Honeywell

Experion PKS
SIM-C300 User's Guide

EPDOC-X133-en-431A
February 2015

Release 431

Honeywell

Document	Release	Issue	Date
EPDOC-X133-en-431A	431	0	February 2015

Disclaimer

Contents

1 About this guide

This document provides guidelines to use the SIM-C300 modules.

Revision history

Revision	Date	Description
A	February 2015	Initial release of the document.

Intended audience

This guide is intended for the following users:

- Persons responsible for system planning, initial hardware installation, and control strategy configuration
- Operators who help to maintain control system operations on a day-by-day basis
- Service persons responsible for routine maintenance of control hardware and who also diagnose and repair faults.

Prerequisite skills

It is assumed that you should have some knowledge of Experion control systems and experience of working in a Microsoft Windows environment.

Related documentation

PDF Collection is the online documentation library for the Experion system. It is provided on a compact disc and can be installed on a suitable personal computer.

Within PDF Collection information for both the Series C release and other Experion control hardware, such as Process Manager I/O, Series C I/O, and Series A Chassis I/O can be found.

Listed here are PDF Collection documents that contain general information for planning and implementing control hardware and network communications in your Experion system:

C300 Controller User's Guide - This guide provides information that assists you in planning and designing activities, as well as the installation, operation, and troubleshooting of C300 Process Controllers in an Experion system.

Control Building Guide - This guide contains basic tasks within the Control Builder application such as configuring hardware devices, continuous control strategies, and sequential control strategies. Only representative forms are shown to illustrate a procedure/concept.

Series C I/O User's Guide - The guide contains planning and implementation information for Series C I/O modules. Module types include: AI, AO, LLMUX, DI and DO.

SIM-ACE User's Guide - This document covers placing the ACE FB in simulation and other procedures for interacting with the ACE simulation functionality.

SIM-C200E Implementations Guide - This document provides information about simulating your C200 Controller control strategy with or without Honeywell's UniSim Operations simulator application.

Control Hardware Planning Guide - Provides general information to assist you in planning and design of control hardware in an Experion system. Control hardware includes C200 Controllers, Series A Chassis I/O and

FIMs, also, all I/O families, (except Series C I/O). It includes some supervisory network considerations for general reference.

Process Manager I/O Troubleshooting and Maintenance Guide - Guide features notification messages (soft fail codes and hard fail codes), service procedures and parts lists for PMIO I/O control hardware.

Various UniSim documents:

- UniSim documentation - installation and node administration information residing in the UniSim bookset within PDF Collection.
- UniSim Operations R310 Help - online help within the UniSim application
- UniSim Operations Shadow Plant Configuration Guide - online help within the UniSim application

2 SIMC300 - Simulation for C300

Related topics

“Getting started” on page 8
“Supported function blocks and IO families” on page 9
“Licensing scheme for SIMC300” on page 11

2.1 Getting started

This section includes information that compares the SIMC300 and C300.

Topic	Link
SIMC300 and C300 comparison	“SIMC300 and C300 comparison” on page 8

2.1.1 SIMC300 and C300 comparison

The following is applied to the SIMC300 and C300:

- They share the same template.
- Converting from a SIMC300 to a C300 does not impact most parameters. (MODISREDUN is disabled when the C300 is converted to a SIMC300).
- SIMC300 supports full simulation of C300 controller including:
  - load and execution of all function blocks supported by C300 controller.
  - EE executing with the same base execution cycles, blocks executing with the same block execution period selections.
  - interaction with Series C I/O, PM I/O, and ControlNet resident I/O.
- Simple importing of a set of strategies from a on-process to an OTS system. You only need to:
  - check the SIMTARGET checkbox, and
  - give a proper IP address or Hostname if you chose to maintain the original configuration
- In the R310 Release, SIMC300 is released as non-redundant platform.
  - The option of configuring a SIMC300 as redundant controller is disabled.
  - Converting a redundant C300 controller directly to a SIMC300 is also not allowed. An error message is returned.

    The following is recommended:

    You convert the redundant C300 to a non-redundant controller, and then convert it to a SIMC300 controller.
- SIMC300 supports both FTE and Ethernet network configurations. Unlike SIMC200E, it does not support CNet.
- SIMC300 requires Experion server and Station to support Operator Interface, History, and other functions just like the C300 controller.
- SIMC300 accepts commands from UniSim.
- SIMC300 supports the same Checkpoint save/restore functions that are supported by C300 controller.
- Multiple SIMC300s are allowed to be running on the same node (up to 5 physical CPUs when execution cycle determinism is guaranteed, or 20/Dual CPU when execution cycle determinism is not guaranteed).

Attention

Simulation in high capacity must be run in a separate simulation node where as those in low capacity must be run in the Engineering Station.

2.2 Supported function blocks and IO families

All of the Experion function blocks that are supported by C300 in the R310 release also are supported by SIMC300. This means those function blocks can be assigned and loaded to SIMC300 environment. UCNIF and CAB function blocks are not supported by C300, therefore, are not supported by SIMC300. An error is generated if you attempt to assign/import/load these unsupported block types to either C300 or SIMC300.

The following table demonstrates what type of function block family is supported in SIMC300 R310 Release.

Table 1: FBs supported in SIMC300

Function Block library	C300	SIMC300
STATIC BLOCK		
AUXILIARY	**X**	**X**
DEVCTL	**X**	**X**
DATAACQ	**X**	**X**
IOMODULE	**X**	**X**
IOCHANNEL	**X**	**X**
LOGIC	**X**	**X**
MATH	**X**	**X**
RCM	**X**	**X**
REGCTL	**X**	**X**
SCM	**X**	**X**
UCM	**X**	**X**
UCNIF	n/a	n/a
UTILITY	**X**	**X**
PCDI	**X**	**X**
POWERGEN	**X**	**X**
CCL BLOCK		
AGA	**X**	**X**
EXCHANGE	**X**	**X**
ABDRIVE	**X**	**X**
DNET	**X**	**X**
HARTIO	**X**	**X**
PBUSIF	**X**	**X**
PULSEINPUT	**X**	**X**
RAIL_IO	**X**	**X**
RAIL_IO_HAZ	**X**	**X**
CUSTOM BLOCK		
CAB	n/a	n/a
CDB	**X**	**X**
PHASE	**X**	**X**
IO		
PMIO	**X**	**X**

Function Block library	C300	SIMC300
SERIES C IO	X	X

Table 2: 1756 Series A I/O modules and channels

IOM Name	Description	Associated Blocks	Channel Name	Channels
TC-IAH061	Analog Input, Isolated	CEE	AICHANNEL	6
TC-IHA161	Analog Input	CEE	AICHANNEL	16
TC-IDA161	Digital Input	CEE	DICHANNEL	16
TC-IDD321	Digital Input	CEE	DICHANNEL	32
TC-IDJ161	Digital Input, Isolated	CEE	DICHANNEL	16
TC-IDK161	Digital Input, Isolated	CEE	DICHANNEL	16
TC-IDW161	Digital Input, Isolated	CEE	DICHANNEL	16
TC-IDX081	Digital Input, Diagnostic	CEE	DICHANNEL	8
TC-IDX161	Digital Input, Diagnostic	CEE	DICHANNEL	16
TC-IXL061	Analog Input, Thermocouple (TC/mV)	CEE	AOCHANNEL	6
TC-IXL062	Analog Input, Thermocouple (TC/mV)	CEE	AOCHANNEL	6
TC-IXR061	Analog Input, RTD	CEE	AOCHANNEL	6
TC-MUX021	Serial Interface	CEE	SIFLAGARRCH SINUMARRC SITEXTARRCH	32
TC-OAH061	Analog Output, Isolated	CEE	AOCHANNEL	6
TC-OAV061	Analog Output, Isolated	CEE	AOCHANNEL	6
TC-OAV081	Analog Output	CEE	AOCHANNEL	8
TC-ODA161	Digital Output	CEE	DOCHANNEL	16
TC-ODD321	Digital Output	CEE	DOCHANNEL	32
TC-ODJ161	Digital Output, Isolated	CEE	DOCHANNEL	16
TC-ODK161	Digital Output, Isolated	CEE	DOCHANNEL	16
TC-ODX081	Digital Output, Diagnostic	CEE	DOCHANNEL	8
TC-ODX161	Digital Output, Diagnostic	CEE	DOCHANNEL	16
TC-ORC081	Relay Output, Isolated	CEE	PWMCHANNEL	8
TC-ORC161	Relay Output, Isolated	CEE	PWMCHANNEL	16

2.3 Licensing scheme for SIMC300

A new license key - CX00 Simulation Environment is introduced in R310 Release for the purpose of tracking the usage of SIMC200E, SIMC300 and future SIMX00 controllers.

It is prohibited to load a SIMC200E or a SIMC300 instance if the CX00 Simulation Environment license was not purchased as part of the license package

SIMC300 activity	If license is available	If license is not available
Create	Can be created beyond the range set by the license	Can be created beyond the range set by the license
Load	Can be created	Cannot be loaded

If the system does have a valid CX00 simulation environment license, SIMC200E and SIMC300 are allowed to be created even if the total instances of SIMC200E and SIMC300 has reached the purchased license limit. SIMC200E and SIMC300 can be loaded only if it is within the purchased license range.

