

**Festo CPX-AP-A Driver Block Library for
PCS7**

Commissioning devices using the PCS7 driver block library- for
CPX-AP-A Bus Node

CPX-AP-A, Siemens
PCS7

TitleFesto CPX-AP-A Driver Block Library for PCS7
Version1.0.0.0
Document no. 000001
Originalen
AuthorFesto

Last saved 16.09.2025

General information:

This document is intended for qualified, trained and instructed professionals. The data provided in this document are no guaranteed specifications, in particular with regard to functionality, condition or quality in the legal sense. The information in this document is intended only as simple indications for the implementation of a specific, hypothetical application, and in no way as a substitute for the operating instructions of the respective manufacturers or the design and testing of the respective application by the user. The respective operating instructions for Festo products can be found under www.festo.com. The user of this document must ensure that each function described herein works properly in his application. The user remains solely responsible for his or her own use of this document, even by studying this document and using the information mentioned therein. This also applies to any software (in source code and/or object code) that is made available to the user as an appendix to this document.

Rights of use of the software:

If software is made available to the user as an appendix to this document, the user is granted a simple and unlimited right of use hereto. This right also includes the processing and distribution of the software to third parties in edited or unedited form. The User is not permitted to use the name "Festo" to endorse or promote products derived from the Software without express prior written permission.

Software is provided to the user "as is". Festo does not assume any warranty or guarantee with regard to software. Festo's liability for damages of any kind arising from the use of the software is limited to intent and gross negligence.

Legal Notices:

©Festo SE & Co. KG, all rights reserved. A change in content and form is only permitted with the express written consent of Festo SE & Co. KG. Festo grants the user the right to reproduce this document in a form that remains unchanged in terms of content and form and to pass it on to third parties.

Table of contents

1	Introduction	5
1.1	Description	5
1.2	Installation	5
1.3	Version History	5
1.4	Change History	6
1.5	Supported Modules	6
2	Terminal/Rack Blocks	7
2.1	APARACK(FB6000)	7
2.1.1	Description.....	7
2.1.2	Calling OBs.....	7
2.1.3	Automatic Configuration through “Generate Module Drivers”	7
2.1.4	Manual Configuration	8
2.1.5	Function Block in CFC	8
2.1.6	Interface.....	8
2.1.7	Messages.....	10
3	Module Driver Blocks	11
3.1	APAMODID(FB6001)	11
3.1.1	Description.....	11
3.1.2	Calling OBs.....	11
3.1.3	Automatic Configuration through “Generate Module Drivers”	11
3.1.4	Manual Configuration	11
3.1.5	Function Block in CFC	12
3.1.6	Interface.....	12
3.1.7	Messages.....	13
3.2	APAVTSA(FB6002)	14
3.2.1	Description.....	14
3.2.2	Calling OBs.....	14
3.2.3	Automatic Configuration through “Generate Module Drivers”	14
3.2.4	Manual Configuration	14
3.2.5	Function Block in CFC	15
3.2.6	Interface.....	15
3.2.7	Messages.....	16
3.3	APAVTUG(FB6010)	17
3.3.1	Description.....	17
3.3.2	Calling OBs.....	17
3.3.3	Automatic Configuration through “Generate Module Drivers”	17
3.3.4	Manual Configuration	17
3.3.5	Function Block in CFC	18
3.3.6	Interface.....	18
3.3.7	Messages.....	19

3.4	CPXAPAPN(FB6009)	20
3.4.1	Description.....	20
3.4.2	Calling OBs.....	20
3.4.3	Automatic Configuration through “Generate Module Drivers”	20
3.4.4	Manual Configuration	20
3.4.5	Function Block in CFC.....	21
3.4.6	Interface.....	21
3.4.7	Messages.....	22
4	Channel Driver Blocks	24
4.1	CHAPADI(FC6003)	24
4.1.1	Description.....	24
4.1.2	Calling OBs.....	24
4.1.3	Automatic Configuration with “Generate Module Drivers”	24
4.1.4	Function Block in CFC.....	24
4.1.5	Quality Code	25
4.1.6	Operation modes:	25
4.1.7	Interface.....	26
4.2	CHAPADO(FC6004).....	27
4.2.1	Description.....	27
4.2.2	Calling OBs.....	27
4.2.3	Automatic Configuration with “Generate Module Drivers”	27
4.2.4	Function Block in CFC.....	27
4.2.5	Quality Code	27
4.2.6	Operation modes:	28
4.2.7	Interface.....	28
4.3	APADiln(FC6003).....	29
4.3.1	Description.....	29
4.3.2	Calling OBs.....	29
4.3.3	Automatic Configuration with “Generate Module Drivers”	29
4.3.4	Function Block in CFC.....	29
4.3.5	Quality Code	30
4.3.6	Operation modes:	30
4.3.7	Interface.....	31
4.4	APADiOu(FC653)	32
4.4.1	Description.....	32
4.4.2	Calling OBs.....	32
4.4.3	Automatic Configuration with “Generate Module Drivers”	32
4.4.4	Function Block in CFC.....	32
4.4.5	Quality Code	33
4.4.6	Operation modes:	33
4.4.7	Interface.....	34

1 Introduction

The Festo CPX-AP-A driver library makes it possible to easily connect a Festo CPX-AP-A system to a SIMATIC PCS7 control system. The library supports the automatic generation of the driver connections using "Generate Module Drivers" from the Hardware configuration. Every channel of the CPX-AP-A system can be controlled, monitored and diagnosed.

1.1 Description

The CPX-AP-A Driver Library makes use of AP Parameter Read function to provide module and channel specific diagnostics in Function Block as well as OS Alarm. The Library has 3 Hierarchies/families.

1. Terminal/Rack
2. Module
3. Channel

This library consists of 11 Function blocks which are automatically interconnected and parameterized when "Generate module drivers" function is selected when compiling an S7 400 system. Each block is called in their respective charts having suffix "@CPX_APA_". Siemens specific basis library blocks like OB_DIAG1_PN, AND, ALARM_8P are also called.

