With sequencers, status-dependent and event-driven execution is possible in SFC. An SFC chart can include up to 8 sequencers and an SFC type up to 32 sequencers that can be controlled by defining different start conditions.

The working window in SFC displays one sequencer. You can change to a different sequencer easily using the tabs at the bottom edge of the window.

When you create a new chart or type, a sequencer with the name “RUN” and the start condition RUN=TRUE is created (Note: This corresponds to a V5 chart). The start conditions are formulated in the same way as the transition conditions (see Section 1.7.4, “What is a Transition?”). An empty start condition, in contrast to the transition, is evaluated as FALSE; in other words, the sequencer is never executed.

Each sequencer contains not only this start condition but also the “Priority” attribute with which a start order can be specified when there are simultaneously satisfied conditions for several sequencers (“Start Condition” tab in the “Sequencer Properties” dialog). If the priority is the same and the condition is satisfied, the position of the tab decides the order in which they are processed on the CPU (analogous to the alternative sequence, see Section 3.3.7, Processing an Alternative Sequence).

A cyclic action can also be configured for a sequencer. The cyclic action consists of a part that is executed before the cyclic sequencer processing, the preprocessing and a part that is executed after the cyclic sequencer processing, the postprocessing.
1.7.2 What are Sequence Path Elements?

An SFC chart consists of 1 to 8 and an SFC type of 1 to 32 sequencers each with a sequence of sequence path elements. These elements include the following:

- Step
- Transition

And below a sequence (can be freely positioned):

- Text

The remaining elements are structures made up of different elements:

- Sequence
- Simultaneous sequence
- Alternative sequence
- Loop
- Jump

Identifying "Steps" and "Transitions"

The basic elements, step and transition, have a name that is unique within the sequencer. When the editor creates one of these elements, it assigns a consecutive number that you can modify and change to a name with up to 16 characters. This name must not consist exclusively of numbers.

You can use the optional comment to add comments about the functionality of the element. A comment can take up several lines and consists of up to 80 characters although only 16 characters are displayed to the right of the chart element.

If you position the mouse pointer on the step or the comment, the name with a maximum of 16 characters will be displayed and the comment with up to 50 characters as brief information.

Screen Display

All the elements of an SFC, including the links are displayed in white with black print in the unselected and unedited state.
Selected elements are displayed in blue.
Edited steps or transitions (whose object properties have been changed) are displayed in gray with black print.

Note: The colors explained here are the default settings, some of which can be modified (refer to the SFC online help).
1.7.3  What is a Step?

The step is a control instance for processing the actions associated with it on the AS. Per step, you can configure up to three actions (initialization, processing, termination).

[S7] An action is a collection of statements and is formulated as follows:

- Assignments for assigning parameters to CFC blocks or shared resources, for example:
  
  ```
  Settemp := 100
  XYZ.pump.on := TRUE
  ```

- Activation or deactivation of an SFC or a run-time group, for example:
  
  ```
  SFC_1.INTOFF := TRUE
  ABL_1.EN := FALSE
  ```

Note: Make sure that you keep to the specific rules for address assignments on other target systems.

In an SFC, you can use an initial step, a variable number of normal steps (maximum 253) and a final step. When a sequencer is created, an initial step, a final step, and one transition are created automatically. These three elements form the initial state of an SFC that you can edit by adding further chart elements.

The initial step is activated immediately without querying conditions as soon as the chart is started and the actions associated with it are executed according to the state of the successor transition. The final step does not have a successor transition; all actions are performed exactly once.

You can neither create nor delete the initial or final steps. This makes sure that there is always one initial and one final step in a sequencer.

All other steps apart from the initial and final steps are normal steps.
1.7.4 What is a Transition?

A transition contains the condition with which a sequential control system passes control from one step to the next. Several conditions can be logically combined using Boolean operators. The result of the logic operation decides whether control is passed to the next step.

![Figure 1-4: Icon of the Transition](image)

[S7] The result of a transition condition is obtained from a Boolean expression formed by logic operations on shared addresses, CFC block I/Os, run-time group states, and the SFC state.

During compilation, an empty transition is given the default value TRUE. This default is necessary because while the formulation of a condition is optional, a defined value is required on the AS to allow control to be passed on.

If several transitions become valid at the same time (in alternative sequences, in a loop or in jumps) the system automatically assigns priority from left to right in descending order.

1.7.5 What is a Text?

With the “text” chart element, you can insert any static texts (free texts) in a chart as required.

The text object is a box with a character string with one or more lines. During the analysis phase, this allows you to insert descriptive texts into the SFC that can later be replaced by automation functions.

These free texts are not embedded in the topological sequencer structure and are therefore not repositioned if you change the topology but remain where they are.
1.7.6 What is a Sequence and What is a Sequencer?

A sequence is a path made up of steps and transitions that can be created with a selectable length and inserted in the sequential control system.

A self-contained sequence within a sequential control system forms a sequence path, for example between the divergence and convergence of a parallel or alternative sequence. In the SFC a full sequencer can also be called a sequence path, this runs from the initial step to the final step (see Figure 1-5).

Figure 1-5: Sequences
1.7.7 **What is a Simultaneous Sequence?**

If the control diverges along two or more sequence paths that should be executed at the same time, simultaneous sequences are used. A simultaneous sequence consists of at least two sequence paths that are executed at the same time. A simultaneous sequence is always preceded by a transition (or an alternative sequence). The simultaneous sequence paths end in a simultaneous convergence that is always followed by a transition (or alternative sequences). The successor transition executes only when all actions of the steps at the end of every sequence path have been executed (apart from the “termination” action) and the condition for passing on control is satisfied (synchronization).

![Simultaneous Sequence Diagram](image)

**Figure 1-6: Example a Simultaneous Sequence with Four Sequence Paths**
1.7.8 What is an Alternative Sequence?

If control diverges into two or more sequence paths of which one and only one should be executed, alternative sequences are used.

An alternative sequence consists of at least two sequence paths, of which only one will be executed dependent on the state of the first transition in the individual sequence paths. This means that the path selected is the path whose transition is satisfied first. If more than one transition is true simultaneously, the sequence path furthest left with a true transition is executed.

Alternative sequences must be preceded and followed by a step (or simultaneous sequence).

![Figure 1-7: Example of an Alternative Sequence with Four Sequence Paths](image-url)
1.7.9 What is a Loop?

If you want a section of the sequencer to be repeated depending on a transition, you use a loop.

A loop consists of a sequence and a return path with a transition that encloses the sequence (see Figure 1-7). The start of the loop must be immediately following a step and the return path must converge again immediately before a step.

![DIAGRAM: Example of a Loop](image)

The transition of the return path is queried after the successor transition. If the successor transition and the return path transition are true at the same time, the step (or simultaneous sequence) following the successor transition is executed.

**Note:**

Return paths from within or into simultaneous or alternative sequences are not possible.
1.7.10 What is a Jump?

Depending on a transition condition, jumps can be used to continue the execution of the sequence path at any step within the same sequence path.

![Diagram showing the origin of a jump with three jumps]

A jump always leads immediately out of the sequence following a step (origin of the jump). Several jumps are also possible.

A jump consists of an initiating transition and an arrow specifying the jump target. The name of the step to which control is passed if the transition is true is specified as the name of the step or ???, if the jump destination is as yet unknown.

The origin and target of a jump must always be a step.

Note:
With jumps to a sequence path or from a sequence path of a simultaneous sequence, remember the possible consequences when running the chart on the AS. For further information, refer to the online help.
2 Working with the SFC Editor

Introduction

With the SFC editor, you can create sequential control systems graphically and specify the actions and step control conditions. From beginning (creating the chart or type) to end (compiling and downloading to the AS), the editor provides all the functions required.

How to use the editor is described in this chapter.

2.1 Working with Charts, Types, and Instances

2.1.1 How to Create a Chart

- SIMATIC Manager

You create an SFC chart with the SIMATIC Manager by opening the chart folder of the project in the component view or the hierarchy folder in the plant view and insert the chart there ("Insert > S7 Software > SFC" or "Insert > Technological Objects > SFC"). The chart is given a standard name by the system, for example SFC(1) that you can change. The name must be unique on the CPU. This is checked by the system.

- SFC Editor

Open the “New” dialog box in SFC using the menu command “SFC > New...”. Select the project and the chart folder in the component view.

[S7] In the plant view or the process object view, open the hierarchy folder of the project in which you want to create the chart.

In the "Object type" box, select "SFC" from the drop-down list box and enter a chart name in the "Object name" box. The name must be unique in the chart folder; this is checked by the system. When you click “OK”, a new window is opened with the SFC chart (initial status).
2.1.2 [S7] Creating an SFC Type

You create an SFC type as follows:

- In the component view of the SIMATIC Manager with the chart folder selected, right-click to open the context-sensitive menu ("Insert New Object > SFC Type") or select the menu command "Paste > S7 Software > SFC Type".
- In the SFC editor with the menu command "SFC > New...". In the "New" dialog box, select "SFC Type" in the drop-down list box of the "Object type" box.

The next free FB number is automatically reserved for the SFC type and this is copied to the block folder as a type template. This allows you to configure messages and instances of the type once it is created without having to compile the type. The FB number can be modified later in the Object Properties dialog.

When you first create an SFC type, the blocks required for compiling are copied to the current program and then managed on the ES. The blocks are included in the supplied block library.

**Note:** SFC types cannot be assigned to a hierarchy folder in the plant view, since they themselves are not relevant to the running of a project (from the point of view of the process being automated).

2.1.3 [S7] Creating an SFC Instance

You create an SFC instance by dragging the SFC type from the block catalog to the CFC chart in CFC.

The SFC types in the chart folder are displayed in the CFC block catalog (in "All blocks" and in the folder of the family if they are assigned to a family, otherwise in the "Other blocks" folder).

The SFC instance is represented like a CFC instance block. If there is not enough space to position the SFC instance; in other words, it overlaps one or more existing objects, it is displayed as an "overlapping block" (light gray and without visible I/Os). After moving them to a free position in the chart, the overlapping blocks are displayed as "normal" blocks again.

You can assign parameter values to the SFC instance in the CFC chart and interconnect it.
2.1.4 Opening a Chart or Type

- SIMATIC Manager:
  You can open a chart or type in the SIMATIC Manager by double-clicking on
  the required icon in the project in the Charts folder of the S7 program. The SFC
  editor is then started and the selected chart or type is opened.

- SFC Editor:
  In the "Open" dialog of the SFC editor, you select "SFC" or "SFC type" from the
  drop-down list box in the "Object type" box and then select the required object.
  In the "SFC" menu of the SFC editor, you will see the last SFCs to be edited
  shown as menu entries. If you select one of these names, the relevant chart or
  type is opened or, if it is already open, is displayed in the foreground.

2.1.5 [S7] Opening an SFC Instance

You can open SFC instances in the CFC chart. The SFC is started with the
 topology of the SFC instance. This topology is only viewable in the edit mode and
 cannot be modified. The properties of the SFC instance and the interface can be
 modified.

2.1.6 Copying Charts

With the SIMATIC Manager, you can copy charts; in other words, you can transfer
 tested substructures or even entire structures from one CPU to another CPU of the
 same type or copy them within the same CPU. Existing references are not lost if
 the relevant charts are copied together at one time.

For more detailed information on copying, refer to the SFC online help.

2.1.7 [S7] Copying and Moving SFC Types

SFC types are copied in the SIMATIC Manager. The run-time objects belonging to
 the SFC type are also copied. If the SFC type is not up to date (time stamp of the
 FB is older than the time stamp of the SFC type), a message will be displayed. If
 the SFC type already exists at the destination (SFC type with the same name), this
 is overwritten following a prompt for confirmation and any properties that differ from
 the previous type are passed on to the SFC instances.

SFC types are moved in the SIMATIC Manager. SFC types can only be moved
 when there are no SFC instances of the SFC type in the source. The run-time
 objects belonging to the SFC type are also moved. If the SFC type already exists
 at the destination (SFC type with the same name), this is overwritten following a
 prompt for confirmation and any differences compared with the previous type are
 passed on to the SFC instances.
2.1.8 [S7] Copying and Moving SFC Instances

You can copy or move SFC instances in the CFC chart, between CFC charts or indirectly by copying/moving the CFC chart in the SIMATIC Manager. The run-time objects belonging to the SFC instance are also copied/moved.

If you copy an SFC instance within a CFC chart or between CFC charts of the same chart folder or copy a CFC chart within a chart folder, the SFC instance is also copied. The run-time objects belonging to the SFC instance are also copied. If you copy an SFC instance between CFC charts from different chart folders or copy a CFC chart to a different chart folder, the SFC type is also copied.

If you move an SFC instance within a CFC chart, you merely change the position of the SFC instance. If you move an SFC instance between CFC charts of the same chart folder, the SFC instance is also moved. The run-time objects belonging to the SFC instance are retained. If you move an SFC instance between CFC charts from different chart folders or move a CFC chart to a different chart folder, the SFC type is also copied or moved.

2.1.9 Deleting Charts and Types

You delete SFC charts and SFC types only in the SIMATIC Manager.

- You delete SFC charts in the same way as other objects (hierarchy folders, OS pictures, ...) by selecting the object and pressing the DEL key.

- [S7] You can only delete SFC types, when no SFC instances of the SFC type exist. If instances of an SFC type exist, a message to this effect is displayed.
  
  The run-time objects belonging to the SFC type are also deleted.

You cannot delete charts or types in the SFC editor.

2.1.10 [S7] Deleting SFC Instances

You delete SFC instances in the CFC chart or indirectly by deleting the CFC chart in the SIMATIC Manager. The run-time objects belonging to the SFC instance are also deleted.
2.1.11 **Representation and Interconnection of the External View**

The interface of the SFC chart is displayed as a graphic “external view” in a CFC chart.

With the menu command “View > External View”, you open CFC with a window displaying the external view of the SFC chart.

**Display**

The external view displays the SFC like a block; in other words with the standard interface derived from the SFC run-time system. To distinguish it from CFC blocks and hierarchical charts, the external view has the “SFC chart” icon in the header. The block name is the same as the SFC chart name and cannot be modified.

**Interconnection**

You can assign textual interconnections to the I/Os and/or interconnect them with compatible I/Os of other objects or with shared addresses. All interconnections are via the sheet bar which means that you cannot place any objects (for example, blocks) in this window.

You cannot make modifications to the interface here, in other words, you cannot open the editor.

**Properties**

You can display the object properties for the entire interface (double-click in the header of the external view) or for the individual I/Os (double-click on an interconnection).
2.2 Properties of Charts and Types

2.2.1 Adapting Chart Properties

You can modify the chart properties for the active chart. With the menu command "SFC > Properties", you open the properties dialog box.

You can modify the chart properties in the following three tabs.

• General
  This tab is used to enter or modify the chart name, the author and the comment.

• Operating Parameters AS
  Here, you can make the settings for the initial status of the chart. These include: "step control mode", "mode", "command output", "cyclic operation" and "time monitoring" as well as the options for starting the chart: "Autostart" and "Use default operating parameters when SFC chart starts".

• OS
  If the “Transfer chart to OS for visualization” option is set, the SFC chart is transferred to the OS automatically with the next OS compilation.

For a description of the operating parameters, refer to Section 3.2, How the SFC Runs and in the SFC online help.

2.2.2 [S7] Adapting Type Properties

You can display and modify the properties for the active SFC type. The “SFC > Properties...” menu command opens a dialog box. You can modify the properties in the following three tabs.

• General
  This tab includes the entry and modification of the type name, the author, the version, the family, the FB number, and the comment.

• Operating Parameters AS
  In this tab, you can set the defaults for the initial status of the SFC instances created from this type. These include: "step control mode", "mode", "command output", "cyclic operation", and "time monitoring" as well as the options for starting the SFC instance: "Autostart" and "Use default operating parameters when SFC chart starts".
• **Options**
  In this tab, you can classify the SFC type for SIMATIC BATCH:
  
  - The category
    "None" ➔ there is no classification
    "EOP" ➔ the SFC type is classified as an "operation type"
    "EPH" ➔ the SFC type is classified as a "phase type".
  
