



Siemens
Industry
Online
Support

Library of Unit Control (LUC)

STEP 7 / WinCC Unified (TIA PORTAL V19)

<https://support.industry.siemens.com/cs/ww/en/view/109974940>

SIEMENS

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 under <https://www.siemens.com/cert>.

Table of contents

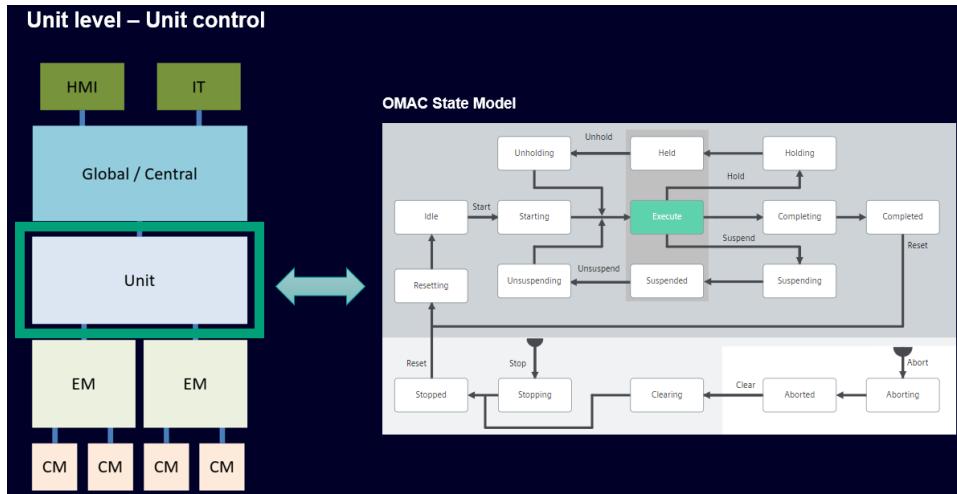
1	Introduction	5
1.1	Overview.....	5
1.2	General	6
1.3	Complementary Libraries	6
1.4	Hardware and software requirements	7
2	Library of Unit Control (LUC) Concept	8
2.1	OMAC State Model and LPM LV2022.....	8
2.2	ISA-88 Standard	9
2.2.1	Unit Level.....	10
2.2.2	Equipment Module Level	12
2.3	Structure and design of the library	13
2.3.1	Functions and Function Blocks FCs/FBs	13
2.3.2	User Defined Types UDTs.....	15
3	Program blocks	17
3.1	LUC_InterlinkToParent (FC / 1.2.0).....	17
3.2	LUC_InterfaceManager (FB / 1.2.0)	20
3.3	LUC_StatusCollector (FB / 1.2.0)	24
3.4	HelperFunctions	28
3.4.1	LUC_ModeStateToString (FC / 1.2.0)	28
3.4.2	LUC_SetUnitStatus (FC / 1.2.0)	30
3.5	History.....	34
3.5.1	LUC_UpdateModeStateHistory (FC / 1.2.0)	34
3.6	RemoteControl.....	36
3.6.1	LUC_RemoteCtrl_OpcUa (FB / 1.2.0)	36
3.7	RemoteControl / OPC_UA_Methods.....	39
3.7.1	LUC_OpcUaMethod_Abort (FB / 1.2.0)	39
3.7.2	LUC_OpcUaMethod_Clear (FB / 1.2.0)	41
3.7.3	LUC_OpcUaMethod_Complete (FB / 1.2.0).....	43
3.7.4	LUC_OpcUaMethod_Hold (FB / 1.2.0).....	45
3.7.5	LUC_OpcUaMethod_Reset (FB / 1.2.0).....	47
3.7.6	LUC_OpcUaMethod_SetUnitMode (FB / 1.2.0).....	49
3.7.7	LUC_OpcUaMethod_SetUnitState (FB / 1.2.0).....	51
3.7.8	LUC_OpcUaMethod_Start (FB / 1.2.0)	53
3.7.9	LUC_OpcUaMethod_StateComplete (FB / 1.2.0)	55
3.7.10	LUC_OpcUaMethod_Stop (FB / 1.2.0).....	57
3.7.11	LUC_OpcUaMethod_Suspend (FB / 1.2.0).....	59
3.7.12	LUC_OpcUaMethod_Unhold (FB / 1.2.0).....	61
3.7.13	LUC_OpcUaMethod_Unsuspend (FB / 1.2.0)	63
3.8	SignalStack.....	65
3.8.1	LUC_SignalStack (FB / 1.2.0)	65
3.9	StateMachineConfiguration	73
3.9.1	LUC_StateMachineConfig (FC / 1.2.0).....	73
4	PLC data types	75
4.1	LUC_typeCommands (UDT / V1.1.0).....	75
4.2	LUC_typeDiagnostics (UDT / V1.1.0).....	75
4.3	LUC_typeEmFeedback (UDT / V1.1.0)	75
4.4	LUC_typeInterlinkMode (UDT / V1.2.0).....	75

Table of contents

4.5	LUC_typeManagerInterface (UDT / V1.2.0)	76
4.6	LUC_typeModeRequests (UDT / V1.1.0)	76
4.7	LUC_typeModes (UDT / V1.1.0).....	76
4.8	LUC_typeMonitoring (UDT / V1.2.0)	77
4.9	LUC_typeStateCommands (UDT / V1.1.0)	78
4.10	LUC_typeStates (UDT / V1.1.0)	78
4.11	LUC_typeUnitModeStateTimes (UDT / V1.2.0).....	79
4.12	HelperFunctions	80
4.12.1	LUC_typeButtons (UDT / V1.2.0)	80
4.13	History.....	81
4.13.1	LUC_typeModeState (UDT / V1.2.0).....	81
4.14	RemoteControl.....	82
4.14.1	LUC_typeRemoteCtrlConfiguration (UDT / V1.1.0)	82
4.15	SignalStack	83
4.15.1	LUC_typeSignalStackByBool (UDT / V1.1.0).....	83
4.15.2	LUC_typeSignalStackByDInt (UDT / V1.1.0)	83
4.15.3	LUC_typeSignalStackByDWord (UDT / V1.1.0).....	83
4.15.4	LUC_typeSignalStackByStatus (UDT / V1.1.0).....	84
4.15.5	LUC_typeSignalStackConfiguration (UDT / V1.1.0).....	84
4.15.6	LUC_typeSignalStackStateBehaviour (UDT / V1.1.0)	85
4.16	StateMachineConfiguration	86
4.16.1	LUC_typeCompleteCommandConfiguration (UDT / V1.1.0).....	86
4.16.2	LUC_typeHoldCommandConfiguration (UDT / V1.1.0).....	86
4.16.3	LUC_typeModeConfiguration (UDT / V1.2.0)	86
4.16.4	LUC_typeModeTransitionConfiguration (UDT / V1.1.0)	87
4.16.5	LUC_typeStateConfiguration (UDT / V1.1.0)	87
4.16.6	LUC_typeStateMachineConfiguration (UDT / V1.2.0).....	88
5	Operation description (Operator).....	89
5.1	Mode Request Faceplate	89
5.2	State Command Faceplate.....	90
5.3	Interlink to parent faceplate	91
5.4	This library provides a faceplate to link or unlink an equipment module from a unit. The faceplate contains the following features:	91
5.5	Mode state history	92
5.6	Signal stack	92
5.7	State Model.....	97
6	LUC Custom Style.....	98
7	Appendix.....	99
7.1	Service and support.....	99
7.2	Links and Literature	100
7.3	Change documentation	101
7.4	Change log	102

1 Introduction

1.1 Overview



The Library of Unit Control (LUC) provides function blocks that simplify LPML OMAC state unit handling according to the ISA-88 standard.

Note

All blocks in the LUC were created in accordance with the Programming Style Guide.

<https://support.industry.siemens.com/cs/ww/en/view/81318674>

1.2 General

All blocks in the *Library of Unit Controls* LUC library can be freely used in conjunction with SIMATIC S7-1200 and SIMATIC S7-1500 controllers. The blocks are stored as a type in the library. This allows you to benefit from the following advantages:

- Central update function of library items
- Versioning of library elements

Note

Information on general library handling can be found in the library handling guide
<https://support.industry.siemens.com/cs/ww/en/view/109747503>

and in the programming guide for S7-1200/1500 in the “Libraries” chapter
<https://support.industry.siemens.com/cs/ww/en/view/81318674>

More information on libraries in the TIA Portal:

1. Libraries topic page
 - <https://support.industry.siemens.com/cs/ww/en/view/109738702>
2. How do you open libraries in STEP 7 (TIA Portal)?
 - <https://support.industry.siemens.com/cs/ww/en/view/37364723>
3. Automate in less than 10 minutes TIA Portal: Time Savers - Global Libraries
 - <https://support.industry.siemens.com/cs/ww/en/view/78529894>
4. Which elements from STEP 7 (TIA Portal) can be stored in a library as a type or as a master copy?
 - <https://support.industry.siemens.com/cs/ww/en/view/109476862>
5. How can you automatically open a global library when starting TIA Portal V13 and higher and use it as a company library, for example?
 - <https://support.industry.siemens.com/cs/ww/en/view/100451450>
6. Library with PLC data types (LPD) for STEP 7 (TIA Portal) and SIMATIC S7-1200 / S7-1500
 - <https://support.industry.siemens.com/cs/ww/en/view/109482396>

1.3 Complementary Libraries

The *Library of Unit Control* (LUC) extends the functionality of the OMAC LPML library. Therefore, to employ LUC, integration of the OMAC LPML library into the project is required.

LPML OMAC PackML library

The OMAC PackML library (LPML) provides a user-friendly basis for the configuration and use of an OMAC-compliant mode and state manager for SIMATIC and SIMOTION.

<https://support.industry.siemens.com/cs/ww/en/view/109821198>

1.4 Hardware and software requirements

Requirements for this library

To be able to use the functionality of the library described here, the following hardware and software requirements must be met.

Hardware

Modules can be used with

- S7-1200
- S7-1500(TF) / Software Controller
- Simulation with S7-PLCSIM and PLCSIM Advanced
- WinCC Unified Panel

Software

- From TIA Portal V19
- STEP 7 Professional
- WinCC Unified

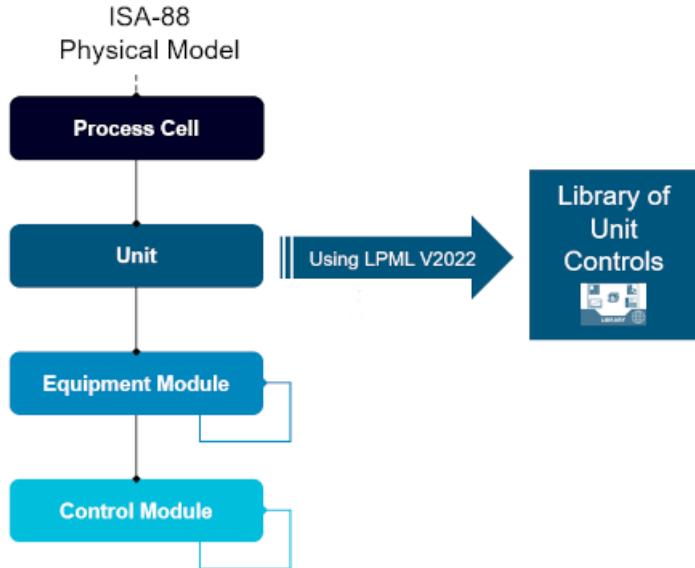
Note

In general, it is possible to open a library with STEP 7 Basic, although STEP 7 Professional elements (e.g. SIMATIC S7-1500 controller) are included. In this case you will be informed with a message when opening the library.

All elements (types and copy templates) can be used if they are supported by the hardware installed in the TIA Portal. If you try to copy elements with STEP 7 Basic from the library that are not supported (e.g. SIMATIC S7-1500 controller), an error message is displayed.

2 Library of Unit Control (LUC) Concept

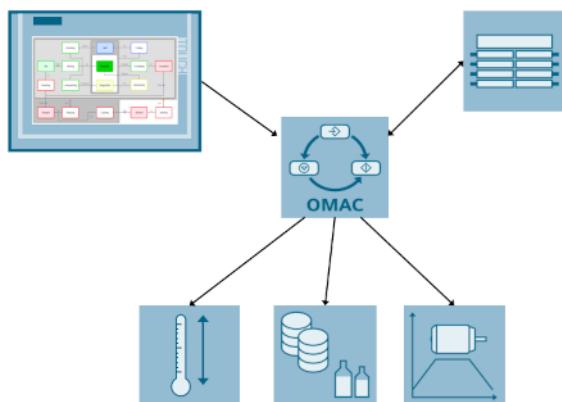
The Library of Unit Control (LUC) provides helpful function blocks on unit level, simplifying the control and monitoring of LPML OMAC state model.



The library of Unit Control handles the OMAC state model of a unit according to the ISA 88 structure.

2.1 OMAC State Model and LPMLV2022

A production machine consists of different unit/machine modes (e.g. manual or production mode) and states (e.g. stopped or aborting). The unit state can be controlled with commands (e.g. start or stop). The library LPMLV2022 provides standardized blocks to control the modes and states of a unit according to the OMAC state model.

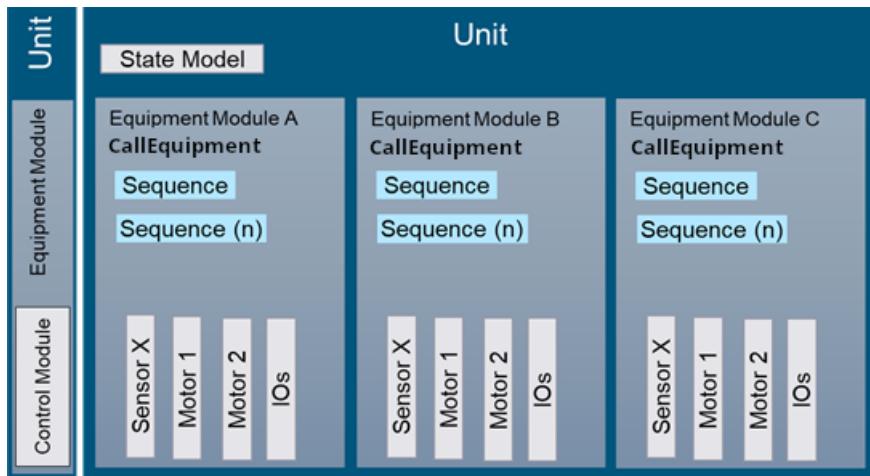


The Library of Unit Control is based on the Library PackML V2022 (LPM LV2022). The Library of Unit Control provides standardized blocks to sync the OMAC State Model standard with the control and monitoring of a unit according to ISA-88 (See next chapter).

2.2 ISA-88 Standard

The ISA (International Society of Automation) decided to provide a standard which describes a method to structure batch processes. It defines terminology and models that makes design and operation of batch plants easier and uniform. This standard is named ISA-88 and is an important foundation for the modular programming structure of production machines.

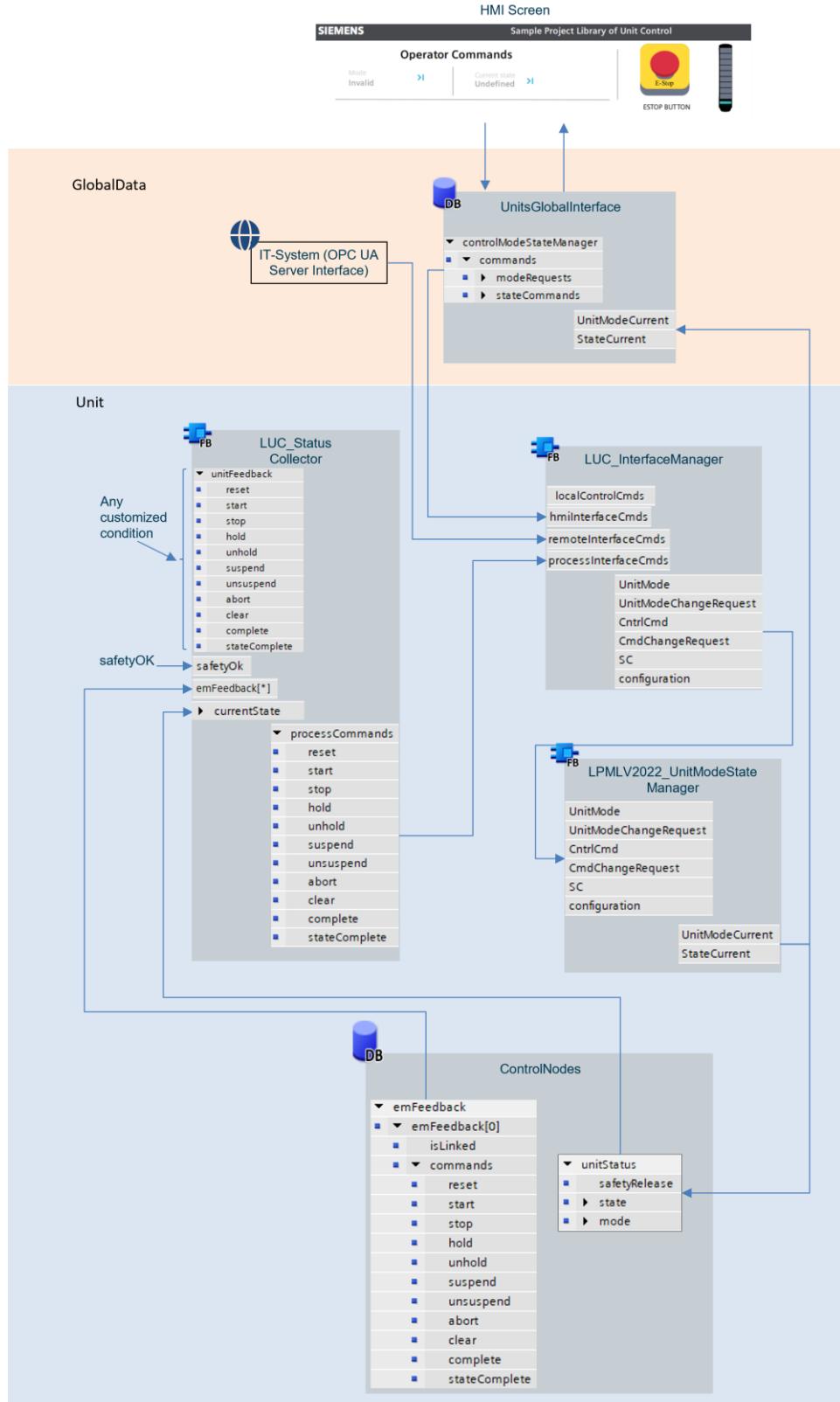
The Library of Unit Control manages the interactions between the ISA-88 layers, from the unit down to the equipment modules.



In the following chapters the implementation of the ISA-88 standard with the help of the Library of Unit Control is explained.

2.2.1 Unit Level

The following figure illustrates the structure of how the LPML OMAC state unit is handled and the preprocessing of the operator or remote commands (e.g. via OPC UA).



The unit OMAC state model is configured in PLC StartUp Organization Block (OB 100) by the Configuration DB. The State model is controlled and monitored via the *UnitsGlobalInterface* DB from HMI or remote (OPC UA).

The processing between the commands and the LPML_UnitModeStateManager is carried out through the following blocks:

- **FB LUC_StatusCollector:** The block receives commands from the equipment modules or any customized condition and evaluates them considering the state of the unit and the Safety. Finally, the *processCommands* are obtained, which are the commands that then would be involved in the *LUC_InterfaceManager* block.
- **FB LUC_InterfaceManager:** The operator can change the unit mode and state through different interfaces (e.g. via HMI, OPC UA or any local customized condition). In this block the collected commands are evaluated considering the *processCommands* obtained from the unit and equipment module status. After evaluating the commands received from the different interfaces, retrieves the corresponding commands to be assigned to the *LPMLV2022_UnitModeStateManager* block to perform the mode or state change of the Unit Model State.

There are more helpful blocks related to manage the unit level:

- **FC LUC_StateMachineConfig:** This block transfers the user settings of the enabled modes, states, transitions to other modes, hold and complete commands of the OMAC model state for the configured unit. This block should be called in the StartUp Organization Block (OB100).
- **FB LUC_RemoteCtrl_OpcUa:** This Block represents the call environment for the OPC UA Methods belonging to the *LUC_InterfaceManager*. It calls all the available methods to control the *LUC_InterfaceManager*.
- **FB LUC_SignalStack:** This Block represents a Signal Stack object, to set the signal stack status according to the current global unit-state from the *LUC_InterfaceManager*.

These blocks require user settings, so it is recommended to create a *ConfigurationDB* with the following structure. For more information, see each block description.

Configuration	Name	Data type	Start value	Ret...	Acces...	Writ...	Visibil...	Set...	Supervision	Comment
Static	stateConfiguration	"LUC_typeStateMachineConfiguration"		<input checked="" type="checkbox"/>		Configuration of states disabled for each mode				
	enableModes	"LUC_typeModeConfiguration"		<input checked="" type="checkbox"/>		Configure which modes are enabled				
	enableStates	Array[0.."LPMVL2022_MAX_MODES_U..."		<input checked="" type="checkbox"/>		Configure which states are enabled for each mode				
	enableModeTransi...	Array[0.."LPMVL2022_MAX_MODES_U..."		<input checked="" type="checkbox"/>		Configure in which states a mode transition is allowed				
	enableHoldComma...	"LUC_typeHoldCommandConfiguration"		<input checked="" type="checkbox"/>		Configure in which states a hold command is possible				
	enableCompleteC...	"LUC_typeCompleteCommandConfigu..."		<input checked="" type="checkbox"/>		Configure in which states a complete command is possible				
	stateModeManager	"LPMVL2022_typeConfiguration"		<input checked="" type="checkbox"/>		Support: tech.team.motioncontrol@siemens.com				
	EnabledModesCfg	DWord	16#0000...0000	<input checked="" type="checkbox"/>		Bit locations within the DWORD represent Mode numbers. Avel...				
	DisabledStatesCg	Array[0.."LPMVL2022_MAX_MODES_U..."		<input checked="" type="checkbox"/>		The array index represents the Mode number. Bit locations wit...				
	ModeTransitionCg	Array[0.."LPMVL2022_MAX_MODES_U..."		<input checked="" type="checkbox"/>		The array index represents the Mode number. Bit locations wit...				
	holdCmdCfg	DWord	16#0000...0000	<input checked="" type="checkbox"/>		Bit locations within the DWORD value represent State numbers...				
	completeCmdCg	DWord	16#0000...0000	<input checked="" type="checkbox"/>		Bit locations within the DWORD value represent State numbers...				
	remoteCtrlOpcUa	"LUC_typeRemoteCtrlConfiguration"		<input checked="" type="checkbox"/>		OPC UA Method calls parameters, to parametrize the functiona...				
	enableMethods	Struct		<input checked="" type="checkbox"/>		Enabling of methods				
	signalStack	"LUC_typeSignalStackConfiguration"		<input checked="" type="checkbox"/>		Behaviour settings for the signal stack - machine modes				
	undefined	"LUC_typeSignalStackStateBehaviour"		<input checked="" type="checkbox"/>		Stack light status - machine state "undefined"				
	clearing	"LUC_typeSignalStackStateBehaviour"		<input checked="" type="checkbox"/>		Stack light status - machine state "clearing"				
	stopped	"LUC_typeSignalStackStateBehaviour"		<input checked="" type="checkbox"/>		Stack light status - machine state "stopped"				
	starting	"LUC_typeSignalStackStateBehaviour"		<input checked="" type="checkbox"/>		Stack light status - machine state "starting"				
	idle	"LUC_typeSignalStackStateBehaviour"		<input checked="" type="checkbox"/>		Stack light status - machine state "idle"				
	suspended	"LUC_typeSignalStackStateBehaviour"		<input checked="" type="checkbox"/>		Stack light status - machine state "suspended"				
	execute	"LUC_typeSignalStackStateBehaviour"		<input checked="" type="checkbox"/>		Stack light status - machine state "execute"				
	stopping	"LUC_typeSignalStackStateBehaviour"		<input checked="" type="checkbox"/>		Stack light status - machine state "stopping"				
	aborting	"LUC_typeSignalStackStateBehaviour"		<input checked="" type="checkbox"/>		Stack light status - machine state "aborting"				
	aborted	"LUC_typeSignalStackStateBehaviour"		<input checked="" type="checkbox"/>		Stack light status - machine state "aborted"				
	holding	"LUC_typeSignalStackStateBehaviour"		<input checked="" type="checkbox"/>		Stack light status - machine state "holding"				
	held	"LUC_typeSignalStackStateBehaviour"		<input checked="" type="checkbox"/>		Stack light status - machine state "held"				
	unholding	"LUC_typeSignalStackStateBehaviour"		<input checked="" type="checkbox"/>		Stack light status - machine state "unholding"				
	suspending	"LUC_typeSignalStackStateBehaviour"		<input checked="" type="checkbox"/>		Stack light status - machine state "suspending"				
	unsuspending	"LUC_typeSignalStackStateBehaviour"		<input checked="" type="checkbox"/>		Stack light status - machine state "unsuspending"				
	resetting	"LUC_typeSignalStackStateBehaviour"		<input checked="" type="checkbox"/>		Stack light status - machine state "resetting"				
	completing	"LUC_typeSignalStackStateBehaviour"		<input checked="" type="checkbox"/>		Stack light status - machine state "completing"				
	completed	"LUC_typeSignalStackStateBehaviour"		<input checked="" type="checkbox"/>		Stack light status - machine state "completed"				

2.2.2 Equipment Module Level

The library of Unit Control also manages the connection or link between the unit and the different equipment modules. For that the Library of Unit Control provides the following block:

- **FC LUC_InterlinkToParent:** This block is used to link or unlink an equipment module from a unit. In case the equipment module is linked, this information is sent to *HmiInterface DB*. From the unit point of view, if an equipment module is linked, the unit takes the feedback from the equipment module into account. In case it is unlinked, the unit would not take the feedback of the equipment module into account.

