

SIEMENS

Ingenuity for life



SIMIT Getting Started

SIMIT V10.0

<https://support.industry.siemens.com/cs/ww/en/view/109767324>

Siemens
Industry
Online
Support



Legal information

Use of application examples

Application examples illustrate the solution of automation tasks through an interaction of several components in the form of text, graphics and/or software modules. The application examples are a free service by Siemens AG and/or a subsidiary of Siemens AG ("Siemens"). They are non-binding and make no claim to completeness or functionality regarding configuration and equipment. The application examples merely offer help with typical tasks; they do not constitute customer-specific solutions. You yourself are responsible for the proper and safe operation of the products in accordance with applicable regulations and must also check the function of the respective application example and customize it for your system.

Siemens grants you the non-exclusive, non-sublicensable and non-transferable right to have the application examples used by technically trained personnel. Any change to the application examples is your responsibility. Sharing the application examples with third parties or copying the application examples or excerpts thereof is permitted only in combination with your own products. The application examples are not required to undergo the customary tests and quality inspections of a chargeable product; they may have functional and performance defects as well as errors. It is your responsibility to use them in such a manner that any malfunctions that may occur do not result in property damage or injury to persons.

Disclaimer of liability

Siemens shall not assume any liability, for any legal reason whatsoever, including, without limitation, liability for the usability, availability, completeness and freedom from defects of the application examples as well as for related information, configuration and performance data and any damage caused thereby. This shall not apply in cases of mandatory liability, for example under the German Product Liability Act, or in cases of intent, gross negligence, or culpable loss of life, bodily injury or damage to health, non-compliance with a guarantee, fraudulent non-disclosure of a defect, or culpable breach of material contractual obligations. Claims for damages arising from a breach of material contractual obligations shall however be limited to the foreseeable damage typical of the type of agreement, unless liability arises from intent or gross negligence or is based on loss of life, bodily injury or damage to health. The foregoing provisions do not imply any change in the burden of proof to your detriment. You shall indemnify Siemens against existing or future claims of third parties in this connection except where Siemens is mandatorily liable.

By using the application examples you acknowledge that Siemens cannot be held liable for any damage beyond the liability provisions described.

Other information

Siemens reserves the right to make changes to the application examples at any time without notice. In case of discrepancies between the suggestions in the application examples and other Siemens publications such as catalogs, the content of the other documentation shall have precedence.

The Siemens terms of use (<https://support.industry.siemens.com>) shall also apply.

Security information

Siemens provides products and solutions with industrial security functions that support the secure operation of plants, systems, machines and networks.

In order to protect plants, systems, machines and networks against cyber threats, it is necessary to implement – and continuously maintain – a holistic, state-of-the-art industrial security concept. Siemens' products and solutions constitute one element of such a concept.

Customers are responsible for preventing unauthorized access to their plants, systems, machines and networks. Such systems, machines and components should only be connected to an enterprise network or the Internet if and to the extent such a connection is necessary and only when appropriate security measures (e.g. firewalls and/or network segmentation) are in place. For additional information on industrial security measures that may be implemented, please visit <https://www.siemens.com/industrialsecurity>.

Siemens' products and solutions undergo continuous development to make them more secure. Siemens strongly recommends that product updates are applied as soon as they are available and that the latest product versions are used. Use of product versions that are no longer supported, and failure to apply the latest updates may increase customer's exposure to cyber threats.

To stay informed about product updates, subscribe to the Siemens Industrial Security RSS Feed at: <https://www.siemens.com/industrialsecurity>.

Table of contents

	Legal information	2
1	Introduction	5
1.1	Overview	5
1.2	Principle of operation	5
1.3	Components used	5
2	Theory	6
2.1	Presentation of the target project	6
	Task	6
2.2	Creating a project	6
	Project creation.....	6
2.3	User interface of SIMIT SP	6
	Left screen edge	6
	Right screen edge.....	7
	Top screen edge.....	8
	Lower screen edge	8
2.4	Diagram structure/setting up a project.....	8
	Project tree	8
2.5	Libraries in SIMIT SP	9
	Base libraries.....	9
2.6	Operation of the interface	10
	Drag n' drop.....	10
2.7	Signals	10
	Connecting signals	10
2.8	Animations	11
	Animations in SIMIT SP	11
2.9	Block logic.....	12
	Logic blocks.....	12
2.10	Macros	13
	Macro editor	13
	Inputs/outputs.....	13
2.11	Component editor CTE	14
	CTE user interface.....	14
2.12	Connection with SIMATIC S7-PLCSIM Advanced	15
	Creating a connection.....	15
	Simulating with SIMATIC S7-PLCSIM Advanced.....	16
	Signal table.....	17
	Set up inputs and outputs	17
2.13	Digression: Control technology	18
	Building a control loop.....	18
	Use in the project.....	18
2.14	Graphically depicting signals.....	19
3	In practice	20
3.1	Basic	20
3.1.1	Task 1: Project	20
3.1.2	Task 2: Project tree.....	20
3.1.3	Task 3: Graphics	20
3.1.4	Task 4: Controls	20
3.1.5	Task 5: Animations1	20
3.1.6	Task 6: Logic1	21
3.1.7	Task 7: Logic2.....	21
3.1.8	Task 8: Animation2.....	21
3.1.9	Task 9: Macros.....	22
3.1.10	Task 10: ComponentTypeEditor CTE.....	22

Table of contents

3.1.11	Task 11: Finish Project	22
3.2	Extension	23
3.2.1	Task 12: Connection.....	23
3.2.2	Task 13: InOut.....	23
3.2.3	Task 14: PT1	23
3.2.4	Task 15: Monitoring	24
3.2.5	Task 16: Finish Project	24
4	Appendix.....	25
4.1	Service and support.....	25
4.2	Application support	26
4.3	Links and literature	26
4.4	Change documentation.....	26

1 Introduction

1.1 Overview

Getting Started gives you an insight into SIMIT's basic features. The interface and navigation of this simulation software are explained, and the first project is built step by step by using tasks. In the extensions, SIMATIC S7-PLCSIM Advanced uses and explains a coupling with a TIA portal project.

The aim is to convey basic knowledge in order to facilitate the entry into SIMIT.