  - Allow operator instructions on the OS; in other words, allow input of values in the operator dialog.

2.2.3 **[S7] Adapting Instance Properties**

You can display and modify the properties for the SFC instance opened in the CFC. The "SFC > Properties..." menu command opens a dialog box with the three following tabs:

• **General**
  This tab is used to enter or modify the instance name and the comment. All other properties (see SFC type) can only be read and not modified.

• **Operating Parameters AS**
  In this tab, you can change the SFC instance operating parameters (see SFC Type).

• **Options**
  In this tab, you can display the options set for the SFC type for SIMATIC BATCH:
2.3 The Run-Time Properties

The run-time properties of an SFC chart or SFC instance determine how the SFC is included in the execution of the entire structure on the AS. These properties are vital to the performance of the AS in terms of reaction times, dead times, or the stability of time-dependent structures, for example control loops.

An SFC instance is treated like a CFC block and therefore only the SFC chart is dealt with below.

You edit the Run Sequence with the run sequence editor. This is started with the menu command “Edit > Run Sequence...” or with the button in the toolbar.

2.3.1 Run Sequence

Each SFC chart is installed in a run sequence.

Each SFC chart must be installed in at least two tasks; in the

- task for the startup behavior ([S7] OB100)
- task for normal execution ([S7] e.g. OB35).

Changing the Run Sequence

To change the run sequence, select the SFC chart icon, select “Cut”, then select the required task and then “Paste”. If you have selected a task, the SFC chart is installed at the beginning of the task. If you have selected an object within the task, the SFC chart is installed after it.

As an alternative to cut / paste, you can also drag a chart from an open task to another task with the mouse.

Removing an SFC Chart from a Task

To remove a chart from a task, select the chart and delete it with the “Delete” function or with the DEL key. Before the chart is deleted, you are prompted to confirm your intention.

Installing an SFC Chart in a Run-Time Group

You create a run-time group with the menu command “Insert Run-Time Group...” for the selected task (in the “Edit” menu or in the context-sensitive menu). In the dialog box, you enter the name and any comment you require and the run-time attributes for the scan rate and phase offset.

Install the SFC chart in the run-time group as usual (same procedure as installation in a task).
2.3.2 Run-Time Groups

SFC charts can be installed in run-time groups if you want them to have the attributes scan rate and/or phase offset. The attributes can be set only with the object properties of the run-time group; in other words, all charts of the run-time group have the same “scan rate” and “phase offset”.

By using SFC charts in run-time groups, technologically-oriented groups can be formed in conjunction with CFC charts. From a technological point of view, a better structuring of the project can be achieved that promises a considerable improvement in performance when configurations are changed (among other things due to shorter compilation times).

2.3.2.1 Run-Time Attributes of the Run-Time Group

A run-time group has the following three attributes:

- Enable
- Scan rate
- Phase offset

[S7] The Enable Attribute

The run-time group is activated and deactivated with the enable attribute (on=1, off=0). As long as 0 is set, the run-time group will not be run regardless of any other conditions.

The enable attribute can be set dynamically. In this case, for example, the value of a block output or the statement of a step decides whether or not the run-time group is activated or deactivated.

[S7] The Attributes “Scan Rate” and “Phase Offset”

These attributes cannot be assigned directly to an SFC chart. Charts can only be given these attributes by installing them in a run-time group from which they inherit the selected attributes.

An SFC chart that is not installed in a run-time group has the default: “scan rate = 1” and “phase offset = 0”.

If you want different SFC charts on a CPU to run with different run-time attributes, they must be installed in different run-time groups.

Note: Since the SFC chart does not have its own (modifiable) run-time properties, the object properties of the selected SFC chart cannot be opened in the run sequence.
[S7] Changing the Scan Rate and Phase Offset

If you want to modify the run-time attributes, select the SFC chart in the Run Sequence window and select the “Object Properties” menu command (context-sensitive menu or “Edit” menu).

• **Scan rate:**  
  The scan rate specifies whether the SFC chart is executed by the task each time the task is run or only in every n-th run. Where “n” is an integer (n=2^t, where 0 <= t <= 15). The steps are a multiple of the basic cycle rate of the task.  
  Default: 1 (execute every run)

**Example:**
Basic cycle of a cyclic interrupt (OB33): 500 ms  
Possible cycle rates with scan rate: 1s, 2s, 4s, 8s, 16s etc.

• **Phase offset:**  
  The phase offset can be used to achieve a better distribution of load on the CPU. It must be considered in conjunction with “n”, the scan rate. The SFC chart is processed as often as specified by “n”, offset in each case by “m” units of the cycle.  
  Where “m” is an integer and 0 ≤ m ≤ (n-1)  
  Default: 0 (no phase offset)

**Example:**
Basic cycle of a cyclic interrupt (OB33): 500 ms  
Scan rate: **16**. The SFC chart is executed every 8 seconds (0.5s x 16).  
Phase offset: **3**. The SFC chart is executed after 1.5s; 9.5s; 17.5s etc.

⚠️ **Caution!**  
Whenever possible, you should only use the scan rate and phase offset in the tasks that execute in defined cycles; in other words, with cyclic interrupts. In all other tasks you should be extremely careful, particularly with hardware interrupts and special tasks. Here, you should not change the default scan rate=1 and phase offset=0.
2.4 Configuring Sequential Control Systems

Requirements
Before you can configure sequential control systems, you must first create the required basic control functions with CFC and/or STEP 7 tools. At the same time, the AS blocks to be used in the SFC charts or SFC instances are also inserted. Automation functions that do not yet exist can be added later and then used in an SFC.

Procedure
When you configure a sequential control system (SFC chart or SFC type),
• create the sequence topology with the required number of sequencers and the required arrangement of SFC elements.
• configure the start condition, the preprocessing, and the postprocessing in the Properties dialog of the sequencers.
• configure the steps and transitions, the actions and conditions in the Properties dialog of the steps and transitions. These procedures are dealt with in greater detail in the following topics.

Color Settings
The objects of a chart are displayed in different colors depending on their current state. The elements of an unselected sequential control system, for example, are displayed in “white” (parameters not set) or “gray” (parameters set) and in “blue” when they are selected.

With the “Options > Customize > Colors...” menu command, you can select your own color scheme for certain elements.
2.4.1 Creating the Sequencer Topology

Appearance of the Sequencer

In its initial state, the new SFC consists of one sequencer; this can, however, be extended to up to 8 (SFC chart) or 32 (SFC type) sequencers. Each sequencer is created in a separate working window; You can change from one sequencer to the next with the tabs at the lower edge of the window.

A newly created sequencer ("Insert > Sequence > ..." menu command) is inserted in its initial status consisting of an initial step, transition and final step, at a selected position in the SFC and a tab is added at the lower edge of the window. Each tab contains the name of the sequencer (RUN, SEQ1, ...).

See also Section 2.4.2 Configuring Several Sequencers

If you insert or delete SFC elements in the sequencer, its layout is changed automatically according to predefined rules. These determine the spacing between elements, the size of steps and transitions, the alignment of alternative sequences etc. You can change the display/layout rules at any time (Options > Customize > Display... menu command).

You can center the sequencer topology in the window. With the zoom functions, you can increase or reduce the size of the display (in percentage steps determined by the zoom factor).

Adding Elements

To add further elements to the SFC, select the icon of the required element in the element bar.

The mouse pointer changes its appearance from an arrow to the selected symbol with a positioning cross. To insert the sequence element, position the cross at the required position (the insert position is indicated by a green line) and then click the left mouse button. The inserted elements are selected and displayed in color.

Syntax Rules

The sequencer topology is formed by the sequences of steps and transitions. The fundamental rule of the sequencer topology is that a step (S) must be followed by a transition (T) and a transition must always be followed by a step (sequence : S-T-S or T-S-T). The editor automatically adheres to the rules.

Example:

If you insert a simultaneous sequence in a sequencer following a transition and before a step, a transition is created automatically before the step since the syntax rules require a transition before and after a simultaneous sequence.
2.4.2 Configuring Several Sequencers

An SFC can contain several sequencers that can be used for different applications. By specifying different start conditions, you can arrange for a particular sequencer to start when a particular event occurs. You can, for example, configure a separate sequencer for every operating state (ready, active, error, ...) or for every control strategy (heating, cooling, tempering, ...).

Note:

There are ready-made sequencer templates available in the "SFC Library". You can copy these templates and adapt them to your own purposes.

[S7] Start Condition of the Sequencers

The first sequencer of a chart or type has the condition "SFC.RUN=1"; the start condition of every further sequencer you add is empty and therefore not satisfied and it will never execute. In contrast to newly created transitions that are always satisfied, a new sequencer must always be given a defined start condition (Sequencer Properties > Start Condition tab).

Since it is possible that several start conditions can be satisfied at the same time, you can assign different priorities to the individual sequencers (Sequencer Properties > General tab, Priority: 1 to 32).

You can formulate the start condition of a sequencer so that the status of the operating state logic is checked and the appropriate sequencer executes when the SFC is in a particular state.

Examples

Example 1: You configure a sequencer with a start condition that queries one of the control strategies of the SFC. The formulation is, for example, "SFC.QCS=1". If the SFC is set to this control strategy, the sequencer executes and is not dependent on the operating state of the SFC.

Example 2: You configure a sequencer with the start condition "SFC.IDLE=1". This sequencer executes when the operating state is "Ready".

Example 3: Any process status can also be queried as the start condition. You interconnect it with the external signal "LOCKERROR" (input of the SFC) →, the SFC changes to the "Error" state if the error signal is applied. You also configure a sequencer for handling the problem whose start condition is, for example, "SFC.ERROR=1 AND process status=1".

Example 4: As an alternative to example 3, problem handling can also execute without a state change. To achieve this, you configure a sequencer with the start condition "process status=1" and assign it high priority. This sequencer always executes when the problem occurs and the sequencer currently executing has a lower priority than the sequencer for handling the problem. In this case, you do not interconnect the process status with the "LOCKERROR" input otherwise this would mean a change to the "Error" state.
Notes on Configuration

- You can insert a new sequencer consisting of an initial step, a transition and a final step with the menu commands "Insert > Sequence > Before Current Sequence" or "Insert > Sequence > At End".

- Ready-made sequencers are already available for various standard scenarios. These sequencer templates are available in the "SFC Library". You can copy these templates and adapt them to your own purposes.

- You can copy sequencers and paste them again or move them to change their order.

- The names of the steps and transitions must be unique within a sequencer, the same names can be used in different sequencers.

- You can also configure a cyclic action for each sequencer. The cyclic action consists of a part that is executed before the cyclic sequencer processing, the preprocessing and a part that is executed after the cyclic sequencer processing, the postprocessing. You configure both parts in the "Sequencer Properties" dialog box. The dialog box contains the "Preprocessing" and "Postprocessing" tabs whose structure corresponds to the processing phases of steps.
2.5 Creating SFC Elements

2.5.1 Creating a Sequence

When you create a sequence, then depending on the position a Step-Transition sequence (ST) or a Transition-Step sequence (TS) is created (see Figure 2-1). You create a sequence by clicking a vertical link in the chart between a step and transition or between a transition and step.

Figure 2-1: Creating a TS or ST Sequence Depending on the Location

If you move the mouse pointer to a permitted insertion point in the existing sequence, you will see a horizontal green line.

By holding down the left mouse button and dragging the pointer vertically, you can select the length of the sequence you are inserting. The current length that will be inserted (number of ST/TS pairs) is displayed as a number at the insertion point.
2.5.2 Creating and Extending a Simultaneous Sequence

When you create a simultaneous sequence, two sequence paths are generated each consisting of one step. Depending on the point of insertion, a further transition is added automatically before or after the simultaneous sequence to ensure that the syntax is maintained.

If you open a lasso (in the Edit mode) around the elements of a sequence path, the enclosed elements become part of the left sequence path of the generated simultaneous sequence.

You can add further sequence paths to a simultaneous sequence or delete paths and also insert them in a different sequence. You can move a sequence path within a simultaneous sequence or to any other position in the chart (except in the return branch of a loop). If you delete the second last path, the remaining path is simply integrated in the surrounding structure and the simultaneous sequence is eliminated.

To add further sequence paths, simultaneous sequences, or alternative sequences to a simultaneous sequence, change to the required insert mode and click the mouse with the positioning cross located on the upper or lower double line.

If you move the mouse pointer to a permitted insertion point in the existing sequencer, you will see a horizontal green line. Within the simultaneous sequence (in the vicinity of the upper simultaneous divergence or lower simultaneous convergence), the vertical green line indicates that you are inserting a further sequence path. If, for example, you insert an alternative sequence beside a sequence path, an extra step is created before and after the sequence to maintain the correct syntax.
2.5.3 Creating and Extending an Alternative Sequence

When you create an alternative sequence, two sequence paths are generated each containing one transition. Depending on the point of insertion, a further step is added automatically before or after the alternative sequence to ensure that the syntax is maintained.

If you open a lasso (in the insert mode) around the elements of a sequence path, the enclosed elements become components of the left sequence path of the generated alternative sequence (refer to the description “Creating a Simultaneous Sequence”).

You can add further sequence paths to an alternative sequence or delete paths and also insert them in a different sequence. You can move sequence paths within the alternative sequence or to any other position in the sequencer. If you delete the second last path, the remaining path is simply integrated in the surrounding structure and the alternative sequence is eliminated.

To add further sequence paths, simultaneous sequences, or alternative sequences to an alternative sequence, change to the required insert mode and click the mouse with the positioning crosshair located on the divergence/convergence line.

If you move the mouse pointer to a permitted insertion point in the existing sequencer, you will see a horizontal green line. Within the alternative sequence (in the vicinity of the upper divergence or lower convergence beside the sequence path), the vertical green line indicates that you are inserting a further sequence path. If, for example, you insert a simultaneous sequence beside a sequence path, an extra transition is created before and after the sequence to maintain the correct syntax.
2.5.4 Creating a Loop

When you create a loop, a sequence path (that can consist of a single step) and a return path with a transition are generated.

You can create loops around existing sequences. You select the beginning and end of the loop by positioning the mouse pointer on the vertical link, holding down the left mouse button and dragging vertically to the required position and releasing the button. The syntax is maintained by adding whatever elements are necessary. If, for example, you create a loop around a transition, the enclosed sequence then consists of this transition and a step before and after it. Below the loop, a further transition is added.

The start and end point of a loop cannot be modified later. You can, however, move the elements you want to be included in the sequence of the loop into the loop and achieve the same result.
2.5.5 Creating a Jump

When you insert a jump, a transition is created with an arrow and information about the destination of the jump.

When inserting loop, click on the vertical connecting line of the sequence immediately following a step. With a single click, you create a jump with an undefined destination. The destination is displayed as question marks (???).

If you require more than one jump from a step, click on the horizontal line of the jump branch. The branch with the jumps is then extended by a jump with each click.

When you insert the jump, you can also select the jump destination directly. Drag the mouse from the point of origin of the jump directly to the destination step and then release the mouse button. Instead of the question mark, the name of the step is now entered as the jump destination.

Note:
With jumps to a sequence path or from a sequence path of a simultaneous sequence, remember the possible consequences when running the chart on the AS. For further information, refer to the online help.

Changing the Destination of a Jump

You specify the destination of the jump by changing the name (???) in the object properties of the destination. Double-click the destination to open a dialog box. All the existing steps of the chart are listed and can be sorted. From this list, you select the step name for the destination of the jump.

Note:
If the jump destination is deleted, all the jumps to this step become undefined. If the step name of a jump destination is changed later, all the jumps to the step are automatically adapted.
2.5.6 Creating and Editing Text Objects

You can insert, delete, copy, and move a text object at any (free) position in the chart (you cannot drag to other charts).

After inserting a text object using the button in the toolbar or selecting the "Insert > Text" menu command, an open text box is displayed in the window. The text cursor is active and you can begin editing immediately. A line break is added automatically at the right edge of the box. If you enter more text than can be displayed in the box, the size of the box is not increased automatically and the text is moved out of the visible area. You can make the entire text visible by increasing the size of the box manually.