Use Cases for *FC LUC_InterlinkToParent*:

- If one of the equipment modules of a unit is sending a *StopCommand*, the unit will change into *Stopping* state. If the designed machine process supports this, the user can unlink the Equipment and continue the production during the unlinked module is sending a *cmdStop*.
- In case the operator wants to do maintenance in the unit for each equipment module separately, the user can unlink the equipment module on which maintenance will be done.
- For specific case when an equipment module is not available physically, this one can be unliked from the unit to proceed with the work without any inconvenience.

2.3 Structure and design of the library

The following chapters describe all blocks of the library “Library of Unit Control”. The chapters are structured in the same way as the library itself.

2.3.1 Functions and Function Blocks FCs/FBs

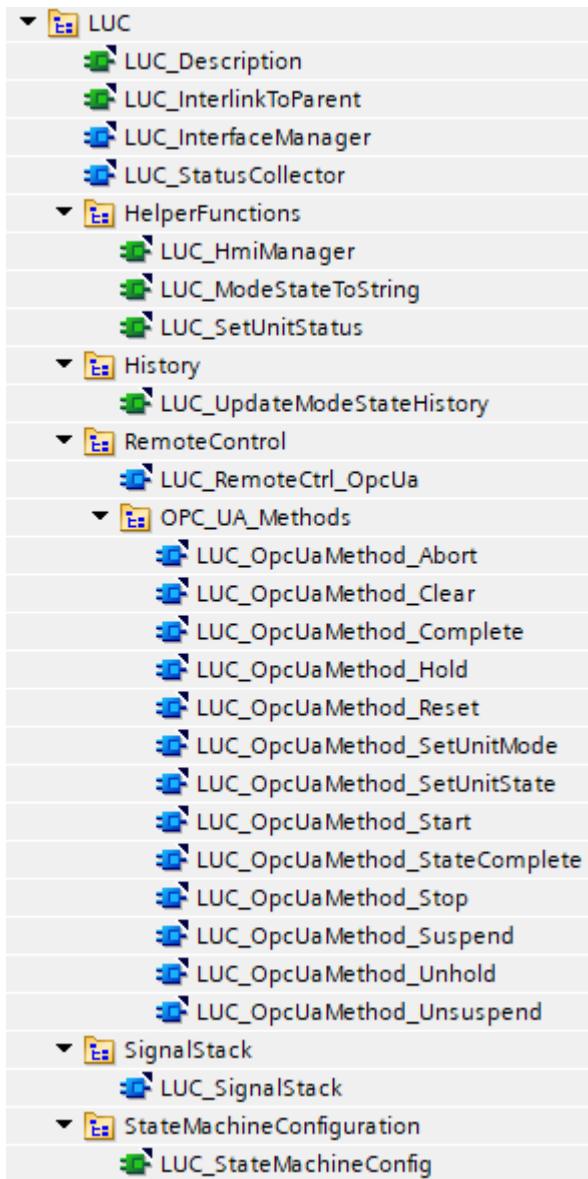


Table: Main Functions of Library of Unit Control

Name	Version
LUC_InterfaceManager	V1.2
LUC_StatusCollector	V1.2
LUC_InterlinkToParent	V1.2

2 Library of Unit Control (LUC) Concept

Table: Helper Functions of Library of Unit Control

Name	Version
LUC_HmiManager	V1.2
LUC_ModeStateToString	V1.2
LUC_SetUnitStatus	V1.2

Table: History of Library of Unit Control

Name	Version
LUC_UpdateModeStateHistory	V1.2

Table: Remote Control and OPC UA Methods of Library of Unit Control

Name	Version
LUC_RemoteCtrl_OpcUa	V1.2
LUC_OpcUaMethod_Abort	V1.2
LUC_OpcUaMethod_Clear	V1.2
LUC_OpcUaMethod_Complete	V1.2
LUC_OpcUaMethod_Hold	V1.2
LUC_OpcUaMethod_Reset	V1.2
LUC_OpcUaMethod_SetUnitMode	V1.2
LUC_OpcUaMethod_SetUnitState	V1.2
LUC_OpcUaMethod_Start	V1.2
LUC_OpcUaMethod_StateComplete	V1.2
LUC_OpcUaMethod_Stop	V1.2
LUC_OpcUaMethod_Suspend	V1.2
LUC_OpcUaMethod_Unhold	V1.2
LUC_OpcUaMethod_Unsuspend	V1.2

Table: Signal Stack of Library of Unit Control

Name	Version
LUC_SignalStack	V1.2

Table: State Machine Configuration of Library of Unit Control

Name	Version
LUC_StateMachineConfig	V1.2

2.3.2 User Defined Types UDTs

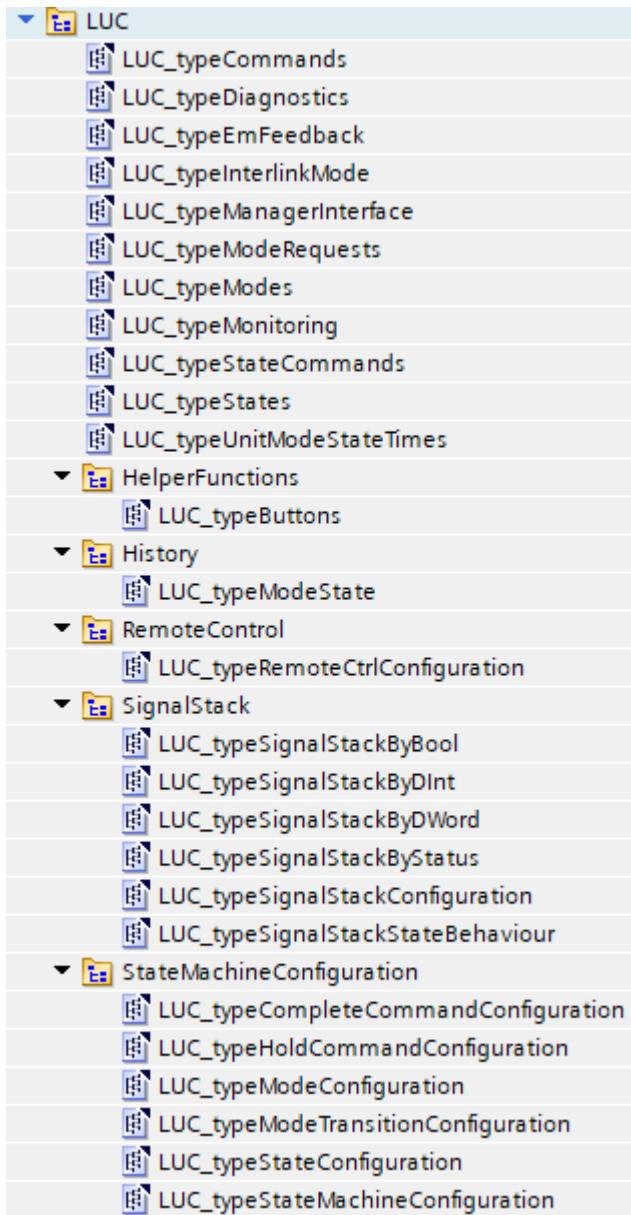


Table: Main User Defined Types (UDTs) of Library of Unit Control

Name	Version
LUC_typeCommands	V1.1
LUC_typeDiagnostics	V1.1
LUC_typeEmFeedback	V1.2
LUC_typeInterlinkMode	V1.2
LUC_typeManagerInterface	V1.2
LUC_typeModeRequests	V1.1
LUC_typeModes	V1.2
LUC_typeMonitoring	V1.2
LUC_typeStateCommands	V1.1
LUC_typeStates	V1.2
LUC_typeUnitModeStateTimes	V1.2

Table: Helper Functions User Defined Types (UDTs) of Library of Unit Control

Name	Version
LUC_typeButtons	V1.2

Table: History User Defined Types (UDTs) of Library of Unit Control

Name	Version
LUC_typeModeState	V1.2

Table: Remote Control User Defined Types (UDTs) of Library of Unit Control

Name	Version
LUC_typRemoteCtrlConfiguration	V1.1

Table: Signal Stack User Defined Types (UDTs) of Library of Unit Control

Name	Version
LUC_typeSignalStackByBool	V1.1
LUC_typeSignalStackByDInt	V1.2
LUC_typeSignalStackByDWord	V1.2
LUC_typeSignalStackByStatus	V1.2
LUC_typeSignalStackConfiguration	V1.1
LUC_typeSignalStackStateBehaviour	V1.1

Table: Mode State Manager User Defined Types (UDTs) of Library of Unit Control

Name	Version
LUC_typeCompleteCommandConfiguration	V1.2
LUC_typeHoldCommandConfiguration	V1.2
LUC_typeModeConfiguration	V1.2
LUC_typeModeTransitionConfiguration	V1.2
LUC_typeStateConfiguration	V1.2
LUC_typeStateMachineConfiguration	V1.2

3 Program blocks

3.1 LUC_InterlinkToParent (FC / 1.2.0)

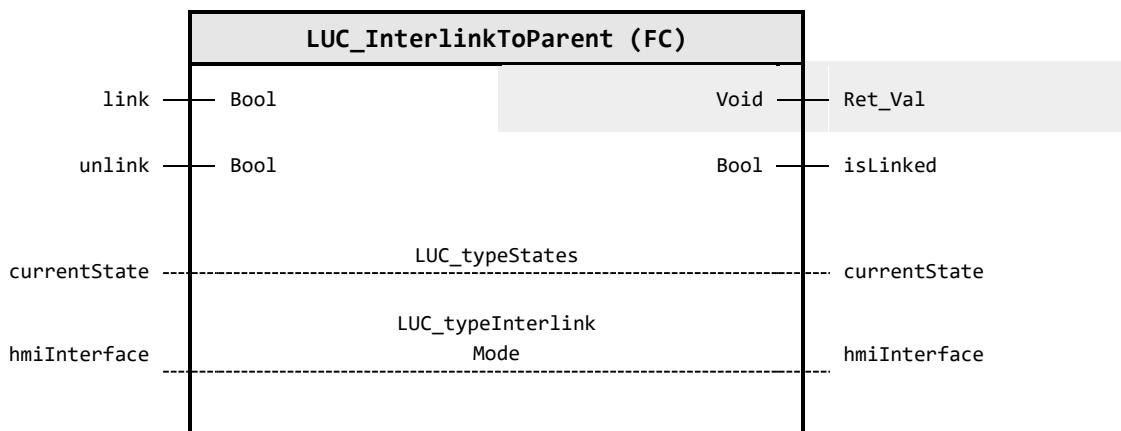
Author: Siemens Digital Industries

Short description

This function links or unlinks an EM from the Unit

Interface description

Block interface



Input parameter

Identifier	Data type	Description
link	Bool	Command to link module to parent
unlink	Bool	Command to unlink module from parent

Output parameter

Identifier	Data type	Description
Ret_Val	Void	Void - Function has no return value
isLinked	Bool	Feedback from the EM to the Unit

In/Out parameter

Identifier	Data type	Description
currentState	LUC_typeStates	Current state of the unit
hmiInterface	LUC_typeInterlinkMode	Connection to DB HmiInterface of the EM

User defined datatype(s)**LUC_typeStates (UDT / V1.1.0)**

Identifier	Data type	Default value	Description
current	DInt	0	Current unit state
clearing	Bool	FALSE	The unit state "Clearing" is active
stopped	Bool	FALSE	The unit state "Stopped" is active
starting	Bool	FALSE	The unit state "Starting" is active
idle	Bool	FALSE	The unit state "Idle" is active
suspended	Bool	FALSE	The unit state "Suspended" is active
execute	Bool	FALSE	The unit state "Execute" is active
stopping	Bool	FALSE	The unit state "Stopping" is active
aborting	Bool	FALSE	The unit state "Aborting" is active
aborted	Bool	FALSE	The unit state "Aborted" is active
holding	Bool	FALSE	The unit state "Holding" is active
held	Bool	FALSE	The unit state "Held" is active
unholding	Bool	FALSE	The unit state "Unholding" is active
suspending	Bool	FALSE	The unit state "Suspending" is active
unsuspending	Bool	FALSE	The unit state "Unsuspending" is active
resetting	Bool	FALSE	The unit state "Resetting" is active
completing	Bool	FALSE	The unit state "Completing" is active
complete	Bool	FALSE	The unit state "Complete" is active

LUC_typeInterlinkMode (UDT / V1.2.0)

Link and unlink from parent module

Identifier	Data type	Default value	Description
link	Bool	FALSE	Command to link module to parent
unlink	Bool	FALSE	Command to unlink module from parent
isLinked	Bool	true	Equipment module is linked to parent

Functional description

This block enables an Equipment Module to be linked or unlinked from a Unit. The process of linking or unlinking it is only possible when the parent (the Unit that contains the respective Equipment Module) is in the active state of *Aborted*, *Clearing* or *Stopped*.

- Link an Equipment Module to a Unit: The Equipment Module must be not linked to the Unit and the *link* command must be enabled from the HMI Interface or from the input of the block.
- Unlink an Equipment Module from a Unit: The Equipment Module must be linked to the Unit and the *unlink* command must be enabled from the HMI Interface or from the input of the block.

Change log

Version & Date	Change description
1.0.0 15.12.2023	SIMATIC Systems Support First released version
1.1.0 21.06.2024	SIMATIC Systems Support Implemented the equipment module sharing of the linked status with the ControlNodes DB and HmiInterface DB of the Equipment Module.
1.2.0 05.09.2024	SIMATIC Systems Support Modified the block interface, removed ControlNodes interface. The equipment module can be linked or unlinked to/from the unit through the block interface inputs.

3.2 LUC_InterfaceManager (FB / 1.2.0)

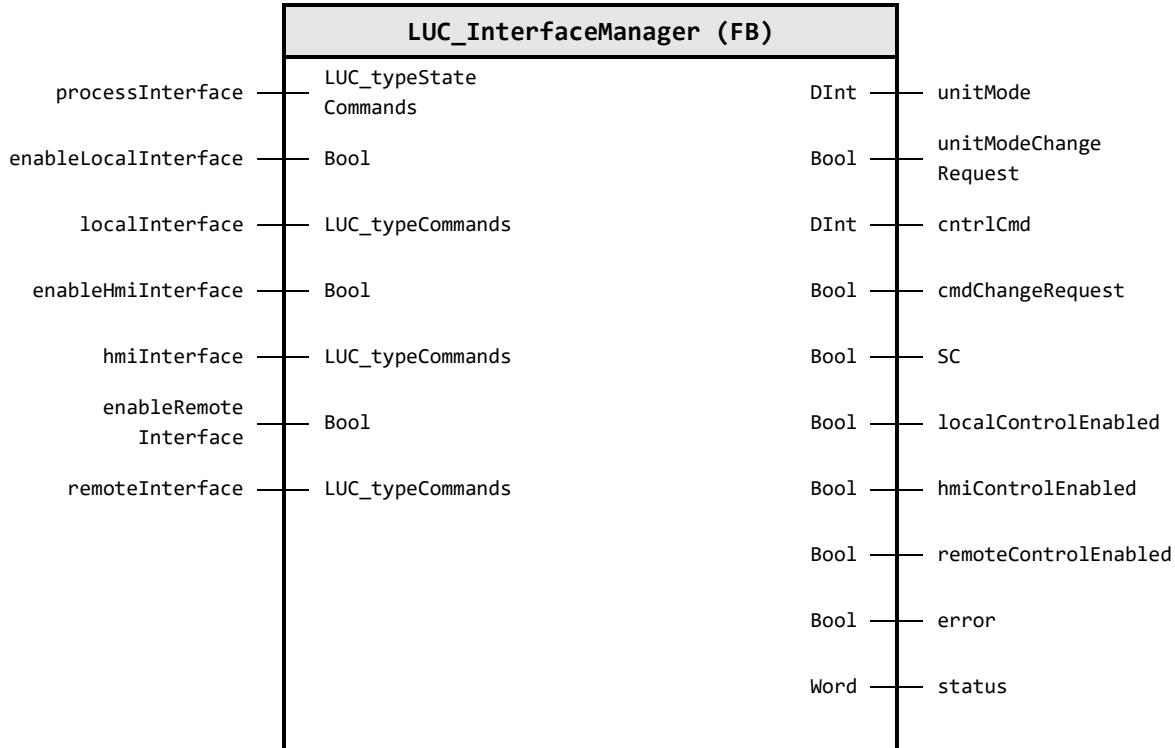
Author: Siemens Digital Industries

Short description

This function block collects commands from different interfaces and summarizes them in the outputs. The main use of this block is to manage the LPMLV2022_UnitModeStateManager when the commands come from different interfaces.

Interface description

Block interface



Input parameter

Identifier	Data type	Default value	Description
processInterface	LUC_typeStateCommands	---	Commands from the own unit and lower level modules
enableLocalInterface	Bool	FALSE	Enable control via the local interface
localInterface	LUC_typeCommands	---	Operator commands from a local systems (i.e. pushbuttons)
enableHmiInterface	Bool	FALSE	Enable control via the hmi interface
hmiInterface	LUC_typeCommands	---	Operator commands from HMI
enableRemoteInterface	Bool	FALSE	Enable control via the remote interface
remoteInterface	LUC_typeCommands	---	Operator commands from remote (i.e. OPC UA)

Output parameter

Identifier	Data type	Description
unitMode	DIInt	Requested unit mode
unitModeChangeRequest	Bool	Request unit mode
cntrlCmd	DIInt	Request control command
cmdChangeRequest	Bool	Enable change into requested state
SC	Bool	State change from FALSE to TRUE (rising edge) triggers state complete signal
localControlEnabled	Bool	Control via local interface enabled
hmiControlEnabled	Bool	Control via hmi interface enabled
remoteControlEnabled	Bool	Control via remote interface enabled
error	Bool	Error occurred
status	Word	Status information

Status & Error codes

Code / Value	Identifier / Description
16#0000	STATUS_NO_ERROR_OCCURRED No error occurred during execution
16#8006	ERR_MISSING_LOCAL_CTRL_ENABLE Missing release signal for local control "enableLocalControl"
16#8007	ERR_MISSING_HMI_CTRL_ENABLE Missing release signal for local control "enableHmiControl"
16#8008	ERR_MISSING_REMOTE_CTRL_ENABLE Missing release signal for remote control "enableRemoteControl"

User defined datatype(s)

LUC_typeStateCommands (UDT / V1.1.0)

Unit State Commands

Identifier	Data type	Default value	Description
reset	Bool	FALSE	OMAC PackMLV2022 control command Reset
start	Bool	FALSE	OMAC PackMLV2022 control command Start
stop	Bool	FALSE	OMAC PackMLV2022 control command Stop
hold	Bool	FALSE	OMAC PackMLV2022 control command Hold
unhold	Bool	FALSE	OMAC PackMLV2022 control command Unhold
suspend	Bool	FALSE	OMAC PackMLV2022 control command Suspend
unsuspend	Bool	FALSE	OMAC PackMLV2022 control command Unsuspend
abort	Bool	FALSE	OMAC PackMLV2022 control command Abort
clear	Bool	FALSE	OMAC PackMLV2022 control command Clear
complete	Bool	FALSE	OMAC PackMLV2022 control command Complete
stateComplete	Bool	FALSE	OMAC PackMLV2022 control command State Complete

LUC_typeCommands (UDT / V1.1.0)

Module related commands from external systems

Identifier	Data type	Default value	Description
modeRequests	LUC_typeModeRequests	---	Commands to request a mode change
stateCommands	LUC_typeStateCommands	---	Commands to request a state change

Functional description

Description of input interfaces

For the purposes of changing the mode and/or the state of the unit, the LUC_InterfaceManager can receive commands through the process, local, HMI or remote interfaces. In case there are more than one interface of control enabled and there are state commands and mode requested at the same time, the prioritization is the following:

- State Request Commands. The priorities in the state requested commands are:
 - a. *Abort* has the priority.
 - b. *Stop* has the second priority.
 - c. In case of no abort or stop command, priority is sorted as following: *Reset, Start, Hold, Unhold, Suspend, Unsuspend, Clear, Complete, State_Complete*.
- Mode Request. The priority in the modes requested is dependent on the mode requested. The hierarchy is the following:
 - a. Production.
 - b. Maintenance.
 - c. Manual.
 - d. User_Mode_01.
 - e. User_Mode_02.
 - f. User_Mode_03.
 - g. User_Mode_04.
 - h. User_Mode_05.

Process Commands

This variable reads the state commands that come from the block *LUC_StatusCollector* (where the Equipment Modules state commands are collected and evaluated).

Local interface

This is used in the scenario in which mode/state change commands come from other PLC blocks within the unit automation solution, the LUC_InterfaceManager has a local interface (*localInterfaceCmds*) To allow the control through local interface *enableLocalControl* must be enabled.

HMI interface

This is used in case of controlling mode/state through an HMI, the LUC_InterfaceManager has an HMI interface (*hmiInterfaceCmds*) To allow the control through local interface *enableHmiControl* must be enabled.

Remote interface

For the purposes of changing the production mode and/or unit state of the machine remotely, the LUC_InterfaceManager has a remote interface (*remoteInterfaceCmds*). This interface is then used by the LUC_RemoteCtrl_OpcUa block for the purpose of receiving mode/state commands. For the OPC UA commands to have any effect, remote control must be enabled.

Monitoring the LUC_InterfaceManager

The current status of the block is known through the output interface :

Output interface

The output interface is typically used for interfacing the block with other PLC blocks and it provides feedback about:

- Requested mode/stateCmd
 - Mode/stateCmd boolean change request
 - StateComplete
 - Local Control Enabled
 - HMI Control Enabled
 - Remote Control Enabled
 - Status word
 - Diagnostics structure that contains a buffer with precise information about mode/state change with timestamp.

Change log

Version & Date	Change description
1.0.0	SIMATIC Systems Support
15.12.2023	First released version
1.1.0	SIMATIC Systems Support
21.06.2024	Implemented 3 different interfaces of control (local, hmi, remote)and diagnosis status through just a word value "status".
1.2.0	SIMATIC Systems Support
05.09.2024	Optimized code.

3.3 LUC_StatusCollector (FB / 1.2.0)

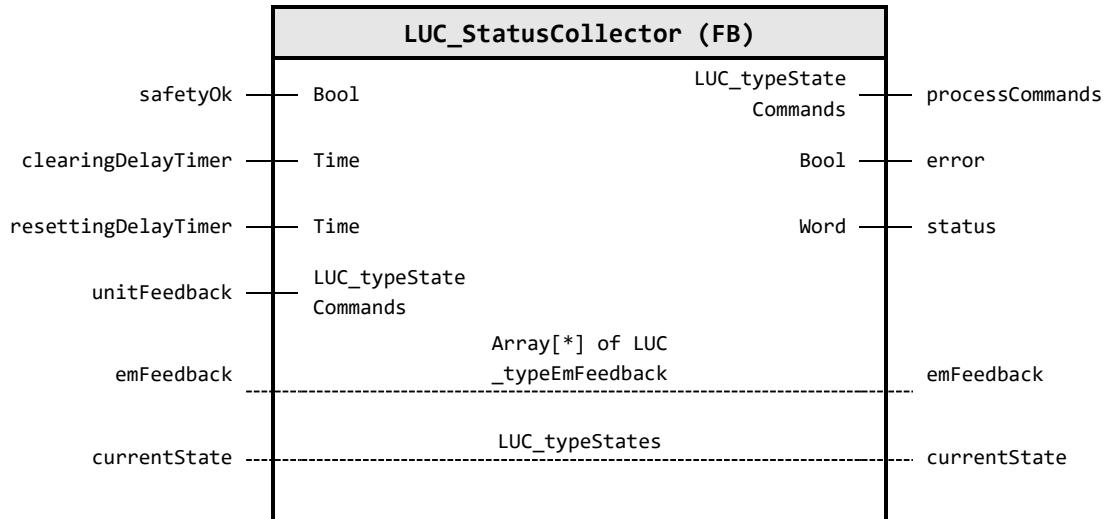
Author: Siemens Digital Industries

Short description

Reads the commands of multiple equipment modules and summarizes to unit status.