Note

If you have questions about components, you can call the help by marking the component and pressing the "F1" button.

1.2 Principle of operation

Getting Started consists of a theoretical part and a practical part. First, all important sections are explained and then appropriate tasks are set. By dearchiving the respective backup project, you get to the booth before the task at hand. So you can always access a secured state.

1.3 Components used

This application example was created using these software components:

Table 1-1

Components	Quantity	Article number	Note
SIMIT SP V10.0 ENGINEERING S	1	6DL8913-0AK00-0AB5	The application example can also be tested with "SIMIT SP DEMO" (3).
SIMATIC STEP 7 Prof. V15.1	1	6ES7822-1AA05-0YA5	Required for sections 2.12 and 3.2
SIMATIC S7-PLCSIM Advanced V2.0 SP1	1	6ES7823-1FA01-0YA5	Required for sections 2.12 and 3.2
SIMIT SIM V10.0 COMPONENT TYPE EDITOR	1	6DL8913-0EK00-0AB5	Required for sections 2.11 and 3.1.10

In addition to this document, Getting Started also consists of a sample project. This contains the "Basic" and "Extension" folders, which include sample solutions, layouts for graphic objects, and all required project data. The project statuses for the task at the "Backup" sub-folder are stored. To open these project statuses, the file must be archived in SIMIT.

This application example consists of the following components:

Table 1-2

Components	File name
This documentation	109767324_SIMIT-GettingStarted_DOC_V1.0_en.pdf
Sample project for SIMIT V10.0	109767324_SIMIT-GettingStarted_PROJ_V1.0.zip

2 Theory

2.1 Presentation of the target project

Task

BringBot GmbH develops robots designed to make their customers' lives easier by taking things from one place to another. In order for this to work reliably, all robots undergo a continuous test on a treadmill.

BringBot GmbH has commissioned a new treadmill. In order to save time and reduce costs, all employees should be trained in advance to supervise the tests. In addition, a digital twin of the test setup is to be created in order to be able to virtually test updates and changes before the actual belt is implemented.

In the first part of the order, SIMIT Simulation Platform (SP) will be used to meet the requirements of BringBot GmbH.

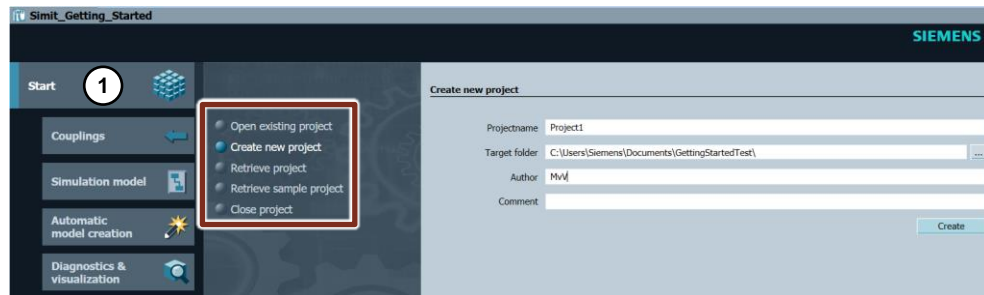
To help you make use of a secure status of backup projects at any time.

2.2 Creating a project

Project creation

In SIMIT SP, new projects can be created, existing ones can be opened, degenerated and closed. For this purpose, the "Start" navigation bar must be ① selected in the portal view.

Figure 2-1



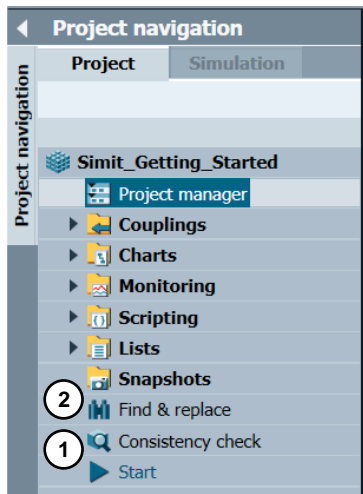
2.3 User interface of SIMIT SP

Left screen edge

The left edge of the screen is used to navigate between different views. Different diagrams can be opened there, for example. At the bottom left, you can switch between the project view and the portal view.

SIMIT SP provides the opportunity to check the consistency of the charts in the project view. ① This shows errors and alerts with localization. The affected diagram can then be clicked directly on the affected diagram and the error can be fixed. You can use "Find & replace" ②, for example, to change signal names project-wide or to search for the use of a specific signal.

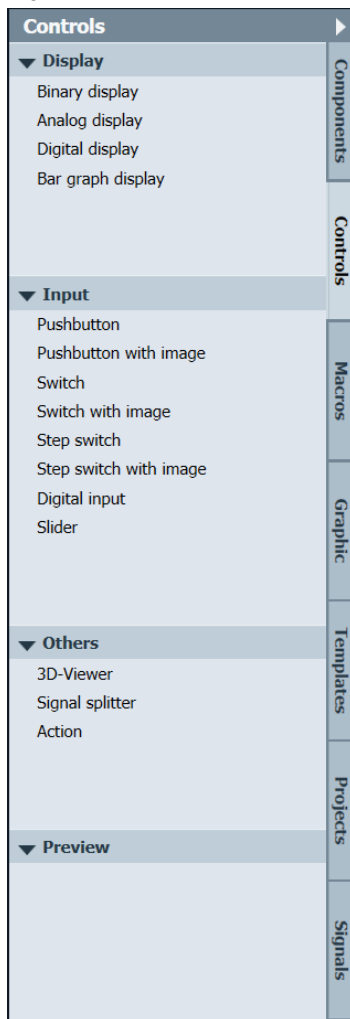
Figure 2-2



Right screen edge

The right edge of the screen gives you access to different components, graphical elements, macros or signals. Clicking on the tab in each category changes the choice of items available.

Figure 2-3

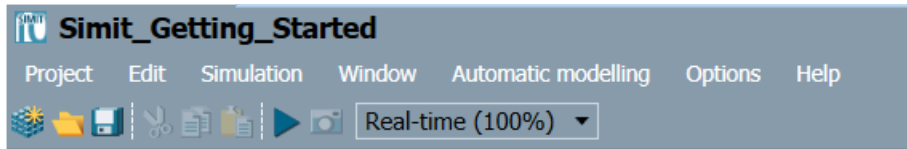


Top screen edge

At the top of the screen, you can manage, open, or save your project. Clicking on the blue arrow the simulation is started and SIMIT SP switches to simulation mode. This can be seen in the orange color of the surface. To finish the simulation, press the button .