To change the size of a box, click on the box handles and drag with the mouse until the required size is reached. If you change the width of the box, the length of the text lines is automatically adapted. You can pick up the box using the handles of the frame and move it to any position in the window.

When you open a text box, the cursor is positioned at the point in the text at which you clicked with the mouse. You can exit the editing mode and close the text box by clicking outside the text box with the mouse.

**Note:** If elements of the sequence topology are covered by the text object, the text object is displayed as a frame with a transparent surface (no content). The chart elements below it remain visible.

Copying, Moving, Deleting

With the mouse pointer over the selected text box (displayed in a frame), open the context-sensitive menu with the right mouse button. This contains the menu commands: "Cut Text Object", "Copy Text Object", Delete Text Object".

To paste, click on a free position in the chart and then select the “Paste” menu command (context-sensitive menu or “Edit” menu). Click again at the required position to insert the text object (the mouse pointer is displayed as a symbol for “inserting” or “copying”).
2.6 Editing SFC Elements

In the “Edit” menu (and in the context-sensitive menu), you will find further functions for editing the SFC.

**Copy:**
You can copy the selected elements of a sequence path that form a syntactical unit (sequence of elements without gaps) and position them at a different, syntactically correct position within the sequence path or in a different sequence path on the same or another CPU. If necessary, new names may be assigned to the copied elements automatically. The copied elements contain the same actions or conditions as the originals.

Copying jumps: If you copy a sequence that contains a jump and the step of the jump destination, the jump destination is adapted appropriately in the copy. If you copy a sequence that contains a jump but the destination of the jump is not within the copied object, the jump destination is undefined (???).

**Cut and Paste:** You can move the selected elements of a sequence that form a syntactical unit (no gaps) to another syntactically correct position within the sequencer or to another sequencer of the same or a different CPU (cut and paste).

**Delete:** The selected elements are removed from the sequencer topology following a prompt for confirmation (“Do you really want to delete the selected objects?”). If you delete only one element from a syntactical unit, the syntax is immediately restored by entering a new element to suit the syntax (this has no parameters assigned). This means that you have only deleted the parameter settings for the object.

The last step of a sequence in a simultaneous sequence cannot be deleted. To delete a sequence consisting of only one last step, you must select the sequence path by clicking on the vertical link.

The same principle applies to alternative sequences.

If you only select the transition in a jump, its content is deleted; if you select the jump destination (transition and jump destination are selected), the jump is deleted.
2.7 Editing in the Object Properties Dialog

The steps and transitions entered in the sequence topology must now be linked to the “block world” or basic control. You do this in the “Properties” dialog for the steps and transitions (Sections 2.7.1 and 2.7.2).

2.7.1 Editing Object Properties: Step

You open the object properties by double-clicking the step you want to edit or using the menu command “Object Properties...” of the “Edit” menu or the context-sensitive menu if a step is already selected. A dialog box is displayed in which you specify the properties or formulate the actions.

When you edit the step, the Properties dialog is divided into four tabs:

“General” Tab

In the "General" tab, you can enter or modify the step name, minimum and maximum execution time, the comment for the step and the OS comment. In the "Number" box, you can see the unique number of the step in the sequencer as assigned by SFC.

By clicking the “Confirmation” check box, you can assign a flag to the step. This flag determines the behavior of the step when the chart is run on the AS in the “T / T and C” mode in other words, transitions following a step with this flag only become active and pass control to successor steps when they are satisfied and confirmed by the operator (as in the “T and C” mode). Without these flags, the successor transitions become active once their conditions are satisfied (as in the “T” mode).

In the Minimum: box, you can set the minimum time that a step should remain active regardless of whether the successor transition is already satisfied.

With the Maximum: setting, you can specify a time for the time monitoring limiting the maximum time that the step can be active.

Comment and OS comment boxes, you can enter comment text, for example, describing the actions to be executed by the step (maximum characters for step: 80, for OS: 512). The OS comment is used in process control for visualizing the step.
Initialization/Processing/Termination Tabs

The tabs for the actions (processing phases) Initialization, Processing and Termination are structured identically. Here, you configure the statements that will control the process during the initial, normal, and final processing of the step (see Section 3.3.7 Phases of a Step).

The statements are adopted as OS comments if the check box is selected in the relevant line.

Formulating Actions

You enter the statements for actions in a formatted dialog. For each step, you can formulate up to 50 statements per action. Ten of these are visible in the dialog box. You can scroll the list with the scroll bar.

If you position the mouse pointer on an input box, the complete entry is displayed as fast information including the data type and object type. Example:

3.       DP666\Reactor1\Motor3\CFC8.Ctrllr2.P_SEL [BOOL] CFC

A double backslash (\\) is displayed between the path and the CFC name.

In unselected lines, you can use the “Copy/Paste Action” functions in the context-sensitive menu to copy the statements of a complete action and paste them into another action. Using this function, you can, for example, copy the statements from the “Initialization” action to the “Termination” action and then change the settings “TRUE” to “FALSE”.

You will find the formulation options in the online help for SFC.

Entering Addresses

Using the “Browse” dialog:

When you browse (for example through CFC charts), all the available objects of the chart folder are found and displayed. Using a filter, you can list the I/Os that are of interest in this particular phase of configuration. The filter is active only when the criteria are clearly defined; in other words, if an address is, for example, a constant, all I/Os are displayed unfiltered and can be selected for the second address.

You can insert the selected I/O in the address field with “Apply”, by double-clicking or by dragging.

From the I/O window:

You can also drag the I/Os of the SFC directly from the window of the I/Os to the address box of the Properties dialog.

From the CFC chart:

With the CFC chart open, you can select the block I/Os and can drag them to the address box of the Properties dialog.

If you insert a block I/O with a value identifier in the address box, the value identifier is also entered in the right (empty) address box. If there is more than one value identifier for this I/O, you can open the drop-down list box with the value identifiers in the right address field by pressing the key combination ALT + DOWN.

In the drop-down list box, you can select the required value identifier for this address.
To allow the value identifiers to be displayed in SFC, the option “Parameter: Value identifier” must be selected in “Options > Customize > Layout...”.

**By editing:**
With text entries, make sure that the names are consistent. A symbol that does not exist in the symbol list (or a referenced symbol that is subsequently renamed) cannot be checked in the editor and is set up as a textual interconnection.

In the addresses, you can enter CFC block I/Os whose block does not yet really exist in the CFC chart. These instructions are displayed in the statement list on a yellow background and are set up as textual interconnections.

**Example:** Turning the SFC chart on and off
[S7] The statement “<SFCchart>.INTONOFF := TRUE”, activates an SFC chart. The statement “<SFCchart>.INTONOFF := FALSE” deactivates an SFC chart (its final step is executed and then the chart is deactivated).

**Note:**
When you enter statements, they are checked so that correct syntactic and semantic formulation is guaranteed. The program, for example, also checks whether the data types of the logically combined addresses are compatible.

With SFC access to CFC blocks, please remember the following:

- Being able to modify a type centrally means that it is possible to replace or modify blocks of which CFC block instances have already been generated. The type changes are also made to the CFC blocks. If there are SFC accesses to modified blocks, these modifications must also be made to the addresses in actions and transitions.

- Access to CFC blocks can also be modified in the relevant CFC chart. This modification is, however, restricted to “rewiring” SFC accesses; in other words moving the access to another block I/O (with ALT + Drag & Drop).

**Documenting the Configuration**

You can document the configured actions of the step.

With the "Print" button in the Object Properties dialog, you obtain a log of the step with information about the properties and assignments for initialization, processing and termination.
2.7.2 Edit Object Properties: Transition

You open the object properties by double-clicking the transition you want to edit or using the menu command “Object Properties...” of the “Edit” menu or the context-sensitive menu if a transition is already selected. A dialog box with three tabs is opened in which you can specify the properties, the conditions and the OS comments.

“General” Tab

In the “General” tab, you can enter or change the name and comment.

“Condition” Tab

In the “Condition”, you specify the step control conditions for the selected transition.

Figure 2-2: Logical Combination of Conditions with Three-Stage Transition Logic
Formulating the Conditions of a Transition

You formulate a transition as a Boolean expression, that can consist of 2 x 5 and 2 x 3 conditions. The conditions are combined using three-stage transition logic.

The Boolean operators are designed as buttons. With a simple mouse click on the operator, you can change it from "AND (\&)" to "OR (≥1)". To make a “NAND” from an “AND” and a “NOR” from an “OR”, click the output of the operator. The negation is displayed by a period in bold print on the output line.

You will find the formulation options in the online help for SFC.

If you position the mouse pointer on an input box, the complete entry is displayed as fast information including the data type and object type.

You should whenever possible enter the addresses using the “Browse” selection dialog or from the open CFC chart since this means that the addresses are unique. As when selecting addresses for the steps, you can also enter the value identifiers of the I/Os for the transitions.

Note:

When you enter conditions, they are checked for consistency so that correct syntactic and semantic formulation of the subexpression is guaranteed. The program also checks whether the data types of the logically combined addresses are compatible.

The predefined three-stage logic is normally adequate. If you require more complex formulations, you can create them as a CFC chart and enter the results calculated in the chart as the address for transition conditions.
“OS Comment” Tab

Here, you can enter a comment text for each condition that is displayed on the OS during process control.

When you first open the “OS Comment” dialog, the formulated condition is entered as an OS comment and can then be changed as required.

• Unless you edit and modify the OS comment, the OS comment is adapted automatically whenever the compare condition is changed. This automatic updating also applies to renaming, copying, moving or deleting the referenced CFC blocks.

• If you change the condition belonging to a comment text that you have edited, this is not automatically adapted. In this case, a dialog box is opened telling you that you should check the comment text. You can cancel the change in this dialog box by clicking the “Cancel” button.

• You can initialize an individual OS comment in the SFC editor by deleting the OS comment. The text from the compare condition is then used again as the OS comment and from this time onwards is automatically adapted whenever the condition is changed providing you do not edit the OS comment.
  As an alternative: You can enter the original OS comment (text of the condition) again using the “Use Default” context-sensitive menu command.

• You can edit the OS comments later.
  With the “Options > Edit OS Comments...” menu command, a dialog is opened in which you can select how the OS comments are handled in the transitions:
  - leave unchanged or
  - use all conditions.
  In this dialog, you can also decide on the range to which the processing of the OS comments applies:
  - entire chart folder
  - the current chart
  - the selected steps/transitions.

Documenting the Configuration

You can document the configured conditions of the transition. With the “Print” button in the Object Properties dialog, you obtain a log of the transition with information about the properties and parameters of the conditions.
2.8 [S7] Creating an SFC Type

There are two possible ways of creating and modifying an SFC type:

- **Creating and modifying in a library** has the advantage that the master for the SFC type is always in the library and that the test project can still run until a new version of the SFC type is adopted.

- **Creating and modifying in a project** has the advantage that every change to the SFC type can be checked immediately since you are working directly with the master.

Creating the SFC Type in the Project

You can only create an SFC type when a PCS 7 project already exists.

**Basic Procedure:**

- **Preparations:**
  - With LAD/FBD/STL: Definition of which I/Os of the blocks (of the block library you are using) need to be connected to the SFC types (using the system attribute "S7_contact" for the block contacts). (This has already been done for the technological blocks from the PCS 7 library.)
  - In CFC: Import the blocks ("Options > Block Types...")

- **Creating (SIMATIC Manager):**
  - Create a new SFC type ("Insert New Object > SFC Type")
  - Adapt the SFC type name
  - Open the SFC type

- **Adaptation (SFC Editor):**
  - Adapt the properties ("SFC > Properties...");
    "General" tab: Author, Version, Family, FB number, Comment;
    "Operating Parameters AS": Options for "Defaults" and "Start Options".

- **Characteristics (SFC Editor), see Section 2.9**
  - Definition of the characteristics ("View > Characteristics")
  - Add the control strategies
  - Add the setpoints
  - Add the process values, block contacts etc.
• Sequencers (SFC Editor)
  - Adding/editing the sequencers
  - Editing the start condition (sequencer properties)
  - Possibly editing preprocessing/postprocessing of the sequencer
  - Configuring Sequencers

Note:
There are ready-made sequencer templates available in the "SFC Library". You can copy these templates and adapt them to your own purposes.
2.9 [S7] Configuring in the Characteristics Dialog

What are characteristics?

A characteristic is the technology-oriented functionality of an SFC type as described by a set of attributes.

A characteristic is described as follows:

- by the attributes such as data type, initial value, upper limit, ...
- by the I/O names and the I/O group, this is the interface description of the I/Os required for the characteristic.
- by mapping the attributes to the I/Os of the I/O group or to the faceplate.

The following characteristics are available for creating the type:

- **Control strategies**
  Control strategies are used for the structuring of an SFC type as required for process control. With the Characteristics dialog, you can define control strategies so that these can be used in the sequencers (for example heating, cooling, ...). The control strategy can be set by the operator or by a higher-level controller such as SIMATIC BATCH. The control strategies are optional.

- **Setpoints**
  Setpoints are used to control the SFC type. These can be set by the operator or by a higher-level controller such as SIMATIC BATCH. When using control strategies, all setpoints are initially assigned to all control strategies. Using the Object Properties dialog of a setpoint, the setpoint can be assigned selectively to specific control strategies.

- **Process values**
  Process values are used to control the SFC type based on process signals (for example, the value for the fill level).

- **Control values**
  Control values are used to control external logic with this SFC type.

- **Parameters**
  Parameters are used to modify the behavior of the instances of the SFC type.

- **Bit memory**
  Bit memory is used to store values temporarily. Memory bits are created as static variables, that are not visible in the interface displayed in SFC.

- **Timers**
  Timer objects are used to configure time-driven sequences in an SFC type. Timer objects are implemented with a standard timer block (TIMER_P) that supports various modes (pulse, extended pulse, on delay, latching on delay, off delay). The TIMER_P instance is created when the SFC type is compiled and is embedded as a multiple instance block in the SFC type.
• **Instruction texts**
  Instruction texts are additional texts for display in the faceplate. Here, text lines containing additional information for the operator can be displayed. Via an output (OPTIPNO) in the interface, the texts defined in the Characteristics dialog can be displayed by setting the output in a step. These can be acknowledged by the operator.
  There are various standard instruction texts that are displayed, for example during the setpoint check (for example "lower limit violated").
  An instruction text is not connected to the message system.

• **Block contacts**
  Block contacts represent blocks of the basic control. In the interface of the SFC type, elements are defined that can be used in the sequencers (see also Section 2.10, The I/O Groups).

• **Position texts**
  The position text is used to display the current position within the sequencers. It consists of a number and an assigned text. The text is displayed in the SFC faceplate as an alternative to the step text.
  The number of the position can be evaluated by a higher-level control system (for example BATCH or SFC).
  The position number is set from within individual steps. Using the same number, various steps can be grouped together.
2.10 [S7] The I/O Groups

To define I/O groups, there are templates that describe which I/Os are required for a characteristic. The names of the individual I/O elements are made up of the I/O names configured in the characteristic and a fixed part.

The interface descriptions for the characteristics control strategies, setpoints, process values, control values, parameters, bit memory, timers, instruction texts, and position texts are predefined. Apart from the control strategies, instruction texts, and position texts where the I/O group is used exactly once with definable enumerations, instances of the characteristics can be used more than once.

Block Contacts

To link the blocks of the basic control using block contacts, you can define block I/O groups for the block types. These are specified by selecting I/Os of the block type.

The I/O groups are defined for a block type in the programming language of the block type as the system attribute "S7_contact". When necessary, you can make project-specific adaptations to the I/O groups for the supplied block types:

The system attribute "S7_contact = true" means that the corresponding I/O belongs to the I/O group of the block type. There is one block I/O group for each block type.

If different I/O groups are required for a block type, this can be achieved as follows:

- The I/O group is defined as a set of all required I/Os
- The block type is copied or an empty block is created with the block interface of the original block reduced to the I/Os required as the I/O group. The required I/O group is defined in the created block type. The resulting block contact (the SFC instance) can later be interconnected with an instance of a different "suitable" block type.