Interface description

Block interface



Input parameter

Identifier	Data type	Default value	Description
safetyOk	Bool	FALSE	Safety release coming from the safety program
clearingDelayTimer	Time	T#2000ms	During that time, command abort is suppressed in state clearing
resettingDelayTimer	Time	T#5000ms	During that time, command stop is suppressed in state clearing
unitFeedback	LUC_typeStateCommands	---	Unit State Commands

Output parameter

Identifier	Data type	Description
processCommands	LUC_typeStateCommands	Struct for monitoring the commands to request a state change
error	Bool	Error occurred
status	Word	Status information

In/Out parameter

Identifier	Data type	Description
emFeedback	Array[*] of LUC_typeEmFeedback	Connection to DB ControlNodes of the unit
currentState	LUC_typeStates	Current state of the unit

User defined datatype(s)**LUC_typeStateCommands (UDT / V1.1.0)**

Unit State Commands

Identifier	Data type	Default value	Description
reset	Bool	FALSE	OMAC PackMLV2022 control command Reset
start	Bool	FALSE	OMAC PackMLV2022 control command Start
stop	Bool	FALSE	OMAC PackMLV2022 control command Stop
hold	Bool	FALSE	OMAC PackMLV2022 control command Hold
unhold	Bool	FALSE	OMAC PackMLV2022 control command Unhold
suspend	Bool	FALSE	OMAC PackMLV2022 control command Suspend
unsuspend	Bool	FALSE	OMAC PackMLV2022 control command Unsuspend
abort	Bool	FALSE	OMAC PackMLV2022 control command Abort
clear	Bool	FALSE	OMAC PackMLV2022 control command Clear
complete	Bool	FALSE	OMAC PackMLV2022 control command Complete
stateComplete	Bool	FALSE	OMAC PackMLV2022 control command State Complete

LUC_typeEmFeedback (UDT / V1.1.0)

Identifier	Data type	Default value	Description
isLinked	Bool	true	EM is linked to the parent (usually the unit)
commands	LUC_typeStateCommands	---	Unit State Commands

LUC_typeStates (UDT / V1.1.0)

Identifier	Data type	Default value	Description
current	DIInt	0	Current unit state
clearing	Bool	FALSE	The unit state "Clearing" is active
stopped	Bool	FALSE	The unit state "Stopped" is active
starting	Bool	FALSE	The unit state "Starting" is active
idle	Bool	FALSE	The unit state "Idle" is active
suspended	Bool	FALSE	The unit state "Suspended" is active
execute	Bool	FALSE	The unit state "Execute" is active
stopping	Bool	FALSE	The unit state "Stopping" is active
aborting	Bool	FALSE	The unit state "Aborting" is active
aborted	Bool	FALSE	The unit state "Aborted" is active
holding	Bool	FALSE	The unit state "Holding" is active
held	Bool	FALSE	The unit state "Held" is active
unholding	Bool	FALSE	The unit state "Unholding" is active
suspending	Bool	FALSE	The unit state "Suspending" is active
unsuspending	Bool	FALSE	The unit state "Unsuspending" is active
resetting	Bool	FALSE	The unit state "Resetting" is active
completing	Bool	FALSE	The unit state "Completing" is active
complete	Bool	FALSE	The unit state "Complete" is active

Functional description

The block reads the *emFeedback* of all equipment modules that are linked to the unit and the *unitStatus* of and defines the *processCommands* for the main mode state manager. The commands *reset*, *start*, *stop*, *hold*, *unhold*, *suspend*, *unsuspend*, *abort*, *clear_* and *complete* can also be set from outside to define the *processCommands* independently from the *emStatus* commands.

The *processCommands* are enabled depending on the following:

reset:

The *processCommands.reset* will be set if:

- There is a *cmdReset* feedback from any Equipment Module.
- The input command *reset* is set.

start:

The *processCommands.start* will be set if:

- There is a *cmdStart* feedback from any Equipment Module.
- The input command *start* is set.

stop:

The *processCommands.stop* will be set if:

- Any Equipment Module reports a *cmdStop*.
- The input command *stop* is set.

The *processCommands.stop* will be reset if the Unit is in *Resetting* state during *resettingDelayTime*, to give the Unit time to reset faults.

hold:

The *processCommands.hold* will be set if:

- Any Equipment Module reports a *cmdHold*.
- The input command *hold* is set.

unhold:

The *processCommands.unhold* will be set if:

- Any Equipment Module reports a *cmdUnhold*.
- The input command *unhold* is set.

suspend:

The *processCommands.suspend* will be set if:

- Any Equipment Module reports a *cmdSuspend*.
- The input command *suspend* is set.

unsuspend:

The *processCommands.unsuspend* will be set if:

- Any Equipment Module reports a *cmdUnsuspend*.
- The input command *unsuspend* is set.

abort:

The `processCommands.abort` will be set if:

- Any Equipment Module reports a `cmdAbort`
- The input command `abort` is set.
- `SafetyOK` is not enabled.

The `processCommands.abort` will be reset if the Unit is in *Clearing* state during `clearingDelayTime`, to give the Unit time to clear faults.

clear:

The `processCommands.clear` will be set if:

- Any Equipment Module reports a `cmdClear`.
- The input command `clear` is set.

complete:

The `processCommands.complete` will be set if:

- Any Equipment Module reports a `cmdComplete`.
- The input command `complete` is set.

stateComplete:

The `processCommands.stateComplete` will be set only when all Equipment Module reports a `stateComplete` and the `safetyOk` bit is enabled.

Change log

Version & Date	Change description
1.0.0 12.12.2023	SIMATIC Systems Support First released version
1.1.0 21.06.2024	SIMATIC Systems Support Implemented new input commands (that can set the <code>processCommands</code> are extended with the implementation of new commands. In addition, the Unit Status is taken into account to set the <code>processCommands</code> .
1.2.0 05.09.2024	SIMATIC Systems Support Optimized code.

3.4 HelperFunctions

3.4.1 LUC_ModeStateToString (FC / 1.2.0)

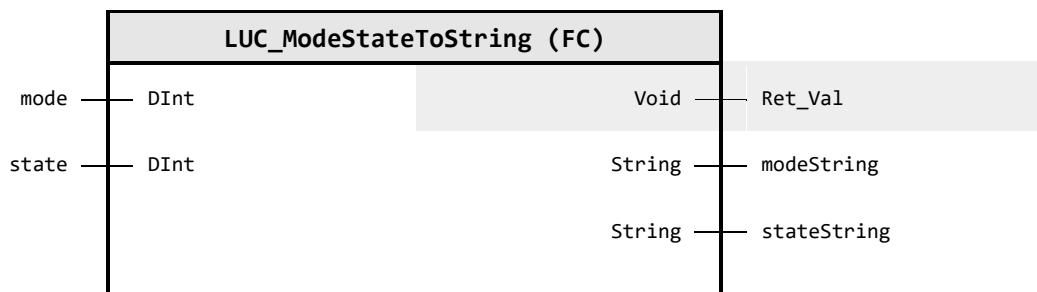
Author: Siemens Digital Industries

Short description

Converts the DIInt Value of the *mode* and *state* into a string

Interface description

Block interface



Input parameter

Identifier	Data type	Description
mode	DIInt	Current mode as DINT
state	DIInt	Current state as DINT

Output parameter

Identifier	Data type	Description
Ret_Val	Void	Void - Function has no return value
modeString	String	Current mode as string
stateString	String	Current state as string

Functional description

Relation between DIint *mode* and *LUC_ModeToString* string values

MODE (DIint)	LUC_ModeToString (String)
0	Invalid
1	Production
2	Maintenance
3	Manual
4	UserMode01
5	UserMode02
6	UserMode03
7	UserMode04
8	UserMode05
9	UserMode06
10	UserMode07
11	UserMode08
12	UserMode09

3 Program blocks

MODE (DInt)	LUC_ModeToString (String)
13	UserMode10
14	UserMode11
15	UserMode12
16	UserMode13

Relation between Dint state and LUC_StateToString String

STATE (DInt)	LUC_StateToString (LUC_typeStates)
0	Undefined
1	Clearing
2	Stopped
3	Starting
4	Idle
5	Suspended
6	Execute
7	Stopping
8	Aborting
9	Aborted
10	Holding
11	Held
12	Unholding
13	Suspending
14	Resetting
15	Clearing
16	Completing
17	Completed

Change log

Version & Date	Change description
1.0.0 26.04.2024	SIMATIC Systems Support First released version
1.1.0 21.06.2024	SIMATIC Systems Support Implemented the mode and state conversion in the same block.
1.2.0 05.09.2024	SIMATIC Systems Support Optimized code.

3.4.2 LUC_SetUnitStatus (FC / 1.2.0)

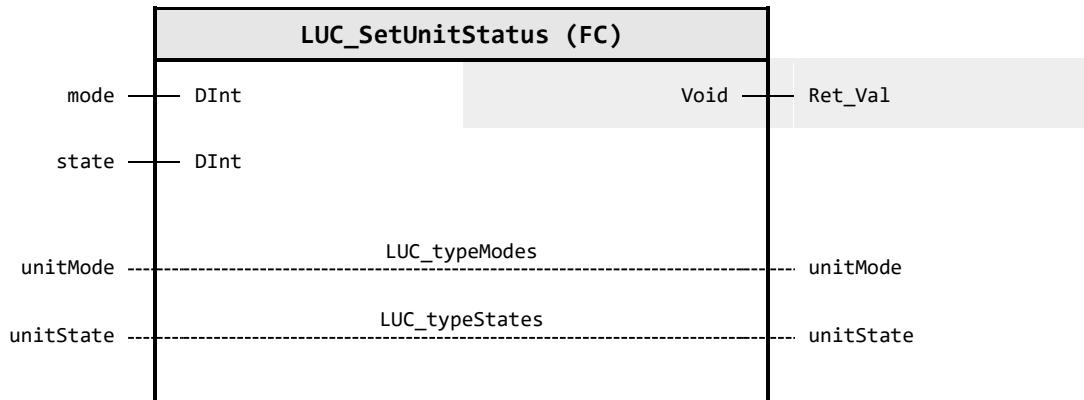
Author: Siemens Digital Industries

Short description

Translate from Dint *state* and *mode* to *LUC_typeStates* and *LUC_typeModes* structures respectively.

Interface description

Block interface



Input parameter

Identifier	Data type	Description
mode	DInt	Active mode of the unit expressed in a Dint data type
state	DInt	Active state of the unit expressed in a Dint data type

Output parameter

Identifier	Data type	Description
Ret_Val	Void	Void - Function has no return value

In/Out parameter

Identifier	Data type	Description
unitMode	LUC_typeModes	Active mode of the unit expressed in a boolean structure
unitState	LUC_typeStates	Active state of the unit expressed in a boolean structure

User defined datatype(s)**LUC_typeModes (UDT / V1.1.0)**

Identifier	Data type	Default value	Description
current	DInt	0	Current operating mode
production	Bool	FALSE	The operating mode "Production" is active
maintenance	Bool	FALSE	The operating mode "Maintenance" is active
manual	Bool	FALSE	The operating mode "Manual" is active
userMode01	Bool	FALSE	The operating mode "UserMode1" is active
userMode02	Bool	FALSE	The operating mode "UserMode2" is active
userMode03	Bool	FALSE	The operating mode "UserMode3" is active
userMode04	Bool	FALSE	The operating mode "UserMode4" is active
userMode05	Bool	FALSE	The operating mode "UserMode5" is active
userMode06	Bool	FALSE	The operating mode "UserMode6" is active
userMode07	Bool	FALSE	The operating mode "UserMode7" is active
userMode08	Bool	FALSE	The operating mode "UserMode8" is active
userMode09	Bool	FALSE	The operating mode "UserMode9" is active
userMode10	Bool	FALSE	The operating mode "UserMode10" is active
userMode11	Bool	FALSE	The operating mode "UserMode11" is active
userMode12	Bool	FALSE	The operating mode "UserMode12" is active
userMode13	Bool	FALSE	The operating mode "UserMode13" is active

LUC_typeStates (UDT / V1.1.0)

Identifier	Data type	Default value	Description
current	DInt	0	Current unit state
clearing	Bool	FALSE	The unit state "Clearing" is active
stopped	Bool	FALSE	The unit state "Stopped" is active
starting	Bool	FALSE	The unit state "Starting" is active
idle	Bool	FALSE	The unit state "Idle" is active
suspended	Bool	FALSE	The unit state "Suspended" is active
execute	Bool	FALSE	The unit state "Execute" is active
stopping	Bool	FALSE	The unit state "Stopping" is active
aborting	Bool	FALSE	The unit state "Aborting" is active
aborted	Bool	FALSE	The unit state "Aborted" is active
holding	Bool	FALSE	The unit state "Holding" is active
held	Bool	FALSE	The unit state "Held" is active
unholding	Bool	FALSE	The unit state "Unholding" is active
suspending	Bool	FALSE	The unit state "Suspending" is active
unsuspending	Bool	FALSE	The unit state "Unsuspending" is active
resetting	Bool	FALSE	The unit state "Resetting" is active
completing	Bool	FALSE	The unit state "Completing" is active
complete	Bool	FALSE	The unit state "Complete" is active

Functional description

Relation between Dint mode and modes structure

MODE (DINT)	MODES (LUC_typeModes)
0	Invalid
1	Production
2	Maintenance
3	Manual
4	UserMode01
5	UserMode02
6	UserMode03
7	UserMode04
8	UserMode05
9	UserMode06
10	UserMode07
11	UserMode08
12	UserMode09
13	UserMode10
14	UserMode11
15	UserMode12
16	UserMode13

Relation between Dint state and states structure

STATE (DInt)	STATE (LUC_typeStates)
0	Undefined
1	Clearing
2	Stopped
3	Starting
4	Idle
5	Suspended
6	Execute
7	Stopping
8	Aborting
9	Aborted
10	Holding
11	Held
12	Unholding
13	Suspending
14	Resetting
15	Clearing
16	Completing
17	Completed

Change log

Version & Date	Change description
1.0.0 15.12.2023	SIMATIC Systems Support First released version
1.1.0 21.06.2024	SIMATIC Systems Support Included LPMLV2022 constants as a local constants in the block
1.2.0 05.09.2024	SIMATIC Systems Support Optimized code.

3.5 History

3.5.1 LUC_UpdateModeStateHistory (FC / V1.2.0)

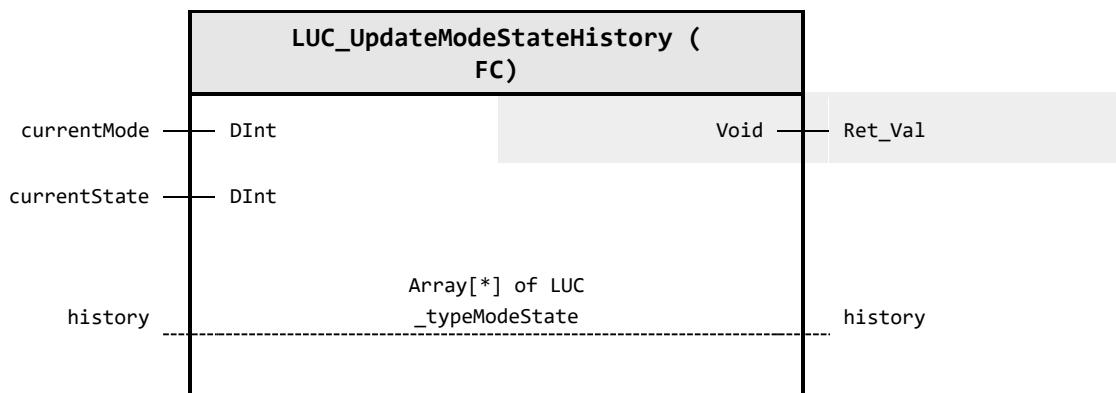
Author: Siemens Digital Industries

Short description

Update the mode and state history.

Interface description

Block interface



Input parameter

Identifier	Data type	Description
currentMode	DInt	Current unit mode
currentState	DInt	Current unit state

Output parameter

Identifier	Data type	Description
Ret_Val	Void	Void - Function has no return value

In/Out parameter

Identifier	Data type	Description
history	Array[*] of LUC_typeModeState	Duration of stay in one mode and state

User defined datatype(s)

LUC_typeModeState (UDT / V1.2.0)

Duration of stay in one mode and state

Identifier	Data type	Default value	Description
mode	DInt	0	OMAC identifier of mode
state	DInt	0	OMAC identifier of state
duration	LReal	0.0	Time stayed in that status

Functional description

Update the mode and state history. When mode or state change occur, save current mode and state in history[0], and the previous ones in ascending order.

Change log

Version & Date	Change description
1.1.0 21.06.2024	SIMATIC Systems Support First released version
1.2.0 05.09.2024	SIMATIC Systems Support Optimized code.

3.6 RemoteControl

3.6.1 LUC_RemoteCtrl_OpcUa (FB / 1.2.0)

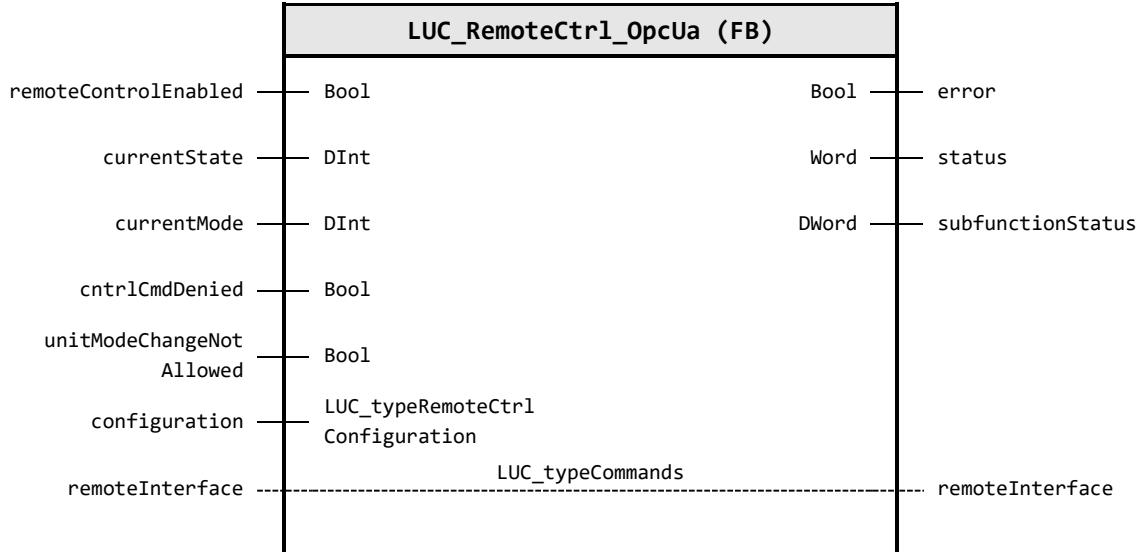
Author: Siemens Digital Industries

Short description

This block calls the OPC UA methods to remote control the Unit. The remote interface is used as an input of the *LUC_InterfaceManager*.

Interface description

Block interface



Input parameter

Identifier	Data type	Default value	Description
remoteControlEnabled	Bool	FALSE	Remote control is active
currentState	DIInt	LPMLV2022_STATE_STOPPED	Current executed state
currentMode	DIInt	LPMLV2022_MODE_MANUAL	Current executed mode
cntrlCmdDenied	Bool	FALSE	Control command denied (output is reset with the next successful CntrlCmd or if input 'CntrlCmd' is set to 0 or if input 'CmdChangeRequest' is set to FALSE)
unitModeChangeNotAllowed	Bool	FALSE	Requested unit mode change denied
configuration	LUC_typeRemoteCtrlConfiguration	---	OPC UA Method calls parameters, to parametrize the functionality of the OPC UA Method calls belonging to the UnitModeStatemanager to enable / disable the method calls

Output parameter

Identifier	Data type	Description
error	Bool	Error occurred
status	Word	Status information
subfunctionStatus	DWord	Status of called subfunction when error occurs

In/Out parameter

Identifier	Data type	Description
remoteInterface	LUC_typeCommands	Module related commands from external systems

Status & Error codes

Code / Value	Identifier / Description
16#0000	STATUS_NO_ERROR No Error during call occurred.
16#7000	STATUS_NO_CALL No call of FB
16#8611	ERR_METHOD_RESET Error in method for reset
16#8612	ERR_METHOD_START Error in method for start
16#8613	ERR_METHOD_STOP Error in method for stop
16#8614	ERR_METHOD_HOLD Error in method for hold
16#8615	ERR_METHOD_UNHOLD Error in method for unhold
16#8616	ERR_METHOD_SUSPEND Error in method for suspend
16#8617	ERR_METHOD_UNSUSPEND Error in method for unsuspend
16#8618	ERR_METHOD_ABORT Error in method for abort
16#8619	ERR_METHOD_CLEAR Error in method for clear
16#8620	ERR_METHOD_COMPLETE Error in method for complete
16#8621	ERR_METHOD_STATE_COMPLETE Error in method for state complete
16#8631	ERR_METHOD_REQUEST_UNIT_MODE Error in method for RequestUnitMode
16#8632	ERR_METHOD_REQUEST_UNIT_STATE Error in method for RequestUnitState

User defined datatype(s)

LUC_typeRemoteCtrlConfiguration (UDT / V1.1.0)

OPC UA Method calls parameters, to parametrize the functionality of the OPC UA Method calls belonging to the UnitModeStatemanager to enable / disable the method calls

Identifier	Data type	Default value	Description
enableMethods	Struct	---	Enabling of methods
reset	Bool	TRUE	Enable command reset
start	Bool	TRUE	Enable command start
stop	Bool	TRUE	Enable command stop
hold	Bool	TRUE	Enable command hold
unhold	Bool	TRUE	Enable command unhold
suspend	Bool	TRUE	Enable command suspend
unsuspend	Bool	TRUE	Enable command unsuspend
abort	Bool	TRUE	Enable command abort
clear	Bool	TRUE	Enable command clear
complete	Bool	TRUE	Enable command complet
stateComplete	Bool	TRUE	Enable command state completed
requestUnitMode	Bool	TRUE	Enable command request unit mode change
requestUnitState	Bool	TRUE	Enable command request unit state change

LUC_typeCommands (UDT / V1.1.0)

Module related commands from external systems

Identifier	Data type	Default value	Description
modeRequests	LUC_typeModeRequests	---	Commands to request a mode change
stateCommands	LUC_typeStateCommands	---	Commands to request a state change

Functional description

This module provides a collection of OPC UA methods to control the *LUC_InterfaceManager* module via the remote interface.

To ensure the simplest possible use, it is only necessary to connect the instance of the *LUC_InterfaceManager*.

In the configuration interface, individual methods can be selected or deselected.
Make sure that the instance DB is correctly defined in the OPC UA interface.

Change log

Version & Date	Change description
1.0.0 15.12.2023	SIMATIC Systems Support First released version
1.1.0 21.06.2024	SIMATIC Systems Support Extend number of existing modes from 8 to 15.
1.2.0 05.09.2024	SIMATIC Systems Support Fixed SetUnitMode and SetUnitState bug. Removed the forcing of the ENO to true.

3.7 RemoteControl / OPC_UA_Methods

3.7.1 LUC_OpcUaMethod_Abort (FB / 1.2.0)

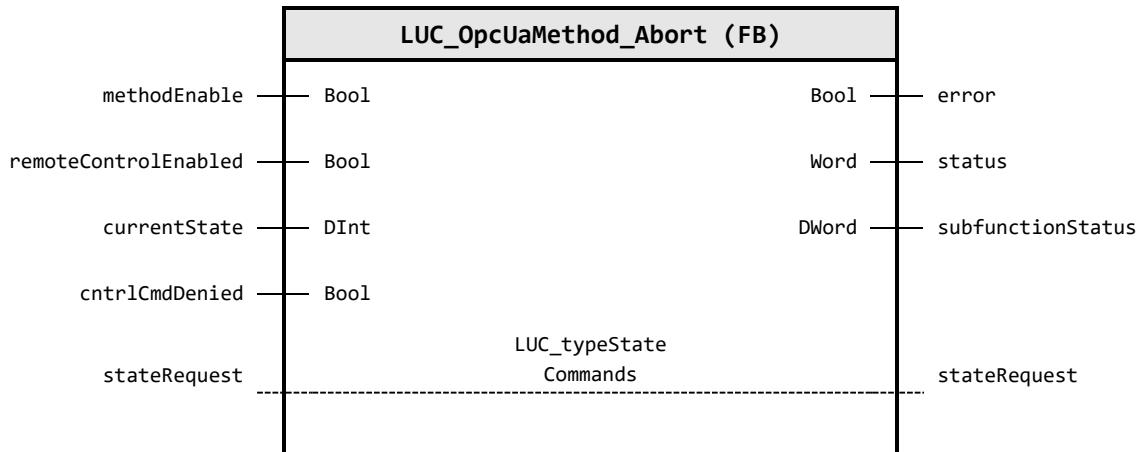
Author: Siemens Digital Industries

Short description

This Block represents one OPC UA Method interface to send a defined command to the *LUC_InterfaceManager* instance. The Method is called in the method collector *LUC_RemoteCtrl_OpcUa*, where all other available OPC UA methods are called as well. Therefore, there is no need to call this single method.