Object name	Symbolic name	Created in language	Size in the work me.	Type	Version (Header)	Unlinked	Name (Header)	Author	Standard block	Non-Retain	Last interface change	OS write-protected in	Last modified	Mod
FB0000	APARACK	SCL	18406	Function Block	1.0	---	APARACK	FESTO	---	---	08/09/2025 08:55.5	---	08/09/2025 09:37:37 AM	---
FB0001	APAMODID	SCL	1025	Function Block	1.0	---	APAMODID	FESTO	---	---	08/09/2025 07:03.4	---	08/09/2025 09:38:49 AM	---
FB0002	APAPISA	SCL	1452	Function Block	1.0	---	APAPISA	FESTO	---	---	08/09/2025 07:05.0	---	08/09/2025 09:40:59 AM	---
FB0007	APAPAPA	SCL	7369	Function Block	1.0	---	APAPAPA	FESTO	---	---	03/10/2025 04:50.4	---	03/10/2025 04:51:21 AM	---
FB0009	CPAPAPN	SCL	4126	Function Block	1.0	---	CPAPAPN	FESTO	---	---	08/05/2025 11:41.3	---	08/05/2025 11:41:36 AM	---
FB0010	APAWTUG	SCL	1432	Function Block	1.0	---	APAWTUG	FESTO	---	---	08/09/2025 07:11.3	---	08/09/2025 09:42:52 AM	---
FC0003	CHAPADI	SCL	256	Function	1.0	---	CHAPADI	FESTO	---	---	03/10/2025 05:03.0	---	03/10/2025 05:03:05 AM	---
FC0004	CHAPADO	SCL	194	Function	1.0	---	CHAPADO	FESTO	---	---	03/10/2025 05:03.5	---	03/10/2025 05:04:06 AM	---
FC0005	APADIN	SCL	630	Function	1.0	---	APADIN	FESTO	---	---	03/10/2025 05:00.2	---	03/10/2025 05:00:28 AM	---
FC0006	APADOU	SCL	892	Function	1.0	---	APADOU	FESTO	---	---	03/10/2025 05:02.0	---	03/10/2025 05:02:09 AM	---
FC0008	FCBjst_Word	SCL	148	Function	1.0	---	FCBjst_Word	Festo	---	---	03/10/2025 04:49.2	---	03/10/2025 04:49:22 AM	---
SFB4	TON	STL	---	System function block	1.0	---	TON	SIMATIC	---	---	08/09/1996 09:21.4	---	08/09/1996 09:21:48 AM	---
SFB35	ALARM_8P	STL	---	System function block	1.1	---	ALARM_8P	SIMATIC	---	---	12/13/1995 05:11.5	---	12/06/1996 12:22:01 PM	---
SFB52	RDRREC	STL	---	System function block	1.0	---	RDRREC	SIMATIC	---	---	11/23/2000 03:30.0	---	11/23/2000 03:30:00 PM	---
SFB53	VRREC	STL	---	System function block	1.0	---	VRREC	SIMATIC	---	---	11/23/2000 03:30.0	---	11/23/2000 03:30:00 PM	---

1.2 Installation

The library is found as an executable file in Festo Support Portal. After downloading, run the installation file.

1.3 Version History

Version	Date	Description
1.0.0.0	09-09-2025	Initial Driver Block Library Release

1.4 Change History

Version	Date	Description
1.0.0.0	09-09-2025	Release: New Function Blocks: CHAPADI: Digital Input Channel Driver Block CHAPADO: Digital Output Channel Driver Block APADiIn: Digital Input Channel Driver Block (APL) APADiOu: Digital Output Channel Driver Block (APL) CPXAPAPN: APA PN Head Module Driver Block APAMODID: 16 Digital Input module Driver Block APAVTSA: APA-VTSA Interface Module Driver Block APAVTUG: APA-VTUG Interface Module Driver Block APARACK: APA Rack Driver Block

1.5 Supported Modules

The following hardware modules are supported by the Festo CPX-AP-A driver library for PCS 7:

Type	GSD/GSDML Entry	Description	Release
16DI-D	CPX-AP-A-16DI-D-M12-5P	16 Channel Digital Input Module	1.0.0.0
VTSA	VABA-S6-1-X5	CPX-AP-A to VTSA-F Pneumatic-Interface with 32 Outputs	1.0.0.0
VTUG	VAEM-L1-S-12-AP	VTUG electrical interface with 24 outputs	1.0.0.0
VTUG	VAEM-L1-S-24-AP	VTUG electrical interface with 48 outputs	1.0.0.0

The following Fieldbus Modules are supported by the Festo CPX-E driver library for PCS 7:

Name	Fieldbus	Description	Release
CPX-AP-A-PN-12	Profinet	PROFINET interface for CPX-AP-A, 2x M12 D-coded	1.0.0.0

2 Terminal/Rack Blocks

2.1 APARACK(FB6000)

2.1.1 Description

This is the main function block that represents the APA system. The main task of this function block is to handle AP Parameters through APAPARA block. This function block reads Parameter 20916 Instance 0 and maps the system status consisting of 80 Bytes into a structure named "SYS_Diag". Once the system status is stored, the function block sequentially checks individual modules for errors using the Parameter 20195 and instance as the respective module number. After inspecting each module for errors, it fills a structure containing the diagnostic data. This structure is interconnected with respective module driver function block. The module driver blocks further breakdown structure to retrieve alarms and messages.

If the OB_DIAG1 block superior to APARACK detects a slave failure (complete DP/PN slave missing or not working), this is signaled via the bRackF connector to APARACK which in turn forwards the information to the module driver blocks via its bRackFOut connector. If the OB_DIAG1 block signals a subnet/redundancy error via the bSubn1Err and/or bSubn2Err connectors, the error information is forwarded to the module driver blocks by setting bSubnErr.

2.1.2 Calling OBs

Type	Description
OB 1	Cyclic Program

2.1.3 Automatic Configuration through "Generate Module Drivers"

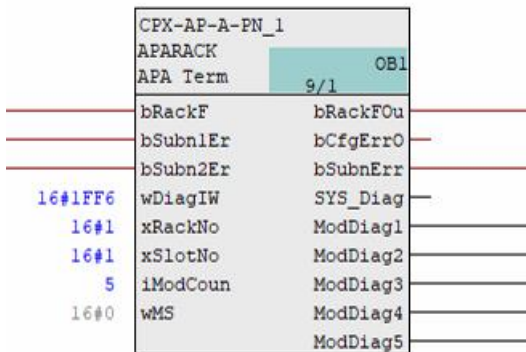
For each CPX-AP-A terminal found in the hardware configuration, the following actions are executed automatically by the "Generate Module Drivers" function:

1. OB_DIAG1 is installed in the run sequence after the corresponding SUBNET block (for monitoring the Profibus DP slave/Profinet IO device upon failure and recovers; see Siemens Help for PCS 7/Basis Library for more details).
2. APARACK is installed in the run sequence after the previously installed OB_DIAG1 block.
3. The wDiagIW, xRackNo, xSlotNo, iModCount, xSubn1Id and xSubn2Id signals are configured.
4. The EN input of OB_DIAG1 is interconnected with the output of an AND block whose inputs are interconnected to the EN_SUBx output of the OB_BEGIN block, and to the EN_Rx output of the SUBNET block.
5. The CPU_DIAG and CPU_OB5x OUT structures of the OB_BEGIN block and SUB_DIAG of the SUBNET block are interconnected with the IN_OUT structures of the same name of the OB_DIAG1 block.
6. The bRackF, bSubn1Err and bSubn2Err signals of the APARACK block are interconnected with the RACKF, SUBN1ERR and SUBN2ERR signals of the OB_DIAG1 block.
7. The bRackFOut, bSubnErr and ModDiagx signals are configured by the subordinate module drivers APAMODx.

2.1.4 Manual Configuration

If the terminal's module configuration is changed manually, verify that all the signals specified in **section 2.1.3** are set up correctly. It is recommended that this block is set up by using generate module driver function automatically.

2.1.5 Function Block in CFC



2.1.6 Interface

Name: Visible in the CFC

Name: Not visible in the CFC (can be changed via Properties)

Name: Output, otherwise input(ACC_ID is IN_OUT).