3 SIMC300 - Planning

Related topics

“SIMC300 simulation environment” on page 14
“HART Device Simulation” on page 16

3.1 SIMC300 simulation environment

In some cases simulation and on-process environments exists on different systems utilizing different Engineering Repository Databases (ERDBs). If this is the case, a one-time:

- import/export operation, or
- engineering database backup/restore

from simulation to on-process (or on-process to simulation) and loading of all the controllers is necessary.

Topic	Link
SIMC300 usage	“SIMC300 usage” on page 14
Interoperability of SIMC300 with other environments	“Interoperability of SIMC300 with other environments” on page 14

Related topics

“SIMC300 usage” on page 14
“Interoperability of SIMC300 with other environments” on page 14

3.1.1 SIMC300 usage

The following are the *possible simulation scenarios* that SIMC300 supports:

- C300 hardware is not available - create control strategies in a SIMC300 environment.
- Using an existing on-process strategy - bringing it into a SIMC300 environment.
- Creating a simulation strategy - then bringing it into an on-process environment.
- Using the same ERDB for simulation and on-process work (although you must modify the SIMTARGET from C300 to SIMC300).
- Moving an ERDB from on-process to simulation.

3.1.2 Interoperability of SIMC300 with other environments

Experion R310 is the first official release for SIMC300. The Experion server and station have to be on the R310 release also to allow proper functionality between the on-process and the OTS systems.

Refer to the following table for the interoperability between C300 and other environments.

Table 3: Table 3 SIMC300 and other environments

SIMC300 and …	Relationship
Server/Station	The engineering tool and database must match the release of the SIMC300. *If:* SIMC300 = release 310.1 *Then*: EPKS server and station = release 310.1 engineering too and database = release 310.1
Other controllers	The general operability principle allows the peer-to-peer communication to one release back. If the node is at Experion R310 with SIMC300, then peer-to-peer communication to SIMC200E is allowed.

SIMC300 and ...	Relationship
UniSim	It is recommended to use the corresponding release of UniSim and Experion. This allows the matching of the on-process and OTS functions *If* Experion = release R310.1 *Then:* UniSim = release R310

SIMC300 and other simulation environments

SIMC300 has full bidirectional connectivity with other simulation environments. This means SIMC300 can either read-from or write-to other simulation environments. Therefore, other simulation environments can also read-from or write to the SIMC300 environment.

Tip

Only SIMC300 initiated read connection or **on-process environment** initiated store connections are allowed between SIMC300 and an **on-process controller.** All other types of peer communications are rejected.

SIMC300 and communication between two different servers

Communication between two execution environments on two different servers must use SIMACE to communicate with OPC Gateway and Inter Cluster Gateway. In order to write data from a SIMACE, the OPC Gateway must be specified as a Simulation OPC Gateway. This is done by setting SIMENABLE as TRUE on an OPC Gateway platform block.

Table 4: SIMC300 and read/write permissions

Function block family	SIMC300 as initiator	SIMC300 as responder		
Read	Write	Read	Write	
C200	n/a	n/a	n/a	n/a
C300	**X**	n/a	n/a	**X**
ACE	**X**	n/a	n/a	**X**
IOLIM	n/a	n/a	n/a	n/a
FIM2	**X**	n/a	n/a	**X**
FIM4	**X**	n/a	n/a	**X**
LIOM	n/a	n/a	n/a	n/a
SM	n/a	n/a	n/a	n/a
EHG	n/a	n/a	n/a	n/a
SIMC200E	**X**	**X**	**X**	**X**
SIMC300	**X**	**X**	**X**	**X**
SIMACE	**X**	**X**	**X**	**X**
SIMIOLIM	n/a	n/a	n/a	n/a
FF-SIM2	n/a	n/a	n/a	n/a
FF-SIM4	n/a	n/a	n/a	n/a
SIM-EHG	n/a	n/a	n/a	n/a

3.2 HART Device Simulation

3.2.1 Supported HART functions in simulation

In the simulation environment, the HART channel block functions, support:

- Monitoring HART device.
  - Configuration parameters.
  - Status parameters including Command 48.
- Building HART device network in FDM.

3.2.2 Monitoring HART device parameters

SIMC300 supports the monitoring of the following HART channel block parameters from Control Builder, FDM and SWMUX.

Table 5: Supported HART Device Parameters

Parameter	Description	Value Read	Remarks
HDEVMFG	Manufacturer	HDVMFGCD	Configured value
HDEVTYPE	Type	HDVTYPCD	Configured value
HDEVID	Id (Serial Number)	HDEVIDCD	Configured value
HDEVREV	Revision	HDVREVCD	Configured value
HHWREV	Hardware Revision	“0”	Default value
HSWREV	Software Revision	“0”	Default value
HFLAGS	HART Device Flags	“0”	Default value
HNMSMINPRE	Minimum M to S Preambles	“2”	Default value
HUCMDREV	Universal Command Revision	“HART Version 6”	Default value
HCOMSTS	HART Communication Status	“OK”	Default value
HDEVST	General Device Status	dynamic	Read and Write support
HTAG	Tag	Null	Default value
HDEVTYPENAME	Type (Name)	HDVTYPCDNAME	Configured value
HCMD48BT[1..200]	Command 48 Bits	dynamic	Read and Write support
HGCHNGFL	HART Global Change Flag	dynamic	Read and Write support
HDEVST	HART Device Status	dynamic	Read support
IOM.HGCHNG1	IOM HART Global Change Flag	dynamic	Read support

3.2.3 Non-supported HART functions in simulation

The following FDM operations are not supported for HART devices assigned to SIMC300:

- Configuring a field device
- Offline Configuration - Offline Download to device, Bulk Offline download
- Accessing Device Specific Parameters
- Executing HART Field Device methods

- Viewing Device Status
- Viewing and editing Device Properties
- Comparing Device Configuration
- Device History
- Exporting Online Configuration
- Saving an Online Configuration as an Offline Configuration
- Renaming FDM tags

4 SIMC300 - Installation

Related topics

“Installing SIMC300” on page 20

4.1 Installing SIMC300

This section identifies where to find the various SIMC300 installation options.

Topic	Link
Selecting the simulation environment	“Selecting the simulation environment” on page 20

4.1.1 Selecting the simulation environment

There are multiple selection options when installing the simulation environment for an Experion system.

Refer to the *Supplementary Installation Tasks Guide* (SITG) for assistance in installing the simulation environment.

5 SIMC300 - Configuration

Related topics

5.1 Configuration overview

This section includes information that you use to configure the SIMC300 environment from within your Experion system.

Topic	Link
Configuration Studio	“Configuration Studio” on page 22
Control Builder	“Control Builder” on page 22

Related topics

“Configuration Studio” on page 22
“Control Builder” on page 22

5.1.1 Configuration Studio

Configuration Studio is the central location from which you can access engineering tools and applications to configure your Experion system. When you choose Control Strategy in the Configuration Explorer tree and then choose the task Configure a Control Strategy, Control Builder is launched so you can configure Series C hardware modules, build the process control strategies for your system and configure the SIMC300 environment.

5.1.2 Control Builder

Control Builder is the application used to create and configure Series C hardware modules and function blocks so you can build process control strategies for your system. Control Builder is accessed through Configuration Explorer.

- If you are familiar with using Control Builder, then most of the same rules apply for configuring, loading and monitoring when implementing Series C control hardware.
- If you are new to Control Builder, you should first refer to the *Control Building Guide* to familiarize yourself with the application and its capabilities.
- To complete some configuration selections for the SIMC300, you should refer to the C300 Controller User's Guide.

Attention

The information and procedures presented in this section apply to using Control Builder for configuring the C300 Controller. Some procedures cover the creation and configuration of the Control Execution Environment block (CEEC300) where you create your control strategies. This section does not attempt to provide all details for using Control Builder in configuring the numerous components that are associated with the Experion system.

Please refer to the *Control Building Guide* for additional information and procedures.

5.2 Creating an IO module for simulation

SIMC300 is comprised of four function blocks:

- one platform block (C300 FB)
  - The platform block (C300 FB) represents the C300 simulation application in Experion system. It contains the CEE block and the two IOLINK blocks.
- one execution block (CEE FB), and
- two I/O link blocks (IOLINK1 and IOLINK2 FB).

The following selections need to occur on the C300 - SIMC300 configuration form:

- Checking the Load to Simulation checkbox The Load to Simulation Environment checkbox changes the platform between SIMC300 and C300 environment. All configuration options that exist for the C300 also apply to the SIMC300, with the exception of module redundant configuration.
- Setting the Device Index parameter to a non-used odd integer number within the range of 1 ~ 255
- Either assign the IP address or the host name of the node that the SIMC300 is running. Once one is configured (IP address or host name), the other one is automatically resolved.

Figure 1: C300 and SIMC300 icons in Control Builder

5.2.1 Creating a SIMC300 controller

The C300 and SIMC300 share the same template.