Interface description

Block interface



Input parameter

Identifier	Data type	Default value	Description
methodEnable	Bool	FALSE	Method is enabled
remoteControlEnabled	Bool	FALSE	Remote control is active
currentState	DInt	LPMLV2022_STATE_STOPPED	Current executed state
cntrlCmdDenied	Bool	FALSE	Control command denied (output is reset with the next successful CntrlCmd or if input 'CntrlCmd' is set to 0 or if input 'CmdChangeRequest' is set to FALSE)

Output parameter

Identifier	Data type	Description
error	Bool	Error occurred
status	Word	Status identifier
subfunctionStatus	DWord	Status of called subfunction when error occurs

In/Out parameter

Identifier	Data type	Description
stateRequest	LUC_typeStateCommands	Commands to request a state change

Status & Error codes

Code / Value	Identifier / Description
16#0000	STATUS_EXECUTION_FINISHED Execution finished without errors
16#7000	STATUS_NO_CALL No call of FB
16#8600	ERR_METHOD_PRE Error in block OPC_UA_ServerMethodPre
16#8601	ERR_METHOD_POST Error in block OPC_UA_ServerMethodPost

User defined datatype(s)

LUC_typeStateCommands (UDT / V1.1.0)

Unit State Commands

Identifier	Data type	Default value	Description
reset	Bool	FALSE	OMAC PackMLV2022 control command Reset
start	Bool	FALSE	OMAC PackMLV2022 control command Start
stop	Bool	FALSE	OMAC PackMLV2022 control command Stop
hold	Bool	FALSE	OMAC PackMLV2022 control command Hold
unhold	Bool	FALSE	OMAC PackMLV2022 control command Unhold
suspend	Bool	FALSE	OMAC PackMLV2022 control command Suspend
unsuspend	Bool	FALSE	OMAC PackMLV2022 control command Unsuspend
abort	Bool	FALSE	OMAC PackMLV2022 control command Abort
clear	Bool	FALSE	OMAC PackMLV2022 control command Clear
complete	Bool	FALSE	OMAC PackMLV2022 control command Complete
stateComplete	Bool	FALSE	OMAC PackMLV2022 control command State Complete

Change log

Version & Date	Change description
1.0.0 15.12.2023	SIMATIC Systems Support First released version
1.1.0 21.06.2024	SIMATIC Systems Support Included LPMLV2022 constants as a local constants in the block
1.2.0 05.09.2024	SIMATIC Systems Support Removed the forcing of the ENO to true.

3.7.2 LUC_OpcUaMethod_Clear (FB / 1.2.0)

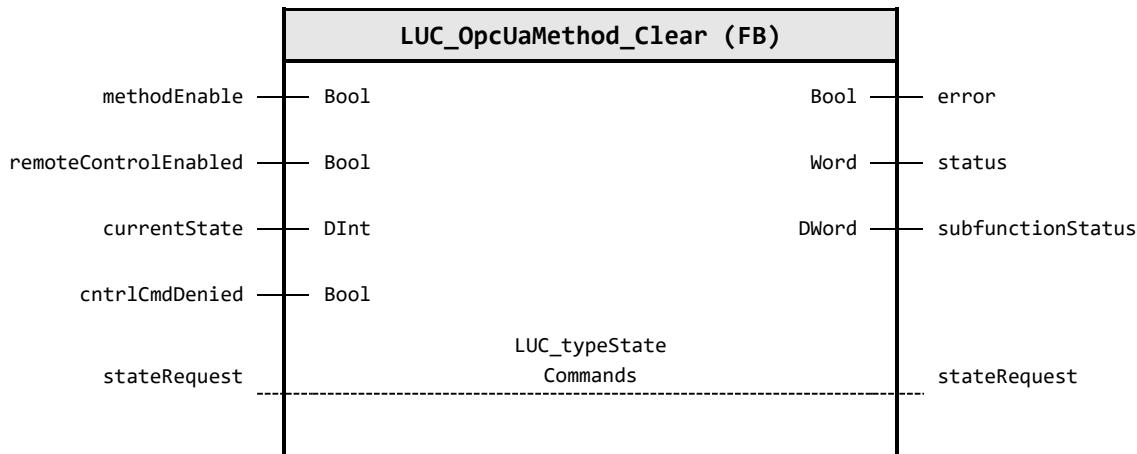
Author: Siemens Digital Industries

Short description

This Block represents one OPC UA Method interface to send a defined command to the *LUC_InterfaceManager* instance. The Method is called in the method collector *LUC_RemoteCtrl_OpcUa*, where all other available OPC UA methods are called as well. Therefore, there is no need to call this single method.

Interface description

Block interface



Input parameter

Identifier	Data type	Default value	Description
methodEnable	Bool	FALSE	Method is enabled
remoteControl Enabled	Bool	FALSE	Remote control is active
currentState	DInt	LPMLV2022_ STATE_STOP PED	Current executed state
cntrlCmdDenie d	Bool	FALSE	Control command denied (output is reset with the next successful CntrlCmd or if input 'CntrlCmd' is set to 0 or if input 'CmdChangeRequest' is set to FALSE)

Output parameter

Identifier	Data type	Description
error	Bool	Error occurred
status	Word	Status identifier
subfunctionSta tus	DWord	Status of called subfunction when error occurs

In/Out parameter

Identifier	Data type	Description
stateRequest	LUC_typeStat eCommands	Commands to request a state change

Status & Error codes

Code / Value	Identifier / Description
16#0000	STATUS_EXECUTION_FINISHED Execution finished without errors
16#7000	STATUS_NO_CALL No call of FB
16#8600	ERR_METHOD_PRE Error in block OPC_UA_ServerMethodPre
16#8601	ERR_METHOD_POST Error in block OPC_UA_ServerMethodPost

User defined datatype(s)

LUC_typeStateCommands (UDT / V1.1.0)

Unit State Commands

Identifier	Data type	Default value	Description
reset	Bool	FALSE	OMAC PackMLV2022 control command Reset
start	Bool	FALSE	OMAC PackMLV2022 control command Start
stop	Bool	FALSE	OMAC PackMLV2022 control command Stop
hold	Bool	FALSE	OMAC PackMLV2022 control command Hold
unhold	Bool	FALSE	OMAC PackMLV2022 control command Unhold
suspend	Bool	FALSE	OMAC PackMLV2022 control command Suspend
unsuspend	Bool	FALSE	OMAC PackMLV2022 control command Unsuspend
abort	Bool	FALSE	OMAC PackMLV2022 control command Abort
clear	Bool	FALSE	OMAC PackMLV2022 control command Clear
complete	Bool	FALSE	OMAC PackMLV2022 control command Complete
stateComplete	Bool	FALSE	OMAC PackMLV2022 control command State Complete

Change log

Version & Date	Change description
1.0.0 15.12.2023	SIMATIC Systems Support First released version
1.1.0 21.06.2024	SIMATIC Systems Support Included LPMLV2022 constants as a local constants in the block
1.2.0 05.09.2024	SIMATIC Systems Support Removed the forcing of the ENO to true.

3.7.3 LUC_OpcUaMethod_Complete (FB / 1.2.0)

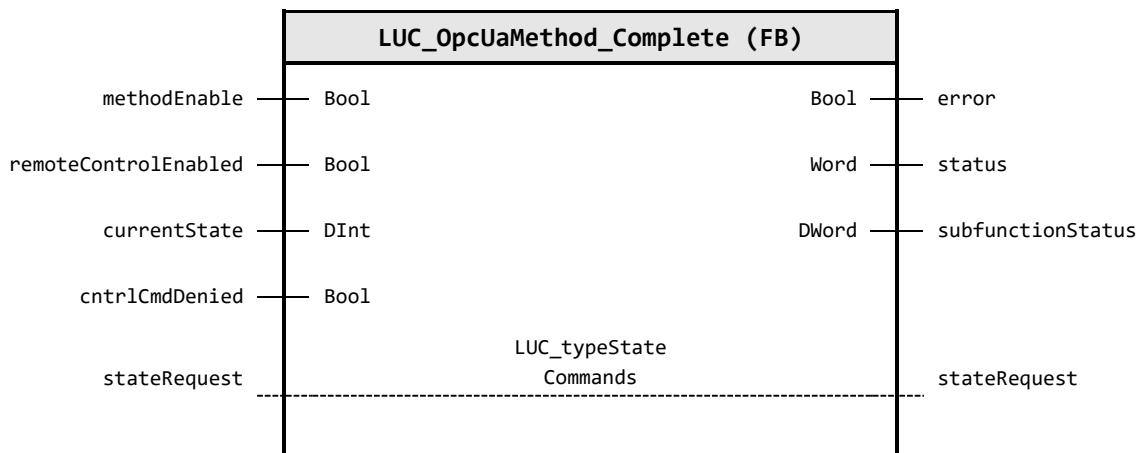
Author: Siemens Digital Industries

Short description

This Block represents one OPC UA Method interface to send a defined command to the *LUC_InterfaceManager* instance. The Method is called in the method collector *LUC_RemoteCtrl_OpcUa*, where all other available OPC UA methods are called as well. Therefore, there is no need to call this single method.

Interface description

Block interface



Input parameter

Identifier	Data type	Default value	Description
methodEnable	Bool	FALSE	Method is enabled
remoteControl Enabled	Bool	FALSE	Remote control is active
currentState	DInt	LPMLV2022_STATE_STOPPED	Current executed state
cntrlCmdDenied	Bool	FALSE	Control command denied (output is reset with the next successful CntrlCmd or if input 'CntrlCmd' is set to 0 or if input 'CmdChangeRequest' is set to FALSE)

Output parameter

Identifier	Data type	Description
error	Bool	Error occurred
status	Word	Status identifier
subfunctionStatus	DWord	Status of called subfunction when error occurs

In/Out parameter

Identifier	Data type	Description
stateRequest	LUC_typeStateCommands	Commands to request a state change

3 Program blocks

Status & Error codes

Code / Value	Identifier / Description
16#0000	STATUS_EXECUTION_FINISHED Execution finished without errors
16#7000	STATUS_NO_CALL No call of FB
16#8600	ERR_METHOD_PRE Error in block OPC_UA_ServerMethodPre
16#8601	ERR_METHOD_POST Error in block OPC_UA_ServerMethodPost

User defined datatype(s)

LUC_typeStateCommands (UDT / V1.1.0)

Unit State Commands

Identifier	Data type	Default value	Description
reset	Bool	FALSE	OMAC PackMLV2022 control command Reset
start	Bool	FALSE	OMAC PackMLV2022 control command Start
stop	Bool	FALSE	OMAC PackMLV2022 control command Stop
hold	Bool	FALSE	OMAC PackMLV2022 control command Hold
unhold	Bool	FALSE	OMAC PackMLV2022 control command Unhold
suspend	Bool	FALSE	OMAC PackMLV2022 control command Suspend
unsuspend	Bool	FALSE	OMAC PackMLV2022 control command Unsuspend
abort	Bool	FALSE	OMAC PackMLV2022 control command Abort
clear	Bool	FALSE	OMAC PackMLV2022 control command Clear
complete	Bool	FALSE	OMAC PackMLV2022 control command Complete
stateComplete	Bool	FALSE	OMAC PackMLV2022 control command State Complete

Change log

Version & Date	Change description
1.0.0 15.12.2023	SIMATIC Systems Support First released version
1.1.0 21.06.2024	SIMATIC Systems Support Included LPMLV2022 constants as a local constants in the block
1.2.0 05.09.2024	SIMATIC Systems Support Removed the forcing of the ENO to true.

3.7.4 LUC_OpcUaMethod_Hold (FB / 1.2.0)

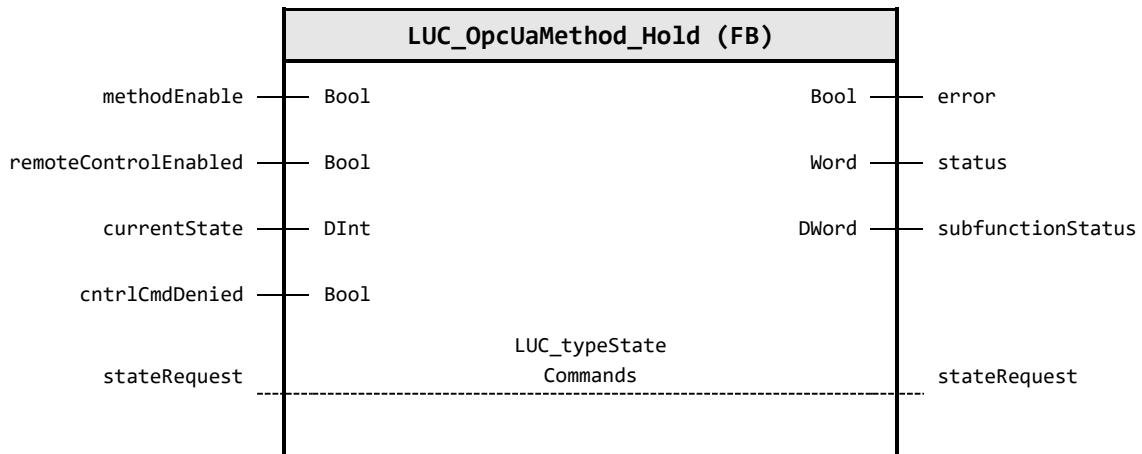
Author: Siemens Digital Industries

Short description

This Block represents one OPC UA Method interface to send a defined command to the *LUC_InterfaceManager* instance. The Method is called in the method collector *LUC_RemoteCtrl_OpcUa*, where all other available OPC UA methods are called as well. Therefore, there is no need to call this single method.

Interface description

Block interface



Input parameter

Identifier	Data type	Default value	Description
methodEnable	Bool	FALSE	Method is enabled
remoteControl Enabled	Bool	FALSE	Remote control is active
currentState	DInt	LPMLV2022_STATE_STOPPED	Current executed state
cntrlCmdDenied	Bool	FALSE	Control command denied (output is reset with the next successful CntrlCmd or if input 'CntrlCmd' is set to 0 or if input 'CmdChangeRequest' is set to FALSE)

Output parameter

Identifier	Data type	Description
error	Bool	Error occurred
status	Word	Status identifier
subfunctionStatus	DWord	Status of called subfunction when error occurs

In/Out parameter

Identifier	Data type	Description
stateRequest	LUC_typeStateCommands	Commands to request a state change

3 Program blocks

Status & Error codes

Code / Value	Identifier / Description
16#0000	STATUS_EXECUTION_FINISHED Execution finished without errors
16#7000	STATUS_NO_CALL No call of FB
16#8600	ERR_METHOD_PRE Error in block OPC_UA_ServerMethodPre
16#8601	ERR_METHOD_POST Error in block OPC_UA_ServerMethodPost

User defined datatype(s)

LUC_typeStateCommands (UDT / V1.1.0)

Unit State Commands

Identifier	Data type	Default value	Description
reset	Bool	FALSE	OMAC PackMLV2022 control command Reset
start	Bool	FALSE	OMAC PackMLV2022 control command Start
stop	Bool	FALSE	OMAC PackMLV2022 control command Stop
hold	Bool	FALSE	OMAC PackMLV2022 control command Hold
unhold	Bool	FALSE	OMAC PackMLV2022 control command Unhold
suspend	Bool	FALSE	OMAC PackMLV2022 control command Suspend
unsuspend	Bool	FALSE	OMAC PackMLV2022 control command Unsuspend
abort	Bool	FALSE	OMAC PackMLV2022 control command Abort
clear	Bool	FALSE	OMAC PackMLV2022 control command Clear
complete	Bool	FALSE	OMAC PackMLV2022 control command Complete
stateComplete	Bool	FALSE	OMAC PackMLV2022 control command State Complete

Change log

Version & Date	Change description
1.0.0 15.12.2023	SIMATIC Systems Support First released version
1.1.0 21.06.2024	SIMATIC Systems Support Included LPMLV2022 constants as a local constants in the block
1.2.0 05.09.2024	SIMATIC Systems Support Removed the forcing of the ENO to true.

3.7.5 LUC_OpcUaMethod_Reset (FB / 1.2.0)

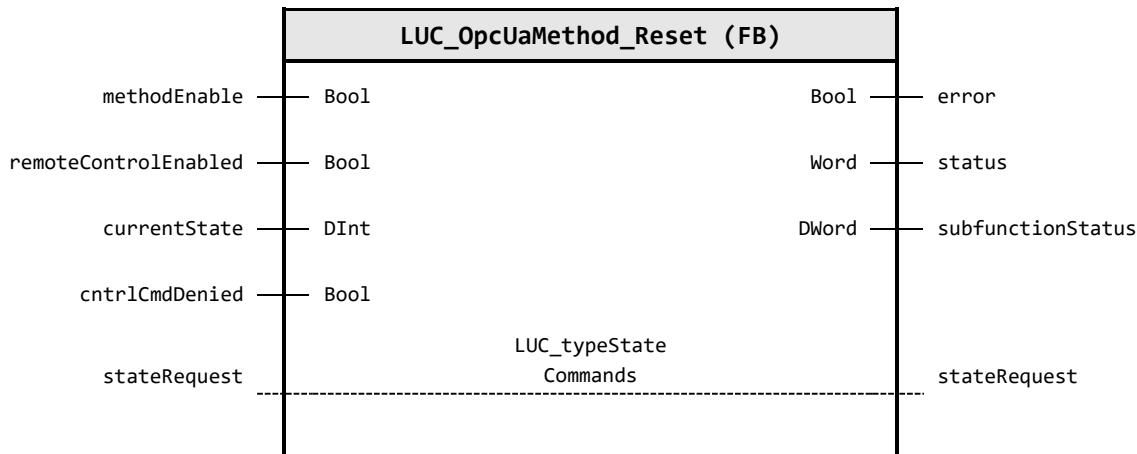
Author: Siemens Digital Industries

Short description

This Block represents one OPC UA Method interface to send a defined command to the *LUC_InterfaceManager* instance. The Method is called in the method collector *LUC_RemoteCtrl_OpcUa*, where all other available OPC UA methods are called as well. Therefore, there is no need to call this single method.

Interface description

Block interface



Input parameter

Identifier	Data type	Default value	Description
methodEnable	Bool	FALSE	Method is enabled
remoteControlEnabled	Bool	FALSE	Remote control is active
currentState	DIInt	LPMLV2022_STATE_STOPPED	Current state of the unit
cntrlCmdDenied	Bool	FALSE	Control command denied (output is reset with the next successful CntrlCmd or if input 'CntrlCmd' is set to 0 or if input 'CmdChangeRequest' is set to FALSE)

Output parameter

Identifier	Data type	Description
error	Bool	Error occurred
status	Word	Status identifier
subfunctionStatus	DWord	Status of called subfunction when error occurs

In/Out parameter

Identifier	Data type	Description
stateRequest	LUC_typeStateCommands	Commands to request a state change

Status & Error codes

Code / Value	Identifier / Description
16#0000	STATUS_EXECUTION_FINISHED Execution finished without errors
16#7000	STATUS_NO_CALL No call of FB
16#8600	ERR_METHOD_PRE Error in block OPC_UA_ServerMethodPre
16#8601	ERR_METHOD_POST Error in block OPC_UA_ServerMethodPost

User defined datatype(s)

LUC_typeStateCommands (UDT / V1.1.0)

Unit State Commands

Identifier	Data type	Default value	Description
reset	Bool	FALSE	OMAC PackMLV2022 control command Reset
start	Bool	FALSE	OMAC PackMLV2022 control command Start
stop	Bool	FALSE	OMAC PackMLV2022 control command Stop
hold	Bool	FALSE	OMAC PackMLV2022 control command Hold
unhold	Bool	FALSE	OMAC PackMLV2022 control command Unhold
suspend	Bool	FALSE	OMAC PackMLV2022 control command Suspend
unsuspend	Bool	FALSE	OMAC PackMLV2022 control command Unsuspend
abort	Bool	FALSE	OMAC PackMLV2022 control command Abort
clear	Bool	FALSE	OMAC PackMLV2022 control command Clear
complete	Bool	FALSE	OMAC PackMLV2022 control command Complete
stateComplete	Bool	FALSE	OMAC PackMLV2022 control command State Complete

Change log

Version & Date	Change description
1.0.0 15.12.2023	SIMATIC Systems Support First released version
1.1.0 21.06.2024	SIMATIC Systems Support Included LPMLV2022 constants as a local constants in the block
1.2.0 05.09.2024	SIMATIC Systems Support Removed the forcing of the ENO to true.

3.7.6 LUC_OpcUaMethod_SetUnitMode (FB / 1.2.0)

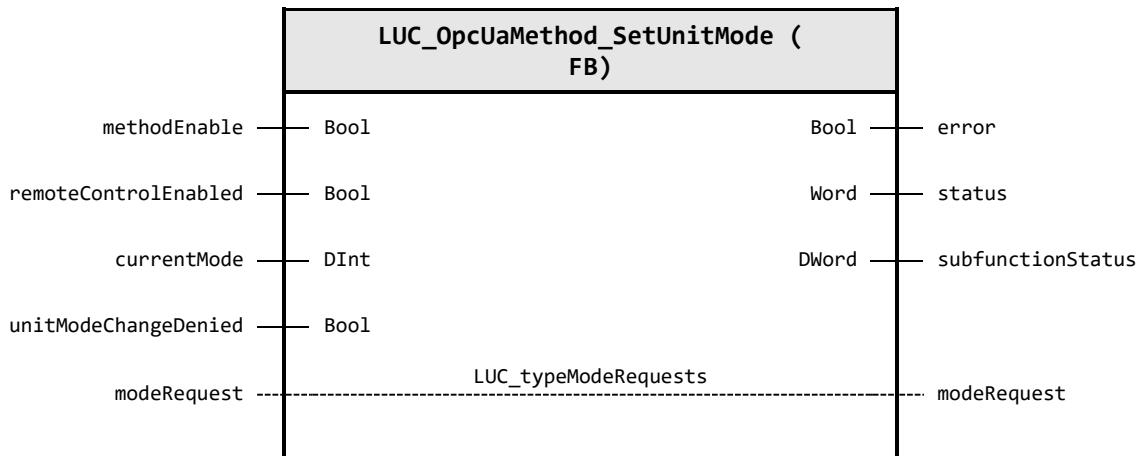
Author: Siemens Digital Industries

Short description

This Block represents one OPC UA Method interface to send a defined command to the *LUC_InterfaceManager* instance. The Method is called in the method collector *LUC_RemoteCtrl_OpcUa*, where all other available OPC UA methods are called as well. Therefore, there is no need to call this single method.

Interface description

Block interface



Input parameter

Identifier	Data type	Default value	Description
methodEnable	Bool	FALSE	Method is enabled
remoteControl Enabled	Bool	FALSE	Remote control is active
currentMode	DInt	LPMLV2022_ STATE_STOP PED	Current selected mode of the unit
unitModeChan geDenied	Bool	FALSE	Requested unit mode change denied (output is reset with the next successful unit mode change or if input 'UnitMode' is set to 0 or if input 'UnitModeChangeRequest' is set to FALSE)

Output parameter

Identifier	Data type	Description
error	Bool	Error occurred
status	Word	Status identifier
subfunctionSta tus	DWord	Status of called subfunction when error occurs

In/Out parameter

Identifier	Data type	Description
modeRequest	LUC_typeMod eRequests	Commands to request a mode change

3 Program blocks

Status & Error codes

Code / Value	Identifier / Description
16#0000	STATUS_EXECUTION_FINISHED Execution finished without errors
16#7000	STATUS_NO_CALL No call of FB
16#8600	ERR_METHOD_PRE Error in block OPC_UA_ServerMethodPre
16#8601	ERR_METHOD_POST Error in block OPC_UA_ServerMethodPost

User defined datatype(s)

LUC_typeModeRequests (UDT / V1.1.0)

Unit Mode requests

Identifier	Data type	Default value	Description
production	Bool	false	Request to change to "Production"
maintenance	Bool	false	Request to change to "Maintenance"
manual	Bool	false	Request to change to "Manual"
userMode01	Bool	false	Request to change to "UserMode01"
userMode02	Bool	false	Request to change to "UserMode02"
userMode03	Bool	false	Request to change to "UserMode03"
userMode04	Bool	false	Request to change to "UserMode04"
userMode05	Bool	false	Request to change to "UserMode05"
userMode06	Bool	false	Request to change to "UserMode06"
userMode07	Bool	false	Request to change to "UserMode07"
userMode08	Bool	false	Request to change to "UserMode08"
userMode09	Bool	false	Request to change to "UserMode09"
userMode10	Bool	false	Request to change to "UserMode10"
userMode11	Bool	false	Request to change to "UserMode11"
userMode12	Bool	false	Request to change to "UserMode12"
userMode13	Bool	false	Request to change to "UserMode13"

Change log

Version & Date	Change description
1.0.0 15.12.2023	SIMATIC Systems Support First released version
1.1.0 21.06.2024	SIMATIC Systems Support Included LPMLV2022 constants as a local constants in the block. Extended number of existing modes from 8 to 15.
1.2.0 05.09.2024	SIMATIC Systems Support Removed the forcing of the ENO to true.