Note In simulation mode, variables can be manipulated and observed, but no charts can be altered.

Figure 2-4

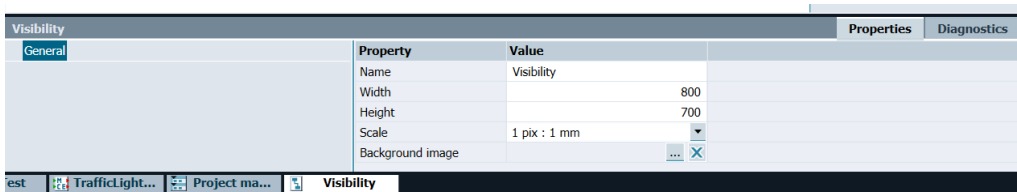


Lower screen edge

The lower screen edge shows the possible settings depending on the choice of the component. There, parameters can be set and connections can be created. If you want to customize a component, parametrize it, or animate it, you can also find the settings in the property window.

Note The property window of a component becomes visible when the component is selected.

Figure 2-5



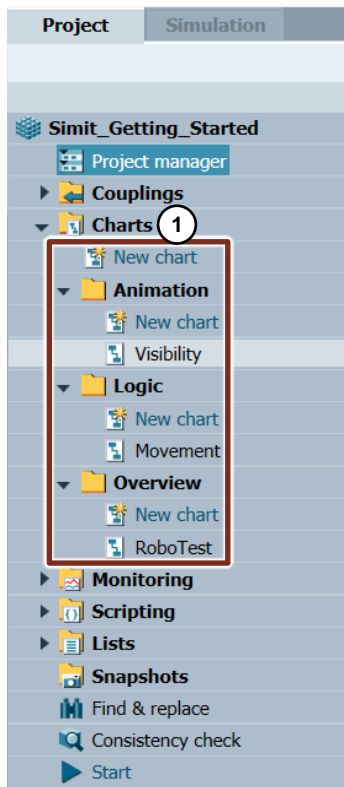
2.4 Diagram structure/setting up a project

Project tree

In order to maintain clarity as the project size increases, it is advisable to create a project tree. This is realized with different folders and is best structured according to topics.

Note With the right click on diagrams ① new folders and in turn diagrams can be created.

Figure 2-6



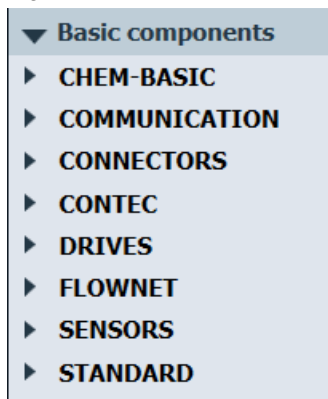
2.5 Libraries in SIMIT SP

Base libraries

After opening a diagram, you get into the editing mode and you will be listed under basic components the different libraries. Depending on the requirements, you will find components in the libraries. For example, with the library "FLOWNET" you can simulate pipe systems, pressure and temperature of gases, liquids, etc. The library "CONTEC" provides components for paths and rail vehicles, as well as belts, while "DRIVES" provides components for simulating drives.

Note All that is needed is the "STANDARD" library for this Getting Started.

Figure 2-7



2.6 Operation of the interface

Drag n' drop

Components can be pulled directly from the library into the diagram in SIMIT SP by drag n' drop. Signals, as well as graphics elements or controls, can also be pulled directly from the list, just like components, and, for example, parametrized as the input of another component.

2.7 Signals

Connecting signals



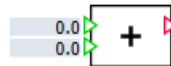
Entrances and exits of a component are represented by colored triangles. Entrances on the left in green , exits to the right are pictured in red . If the signal type is compatible, a connection can be drawn from one component to another by drag n' drop. To do this, click on the exit and then on the input to be connected (or vice versa).

Figure 2-8




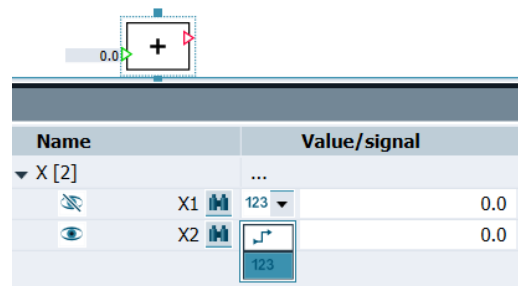

In SIMIT SP, inputs can also be described directly with a constant. By double-clicking on the input value, you can edit the constant. Entrances and exits can also be switched invisibly. To do this, the little eye  on the left side of the names must be clicked.

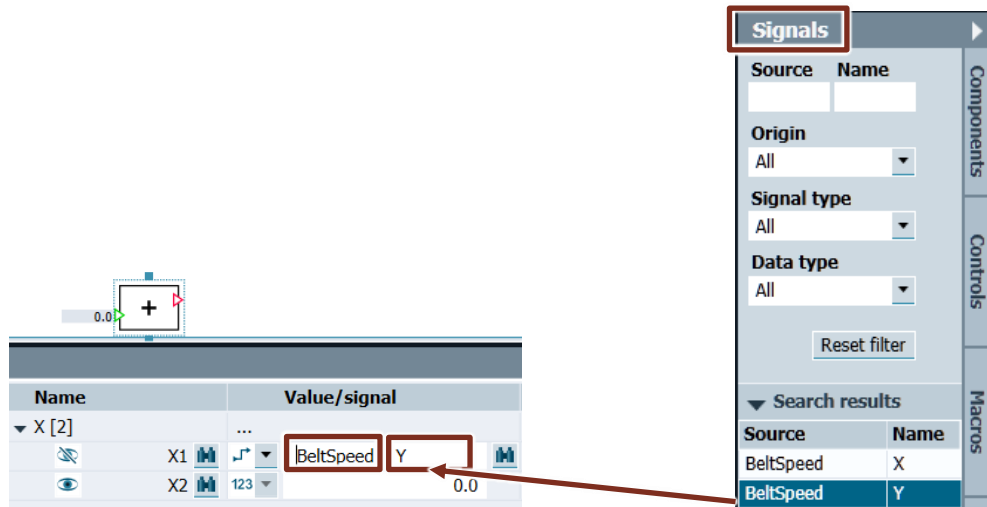
Figure 2-9



For invisible ports, you can choose "implicit connection" via signal default (represented by the buckled arrow .