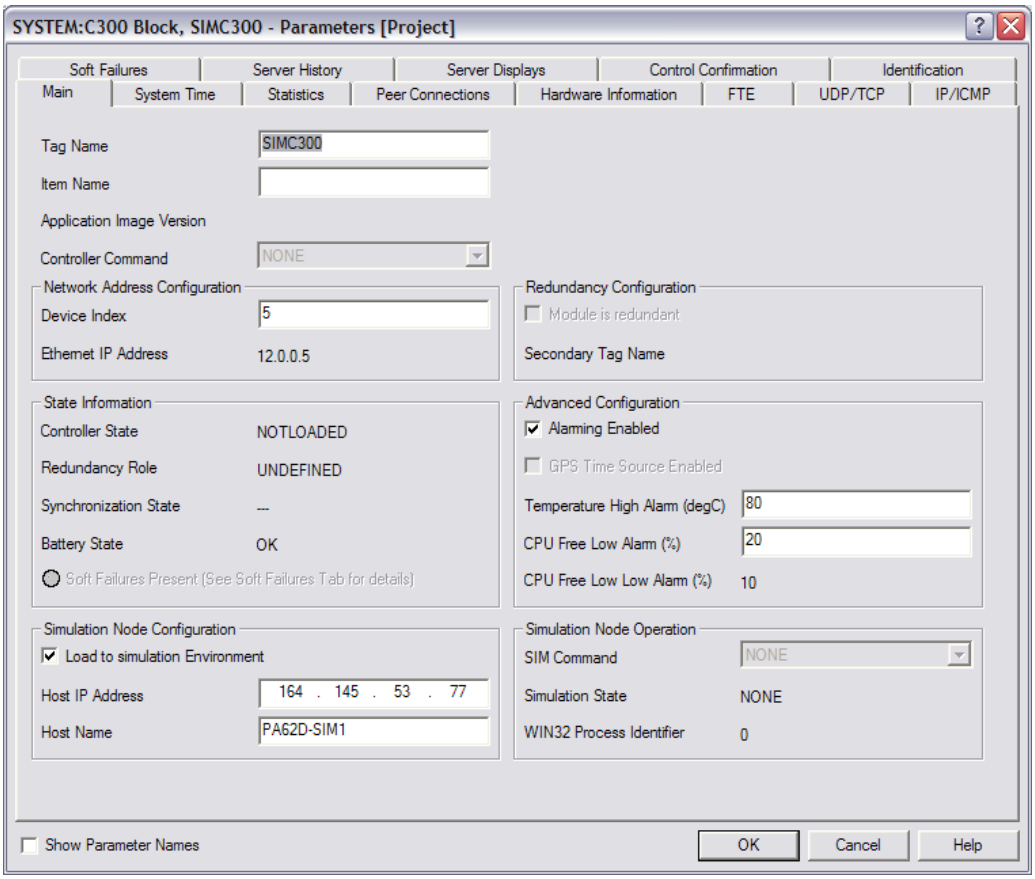
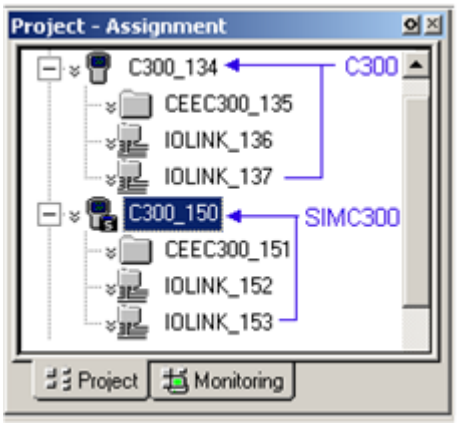


Figure 2: C300 FB configured as SIMC300

To create a SIMC300 controller

1. Click **File -** > **New -** > **Controllers -** > **C300 - Controller (2 I/O links).**
   Result: The C300 Block, SIMC300 configuration form appears.
2. Type in desired name for the SIMC300 of up to 16 characters or accept the default in the **Tag Name** field.
3. Click in the **Device Index** field (DEVICEIDX parameter).
4. Type a non-used odd integer number in the range of 1 through 255.
5. Check the **Load Simulation Environment** (SIMTARGET parameter) check box.
   Note: Unchecking the **Load Simulation Environment** check box causes the SIMC300 to revert back to C300. All configuration options (except Module redundant configuration) that are specific to SIMC300 are applicable to C300.
6. Type the IP address (or the host name of the node) of where the SIMC300 is located.
   Result: The following is an example of the SIMC300 block icons that now appear in the project window:



7 All additional configuration selections for the SIMC300 are identical in the manner to which the C300 selections are made (with the exception of redundancy determination).
Refer to the Controller User's Guide for support with these selections.

5.2.2 Configuring a CEEC300 block in simulation

The CEE300 function block publishes parameters reflecting status and configuration of the execution environment. It monitors the peer connection between itself and other execution environments, as well as the local data transfer between itself and two IOLINKs within the same SIMC300.

There are two parameters introduced to CEE FB configuration form that are meaningful only when the controller is running as SIMC300 - NOTIFINHIBIT SIMSTATE.

Refer to the following, *To configure a CEEC300 function block* in PDF Collection for the procedure to configure the CEEC300 block for simulation.

5.2.3 Configuring IOLINK block in simulation

Two IOLINK blocks are created automatically when a SIMC300 Controller function block is added to the Project tab.

Refer to the following, *To configure IOLINK blocks* in PDF Collection for the procedure to configure the IOLINK block for simulation.

5.2.4 Creating a Series A I/O module for simulation

1 Click **File -** > **New -** > **I/O Modules -** > **[desired I/O module].**
Result: The I/O module's configuration form appears.

2 On the Main Tab of the form, enter the appropriate I/O hardware location data in the applicable fields such as; the IOM Slotnum in Chassis, Uplink CNB in IO Rack, and Downlink CNB Slot Number for Chassis I/O. If necessary, access the on-line help for assistance during this step.

3 Click the OK button to create an instance of the I/O Module block in the Project. tree.

4 If multiple IOMs are required, repeat this procedure for each occurrence of the IOM, even if the type is the same as a previously created IOM.

5 Complete the configuration of each module before loading it to the Controller. Use the appropriate configuration procedure in this document or the applicable I/O component Implementation Guide for reference along with the on-line help.

6 All additional configuration selections for the I/O module for simulation are identical in the manner to which the I/O module selections are made (with the exception of redundancy determination).

7 Refer to the C300 Controller User's Guide for support with these selections.

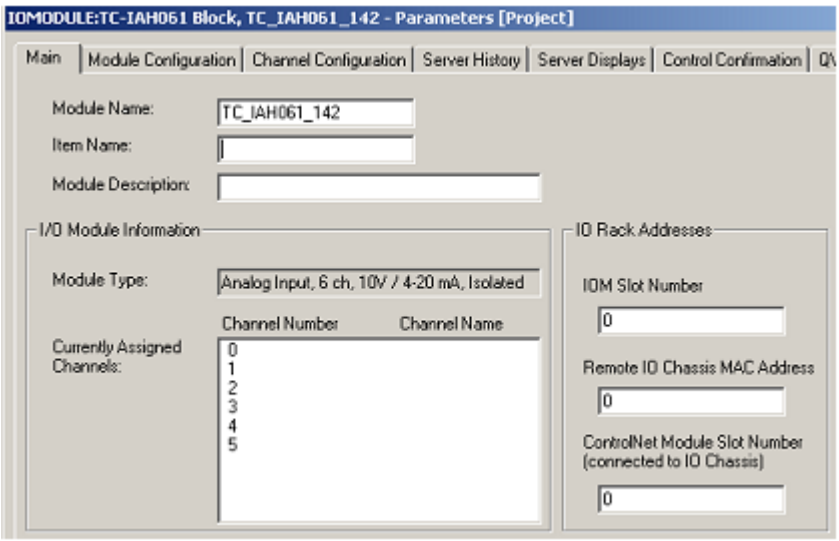
5.2.5 1756 Series A in simulation

Use the following links to access 1756 modules/channels information that support SIMC300 simulation.

- Refer to ‘Table – 1756 Series A I/O modules and channels’ for a listing of the 1756 modules that support simulation with SIMC300.

5.2.6 Creating a 1756 I/O module for simulation

1 Click **File --** > **New -** > **I/O Modules -** > **I0MODULE -** > **[desired I/O module].**.
Result: The I/O module's configuration form appears.



2 On the Main Tab of the form, enter the appropriate I/O hardware data in the applicable fields such as;
Module Name:

Item Name

Module Description

The IOM requires a properly defined path consisting of the following:

IOM Slot Number (IOMSLOT): The limit is 39

Remote IO Chassis MAC Address (ULCNBMAC) : The limit is 20

ControlNet Module Slot Number (DLCNBSLOT): The limit is 20

Also required is the FTEBBLOCK although not visible on the IOM until the IOM block is assigned to a SIMC300 CEEFB. Then the list of applicable FTEB's is available.

Tip
The following occurs if there is any invalid configuration

- A warning will be returned at configuration time.
- An error will be returned at the load time
- The load operation will fail.

IOMs that have the same path with a loaded IOM will also be rejected during load time.

3 Click the OK button to create an instance of the I/O Module block in the Project. tree.

4 If multiple IOMs are required, repeat this procedure for each occurrence of the IOM, even if the type is the same as a previously created IOM.

5 Complete the configuration of each module before loading it to the Controller. Use the appropriate configuration procedure in this document or the applicable I/O component Implementation Guide for reference along with the on-line help.

5.2.7 Creating a Series C I/O module for simulation

1 Drag the Series C I/O module to the IOLINK block of the SIMC300.

2 Double-click the I/O module.
The I/O module configuration form appears.

3 Complete the configuration of the I/O module. For more information about configuring the I/O module, refer to the Series C I/O User's Guide.

4 Assign a channel to the CM.
For more information about adding IO channel blocks to a CM, refer to the *C300 Controller User's Guide*.