3.7.7 LUC_OpcUaMethod_SetUnitState (FB / 1.2.0)

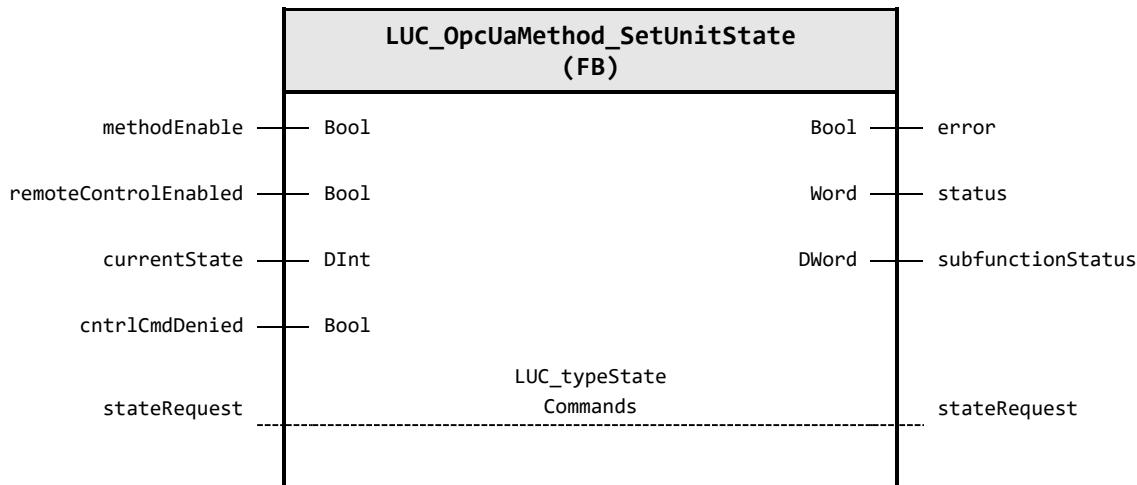
Author: Siemens Digital Industries

Short description

This Block represents one OPC UA Method interface to send a defined command to the *LUC_InterfaceManager* instance. The Method is called in the method collector *LUC_RemoteCtrl1_OpcUa*, where all other available OPC UA methods are called as well. Therefore, there is no need to call this single method.

Interface description

Block interface



Input parameter

Identifier	Data type	Default value	Description
methodEnable	Bool	FALSE	Method is enabled
remoteControlEnabled	Bool	FALSE	Remote control is active
currentState	DInt	LPMLV2022_STATE_STOPPED	Current state of the unit
cntrlCmdDenied	Bool	FALSE	Control command denied (output is reset with the next successful CntrlCmd or if input 'CntrlCmd' is set to 0 or if input 'CmdChangeRequest' is set to FALSE)

Output parameter

Identifier	Data type	Description
error	Bool	Error occurred
status	Word	Status identifier
subfunctionStatus	DWord	Status of called subfunction when error occurs

In/Out parameter

Identifier	Data type	Description
stateRequest	LUC_typeStateCommands	Commands to request a state change

Status & Error codes

Code / Value	Identifier / Description
16#0000	STATUS_EXECUTION_FINISHED Execution finished without errors
16#7000	STATUS_NO_CALL No call of FB
16#8600	ERR_METHOD_PRE Error in block OPC_UA_ServerMethodPre
16#8601	ERR_METHOD_POST Error in block OPC_UA_ServerMethodPost

User defined datatype(s)

LUC_typeStateCommands (UDT / V1.1.0)

Unit State Commands

Identifier	Data type	Default value	Description
reset	Bool	FALSE	OMAC PackMLV2022 control command Reset
start	Bool	FALSE	OMAC PackMLV2022 control command Start
stop	Bool	FALSE	OMAC PackMLV2022 control command Stop
hold	Bool	FALSE	OMAC PackMLV2022 control command Hold
unhold	Bool	FALSE	OMAC PackMLV2022 control command Unhold
suspend	Bool	FALSE	OMAC PackMLV2022 control command Suspend
unsuspend	Bool	FALSE	OMAC PackMLV2022 control command Unsuspend
abort	Bool	FALSE	OMAC PackMLV2022 control command Abort
clear	Bool	FALSE	OMAC PackMLV2022 control command Clear
complete	Bool	FALSE	OMAC PackMLV2022 control command Complete
stateComplete	Bool	FALSE	OMAC PackMLV2022 control command State Complete

Change log

Version & Date	Change description
1.0.0 15.12.2023	SIMATIC Systems Support First released version
1.1.0 21.06.2024	SIMATIC Systems Support Included LPMLV2022 constants as a local constants in the block
1.2.0 05.09.2024	SIMATIC Systems Support Modified the interface for _statLastCycleState_ from Temp to Static. Removed the forcing of the ENO to true.

3.7.8 LUC_OpcUaMethod_Start (FB / 1.2.0)

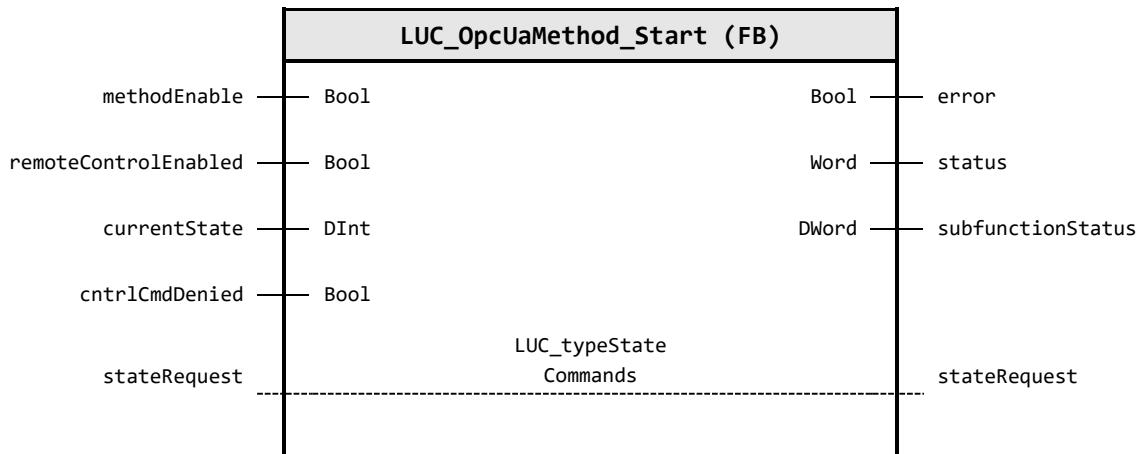
Author: Siemens Digital Industries

Short description

This Block represents one OPC UA Method interface to send a defined command to the *LUC_InterfaceManager* instance. The Method is called in the method collector *LUC_RemoteCtrl_OpcUa*, where all other available OPC UA methods are called as well. Therefore, there is no need to call this single method.

Interface description

Block interface



Input parameter

Identifier	Data type	Default value	Description
methodEnable	Bool	FALSE	Method is enabled
remoteControl Enabled	Bool	FALSE	Remote control is active
currentState	DIInt	LPMLV2022_ STATE_STOP PED	Current executed state
cntrlCmdDenie d	Bool	FALSE	Control command denied (output is reset with the next successful CntrlCmd or if input 'CntrlCmd' is set to 0 or if input 'CmdChangeRequest' is set to FALSE)

Output parameter

Identifier	Data type	Description
error	Bool	Error occurred
status	Word	Status identifier
subfunctionSta tus	DWord	Status of called subfunction when error occurs

In/Out parameter

Identifier	Data type	Description
stateRequest	LUC_typeStat eCommands	Commands to request a state change

3 Program blocks

Status & Error codes

Code / Value	Identifier / Description
16#0000	STATUS_EXECUTION_FINISHED Execution finished without errors
16#7000	STATUS_NO_CALL No call of FB
16#8600	ERR_METHOD_PRE Error in block OPC_UA_ServerMethodPre
16#8601	ERR_METHOD_POST Error in block OPC_UA_ServerMethodPost

User defined datatype(s)

LUC_typeStateCommands (UDT / V1.1.0)

Unit State Commands

Identifier	Data type	Default value	Description
reset	Bool	FALSE	OMAC PackMLV2022 control command Reset
start	Bool	FALSE	OMAC PackMLV2022 control command Start
stop	Bool	FALSE	OMAC PackMLV2022 control command Stop
hold	Bool	FALSE	OMAC PackMLV2022 control command Hold
unhold	Bool	FALSE	OMAC PackMLV2022 control command Unhold
suspend	Bool	FALSE	OMAC PackMLV2022 control command Suspend
unsuspend	Bool	FALSE	OMAC PackMLV2022 control command Unsuspend
abort	Bool	FALSE	OMAC PackMLV2022 control command Abort
clear	Bool	FALSE	OMAC PackMLV2022 control command Clear
complete	Bool	FALSE	OMAC PackMLV2022 control command Complete
stateComplete	Bool	FALSE	OMAC PackMLV2022 control command State Complete

Change log

Version & Date	Change description
1.0.0 15.12.2023	SIMATIC Systems Support First released version
1.1.0 21.06.2024	SIMATIC Systems Support Included LPMLV2022 constants as a local constants in the block
1.2.0 05.09.2024	SIMATIC Systems Support Removed the forcing of the ENO to true.

3.7.9 LUC_OpcUaMethod_StateComplete (FB / 1.2.0)

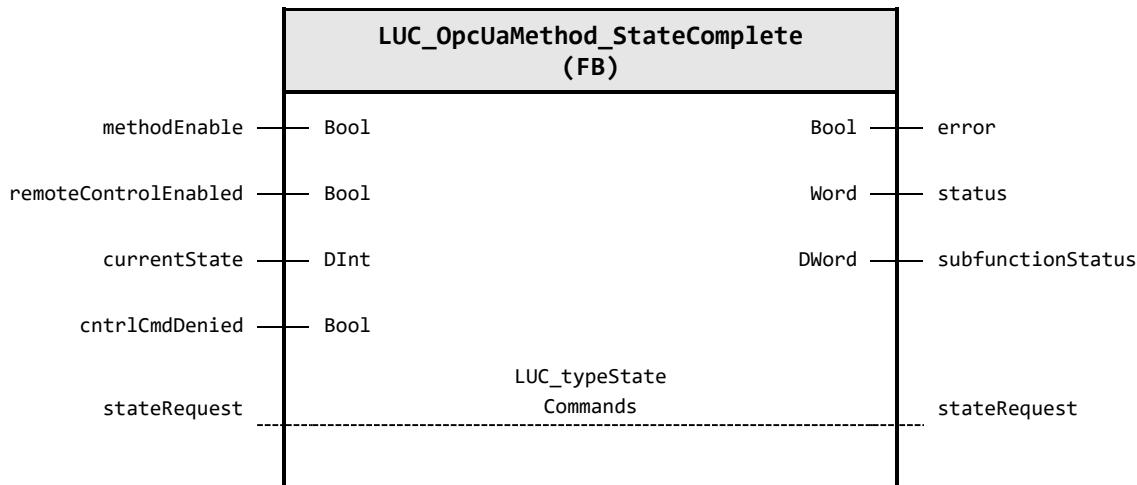
Author: Siemens Digital Industries

Short description

This Block represents one OPC UA Method interface to send a defined command to the *LUC_InterfaceManager* instance. The Method is called in the method collector *LUC_RemoteCtrl_OpcUa*, where all other available OPC UA methods are called as well. Therefore, there is no need to call this single method.

Interface description

Block interface



Input parameter

Identifier	Data type	Default value	Description
methodEnable	Bool	FALSE	Method is enabled
remoteControlEnabled	Bool	FALSE	Remote control is active
currentState	DInt	LPMLV2022_STATE_STOPPED	Current executed state
cntrlCmdDenied	Bool	FALSE	Control command denied (output is reset with the next successful CntrlCmd or if input 'CntrlCmd' is set to 0 or if input 'CmdChangeRequest' is set to FALSE)

Output parameter

Identifier	Data type	Description
error	Bool	Error occurred
status	Word	Status identifier
subfunctionStatus	DWord	Status of called subfunction when error occurs

In/Out parameter

Identifier	Data type	Description
stateRequest	LUC_typeStateCommands	Commands to request a state change

Status & Error codes

Code / Value	Identifier / Description
16#0000	STATUS_EXECUTION_FINISHED Execution finished without errors
16#7000	STATUS_NO_CALL No call of FB
16#8600	ERR_METHOD_PRE Error in block OPC_UA_ServerMethodPre
16#8601	ERR_METHOD_POST Error in block OPC_UA_ServerMethodPost

User defined datatype(s)

LUC_typeStateCommands (UDT / V1.1.0)

Unit State Commands

Identifier	Data type	Default value	Description
reset	Bool	FALSE	OMAC PackMLV2022 control command Reset
start	Bool	FALSE	OMAC PackMLV2022 control command Start
stop	Bool	FALSE	OMAC PackMLV2022 control command Stop
hold	Bool	FALSE	OMAC PackMLV2022 control command Hold
unhold	Bool	FALSE	OMAC PackMLV2022 control command Unhold
suspend	Bool	FALSE	OMAC PackMLV2022 control command Suspend
unsuspend	Bool	FALSE	OMAC PackMLV2022 control command Unsuspend
abort	Bool	FALSE	OMAC PackMLV2022 control command Abort
clear	Bool	FALSE	OMAC PackMLV2022 control command Clear
complete	Bool	FALSE	OMAC PackMLV2022 control command Complete
stateComplete	Bool	FALSE	OMAC PackMLV2022 control command State Complete

Change log

Version & Date	Change description
1.0.0 15.12.2023	SIMATIC Systems Support First released version
1.1.0 21.06.2024	SIMATIC Systems Support Included LPMLV2022 constants as a local constants in the block
1.2.0 05.09.2024	SIMATIC Systems Support Removed the forcing of the ENO to true.

3.7.10 LUC_OpcUaMethod_Stop (FB / 1.2.0)

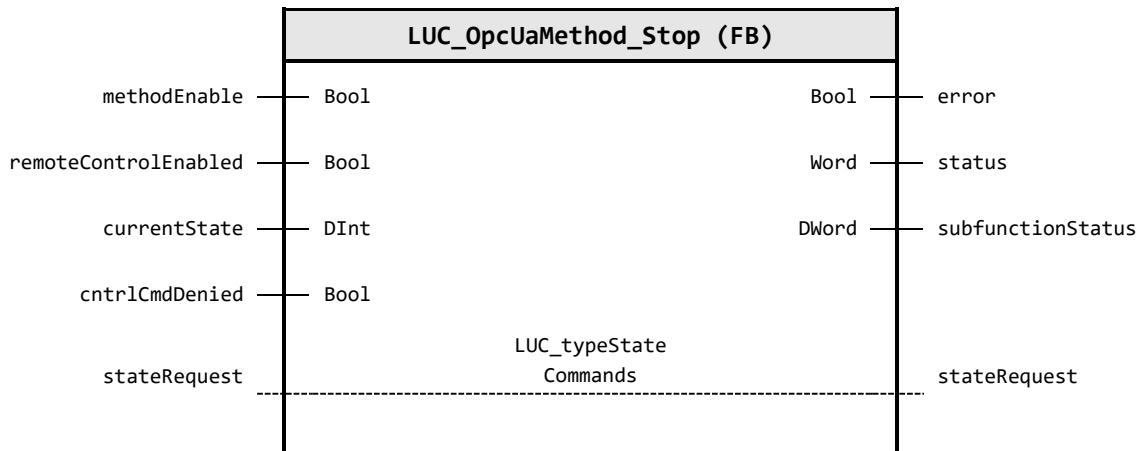
Author: Siemens Digital Industries

Short description

This Block represents one OPC UA Method interface to send a defined command to the *LUC_InterfaceManager* instance. The Method is called in the method collector *LUC_RemoteCtrl_OpcUa*, where all other available OPC UA methods are called as well. Therefore, there is no need to call this single method.

Interface description

Block interface



Input parameter

Identifier	Data type	Default value	Description
methodEnable	Bool	FALSE	Method is enabled
remoteControlEnabled	Bool	FALSE	Remote control is active
currentState	DInt	LPMLV2022_STATE_STOPPED	Current executed state
cntrlCmdDenied	Bool	FALSE	Control command denied (output is reset with the next successful CntrlCmd or if input 'CntrlCmd' is set to 0 or if input 'CmdChangeRequest' is set to FALSE)

Output parameter

Identifier	Data type	Description
error	Bool	Error occurred
status	Word	Status identifier
subfunctionStatus	DWord	Status of called subfunction when error occurs

In/Out parameter

Identifier	Data type	Description
stateRequest	LUC_typeStateCommands	Commands to request a state change

Status & Error codes

Code / Value	Identifier / Description
16#0000	STATUS_EXECUTION_FINISHED Execution finished without errors
16#7000	STATUS_NO_CALL No call of FB
16#8600	ERR_METHOD_PRE Error in block OPC_UA_ServerMethodPre
16#8601	ERR_METHOD_POST Error in block OPC_UA_ServerMethodPost

User defined datatype(s)

LUC_typeStateCommands (UDT / V1.1.0)

Unit State Commands

Identifier	Data type	Default value	Description
reset	Bool	FALSE	OMAC PackMLV2022 control command Reset
start	Bool	FALSE	OMAC PackMLV2022 control command Start
stop	Bool	FALSE	OMAC PackMLV2022 control command Stop
hold	Bool	FALSE	OMAC PackMLV2022 control command Hold
unhold	Bool	FALSE	OMAC PackMLV2022 control command Unhold
suspend	Bool	FALSE	OMAC PackMLV2022 control command Suspend
unsuspend	Bool	FALSE	OMAC PackMLV2022 control command Unsuspend
abort	Bool	FALSE	OMAC PackMLV2022 control command Abort
clear	Bool	FALSE	OMAC PackMLV2022 control command Clear
complete	Bool	FALSE	OMAC PackMLV2022 control command Complete
stateComplete	Bool	FALSE	OMAC PackMLV2022 control command State Complete

Change log

Version & Date	Change description
1.0.0 15.12.2023	SIMATIC Systems Support First released version
1.1.0 21.06.2024	SIMATIC Systems Support Included LPMLV2022 constants as a local constants in the block
1.2.0 05.09.2024	SIMATIC Systems Support Removed the forcing of the ENO to true.

3.7.11 LUC_OpcUaMethod_Suspend (FB / 1.2.0)

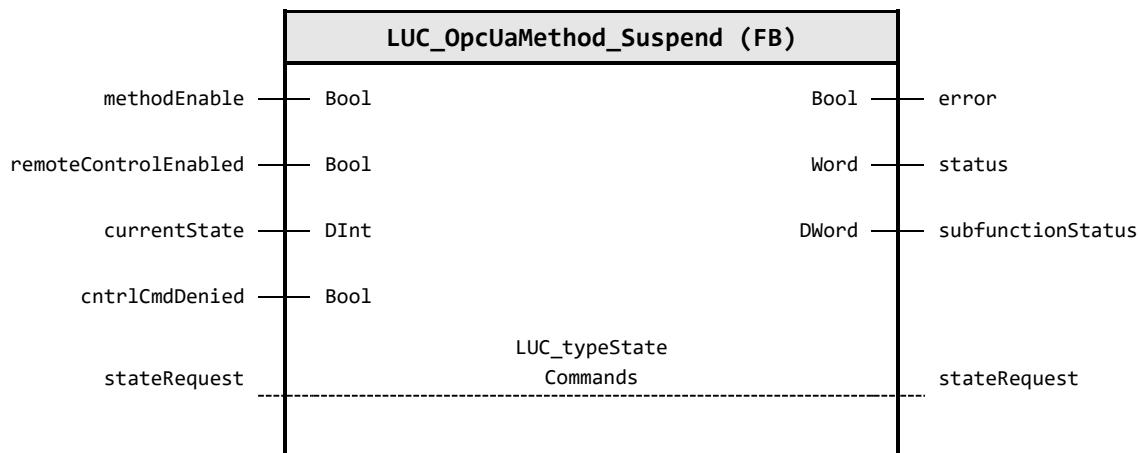
Author: Siemens Digital Industries

Short description

This Block represents one OPC UA Method interface to send a defined command to the *LUC_InterfaceManager* instance. The Method is called in the method collector *LUC_RemoteCtrl_OpcUa*, where all other available OPC UA methods are called as well. Therefore, there is no need to call this single method.

Interface description

Block interface



Input parameter

Identifier	Data type	Default value	Description
methodEnable	Bool	FALSE	Method is enabled
remoteControl Enabled	Bool	FALSE	Remote control is active
currentState	DInt	LPMLV2022_ STATE_STOP PED	Current executed state
cntrlCmdDenie d	Bool	FALSE	Control command denied (output is reset with the next successful CntrlCmd or if input 'CntrlCmd' is set to 0 or if input 'CmdChangeRequest' is set to FALSE)

Output parameter

Identifier	Data type	Description
error	Bool	Error occurred
status	Word	Status identifier
subfunctionSta tus	DWord	Status of called subfunction when error occurs

In/Out parameter

Identifier	Data type	Description
stateRequest	LUC_typeStat eCommands	Commands to request a state change

Status & Error codes

Code / Value	Identifier / Description
16#0000	STATUS_EXECUTION_FINISHED Execution finished without errors
16#7000	STATUS_NO_CALL No call of FB
16#8600	ERR_METHOD_PRE Error in block OPC_UA_ServerMethodPre
16#8601	ERR_METHOD_POST Error in block OPC_UA_ServerMethodPost

User defined datatype(s)

LUC_typeStateCommands (UDT / V1.1.0)

Unit State Commands

Identifier	Data type	Default value	Description
reset	Bool	FALSE	OMAC PackMLV2022 control command Reset
start	Bool	FALSE	OMAC PackMLV2022 control command Start
stop	Bool	FALSE	OMAC PackMLV2022 control command Stop
hold	Bool	FALSE	OMAC PackMLV2022 control command Hold
unhold	Bool	FALSE	OMAC PackMLV2022 control command Unhold
suspend	Bool	FALSE	OMAC PackMLV2022 control command Suspend
unsuspend	Bool	FALSE	OMAC PackMLV2022 control command Unsuspend
abort	Bool	FALSE	OMAC PackMLV2022 control command Abort
clear	Bool	FALSE	OMAC PackMLV2022 control command Clear
complete	Bool	FALSE	OMAC PackMLV2022 control command Complete
stateComplete	Bool	FALSE	OMAC PackMLV2022 control command State Complete

Change log

Version & Date	Change description
1.0.0 15.12.2023	SIMATIC Systems Support First released version
1.1.0 21.06.2024	SIMATIC Systems Support Included LPMLV2022 constants as a local constants in the block
1.2.0 05.09.2024	SIMATIC Systems Support Removed the forcing of the ENO to true.

3.7.12 LUC_OpcUaMethod_Unhold (FB / 1.2.0)

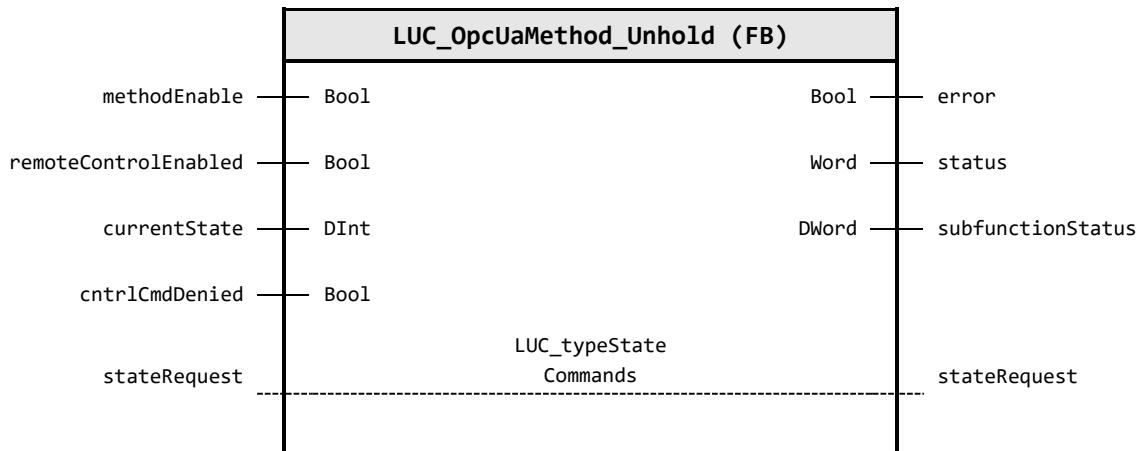
Author: Siemens Digital Industries

Short description

This Block represents one OPC UA Method interface to send a defined command to the *LUC_InterfaceManager* instance. The Method is called in the method collector *LUC_RemoteCtrl_OpcUa*, where all other available OPC UA methods are called as well. Therefore, there is no need to call this single method.