If the type of connection is changed to "implicit connection," a suitable signal from the signal list on the right can be selected after consistency check and drawn directly into the field. This increases clarity on large diagrams with many connections. At the same time, at a glance, it is no longer clear which component is connected to which other component.

Figure 2-10



You can also enter the signal directly. The signal information always consists of the name of the source and the port.

2.8 Animations

Animations in SIMIT SP

Animations can be used to represent the processes more realistically. For example, graphic objects can be moved or enlarged. If a graphic object is clicked, an animation can be set in the settings under "Animations." For the animation, a signal is required as a "trigger" or "comparison size," to which the object then moves, for example, in a linear manner. The final position of the object is represented in the diagram by a red marker and can be shifted when the left mouse button is pressed.

Figure 2-11

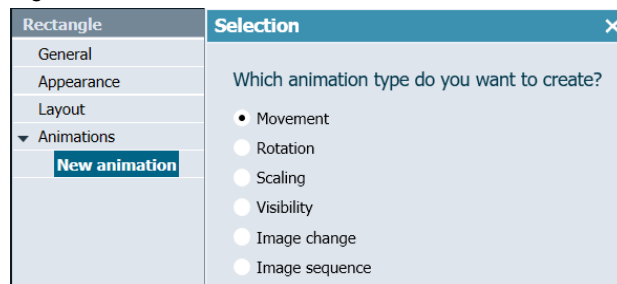
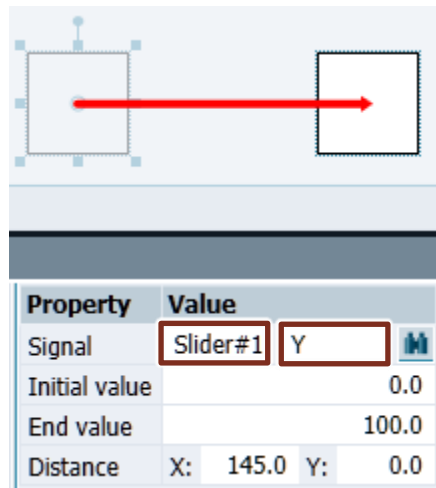


Figure 2-12



Drag the trigger signal from the signal list & drop into the property window or enter the source and connection name directly.

2.9 Block logic

Logic blocks

For logical circuits, SIMIT SP offers a number of building blocks in the library "STANDARD." Depending on the type of data, components are divided into analog, binary, and integer. "Math" also lists mathematical functions in the form of building blocks.

Figure 2-13



2.10 Macros

Macro editor

Logical circuits or contiguous components, which can be used and summarized several times in the project, can also be defined as macro. A macro forms a unit with entrances and outputs and can be considered a diagram in the diagram. A macro is created under macro "new macro." →


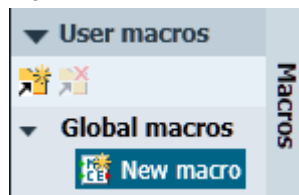
In SIMIT SP, you can integrate folders under "Own Makros/User macros"  via the Icon ("Library Open/Open library") and thus export or import macros. Exported macros have the file "simmac."

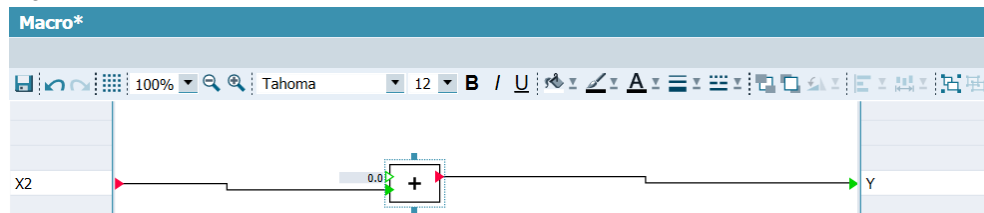
Figure 2-14



Inputs/outputs

In the macro editor there are inputs on the left and outputs on the right edge. If an input is connected to the left side, a new input is created and named. This is then created as an input at the later macro. All inputs and outputs that are not specially created are not parametrized later when using the macro. In the macro, components can be used as in the diagram. After saving, the macro can then be placed in the diagram and connected.

Figure 2-15



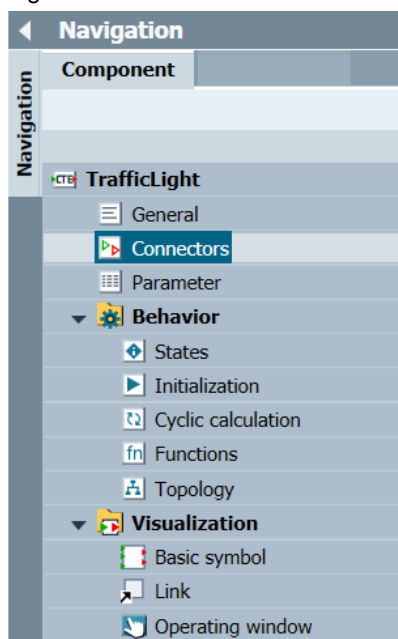
Each macro inserted into a diagram contains the processing status at the time of integration. Circuit changes of the macro in the macro editor are thus only taken over in the diagram after the macro has been replaced. You can also drag macros from diagrams into the "user macros" or "project macros" collection, then edit them.


2.11 Component editor CTE

CTE user interface

The component editor in a standalone application to create your own components and requires its own license (Table 1-1). Inputs and outputs can be created under "Connectors." In addition, parameters can be created, which later allow different settings in SIMIT SP. The "Behavior" buttons are necessary for more complex components and are not discussed here in more detail. The graphical interface of the component can be changed to "Basic Symbol." Animations can also be inserted there. The "operating window" opens when you double the level on the component in SIMIT CTE and can show analog values or settings, for example. After storing the component in the "Global components" folder, it is available in SIMIT SP.