By defining an instance of an I/O group (in other words the characteristic "block contact") in the characteristics dialog, the selected block I/Os are then created in the SFC type. The I/Os of the type IN and OUT are created as their opposites in the SFC type. The IN_OUT I/Os of the block type are created as OUT in the SFC type. This allows a concrete CFC instance of the block type to be interconnected with the corresponding I/Os of an SFC instance. An instance of the "block contact" characteristic is therefore always assigned to exactly one block type. Which block type this is, can be seen in the Block column in the right-hand pane.

This allows the independent configuration, commissioning and visualization of blocks of basic control. At the same time, SFC types can be configured on the basis of block contacts and later interconnected with the blocks of basic control.
2.11  [S7] Configuring Messages

Configuring Messages in SFC

You can start configuring messages in SFC with the menu command “SFC > Messages...”.

You can configure specific message texts for each SFC chart/SFC type. You can modify the message texts in a dialog (for example, for distinguishing messages of different charts/types).

Note:

You must have already configured messages for the SFC type before you create SFC instances. Subsequent changes to the message properties are not automatically passed on the SFC instances. If you want SFC instances to have the new message properties, you must recreate the instances.

Two message events have standard texts:

• Operator prompt
• Step Timeout

With an SFC type, an additional 10 status messages have default texts.

Operator Prompt

The operator prompt is a message that does not require acknowledgment and that is signaled in an SFC chart with the NOTIFY message block and in an SFC type with NOTIFY_8P.

Step timeout

The "step error" is a process control message that must be acknowledged and is signaled along with 3 associated values using the message block ALARM_8P. The remaining 7 free messages and associated values can be used as required.

The standard interface therefore has I/Os via which messages can be triggered (SIG_2 ... SIG_8) and associated values AUX_PR04 ... AUX_PR10 (only with an SFC type). These I/Os can be used by having interconnections in the actions of the steps or by having direct block interconnections.

When configuring messages, please remember that the messages of ALARM_8P must only be assigned message classes with mandatory acknowledgment.
2.12 Compiling and Downloading

General Aspects of Compiling

When compiling (scope: entire program) all charts, SFC types, and SFC instances of the current chart folder are converted to source language and then compiled. After modifying the SFC chart (SFC type, SFC instance), you can compile so that only the changes are compiled.

When you compile, first all SFC types then all SFC instances and then all SFC charts are compiled. Finally the CFC charts are compiled.

The SFC type is compiled only when a modification has been made to the type since the last compilation that is relevant for compilation.

After providing the SFC type with an interface, a function block type is generated containing the I/Os defined in the interface and the configured sequencers. As with the SFC chart, an action and transition FC are also generated.

Saving Settings without Compiling/Downloading

You can save the settings in the dialog box without starting compilation or a download (“Apply” button). This can, for example, be useful when you want to use the "Compile and download function" function in the SIMATIC Manager. Refer to the online help of the SIMATIC Manager.
2.12.1 Compile

[S7] Customizing the Compiler

With the “Customize > Compilation...” command in the “Options” menu, you open a dialog box in which you will see information about the resources used in conjunction with compiling charts. Here, you can

• decide which warning limits will apply so that possible dangers are detected before you download.

• decide which resources should remain unused during compilation of the charts of the current chart folder.

• view the statistics showing how many resources (DBs, FCs) are available for compiling and how many are already being used.

[S7] Compiling

The “SFC > Compile...” menu command opens a dialog box in which you can select one of the option buttons "Entire program" (all objects are compiled) or "Changes only" (only the objects changed since the last compilation are compiled).

• Option Delete empty run-time groups: If this option is set, the empty run-time groups are deleted prior to compilation.
  The empty run-time groups can occur as a result of copying when branching and merging projects. When these empty run-time groups are created, the original names of the run-time groups are lost due to implicit incrementation of the numbers.

• Close textual interconnections option: If this option is set, all textual interconnections for which the referenced interconnection partner exists are closed and turned into real interconnections.
  Note: If the option is not set or if the textual interconnections cannot be closed, they are ignored during compilation. A warning is entered in the log indicating which I/Os still have a textual interconnection. One unclosed Textual interconnection is also tolerated in a download.

• The options "generate module drivers" and "update sampling time" are used in CFC and are relevant only when CFC charts have been modified that are also compiled in this S7 program.

Select the type of compilation you require (“Entire program" or “Changes only") and start the compilation with “OK".

The charts of the current program (chart folder) are checked for consistency and then compiled.
Consistency Check

During compilation, a consistency check is run automatically. The messages are written to a log file.

If errors and warnings occur, you can jump to the relevant SFC by double-clicking the message.

You can also run the consistency check without a full compilation by selecting “SFC > Consistency Check...”.

Logs

Following the consistency check or compilation, you can display and print out the messages of both activities using the menu command “Options > Logs”.

"Options > Logs ..." menu command).

[S7] AS Comparison

If you want to compare the time stamp of the last modifications before downloading, you can start the “Compare” function in the “AS” menu. A dialog box with the date and time of the following is displayed:

• The last download-relevant change
• The last offline program change
• The last online program change

If the time stamp of the last download-relevant change is older than the last offline program change, this has no further effects on the program execution on the CPU; you do not need to download the program again.

If the time stamp of the offline program modification is older than the download-relevant modification, you must compile the charts and download them to the CPU so that they match.

If the time stamp of the online program modification is older than the time stamp of the offline program modification, you must download the user program from the PC/PG to the CPU so that they match.

Note on H CPUs: If the H CPU is in the solo mode, for example after the failure of a CPU and there was a CPU failover, an online access results in a dialog being displayed. In this dialog, you can select the required CPU. In the redundant mode, this dialog does not appear.
2.12.2 [S7] Downloading

After you have compiled, you can download the user program to the AS.

The following requirements must be met before you can download:

- There must be a connection between the CPU and your PG/PC.
- The Edit mode is set.

With the programs created in SFC, you must always download to the AS from SFC (or CFC), since only this download function guarantees the consistency of the configuration data with the AS data. The same download function is also used if you select the "AS > Compile and Download Objects" menu command in the SIMATIC Manager.

On the other hand, copying blocks in the "Offline Block Folder" to insert them in the "Online Block Folder" is not permitted.

If these requirements are met, you can start the download with the menu command "AS > Download...".

If you have made download-relevant changes in the configuration and have not compiled since you made the changes, you will be prompted to compile before you download. If the compilation is free of errors, the download will be started automatically when compilation is completed.

In the "Download" dialog, you can choose between "Entire program" (CPU in STOP) and "Changes" (CPU can be in RUN-P).

Downloading the Entire Program

If you select "Entire program", all the charts of the active chart folder are downloaded to the CPU. After prompting you for confirmation, the CPU is set to "STOP" and all the blocks on the CPU are deleted.

**Note on H CPUs:** If the H CPU is in the solo mode, for example after the failure of a CPU and there was a CPU failover, an online access results in a dialog being displayed. In this dialog, you can select the required CPU. In the redundant mode, this dialog does not appear.

Downloading Changes Only

If you select "Changes only" in the "RUN-P" CPU status, you can download the configuration changes to the AS without needing to change the CPU to STOP. With this type of download, you only download changes that have been made since the last download.
[S7] Notes on Downloading Changes

- If the sequencer topology of SFC charts/SFC types has been changed (steps or transitions have been added, deleted, copied, moved, jump destination changed...), and if the modified sequencer is active, these charts or all instances of the type must be deactivated. Deactivating before the download and reactivating after the download is handled by the downloader following positive acknowledgment of a query. Otherwise, the download is aborted.

- Modifications to the interface of the SFC type are transferred to the SFC instances immediately. The SFC instances must therefore be deactivated during downloads and execution stopped on the CPU. Deactivating before the download and reactivating after the download is handled by the downloader following positive acknowledgment of a query. Otherwise, the download is aborted.

- If SFC charts/types/instances have been modified (chart properties, object properties are the steps/transitions) without changing their structure, you can download the charts after they have been compiled while the CPU is in RUN without needing to deactivate be modified SFC chart.

- If you have not changed the SFC itself, but only the objects that are accessed (for example a symbol in the symbol table, run-time groups, block I/O), you do not need to deactivate the SFC before downloading changes.

- After downloading changes, the halted SFC is not started with the property “Autostart: on” but must be started again by the operator or via the external view/SFC instance.

- **F systems**: You can only download changes to programs with modified F components after entering an F password. Without this legitimization, downloading is aborted.

- **H systems**: If you download program changes to a CPU operating in the solo mode and then “switchover with modified configuration” (menu command “AS > Operating Mode...”), these changes are lost and you then have to download the entire program.
  Remedy: Download in the redundant mode. In this case, you must make sure that the operating mode remains unchanged until the download is completed.

---

Caution!
Read the information about the causes of stoppages when downloading changes in the online help.
2.13 [S7] Parameter-Controlled Operation

Below, you will find brief information about parameter-controlled operation and how it is configured. For a more detailed description with examples, refer to the online help of SFC.

What does parameter control mean?
Apart from fixed sequential control systems, parameter-controlled systems are used in batch processes. Parameter-controlled sequential control systems have variable parameters.

These variable parameters are located in a shared data block (recipe data block). You can give the recipe data block a symbolic name, for example “RecParDB”.

The variables of the recipe data block are values assigned to the parameters of the basic control during configuration.

Configuring Parameter-Controlled Operation
You configure parameter-controlled operation with the SFC editor. The procedure is basically the same as when configuring the sequential control system. During processing in the Object Properties dialog, you assign the values from the recipe data block to the parameters of the basic control.

Modification on the OS
By changing the content of the recipe data block, parameter-controlled operation is possible with different parameter sets. The parameters are changed on the OS.
3 Sequential Control System on the AS

Introduction
This chapter describes the functions of sequential control systems on the AS, when steps and transitions are processed, and the effects of a statement in the processing phase (action) of a step.

3.1 General

Requirements
The SFC created with the SFC editor is defined by the sequencers (start condition and pre/postprocessing) and their topologies, by the actions of the steps, the conditions of the transitions, and by the run-time properties, the program has been compiled and downloaded to the AS.

State of the Sequential Control System
After downloading to the AS, the sequential control system is in a defined state. This defined state is specified with the SFC editor (default: "Ready"). By assigning start conditions and specifying the order of execution for the sequencers, different program sequences are executed on the AS depending on the SFC state.

You can modify how a chart is executed either by changing the operating parameters using the SFC editor during test and commissioning or on the OS, for example by changing the step control mode “step control with transition only (T)” to “step control confirmation by operator (C)” or changing the time monitoring from “Off” to “On”.

Interaction with the Basic Control
The sequential control system on the AS has links with the basic control in the form of action and transition functions. If the controller also requires parameters, there are also links to the parameter data.

Each SFC has a certain run behavior assigned to it. The basic control with the blocks placed in the CFC charts can execute differently from the SFC itself. This can be controlled by placing the external view of an SFC chart in the CFC chart. In the same way, an SFC instance is controlled by the blocks placed in the CFC chart.
The structure of the run-time system allows the sequential control system and the blocks of the basic control to run in different cycles to reduce the cycle load. In the same way, SFCs can be installed in run-time groups and can then have different scan rates/phase offsets.

3.2 How the SFC Runs

3.2.1 [S7] Mode

The mode decides whether the running of the program is controlled by the operator or is automatically controlled by the AS program.

The following modes are possible for an SFC chart:

- **AUTO** (process mode)
  Control is automatic. The step control modes “T” (SCT input) and “T / T and C” (input SCT_TAC) can be set. Control by the AS program is implemented in the CFC chart by suitable assignment of parameter values or interconnection of inputs of the external view of the SFC chart or SFC instance.

- **Manual** (operator mode)
  Execution of the chart is controlled manually by an operator (for example during commissioning or with SFV). All step control modes are permitted.

Enabling the changeover to AUTO can also be set in the external view or in the SFC instance; in other words, AUTO can be enabled explicitly by the operator per SFC with a central command or by the program.

3.2.2 [S7] Step Control Mode

The way in which control is passed on by the transitions can be specified with various step control modes. It is possible to change the step control modes in all operating modes. The individual step control modes are mutually exclusive.

Table 3-1: Step Control Modes

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Step control mode...</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>Transition</td>
</tr>
<tr>
<td>C</td>
<td>Confirmation by operator</td>
</tr>
<tr>
<td>T and C</td>
<td>Transition and confirmation by operator</td>
</tr>
<tr>
<td>T or C</td>
<td>Transition or confirmation by operator</td>
</tr>
<tr>
<td>T / T and C</td>
<td>Step-specific confirmation by the operator</td>
</tr>
</tbody>
</table>

For a more detailed description of the step control modes, refer to the online help of SFC.

**Note**

In the step control modes “C” and “T or C”, the minimum run time of the step can be overridden by the operator.
3.2.3 Chart Execution Options

With the chart execution options (for example cyclic operation, time monitoring etc.), you can specify how the sequential control system executes. The individual execution options can be combined: The following chart execution options are available and can be changed in the "SFC > Properties: Operating Parameters AS" tab:

- If "cyclic operation: on" is set, execution continues automatically with "Starting" following on from "Completed".
  A sequencer to be processed in the "Completed" state is exited immediately in cyclic operation; in other words only the initial and final step are processed.
  If "cyclic operation: off" is set, the sequential control system remains in the "completed" state.
  As long as no command to exit the state is received, "completed" continues to be processed cyclically.
  Note: This applies to all states that can only be exited as a result of a command.

- With "time monitoring: on", after a step has been activated, the active time of the step is continuously compared with the monitoring time (object properties of the step, "General" tab, option "Maximum run time") and a step error is reported if the time is exceeded.
  With "time monitoring: off", the active time and monitoring time are not compared.

- An SFC with the "Autostart: on" property starts immediately without any further operator input after a CPU restart; in other words, it changes to the "starting" state. After downloading changes (CPU does not change to STOP), there is no automatic start and the SFC must be started again by the operator or the program (for example, external view in CFC).
  The setting for "Autostart" can only be changed in the "Chart > Properties" dialog in the "Operating Parameters AS" tab.

- With "Use default operating parameters when SFC starts: on", all the default settings / options set in the run-time properties become effective again when the SFC chart is started. The options may, for example, have been changed in the test mode.
  The defaults for the execution options are listed in Section 3.3.
3.3 How the Sequential Control System Behaves during Operation

What Determines How a Sequential Control System Behaves?

The behavior of a sequential control system depends on the operating parameters: operating state, mode, step control mode, and the chart execution options.

You can set the operating parameters when testing and commissioning or at the OS with SFC Visualization (except for the "Autostart" parameter that can only be set in the Properties dialog for the SFC chart or SFC type in the "Operating Parameters AS" tab).

The defaults of the operating parameters are as follows:

- Operating mode: Manual (operator mode)
- Step control mode: T (process-controlled)
- Chart Execution Options:
  - Command output: on
  - Cyclic operation: off
  - Time monitoring: off
- Start up Options:
  - Autostart: off
  - Use default operating parameters when SFC starts: off

The following operating mode is set after the CPU has started up:
- Ready (when Autostart = off)
- Starting (when Autostart = on)
3.3.1 [S7] The Operating States

How is the operating state influenced?

- In the test mode with the commands.
- In the process mode (AUTO) by interconnections with the external view of the SFC chart and parameter settings from a higher-level control program, for example, SIMATIC BATCH.

What does the operating state indicate?

The operating state of the sequential control system indicates the current state and, for example, whether operator intervention is necessary for continued operation or which commands are possible to change to a different operating state.

What are the tasks of the operating state logic?

The operating state logic of SFC (SFC-OSL) describes

- the states that can be adopted by an SFC chart or an SFC instance.
- the state changes possible in a particular state.
- the events that bring about a state change.

In addition to the SFC-OSL, there is also a separate simpler operating state logic for the sequencers configured in an SFC, the sequencer OSL. This describes

- the states that can be adopted by a sequencer.
- the state changes possible in a particular state.
- the events that bring about a state change.