Interface description

Block interface



Input parameter

Identifier	Data type	Default value	Description
methodEnable	Bool	FALSE	Method is enabled
remoteControl Enabled	Bool	FALSE	Remote control is active
currentState	DInt	LPMLV2022_ STATE_STOP PED	Current executed state
cntrlCmdDenie d	Bool	FALSE	Control command denied (output is reset with the next successful CntrlCmd or if input 'CntrlCmd' is set to 0 or if input 'CmdChangeRequest' is set to FALSE)

Output parameter

Identifier	Data type	Description
error	Bool	Error occurred
status	Word	Status identifier
subfunctionSta tus	DWord	Status of called subfunction when error occurs

In/Out parameter

Identifier	Data type	Description
stateRequest	LUC_typeStat eCommands	Commands to request a state change

Status & Error codes

Code / Value	Identifier / Description
16#0000	STATUS_EXECUTION_FINISHED Execution finished without errors
16#7000	STATUS_NO_CALL No call of FB
16#8600	ERR_METHOD_PRE Error in block OPC_UA_ServerMethodPre
16#8601	ERR_METHOD_POST Error in block OPC_UA_ServerMethodPost

User defined datatype(s)

LUC_typeStateCommands (UDT / V1.1.0)

Unit State Commands

Identifier	Data type	Default value	Description
reset	Bool	FALSE	OMAC PackMLV2022 control command Reset
start	Bool	FALSE	OMAC PackMLV2022 control command Start
stop	Bool	FALSE	OMAC PackMLV2022 control command Stop
hold	Bool	FALSE	OMAC PackMLV2022 control command Hold
unhold	Bool	FALSE	OMAC PackMLV2022 control command Unhold
suspend	Bool	FALSE	OMAC PackMLV2022 control command Suspend
unsuspend	Bool	FALSE	OMAC PackMLV2022 control command Unsuspend
abort	Bool	FALSE	OMAC PackMLV2022 control command Abort
clear	Bool	FALSE	OMAC PackMLV2022 control command Clear
complete	Bool	FALSE	OMAC PackMLV2022 control command Complete
stateComplete	Bool	FALSE	OMAC PackMLV2022 control command State Complete

Change log

Version & Date	Change description
1.0.0 15.12.2023	SIMATIC Systems Support First released version
1.1.0 21.06.2024	SIMATIC Systems Support Included LPMLV2022 constants as a local constants in the block
1.2.0 05.09.2024	SIMATIC Systems Support Removed the forcing of the ENO to true.

3.7.13 LUC_OpcUaMethod_Unsuspend (FB / 1.2.0)

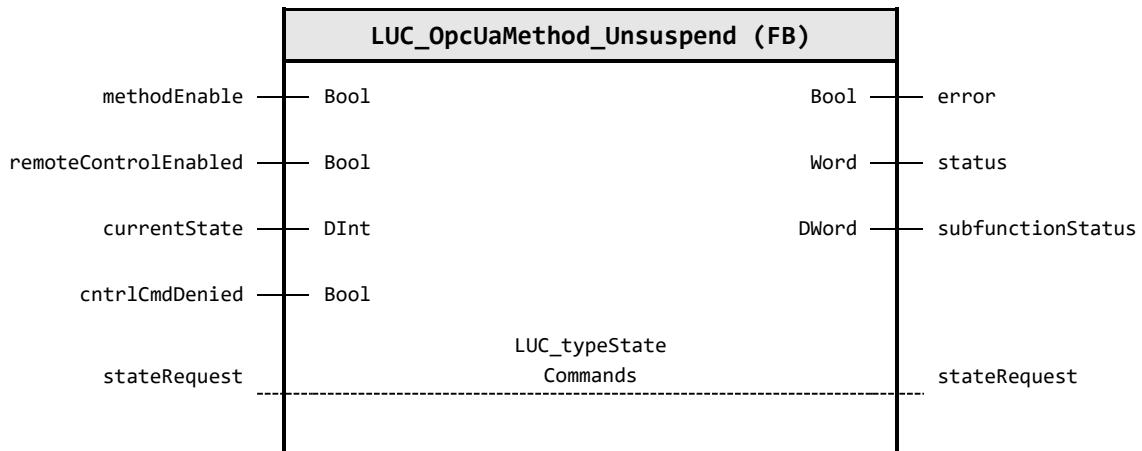
Author: Siemens Digital Industries

Short description

This Block represents one OPC UA Method interface to send a defined command to the *LUC_InterfaceManager* instance. The Method is called in the method collector *LUC_RemoteCtrl_OpcUa*, where all other available OPC UA methods are called as well. Therefore, there is no need to call this single method.

Interface description

Block interface



Input parameter

Identifier	Data type	Default value	Description
methodEnable	Bool	FALSE	Method is enabled
remoteControl Enabled	Bool	FALSE	Remote control is active
currentState	DInt	LPMLV2022_ STATE_STOP PED	Current executed state
cntrlCmdDenie d	Bool	FALSE	Control command denied (output is reset with the next successful CntrlCmd or if input 'CntrlCmd' is set to 0 or if input 'CmdChangeRequest' is set to FALSE)

Output parameter

Identifier	Data type	Description
error	Bool	Error occurred
status	Word	Status identifier
subfunctionSta tus	DWord	Status of called subfunction when error occurs

In/Out parameter

Identifier	Data type	Description
stateRequest	LUC_typeStat eCommands	Commands to request a state change

Status & Error codes

Code / Value	Identifier / Description
16#0000	STATUS_EXECUTION_FINISHED Execution finished without errors
16#7000	STATUS_NO_CALL No call of FB
16#8600	ERR_METHOD_PRE Error in block OPC_UA_ServerMethodPre
16#8601	ERR_METHOD_POST Error in block OPC_UA_ServerMethodPost

User defined datatype(s)

LUC_typeStateCommands (UDT / V1.1.0)

Unit State Commands

Identifier	Data type	Default value	Description
reset	Bool	FALSE	OMAC PackMLV2022 control command Reset
start	Bool	FALSE	OMAC PackMLV2022 control command Start
stop	Bool	FALSE	OMAC PackMLV2022 control command Stop
hold	Bool	FALSE	OMAC PackMLV2022 control command Hold
unhold	Bool	FALSE	OMAC PackMLV2022 control command Unhold
suspend	Bool	FALSE	OMAC PackMLV2022 control command Suspend
unsuspend	Bool	FALSE	OMAC PackMLV2022 control command Unsuspend
abort	Bool	FALSE	OMAC PackMLV2022 control command Abort
clear	Bool	FALSE	OMAC PackMLV2022 control command Clear
complete	Bool	FALSE	OMAC PackMLV2022 control command Complete
stateComplete	Bool	FALSE	OMAC PackMLV2022 control command State Complete

Change log

Version & Date	Change description
1.0.0 15.12.2023	SIMATIC Systems Support First released version
1.1.0 21.06.2024	SIMATIC Systems Support Included LPMLV2022 constants as a local constants in the block
1.2.0 05.09.2024	SIMATIC Systems Support Removed the forcing of the ENO to true.

3.8 SignalStack

3.8.1 LUC_SignalStack (FB / 1.2.0)

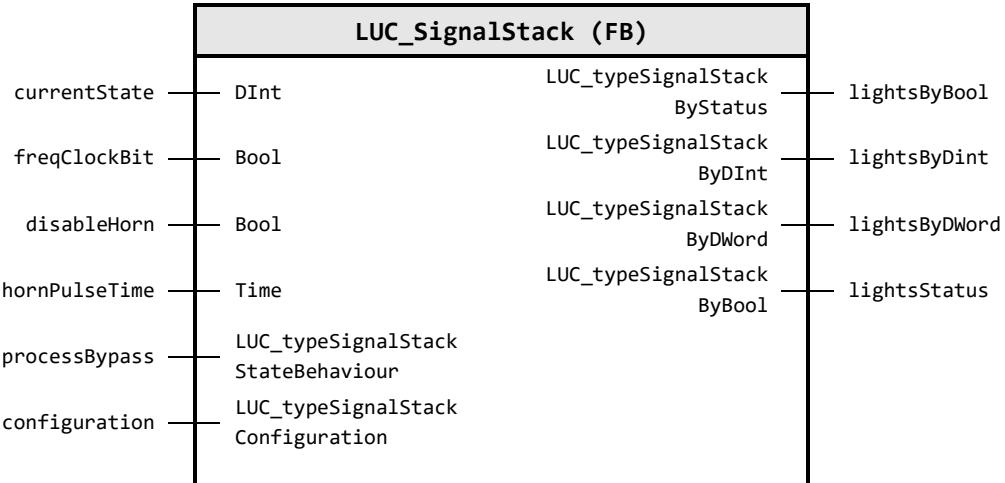
Author: Siemens Digital Industries

Short description

This block represents a Signal Stack object, to set the signal stack outputs according to the current global Unit state from the [LUC_ModeStateManager](#).

Interface description

Block interface



Input parameter

Identifier	Data type	Default value	Description
currentState	DInt	0	Current state of the unit
freqClockBit	Bool	FALSE	Signal with specified frequency for any light of the stack
disableHorn	Bool	FALSE	Disable Horn
hornPulseTime	Time	T#0s	Horn Pulse time - the Horn is turned on as long the pulse time is not expired. If the time is set to t#0s, continuous horn sound can be achieved.
processBypass	LUC_typeSignalStackStateBehaviour	---	Bypass for directly setting the signals in the SignalStack, concatenate the configuration with OR
configuration	LUC_typeSignalStackConfig	---	Behaviour settings for the signal stack - machine modes

Output parameter

Identifier	Data type	Description
lightsByBool	LUC_typeSignalStackByStatus	Stack Light Signals by Bool
lightsByDint	LUC_typeSignalStackByDInt	Stack Light Signals by DINT
lightsByDWord	LUC_typeSignalStackByDWord	Stack Light Signals by DWord

3 Program blocks

Identifier	Data type	Description
lightsStatus	LUC_typeSign aiStackByBool	Signal stack type used for a default machine signal stack, includes.

User defined datatype(s)

LUC_typeSignalStackStateBehaviour (UDT / V1.1.0)

Behavior settings for the signal stack - belonging to machine states

Identifier	Data type	Default value	Description
static	LUC_typeSign aiStackByBool	---	Signal stack type used for a default machine signal stack, includes.
flashing	LUC_typeSign aiStackByBool	---	Signal stack type used for a default machine signal stack, includes.

LUC_typeSignalStackConfiguration (UDT / V1.1.0)

Behaviour settings for the signal stack - machine modes

Identifier	Data type	Default value	Description
undefined	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Undefined"
clearing	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Clearing"
stopped	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Stopped"
starting	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Starting"
idle	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Idle"
suspended	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Suspended"
execute	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Execute"
stopping	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Stopping"
aborting	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Aborting"
aborted	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Aborted"
holding	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Holding"
held	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Held"
unholding	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Unholding"
suspending	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Suspending"
unsuspending	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Unsuspending"
resetting	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Resetting"
completing	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Completing"
completed	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Completed"

LUC_typeSignalStackByStatus (UDT / V1.1.0)

Stack Light Signals by Bool

Identifier	Data type	Default value	Description
redSolid	Bool	FALSE	Red light solid
redFlashing	Bool	FALSE	Red light flashing
yellowSolid	Bool	FALSE	Yellow light solid
yellowFlashing	Bool	FALSE	Yellow light flashing
blueSolid	Bool	FALSE	Blue light solid
blueFlashing	Bool	FALSE	Blue light flashing
greenSolid	Bool	FALSE	Green light solid
greenFlashing	Bool	FALSE	Green light flashing
hornSolid	Bool	FALSE	Horn solid
hornFlashing	Bool	FALSE	Horn flashing
whiteSolid	Bool	FALSE	White light solid
whiteFlashing	Bool	FALSE	White light flashing
user1Solid	Bool	FALSE	User defined light solid
user1Flashing	Bool	FALSE	User defined light flashing
user2Solid	Bool	FALSE	User defined light solid
user2Flashing	Bool	FALSE	User defined light flashing

LUC_typeSignalStackByDInt (UDT / V1.1.0)

Stack Light Signals by DINT

Identifier	Data type	Default value	Description
red	DInt	0	Red light active
yellow	DInt	0	Yellow light active
blue	DInt	0	Blue light active
green	DInt	0	Green light active
horn	DInt	0	Horn active
white	DInt	0	White light active
user1	DInt	0	User defined light 1 active
user2	DInt	0	User defined light 2 active

LUC_typeSignalStackByDWord (UDT / V1.1.0)

Stack Light Signals by DWord

Identifier	Data type	Default value	Description
statusByDWor d	DWord	16#00000000	Stack light status. Indicates the current active lights and their behaviour.

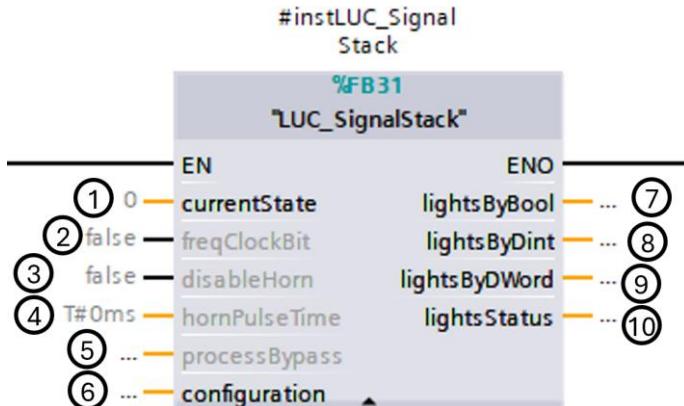
LUC_typeSignalStackByBool (UDT / V1.1.0)

Signal stack type used for a default machine signal stack, includes.

Identifier	Data type	Default value	Description
red	Bool	FALSE	Signal light active- Red
yellow	Bool	FALSE	Signal light active- Yellow
blue	Bool	FALSE	Signal light active- Blue
green	Bool	FALSE	Signal light active - Green
horn	Bool	FALSE	Signal horn - The horn will sound with the configured horn pulse time
white	Bool	FALSE	Signal light active- White
user1	Bool	FALSE	Signal active- Userdefined 1
user2	Bool	FALSE	Signal active- Userdefined 2

Functional description**How to use the LUC_SignalStack block in your application**

To use the block, you must choose the type of block instantiation fitting to your application. In most applications, the block will be called as a multi-instance. Once instantiated, inputs, inOuts and outputs should be connected, so that the block can interact with the rest of your automation solution. In the following picture, an example of the block call is shown.



The picture shows the following variables of the interface:

1. The *current state* of the unit.
2. The *freqClockBit* input is a boolean signal with a user defined frequency to make a Stack Light or Horn flash.
3. The *hornPulseTime* of the Stack Light.
4. The *hornPulseTime* input is the duration of the pulse of the horn. If time is T#0s, the horn would have a continuous sound.
5. The *processBypass* input. It allows you to bypass the block configuration and set the lights and horn statically/flashing ON and OFF regardless of the production state of the unit.
6. The configuration input. It provides the functionality of selecting which light/horn is going to be ON/OFF or FLASHING for each state of the unit. Typically, a global DB for configuration is connected here. More details about the configuration possibilities are described in the section below (Structure of LUC_StackLight configuration).
7. The horn/light outputs. They provide an unstructured way of connecting the signal stack periphery to the block. The output logic is boolean and monitors the different status that each light or horn could be in. For a structured way of connection, by means of a UDT, use object number 6, described below.

8. The horn/light outputs. They provide an unstructured way of connecting the signal stack periphery to the block. The output logic is DIInt and monitors the different status that each light or horn could be in.
For a structured way of connection, by means of a UDT, use object number 6, described below.

LightsByDint	Value = 1	Value = 2	Value = 3
Red	Off	On	Flashing
Yellow	Off	On	Flashing
Blue	Off	On	Flashing
Blue	Off	On	Flashing
Green	Off	On	Flashing
Horn	Off	On	Flashing
White	Off	On	Flashing
User 1	Off	On	Flashing
User 2	Off	On	Flashing

9. The horn/light outputs. They provide an unstructured way of connecting the signal stack periphery to the block. The output logic is a DWord and monitors the different status that each light or horn could be in. Each bit refers to a different status of the lights. For a structured way of connection, by means of a UDT, use object number 6, described below.

Bit Position	Meaning
0	Red solid
1	Red flashing
2	Yellow solid
3	Yellow flashing
4	Blue solid
5	Blue flashing
6	Green solid
7	Green flashing
8	Horn solid
9	Horn flashing
10	White solid
11	White flashing
12..31	User Defined

10. The stackConnectPeriphery output. It provides a structure to connect the lighstack and horn periphery. The output logic is boolean.

'Structure of LUC_StackLight configuration'

In the following picture, a visual representation of the configuration structure is shown.

LUC_typeSignalStackConfiguration		
	Name	Data type
▶ undefined	"LUC_typeSignalStackStateBehaviour"	
▶ clearing	"LUC_typeSignalStackStateBehaviour"	
▶ stopped	"LUC_typeSignalStackStateBehaviour"	
▶ starting	"LUC_typeSignalStackStateBehaviour"	
▶ idle	"LUC_typeSignalStackStateBehaviour"	
▶ suspended	"LUC_typeSignalStackStateBehaviour"	
▶ execute	"LUC_typeSignalStackStateBehaviour"	
▶ stopping	"LUC_typeSignalStackStateBehaviour"	
▶ aborting	"LUC_typeSignalStackStateBehaviour"	
▶ aborted	"LUC_typeSignalStackStateBehaviour"	
▶ holding	"LUC_typeSignalStackStateBehaviour"	
▶ held	"LUC_typeSignalStackStateBehaviour"	
▶ unholding	"LUC_typeSignalStackStateBehaviour"	
▶ suspending	"LUC_typeSignalStackStateBehaviour"	
▶ unsuspending	"LUC_typeSignalStackStateBehaviour"	
▶ resetting	"LUC_typeSignalStackStateBehaviour"	
▶ completing	"LUC_typeSignalStackStateBehaviour"	
▶ completed	"LUC_typeSignalStackStateBehaviour"	

According to each state of OMAC model, the user can configure the light and horn behavior. They can be configured as OFF, ON, FLASHING. If the user needs also a different configuration for the different Unit production modes (Production, Maintenance, Manual, etc.) use the LUC_SignalStackExtended block.

3 Program blocks

LUC_typeSignalStackConfiguration				
	Name	Data type	Default value	Acc
►	undefined	"LUC_typeSignalStackStateBehaviour"		
►	clearing	"LUC_typeSignalStackStateBehaviour"		
►	static	"LUC_typeSignalStackByBool"		
►	red	Bool	false	
►	yellow	Bool	false	
►	blue	Bool	false	
►	green	Bool	false	
►	horn	Bool	false	
►	white	Bool	false	
►	user1	Bool	false	
►	user2	Bool	false	
►	flashing	"LUC_typeSignalStackByBool"		
►	red	Bool	TRUE	
►	yellow	Bool	false	
►	blue	Bool	false	
►	green	Bool	false	
►	horn	Bool	false	
►	white	Bool	false	
►	user1	Bool	false	
►	user2	Bool	false	
►	stopped	"LUC_typeSignalStackStateBehaviour"		
►	starting	"LUC_typeSignalStackStateBehaviour"		
►	idle	"LUC_typeSignalStackStateBehaviour"		
►	suspended	"LUC_typeSignalStackStateBehaviour"		
►	execute	"LUC_typeSignalStackStateBehaviour"		
►	stopping	"LUC_typeSignalStackStateBehaviour"		
►	aborting	"LUC_typeSignalStackStateBehaviour"		
►	aborted	"LUC_typeSignalStackStateBehaviour"		
►	holding	"LUC_typeSignalStackStateBehaviour"		
►	held	"LUC_typeSignalStackStateBehaviour"		
►	unholding	"LUC_typeSignalStackStateBehaviour"		
►	suspending	"LUC_typeSignalStackStateBehaviour"		

'Connection with HMI interface'

See chapter 6.5. Signal stack.

Change log

Version & Date	Change description
1.0.0 15.12.2023	SIMATIC Systems Support First released version
1.1.0 21.06.2024	SIMATIC Systems Support Implemented 4 different ways to monitor Stack lights and horn (lightsByBool, lightsByDint, lightsByDWord and lightsStatus)
1.2.0 05.09.2024	SIMATIC Systems Support Optimized code.

3.9 StateMachineConfiguration

3.9.1 LUC_StateMachineConfig (FC / V1.2.0)

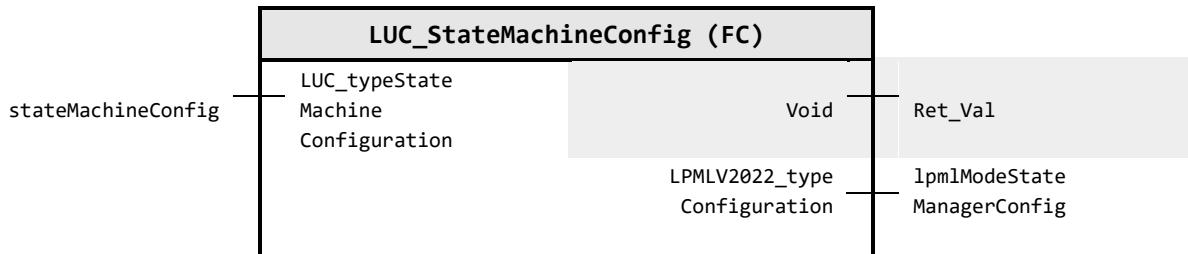
Author: Siemens Digital Industries

Short description

This block transforms the boolean configuration done in the DB Configuration of the disabled states and mode transitions into a standard structure. The standard user defined type *LPMLV2022_typeConfiguration* will be used in the *LPML_ModeStateManager*.

Interface description

Block interface



Input parameter

Identifier	Data type	Description
stateMachineConfig	LUC_typeStateMachineConfiguration	Configuration of the state machine in a boolean structure

Output parameter

Identifier	Data type	Description
Ret_Val	Void	Void - Function has no return value
lpmlModeStateManagerConfig	LPMLV2022_typeConfiguration	Configuration of the state machine in a DWord structure

User defined datatype(s)

LUC_typeStateMachineConfiguration (UDT / V1.2.0)

Configuration of the state machine

Identifier	Data type	Default value	Description
enableModes	LUC_typeModeConfiguration	---	Configure which modes are enabled
enableStates	Array[0..LPML_V2022_MAX_MODES_UPPER_LIM] of LUC_typeStateConfiguration	---	Configure which states are enabled for each mode

3 Program blocks

Identifier	Data type	Default value	Description
enableModeTransitions	Array[0..LPML_V2022_MAX_MODES_UPPER_LIM] of LUC_typeModeTransitionConfiguration	---	Configure in which states a mode transition is allowed
enableHoldCommand	LUC_typeHoldCommandConfiguration	---	Configure in which states a hold command is possible
enableCompleteCommand	LUC_typeCompleteCommandConfiguration	---	Configure in which states a complete command is possible

Functional description

To obtain the *stateModeManagerConfiguration* is mandatory to create a **Configuration DB** where the states and transitions of each mode of the model state should be be configured.

- Modes. The user can define to which modes is permitted to do the mode request change from each mode. The structure to define them is by boolean variables.
- States. The definition of the states of each mode is configured by enabling the states that won't be used per each mode. The structure to define them is by boolean variables.
- Transitions. The definition of the transitions from which state is allowed to change the mode of a Unit. The structure to define them is by boolean variables.
- Hold. The definition of the states from which the Unit will be able to change into *Hold* state. The structure to define them is by boolean variables.
- Complete. The definition of the states from which the Unit will be able to change into *Complete* state. The structure to define them is by boolean variables.