Lock (L): X => Cannot be changed by the user

Set (S): V => Signal is set automatically

C => Signal is connected automatically

X => Signal is changed during the runtime

Name (I/O)	Description	Data type	Default	L	S
bRackF	From OB_DIAG1: PNIO device broken (1=error)	BOOL	FALSE	C	X
bSubn1Err	From OB_DIAG1: PNIO subnet 1 failure (1=error)	BOOL	FALSE	C	X
bSubn2Err	From OB_DIAG1: PNIO redundant failure (1=error)	BOOL	FALSE	C	X
xSubn1Id	ID of primary DP master system	BYTE	16#FF	X	V
xSubn2Id	ID of redundant DP master system	BYTE	16#FF	X	V
wDiagIW	Diagnostic Address of CPX-APA-PN	WORD	0	X	V
xRackNo	PNIO device number	BYTE	0	X	V
xSlotNo	Slot number of head module	BYTE	0	X	V
wMS	Maintenance State	WORD	0	X	V
ACC_ID	1=parameters changed by driver generator, re-initialize	BOOL	FALSE	X	V
bRackFOu	1=PNIO device broken Output	BOOL	FALSE	C	X
bCfgrErrOut	1=Configuration error Output	BOOL	FALSE	C	X
bSubnErrOut	PNIO subnet failure (1=error)	BOOL	FALSE	C	X
wMSOut	Maintenance state Output	WORD	0	C	X
SYS_Diag	System Diagnostic State	STRUCT	-		X
ModDiag1	Diagnostic Information of Module at slot 1	STRUCT	-	C	X

[illegible]

ModDiag64	Diagnostic Information of Module at slot 64	STRUCT	-	C	X
ModDiag65	Diagnostic Information of Module at slot 65	STRUCT	-	C	X
ModDiag66	Diagnostic Information of Module at slot 66	STRUCT	-	C	X
ModDiag67	Diagnostic Information of Module at slot 67	STRUCT	-	C	X
ModDiag68	Diagnostic Information of Module at slot 68	STRUCT	-	C	X
ModDiag69	Diagnostic Information of Module at slot 69	STRUCT	-	C	X
ModDiag70	Diagnostic Information of Module at slot 70	STRUCT	-	C	X
ModDiag71	Diagnostic Information of Module at slot 71	STRUCT	-	C	X
ModDiag72	Diagnostic Information of Module at slot 72	STRUCT	-	C	X
ModDiag73	Diagnostic Information of Module at slot 73	STRUCT	-	C	X
ModDiag74	Diagnostic Information of Module at slot 74	STRUCT	-	C	X
ModDiag75	Diagnostic Information of Module at slot 75	STRUCT	-	C	X
ModDiag76	Diagnostic Information of Module at slot 76	STRUCT	-	C	X
ModDiag77	Diagnostic Information of Module at slot 77	STRUCT	-	C	X
ModDiag78	Diagnostic Information of Module at slot 78	STRUCT	-	C	X
ModDiag79	Diagnostic Information of Module at slot 79	STRUCT	-	C	X
ModDiag80	Diagnostic Information of Module at slot 80	STRUCT	-	C	X

2.1.7 Messages

APARACK does not have any messages to report to OS.

3 Module Driver Blocks

3.1 APAMODID(FB6001)

3.1.1 Description

The APAMODID function block represents a CPX-AP-A Digital Input Module with channel diagnostics(16DI-D) in the system. The status and diagnostic information is forwarded to the relevant channel through ModeOut parameter. The necessary connections are generated automatically.

Functionally, this block receives Status of the module through ModDiag input from APARACK and decodes the input. The error code is displayed in the output dErrorCode. It also creates OS alarms of errors.

3.1.2 Calling OBs

Type	Description
OB 1	Cyclic Program
OB 82	Diagnostic Interrupt Organization Block
OB 83	Insert/Remove module Interrupt
OB 100	Warm Restart

3.1.3 Automatic Configuration through “Generate Module Drivers”

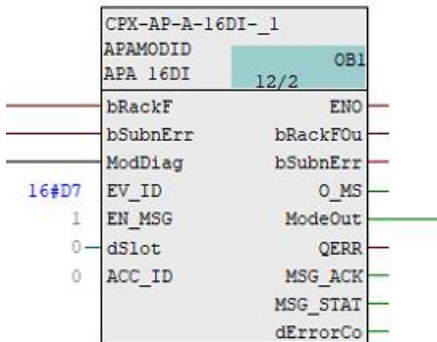
For each CPX-AP-A 16 DI-D module found in the hardware configuration, the following actions are executed automatically by the "Generate Module Drivers" function:

1. APAMODID is installed in the run sequence.
2. bRackF, bSubnErr, ModDiag inputs are connected with respective APARACK block outputs (bRackFOut, bSubnErr, ModDiagx).
3. EV_ID, EN_MSG, ACC_ID, QERR, MSG_ACK, MSG_STAT parameters are configured to setup OS message creation.

3.1.4 Manual Configuration

If the module configuration is changed manually, verify that all the signals specified in **section 3.1.3** are set up correctly. It is recommended that this block is set up by using generate module driver function automatically.

3.1.5 Function Block in CFC



3.1.6 Interface

Name: Visible in the CFC

Name: Not visible in the CFC (can be changed via Properties)

Name: Output, otherwise input(ACC_ID is IN_OUT).

Lock (L): X => Cannot be changed by the user

Set (S): V => Signal is set automatically

C => Signal is connected automatically

X => Signal is changed during the runtime

Name (I/O)	Description	Data type	Default	L	S
bRackF	From APARACK: PNIO device broken (1=error)	BOOL	FALSE	C	X
bSubnErr	From APARACK: PNIO system failure (1=error)	BOOL	FALSE	C	X
ModDiag	ID of primary DP master system	STRUCT		C	X
xSubn2Id	ID of redundant DP master system	STRUCT		C	X
EV_ID	Message ID	DWORD	0	X	V
EN_MSG	Enable messages	WORD	0	X	V
ACC_ID	1=parameters changed by driver generator, re-initialize	BOOL	FALSE	X	V
bRackFOut	1= PNIO device broken Output	BOOL	FALSE	C	X
bSubnErrOut	PNIO subnet failure (1=error)	BOOL	FALSE	C	X
O_MS	Maintenance state to Channel Driver Block	DWORD	0		
ModeOut	Mode of Channels connected to the current module	DWORD	0	C	X
QERR	1=program error (ALARM_8P failed, see MSG_STAT)	BOOL	FALSE	C	X
MSG_ACK	Acknowledge status for messages	WORD	0		X
MSG_STAT	Error Information of ALARM_8P	WORD	0		X
dErrorCode	Error Code of the module (see manual)	DWORD	0		

3.1.7 Messages

ALARM_8P Instance	Message Number	Default Message Text	Message Class
EV_ID	SIG1	Short Circuit/Overload in sensor supply	S
EV_ID	SIG2	Undervoltage in logic supply (PS) 24 VDC	S
EV_ID	SIG3	Overvoltage in logic supply (PS) 24 VDC	S
EV_ID	SIG4	Communication to AP Master Interrupted	S
EV_ID	SIG5	Output process data watchdog of the AP module expired	S
EV_ID	SIG6	Input process data watchdog of the AP module expired	S
EV_ID	SIG7	Device in emergency operation system state	S
EV_ID	SIG8	(not used)	N/A

3.2 APAVTSA(FB6002)

3.2.1 Description

The APAVTSA function block represents a CPX-AP-A VTSA valve terminal module(VABA-S6-1-X5) in the system. The status and diagnostic information is forwarded to the relevant channel through ModeOut parameter. The necessary connections are generated automatically.