Figure 2-16



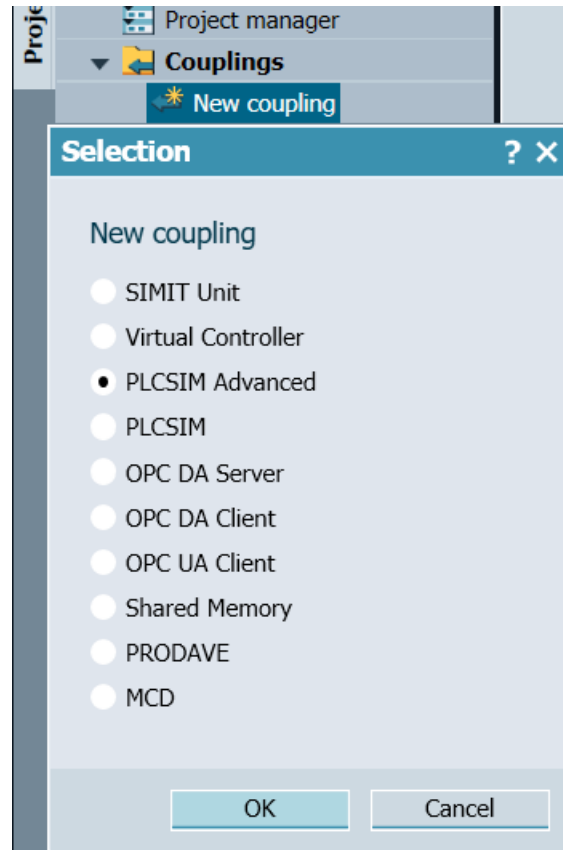
In SIMIT SP you can include folders under "Your own components/User components" via the icon  ("Open library/Open library") and thus export or import components. Exported components have the file extension "simcmp."

2.12 Connection with SIMATIC S7-PLCSIM Advanced

Creating a connection

Under "couplings" a new coupling can be created in the navigation bar. The type of coupling can be chosen by clicking.

Figure 2-17



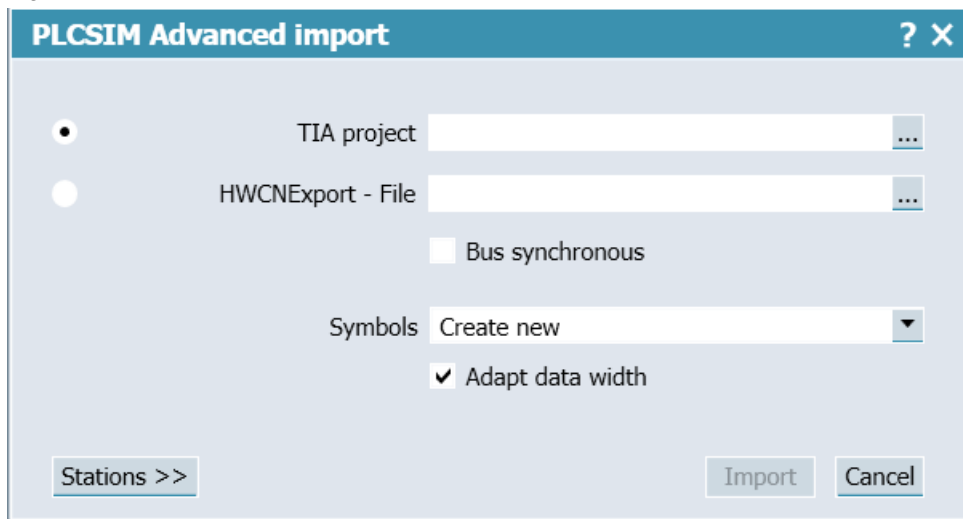
When selecting "PLCSIM Advanced," you then have to select the corresponding TIA Portal project to import its signals. To this end, SIMIT SP is accessing the TIA Portal project through Openness. This must include the Windows user of the user group "Siemens TIA Openness" (see [4](#), section: "Adding users to the "Siemens TIA Openness" user group).

You may need to grant access to the TIA portal. Check the PopUp window in the background.

Note

At TIA Portal < V15.1, the TIA Portal project must be closed.

Figure 2-18



Once the import has been completed, it can be adjusted in the properties window, whether the communication with SIMATIC S7-PLCSIM Advanced should take place via TCP/IP or via the SOFTBUS.

Figure 2-19

Localhost	
Property	Value
Host address	127.0.0.1
Communication interface	SOFTBUS

Simulating with SIMATIC S7-PLCSIM Advanced

SIMIT SP imports only the interfaces and not the data for the CPU. If the simulation is started in SIMIT SP, SIMIT SP creates an instance in S7-PLCSIM Advanced. With the launch of S7-PLCSIM Advanced, you can also see the created instance in the graphic interface or via the icon in the taskbar of Windows.

Note In this context, the instance is always created by SIMIT SP. There is no need to create your own instance.

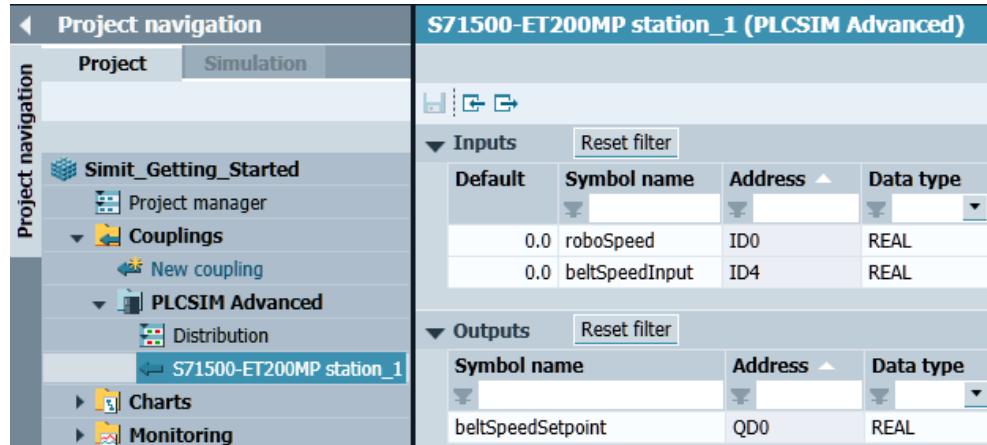
This S7 PLCSIM Advanced instance, designed by SIMIT SP, must then be downloaded from the TIA Portal.