The relationship between SFC-OSL and sequencer OSL results from the possibility of being able to execute one (of several) sequencers of the SFC in every state of the SFC-OSL.
3.3.2 Operating State Logic for SFC (SFC-OSL)

The current state of the SFC-OSL can be changed by the following events:

- Commands (Start, Resume, Hold, ...) in the "MANUAL" or "AUTO" modes.
- External signals (inputs of the SFC, commands from another SFC, ...).
- Internal signals (command from own sequencers, from the test mode or SFC Visualization).
- Implicit state change.

The operating state logic of an SFC is defined by the diagram of state changes:

![Figure 3-1: Diagram of State Changes for SFC-OSL](image)

The numbers in the diagram are identifiers for the individual states whose significance is explained in the following table.
Operating States (SFC-OSL)

Table 3-2: Operating States: SFC

<table>
<thead>
<tr>
<th>No.</th>
<th>Status</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ready</td>
<td>Basic state; waiting for start command.</td>
</tr>
<tr>
<td>2</td>
<td>Starting</td>
<td>Processing start after start command.</td>
</tr>
<tr>
<td>3</td>
<td>Active</td>
<td>Normal execution after completing startup.</td>
</tr>
<tr>
<td>4</td>
<td>Completing</td>
<td>Completing execution after the complete command or implicit completion.</td>
</tr>
<tr>
<td>5</td>
<td>Error (completing)</td>
<td>Error processing during completion.</td>
</tr>
<tr>
<td>6</td>
<td>Completed</td>
<td>Final processing completed; waiting for reset or start command.</td>
</tr>
<tr>
<td>7</td>
<td>Holding</td>
<td>Processing hold after hold command.</td>
</tr>
<tr>
<td>8</td>
<td>Held</td>
<td>Hold processing completed; waiting for resume command.</td>
</tr>
<tr>
<td>9</td>
<td>Resuming</td>
<td>Processing resume after resume command.</td>
</tr>
<tr>
<td>10</td>
<td>Error</td>
<td>Error processing after error.</td>
</tr>
<tr>
<td>11</td>
<td>Held (error)</td>
<td>Error processing completed and error no longer present; waiting for resume command.</td>
</tr>
<tr>
<td>12</td>
<td>Resuming (error)</td>
<td>Processing resume after resume command.</td>
</tr>
<tr>
<td>13</td>
<td>Aborting</td>
<td>Processing abort after abort command.</td>
</tr>
<tr>
<td>14</td>
<td>Aborted</td>
<td>Abort processing completed; waiting for reset or start command.</td>
</tr>
<tr>
<td>15</td>
<td>Stopping</td>
<td>Processing stop after stop command.</td>
</tr>
<tr>
<td>16</td>
<td>Stopped</td>
<td>Stop processing completed; waiting for reset command.</td>
</tr>
</tbody>
</table>

For more information on the state changes and state control via SFC I/Os, refer to the SFC online help, topic "Operating State Logic for SFC".
3.3.3 Operating State Logic for Sequencers (sequencer OSL)

The processing of sequencers is controlled by the sequencer OSL.

The sequencer OSL is run through independent of the SFC-OSL when a sequencer executes. This means that the sequencer has a state that must be distinguished from the state of the SFC. The SFC-OSL can, for example, be in the “holding” state, whereas (due to the processing of the sequencer for the "holding" state), the sequencer OSL is in the “active” state. The processing of the sequencer OSL is subject to the SFC OSL so that state changes in the SFC OSL usually mean a state change in the sequencer OSL.

The operating state logic of the sequencer is defined by the diagram of state changes for the sequencer OSL:

![Diagram of State Changes for Sequencer OSL]

Legend:
- States exited due to commands
- Operator commands
- Implicit state changes triggered by SFC

Figure 3-2: Diagram of State Changes for Sequencer OSL
## The states of the sequencer OSL

Table 3-3: Operating States: Sequencers

<table>
<thead>
<tr>
<th>No.</th>
<th>Status</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ready</td>
<td>Basic state; waiting for start command.</td>
</tr>
<tr>
<td>2</td>
<td>Active</td>
<td>Normal execution.</td>
</tr>
<tr>
<td>3</td>
<td>Completed</td>
<td>Normal execution completed; waiting for reset or start command.</td>
</tr>
<tr>
<td>4</td>
<td>Held</td>
<td>Hold processing completed; waiting for resume command.</td>
</tr>
<tr>
<td>5</td>
<td>Aborted</td>
<td>Abort processing completed; waiting for reset or start command.</td>
</tr>
</tbody>
</table>
3.3.4 Commands

With the following commands, you can set or modify the operating states of the SFC (in the test mode or in SFC Visualization):

All commands are permitted both in the "AUTO" and in the "MANUAL" mode; in the "MANUAL" mode, they do, however, depend on a command enable.

If several commands are pending at the same time (for example, external and internal commands), an interconnection error is displayed (LI_ERR=1). How the commands are then processed, is described in the SFC online help, topic "Priority of the Commands".

Table 3-4: Commands for the SFC

<table>
<thead>
<tr>
<th>Button</th>
<th>Command</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>🔴</td>
<td>Start</td>
<td>Trigger start processing by changing to the &quot;Starting&quot; state.</td>
</tr>
<tr>
<td>⏸️</td>
<td>Hold</td>
<td>Trigger hold processing by changing to the &quot;Holding&quot; state.</td>
</tr>
<tr>
<td>🔶</td>
<td>Resume</td>
<td>Trigger resume processing by changing to the &quot;Resuming&quot; or &quot;Resuming (error)&quot; state.</td>
</tr>
<tr>
<td>✗</td>
<td>Abort</td>
<td>Trigger abort processing by changing to the &quot;Aborting&quot; state.</td>
</tr>
<tr>
<td>✅</td>
<td>Complete</td>
<td>Trigger complete processing by changing to the &quot;Completing&quot; state.</td>
</tr>
<tr>
<td>🔴</td>
<td>Stop</td>
<td>Trigger stop processing by changing to the &quot;Stopping&quot; state.</td>
</tr>
<tr>
<td>🔶</td>
<td>Restart</td>
<td>Trigger start processing by changing to the &quot;Starting&quot; state.</td>
</tr>
<tr>
<td>✗</td>
<td>Reset</td>
<td>Change to the &quot;Ready&quot; state.</td>
</tr>
<tr>
<td>✗</td>
<td>Error</td>
<td>Trigger error processing by changing to the &quot;Error&quot; or &quot;Error (completing)&quot; state.</td>
</tr>
</tbody>
</table>

Table 3-5: Commands for the Sequencer

<table>
<thead>
<tr>
<th>Command</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start / Resume</td>
<td>Trigger or resume sequencer execution by changing to the &quot;Active&quot; state.</td>
</tr>
<tr>
<td>Hold</td>
<td>Hold sequencer execution by changing to the &quot;Held&quot; state.</td>
</tr>
<tr>
<td>Restart</td>
<td>Restart sequencer execution by changing to the &quot;Active&quot; state.</td>
</tr>
<tr>
<td>Abort</td>
<td>Abort sequencer execution by changing to the &quot;Aborted&quot; state.</td>
</tr>
</tbody>
</table>
3.3.5 Sequencer Execution

In the following description of the basic principles of cyclic execution of an SFC, it is assumed that the SFC contains several sequencers with freely defined start conditions.

Sequence

The execution of the SFC depends on the input signals applied to the SFC interface and the internal status of the SFC. After downloading a chart folder to the programmable controller, all the SFCs are in the "Ready" state.

- The mode, commands, and external and internal signals are queried to establish how the SFC will execute, in other words, which state the SFC should adopt based on the input signals.
- All the start conditions of the sequencers are checked to determine which sequencer will execute on the basis of the checks made up to now.

SFC State Unchanged

If the SFC state is unchanged, the sequencer that will execute is the one whose start condition is satisfied and that has the highest priority of all sequencers with a satisfied start condition. If there are several sequencers with a satisfied start condition and the same priority, then (analogous to the execution of alternative sequences) the sequencer with the tab furthest left in the editor will execute. If the sequencer that was executing is different from the newly selected sequencer, the previous sequencer aborts and the new sequencer starts.

The start condition triggers the start of a sequencer. While the sequencer executes, it is not necessary for the start condition to remain satisfied. The sequencer executes until it is completed or until a sequencer with a higher priority and satisfied start condition needs to execute or until the sequencer is aborted or held due to a status change.

In the "Ready", "Completed", "Aborted", "Stopped", "Held", "Held (error)", "Error" and "Active" states (when SELFCOMP=0), sequencers execute until the state is exited due to a command. As a result, a sequencer is repeated any number of times if the start condition is satisfied and there is no command applied. If you want to avoid this, the last transition of the sequencer must be configured with an unsatisfied condition. As a result, the sequencer "hangs" at this transition and this state is exited only with a command.
SFC State Changed

If the SFC state has changed, the SFC executes the previous and the new sequencer depending on the state change that occurred, as follows:

If the state change is from "active" to "holding", the previous sequencer is held or aborted (depending on RUNHOLD) and the new sequencer starts.

There is a state change from "resume" or "resuming (error)" to "active" when the previous sequencer has executed to completion. The new sequencer resumes or starts at the change from "resuming" (depending on RUNHOLD) and starts at the change from "resuming (error)".

If there is an implicit state change, the change of sequencer takes place when the first sequencer is completed. If there is no sequencer with a satisfied start condition, the implicit change is made immediately and the new sequencer is started.

With all other state changes resulting from commands, external or internal signals, the previous sequencer aborts if it has not yet completed and the new sequencer starts.

Notes

- When a sequencer aborts, the active steps always execute completely and then the final step executes. The completion of the active steps and the initialization or execution of the final step take place in one cycle.

- In an SFC, a sequencer can execute even if the SFC has not received a start command. This is the situation, for example, when the start condition of one or more sequencers is satisfied in the "Ready" state or when the start condition is not dependent on states.

Target Step for the Sequencer

the start response of a sequencer that will become active in the future can be influenced by setting a target step for this sequencer in step actions of the currently active sequencer (SFC.TARGETSEQ:=2; SFC.TARGETSTEP:=5). This corresponds to setting a target step manually during testing and commissioning, the difference in this case is, however, that all other target steps including those set manually are reset. This means that the required initial step is specified for the sequencer that will start later.

Please note that target steps cannot be set in simultaneous sequences.

A target step is taken into account when a sequencer starts or resumes and is then deleted.

Since the required execution and also the target step generally depend on the previous execution of the SFC, the last active sequencer and the last active step can be checked in a transition condition (SFC.LASTSEQ:=3; SFC.LASTSTEP:=2) to allow you to set different target steps depending on how the SFC executed prior to this.

Sequencers and steps are identified by their numbers. These are displayed in the properties dialogs and can be used to configure target steps.
[S7] The Influence of the Control Strategy

A change in the control strategy is permitted in all states. The state is retained but the current sequencer is aborted and this is followed by a check to determine whether a different sequencer should execute. This means that not only a state change but also a change in the control strategy can influence how an SFC is executed.

The processing of the state and control strategy change is handled by the operating state logic (OSL).

Summary of the Execution of a Sequencer

If you want to put together the processing of the "active" state and the "holding", "held", and "resuming" states as an alternative sequence in a sequencer, the RUNHOLD input of the SFC must be set to TRUE. If RUNHOLD is set to FALSE, the mechanism for the sequencer change would not function (a sequencer cannot be held and then (re)started and then resumed).
3.3.6 Starting an SFC (chart or instance)

An SFC starts only when the constraints are satisfied. The start enable must be set (ENSTART=1) and the SFC must be in a state in which starting is permitted. The following conditions must also be met:

- There is no interconnection error (LI_ERR=1).
- None of the signals INTERROR, LOCKERROR, LOCKCOMPLETE, LOCKHOLD, LOCKABORT or LOCKSTOP is active at the same time.
- There is no operator error (OP_ERR=1) in the MANUAL mode.

For an SFC instance, when using control strategies, one of the defined control strategies (CS=<defined control strategy>) must be selected. If CS==0 or CS>CS_HL, it is not possible to start the SFC instance. If no control strategies are configured, the I/Os CS, CS_LL, CS_HL are assigned the value 0 during compilation and must not be modified.

For an SFC instance, prior to starting in the AUTO mode, parameter transfer (PARAM=1) must be set. The transferred setpoints are then checked and the start disable is reset (QDIS_START=0) if the parameters could be accepted. The start disable remains set as long as no parameters have been transferred by setting PARAM. Following a successful start, the start disable is set until the next parameter transfer.

If one of the conditions listed above is not satisfied, the start disable (QDIS_START=1) is set and the start is not executed and the "Start" button is inoperable.

Starting in the "Active" State

To be able to start in the "active" state, the additional start enable must be set (ENASTART=1) and self completion must be deactivated (SELFCOMP=0). In the AUTO mode, continuous operation must also be activated (CONT=1).

The start condition of the sequencer required to execute in the "active" state must also make sure that the sequencer has not already executed to completion (READY_TC=0). This prevents the sequencer being repeated cyclically when the start condition is satisfied.

When an SFC starts, the ready identifier is reset (READY_TC=0) and is set again (READY_TC=1) on completion of the sequencer executing in the "active" state.

An example of the start condition would then be: RUN==TRUE AND READY_TC==FALSE.
3.3.7 Execution of a Sequential Control System

Phases of a Step

Each step is divided into three phases (actions):

- Initialization is the action for first-time execution
- Execution is the action for cyclic execution
- Termination is the action for last execution

The schematic shows the phases of a step in conjunction with a successor transition: on the left the elements in the sequencer topology, on the right, the corresponding phases.

![Phases of a Step in Conjunction with a Follow-on Transition](image)

Figure 3-3: Phases of a Step in Conjunction with a Follow-on Transition

Processing Steps and Transitions

The initial step is activated when the chart is started without querying conditions and its actions are executed.

A (normal) step can have the states “active” and “inactive”. A step becomes active after the preceding transition passes control to it. The actions are then triggered and executed. A step becomes inactive after the successor transition is satisfied.

If there is an abort, the termination processing of the previously active step is executed and the initialization of the final step is started (overlapping in the same cycle).
A transition has the states “FALSE” and “TRUE”. The state of the successor transition of the active step is queried. If the successor transition is true (the condition is true), the previous step is deactivated and the next step activated. If a minimum run time is configured, the transition is queried depending on the step control mode only after this time has elapsed.

The actions of the final step are executed once only.

![Diagram](image)

Figure 3-4: Time Response in Executing Steps and Transitions in a Sequential Control System

When the chart progresses from one step to the next, the predecessor step is terminated in the same cycle as the first action (initialization or execution) of the next step. This allows the “non-latching behavior” specified in IEC 1131 - 3 to be achieved.

**Example:** In step S4, the execution opens a valve and in the termination phase this is closed again. If the same valve is opened again in the first action of the next step (S5), the overlapping of the two actions (both in one cycle) means that the valve is not closed.
Special Situations

The example in Figure 3-4 shows the response when all three actions of a step are configured.

Other combinations are also possible:

- If no “initialization” is configured, the execution begins immediately when the step is activated.
- If no “termination” is configured, the step is deactivated immediately when the transition is true.

The minimum time in which a step is active depends on the number of configured actions (for normal steps 1 to 2 processing phases, with the final step up to 3). If a minimum run time is set for a step, the step remains active for at least this time even if the transition condition is satisfied earlier.

Processing a Simultaneous Sequence

The simultaneous sequence paths are executed more or less simultaneously in one cycle (in other words, the entire simultaneous sequence). The simultaneous sequence paths are executed independent of each other.

The transition after the simultaneous sequence becomes true when all steps at the end of the sequence paths are active and the conditions are satisfied.

Processing an Alternative Sequence

The path of an alternative sequence that is executed is the sequence with the transition whose condition is true first.

If several conditions are true at the same time, the transition furthest left in the chart topology is activated.

Note:
In alternative sequences, there must not be no transition without parameters at the start of a sequence path. Reason: Transitions without parameters are always TRUE and are therefore automatically satisfied. This means that they are always true before a transition with parameters.
**Processing a Loop**

Figure 3-5 illustrates the phases of a loop: To the left the elements in the chart topology, to the right the corresponding phases.