Functionally, this block receives Status of the module through ModDiag input from APARACK and decodes the input. The error code is displayed in the output dErrorCode. It also creates OS alarms of errors.

3.2.2 Calling OBs

Type	Description
OB 1	Cyclic Program
OB 100	Warm Restart

3.2.3 Automatic Configuration through “Generate Module Drivers”

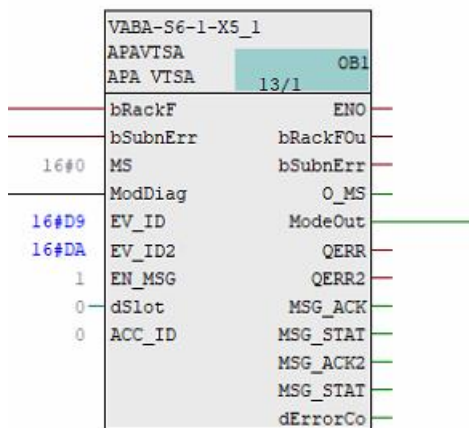
For each CPX-AP-A VABA-S6-1-X5 module found in the hardware configuration, the following actions are executed automatically by the "Generate Module Drivers" function:

4. APAVTSA is installed in the run sequence.
5. bRackF, bSubnErr, ModDiag inputs are connected with respective APARACK block outputs (bRackFOut, bSubnErr, ModDiagx).
6. EV_ID, EN_MSG, ACC_ID, QERR, MSG_ACK, MSG_STAT, EV_ID2, ACC_ID2, QERR2, MSG_ACK2, MSG_STAT2 parameters are configured to setup OS message creation.

3.2.4 Manual Configuration

If the module configuration is changed manually, verify that all the signals specified in **section 3.2.3** are set up correctly. It is recommended that this block is set up by using generate module driver function automatically.

3.2.5 Function Block in CFC



3.2.6 Interface

Name: Visible in the CFC

Name: Not visible in the CFC (can be changed via Properties)

Name: Output, otherwise input(ACC_ID is IN_OUT).

Lock (L): X => Cannot be changed by the user

Set (S): V => Signal is set automatically

C => Signal is connected automatically

X => Signal is changed during the runtime

Name (I/O)	Description	Data type	Default	L	S
bRackF	From APARACK: PNIO device broken (1=error)	BOOL	FALSE	C	X
bSubnErr	From APARACK: PNIO system failure (1=error)	BOOL	FALSE	C	X
ModDiag	ID of primary DP master system	STRUCT		C	X
EV_ID	Message ID	DWORD	0	X	V
EV_ID2	Message ID 2	DWORD	0	X	V
EN_MSG	Enable messages	WORD	0	X	V
ACC_ID	1=parameters changed by driver generator, re-initialize	BOOL	FALSE	X	V
bRackFOu	1= PNIO device broken Output	BOOL	FALSE	C	X
bSubnErrOut	PNIO subnet failure (1=error)	BOOL	FALSE	C	X
O_MS	Maintenance state to Channel Driver Block	DWORD	0		
ModeOut	Mode of Channels connected to the current module	DWORD	0	C	X
QERR	1=program error (ALARM_8P failed, see MSG_STAT)	BOOL	FALSE	C	X
MSG_ACK	Acknowledge status for messages	WORD	0		X
MSG_STAT	Error Information of ALARM_8P	WORD	0		X
QERR2	1=program error (ALARM_8P failed, see MSG_STAT)	BOOL	FALSE	C	X
MSG_ACK2	Acknowledge status for messages	WORD	0		X
MSG_STAT2	Error Information of ALARM_8P	WORD	0		X
dErrorCode	Error Code of the module (see manual)	DWORD	0		

3.2.7 Messages

ALARM_8P Instance	Message Number	Default Message Text	Message Class
EV_ID	SIG1	Short Circuit/Overload in output signal	S
EV_ID	SIG2	Undervoltage in logic supply (PS) 24 VDC	S
EV_ID	SIG3	Overvoltage in logic supply (PS) 24 VDC	S
EV_ID	SIG4	Undervoltage in load supply (PS) 24 VDC	S
EV_ID	SIG5	Overvoltage in load supply (PS) 24 VDC	S
EV_ID	SIG6	Valve coil not connected	S
EV_ID	SIG7	Communication to AP master interrupted	S
EV_ID	SIG8	Output process data watchdog of the AP module expired	S
EV_ID2	SIG1	System state emergency operation mode	S
EV_ID2	SIG2	Condition counter for valve coil reached	S
EV_ID2	SIG3	Shutdown of load supply (PL) 24 V DC	S
EV_ID2	SIG4	(not used)	N/A
EV_ID2	SIG5	(not used)	N/A
EV_ID2	SIG6	(not used)	N/A
EV_ID2	SIG7	(not used)	N/A
EV_ID2	SIG8	(not used)	N/A

3.3 APAVTUG(FB6010)

3.3.1 Description

The APAVTUG function block represents a CPX-AP-A VTUG valve terminal module(VAEM-L1-S-12/24) in the system. The status and diagnostic information is forwarded to the relevant channel through ModeOut parameter. The necessary connections are generated automatically.

Functionally, this block receives Status of the module through ModDiag input from APARACK and decodes the input. The error code is displayed in the output dErrorCode. It also creates OS alarms of errors.

3.3.2 Calling OBs

Type	Description
OB 1	Cyclic Program
OB 100	Warm Restart

3.3.3 Automatic Configuration through “Generate Module Drivers”

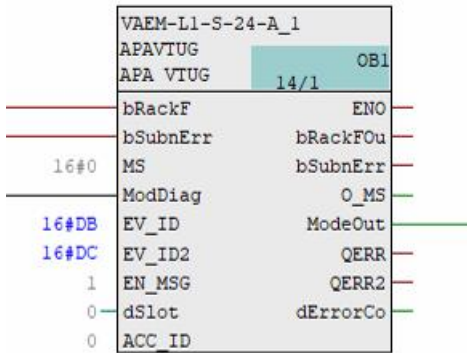
For each CPX-AP-A VAEM-L1-S-12/24 module found in the hardware configuration, the following actions are executed automatically by the "Generate Module Drivers" function:

7. APAVTUG is installed in the run sequence.
8. bRackF, bSubnErr, ModDiag inputs are connected with respective APARACK block outputs (bRackFOut, bSubnErr, ModDiagx).
9. EV_ID, EN_MSG, ACC_ID, QERR, MSG_ACK, MSG_STAT, EV_ID2, ACC_ID2, QERR2, MSG_ACK2, MSG_STAT2 parameters are configured to setup OS message creation.

3.3.4 Manual Configuration

If the module configuration is changed manually, verify that all the signals specified in **section 3.3.3** are set up correctly. It is recommended that this block is set up by using generate module driver function automatically.