5 Double-click the channel block.
The channel block configuration form appears.

6 Click the **Configuration Parameters** tab.

7 Add **SIMMODE** parameter as a configuration parameter.
The **SIMMODE** parameter appears on the channel block's faceplate.

8 Click the **Monitoring Parameters** tab.

9 Add **SIMVALUE** parameter as a monitoring parameter.

10 Click the **Block Pins** tab.

11 Select the **SIMVALUE** parameters and expose it as a block pin.

12 Click **OK**.
The **SIMVALUE** parameter appears on the channel block's faceplate.

13 Double-click the **SIMMODE** parameter on the faceplate and select one of the following options.

a **None** - If simulation is not required.

b **DirectSub** - If you want the operator to write the target value manually.

c **SimValSub** - If you want the operator to write the target value or if the target value can be fetched from other program.

Note: You can select the simulation mode while configuring the channel or after loading the module.

14 Double-click the **SIMVALUE** parameter on the faceplate and enter the simulation value.
When you enter the **SIMVALUE**, the same value is reflected in the PV parameter.

Note: You can enter the simulation value only after loading the module.

15 Click **OK**.
You can activate the CEE to start the simulation.

Attention
The Series C Pulse Input Module also supports simulation with SIM-C300.

5.2.8 Defining the FTEB for SIMC300

The following rules apply to the FTEB for SIMC300:

- Maximum number of IOMs allowed to be assigned to the same FTEB is **16**.
- Maximum number of FTEBs allowed to be used for a particular SIMC300 is **8**.
- Maximum number of IOMs allowed to be loaded to a particular SIMC300 in total is **64**.
- Redundant FTE Bridge is not allowed.

There is a load failure for the IOM for any violation of the above rules.

5.3 Creating a Control Module for simulation

The SIMC300 environment allows building a Control Strategy, where a Control Module (CM) is created and function blocks are inserted and connected with other function blocks.

WARNING
All edits done on project-related objects must be reloaded to the controller before those edits can be seen in the controller. See *Control Strategy Loading* for information on how to load control strategy objects.

To create a Control Module for simulation

- Refer to the following, *To create and save a Control Module* in PDF Collection for the procedure to configure the IOLINK block for simulation.

5.4 Assigning a Control Module and IOM to a CEEC300 block in a SIMC300 environment

Once a Control Module (CM) or Sequential Control Module (SCM) is created, you can assign it to a CEEC300 block of a SIMC300. Use the following procedure as a general guide to assign configured CMs and I/O Modules (IOMs) to the CEEC300 block.

Note that in the SIMC300 controller environment, Chassis IOMs and Rail IO modules can be assigned only to the CEEC300 block.

For additional information on CM assignment, see the *Control Building Guide*; or for SCM assignment, the *Sequential Control User's Guide*.

Attention

- Before Control Builder allows you to associate an IOM to an IOCHANNEL block, it checks to make sure that the CM and IOM are assigned to the same CEE
- All edit windows (such as CM charts) must be closed before proceeding with this procedure or a lock contention may occur. To resolve these types of lock contentions, close the open CM chart and attempt to open the CM chart again.

To assign a Control Module and IOM to a CEE300 block in a SIMC300 environment

1 The steps to assign a Control Module and IOM to a CEEC300 block are the same in the on-process and simulation environment.

2 Refer to the following, *To assign Control Modules and IOMs to a CEE* for the procedure to activate the CEE for simulation.

5.5 Assigning an IOP to an IOLINK in a SIMC300 environment

To assign an IOP to an IOLINK block in a SIMC300 environment

1. The steps to assign a Control Module and IOM to a CEEC300 block are the same in the on-process and simulation environment.
2. Refer to the following, *To assign Control Modules and IOMs to a CEE* for the procedure to activate the CEE for simulation.

5.6 Adding an IO Channel to a Control Module in a simulated environment

An IO Channel block represents a channel in one of the various IO modules (IOMs). The IO Channel blocks can be added to a Control Module (CM) in the Project tab to build a process control strategy.

The procedure is a drag and drop operation and is the same for any type IO Channel, whether it is an AI channel in a Series C IO module or an DO channel in a Series A IO module.

To add IO Channel blocks to a Control Module in a simulated environment

- Refer to the following, *To add IO Channel blocks to a Control Module chart* for the procedure to activate the CEE for simulation.

5.7 Converting C300 to SIMC300

This section includes information that you use to convert a C300 to a SIMC300.

Topic	Link
To convert a C300 to a SIMC300	“Converting a C300 to a SIMC300” on page 32
To convert a redundant C300 to a SIMC300	“Converting a redundant C300 to a SIMC300” on page 33
To convert a redundant C300 Controller to a non-redundant controller	“Converting a redundant C300 Controller to a non-redundant controller” on page 33

Related topics

“Converting a C300 to a SIMC300” on page 32

“Converting a redundant C300 to a SIMC300” on page 33

“Converting a redundant C300 Controller to a non-redundant controller” on page 33

5.7.1 Converting a C300 to a SIMC300

The C300 and SIMC300 share the same template and configuration forms. The process to convert an existing C300 to a SIMC300 requires that you:

- Delete the C300 from your monitoring tab
- Define the C300 as a SIMC300
- Reload the controller as a SIMC300

To convert a C300 to a SIMC300

1 In the Control Builder Monitoring tab, delete all assigned blocks under the SIMC300/CEE/IOLINKs

2 In the Project window, double-click on the CEEC300 block icon.
Result: Calls up CEEC300 Block configuration form

3 From the Powerup Restart Settings section, click the CEE State's down-arrow button and select the IDLE state.

Result: This places both the C300 and the CEE in an IDLE state.

NOTE: The Simulation State of the C300/CEE/IOLINK blocks are not evaluated.

4 In the Project tab, double-click the C300 block icon that is to be placed in simulated.
Result: Calls up C300 configuration form

5 On the C300 configuration form verify the following:

a Type in desired name for the SIMC300 of up to 16 characters or accept the default in the **Tag Name** field.

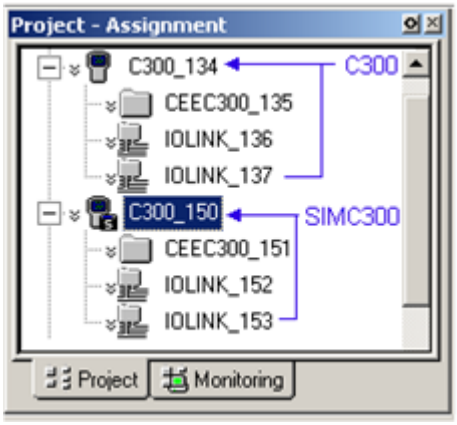
b Click in the **Device Index** field (DEVICEIDX parameter). Type a non-used odd integer number in the range of 1 through 509.

6 Check the **Load Simulation Environment** (SIMTARGET parameter) check box.

Note: Unchecking the **Load Simulation Environment** check box causes the SIMC300 to revert back to C300. All configuration options (except Module redundant configuration) that are specific to SIMC300 are applicable to C300.

7 Type the IP address (or the host name of the node) of where the SIMC300 is located.

Result: The following is an example of the SIMC300 block icons that now appear in the project window:



8 All additional configuration selections for the SIMC300 are identical in the manner to which the C300 selections are made (with the exception of redundancy determination).

Refer to the C300 Controller User's Guide for support with these selections.

5.7.2 Converting a redundant C300 to a SIMC300

Configuring a SIMC300 as a redundant controller is not allowed. An error message is returned.

Converting a redundant C300 controller directly to a SIMC300 is also not allowed. An error message is returned.

The following is recommended:

1. You convert the redundant C300 to a non-redundant controller. Refer to the following: “Converting a redundant C300 Controller to a non-redundant controller” on page 33
2. And then convert the C300 to a SIMC300 controller. Refer to the following: “Creating a SIMC300 controller” on page 23

Note: Once the checkbox **Module is Redundant** (MODISREDUN parameter) is unchecked, the secondary controller is automatically deleted from database and the view tree.

5.7.3 Converting a redundant C300 Controller to a non-redundant controller

1 Disconnect the redundancy cable from the primary C300 Controller.

Result: Alarms are generated.

2 In the Monitor view, right click on the secondary C300 Controller block.

Choose delete.

Result: The secondary C300 Controller icon disappears from the tree view.

3 In the Project view, right click on the primary C300 Controller icon and choose Module Properties.

Result: Calls up the primary C300 Controller configuration form.

4 Uncheck the Module is redundant check box.

5 Click the OK button.

Result:

- The secondary C300 Controller icon is deleted from the project view,
- The non-redundant C300 Controller icon changes from a redundant to a non-redundant icon
- The non-redundant C300 icon shows a delta

6 Click non-redundant C300 block icon in Project view.

Perform a Load to the controller.

The delta sign should disappear from the C300 Controller icon in the Project view. The C300 Controller icon in the Monitor view should indicate the controller is now non-redundant.

5.8 Understanding SIMC300 parameters

This section includes information that you use to understand SIMC300 parameters.