![Figure 3-5: Phases of a Loop](image)

**Processing a Jump**

The jump is executed when the transition of the jump is satisfied.

![Figure 3-6: Example: Jumps from a Successor Transition and an Alternative Sequence](image)

If there is more than one jump following the origin of a jump (S4), then (just as in alternative sequences), the jump whose transition is satisfied first is executed. If several transitions are true at the same time, the transition furthest left is activated. If the next element in the sequence is not a successor transition (such as T4 on the left in Figure 3-6 but an alternative sequence, all the transitions of the alternative paths (T5 and T7) are queried before the transitions of the jumps.
4 Test and Startup

Overview
To support you when you put a new program into operation, the SFC editor provides test functions that allow you to monitor the operation of the charts on the CPU, to control the running of the chart, and to change settings.
This chapter describes how to monitor and control the sequential control system when testing.

4.1 Activating the Test

Requirements for Testing
The sequential control system (chart or instance) to be tested and the basic control functions (CFC charts) must be loaded on the CPU.
The chart is opened in SFC or the SIMATIC Manager, the SFC instance in the CFC chart.

[S7] The Modes for Testing
Before you switch over to testing, you can select between the "process mode" and the "laboratory mode".
In the process mode, the communication for online dynamic display of the SFCs is restricted and causes only limited extra load on the CP and bus. If overload occurs in this test mode, a message is displayed to indicate the limit of bus load has been reached. In this case, you should stop the testing of the SFCs that are not absolutely necessary for the test.
The laboratory mode allows convenient and efficient testing and commissioning. In the laboratory mode, in contrast to the process mode, communication for online dynamic display of SFCs is unrestricted.
You set the mode for testing with the menu commands “Debug > Process Mode” or “Debug > Laboratory Mode”.

Switching over to the Test Mode

You change from the edit mode to the test mode by clicking the button \( \text{ } \) in the toolbar or by selecting the “Debug > Test Mode” menu command. During the Test mode, you can change to the Edit mode at any time.

The mode change relates to the currently active SFC. This SFC is displayed dynamically in its overview and is cyclically updated (you can set the watching cycle with the “Debug > Test Settings...” menu command.)

**Note:**
If you change the test settings, you change the watching cycle for all charts of this CPU.

After you change to the Test mode, the current state of the chart is displayed. This means that a sequential control system that has already started cannot always be monitored or controlled from the start. This is, for example, the case with charts that are started immediately after they are downloaded to the CPU without any operator command being required (autostart).

**[S7] Note on H CPUs:** If the H CPU is in the solo mode, for example after the failure of a CPU and there was a CPU failover, an online access (in this case, download) results in a dialog being displayed. In this dialog, you can select the required CPU. In the redundant mode, this dialog does not appear.
4.2 Display in the Test Mode

The window of the SFC chart or SFC instance in the test mode has an extra operator input and display section at the bottom edge of the window compared with the edit mode. All operator input relates to the chart or the instance (not to the sequencer).

Figure 4-1: SFC Window in the Test Mode (without element bar and toolbar)

The following elements are displayed (from left to right):

- in the fields of the first row:
  - Symbol and name of the SFC state
  - State display and name of the active sequencer
  - State display and name of the held sequencer
  - Field with drop-down list box for displaying and modifying the step control modes
• in the fields of the second row:
  - The current mode (MANUAL / AUTO)
  - "CONTINUOUS" state flag for smooth switchover (for example, in SIMATIC BATCH to avoid having to turn off the SFC being used when changing from one recipe operation to another. Displayed when the CONT input is set to 1.
  - Status display "READY T.C." when the SFC is not self terminating (SELFCOMP = 0) and waits for the "complete" command in the active state.
  - The display 🔄 for an interconnection error (or empty field)
  - The display 🟢 for an operator error (or empty field)
  - The display 🟥 for a step error (or empty field)
  - The display 🟦 for an operator prompt (or empty field)
  - The Confirm All button 🟢

• The Buttons
  - for selecting the "MANUAL" or "AUTO" mode
  - for enabling the switchover to "AUTO" 🟢.
    When this has been enabled, the button changes to 🟢.

• The buttons for the commands:
  - Start 🎁
  - Hold 🎁
  - Resume 🎁
  - Abort 🎁
  - Complete 🎁
  - Stop 🎁
  - Restart 🎁
  - Reset 🎁
  - Error 🎁

• The check boxes for activating and deactivating the execution options "command output", "cyclic operation", "time monitoring".

Operator Prompt
The operator prompt (not with step control mode "T") is displayed in a box beside the transition icon.

After clicking on the field (or 🟢) and continuing execution, the operator prompt disappears again.
Displaying the States

In the test mode of the SFC, both the SFC and the sequencers with their start conditions are made dynamic. The current states of the SFC and sequencer are displayed in the operator input and display section of the SFC (see above).

The various states of the sequential control system, the steps, the transitions are visualized by different colors and icons.

In addition to the color frame of the steps, a status display also appears. The icon of the status display is therefore an additional indicator of the current states in the chart in case the colors cannot be clearly distinguished. The colors of the state displays cannot be changed.

Table 4-1: Appearance of the State Displays

<table>
<thead>
<tr>
<th>Step state</th>
<th>Step color</th>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>inactive (not executed)</td>
<td>gray</td>
<td></td>
</tr>
<tr>
<td>inactive (executed)</td>
<td>dark green</td>
<td>✓</td>
</tr>
<tr>
<td>active</td>
<td>light green</td>
<td>▲</td>
</tr>
<tr>
<td>held</td>
<td>yellow</td>
<td>II</td>
</tr>
<tr>
<td>error</td>
<td>red</td>
<td>❌</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Transition State</th>
<th>Transition Color</th>
</tr>
</thead>
<tbody>
<tr>
<td>inactive</td>
<td>gray</td>
</tr>
<tr>
<td>satisfied</td>
<td>dark red</td>
</tr>
<tr>
<td>not satisfied</td>
<td>dark green</td>
</tr>
</tbody>
</table>

Display of the CPU mode

The operating mode of the CPU is displayed in the status bar (right information box) in color (green = RUN, red = STOP). The test mode (Laboratory) or (Process) is also indicated.
4.3 Operator Control and Monitoring of the Sequential Control System

Setting the Test Environment

With the "Debug > Test Settings..." menu command, you can open a dialog box in which you can change the watching cycle for the current program (default 2 s). You can set cycles from 1 to 9 and in steps of ten from 10 to 90 seconds.

The same cycle time as for CFC is stored in the chart folder. This means that the same monitoring cycle is valid for both applications (SFC and CFC).

Operator Control and Monitoring

You operate and monitor the SFC in the overview display. Here, you can change the statuses, modes, step control modes, and the chart options as required.

This always relates to the sequencer in the foreground selected by its tab; in other words, there is no implicit changeover to the current sequencer.

If you double-click on a step or a transition, a dialog box appears resembling the object properties dialog in the Edit mode. You can also display the object properties for step and transition at the same time by selecting the required transition and opening both dialog boxes by double-clicking a step (or the other way round by selecting a step and double-clicking the transition). To be able to see both dialog boxes at the same time, the step and transition do not need to belong together.

A selected element is indicated by a blue background.

Confirming in the Sequencer and in the Object Properties Dialog

If a button is displayed for operator intervention or to confirm an error for the monitored step or transition in the sequence path, the dialog box with the object properties also has the corresponding button or buttons added to it.

Note: After a step run-time error has been acknowledged, the appearance of the step returns to the state before the error occurred (for example active = “light green”).

Changes during a Test Run  (SFC chart only, not SFC instance)

The step attributes (confirmation, minimum/maximum run time, constant in assignments) and transition attributes (constant in conditions) that can be modified in the Test mode are adopted on the AS and in the ES data management when the change is made and do not require recompilation and download.

Note: If you want to modify the existing SFC instances, you must exit the test mode and then modify the corresponding SFC type. After compiling the changes and downloading them, all the instances are automatically adapted.
4.3.1 The Object Properties of a Step during Testing

The Properties dialog is divided into four parts. These are the tabs: “General”, “Initialization”, “Processing” and “Termination”.

- **“General” Tab**
  
  The “Name” box in the “General” tab is in a frame, the frame color corresponds to the operating state of the step and is updated constantly (colors: see table in “Default Colors” in the online help).

  You can activate or deactivate the “Confirmation” option and which sets or resets a flag that is evaluated in the “T / T and C” step control mode (step-specific confirmation by user). The sequential control system operates as follows:

  - **process-controlled** for steps without the “Confirmation” option.
    
    Each satisfied successor transition of a step without the “Confirmation” option passes control without operator intervention (corresponds to “T”).
  
  - **operator-controlled** for steps with the “Confirmation” option.
    
    If the successor transition of an active step with the “Confirmation” option is satisfied, an operator prompt is set and control passes to the next step or steps after the prompt has been confirmed (corresponds to “T and C”).

[S7] By setting the “Target step” option, the current step is selected as the target step. This means that

  - The stopped sequencer starts at the selected target step instead of at the initial step with the next "Start" command.
  
  - the held sequencer resumes at the target step after correct processing of the interrupted steps when the “Continue” command is set.

[S7] The target step marker is valid only for the next “Start” or “Continue” command. When you restart the CPU, the target step marker is deleted.

You can also select several steps as the target steps. The user is responsible for selecting the target steps so that the processing is feasible; in other word, that no blockages or endless loops result. See also Section 3.3.5, Editing Sequencers.

If you use "programmed target steps", the target steps sent by the operator in the corresponding sequencers are deleted.

You can change the values of “Run Times Minimum” and “Run Times Maximum”. By clicking in the input box, you open a dialog in which you can change the current setting. Any changes you make are entered in the ES data management when you confirm with OK and take effect in the next processing cycle.

With the boxes “Run Times - Current”, “Time to minimum” and “Time to maximum” you can monitor the run time.

If no values were configured for the run times (time = 0), then “- - -” is displayed in the boxes.

- **“Initialization, Processing, Termination” Tabs**

  The current value of the 1st address is shown in a box to its left. The field to
the right beside the 2nd address contains the current setpoint that you can modify (SFC chart only). If you click the box, you open the “Change Value” dialog box, in which you can enter the new value.

After you close the dialog box, the changed value is written to the ES data management and becomes effective in the next processing cycle.

If the monitoring of the step detects a time error, the confirm button is added to the buttons. This allows you to acknowledge the error from the dialog box.

After a step run-time error has been acknowledged, the appearance of the step returns to the state before the error occurred (for example active = “light green”). Using the “Go To” button, you can jump from the current field of an operand to its point of use (block in the CFC chart, SFC chart, I/O address in HW Config.).
4.3.2 The Object Properties of a Transition during Testing

The Properties dialog of a transition is divided into four parts. These are as follows:

- **General**
  The “Name” box in the “General” tab is in a frame, the frame color corresponds to the operating state of the step or transition and is updated constantly (colors: refer to the table in “Default Colors” in the online help).

- **The current condition (Current Cond.)**
  shows the current state of the conditions

- **The last condition ("Current Cond. -1")**
  shows the state of the condition of the previous execution cycle.

- **The condition after error ("Cond.after Error")**
  shows the state of the condition that led to an error.

**Changing Addresses**

To the left of the first address and to the right of the second address there are fields with the current value of the address. In the SFC chart, you can change the content of both fields (not possible with an SFC instance). If you click one of the boxes, the “Change Value” dialog box is displayed in which you can enter a new value for the address.

After you close the dialog box, the changed value is written to the ES data management and becomes effective in the next processing cycle.

If there is an operator prompt for the monitored transition in the sequencer and the button displayed beside the transition, the dialog box with the object properties also has this button added to it. This allows you to acknowledge the operator prompt from the dialog box.

Using the “Go To” button, you can jump from the current field of an operand to its point of use (block in the CFC chart, SFC chart, I/O address in HW Config.).

**Status of the Logic Operation**

The results of the logic operations of the conditions using Boolean operators are visualized as colored links of varying thickness. A broad, green line means “satisfied”, a thin, red line means “not satisfied” and a thin, black line means “inactive”.

**Update**

The content is updated continuously in the “Current Cond.” tab. In the “Current Cond. -1” and “Cond. after Error” tabs also include the “Update” button. The state of the transition is entered here as it was at the time when the object properties were opened. Using the “Update” function, you can display the current state in a permanently open dialog box.
Test and Startup
5 Documentation

Overview

This chapter describes how you create the documentation for your SFC charts/types/instances. This includes printing the SFC in various views and printing the parameter assignment, the properties, and chart reference data.

5.1 Printing an SFC

You can print the current chart or type by clicking the button in the toolbar or using the "SFC > Print..." menu command. In the dialog box that opens, you can make settings relating to the scope of the printout (Print What) and the layout (Options).

Print What: properties, I/Os, external view (with SFC chart) or alternatively characteristics (with SFC type/instance). For sequencers: properties, normal size, overview, steps/transitions. You can select either normal size or overview, but not both.

• Options (only for "normal size"): alternative sequence left-aligned, comment/text.

Note:
The selected range is always printed for the entire SFC. If you only require a printout of a step or a transition, use the print functions of the in the “Object Properties” dialog of the selected element.
5.2 Defining Footers

With the SFC > Footers... menu command, you can open a dialog box in which you enter the text printed in the footers of every printed page.

With the DOCPRO optional package, you can print the SFC chart/type along with footer data. In the footer data, a distinction is made between global data and local data specific to an SFC.

You can enter the global data for the project using DOCPRO or the SIMATIC Manager; the local specific data must be entered with the SFC editor. Remember that the data specific to an SFC overwrites the entries of the global data for the particular SFC.

You can also enter specific data even when you have not installed the DOCPRO optional package. This data is saved and can be printed later when DOCPRO is available for print jobs.

You can enter the SFC-specific footer data in the active fields of the tabs "Part 1" to "Part 4" and "Free fields". These include the type of document, date created, document number, dates modified, free texts etc.

In the global footers, you can enter keywords that are replaced by the actual texts when you print out. The keywords that can be used and their meaning (origin) are explained in the online help of SFC.

5.3 Chart Reference Data

With the "Options > Chart Reference Data..." menu command or by clicking, you start an application that creates comprehensive documentation listing, for example, the run sequence, the cross-references of the addresses etc.

With a search function, you can search for specified addresses, symbols, I/Os etc. This helps you to check your configuration structure. With the "Cross-References Chart Element -> Address" list, you can, for example, check which addresses are used and how often and whether the write access is synchronized.

You do not need to close the window with the chart reference data if you want to continue working with the SFC editor. This means that you can view the lists while working with the SFC editor.

For a detailed description of the “Chart Reference Data”, refer to the CFC online help.
A Technical Specifications

A.1 Technical Specifications

Hardware Requirements
- PG or PC with:
  - Pentium Processor
  - RAM 256 Mbytes (or more)
  - Hard disk 500 Mbytes (less RAM)
  - Graphics card VGA 640 x 480
    (recommended: SVGA 1024 x 768 or higher)
- SIMATIC S7-400

Software Requirements
- Microsoft Windows NT (SP 5) / Windows 2000 (SP 3)
- STEP 7
- SCL Compiler
- CFC
  An authorization for STEP 7, SCL, CFC and SFC must be installed.