3.3.5 Function Block in CFC



3.3.6 Interface

Name: Visible in the CFC

Name: Not visible in the CFC (can be changed via Properties)

Name: Output, otherwise input(ACC_ID is IN_OUT).

Lock (L): X => Cannot be changed by the user

Set (S): V => Signal is set automatically

C => Signal is connected automatically

X => Signal is changed during the runtime

Name (I/O)	Description	Data type	Default	L	S
bRackF	From APARACK: PNIO device broken (1=error)	BOOL	FALSE	C	X
bSubnErr	From APARACK: PNIO system failure (1=error)	BOOL	FALSE	C	X
ModDiag	ID of primary DP master system	STRUCT		C	X
EV_ID	Message ID	DWORD	0	X	V
EV_ID2	Message ID 2	DWORD	0	X	V
EN_MSG	Enable messages	WORD	0	X	V
ACC_ID	1=parameters changed by driver generator, re-initialize	BOOL	FALSE	X	V
bRackFOu	1= PNIO device broken Output	BOOL	FALSE	C	X
bSubnErrOut	PNIO subnet failure (1=error)	BOOL	FALSE	C	X
O_MS	Maintenance state to Channel Driver Block	DWORD	0		
ModeOut	Mode of Channels connected to the current module	DWORD	0	C	X
QERR	1=program error (ALARM_8P failed, see MSG_STAT)	BOOL	FALSE	C	X
MSG_ACK	Acknowledge status for messages	WORD	0		X
MSG_STAT	Error Information of ALARM_8P	WORD	0		X
QERR2	1=program error (ALARM_8P failed, see MSG_STAT)	BOOL	FALSE	C	X
MSG_ACK2	Acknowledge status for messages	WORD	0		X
MSG_STAT2	Error Information of ALARM_8P	WORD	0		X
dErrorCode	Error Code of the module (see manual)	DWORD	0		

3.3.7 Messages

ALARM_8P Instance	Message Number	Default Message Text	Message Class
EV_ID	SIG1	Undervoltage in logic supply (PS) 24 VDC	S
EV_ID	SIG2	Overvoltage in logic supply (PS) 24 VDC	S
EV_ID	SIG3	Undervoltage in load supply (PS) 24 VDC	S
EV_ID	SIG4	Shutdown of load supply (PS) 24 VDC	S
EV_ID	SIG5	Overvoltage in load supply (PS) 24 VDC	S
EV_ID	SIG6	Valve Fault	S
EV_ID	SIG7	Communication to AP master interrupted	S
EV_ID	SIG8	Output process data watchdog of the AP module expired	S
EV_ID2	SIG1	Input process data watchdog of the AP module expired	S
EV_ID2	SIG2	System state emergency operation mode	S
EV_ID2	SIG3	Condition counter for valve coil reached	S
EV_ID2	SIG4	(not used)	N/A
EV_ID2	SIG5	(not used)	N/A
EV_ID2	SIG6	(not used)	N/A
EV_ID2	SIG7	(not used)	N/A
EV_ID2	SIG8	(not used)	N/A

3.4 CPXAPAPN(FB6009)

3.4.1 Description

The CPXAPAPN function block represents a CPX-AP-A PN Head module(CPX-AP-A=PN-M12) in the system. The necessary connections are generated automatically. Functionally, this block receives Status of the module through ModDiag input from APARACK and decodes the input. The error code is displayed in the output dErrorCode. It also creates OS alarms of errors.

3.4.2 Calling OBs

Type	Description
OB 1	Cyclic Program
OB 82	Diagnostic Interrupt Organization Block
OB 83	Insert/Remove module Interrupt
OB 100	Warm Restart

3.4.3 Automatic Configuration through “Generate Module Drivers”

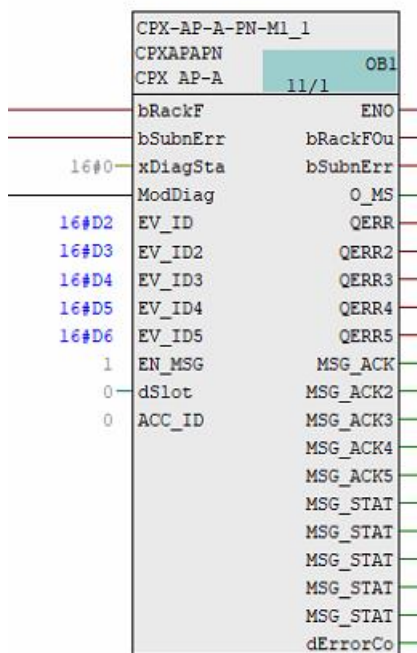
For each CPX-AP-A CPX-AP-A=PN-M12 module found in the hardware configuration, the following actions are executed automatically by the "Generate Module Drivers" function:

1. CPXAPAPN is installed in the run sequence.
2. bRackF, bSubnErr, ModDiag inputs are connected with respective APARACK block outputs (bRackFOut, bSubnErr, ModDiagx).
3. EV_ID, EN_MSG, ACC_ID, QERR, MSG_ACK, MSG_STAT, EV_ID2, ACC_ID2, QERR2, MSG_ACK2, MSG_STAT2, EV_ID3, ACC_ID3, QERR3, MSG_ACK3, MSG_STAT3, EV_ID4, ACC_ID4, QERR4, MSG_ACK4, MSG_STAT4, EV_ID5, ACC_ID5, QERR5, MSG_ACK5, MSG_STAT5 parameters are configured to setup OS message creation.

3.4.4 Manual Configuration

If the module configuration is changed manually, verify that all the signals specified in **section 3.4.3** are set up correctly. It is recommended that this block is set up by using generate module driver function automatically.

3.4.5 Function Block in CFC



3.4.6 Interface

Name: Visible in the CFC

Name: Not visible in the CFC (can be changed via Properties)

Name: Output, otherwise input(ACC_ID is IN_OUT).

Lock (L): X => Cannot be changed by the user

Set (S): V => Signal is set automatically

C => Signal is connected automatically

X => Signal is changed during the runtime

Name (I/O)	Description	Data type	Default	L	S
bRackF	From APARACK: PNIO device broken (1=error)	BOOL	FALSE	C	X
bSubnErr	From APARACK: PNIO system failure (1=error)	BOOL	FALSE	C	X
ModDiag	ID of primary DP master system	STRUCT		C	X
EV_ID	Message ID	DWORD	0	X	V
EV_ID2	Message ID 2	DWORD	0	X	V
EV_ID3	Message ID3	DWORD	0	X	V
EV_ID4	Message ID4	DWORD	0	X	V
EV_ID5	Message ID 5	DWORD	0	X	V
EN_MSG	Enable messages	WORD	0	X	V
ACC_ID	1=parameters changed by driver generator, re-initialize	BOOL	FALSE	X	V
bRackFOut	1= PNIO device broken Output	BOOL	FALSE	C	X
bSubnErrOut	PNIO subnet failure (1=error)	BOOL	FALSE	C	X
O_MS	Maintenance state to Channel Driver Block	DWORD	0		
ModeOut	Mode of Channels connected to the current module	DWORD	0	C	X
QERR	1=program error (ALARM_8P failed, see MSG_STAT)	BOOL	FALSE	C	X
QERR2	1=program error (ALARM_8P failed, see MSG_STAT)	BOOL	FALSE	C	X
QERR3	1=program error (ALARM_8P failed, see MSG_STAT)	BOOL	FALSE	C	X
QERR4	1=program error (ALARM_8P failed, see MSG_STAT)	BOOL	FALSE	C	X
QERR5	1=program error (ALARM_8P failed, see MSG_STAT)	BOOL	FALSE	C	X