Topic	Link
C300 parameters used for simulation	“C300 parameters used for simulation” on page 35
SIMTARGET	“SIMTARGET” on page 36
SIMCOMMAND	“SIMCOMMAND” on page 36
SIMSTATE	“SIMSTATE” on page 36
SPDFACTOR	“SPDFACTOR” on page 37
Supported PM IO functionality for SIMC300	“Supported PM IO functionality for SIMC300” on page 37
Non-supported PM IO functionality for SIMC300	“Non-supported PM IO functionality for SIMC300” on page 37

Related topics

“C300 parameters used for simulation” on page 35
“SIMTARGET” on page 36
“SIMCOMMAND” on page 36
“SIMSTATE” on page 36
“SPDFACTOR” on page 37
“Supported PM IO functionality for SIMC300” on page 37
“Non-supported PM IO functionality for SIMC300” on page 37

5.8.1 C300 parameters used for simulation

Parameter	Description
Configuration parameters	
SIMTARGET	The environment is: C300 if SIMTARGET = FALSE SIMC300 if SIMTARGET = TRUE
HOSTIPPRI	The host IP address where the SIMC300 is running.
HOSTNAMEPRI	The host name where the SIMC300 is running.
PROCESS_ID	The windows process ID is assigned to the SIMC300 when it was created.
SIMCOMMAND	Commands the simulation state.
Run-time parameters	
SIMSTATE	The current state of simulation environment.
TNUMSC3INCON	Number of other C300s/SIMC300s this SIMC300 is connected to as an initiator.

Parameter	Description
TNUMSC3OUTCON	Number of other C300s/SIMC300s this SIMC300 is connected to as a responder.
STEPTIME	The step execution time in ms.
SPDFACTOR	The target Base cycle speed factor.

5.8.2 SIMTARGET

SIMTARGET parameter is located on the C300 configuration form's Main tab. The text appears as “Load to simulation”. The following is true:

- For SIMC300 - Checking the checked box equals SIMTARGET = TRUE
- For C300 - Unchecking the check box equals SIMTARGET = FALSE

5.8.3 SIMCOMMAND

SIMCOMMAND is used to change the state of simulation and can be issued only when SIMTARGET = TRUE

When SIMCOMMAND is	Then
SIMNONE	C300 is *not* in simulation mode and the blocks are being executed. If the SIMC300 is not in simulation (SIMSTATE is NONE), the FREEZE command is rejected, and an error is logged in Experion error log.
SIMRUN	C300 is in simulation mode and the blocks are being executed. UniSim switches SIMSTATE from SIMFREEZE to SIMRUN.
SIMFREEZE	UniSim stops simulation. UniSim issues this command when a save/restore dynamic data snapshot or for Single/ Multi-Step execution is needed. Only the UniSim simulator can successfully change the SIMC300's SIMSTATE to SIMFREEZE.
SIMDISABLE	If UniSim freezes the simulation, then the Engineer can use SIMDISABLE to unfreeze the simulation. On receiving this command SIMSTATE is set to SIMRUN.

5.8.4 SIMSTATE

SIMC300 checks the SIMSTATE value to make sure it is in simulation. The CEE and IOLINKs are also placed in simulation and share the same SIMSTATE as the SIMC300.

Attention

- In C300, SIMSTATE is always SIMNONE.
- In SIMC300, SIMSTATE is never SIMNONE.

When SIMSTATE is	Then
SIMNONE	C300 is *not* in simulation mode. Default setting for the C300.
SIMRUN	C300 is in simulation mode and the blocks are being executed. UniSim or the engineer switches SIMSTATE from SIMFREEZE to SIMRUN.

When SIMSTATE is	Then
SIMFREEZE	UniSim stops simulation to do a Save/Restore Dynamic Data Snapshot or for Single/Multi-Step execution. After a successful Save or Restore, UniSim commands SIMSTATE to SIMRUN. Note: Only the UniSim simulator can successfully change the SIMC300's SIMSTATE to SIMFREEZE.
SIMDISABLE	There is no SIMDISABLE state.

5.8.5 SPDFACTOR

UniSim commands SIMC300 to run slower or faster by setting the value of SPDFACTOR parameter. This parameter is processed only when this SIMC300 controller is in SIMFREEZE.

- If the value is out of the range, the request is rejected with an error indicating the value is out of range. Since SPDFACTOR is a float value, the value requested by a user may not be valid. Therefore the value of parameter SPDFACTOR must be adjusted to the nearest supported value.
- If the value of SPDFACTOR that UniSim commands is within the supported range (0.01 to 5), the value is reset.
- The controller will run at the same speed until the SPDFACTOR is reset.

5.8.6 Supported PM IO functionality for SIMC300

PM IO in SIMC300 is configured the same as PM in the C300 environment. This means that:

- The IOP is allowed to be assigned and loaded to the SIMC300
- The IOP can be configured as redundant or non-redundant. This allows a redundant on-process C300 to retain its original configuration in SIMC300, although the redundancy capability is not supported in SIMC300.
- The PV can be simulated
- A close loop tail-to-mouth wiring can be configured

5.8.7 Non-supported PM IO functionality for SIMC300

The following PM IO functionality is not supported for SIMC300 since the actual hardware is required to invoke these functions:

- Alarming and events (including IOM soft failures, SOE events, etc.)
- HART device parameters
- Smart Transmitter device specific parameters (PM IO only)
- IO redundancy operations (Swap, DisableSync, EnableSync, etc.)
- IOM hardware and maintenance related parameters
- IOM statistics (CPU free averages, minimums, etc.)

6 SIMC300 - Operations

Related topics

6.1 Activating CEE in a SIMC300 environment

For the SIMC300 to begin executing its control strategy, you must activate the CEEC300 block.

Topic	Link
Initial activation order guidelines for SIMC300 CEE	“Initial activation order guidelines for SIMC300 CEE” on page 40
To activate a CEE in a SIMC300 environment	“To activate CEE in a SIMC300 environment” on page 40

6.1.1 Initial activation order guidelines for SIMC300 CEE

Make the initial activation of control strategy components in Control Builder from the Monitoring tab in the following suggested order to minimize possible bad data generated alarms.

Order	Component
1	Control environment components such as: CEEC300 ACE/CEE
2	Process Manager IOMs
3	PMIO I/O Channels
4	Fieldbus contained function blocks
5	Fieldbus device resident blocks
6	Input/Output Modules (IOMs)
7	Control Modules (CMs) and/or Sequential Control Modules (SCMs)

6.1.2 To activate CEE in a SIMC300 environment

The steps to activate the CEEC300 block is the same for both in the on-process and simulation environment.

Refer to the Activating the CEE for the procedure to activate the CEE for simulation.

6.2 Similarities between loading SIMC300 and C300

The C300 User's Guide provides an extensive list of Loading C300 Controller Configuration information that also applies to the SIMC300. It provides sections for the following main topics:

- About load operations
- Initial load order guidelines
- Load components from Project
- Load With Contents command
- Reloading components from project
- Restrictions and conditions for reloading operations
- Upload to the Monitoring database

Refer to the *Load C300 Controller Configuration* for the complete listing of Loading C300 Controller Configuration topics.

6.3 Inactivating the CEE in a SIMC300 environment

The steps to inactivate the CEEC300 block is the same for both in the on-process and simulation environment. Refer to *Setting the CEE inactive* for the procedure to inactivate the CEE for simulation.

6.4 Shutting down the SIMC300

To properly shutdown a SIMC300 the following needs to occur:

- The CEE state is set to IDLE. This also places the SIMC300 in an IDLE state.
- The Simulation state (SIMSTATE) of the C300 block is set to SIMRUN.
- From the Monitoring tab a shutdown command is issued from the C300 configuration form.

Tip

Once the SIMC300 shutdown command is processed:

- all the blocks inside of the SIMC300 become red, excluding the SIMC300
- SIMC300.exe exits from the task manager

To shutdown a SIMC300

1 In the Monitoring window, double-click on the CEEC300 block icon.
Result: Calls up CEEC300 Block configuration form

2 From the Command/State section, click the CEE Command's down-arrow button and select the IDLE state.

3 From the Powerup Restart Settings section, click the CEE State's down-arrow button and select the IDLE state.

Result: This places both the C300 and the CEE in an IDLE state.

4 In the Monitoring window, double-click on the SIMC300 block icon.
Result: Calls up C300 Block configuration form.

5 From the Simulation Node Operation section, click the SIM Command's down-arrow button and select the SIMDISABLE state.

6 In the Monitoring window, double-click on the SIMC300 block icon.
Result: Calls up C300 Block configuration form..

7 On the Main tab, click Controller Command box and select Shutdown from the list.

8 Click the Yes button to confirm the action.

9 This completes the configuration procedure for shutting down the SIMC300.

6.5 Deleting a SIMC300

To delete a SIMC300

To properly delete the SIMC300 the following needs to occur:

- Delete all assigned blocks under SIMC300/CEE/IOLINKs
- The CEE state is set to IDLE. This also places the SIMC300 in an IDLE state.

Tip

Once the SIMC300 is deleted:

- The CEE/IOLINK for that SIMC300 is also deleted
- SIMC300.exe exits from the task manager

1. In the Control Builder Monitoring tab, delete all assigned blocks under the SIMC300/CEE/IOLINKs
2. In the Project window, double-click on the CEEC300 block icon.

   Result: Calls up CEEC300 Block configuration form
3. From the Command/State section, click the CEE Command's down-arrow button and select the IDLE state.
4. From the Powerup Restart Settings section, click the CEE State's down-arrow button and select the IDLE state.

   Result: This places both the C300 and the CEE in an IDLE state.