Note This download is only necessary the first time, as the virtual memory card of the S7-PLCSIM Advanced instance is preserved.
Exception: If the TIA Portal project is changed, this change must be loaded onto the instance.

Signal table

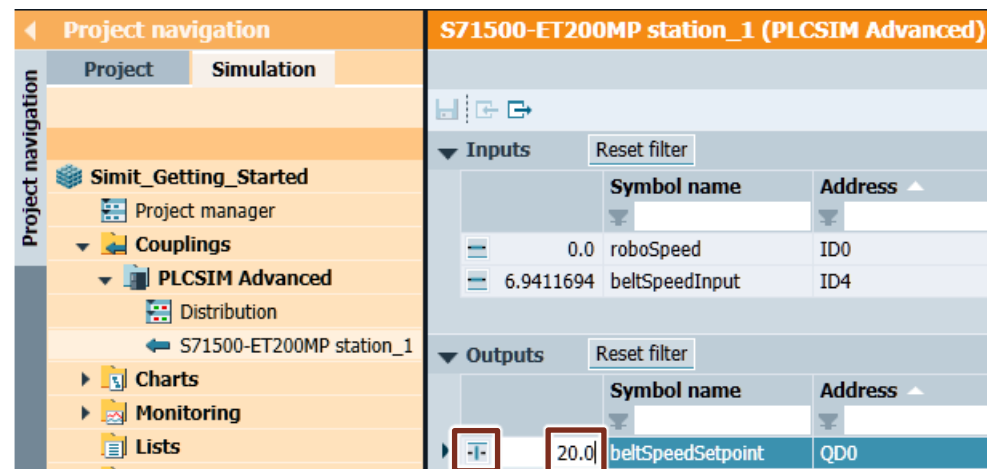
After the successful import of the project, all available inputs and outputs are displayed in a signal table.

Figure 2-20



If the simulation is active, the signals can be observed and altered directly. For signals connected in SIMIT SP, use the small field to the left of the name of the signal . You then enter a value and confirm the input with RETURN. This feature corresponds to the "Forcen" in STEP 7.

Figure 2-21

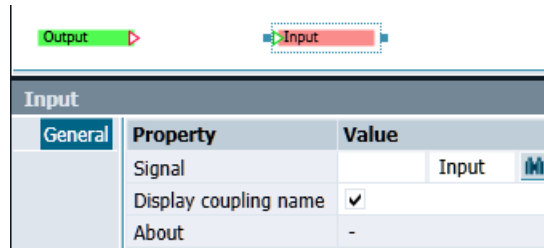


Set up inputs and outputs

Connectors are created to use signals from the PLC in SIMIT SP. It should be noted that inputs in SIMIT SP are defined from the point of view of the PLC and are therefore described by SIMIT SP with a value. An output is read in by SIMIT SP.

Connectors can be found under the basic components "connectors" and can be drawn directly into the diagram. After that, the connector still needs to be applied with the PLC signal.

Figure 2-22

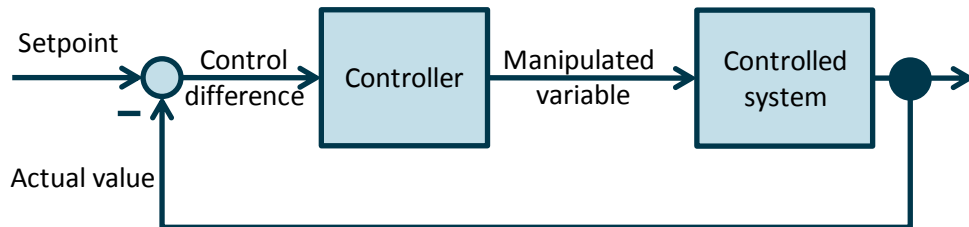


2.13 Digression: Control technology

Note This section is not necessary for task processing

Building a control loop

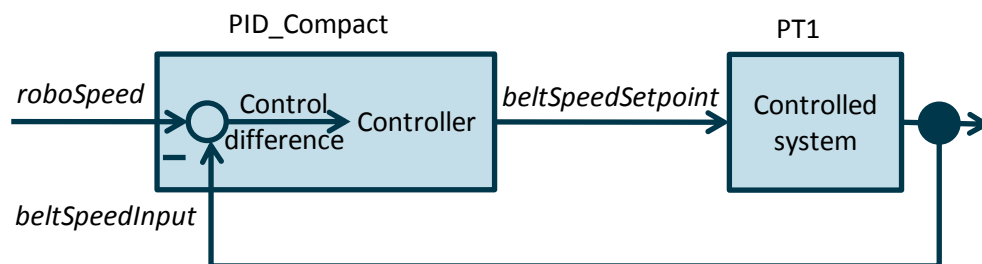
Figure 2-23



A control loop describes how an actual value becomes the desired set point. The set point is given to the control loop from the left and is also called guide size. The difference is formed from the guide size and the rule size (actual value) via the repatriation. This represents the deviation of the two values. The controller uses this rule difference and generates the manipulated variable for the controlled system (actuator). Because this is not ideal and responds to changes delayed, for example, the manipulated variable and the controlled variable are different. The new rule size is then returned and offset against the guide size. This happens until the difference is zero and the actual value reaches the set point.

Use in the project

Figure 2-24



In the TIA Portal project, the speed of the robot ("roboSpeed") is the reference variable to which the belt must adapt. A PID controller "PID_Compact" is used as a controller. PID_Compact calculates the control difference and generates the manipulated variable ("beltSpeedSetpoint") for the treadmill as a control system. This is represented by a delay element (PT1), which mimics the belt's sluggish response to changes. The resulting controlled variable ("beltSpeedInput") represents the current speed of the treadmill.

Note

The relationship between the reference variable, the manipulated variable and the controlled variable becomes more apparent in [Figure 2-26](#).