General Numeric Data

<table>
<thead>
<tr>
<th>Object</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Steps per chart</td>
<td>2 - 255</td>
</tr>
<tr>
<td>Transitions per chart</td>
<td>1 - 255</td>
</tr>
<tr>
<td>Statements per step and action</td>
<td>• 50</td>
</tr>
<tr>
<td>Conditions per transition</td>
<td>• 16</td>
</tr>
<tr>
<td>Sequencers per SFC chart</td>
<td>• 8</td>
</tr>
<tr>
<td>Sequencers per SFC type</td>
<td>• 32</td>
</tr>
<tr>
<td>Steps per sequencer</td>
<td>2 - 255</td>
</tr>
<tr>
<td>Transitions per sequencer</td>
<td>1 - 255</td>
</tr>
</tbody>
</table>
## B Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>Step control mode: confirmation by operator</td>
</tr>
<tr>
<td>C / C++</td>
<td>High-level language for programming computers</td>
</tr>
<tr>
<td>CFC</td>
<td>Continuous Function Chart</td>
</tr>
<tr>
<td>CPU</td>
<td>Central processing unit</td>
</tr>
<tr>
<td>DB</td>
<td>Data block</td>
</tr>
<tr>
<td>ES</td>
<td>Engineering system</td>
</tr>
<tr>
<td>FB</td>
<td>Function block (function block with memory)</td>
</tr>
<tr>
<td>FC</td>
<td>Function (function block without memory)</td>
</tr>
<tr>
<td>HID</td>
<td>Higher level designation</td>
</tr>
<tr>
<td>IEA</td>
<td>Import-Export Assistant</td>
</tr>
<tr>
<td>OB</td>
<td>Organization block</td>
</tr>
<tr>
<td>OCM</td>
<td>Operator control and monitoring (WinCC)</td>
</tr>
<tr>
<td>OS</td>
<td>Operator station</td>
</tr>
<tr>
<td>PC</td>
<td>Personal computer</td>
</tr>
<tr>
<td>Abbreviations</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td><strong>PCS 7</strong></td>
<td>Process control system (SIMATIC)</td>
</tr>
<tr>
<td><strong>PG</strong></td>
<td>Programming device</td>
</tr>
<tr>
<td><strong>PH</strong></td>
<td>Plant hierarchy</td>
</tr>
<tr>
<td><strong>PLC</strong></td>
<td>Programmable controller</td>
</tr>
<tr>
<td><strong>PLC</strong></td>
<td>Programmable logic controller</td>
</tr>
<tr>
<td><strong>SFB</strong></td>
<td>System function block</td>
</tr>
<tr>
<td><strong>SFC</strong></td>
<td>Sequential Function Chart</td>
</tr>
<tr>
<td><strong>SFV</strong></td>
<td>SFC Visualization</td>
</tr>
<tr>
<td><strong>STEP 7</strong></td>
<td>Software development environment for SIMATIC S7 / M7</td>
</tr>
<tr>
<td><strong>T</strong></td>
<td>Step control mode: transition only</td>
</tr>
<tr>
<td><strong>T / T and C</strong></td>
<td>Step control mode: step-specific confirmation by operator</td>
</tr>
<tr>
<td><strong>T and C</strong></td>
<td>Step control mode: transition and confirmation by operator</td>
</tr>
<tr>
<td><strong>T or C</strong></td>
<td>Step control mode: transition or confirmation by operator</td>
</tr>
</tbody>
</table>
Glossary

A

Access
SFC elements or block I/Os can access addresses, charts, block I/Os, or run-time groups. A distinction is made between read and write access.

Action
Actions allow the activation or deactivation of run-time group and SFC charts, as well as modifications to blocks and shared resources and their input parameters. The actions are formulated in SFC in the Properties dialog.

Address
An address is the identifier of the location of data or an area containing data, examples: input I12.1; memory word MW25; data block DB3. Addresses can be specified in absolute or symbolic form.
An address is part of a STEP 7 instruction and tells the processor what it should perform an operation on.
In SFC, the address is part of a statement (step) or condition (transition).

Addressing, absolute
If data is addressed in absolute form, the absolute address is used to access the value with which the operation will be performed. Example: The address Q4.0 identifies bit 0 in byte 4 of the process output image (PIQ).

Addressing, symbolic
If data is addressed using symbols, the address to be processed is specified as a symbol (not as an absolute address). Symbols are assigned to addresses in the symbol table.

Alternative sequence
A structural element of an SFC, that consists of at least two sequence paths. Only the path whose transition condition is satisfied first is processed by the PLC.
Basic control

Basic control is the configuration of the programmable controller (PLC). The configuration tools available are the CFC and SFC editors.

Block

Blocks are separate parts of a user program that are distinguished by their function, their structure, or purpose.

CFC works with ready-made block types that are placed (inserted) in a CFC chart. When a block type is inserted in the chart, a block instance is created. These block instances and their graphic representation are blocks in the sense of CFC.

Block category

The block category identifies different forms of blocks. Block categories include, for example, data blocks (DBs), function blocks (FBs), and functions (FCs).

Block icon

Graphic representation of the most important information of a technological, block that can be controlled and monitored on the PLC. The block icon is typically placed in an OS overview picture. The corresponding faceplate can be called up using the block icon.

Block input

Block I/O that can be interconnected with block outputs and addresses of the same data type or can have parameters assigned to it.

Block instance

A block instance is the implementation of a block type. A block type inserted in a CFC chart becomes an instance. After it is inserted, the block instance has runtime properties and name that is unique within the chart.

Block I/O

Block input or block output

Block output

Block I/O that can be interconnected with block inputs and addresses of the same data type.
C

Catalog
Window in CFC, that can be switched between the catalogs of blocks, charts, libraries, and unplaced blocks (if unplaced blocks exist). You can insert blocks, charts and text boxes from the catalog into a chart.

CFC
Continuous Function Chart.
1. Function chart (CFC chart) with the graphic interconnection of technological functions (blocks).
2. An editor for plant/oriented, graphic configuration of automation tasks. Using CFC, entire software structures are created (CFC charts) from ready-made blocks.

Chart
Depending on the context, a chart in the ES is
- a CFC chart consisting of 1 to 26 chart partitions each with 6 sheets and possibly also overflow pages.
- a nested chart that results when one chart is inserted in another (chart-in-chart).
- an SFC chart consisting of between 1 and a maximum of 8 tabs. Each tab contains a sequencer.

Chart element
Chart elements of the SFC chart are the basic elements (step, transition, text) and structure elements (sequence, simultaneous sequence, alternative sequence, loop and jump).

Chart folder
Folder in the project structure containing charts of a user program.

Chart reference data
Chart reference data is data available to the user in addition to the graphic chart display in the form of lists, for example the list of accesses to shared addresses.

Component view
Device-oriented view in the SIMATIC Manager. The project is displayed with its components (station, module, program ... ); alternative to the plant view or process object view.
Consistency check

Checks the consistency of block types, shared addresses etc. of the chart folder.

Condition

Conditions in a transition allow you to do the following:

- to read values of block I/Os or shared addresses
- To logically combine the read values with a constant or another read value using Boolean operators (\(=\), \(>\), \(<\), ...)  

The result of a condition is a Boolean variable that can be logically combined with the results of other conditions.

Cycle time

The cycle time is the time required by the CPU to execute the user program once.

Data block (DB)

Data blocks are data areas in the user program containing user data. There are shared data blocks that can be accessed by all logic blocks and instance data blocks that are assigned to a particular FB call. In contrast to all other blocks, they contain no instructions.

Data type

A data type specifies how the value of a variable or constant is used at a block I/O. "BOOL", for example, defines a binary variable; "INT" defines a 16-bit integer variable.

Drag-and-drop

With the drag-and-drop function, you can move, copy and insert objects using the mouse.

Using drag and drop:

1. Select the objects by clicking them or by drawing a lasso round them.
2. Position the mouse pointer on an object, press the left mouse button and hold it pressed.
3. If you want to copy, hold down the CTRL key as well (the mouse pointer has the "+" symbol added to it).
4. Move the mouse pointer to the required position and release the button - the objects are inserted.
Dynamic display

In the dynamic display, input and output values of a block in a CFC chart or address values in an SFC chart are updated from the CPU in the Test mode.

E

Edit mode

(Alternative to the ↑ test mode)
In CFC blocks can be inserted, copied, moved, deleted, renamed, have parameter values assigned, or be interconnected.
In SFC the sequential control system is created in this mode. Chart elements can be inserted, copied, moved, deleted, renamed, and assigned parameters.

ES

Abbreviation for "Engineering System". This is a user-friendly, visual configuration system in which the process control system can be created and adapted to different situations.

External view

The external view is the graphic representation of a SFC chart as block (with interface) that allows external interconnection of the I/Os. The external view contains the standard interface that is derived from the SFC run-time system.

The external view of the SFC can be interconnected with any objects (CFC blocks, nested charts, run-time groups, textual references, shared addresses). This means that the SFC chart can also be controlled directly by CFC interconnections.

The external view is displayed in a separate window in the CFC. Other objects cannot be placed in this window. Interconnections to other CFC objects are implemented solely via the sheet bar.

F

Forced manual switchover

Switchover by the operator from the “AUTO” mode to the "MANUAL" mode without a prior enable.
Hierarchy folder

The hierarchy folder is used to structure the plant in a hierarchy. It can contain further hierarchy folders and objects such as CFC / SFC charts, process pictures, reports, additional documents (Excel, Word, ...). The HID of an object is obtained from the name of the hierarchy folder (path) and the object name.

Higher level designation (HID)

The HID is used to identify parts of a plant according to their functions. This is made up of the hierarchical path of the plant hierarchy.

Installation and startup

The CFC/SFC editors provide test functions to allow you to monitor, modify, and change parameter settings on the PLC during installation and startup.

Installing

Procedure with which a block/SFC chart is placed in a run-time group so that it is registered with an organization block (OB).

Instance

When you insert an SFC type or a block type in a CFC chart, an instance is created. Instance in this sense means that it is an instance or usage of the selected SFC type or block type.

Interconnection

In CFC, the connection of an I/O to another element. The value of the interconnected input is fetched from the other end of the interconnection during runtime. Read or write access from an SFC chart to a block I/O in a CFC chart is also known as an interconnection.

I/O

The input or output of a block or chart. I/Os of the same data type can be interconnected or connected to shared addresses. An I/O is parameter, that accepts data for further processing (input) and transfers the resulting data (output).
J

Jump

A jump is a structure element of SFC, with which the execution of an SFC can be continued at a different step in the same SFC depending on a transition condition.

L

Laboratory mode

One of the test modes. The laboratory mode allows convenient and efficient testing and commissioning. In the laboratory mode communication for online dynamic display of SFCs is unrestricted.

As an alternative to the laboratory mode, you can also select the ↑ process mode (restricted test mode). It is not possible to switch over between the process and laboratory modes when testing is active (only in the edit mode).

Library

A folder for objects that can be used again and again. A library is not project-related. Blocks are made available in block libraries sorted according to certain criteria (block families, alphabetical order etc.). Different block libraries are used depending on the PLC or situation.

Loop

Structure element in SFC consisting of a ↑ sequence and a return path; the sequence encloses and contains exactly one transition.

M

Master project

A master project is a project that has had parts of it (for example, charts) distributed into other projects (branch projects) to allow more efficient engineering. After they have been worked on, the parts are merged back into the original project = master project.

Messages, configuring

Creating messages with their attributes and texts. Messages can be configured from within SFC.
Mode

1. CPU:
Using the mode selector of the CPU, you can set the following operating modes:

- RUN with access to the STEP 7 user program,
  for example, with programming device ("RUN-P"),
- RUN with access protection ("RUN"),
- STOP and
- Memory Reset ("MRES").

2. SFC:
The mode decides how the execution of an SFC is controlled.

- AUTO (process mode): Execution is controlled automatically, for example, via the external view of an SFC chart.
- MANUAL (operator mode): Execution of the chart is controlled manually by an operator, for example during commissioning or with SFV (SFC Visualization in WinCC).

Online/offline

In the SIMATIC Manager, objects of the programmable controller are displayed in the online view and the objects of the ES in the offline view. Online, there is a data connection between the PLC and the programming device/PG, offline there is no connection.

Operating mode

1. The programmable controllers of SIMATIC S7 can adopt the following operating modes: STOP, STARTUP, RUN and HOLD.

2. SFC has the following operating states Ready, Starting, Active, Completing, Error (completing), Completed, Holding, Held, Continuing, Error, Held (error), Resuming (error), Aborting, Aborted, Stopping, Stopped.

Operating system

General term for all the functions that control and monitor the execution of the user programs, the distribution of resources to the individual user programs, and the maintenance of the operating mode in conjunction with the hardware (for example standard operating system MS-WINDOWS).
**Operating state logic (OSL)**

The operating state logic of SFC describes

- the states that can be adopted by an SFC chart or an SFC instance.
- the state changes possible in a particular state.
- the events that bring about a state change.

In addition to the SFC-OSL, there is also a separate simpler operating state logic for the sequencers, the sequencer OSL. This describes the states that can occur in a sequencer.

**Organization block (OB)**

Organization blocks form the interface between the operating system of the CPU and the user program. The order of execution of the user program is specified in organization blocks.

An organization block corresponds to a ↑ task.

**OS**

Operator Station. A station for controlling and monitoring the process. In PCS 7, the WinCC software system is used for the OS with which all the process monitoring and control functions can be implemented.

**P**

**Phase offset**

The phase offset shifts the point of activation of the run-time group within a task by a defined time compared with the basic cycle. Phase offset allows a uniform distribution of load within the CPU. See also ↑ scan rate.

**Plant Hierarchy (PH)**

A hierarchy structured according to technological aspects.

**Plant view**

View in the SIMATIC Manager based on technological aspects (plant (process cell), unit, function ...); alternative to the ↑ component view and ↑ process object view.
Postprocessing
A cyclic action can also be configured for each sequencer. The cyclic action consists of a part that is executed before the cyclic sequencer processing, the **preprocessing** and a part that is executed after the cyclic sequencer processing, the **postprocessing**.

Preprocessing
See ↑ Postprocessing.

Process
Sequence of chemical, physical, or biological processes for extraction, production or elimination of substances or products.

Process image
Reserved areas in the RAM of the CPU. Signal states of the input and output modules are entered here.

Process mode
One of the test modes. In the process mode, the communication for online dynamic display of the SFC charts is restricted and causes only limited extra load on the CP and bus.
As an alternative to the process mode, you can also select the ↑ laboratory mode (unrestricted test mode).

Processing phase
A step is divided into three processing phases: initialization, (cyclic ) processing and termination. Each processing phase corresponds to an action with statements.

Process object view
View in the SIMATIC Manager. With the process object view, all the data of the basic control of an entire project can be displayed in process control-oriented view.

Process tag
A neutral object in terms of resources. It is used to connect the PLC configuration world (STEP 7, CFC ...) with the OS configuration world (WinCC). It contains information about its location during run time (for example the network address and the memory area on the PLC) as well as information on specific OS-relevant properties.
Programmable (Logic) controller (PLC)

A programmable controller a SIMATIC S7 PLC, a complete device (PLC with integrated operator panel) in SIMATIC C7 or an automation computer in SIMATIC M7. It is used for closed and open-loop control of processes in process engineering and manufacturing.

PLC is used as a general term for the unit containing the CPUs on which the user program runs. In this case, it includes S7 programmable controllers and M7 automation computers.

Project

A folder for all the automation solutions regardless of the number of stations or modules and how they are networked.

Recipe

A recipe is the procedure for manufacturing a product according to a specific procedure.

Resources

Resources are pools of objects (FBs, FCs, DBs, OBs, bit memory, counters, timers etc.) that can be accessed when configuring and setting parameters for a CFC/SFC chart.

Run-time attribute

Run-time groups have run-time attributes that control their activation. All the blocks and/or SFC charts in a run-time group inherit these attributes from it.

In S7 the tasks are implemented as organization blocks (OBs).

Run-time group

Run-time groups are used to structure tasks. The blocks and/or SFC charts are installed in the run-time groups. Run-time groups have run-time attributes and can be activated and deactivated separately by interconnecting them with a block output or by a statement. If a run-time group is deactivated, none of the objects it contains will be activated any longer.

Run-time properties

The run-time properties of block or SFC chart determine how the block/chart is included in the execution of the entire structure on the PLC. These properties are vital to the performance of the PLC in terms of reaction times, dead times, or the stability of time-dependent structures, for example control loops.
Run sequence
This is a program sequence with which the blocks and/or SFC charts can be installed in the order in which they are executed by the CPU. The run sequence is displayed as a separate window in CFC and contains the OBs of the relevant PLC.

S

S7 program
A folder for the symbol table, the blocks, the source files, and the charts for the programmable S7 modules.