   NOTE: The Simulation State of the C300/CEE/IOLINK blocks is not evaluated.
5. In the Project tab, right-click the SIMC300 icon and select Delete from the menu.
6. This completes the configuration procedure for deleting the SIMC300.

6.6 SIMC300 operator displays

C300, C300 CEE and C300 IOLINK share the same detail display and group faceplate as their simulation counterpart displays SIMC300, SIMC300 CEE, and SIMC300 IOLINK. The displays have been modified to be able to indicate whether it is a simulation or on-process platform.

- When the detail display is used for a simulation, “Simulation” is displayed at the bottom of the detail display.
- When the detail display is used for a simulation and it is in SIMFREEZE, “SIMFREEZE” will be displayed at the bottom of the detail display.

When the detail display is used for a C300, the detail display retains its non-simulation appearance.

C300	SIMC300 in SIMRUN	SIMC300 in SIMFREEZE
C300 	SIMC300 in SIMRUN 	SIMC300 in SIMFREEZE 
C300 CEE 	SIMC300 CEE in SIMRUN 	SIMC300 CEE in SIMFREEZE 
C300 IOLINK 	SIMC300 IOLINK in SIMRUN 	SIMC300 IOLINK in SIMFREEZE 

6.7 Save and restore Snapshot data

This section includes information that you use to save and restore Snapshot data.

Topic	Link
Type of snapshots	“Type of snapshots” on page 46
Reviewing snapshot rules	“Reviewing snapshot rules” on page 47
To create/save a static snapshot	“Creating/saving a static snapshot” on page 48
To restore an older snapshot	“Restoring an older snapshot” on page 48
To create/save a dynamic snapshot	“Creating/saving a dynamic snapshot” on page 49
To restore/load a dynamic snapshot	“Restoring/loading a dynamic snapshot” on page 49

Related topics

“Type of snapshots” on page 46
“Reviewing snapshot rules” on page 47
“Checkpoint operations that can be launched from Control Builder” on page 47
“Creating/saving a static snapshot” on page 48
“Restoring an older snapshot” on page 48
“Creating/saving a dynamic snapshot” on page 49
“Restoring/loading a dynamic snapshot” on page 49

6.7.1 Type of snapshots

By doing a snapshot save you capture the state of a controller at a particular time, allowing:

- for SIMC300 to be able to backtrack to a previous simulation state
- or be able to recover from a power loss scenario

A successfully saved snapshot file can be used to restore the controller to the state at the time it was saved.

Type of snapshot	Description
Structural Snapshot (checkpoint file)	The structural snapshot file contains the controller configuration data plus additional run-time data sufficient to do a warm start. PURPOSE OF A STRUCTURAL SNAPSHOT: Structural snapshots are used to store a binary image of the simulation database. They contain model configuration data as well as state variables. They can be thought of as a database dump or a memory map of the simulation database. Refer to the following: “Creating/saving a static snapshot” on page 48.

Type of snapshot	Description
Non-Structural Snapshot (Dynamic)	The non-structural snapshot file contains the runtime data sufficient to do a hot start of a controller. PURPOSE OF A DYNAMIC SNAPSHOT: During a hot start, the dynamic state of the controller is retained and is also referred to as a dynamic snapshot. It does not contain configuration data, and must be restored to a controller that is identically configured as the one from which the non-structural snapshot was saved. Note: Only UniSim can command a non-structural snapshot. Refer to the following: “Creating/saving a dynamic snapshot” on page 49.
Backtrack Snapshot	A Backtrack Snapshot is a UniSim snapshot set that consists of the: • controllers' non-structural snapshots and • UniSim models' dynamic state PURPOSE OF A BACKTRACK SNAPSHOT: With backtrack snapshots you can reset a model's conditions in order to try various operating techniques, to correct a process upset, or to allow the trainee to practice a procedure several times. To backtrack means to reset both the simulation time and simulation conditions to an earlier state. Backtrack snapshots persist only while a model is open. Once you close a model, all backtrack snapshots are automatically deleted. Refer to the UniSim Operations R310 Help for instructions on creating, deleting and importing instructions a backtrack snapshot.
ASCII Snapshot	An ASCII Snapshot is a UniSim snapshot that consists of: • the controllers' non-structural snapshots and • UniSim models' dynamic state that can be used between simulation sessions. PURPOSE OF ASCII SNAPSHOT: They are an engineering feature, designed to be used for troubleshooting, for migrating model files to a new version of UniSim Operations, or for integrating structural changes to a model. Do not use ASCII snapshots for training purposes or in place of initial condition snapshots. Refer to the UniSim Operations R310 Help for instructions on creating, deleting and importing instructions an ASCII snapshot.

6.7.2 Reviewing snapshot rules

When the controller is in the middle of snapshot, UniSim requests a Save operation. If there is a Checkpoint Restore also happening, these operations may be attempting to function at the same time. To eliminate the possibility of a conflict between a Save activity and a Restore activity happening simultaneously, the following rules must be followed:

If	Then
The controller is doing a snapshot restore	a checkpoint save is not allowed
The controller is doing a snapshot save	a checkpoint restore is not allowed
The controller is doing a snapshot restore	a checkpoint restore is not allowed
The controller is doing a snapshot save	a checkpoint save is allowed

6.7.3 Checkpoint operations that can be launched from Control Builder

The following table summarizes the checkpoint operations that can be launched from Control Builder.

Initiate This Operation . . .	If You Want To . . .
On **Controller** menu or right-click node, click **Checkpoint** > **Schedule Checkpoint Tasks**	Create tasks that periodically checkpoint all the nodes listed in the task. The complete functionality of the **Checkpoint Scheduler** dialog can only be launched from Control Builder.
On **Controller** menu or right-click node, click **Checkpoint** > **Archive Checkpoint Files**	Archive desired set of checkpoint files to a different location.
On **Controller** menu or right-click node, click **Checkpoint** > **Save Checkpoint Manually**	Checkpoint a selected node and store the current configuration and operational data. In addition, you can save a group of nodes based on previously configured manual tasks.
On **Controller** menu or right-click node, click **Checkpoint** > **Restore From Checkpoint**	Restore a failed hardware node or return to a previous configuration/operational condition by choosing a checkpoint file from among multiple checkpoint files. For certain nodes, you can selectively restore lower level hardware modules.
On **Controller** menu or right-click node, click **Checkpoint** > **Rebuild Selected Object(s) and Contents Checkpoint from Monitoring Tab**	Re-construct both the checkpoint base information (CCD info) of that node, and regenerate the node's Latest.cp checkpoint file with just the configuration information. The information for doing this is obtained from that node's loaded information in the Monitor side of the ERDB.

6.7.4 Creating/saving a static snapshot

Creating/Saving a static snapshot

The structural data snapshot saving and restoring is, but commanded by Control Builder or Station.

Tip

All the structural information is saved when the controller is first loaded, and information that can be changed during run-time is saved in the snapshot file. Therefore, a periodic manual checkpoint file save is recommended, but not required.

To create/save a static snapshot

1. In Control Builder with the SIMC300 selected, right-click the node, select **Indent1oint -** > **Save Indent1oint Manually.**
   Result: Save Indent1oint Manually … dialog opens
2. Do one of the following:
   In the Available field select the SIMC300 to be saved. Click the right-arrow key to move the SIMC300 to the To Be Saved field
   Select the SIMC300 in the To Be Saved field.
3. Click Save,
4. This completes the procedure for creating/saving the SIMC300.

6.7.5 Restoring an older snapshot

Once the snapshot restore is completed, UniSim issues a notification recover command to the controller. The snapshot restore event, notification recovery begin event and notification recover end event is saved in server event log.

Tip

A Checkpoint restore can't be performed when a snapshot restore is planned, since both operations write to the controller.

The controller has to be in IDLE and IOM has to be in INACTIVE in order to perform a Checkpoint restore.

To restore an older snapshot

1 In Control Builder with the SIMC300 selected, right-click the node, select **Checkpoint** > **Restore From Checkpoint.**
Result: Restore From Checkpoint … dialog opens

2 Do one of the following:
   a In the left-hand window, select one of the listed controller files.
   b Click Browse to locate the file to restore.

3 Click the proper Restore Scope selection

4 Click Restore.

5 This completes the procedure for restoring the SIMC300.

6.7.6 Creating/saving a dynamic snapshot

The non-structural data snapshot saving and restoring is commanded by UniSim.

Tip
SAVING DYNAMIC SNAPSHOT:

Once the dynamic snapshot is saved all the active alarms get time stamped and saved.

RESTORING DYNAMIC SNAPSHOT:

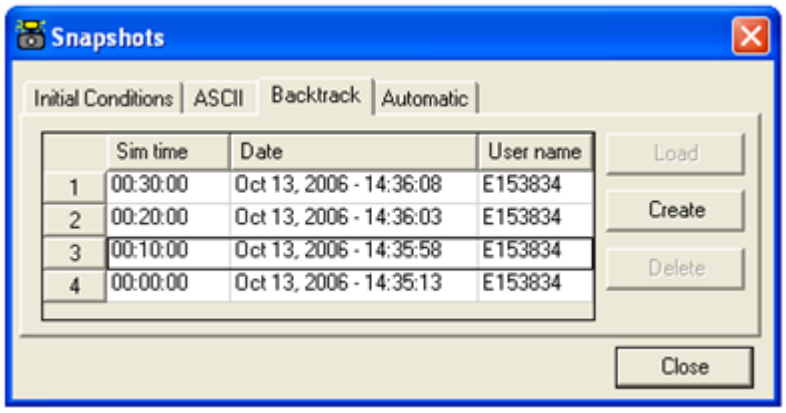
When the dynamic snapshot is restored, the current alarms get cleared and the alarms that were present while the snapshot was saved appear with the timestamp.