2.14 Graphically depicting signals

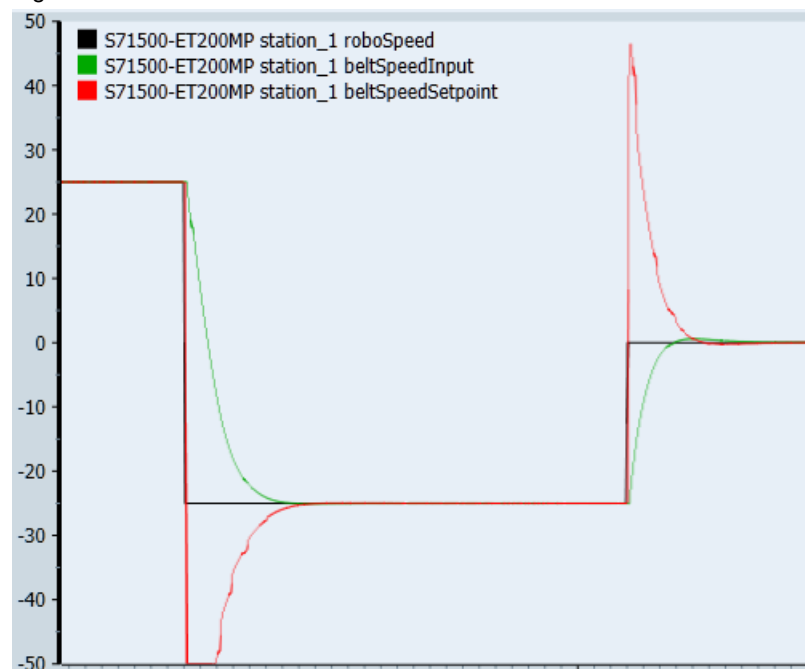
In the navigation bar is the "Monitoring" folder. New graphs can be created here. These must then be connected to the desired signals. To do this, the signal from the signal list must be drawn into a free field of properties. If the signals are to be displayed true to scale, the display range must be defined for all signals with the same limits. It is important to specify the limits in the spelling as in the figure below.

Figure 2-25

Source	Name	Color	Range
S71500-ET200MP station_1	roboSpeed	Black	(-50 .. 50)
S71500-ET200MP station_1	beltSpeedInput	Green	(-50 .. 50)
S71500-ET200MP station_1	beltSpeedSetpoint	Red	(-50 .. 50)

When the simulation is started, the signals are recorded. This history can be stored, if desired. This can be very useful for testing components or projects, as values often change very quickly and can therefore be poorly tracked.

Figure 2-26



3 In practice

3.1 Basic

3.1.1 Task 1: Project

Create a project in SIMIT SP. Choose a meaningful name for this.

3.1.2 Task 2: Project tree

In the project tree, place the "Animation," "Logic" and "Overview" folders under "Charts."

3.1.3 Task 3: Graphics

Create a new diagram in the "Overview" folder. In it, represent the treadmill and the robot in two dimensions (side view) using rectangles. For the robot, you can use the image templates in the "Basic/Layouts" folder.

Note You can add images in the properties of the graphics via "Appearance > Fill color > Image."

3.1.4 Task 4: Controls

To control the speed of the robot, place a slider in the diagram (at this point you can also test other control fields). Insert a display as a test. Connect the display directly to the slider. Start the simulation and look at the results.

Then replace the direct connection with an implicit connection. Start the simulation and check the results.

Note If signals from controls are switched invisibly, they are no longer listed on the signal list after consistency check.

The display is irrelevant to the following tasks and can be removed. Set the range of values of the slider so that it ranges from the negative to the positive.

3.1.5 Task 5: Animations¹

An animation must be inserted so that the robot can move over the treadmill. Select the slider as a signal for movement.

The robot is said to be able to start in the middle of the belt and move to both ends of the belt.

Start the simulation and check the results.

Note The movement in both directions is achieved by setting the animation parameters.

3.1.6 Task 6: Logic1

Testing the robot requires the treadmill to "move". Add a 2nd slider representing the speed of the belt.

Create a diagram in the "Logic" folder. In the diagram, calculate as the difference in the speeds of belt and robot. Select the relative speed as a signal for the movement of the robot.

In the diagram of the "Overview" folder, include a display of the relative speed.

Start the simulation and check the results.

3.1.7 Task 7: Logic2

At the moment, the robot shows the relative speed. Instead, it is now supposed to show the actual location. To do this, extend the diagram in the "Logic" folder.

Note To get from speed to position, an integrator is necessary. These can be found under the ("Basic components > STANDARD > AnalogExtended > INT").

Adjust the parameters of the integrator to the set limits of motion animation and then connect the signal.

Note The larger these limits are chosen, the more evenly the robot moves.
Recommendation: -1000 to 1000

In the diagram of the "Overview" folder, include a digital display for the position of the robot on the treadmill.

Start the simulation and check the results.

3.1.8 Task 8: Animation2

To represent the set speed of the belt, two opposite triangles ($\leftarrow\rightarrow$) are to be placed on the belt.

Note You can draw a triangle with the help of the graphic element "Polyline".

These are to increase depending on the direction of movement.

Create a new diagram for animation logic in the "Animation" folder.

Note The animation "Scaling" is necessary for the representation.

In order to prevent the arrows from scaling into the negative, you need to switch them invisibly in the respective range of values. To do this, check the belt speed for <0 or >0 .

Note The comparative can be found under the basic components: ("Basic components > STANDARD > AnalogExtended > Compare").

Start the simulation and check the results.

Note Note that the running direction of the belt is opposite to the direction of movement of the robot.

3.1.9 Task 9: Macros

The position of the robot on the belt is to be monitored. To do this, create two macros with 3 inputs and 3 outputs each – one for the left end and one for the right end of the belt.

If the robot is in the safe area, the bit "Okay" should be set.

If the robot exceeds the limit for a warning, a bit is to be set for "warning" until the robot is back in the safe area.

If the robot exceeds the limit for an alarm, the bit for "alarm" should also be set until the area is left again.

Create the limits as inputs and the three status bits as exits of the macro.

Note

For the macros, two comparisons and an "AND" operation, as well as 2 negations are necessary. The macro for the limits in negative area must be adjusted in terms of comparisons.