Scan rate
The scan rate is a run-time attribute. It specifies whether a run-time group is executed every time a task is executed or only every nth time. See also phase offset.

SCL
A high-level language similar to Pascal complying with IEC 1131-3 for programming complex solutions on a PLC, for example algorithms and data processing jobs.

Sequence
Structure element in SFC containing a sequence of steps and transitions.

Sequence path
In SFC, a sequence path is a series of elements (in the topology seen as a vertical path). A simultaneous sequence or an alternative sequence consists of at least two sequence paths arranged side-by-side and containing at least one element. In SFC, an entire sequencer can also be understood as a sequence path.

Sequencer
A sequencer is a functionally self-contained unit within a sequential control system. It consists of a series of steps and transitions and begins with an initial step and ends with a final step. Within the SFC, a maximum of 8 (chart) or 32 (type) sequencers can be created and selected using tabs.

Sequencer topology
In the sequencer topology of the SFC, the elements are displayed according to fixed syntax rules (for example, order, spacing, extent and alignment of the elements). When the topology is created, the syntax rules are adhered to automatically by the editor.
Sequential control system
A sequential control system switches from one to the next step dependent on conditions. Sequential control systems are implemented in PCS 7 with SFC charts (with up to 8 sequencers) or SFC types (with up to 32 sequencers).

Shared address
Shared addresses are I/O signals (input/output bits, bytes, words, double words; peripheral input/output bits, bytes, words, double words), memory bits, timers, counters, data block cells, functions, or function blocks. They can be accessed in absolute and symbolic form by SFC and CFC.
Symbolic access uses names (the symbol) of the shared address in the symbol table. For absolute access, the symbol table is not required. The absolute form of the shared address is entered. Symbolic addressing has the advantage that the connection is retained if the address is changed and the symbol stays the same.
When accessing addresses, make sure that the shared address has the same data type as the block I/O in the CFC chart or statement / condition in the SFC chart.

SFC
An SFC chart is a sequential control system that runs as a separate control system within the programmable controller.

SFC instance
See Instance.

SFC type
See Type.

SIMATIC Manager
Graphic user interface for SIMATIC users under Windows 95/98/NT/2000/XP. The SIMATIC Manager is used, for example, to create projects and access libraries.

Start condition
The start condition triggers the start of a sequencer. While the sequencer executes, it is not necessary for the start condition to remain satisfied. The sequencer executes until it is completed or until a sequencer with a higher priority and satisfied start condition needs to execute or until the sequencer is aborted or held due to a status change.
Statement

Within the action of a step, a statement can be used for the following:

- to write to CFC block I/Os
- to assign shared addresses
- to activate and deactivate SFC charts
- to activate and deactivate run-time groups

A statement always consists of a left (first) address, an operator and a right (second) address.

Step

The step is an element of a sequential control system and is the control instance for the execution of the associated actions. A step is divided into three processing phases: initialization, (cyclic) processing and termination.

Step control mode

An SFC can be executed in various step control modes. These affect the way in which the execution of the steps is enabled. You can select the following modes:

- C = confirmation by operator The sequential control system runs controlled by the operator.
- T = transition. The sequential control system runs controlled by the process.
- T or C = transition or confirmation by operator The sequential control system runs controlled by the process or by the operator.
- T and C = transition and confirmation by operator The sequential control system runs controlled by the process and by the operator.
- T / T and C = step-specific confirmation by operator. The sequential control system runs process-controlled in steps without the "confirmation" flag (as in "T") and must be confirmed by the operator at steps with the flag (as in "T and C").

Step types

An SFC has different types of steps: initial step, normal step, final step.

Structure

A structure is a structured data type made up of various elements. An element can be an elementary or a structured data type.

Structure element

Structure elements in SFC consist of an arrangement of basic elements. Structure elements are: "Sequence", "Simultaneous sequence", "Alternative sequence", "Loop" and "Jump".
**Symbol**

A symbol is a name defined by the user according to certain syntactical rules. After its purpose has been defined (for example to represent a variable, data type, jump label, block), it can be used in programming and in operator monitoring and control. Example: address: I5.0, data type: BOOL, symbol: Emer stop.

**Symbol table**

Table for assigning symbols (= name) to addresses for shared data and blocks.

Examples:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Emerstop</td>
<td>I1.7</td>
</tr>
<tr>
<td>Ctrl</td>
<td>FB24</td>
</tr>
</tbody>
</table>

**Target step**

In the test mode (or in SFC visualization on the OS), a step can be selected as the target step. This means that the deactivated sequencer begins again at the selected step instead of the initial step at the next "Start" command and a halted sequencer resumes at the target step after completing execution of the interrupted step(s) when the "resume" command is received.

**Task**

Tasks form the interface between the operating system of the CPU and the user program. The order of execution of the user program is specified in tasks. A task corresponds to an organization block in S7.

**Test mode**

(Alternative to the Edit mode) Mode of CFC/SFC for testing and optimizing user program running online on the CPU. Values of block I/Os and the execution of SFC charts on the CPU can be monitored and modified.

For testing, you can select either the process mode or the laboratory mode.

**Top chart**

A CFC chart that is not nested in another chart and can be displayed in the SIMATIC Manager (nested charts).

**Transition**

A transition is a basic element of SFC and contains the conditions according to which a sequential control system passes control from one step to the successor step.
Type

In SFC, there is not only the object type "SFC chart" but also the object type "SFC type". The SFC type allows the definition of sequential control systems including an interface.

The SFC type cannot run alone. Like a function block type, an SFC type must be placed in a CFC chart to obtain a runnable object, in this case, an SFC instance. To run an SFC instance, both the SFC type and the SFC instance are loaded on the PLC.

Update cycle

In the Test mode, this specifies the intervals at which the watched values of the block I/Os are updated.

User program

The user program contains all the instructions and declarations and the data for processing the signals allowing a plant or process to be controlled. It is assigned to a programmable module (for example CPU, FM) and can be structured in smaller units.

In S7, a user program on the ES consists of the symbol table, the source files, the blocks, and the charts.

Value identifier

Symbolic substitute (text) for defined values of block I/Os of the data types BOOL, BYTE, INT, DINT, WORD and DWORD.
# Index

## A

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action</td>
<td>1-11</td>
</tr>
<tr>
<td>Actions</td>
<td>2-23</td>
</tr>
<tr>
<td>Address selection</td>
<td>2-23</td>
</tr>
<tr>
<td>Address selection: browsing</td>
<td>2-23</td>
</tr>
<tr>
<td>Address selection: editing</td>
<td>2-24</td>
</tr>
<tr>
<td>Address selection: from a CFC chart</td>
<td>2-23</td>
</tr>
<tr>
<td>Alternative sequence</td>
<td>1-15</td>
</tr>
<tr>
<td>Alternative sequence: creating/extending</td>
<td>2-17</td>
</tr>
<tr>
<td>AS comparison</td>
<td>2-36</td>
</tr>
</tbody>
</table>

## B

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic control</td>
<td>3-1</td>
</tr>
<tr>
<td>Bit memory</td>
<td>2-30</td>
</tr>
<tr>
<td>Block contact</td>
<td>2-31</td>
</tr>
<tr>
<td>Block contacts</td>
<td>2-32</td>
</tr>
</tbody>
</table>

## C

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>3-2</td>
</tr>
<tr>
<td>Changes only</td>
<td>2-37</td>
</tr>
<tr>
<td>Changing the destination of a jump</td>
<td>2-19</td>
</tr>
<tr>
<td>Characteristics</td>
<td>2-30</td>
</tr>
<tr>
<td>Chart</td>
<td>2-3</td>
</tr>
<tr>
<td>Chart: copying</td>
<td>2-3</td>
</tr>
<tr>
<td>Chart: opening</td>
<td>2-3</td>
</tr>
<tr>
<td>Chart elements</td>
<td>2-21</td>
</tr>
<tr>
<td>Chart elements: cutting and pasting</td>
<td>2-21</td>
</tr>
<tr>
<td>Chart elements: deleting</td>
<td>2-21</td>
</tr>
<tr>
<td>Chart execution options</td>
<td>3-3</td>
</tr>
<tr>
<td>Chart reference data</td>
<td>5-2</td>
</tr>
<tr>
<td>Color settings</td>
<td>2-11</td>
</tr>
<tr>
<td>Commissioning</td>
<td>4-1</td>
</tr>
<tr>
<td>Compiling</td>
<td>2-35</td>
</tr>
<tr>
<td>Consistency check</td>
<td>2-36</td>
</tr>
<tr>
<td>Control strategy</td>
<td>2-30, 3-13</td>
</tr>
<tr>
<td>Control value</td>
<td>2-30</td>
</tr>
<tr>
<td>Copying</td>
<td>2-21</td>
</tr>
<tr>
<td>Creating an SFC type</td>
<td>2-2, 2-28</td>
</tr>
<tr>
<td>Creating/editing text objects</td>
<td>2-20</td>
</tr>
<tr>
<td>Customize</td>
<td>2-11</td>
</tr>
<tr>
<td>Customize: colors</td>
<td>2-11</td>
</tr>
<tr>
<td>Customize: compilation</td>
<td>2-35</td>
</tr>
<tr>
<td>Customize: the compiler</td>
<td>2-35</td>
</tr>
<tr>
<td>Cut</td>
<td>2-21</td>
</tr>
</tbody>
</table>

## D

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Defaults (operating parameters)</td>
<td>3-4</td>
</tr>
<tr>
<td>Delete</td>
<td>2-21</td>
</tr>
<tr>
<td>DOCPRO</td>
<td>5-2</td>
</tr>
<tr>
<td>Download</td>
<td>2-37</td>
</tr>
<tr>
<td>Downloading F programs</td>
<td>2-38</td>
</tr>
</tbody>
</table>

## E

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entire program</td>
<td>2-37</td>
</tr>
<tr>
<td>Execution on the AS sequential control system</td>
<td>3-15</td>
</tr>
<tr>
<td>External view</td>
<td>1-8, 2-5</td>
</tr>
</tbody>
</table>

## F

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Final step</td>
<td>1-11</td>
</tr>
<tr>
<td>Formulating conditions</td>
<td>2-26</td>
</tr>
</tbody>
</table>

## G

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>General numeric data</td>
<td>A-1</td>
</tr>
</tbody>
</table>

## H

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware requirements</td>
<td>A-1</td>
</tr>
</tbody>
</table>

## I

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>I/O groups</td>
<td>2-32</td>
</tr>
<tr>
<td>Instruction texts</td>
<td>2-31</td>
</tr>
</tbody>
</table>

## J

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jump</td>
<td>1-17</td>
</tr>
<tr>
<td>Jump: copying</td>
<td>2-21</td>
</tr>
<tr>
<td>Jump: creating</td>
<td>2-19</td>
</tr>
</tbody>
</table>

## K

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Keywords (DOCPO)</td>
<td>5-2</td>
</tr>
</tbody>
</table>

## L

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laboratory mode</td>
<td>4-1</td>
</tr>
<tr>
<td>Logs</td>
<td>2-36, 5-3</td>
</tr>
<tr>
<td>Loop</td>
<td>1-16</td>
</tr>
<tr>
<td>Loop: creating</td>
<td>2-18</td>
</tr>
</tbody>
</table>
# Index

## M
- Master data library .......................................................... 1-4
- Mode ..................................................................................... 3-2
- Move ..................................................................................... 2-21
- Multiproject ......................................................................... 1-4

## N
- Normal step ........................................................................... 1-11

## O
- Object properties
  - step ................................................................. 2-22
  - transition ......................................................... 2-25
- Operating state .................................................................. 3-5
- Operator mode (manual) .................................................... 3-2
- OS comment ...................................................................... 2-27

## P
- Parameter ............................................................................ 2-30
- Parameter-controlled operation ........................................ 2-39
- Paste .................................................................................... 2-21
- Phase offset ........................................................................ 2-10
- Plant hierarchy ................................................................... 1-3
- Point of use ........................................................................... 4-8, 4-9
- Position text ........................................................................ 2-31
- Postprocessing .................................................................... 2-11, 2-14
- Preprocessing ....................................................................... 2-11, 2-14
- Process mode ....................................................................... 4-1
- Process mode (Auto) ............................................................ 3-2
- Process value ....................................................................... 2-30
- Processing on the AS
  - alternative sequence ..................................................... 3-17
  - jump ............................................................................... 3-18
  - loop .................................................................................. 3-18
  - simultaneous sequence ................................................. 3-17
  - step / transition ........................................................... 3-15
- Project structure ................................................................... 1-4
- Properties
  - chart ............................................................................... 2-6
  - type .................................................................................. 2-6
- Properties dialog
  - step .................................................................................. 2-22
  - transition ......................................................................... 2-25

## R
- Recipe data block ............................................................... 2-39
- Reference data ..................................................................... 5-2
- Return path .......................................................................... 1-16
- Run-time attributes ............................................................ 2-9
- Run-time group .................................................................. 2-9
- Run-time properties ............................................................ 2-8

## S
- Scan rate ............................................................................... 2-10
- Sequence ............................................................................. 1-13
  - creating ............................................................................ 2-15
- Sequence path ..................................................................... 1-13
- Sequence path elements .................................................... 1-10
- Sequencer ........................................................................... 1-9
- Sequencer OSL .................................................................... 3-8
- Sequencer topology ........................................................... 2-12
- Sequential control system .................................................. 1-1, 2-11
- configuring .......................................................................... 2-11
- Sequential control system behavior .................................. 3-4
- Setpoint ............................................................................... 2-30
- SFC
  - printing ............................................................................. 5-1
  - SFC chart
    - creating ........................................................................ 2-1
  - SFC elements .................................................................... 1-9
  - copying .......................................................................... 2-21
  - editing ............................................................................. 2-21
  - SFC instance .................................................................... 1-7
  - SFC instances
    - copying and moving .................................................... 2-4
  - SFC type ......................................................................... 1-7
  - opening ............................................................................ 2-3
  - SFC types
    - copying and moving .................................................... 2-3
    - SFC-OSL ........................................................................ 3-6
    - SIMATIC Manager ......................................................... 1-2
  - Simultaneous sequence .................................................. 1-14
  - creating/extending ......................................................... 2-16
  - Single project .................................................................... 1-4
  - Software requirements .................................................... A-1
  - start condition .................................................................. 3-1
  - Start condition
    - 1-5, 2-13, 2-29, 3-11, 3-14
    - Start step ....................................................................... 1-11
    - Starting an SFC ............................................................ 3-14
    - Statement ........................................................................ 1-11
  - Statements ........................................................................ 2-23
  - Status display .................................................................... 4-5
  - Step .................................................................................... 1-11
    - during test ...................................................................... 4-7
    - editing ............................................................................ 2-22
    - initialization ................................................................. 2-23
    - processing ...................................................................... 2-23
    - termination ..................................................................... 2-23
  - Step control condition ...................................................... 2-25
  - Step control mode ............................................................ 3-2
    - confirmation by operator ............................................. 3-2
    - step-specific confirmation by the operator ................. 3-2
    - transition ....................................................................... 2-23
    - transition and confirmation by operator ..................... 3-2
    - transition or confirmation by operator ....................... 3-2
  - Syntax rules ....................................................................... 2-12

## SFC for S7

---

SFC for S7

A5E00177375-01
T
  T ...................................................................3-2
  T / T and C .....................................................3-2
  T and C ..........................................................3-2
  T or C .............................................................3-2
  Target step ..................................................3-12, 4-7
  Test settings .................................................4-6
  Testing
    laboratory mode ...........................................4-1
    process mode .............................................4-1
  Text ..................................................................1-12
  Time monitoring .............................................2-22
  Time stamp .....................................................2-36
  Timers ..........................................................2-30

  Transition .....................................................1-12
    condition .....................................................2-25
    during test ..................................................4-9
    editing ..........................................................2-25
    OS comment ..............................................2-27
  Transition condition ......................................1-12
  Type/instance concept .....................................1-7

V
  Value identifier .............................................2-23

W
  Watching cycle .............................................4-6