MSG_ACK	Acknowledge status for messages	WORD	0		X
MSG_ACK2	Acknowledge status for messages	WORD	0		X
MSG_ACK3	Acknowledge status for messages	WORD	0		X
MSG_ACK4	Acknowledge status for messages	WORD	0		X
MSG_ACK5	Acknowledge status for messages	WORD	0		X
MSG_STAT	Error Information of ALARM_8P	WORD	0		X
MSG_STAT2	Error Information of ALARM_8P	WORD	0		X
MSG_STAT3	Error Information of ALARM_8P	WORD	0		X
MSG_STAT4	Error Information of ALARM_8P	WORD	0		X
MSG_STAT5	Error Information of ALARM_8P	WORD	0		X
dErrorCode	Error Code of the module (see manual)	DWORD	0		

3.4.7 Messages

ALARM_8P Instance	Message Number	Default Message Text	Message Class
EV_ID	SIG1	Undervoltage in logic supply (PS) 24 VDC	S
EV_ID	SIG2	Overvoltage in logic supply (PS) 24 VDC	S
EV_ID	SIG3	Undervoltage in load supply (PS) 24 VDC	S
EV_ID	SIG4	Shutdown of load supply (PS) 24 VDC	S
EV_ID	SIG5	Overvoltage in load supply (PS) 24 VDC	S
EV_ID	SIG6	Startup parameter rejected by device	S
EV_ID	SIG7	Device address invalid	S
EV_ID	SIG8	Communication to AP master interrupted	S
EV_ID2	SIG1	Communication to AP module interrupted	S
EV_ID2	SIG2	Output process data watchdog of the bus interface expired	S
EV_ID2	SIG3	Input process data watchdog of the bus interface expired	S
EV_ID2	SIG4	AP Master toggle bit error	S
EV_ID2	SIG5	Multiple AP masters detected	S
EV_ID2	SIG6	System startup failed	S
EV_ID2	SIG7	Topology discovery failed	S
EV_ID2	SIG8	State change after OPERATION failed	S
EV_ID3	SIG1	AP device detected invalid	S
EV_ID3	SIG2	AP system startup delayed	S
EV_ID3	SIG3	New AP Module detected	S
EV_ID3	SIG4	Device not ready	S
EV_ID3	SIG5	Device forced into communication loopback	S
EV_ID3	SIG6	Illegal topology	S
EV_ID3	SIG7	Internal AP process data error	S

ALARM_8P Instance	Message Number	Default Message Text	Message Class
EV_ID3	SIG8	Process data configuration error	S
EV_ID4	SIG1	Remanent memory fault	S
EV_ID4	SIG2	General software error	S
EV_ID4	SIG3	Firmware invalid	S
EV_ID4	SIG4	Initialisation failed	S
EV_ID4	SIG5	Unsupported Full Diagnostic Image version	S
EV_ID4	SIG6	Unsupported diagnostic version	S
EV_ID4	SIG7	APDD Invalid	S
EV_ID4	SIG8	Start-up APDD invalid	S
EV_ID5	SIG1	Device inconsistency detected	S
EV_ID5	SIG2	Firmware update operating mode	S
EV_ID5	SIG3	Diagnostic trace reset	S
EV_ID5	SIG4	(not used)	N/A
EV_ID5	SIG5	(not used)	N/A
EV_ID5	SIG6	(not used)	N/A
EV_ID5	SIG7	(not used)	N/A
EV_ID5	SIG8	(not used)	N/A

4 Channel Driver Blocks

4.1 CHAPADI(FC6003)

4.1.1 Description

CHAPADI is used to configure any digital input signal coming from a Festo CPX-AP-A terminal. This includes CPX-AP-A digital input modules. This driver block has been designed for usage in non APL (Advanced Process Library) projects.

CHAPADI processes a digital signal interconnected by the user from the process image to the bPVIn parameter cyclically (depending on the insertion point chosen by the user). The processed result is supplied at the bPVOut output parameter.

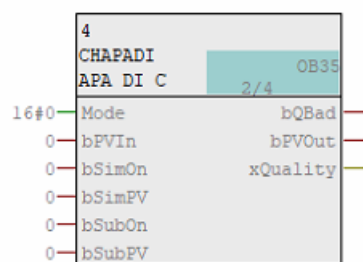
4.1.2 Calling OBs

This block is inserted manually in a convenient CFC. The calling OB depends on the insert point chosen by the user. In most cases, a cyclic interrupt like e. g. OB35 is suitable.

4.1.3 Automatic Configuration with “Generate Module Drivers”

Upon module drivers generation, the CHAPADI function block is automatically interconnected with its respective module driver block (APAMODID). The **Mode** parameter is interconnected with the appropriate **ModeOut** output of the respective module driver block.

4.1.4 Function Block in CFC



4.1.5 Quality Code

The quality of the signal processed is defined by information provided from signal processed internally. From the module driver, the high byte of the value provided at the **Mode** input parameter is evaluated. If the "Generate module Drivers" function is used, this parameter is interconnected automatically with the corresponding **ModeOut** module driver block. The internal processing is based on the setting of the **bSimOn** and **bSubOn** parameters.

The following quality codes are generated by the channel driver block:

Quality Code	Description	Condition
16#60	Simulation Active	1. bSimOn = TRUE
16#80	Valid Value	1. bSimOn = FALSE AND 2. High byte of Mode input = 16#80
16#48	Substituted Value	1. bSimOn = FALSE AND 2. invalid value AND 3. bSubOn = TRUE
16#00	Invalid Value	1. bSimOn = FALSE AND 2. invalid value AND 3. bSubOn = FALSE

The table reflects the priority of the individual quality codes. As can also be seen from the "Condition" column, "Simulation active" overrides all other quality states, while for the "Invalid value" state simulation and value substitution must be deactivated via their respective input parameters.

4.1.6 Operation modes:

1. **Normal Mode:** During normal operation (valid value), the input signal provided at bPVIn is written to the bPVOut output parameter. xQuality is set to 16#80 and bQBad to FALSE.
2. **Simulation Mode:** Simulation can be activated by setting bSimOn = TRUE. The value provided at bSimPV is then written to the bPVOut output parameter, and xQuality is set to 16#60. bQBad is set to FALSE, while bQSim = TRUE indicates the activated simulation mode at the block's interface.
3. **Substitute Value:** If bQSub = TRUE and the value read from the process image is invalid, bQBad = TRUE is set and the value defined at or interconnected to bSubPV is written to the bPVOut output parameter. xQuality is set to 16#48, bQSub = TRUE states that the value currently provided at bPVOut is substituted.
4. **Invalid Value:** If bSubOn is set to FALSE and the value read from the process image (via bPVIn) is invalid, it is nevertheless written to the bPVOut output parameter. bQBad = TRUE indicates the invalid value, and xQuality is set to 16#00.