Configuration									
Name	Data type	Start value	Ret...	Acces...	Writ...	Visibl...	Set...	Supervision	Comment
Static									
stateConfiguration	"LUC_typeStateMachineConfiguration"								Configuration of states disabled for each mode
enableModes	"LUC_typeModeConfiguration"								Configure which modes are enabled
enableStates	Array[0..LPML_V2022_MAX_MODES_UPPER_LIM]								Configure which states are enabled for each mode
enableModeTransitions	Array[0..LPML_V2022_MAX_MODES_UPPER_LIM]								Configure in which states a mode transition is allowed
enableHoldCommand	"LUC_typeHoldCommandConfiguration"								Configure in which states a hold command is possible
enableCompleteCommand	"LUC_typeCompleteCommandConfigu...								Configure in which states a complete command is possible

This block will transform the boolean configuration data into a user data type provided by the LPML library *LPMLV2022_typeConfiguration* :

Configuration									
Name	Data type	Start value	Ret...	Acces...	Writ...	Visibl...	Set...	Supervision	Comment
Static									
stateConfiguration	"LUC_typeStateMachineConfiguration"								Configuration of states disabled for each mode
stateModelManager	"LPMLV2022_typeConfiguration"								Support: tech.team.motioncontrol@siemens.com
EnabledModesCfg	DWord	16#0000_01FE							Bit locations within the DWord represent Mode numbers. A val...
DisabledStatesCfg	Array[0..LPML_V2022_MAX_MODES_UPPER_LIM]								The array index represents the Mode number. Bit locations wit...
ModeTransitionCfg	Array[0..LPML_V2022_MAX_MODES_UPPER_LIM]								The array index represents the Mode number. Bit locations wit...
holdCmdCfg	DWord	16#0000_0060							Bit locations within the DWord value represent state numbers...
completeCmdCfg	DWord	16#0000_0860							Bit locations within the DWord value represent State numbers...

Change log

Version & Date	Change description
1.0.0 15.12.2023	SIMATIC Systems Support First released version
1.1.0 21.06.2024	SIMATIC Systems Support Implemented state model configuration for all possible modes
1.2.0 05.09.2024	SIMATIC Systems Support Optimized code.

4 PLC data types

4.1 LUC_typeCommands (UDT / V1.1.0)

Description

Module related commands from external systems

Parameter description

Identifier	Data type	Default value	Description
modeRequests	LUC_typeModeRequests	---	Commands to request a mode change
stateCommands	LUC_typeStateCommands	---	Commands to request a state change

4.2 LUC_typeDiagnostics (UDT / V1.1.0)

Description

Module related diagnostic information

Parameter description

Identifier	Data type	Default value	Description
interfaceManagerStatus	Word	16#0000	Status of the FB InterfaceManager
modeStateManagerDiagnostics	LPMLV2022_TypeDiagnostics	---	Diagnostics of the FB UnitModeStateManager

4.3 LUC_typeEmFeedback (UDT / V1.1.0)

Description

Parameter description

Identifier	Data type	Default value	Description
isLinked	Bool	true	EM is linked to the parent (usually the unit)
commands	LUC_typeStateCommands	---	Unit State Commands

4.4 LUC_typeInterlinkMode (UDT / V1.2.0)

Description

Link and unlink from parent module

Parameter description

Identifier	Data type	Default value	Description
link	Bool	FALSE	Command to link module to parent
unlink	Bool	FALSE	Command to unlink module from parent
isLinked	Bool	true	Equipment module is linked to parent

4.5 LUC_typeManagerInterface (UDT / V1.2.0)

Description

Module interface for external systems

Parameter description

Identifier	Data type	Default value	Description
commands	LUC_typeCommands	---	Commands to control the unit state model
monitoring	LUC_typeMonitoring	---	Structure to monitor the unit state model
diagnostics	LUC_typeDiagnostics	---	Status of the unit state model

4.6 LUC_typeModeRequests (UDT / V1.1.0)

Description

Unit Mode requests

Parameter description

Identifier	Data type	Default value	Description
production	Bool	false	Request to change to "Production"
maintenance	Bool	false	Request to change to "Maintenance"
manual	Bool	false	Request to change to "Manual"
userMode01	Bool	false	Request to change to "UserMode01"
userMode02	Bool	false	Request to change to "UserMode02"
userMode03	Bool	false	Request to change to "UserMode03"
userMode04	Bool	false	Request to change to "UserMode04"
userMode05	Bool	false	Request to change to "UserMode05"
userMode06	Bool	false	Request to change to "UserMode06"
userMode07	Bool	false	Request to change to "UserMode07"
userMode08	Bool	false	Request to change to "UserMode08"
userMode09	Bool	false	Request to change to "UserMode09"
userMode10	Bool	false	Request to change to "UserMode10"
userMode11	Bool	false	Request to change to "UserMode11"
userMode12	Bool	false	Request to change to "UserMode12"
userMode13	Bool	false	Request to change to "UserMode13"

4.7 LUC_typeModes (UDT / V1.1.0)

Description

Parameter description

Identifier	Data type	Default value	Description
current	DInt	0	Current operating mode
production	Bool	FALSE	The operating mode "Production" is active
maintenance	Bool	FALSE	The operating mode "Maintenance" is active
manual	Bool	FALSE	The operating mode "Manual" is active

Identifier	Data type	Default value	Description
userMode01	Bool	FALSE	The operating mode "UserMode1" is active
userMode02	Bool	FALSE	The operating mode "UserMode2" is active
userMode03	Bool	FALSE	The operating mode "UserMode3" is active
userMode04	Bool	FALSE	The operating mode "UserMode4" is active
userMode05	Bool	FALSE	The operating mode "UserMode5" is active
userMode06	Bool	FALSE	The operating mode "UserMode6" is active
userMode07	Bool	FALSE	The operating mode "UserMode7" is active
userMode08	Bool	FALSE	The operating mode "UserMode8" is active
userMode09	Bool	FALSE	The operating mode "UserMode9" is active
userMode10	Bool	FALSE	The operating mode "UserMode10" is active
userMode11	Bool	FALSE	The operating mode "UserMode11" is active
userMode12	Bool	FALSE	The operating mode "UserMode12" is active
userMode13	Bool	FALSE	The operating mode "UserMode13" is active

4.8 LUC_typeMonitoring (UDT / V1.2.0)

Description

Module related monitoring information

Parameter description

Identifier	Data type	Default value	Description
unitModeCurrent	DInt	0	Current unit mode
unitModeRequested	Bool	FALSE	Requested unit mode change
unitModeChangeInProcess	Bool	FALSE	Unit mode change in process
stateCurrent	DInt	0	Current state
stateRequested	DInt	0	Requested state
stateChangeInProcess	Bool	FALSE	State change in process
curDisabledStates	DWord	16#00000000	Disabled states in current unit mode
curHoldCmdCfg	DWord	16#00000000	Bit locations within the DWORD represent State numbers. A value of 1 in a bit location indicates that the Hold control command is taken into account in the corresponding state
curCompleteCmdCfg	DWord	16#00000000	Bit locations within the DWORD represent State numbers. A value of 1 in a bit location indicates that the Complete control command is taken into account in the corresponding state
curEnabledModes	DWord	16#00000000	Bit locations within the DWORD represent Mode numbers. A value of 1 in a bit location [0-31] indicates that the corresponding Mode number is enabled. Mode 0 is an Invalid Mode, therefore bit 0 is unused
allowedUnitModes	DWord	16#00000000	Bit locations within the DWORD represent Mode numbers. A value of 1 in a bit location [0-31] indicates that a change to the corresponding Mode number is currently possible. Mode 0 is an Invalid Mode, therefore bit 0 is unused
cntrlCmdNotAllowed	Bool	FALSE	Control command is not allowed (output is reset with the next successful CntrlCmd or if input 'CntrlCmd' is set to 0 or if input 'CmdChangeRequest' is set to FALSE)

Identifier	Data type	Default value	Description
localControlEnabled	Bool	FALSE	Local control of the unit mode state manager is enabled
hmiControlEnabled	Bool	FALSE	Hmi control of the unit mode state manager is enabled
remoteControlEnabled	Bool	FALSE	Remote control of the unit mode state manager is enabled

4.9 LUC_typeStateCommands (UDT / V1.1.0)

Description

Unit State Commands

Parameter description

Identifier	Data type	Default value	Description
reset	Bool	FALSE	OMAC PackMLV2022 control command Reset
start	Bool	FALSE	OMAC PackMLV2022 control command Start
stop	Bool	FALSE	OMAC PackMLV2022 control command Stop
hold	Bool	FALSE	OMAC PackMLV2022 control command Hold
unhold	Bool	FALSE	OMAC PackMLV2022 control command Unhold
suspend	Bool	FALSE	OMAC PackMLV2022 control command Suspend
unsuspend	Bool	FALSE	OMAC PackMLV2022 control command Unsuspend
abort	Bool	FALSE	OMAC PackMLV2022 control command Abort
clear	Bool	FALSE	OMAC PackMLV2022 control command Clear
complete	Bool	FALSE	OMAC PackMLV2022 control command Complete
stateComplete	Bool	FALSE	OMAC PackMLV2022 control command State Complete

4.10 LUC_typeStates (UDT / V1.1.0)

Description

Parameter description

Identifier	Data type	Default value	Description
current	DInt	0	Current unit state
clearing	Bool	FALSE	The unit state "Clearing" is active
stopped	Bool	FALSE	The unit state "Stopped" is active
starting	Bool	FALSE	The unit state "Starting" is active
idle	Bool	FALSE	The unit state "Idle" is active
suspended	Bool	FALSE	The unit state "Suspended" is active
execute	Bool	FALSE	The unit state "Execute" is active
stopping	Bool	FALSE	The unit state "Stopping" is active
aborting	Bool	FALSE	The unit state "Aborting" is active
aborted	Bool	FALSE	The unit state "Aborted" is active
holding	Bool	FALSE	The unit state "Holding" is active
held	Bool	FALSE	The unit state "Held" is active
unholding	Bool	FALSE	The unit state "Unholding" is active
suspending	Bool	FALSE	The unit state "Suspending" is active
unsuspending	Bool	FALSE	The unit state "Unsuspending" is active

Identifier	Data type	Default value	Description
resetting	Bool	FALSE	The unit state "Resetting" is active
completing	Bool	FALSE	The unit state "Completing" is active
complete	Bool	FALSE	The unit state "Complete" is active

4.11 LUC_typeUnitModeStateTimes (UDT / V1.2.0)

Description

Statistic of time spent in modes and states

Parameter description

Identifier	Data type	Default value	Description
modeTimeCurrent	DInt	0	Actual time for current mode in seconds
stateTimeCurrent	DInt	0	Actual time for current state in seconds
accTimeSinceReset	DInt	0	Accumulated time since last reset in seconds
actualTimeAllModes	Array[0..LPMLV2022_MODES_UPPER_LIM] of DInt	---	Actual time for all modes in seconds (former name: ModeCurrentTime)
cumulativeTimeAllModes	Array[0..LPMLV2022_MODES_UPPER_LIM] of DInt	---	Cumulative time for all modes in seconds (former name: ModeCumulativeTime)
actualTimeAllStatesOfAllModes	Array[0..LPMLV2022_MODES_UPPER_LIM, 0..LPMLV2022_STATES_UPPER_PER_LIM] of DInt	---	Actual time for all states in seconds (former name: StateCurrentTime)
cumulativeTimeAllStatesOfAllModes	Array[0..LPMLV2022_MODES_UPPER_LIM, 0..LPMLV2022_STATES_UPPER_PER_LIM] of DInt	---	Cumulative time for all states in seconds (former name: StateCumulativeTime)
actualTimeAllStatesOfCurrentMode	Array[0..LPMLV2022_STATES_UPPER_LIM] of DInt	---	Actual time for all states in current mode in seconds
cumulativeTimeAllStatesOfCurrentMode	Array[0..LPMLV2022_STATES_UPPER_LIM] of DInt	---	Cumulative time for all states in current mode in seconds
cumulativeTimeCurrentMode	DInt	0	Cumulative time for current mode in seconds

4.12 HelperFunctions

4.12.1 LUC_typeButtons (UDT / V1.2.0)

Description

Properties of HMI buttons configured from the PLC

Parameter description

Identifier	Data type	Default value	Description
visibility	Word	16#0000	Sets the visibility property in the buttons of the HMI
operatorControlAllowed	Word	16#0000	Sets the operator control allowed property in the buttons of the HMI
active	Word	16#0000	Sets the alternative border color property in the buttons of the HMI

4.13 History

4.13.1 LUC_typeModeState (UDT / V1.2.0)

Description

Duration of stay in one mode and state

Parameter description

Identifier	Data type	Default value	Description
mode	DInt	0	OMAC identifier of mode
state	DInt	0	OMAC identifier of state
duration	LReal	0.0	Time stayed in that status

4.14 RemoteControl

4.14.1 LUC_typeRemoteCtrlConfiguration (UDT / V1.1.0)

Description

OPC UA Method calls parameters, to parametrize the functionality of the OPC UA Method calls belonging to the UnitModeStatemanager to enable / disable the method calls

Parameter description

Identifier	Data type	Default value	Description
enableMethods	Struct	---	Enabling of methods
reset	Bool	TRUE	Enable command reset
start	Bool	TRUE	Enable command start
stop	Bool	TRUE	Enable command stop
hold	Bool	TRUE	Enable command hold
unhold	Bool	TRUE	Enable command unhold
suspend	Bool	TRUE	Enable command suspend
unsuspend	Bool	TRUE	Enable command unsuspend
abort	Bool	TRUE	Enable command abort
clear	Bool	TRUE	Enable command clear
complete	Bool	TRUE	Enable command complet
stateComplete	Bool	TRUE	Enable command state completed
requestUnitMode	Bool	TRUE	Enable command request unit mode change
requestUnitState	Bool	TRUE	Enable command request unit state change

4.15 SignalStack

4.15.1 LUC_typeSignalStackByBool (UDT / V1.1.0)

Description

Signal stack type used for a default machine signal stack, includes.

Parameter description

Identifier	Data type	Default value	Description
red	Bool	FALSE	Signal light active- Red
yellow	Bool	FALSE	Signal light active- Yellow
blue	Bool	FALSE	Signal light active- Blue
green	Bool	FALSE	Signal light active - Green
horn	Bool	FALSE	Signal horn - The horn will sound with the configured horn pulse time
white	Bool	FALSE	Signal light active- White
user1	Bool	FALSE	Signal active- Userdefined 1
user2	Bool	FALSE	Signal active- Userdefined 2

4.15.2 LUC_typeSignalStackByDInt (UDT / V1.1.0)

Description

Stack Light Signals by DINT

Parameter description

Identifier	Data type	Default value	Description
red	DInt	0	Red light active
yellow	DInt	0	Yellow light active
blue	DInt	0	Blue light active
green	DInt	0	Green light active
horn	DInt	0	Horn active
white	DInt	0	White light active
user1	DInt	0	User defined light 1 active
user2	DInt	0	User defined light 2 active

4.15.3 LUC_typeSignalStackByDWord (UDT / V1.1.0)

Description

Stack Light Signals by DWord

Parameter description

Identifier	Data type	Default value	Description
statusByDWor d	DWord	16#00000000	Stack light status. Indicates the current active lights and their behaviour.

4.15.4 LUC_typeSignalStackByStatus (UDT / V1.1.0)

Description

Stack Light Signals by Bool

Parameter description

Identifier	Data type	Default value	Description
redSolid	Bool	FALSE	Red light solid
redFlashing	Bool	FALSE	Red light flashing
yellowSolid	Bool	FALSE	Yellow light solid
yellowFlashing	Bool	FALSE	Yellow light flashing
blueSolid	Bool	FALSE	Blue light solid
blueFlashing	Bool	FALSE	Blue light flashing
greenSolid	Bool	FALSE	Green light solid
greenFlashing	Bool	FALSE	Green light flashing
hornSolid	Bool	FALSE	Horn solid
hornFlashing	Bool	FALSE	Horn flashing
whiteSolid	Bool	FALSE	White light solid
whiteFlashing	Bool	FALSE	White light flashing
user1Solid	Bool	FALSE	User defined light solid
user1Flashing	Bool	FALSE	User defined light flashing
user2Solid	Bool	FALSE	User defined light solid
user2Flashing	Bool	FALSE	User defined light flashing

4.15.5 LUC_typeSignalStackConfiguration (UDT / V1.1.0)

Description

Bahviour settings for the signal stack - machine modes

Parameter description

Identifier	Data type	Default value	Description
undefined	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Undefined"
clearing	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Clearing"
stopped	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Stopped"
starting	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Starting"
idle	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Idle"
suspended	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Suspended"
execute	LUC_typeSign alStackStateB ehaviour	---	Stack light configuration for state "Execute"

Identifier	Data type	Default value	Description
stopping	LUC_typeSignal alStackStateBehaviour	---	Stack light configuration for state "Stopping"
aborting	LUC_typeSignal alStackStateBehaviour	---	Stack light configuration for state "Aborting"
aborted	LUC_typeSignal alStackStateBehaviour	---	Stack light configuration for state "Aborted"
holding	LUC_typeSignal alStackStateBehaviour	---	Stack light configuration for state "Holding"
held	LUC_typeSignal alStackStateBehaviour	---	Stack light configuration for state "Held"
unholding	LUC_typeSignal alStackStateBehaviour	---	Stack light configuration for state "Unholding"
suspending	LUC_typeSignal alStackStateBehaviour	---	Stack light configuration for state "Suspending"
unsuspending	LUC_typeSignal alStackStateBehaviour	---	Stack light configuration for state "Unsuspending"
resetting	LUC_typeSignal alStackStateBehaviour	---	Stack light configuration for state "Resetting"
completing	LUC_typeSignal alStackStateBehaviour	---	Stack light configuration for state "Completing"
completed	LUC_typeSignal alStackStateBehaviour	---	Stack light configuration for state "Completed"

4.15.6 LUC_typeSignalStackStateBehaviour (UDT / V1.1.0)

Description

Behavior settings for the signal stack - belonging to machine states

Parameter description

Identifier	Data type	Default value	Description
static	LUC_typeSignal alStackByBool	---	Signal stack type used for a default machine signal stack, includes.
flashing	LUC_typeSignal alStackByBool	---	Signal stack type used for a default machine signal stack, includes.

4.16 StateMachineConfiguration

4.16.1 LUC_typeCompleteCommandConfiguration (UDT / V1.1.0)

Description

Configure in which states a complete command is possible

Parameter description

Identifier	Data type	Default value	Description
suspended	Bool	true	The Complete control command is taken into account in state Suspended
execute	Bool	true	The Complete control command is taken into account in state Execute
held	Bool	true	The Complete control command is taken into account in state Held

4.16.2 LUC_typeHoldCommandConfiguration (UDT / V1.1.0)

Description

Configure in which states a hold command is possible

Parameter description

Identifier	Data type	Default value	Description
idle	Bool	FALSE	Transition from "Idle" to "Held" is possible
starting	Bool	FALSE	Transition from "Starting" to "Held" is possible
suspended	Bool	FALSE	Transition from "Suspended" to "Held" is possible
execute	Bool	true	Transition from "Execute" to "Held" is possible
unholding	Bool	FALSE	Transition from "Unholding" to "Held" is possible
suspending	Bool	FALSE	Transition from "Suspending" to "Held" is possible
unsuspending	Bool	FALSE	Transition from "Unsuspending" to "Held" is possible
completing	Bool	FALSE	Transition from "Completing" to "Held" is possible
completed	Bool	FALSE	Transition from "Completed" to "Held" is possible

4.16.3 LUC_typeModeConfiguration (UDT / V1.2.0)

Description

Configure which modes are enabled

Parameter description

Identifier	Data type	Default value	Description
production	Bool	true	Enable unit mode Production
maintenance	Bool	true	Enable unit mode Maintenance
manual	Bool	true	Enable mode Manual (Mandatory, always set)
userMode01	Bool	FALSE	Enable user-defined unit mode 01
userMode02	Bool	FALSE	Enable user-defined unit mode 02
userMode03	Bool	FALSE	Enable user-defined unit mode 03
userMode04	Bool	FALSE	Enable user-defined unit mode 04
userMode05	Bool	FALSE	Enable user-defined unit mode 05

Identifier	Data type	Default value	Description
userMode06	Bool	FALSE	Enable user-defined unit mode 06
userMode07	Bool	FALSE	Enable user-defined unit mode 07
userMode08	Bool	FALSE	Enable user-defined unit mode 08
userMode09	Bool	FALSE	Enable user-defined unit mode 09
userMode10	Bool	FALSE	Enable user-defined unit mode 10
userMode11	Bool	FALSE	Enable user-defined unit mode 11
userMode12	Bool	FALSE	Enable user-defined unit mode 12
userMode13	Bool	FALSE	Enable user-defined unit mode 13

4.16.4 LUC_typeModeTransitionConfiguration (UDT / V1.1.0)

Description

Configure in which states a mode transition is allowed

Parameter description

Identifier	Data type	Default value	Description
clearing	Bool	FALSE	Allow a mode transition to/from state Clearing
stopped	Bool	true	Allow a mode transition to/from state Stopped
starting	Bool	FALSE	Allow a mode transition to/from state Starting
idle	Bool	true	Allow a mode transition to/from state Idle
suspended	Bool	FALSE	Allow a mode transition to/from state Suspended
execute	Bool	FALSE	Allow a mode transition to/from state Execute
stopping	Bool	FALSE	Allow a mode transition to/from state Stopping
aborting	Bool	FALSE	Allow a mode transition to/from state Aborting
aborted	Bool	true	Allow a mode transition to/from state Aborted
holding	Bool	FALSE	Allow a mode transition to/from state Holding
held	Bool	FALSE	Allow a mode transition to/from state Held
unholding	Bool	FALSE	Allow a mode transition to/from state Unholding
suspending	Bool	FALSE	Allow a mode transition to/from state Suspending
unsuspending	Bool	FALSE	Allow a mode transition to/from state Unsuspending
resetting	Bool	FALSE	Allow a mode transition to/from state Resetting
completing	Bool	FALSE	Allow a mode transition to/from state Completing
completed	Bool	FALSE	Allow a mode transition to/from state Completed

4.16.5 LUC_typeStateConfiguration (UDT / V1.1.0)

Description

Configure which states are enabled for each mode

Parameter description

Identifier	Data type	Default value	Description
clearing	Bool	true	Enable state Clearing
starting	Bool	true	Enable state Starting
suspended	Bool	true	Enable state Suspended
stopping	Bool	true	Enable state Stopping
aborting	Bool	true	Enable state Aborting
holding	Bool	true	Enable state Holding

Identifier	Data type	Default value	Description
held	Bool	true	Enable state Held
unholding	Bool	true	Enable state Unholding
suspending	Bool	true	Enable state Suspending
unsuspending	Bool	true	Enable state Unsuspending
resetting	Bool	true	Enable state Resetting
completing	Bool	true	Enable state Completing
completed	Bool	true	Enable state Completed

4.16.6 LUC_typeStateMachineConfiguration (UDT / V1.2.0)

Description

Configuration of the state machine

Parameter description

Identifier	Data type	Default value	Description
enableModes	LUC_typeModeConfiguration	---	Configure which modes are enabled
enableStates	Array[0..LPML_V2022_MAX_MODES_UPPER_LIMIT] of LUC_typeStateConfiguration	---	Configure which states are enabled for each mode
enableModeTransitions	Array[0..LPML_V2022_MAX_MODES_UPPER_LIMIT] of LUC_typeModeTransitionConfiguration	---	Configure in which states a mode transition is allowed
enableHoldCommand	LUC_typeHoldCommandConfiguration	---	Configure in which states a hold command is possible
enableCompleteCommand	LUC_typeCompleteCommandConfiguration	---	Configure in which states a complete command is possible

5 Operation description (Operator)

This chapter gives you an overview of the user interfaces and shows you how to operate them.

5.1 Mode Request Faceplate

The mode command allows to change between the different modes of the unit.