To create/save a dynamic snapshot

1 In UniSim select **File** > **Snapshot.**
Result: Snapshot dialog opens

2 Click the Backtrack tab.

3 Click Create.



Result: A new snapshot is created and the Snapshot dialog box is updated.

4 This completes the procedure for creating/saving the dynamic snapshot.

6.7.7 Restoring/loading a dynamic snapshot

The non-structural data snapshot restoring and loading is commanded by UniSim.

Tip

SAVING DYNAMIC SNAPSHOT:

Once the dynamic snapshot is saved all the active alarms get time stamped and saved.

RESTORING DYNAMIC SNAPSHOT:

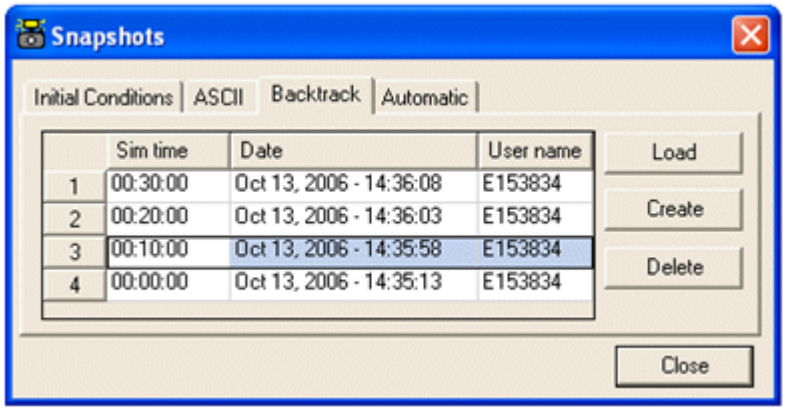
When the dynamic snapshot is restored, the current alarms get cleared and the alarms that were present while the snapshot was saved appear with the timestamp.

To restore/load a dynamic snapshot

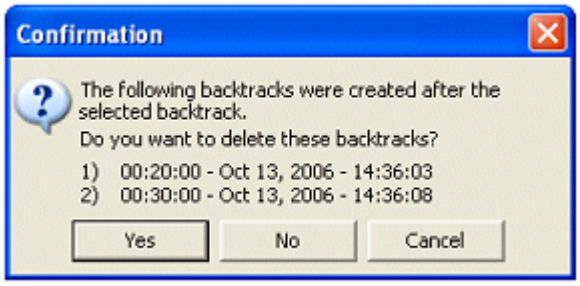
1 In UniSim, select **File -** > **Snapshot.**
Result: Snapshot dialog opens

2 Click the Backtrack tab.

3 Click the snapshot you wish to load:



4 Click Load.



Result: Confirmation dialog box appears.

5 Select one of the following:

- **a** Click Yes to delete the listed backtrack snapshots and load the newly selected snapshot.
- **b** Click No to not delete these backtrack snapshots and load the newly selected snapshot.
- **c** Click Cancel to do neither.

6 This completes the procedure for restoring the SIMC300.

6.8 UniSim operations

Please refer to the following sources for UniSim installation, configuration, and operational information that are not included in this document:

- UniSim documentation - installation and node administration information residing in the UniSim bookset within PDF Collection.
- UniSim Operations R310 Help - online help within the UniSim application
- UniSim Operations Shadow Plant Configuration Guide - online help within the UniSim application

6.8.1 Type of UniSim operations

UniSim sends commands to the SIMC300 in the same manner as the SIMC200E. It requests the status of the SIMSTATE parameter to determine if SIMC300 is in SIMFREEZE or not.

UniSim commands such as: STEP, STEPTOPHASE, STEPTIME, FREEZE and UNFREEZE only impact pulse-related scenarios. They are not meaningful to either PM and Series C IO, since there is no control algorithm executed in IO.

6.8.2 Changing the execution speed of SIMC300

By setting the parameter SPDFACTOR, UniSim commands the SIMC300 to run faster or slower.. The following is true:

- This is only done when the SIMC300 controller is in SIMFREEZE.
- If the value is out of the range, the request is rejected with an error indicating the value is out of range.
- If the value is within the supported range (0.01 to 5), and since SPDFACTOR is a float value: there is a chance that the value requested by user is not applicable. In this case, the value of parameter SPDFACTOR needs to be adjusted to the nearest supported value.
- The controller keeps running at the same speed until the SPDFACTOR is reset.
- The value of SPDFACTOR gets reset only when a command with a new acceptable SPDFACTOR is sent from UniSim.

Execution speed and STEP Operations

STEP command type	Description
STEP	Controller still executes the exact number of steps as what UniSim commands even if the SPDFACTOR is something other than 1X. The controller finishes this command faster or slower based on the value of SPDFACTOR
STEPTOPHASE	Controller still executes to the phase that UniSim commands even the SPDFACTOR is something other than 1X. The controller finishes this command faster or slower based on the value of SPDFACTOR.
STEPTIME	STEPTIME is converted to cycles.

6.8.3 Defining the STEP operations

The SIMC300 controller must be set to SIMFREEZE to allow a UniSim command to be set to STEP, STEPTOPHASE, or STEPTIME.

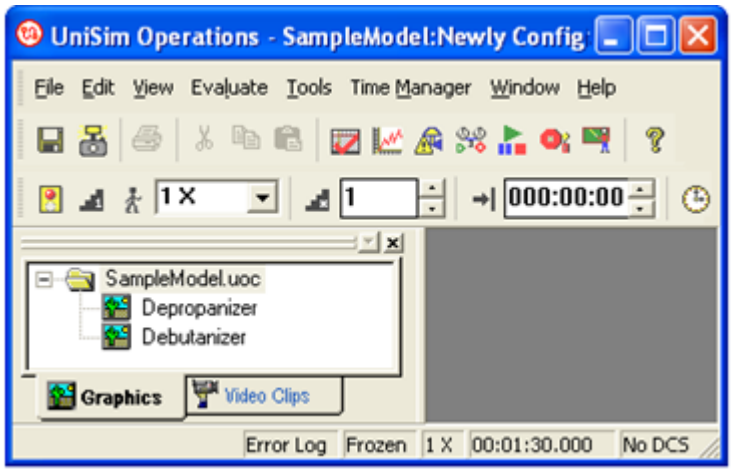
STEP command type (value)	Description
STEP (single or multi-stepping)	The STEP command can be a single-stepping or a multi-stepping. First, the SIMC300 is placed in SIMRUN and then one of the following is executed: • If it is single-stepping: the SIMC300 is executed one cycle and then the SIMC300 is placed SIMFREEZE. • If it is multi-stepping: SIMC300 is executed the number of cycles that is commanded. (Parameter STEP is counting down while the SIMC300 is executing.) Once the parameter STEP reaches to ZERO the SIMC300 is placed in SIMFREEZE. NOTES: • While SIMC300 is in SIMRUN, other STEP, STEPTOPHASE or STEPTIME commands are rejected. • The only way to abort the current STEP command is to issue an SIMFREEZE command to SIMCOMMAND. This resets STEP to ZERO.
	EXAMPLE If a **single step** command is issued: • and the current frozen cycle is 20, SIMC300 CEE is executing in cycle 20. • SIMC300 CEE will be in SIMFREEZE again in cycle 21. If a step of 5 **(multi-stepping)** command is used: • and the current frozen cycle is 20, SIMC300 CEE will start from cycle 20 and stop at cycle 25 • CEE will execute in cycle 20, 21, 22, 23 and 24, and then SIMC300 CEE is in SIMFREEZE again in cycle 25.
STEPTOPHASE (0 to 39)	First, the SIMC300 is placed in SIMRUN until the PHASE that is requested by the STEPTOPHASE command is reached. Then the SIMC300 CEE is placed in SIMFREEZE again. NOTES: • While SIMC300 is in SIMRUN, other STEPTOPHASE, STEP, or STEPTIME commands are rejected. • The only way to abort the current STEPTOPHASE command is to issue an SIMFREEZE command to SIMCOMMAND.
	EXAMPLE: If a STEPTOPHASE command is issued: • and the current frozen cycle is 20 with a value of 3 • SIMC300 CEE will be running from cycle 20 to 39, then from 0 to 2. It will stop at cycle 3. At this point, SIMC300 CEE is in SIMFREEZE.

STEP command type (value)	Description
STEPTIME (milliseconds)	SIMC300 converts from millisecond to the execution cycle based on the base cycle of SIMC300. STEPTIME command: • is accepted only if the STEPTIME is smaller than the base cycle and SIMC300 is executing for the amount of time that the STEPTIME command requested. • while the SIMC300 is executing, the parameter STEPTIME is counting down by the base cycle. Once the STEPTIME is less than the base cycle, the SIMC300 is placed in SIMFREEZE, and the remaining is discarded. NOTES: • While SIMC300 is in SIMRUN, other STEPTIME, STEP, or STEPTOPHASE commands are rejected. • The only way to abort the current STEPTIME command is to issue an SIMFREEZE command to SIMCOMMAND. This resets STEP to ZERO.
	EXAMPLE: If STEPTIME command is issued: • and the current frozen cycle is 20 with a value of 15. SIMC300 will start running. • and the current frozen cycle is 20 with value of 110, SIMC300 will be running in cycle 20 and 21,and will stop at cycle 22.