4.1.7 Interface

Name: Visible in the CFC

Name: Not visible in the CFC (can be changed in Properties)

Name: Output/ IN_OUT

Lock (L): X => Cannot be changed by the user

Set (S): V => Signal is set automatically

C => Signal is connected automatically

X => Signal is changed during the runtime

Name (I/O)	Description	Data type	Default	L	S
Mode	Quality and Mode	DWORD	16#00000000	X	V
bPVIn	Input Value	BOOL	FALSE		
bSimOn	1= activate simulation	BOOL	FALSE		
bSimPV	Simulation Value	BOOL	FALSE		
bSubOn	1= enable failure substitution	BOOL	FALSE		
bSubPV	Substitution value	BOOL	FALSE		
bQBad	1= bad process value	BOOL	FALSE		
bQModErr	1= higher level error	BOOL	FALSE		
bQSim	1= simulation active	BOOL	FALSE		
bQSub	1= failure substitution active	BOOL	FALSE		
bPVOut	Process Value	BOOL	FALSE		
xQuality	Quality code	BYTE	0		

4.2 CHAPADO(FC6004)

4.2.1 Description

CHAPADO is used to configure any digital output signal coming from a Festo CPX-AP-A terminal. This includes CPX-AP-A VTSA/VTUG modules. This driver block has been designed for usage in non APL (Advanced Process Library) projects.

CHAPADO processes a digital signal interconnected by the user from the process image to the bPVIn parameter cyclically (depending on the insertion point chosen by the user). The processed result is supplied at the bPVOut output parameter.

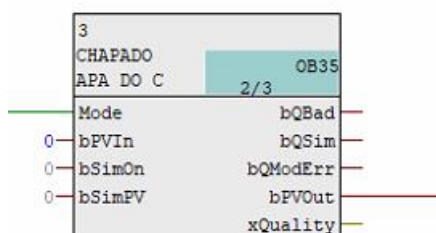
4.2.2 Calling OBs

This block is inserted manually in a convenient CFC. The calling OB depends on the insert point chosen by the user. In most cases, a cyclic interrupt like e. g. OB35 is suitable.

4.2.3 Automatic Configuration with "Generate Module Drivers"

Upon module drivers generation, the CHAPADO function block is automatically interconnected with its respective module driver block (APAVTSA/APAVTUG). The **Mode** parameter is interconnected with the appropriate **ModeOut** output of the respective module driver block.

4.2.4 Function Block in CFC



4.2.5 Quality Code

The quality of the signal processed is defined by information provided from signal processed internally. From the module driver, the high byte of the value provided at the **Mode** input parameter is evaluated. If the "Generate module Drivers" function is used, this parameter is interconnected automatically with the corresponding **ModeOut** module driver block. The internal processing is based on the setting of the **bSimOn** parameter.

The following quality codes are generated by the channel driver block:

Quality Code	Description	Condition
16#60	Simulation Active	1. bSimOn = TRUE
16#80	Valid Value	1. bSimOn = FALSE AND 2. High byte of Mode input = 16#80
16#00	Invalid Value	1. bSimOn = FALSE AND 2. invalid value

4.2.6 Operation modes:

1. **Normal Mode:** During normal operation (valid value), the input signal provided at bPVIn is written to the bPVOut output parameter. xQuality is set to 16#80 and bQBad to FALSE.
2. **Simulation Mode:** Simulation can be activated by setting bSimOn = TRUE. The value provided at bSimPV is then written to the bPVOut output parameter, and xQuality is set to 16#60. bQBad is set to FALSE, while bQSim = TRUE indicates the activated simulation mode at the block's interface.

4.2.7 Interface

Name: Visible in the CFC

Name: Not visible in the CFC (can be changed in Properties)

Name: Output/ IN_OUT

Lock (L): X => Cannot be changed by the user

Set (S): V => Signal is set automatically

C => Signal is connected automatically

X => Signal is changed during the runtime

Name (I/O)	Description	Data type	Default	L	S
Mode	Quality and Mode	DWORD	16#00000000	X	V
bPVIn	Process Value	BOOL	FALSE		
bSimOn	1= activate simulation	BOOL	FALSE		
bSimPV	Simulation Value	BOOL	FALSE		
bQBad	1= bad process value	BOOL	FALSE		
bQModErr	1= higher level error	BOOL	FALSE		
bQSim	1= simulation active	BOOL	FALSE		
bPVOut	Process Value	BOOL	FALSE		
xQuality	Quality code	BYTE	0		

4.3 APADiIn(FC6003)

4.3.1 Description

APADiIn is used to configure any digital input signal coming from a Festo CPX-AP-A terminal. This includes CPX-AP-A digital input modules. This driver block has been designed for usage in APL (Advanced Process Library) projects.

APADiIn processes a digital signal interconnected by the user from the process image to the PV_In parameter cyclically (depending on the insertion point chosen by the user). The processed result is supplied at the PV_Out output parameter.

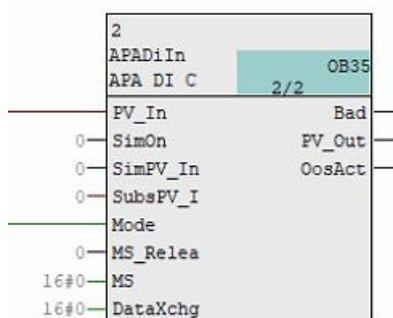
4.3.2 Calling OBs

This block is inserted manually in a convenient CFC. The calling OB depends on the insert point chosen by the user. In most cases, a cyclic interrupt like e. g. OB35 is suitable.

4.3.3 Automatic Configuration with “Generate Module Drivers”

Upon module drivers generation, the CHAPADI function block is automatically interconnected with its respective module driver block (APAMODID). The **Mode** parameter is interconnected with the appropriate **ModeOut** output of the respective module driver block.

4.3.4 Function Block in CFC



4.3.5 Quality Code

The quality of the signal processed is defined by information provided both from module driver externally and signal processed internally. From the module driver, the high byte of the value provided at the Mode input parameter is evaluated. If the "Generate module Drivers" function is used, this parameter is interconnected automatically with the corresponding APAMODx module driver block. An external binary (good/bad) quality signal can be interconnected to the ProImQB input and activated by setting SelQB to TRUE. In this case, both information parts need to show a "good" value (logical AND) for the block to consider the value. The internal processing is based on the setting of the SimOn and Feature.Bit29 (bad value substitution) parameters.

The following quality codes are generated by the channel driver block:

Quality Code	Description	Condition
16#60	Simulation Active or substituted value	1. SimOn = TRUE OR 2. SimOn = FALSE AND 3. Invalid value AND 4. Feature.Bit29 = TRUE
16#80	Valid Value	1. SimOn = FALSE AND 2. High byte of MODE input = 16#80 AND 3. (ProImQB = FALSE or SelQB = FALSE)
16#00	Invalid Value	1. SimOn = FALSE AND 2. invalid value AND 3. Feature.Bit29 = FALSE

The table reflects the priority of the individual quality codes. As can also be seen from the "Condition" column, "Simulation active" overrides all other quality states, while for the "Invalid value" state simulation and value substitution must be deactivated via their respective input parameters.