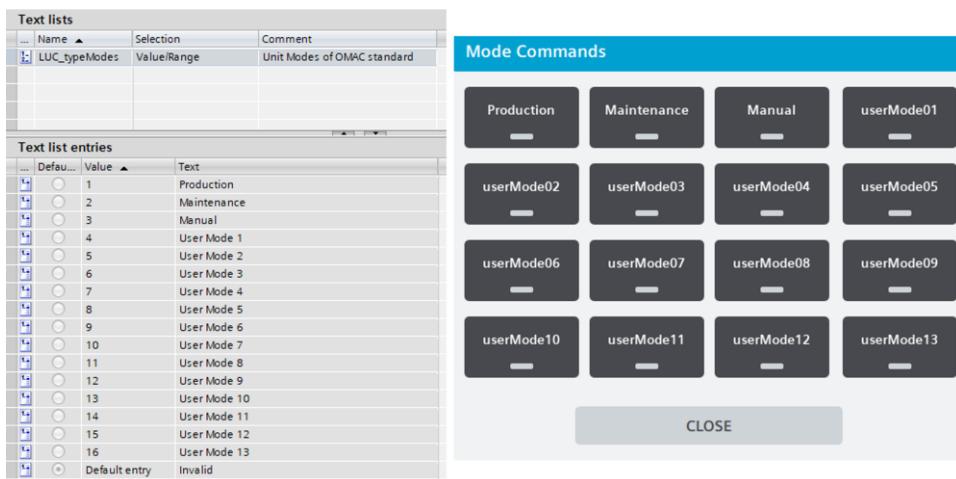
LUC Mode Request Faceplate Interface

The screenshot shows the 'Properties' tab of the LUC Mode Request Faceplate interface. The properties listed include:

Name	Static value	Dynamization (0)
Appearance		
Format		
Miscellaneous		
Connection status	None	
Faceplate type	LUC_ModeRequest V 1.1.1	
Icon		None
Interface		
topLC	HmiMainHeader.hmiToPlc.modeRequests	①
fromPLC	HmiMainHeader.plcToHmi.modeRequestsButtons	②
ButtonTextList	LUC_UnitModesTxLst V 1.1.0	③
Label		
Layer	Layer_0	
Name	Faceplate container_1	
Tab index	0	
Visibility	<input checked="" type="checkbox"/> None	
Security		

Mode request interface

No.	Parameter	Function
1	toPLC	Mode requests sent to the PLC from the HMI through a word type.
2	fromPLC	Properties of the mode request buttons sent from the PLC. a. Visibility b. Operator control allowed c. Background color
3	ButtonTextList	The text of the buttons is customized with a textlist. (See picture below)



5.2 State Command Faceplate

The state command provides the user an overview of the current state of the unit and allows the user to change between the different states. To be able to move to the right state, see the "state model" of the current mode.



LUC State Commands Faceplate Interface

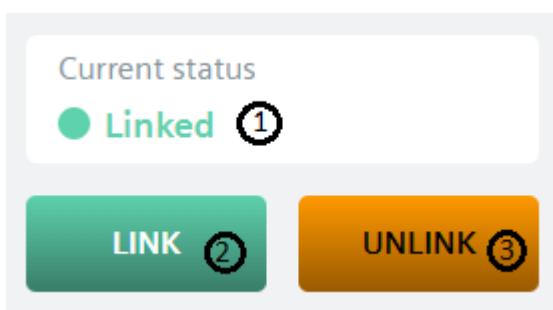
Faceplate container_1 [Faceplate container] [LUC_StateCommands V 1.1.1]		
Properties Events Texts Expressions		
Name	Static value	Dynamiz
▶ Appearance		
▶ Format		
▼ Miscellaneous		
▶ Connection status	None	
Faceplate type	LUC_StateCommands V 1.1.1	
▶ Icon		None
▼ Interface		
toPLC	HmiMainHeader.hmiToPlc.stateCommands	①
fromPLC	HmiMainHeader.plcToHmi.stateCommandsButtons	②
▶ Label		
Layer	Layer_0	
Name	Faceplate container_1	
Tab index	0	
▶ Visibility		<input checked="" type="checkbox"/> None

State command interface

No.	Parameter	Function
1	toPLC	State commands sent to the PLC from the HMI through a word type.
2	fromPLC	Properties of the state commands buttons sent from the PLC. a. Operator control allowed

5.3 Interlink to parent faceplate

5.4 This library provides a faceplate to link or unlink an equipment module from a unit. The faceplate contains the following features:



No.	Object	Function
1	Status	Status of the equipment module. Green: Linked / Orange: Unlinked
2	Link button	Sends a link command, to link the equipment module to the unit. Just allowed when the unit is the state aborted, stopped or idle.
3	Unlink button	Sends a unlink command, to unlink the equipment module from the unit. Just allowed when the unit is the state aborted, stopped or idle.

5.5 Mode state history

A history of the previous states and mode is stored in the PLC. The last 15 steps of the state machine are provided. The FB LUC_UpdateModeStateHistory (called in FB CallUnit) updates that history upon each mode or state change. This information is sent to the LUC_ModeStateHistory faceplate which shows the last 15 steps of the state machine.

No. ①	Mode ②	State ③	Duration ④
Current	Manual	Stopped	[DD hh:mm:ss]
-1	Manual	Stopping	00D 00:00:00
-2	Manual	Execute	00D 00:00:05
-3	Manual	Idle	00D 00:00:03
-4	Manual	Resetting	00D 00:00:01
-5	Manual	Stopped	00D 00:00:02
-6	Manual	Clearing	00D 00:00:00
-7	Manual	Aborted	00D 00:00:04
-8	Manual	Aborting	00D 00:00:02
-9	Manual	Clearing	00D 00:00:02
-10	Manual	Aborted	00D 00:00:09
-11	Manual	Aborting	00D 00:00:08
-12	Manual	Clearing	00D 00:00:02
-13	Manual	Aborted	00D 00:00:03
-14	Manual	Aborting	00D 00:00:12
-15	Manual	Stopped	00D 00:00:00

No.	Object	Function
1	Step number	Number of step stored in the state machine history
2	Mode	Active mode in each step
3	State	Active state in each step
4	Duration	Duration of the step until a mode or state change

5.6 Signal stack

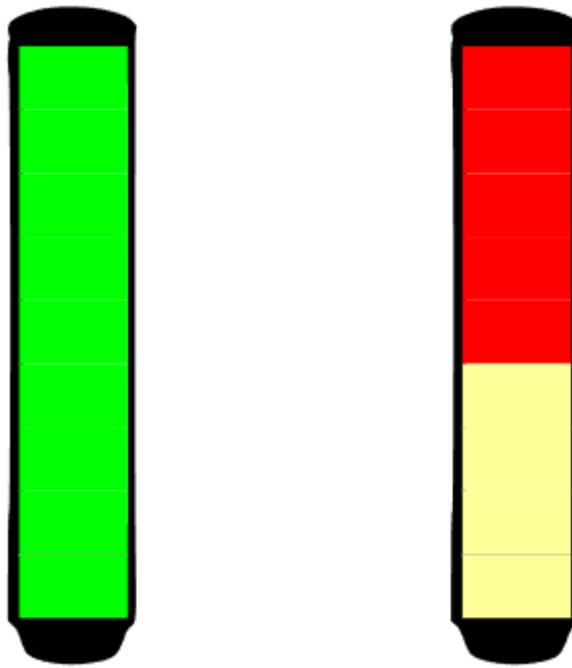
The LUC library provides a Dynamic Widget for a visual representation of the stack light. The stack light shows the current status of a unit.

How the stack light works

The stack light is composed of 9 segments, which are automatically and uniformly distributed among to the status of the unit.

5 Operation description (Operator)

This means that for example, if the active status of the unit is represented with one color, the entire area of the signaling column is illuminated in a single color to ensure maximum visibility.



If the active status of the unit is represented with multiple colors, the illuminated surface is split proportionally. If the segments cannot be uniformly distributed, the last segment is assigned to the color with the highest priority (highest position within the column). If there are several segments left, they are distributed evenly according to the prioritization, i.e. the positioning in the column from top to bottom.

The prioritization is given by the following chart:

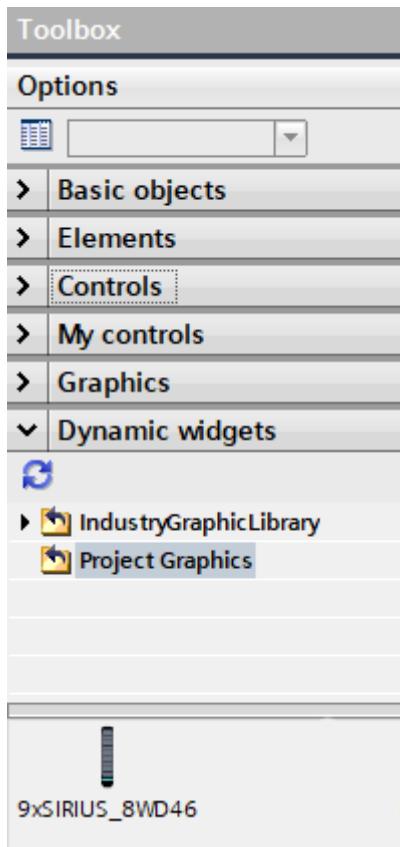
Priority	Color
0	Red
1	Yellow
2	Blue
3	Green
4	White
5	User Defined

How to use a dynamic widget

A dynamic widget is an SVG (Scalable Vector Graphics) object. To use this object in Tia Portal is necessary to have the SVG file stored in the following folder [project folder]\UserFiles\SvgControls.

A screenshot of a file browser window. The address bar shows the path: > LUC_SampleProject > UserFiles > SvgControls. Below the address bar is a table with four columns: Nombre, Fecha de modificación, Tipo, and Tamaño. The first row contains the file '9xSIRIUS_8WD46.svhmi', its modification date '20/05/2024 12:52', its type 'Archivo SVGHMI', and its size '6 KB'. There is also a small icon next to the file name.

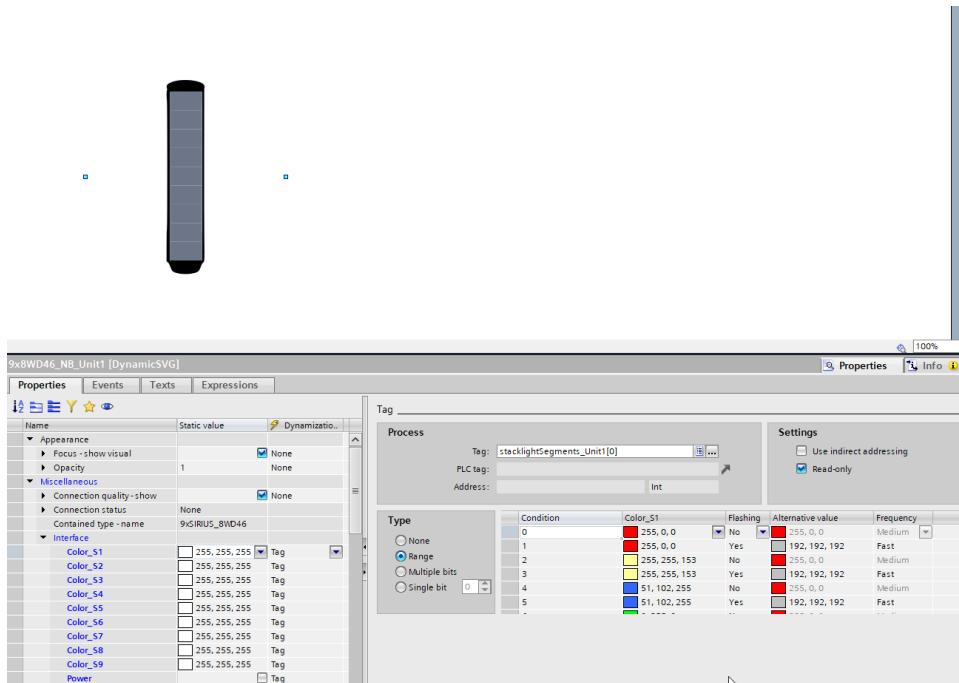
Once the file is stored, the dynamic object will be able to use in the toolbox of the HMI set up.



How to use and configure the stack light

With the LUC library, for the stack light implementation, is delivered a SVG and a script to define how the colors are managed in the segments according to the active colors.

- 1. SVG Stack light.** The user should drag and drop the dynamic widget in the correspondent screen and fill the interface parameters.



The parameters are the following:

- **Segments:** The stack light is composed by 9 segments, where the programmer has to assign a variable for each segment. The colors of each segment is given by the variable mentioned. In this sample project, each value means a different color and flashing status.

Value	Meaning
0	Red solid
1	Red flashing
2	Yellow solid
3	Yellow flashing
4	Blue solid
5	Blue flashing
6	Green solid
7	Green flashing
8	Horn solid
9	Horn flashing
10	White solid
11	White flashing
12..31	User Defined

- **Power status:** The stack light will be powered on in case is required to activate a color. Otherwise, the stack light would be powered off.

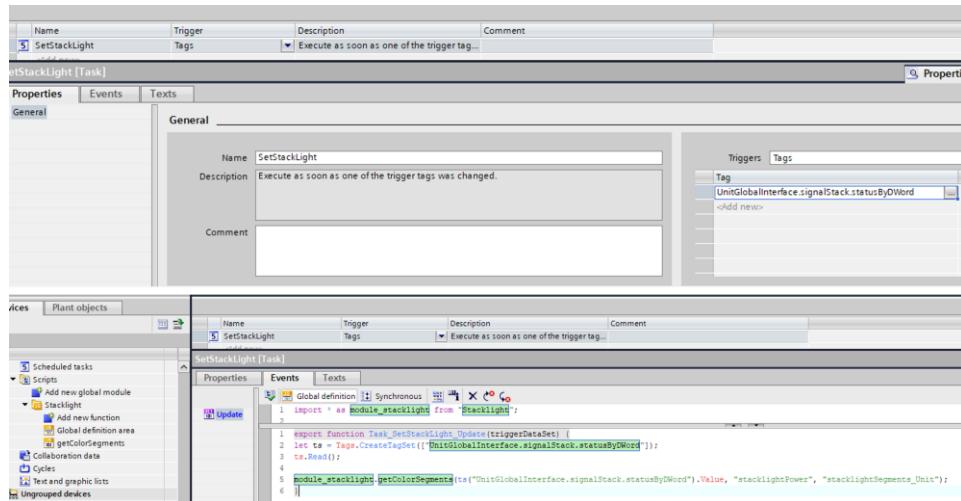
2. Script : *getColorSegments*

The script *getColorSegments* configure the segments according to how many colors are active to show the units status. The script receives the status of the unit (active colors) and writes the configuration of the segments and the power status of the stacklight.

- a. **Segments configuration:** The code define the segments configuration evaluating up to 5 active colors at the same time. In case there are multiple colors active, the assignment of the segments will follow the prioritization show in the table of the subchapter *How the stack light works* :
 - **1 color active.** The same color will be assign to all segments.
 - **2 colors active.** Half of the segments will show one color and the other half the second color.
 - **3 colors active.** Each color will be assigned to the stack light in 3 segments in a row.
 - **4 colors active.** Each color will be assigned to the stack light in 2 segments in a row.
 - **5 colors active.** the first 4 colors will be assigned to the stack light in 2 segments in a row, and the last color will be assigned to the last segment.
- b. **Power status:** The script evaluate if the status of the unit requires to activate any color. So, the stack light will be powered on in case there is required to activate a color. In case is nor required to activate any color, the stack light will be powered off.

The script should be called in a *Scheduled tasks*, where this script will be called every time that the unit status changes. As a proposal, in the sample project the event which calls the script was implemented in the following way:

5 Operation description (Operator)

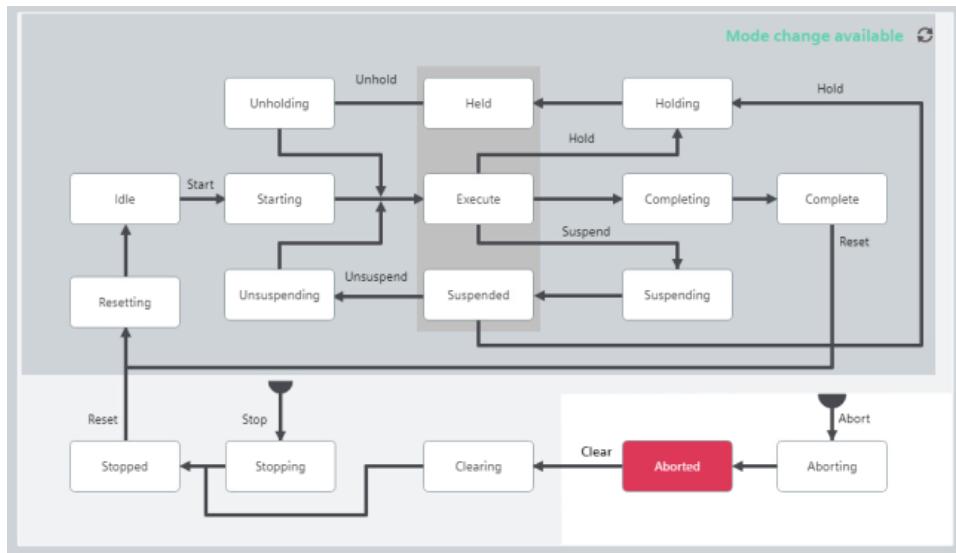


The user should create the HMI Internal variables *stacklightPower* (Int) and *stacklightSegments_Unit* (Array [0..9]of Int). In these variables the configuration of the power status and segments of the stacklight will be written accordingly of the information of the status of the unit *UnitGlobalInterface.signalStack.statusByDWord* send to the script.

5.7 State Model

The state model helps operator understand what the unit is doing and what is needed in order to move into a more productive state. The state model contains a set of predefined states divided into two categories.

1. Waiting states: the unit remains in a wait state until receiving a command to enter the next state. The waiting states include e.g. Idle, stopped, aborted etc.
2. Acting states: are states in which the unit must perform some temporary actions and move automatically into the next state when it is finished. Acting states end with “ing” and include e.g. Resetting, Starting, Clearing etc.



6 LUC Custom Style

In this library a style for the buttons is provided. The custom style file must be stored in the following location,[project folder]\UserFiles\Styles. This style must be used for the faceplates, in specific for the Mode Request and State Commands faceplate.

7 Appendix

7.1 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:

<https://www.siemens.com/industry/supportrequest>

SITRAIN - Digital Industry Academy

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:

<https://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 contract's

You can find detailed information on our range OF services in the service catalog Web page:

<https://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 iOS AND Android:

<https://support.industry.siemens.com/cs/ww/en/sc/2067>

7.2 Links and Literature

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/109974940
\3\	Programming Guidelines and Programming Style guide for SIMATIC S7-1200 and S7-1500 https://support.industry.siemens.com/cs/ww/en/view/81318674
\4\	Guideline on Library Handling in Tia Portal https://support.industry.siemens.com/cs/ww/en/view/109747503
\5\	Libraries in the TIA Portal https://support.industry.siemens.com/cs/ww/en/view/109738702

7.3 Change documentation

Version	Date	Modifications
V1.0.0	12/2023	First Version
V1.1.0	03/2024	Behaviour improvement
V1.2.0	09/2024	Behaviour improvements

7.4 Change log

Version & Date	Change description
V1.2.0 09/2024	<p>UPDATED: LUC_InterfaceManager / V1.2.0 <ul style="list-style-type: none"> • Optimized code. LUC_InterlinkToParent / V1.2.0 <ul style="list-style-type: none"> • Modified the block interface, removed ControlNodes interface. The equipment module can be linked or unlinked to/from the unit through the block interface inputs. LUC_ModeStateToString / V1.2.0 <ul style="list-style-type: none"> • Optimized code. LUC_OpcUaMethod_Abort / V1.2.0 <ul style="list-style-type: none"> • Removed the forcing of the ENO to true. LUC_OpcUaMethod_Clear / V1.2.0 <ul style="list-style-type: none"> • Removed the forcing of the ENO to true. LUC_OpcUaMethod_Complete / V1.2.0 <ul style="list-style-type: none"> • Removed the forcing of the ENO to true. LUC_OpcUaMethod_Hold / V1.2.0 <ul style="list-style-type: none"> • Removed the forcing of the ENO to true. LUC_OpcUaMethod_Reset / V1.2.0 <ul style="list-style-type: none"> • Removed the forcing of the ENO to true. LUC_OpcUaMethod_SetUnitMode / V1.2.0 <ul style="list-style-type: none"> • Removed the forcing of the ENO to true. LUC_OpcUaMethod_SetUnitState / V1.2.0 <ul style="list-style-type: none"> • Modified the interface for <i>statLastCycleState</i> from Temp to Static. Removed the forcing of the ENO to true. LUC_OpcUaMethod_Start / V1.2.0 <ul style="list-style-type: none"> • Removed the forcing of the ENO to true. LUC_OpcUaMethod_StateComplete / V1.2.0 <ul style="list-style-type: none"> • Removed the forcing of the ENO to true. LUC_OpcUaMethod_Stop / V1.2.0 <ul style="list-style-type: none"> • Removed the forcing of the ENO to true. LUC_OpcUaMethod_Suspend / V1.2.0 <ul style="list-style-type: none"> • Removed the forcing of the ENO to true. LUC_OpcUaMethod_Unhold / V1.2.0 <ul style="list-style-type: none"> • Removed the forcing of the ENO to true. LUC_OpcUaMethod_Unsuspend / V1.2.0 <ul style="list-style-type: none"> • Removed the forcing of the ENO to true. LUC_RemoteCtrl_OpcUa / V1.2.0 <ul style="list-style-type: none"> • Fixed SetUnitMode and SetUnitState bug. Removed the forcing of the ENO to true. LUC_SetUnitStatus / V1.2.0 <ul style="list-style-type: none"> • Optimized code. LUC_SignalStack / V1.2.0 <ul style="list-style-type: none"> • Optimized code. LUC_StateMachineConfig / V1.2.0 <ul style="list-style-type: none"> • Optimized code. LUC_StatusCollector / V1.2.0 <ul style="list-style-type: none"> • Optimized code. LUC_UpdateModeStateHistory / V1.2.0 <ul style="list-style-type: none"> • Optimized code. </p>
V1.1.0 06/2024	<p>NEW: LUC_InterfaceManager / V1.1.0 <ul style="list-style-type: none"> • First released version </p>

Version & Date	Change description
	<ul style="list-style-type: none"> Implemented 3 different interfaces of control (local, hmi, remote)and diagnosis status through just a word value “status”. <p>LUC_InterlinkToParent / V1.1.0</p> <ul style="list-style-type: none"> First released version Equipment Module share the linked status with the ControlNodes DB and HmiInterface DB of he Equipment Module. <p>LUC_ModeStateToString / V1.1.0</p> <ul style="list-style-type: none"> First released version Implementation of mode and state conversion in the same block. <p>LUC_OpcUaMethod_Abort / V1.1.0</p> <ul style="list-style-type: none"> First released version LPMLV2022 constants included as a local constants in the block <p>LUC_OpcUaMethod_Clear / V1.1.0</p> <ul style="list-style-type: none"> First released version LPMLV2022 constants included as a local constants in the block <p>LUC_OpcUaMethod_Complete / V1.1.0</p> <ul style="list-style-type: none"> First released version LPMLV2022 constants included as a local constants in the block <p>LUC_OpcUaMethod_Hold / V1.1.0</p> <ul style="list-style-type: none"> First released version LPMLV2022 constants included as a local constants in the block <p>LUC_OpcUaMethod_Reset / V1.1.0</p> <ul style="list-style-type: none"> First released version LPMLV2022 constants included as a local constants in the block <p>LUC_OpcUaMethod_SetUnitMode / V1.1.0</p> <ul style="list-style-type: none"> First released version LPMLV2022 constants included as a local constants in the block. Extended number of existing modes from 8 to 15. <p>LUC_OpcUaMethod_SetUnitState / V1.1.0</p> <ul style="list-style-type: none"> First released version LPMLV2022 constants included as a local constants in the block <p>LUC_OpcUaMethod_Start / V1.1.0</p> <ul style="list-style-type: none"> First released version LPMLV2022 constants included as a local constants in the block <p>LUC_OpcUaMethod_StateComplete / V1.1.0</p> <ul style="list-style-type: none"> First released version LPMLV2022 constants included as a local constants in the block <p>LUC_OpcUaMethod_Stop / V1.1.0</p> <ul style="list-style-type: none"> First released version LPMLV2022 constants included as a local constants in the block <p>LUC_OpcUaMethod_Suspend / V1.1.0</p> <ul style="list-style-type: none"> First released version LPMLV2022 constants included as a local constants in the block <p>LUC_OpcUaMethod_Unhold / V1.1.0</p> <ul style="list-style-type: none"> First released version LPMLV2022 constants included as a local constants in the block <p>LUC_OpcUaMethod_Unsuspend / V1.1.0</p> <ul style="list-style-type: none"> First released version LPMLV2022 constants included as a local constants in the block <p>LUC_RemoteCtrl_OpcUa / V1.1.0</p> <ul style="list-style-type: none"> First released version Extend number of existing modes from 8 to 15. <p>LUC_SetUnitStatus / V1.1.0</p> <ul style="list-style-type: none"> First released version

Version & Date	Change description
	<ul style="list-style-type: none"> • LPMLV2022 constants included as a local constants in the block LUC_SignalStack / V1.1.0 • First released version • Implemented 4 different ways to monitor Stack lights and horn (lightsByBool, lightsByDint, lightsByDWord and lightsStatus) <p>LUC_StateMachineConfig / V1.1.0</p> <ul style="list-style-type: none"> • First released version • Implemented state model configuration for all possible modes <p>LUC_StatusCollector / V1.1.0</p> <ul style="list-style-type: none"> • First released version • In this version, the input commands (that can set the processCommands are extended with the implementation of new commands. In addition, the Unit Status is taken into account to set the processCommands. <p>LUC_UpdateModeStateHistory / V1.1.0</p> <ul style="list-style-type: none"> • First released version