6.8.4 To freeze/unfreeze the simulation

Using UniSim you can and stop (freeze) and start (unfreeze) a model whenever you want. Stopping a simulation means stopping the simulation time and halting current processes.

The current state of the model simulation, either frozen or unfrozen, is displayed on the status bar.



The state is also indicated by the image of the Freeze/Unfreeze icon on the Time Manager toolbar.

Icon	Description
	Indicates the state is frozen. Clicking again unfreezes the simulation
	Indicates the state is unfrozen. Clicking again freezes the simulation

7 SIMC300 - Maintenance

Related topics

“Upgrading SIMC300 software” on page 56
“Migrating SIMC300” on page 57

7.1 Upgrading SIMC300 software

You do not just upgrade the SIMC300 software to a newer version without upgrading the entire Experion application. Please refer to the SIUG installation instructions for upgrading the Experion software.

7.1.1 To upgrade SIMC300 software - general guidelines

The following items are general guidelines to upgrade the SIMC300 software.

- IDLE the SIMC300 CEE.
- Shutdown the SIMC300.
- Install the new version of SIMC300 software.
- Reload the controller and its contents from the project side.
- Restore snapshot data from UniSim.

7.2 Migrating SIMC300

This section includes information that you use to migrate a SIMC300.

Topic	Link
Migration support for SIMC300	“Migration support for SIMC300” on page 57
To migrate a SIMC300	“Migrating a SIMC300” on page 57

7.2.1 Migration support for SIMC300

The simulation environment does not support on-process migration (for SIMC300 or any other simulation environment). SIMC300 migration means carry database and dynamic data from one release to another. There is no dependency on checkpoint for dynamic save/restore. The database can be carried over by:

- database migration during server migration, or by
- importing the exported strategies directly

Dynamic data and initial state can be carried over as long as the migration path is supported from one release to the other.

Tip
Backtrack data has to be saved either as dynamic data or initial state in order to be carried to the new release.

7.2.2 Migrating a SIMC300

Tip
Please refer to the Supplementary Installation Tasks Guide (SIUG) and the Experion Migration User's Guide for instructions to migrate the Experion software.

To migrate a SIMC300

1 Back up the database

If you chose to migrate your database, no additional operation is required. The migration activity handles database migration.

If you chose not to migrate your database, your existing simulation strategies need to be exported and saved before migrating the Experion application.

2 Back up DSD data

Save the dynamic data and initial state.

3 Upgrade software

Migrate the EPKS server, station, UniSim node and SIMC300 by following the SIUG and the Experion Migration User's Guide instructions.

NOTE: Make sure the database migration option is selected if the database needs to be migrated.

4 Restore database

If you chose to migrate your database, no additional operation is required. The migration activity handles database restoration.

If you chose not to migrate your database, your previous exported simulation strategies need to be imported.

5 Load the controller

The controller has to be reloaded from monitoring side since checkpoint data is not migratable.

6 Restore DSD data

Start up OTS session, and then restore DSD from saved dynamic file.

8 SIMC300 - Troubleshooting

Related topics

“Fixing common problems” on page 60
“SIMC300 alarms” on page 61
“Recovering from CDA disconnect” on page 63
“Reporting SIMC300 problems to Honeywell” on page 64

8.1 Fixing common problems

This section identifies some common problems and describes how you might fix them.

Refer to *C300 Controller User's Guide* for a complete list on how to fix Series C C300 (SIMC300) common problems.

Related topics

"Loss of power" on page 60
"Simulation environment was not installed" on page 60

8.1.1 Loss of power

The power supply has failed or the main power source has been shut down or is experiencing a *brownout* or *blackout* condition.

Diagnostic check

- In the **Monitoring** tab, the SIMC300 Controller icon turns red.

Cause

Main power source has been disconnected or shut down either manually or temporarily by *brownout* or *blackout* condition.

Solution

The SIMC300 is recovered by restoring the previous saved checkpoint restore from Control Builder or station.

8.1.2 Simulation environment was not installed

The SIMC300 simulation environment was not included as an installation selection during the install.

Diagnostic check

Attempting to create and load a SIMC300 fails because a simulated controller is not recognized by the system

Cause

Improper selection options during the Experion install.

Solution

There are multiple selection options when installing the simulation environment for an Experion system.

Refer to the *Supplementary Installation* Tasks Guide (SIUG) for assistance in installing the simulation environment

8.2 SIMC300 alarms

The C300 User's Guide provides an extensive list of Troubleshooting information that also applies to the SIMC300. It provides sections for the following main topics:

- Initial checks
- Fixing common problems
- Status and fault messages
- Startup fault scenarios

Refer to the *C300 User's Guide* for the complete listing of C300 Troubleshooting topics.

8.2.1 Notifications

SIMC300 transmits alarms in the manner as the C300. The following information is specific to the SIMC300 notifications:

- CPULO - CPU Free Low Alarm

  When the CPU resources fall to less than the configured limit, a CPULO diagnostic alarm is generated. All SIMC300s that are running on this node report a CPULO alarm. The CPU resources will be obtained from windows task manager.
- OVERTEMPTHLD - Temperature Threshold Alarm

  The following applies for the parameter OVERTEMPTHLD with regards to SIMC300:

  – Default value of the limit 80
  – Disabled temperature alarm 0
  – Default value of the temperature 42

The actual temperature value displayed via SIMC300 is less meaningful since it is not changeable. The value can be manipulated and a temperature alarm can be simulated when the limit is set to a value lower than the default value.

8.2.2 Cycle Overruns

A cycle overrun occurs if the current cycle doesn't complete its execution before the next cycle starts. An alarm is:

- generated when there is accumulated cycle overruns and
- is cleared when there is accumulated time without any cycle overruns

This reporting and clearing of the alarms is based on the multiple of the macro cycle.

- A CEE cycle overrun alarm generates when there are overruns in two consecutive macro cycles.
- The CEE cycle overrun alarm gets RTN'd when there are no overruns in four consecutive macro cycles and the CEE is currently in CEE cycle overrun alarm

8.2.3 CEE Notification

The CEE is responsible for reporting the following notifications. This is the same on both C300 and SIMC300.

- CEE state change - A notification is sent to the server when there is a CEE function state change to respond the new value of parameter CEESTATE
- Memory alarm - a memory limit exceeded alarm is generated by the CEE when the memory usage exceeds the total memory.

  – If the current free memory on the Memory tab shows a negative, an alarm is generated

– When the usage of memory returns to normal, in another word if the currently free memory in Memory tab shows positive number, this alarm will get RTN'd

The parameter NOTIFINHIBIT (CEE Main tab) is introduced to determine if the CEE-wide notifications of a simulation controller are allowed to be detected, reported, and displayed. This parameter pertains to the simulation environments like SIMC300, SIMC200E and SIMACE.

NOTE: For on-process controller it can only be FALSE.

Once this parameter is checked, the notifications that belong to this SIMC300 CEE are detected and reported. The notifications that have been reported and displayed from this SIMC300 CEE are suppressed in Station. When this parameter is unchecked the notifications are detected and reported as a new alarm with the new time stamp.

8.3 Recovering from CDA disconnect

This section includes information that you use ro recover from a CDA disconnect.

8.3.1 Losing the CDA connection

When there is a lost connection between CDA server and the UniSim node: and if the connection is lost more than the CDA time-out tolerable duration (6 seconds), the connection information saved in CDA server for this UniSim node is removed.

8.3.2 Recovering from a CDA disconnect

In order to re-establish the connection between CDA server and UniSim, UniSim has to:

- exit the current simulation session, and
- restart the simulation session

8.4 Reporting SIMC300 problems to Honeywell

If an unexpected behavior is observed and technical assistance is required you should:

- collect the following information
- contact Honeywell TAC for any further information and directions

8.4.1 SIMC300 failures

General SIMC300 operation failure

Collect the following:

- The detail description on the sequence of events and operations that caused the problem or the steps that can be used to reproduce the problem.
- The information from the system that is having problem, such as: the network type, the network configuration, the release version and the patching level.
- The EPKS event logs named as **ErrLog_xx.txt** and **IXP_log_xx.**txt. They both are located under **C:\Documents and Settings\All Users\Application Data\Honeywell\Experion PKS**
- If FTE is the network type: the windows application, system and FTE event log.
- The strategy that triggered the problem.
- The screenshot(s) when problem the occurred.

Operation failure related to UniSim operation

In addition to the information collected to identify SIMC300 operation failures, it is also recommended to save the error log of UniSim.

The error log is found:

- under the same directory where the UniSim model is located
- or go to View -> Error Logs from the UniSim application

In addition to these logs, TAC may request the ErrLog_xx.txt file from the UniSim machine. This log records errors, reported during the save/restore of dynamic snapshots.

Controller crash

As any other win32-based controller, the crash dump of the SIMC300 is saved in directory C:\Program Files\Honeywell\Experion PKS\Engineering Tools\system\bin\user\controller name. It is named with a combination of Executable Name, Tagged Block Name, Executable Process ID and Crash Time (year/month/day/hour/minute/second).

When there is a crash, you send the crash dump along with the system application event log for investigation.