4.3.6 Operation modes:

- 1. Normal Mode:** During normal operation (valid value), the input signal provided at PV_In is written to the PV_Out output parameter. PV_Out.ST is set to 16#80 and Bad to FALSE.
- 2. Simulation Mode:** Simulation can be activated by setting SimOn = TRUE. The value provided at SimPV_In is then written to the PV_Out output parameter, and PV_Out.ST is set to 16#60. Bad is set to FALSE, while SimAct = TRUE indicates the activated simulation mode at the block's interface.
- 3. Substitute Value:** If Feature.Bit29 = TRUE and the value read from the process image is invalid, Bad = TRUE is set and the value defined at or interconnected to SubsPV_In is written to the PV_Out output parameter. PV_Out.ST is set to 16#60. Feature.Bit28 needs to be set to FALSE for Bit29 to become effective.
- 4. Invalid Value:** If Feature.Bit29 (bad value substitution) is set to FALSE and the value read from the process image (via PV_In) is invalid, it is nevertheless written to the PV_Out output parameter. Bad = TRUE indicates the invalid value, and PV_Out.ST is set to 16#00.

4.3.7 Interface

Name: Visible in the CFC

Name: Not visible in the CFC (can be changed in Properties)

Name: Output/ IN_OUT

Lock (L): X => Cannot be changed by the user

Set (S): V => Signal is set automatically

C => Signal is connected automatically

X => Signal is changed during the runtime

Name (I/O)	Description	Data type	Default	L	S
PV_In	Input Value	BOOL	FALSE		
SimOn	1= activate simulation	STRUCT			
SimPV_In	Simulation Value	STRUCT			
SubsPV_In	Substitution value	BOOL	FALSE		
ProImQB	process image quality bit	BOOL	FALSE		
SelQB	1= use process image quality bit	BOOL	FALSE		
Mode	Quality and Mode	DWORD	16#00000000	X	V
Feature	Status of various features	STRUCT			
Bad	bad process value	STRUCT			
ModErr	higher level failure	STRUCT			
SimAct	simulation active	STRUCT			
OosAct	Field device out of service, maintenance in progress.	STRUCT			
PV_Out	Process Value	STRUCT			

4.4 APADiOu(FC653)

4.4.1 Description

APADiOu is used to configure any digital output signal coming from a Festo CPX-AP-A terminal. This includes CPX-AP-A VTSA/VTUG modules. This driver block has been designed for usage in APL (Advanced Process Library) projects.

APADiOu processes a digital signal interconnected by the user from the user program to the PV_In input parameter cyclically (depending on the insert point chosen by the user). The processed result is supplied at the PV_Out output parameter, which needs to be interconnected to the appropriate periphery signal by the user.

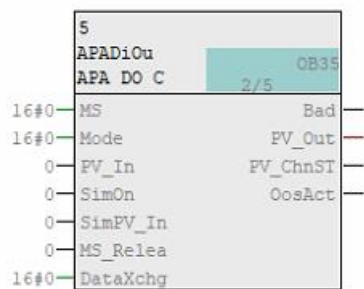
4.4.2 Calling OBs

This block is inserted manually in a convenient CFC. The calling OB depends on the insert point chosen by the user. In most cases, a cyclic interrupt like e. g. OB35 is suitable.

4.4.3 Automatic Configuration with “Generate Module Drivers”

Upon module drivers generation, the APADiOu function block is automatically interconnected with its respective module driver block (APAVTSA/APAVTUG). The **Mode** parameter is interconnected with the appropriate **ModeOut** output of the respective module driver block.

4.4.4 Function Block in CFC



4.4.5 Quality Code

The quality of the signal processed is defined by information provided both from externally and internally.

For the external part, the high byte of the value provided at the Mode input parameter is evaluated. If the "Generate module Drivers" function is used, this parameter is interconnected automatically with the corresponding APAMODx module driver block. The internal part is based on the setting of the SimOn parameter.

The following quality codes are generated by the channel driver block:

Quality Code	Description	Condition
16#60	Simulation Active	SimOn = TRUE
16#80	Valid Value	SimOn = FALSE AND High byte of MODE input = 16#80
16#00	Invalid Value	SimOn = FALSE AND invalid value

The table reflects the priority of the individual quality codes. As can also be seen from the "Condition" column, "Simulation active" overrides all other quality states, while for the "Invalid value" state simulation and value substitution must be deactivated via their respective input parameters.

4.4.6 Operation modes:

- 1. Normal Mode:** During normal operation (valid value), the signal provided at PV_In is written to the PV_Out output parameter. PV_ChnST.ST is set to 16#80 and Bad to FALSE.
- 2. Simulation Mode:** Simulation can be activated by setting the SimOn = TRUE. The value provided at SimPV_In is then written to the PV_Out output parameter, and PV_ChnST.ST is set to 16#60. Bad is set to FALSE, while SimAct = TRUE indicates the activated simulation mode at the block's interface.
If an external simulation is detected via a 16#60 value provided at PV_In.ST and Feature.Bit30 = FALSE, PV_Out is set to FALSE to provoke a powerless state for the digital output and PV_ChnST.ST remains at 16#80. If Feature.Bit30 = TRUE (default), PV_Out is set according to PV_In in spite of the external simulation.
- 3. I/O Fault:** Even if a module error is signaled via the Mode parameter's high byte, the value provided at the PV_In input parameter is always written to the PV_Out output parameter. Bad = TRUE indicates the invalid value, and PV_ChnST.ST is set to 16#00.
- 4. Module Error:** If a higher order module error is detected via a 16#40 high byte in the value provided at Mode, ModErr = TRUE is set.
- 5. Feature Parameter:** APADiOu can be configured using the Feature parameter. Please check the Siemens APL online help for an overview of all reactions which are provided by the Feature parameter. If Bit 30 is set as FALSE, lowest/de-energized value at a block-external simulation is used.

4.4.7 Interface

Name: Visible in the CFC

Name: Not visible in the CFC (can be changed in Properties)

Name: Output/ IN_OUT

Lock (L): X => Cannot be changed by the user

Set (S): V => Signal is set automatically

C => Signal is connected automatically

X => Signal is changed during the runtime

Name (I/O)	Description	Data type	Default	L	S
PV_In	Process Value incl. ST	STRUCT			
SimOn	1= activate simulation	STRUCT			
SimPV_In	Simulation Value	STRUCT			
Mode	Quality and Mode	DWORD	16#00000000	X	V
MS_Release	Maintenance Release	STRUCT			
MS	Maintenance State	DWORD	16#00000000	X	V
Feature	Status of various features	STRUCT			
Bad	bad process value	STRUCT			
ModErr	higher level failure	STRUCT			
SimAct	simulation active	STRUCT			
OosAct	Field device out of service, maintenance in progress.	STRUCT			
PV_Out	Periphery Output Value	STRUCT			
PV_ChnST	State of Output Value	STRUCT			