Insert the macros in the diagram for animations.

Set the limits depending on the scale and check the function.

3.1.10 Task 10: ComponentTypeEditor CTE

When the robot reaches the edge of the treadmill, a warning and alarm are to be graphically displayed on a traffic light. Since this display can be used in many projects, it is a good idea to create your own component. The CTE is to be used for this purpose.

The lights should have the display colors red, yellow and green and can be graphically oriented towards the traffic light. The colors should become visible depending on the circuit.

Note

The component should later be controlled by 3 binary inputs (for red, yellow, green). The inputs can be switched invisibly in the "Use" parameter by selecting the 2nd option and setting the hook at "Implicit Connection."

After storing it in the "Global Components" folder, the traffic light is also listed in SIMIT SP.

At the beginning and end of the treadmill a traffic light is now to be inserted and connected to the logic.

3.1.11 Task 11: Finish Project

Finally, the project is to be compared with the target project.

Have all the functions been implemented correctly?

Has the same solution been used or is there a more efficient solution?

This brings the Getting Started basic package to an end. In the following package, the existing project will be connected to the associated TIA Portal project and it will be launched on a SIMATIC S7-PLCSIM Advanced instance. If the self-created project is functional, it can be used below. If not, the final project of the first package can be used.

3.2 Extension

3.2.1 Task 12: Connection

After testing with the treadmill has been successful, the next goal is to increase the degree of automation. For this purpose, the treadmill should automatically assume a suitable speed, so that the robot does not drive off the belt.

A TIA portal project is available for the scheme. In the SIMIT SP project, connect with S7-PLCSIM Advanced and import the signals from the TIA Portal project. Start the simulation and specify the robot speed "roboSpeed" in the signal table. Does the set point of the belt change ("beltSpeedSetpoint")?

Note The S7 PLCSIM Advanced instance, created by SIMIT SP, has the interfaces, but not the information about the functions behind them. The TIA Portal project must therefore be loaded onto the instance at the first start of the SIMIT SP simulation with the S7-PLCSIM Advanced coupling.

3.2.2 Task 13: InOut

The signals from the TIA Portal project are to be applied in the diagram under "Logic" with entrances and exits. The robot speed "roboSpeed" is to be determined by the slider.

Note Connectors cannot be connected invisibly. Here, for example, an adder can be intermediated as a dummy.

Turn the set point of the belt directly to the actual value to the test. For example, you need an adder as an intermediate component.

Swap the slider for belt speed with a Balkan indicator that displays the actual value of the belt speed.

Perform the consistency check.

Inconsistencies are displayed anywhere in the project where the slider speed is required. The easiest way to resolve these is to use the slider name for the adder (intermediate component).

Test the simulation.

3.2.3 Task 14: PT1

In order to increase the simulation quality, the control system (the band) should now be represented by a PTn element (to be found under "Basic components > STANDARD > AnalogExtended"). To do this, this must be installed between set and actual value (instead of the adder). Leave order N to "1" for a PT1 element. The delay time constant T can be set to "3." To avoid inconsistencies, name the PT1 element as before the adder.

Note In order to avoid rewiring, the PT1 element can also be switched between set point and adder.

The simulation is then to be tested again. Is there a difference from the test before?

3.2.4 Task 15: Monitoring

In order to see more clearly the effects of the PT1 element, a curve image/trend is to be created. It is intended to represent the target and actual value of the belt and the speed of the robot. Uniform values must be chosen for the limits.

The signals are to be observed after a change in robot speed. Is the system of the PID controller recognizable?

What happens if parameter T of the PT1 element is set to 10 and the diagram is rerecorded?

3.2.5 Task 16: Finish Project

The project must be considered in its entirety.

- Are all the required features available?
- Can other features improve simulation?

4 Appendix

4.1 Service and support

Industry Online Support

Do you have any questions or need assistance?

Siemens Industry Online Support offers round the clock access to our entire service and support know-how and portfolio.

The Industry Online Support is the central address for information about our products, solutions and services.

Product information, manuals, downloads, FAQs, application examples and videos – all information is accessible with just a few mouse clicks:

<https://support.industry.siemens.com>

Technical Support

The Technical Support of Siemens Industry provides you fast and competent support regarding all technical queries with numerous tailor-made offers – ranging from basic support to individual support contracts. Please send queries to Technical Support via Web form:

www.siemens.com/industry/supportrequest

SITRAIN – Training for Industry

We support you with our globally available training courses for industry with practical experience, innovative learning methods and a concept that's tailored to the customer's specific needs.

For more information on our offered trainings and courses, as well as their locations and dates, refer to our web page:

www.siemens.com/sitrain

Service offer

Our range of services includes the following:

- Plant data services
- Spare parts services
- Repair services
- On-site and maintenance services
- Retrofitting and modernization services
- Service programs and contracts

You can find detailed information on our range of services in the service catalog web page:

support.industry.siemens.com/cs/sc

Industry Online Support app

You will receive optimum support wherever you are with the "Siemens Industry Online Support" app. The app is available for Apple iOS, Android and Windows Phone:

support.industry.siemens.com/cs/ww/en/sc/2067

4.2 Application support

Siemens AG
 Digital Factory Division
 Factory Automation
 DF FA S SUP E&C
 Gleiwitzerstr. 555
 90475 Nuremberg, Germany
 mailto: simatic.industry@siemens.com

4.3 Links and literature

Table 4-1

No.	Topic
\1\	Siemens Industry Online Support https://support.industry.siemens.com
\2\	Link to the entry page of the application example https://support.industry.siemens.com/cs/ww/en/view/109767324
\3\	Download "Simulation software SIMIT V10.0 Upd1" https://support.industry.siemens.com/cs/ww/en/view/109759702
\4\	Manual "SIMATIC Openness: Automating project creation" https://support.industry.siemens.com/cs/ww/en/view/109477163
\5\	Manual "SIMATIC SIMIT Simulation Platform (V10.0)" https://support.industry.siemens.com/cs/ww/en/view/109759317
\6\	Product release "Sales and delivery release: Simulation software SIMIT V10.0 " https://support.industry.siemens.com/cs/ww/en/view/109758734

4.4 Change documentation

Table 4-2

Version	Date	Change
V1.0	06/2019	First version