

**User Manual**  
**Automation and Telecontrol System**  
**@120**  
Version 3.4

#### **License Provisions**

The software described in this user manual is placed at your disposal according to the conditions in the licensing agreement and may only be used under the conditions described therein.

#### **Data, Illustrations, Changes**

Data and illustrations are not binding. We reserve the right to make changes, which provide technical improvements. We welcome your comments to this publication should you wish to suggest any improvements or changes also if you find any mistakes.

#### **Guarantee**

NO GUARANTEE. This technical documentation is delivered WITHOUT express warranty. OHP accepts no guarantee for the accuracy or usage. The use of this technical documentation and the information contained therein is the responsibility of the user.

#### **Training**

To provide complementary system knowledge OHP offers appropriate courses.

#### **Trademarks**

OHP, the OHP Logo and ProWin are registered trademarks of OHP GmbH.

Other product names mentioned in this user manual could be registered trademarks and/or trademarks of the respective companies.

#### **Copyright**

No part of this document may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior permission of OHP GmbH. Translation into a foreign language is forbidden.

Copyright © 2008 OHP GmbH. All rights reserved

Printed in Germany.

---

## Contents

<b>Part I</b>	<b>General.....</b>	<b>7</b>
<b>1</b>	<b>Validity Note .....</b>	<b>7</b>
<b>2</b>	<b>Introduction .....</b>	<b>7</b>
2.1	System-Neutral Automation Platform.....	7
2.2	Migration from AEG/Schneider Automation and Telecontrol System .....	7
<b>3</b>	<b>System Requirements .....</b>	<b>8</b>
3.1	System Requirements - Programming Software.....	8
3.2	System Requirements - Controller .....	9
3.2.1	@120 System Components and I/O Comparison List .....	9
<b>4</b>	<b>Communication Interfaces of the @120 System.....</b>	<b>11</b>
4.1	Communication Procedures on the ALU 320:.....	11
4.2	Communication Procedures with KOS Networking Modules .....	11
4.3	Modem Modules for Telecontrol Procedures .....	11
4.4	Networking via Modbus .....	12
4.4.1	ModbusTCP Client .....	12
4.4.2	Modbus RS232 Slave / Modbus TCP-Server.....	12
<b>PART II</b>	<b>Configuration and Programming.....</b>	<b>13</b>
<b>1</b>	<b>New Features of the ALU 320 Firmware .....</b>	<b>14</b>
<b>2</b>	<b>Creating a New Project with Multiprog .....</b>	<b>18</b>
<b>3</b>	<b>Change the IP Address for Connection to the ALU 320 .....</b>	<b>19</b>
<b>4</b>	<b>Definition of the Inputs and Outputs.....</b>	<b>20</b>
4.1	Inputs and Outputs with the OHP Equipment List Editor .....	20
4.1.1	Definition Standard I/O Module .....	23
4.2	Inputs and Outputs with the Multiprog I/O Configurator.....	28
4.2.1	Definition Standard I/O Module .....	31
4.3	I/O Configuration for IEC 60870-5-101 or IEC 60870-5-104.....	35
4.4	I/O Configuration for Modbus TCP Client.....	37
4.5	Configuration of the Shared-Memory-Area for Modbus RS232-Slave / TCP-Server.....	39
<b>5</b>	<b>System Markers .....</b>	<b>39</b>
5.1	System Markers - Service .....	40
5.2	System Markers - Slots Occupied in the I/O Configuration.....	41
5.3	System Markers – Module Failure .....	42
5.4	System Markers – Module Fault .....	43
5.5	System Markers – Status Bytes .....	44
5.5.1	Definition - Status Bytes KOS201 .....	44

---

5.5.2	Time-of-Day Transfer via KOS 201/202 .....	45
5.5.3	Definition – Status Bytes ADU 206/306/308 .....	46
5.5.4	Definition – Status Bytes ADU 210/214 .....	46
5.5.5	Definition – Status Bytes Modbus .....	46
<b>6</b>	<b>Configuration of the Initialisation Files .....</b>	<b>47</b>
6.1	Settings in the ALU320.INI .....	47
6.1.1	Enabling Device Drivers .....	48
6.1.2	Setting the IP Address on the ALU320 .....	48
6.1.3	Access via FTP or TELNET .....	48
6.1.4	Time Synchronisation via GPS.....	48
6.2	Settings in the ALURAS.INI .....	50
6.2.1	Setting the Parameter Set [AXRASVPN-PARAMS].....	53
6.2.2	Setting of the Parameter Set [AXRASVPN-PARAMS- T2] .....	54
6.2.3	Setting of the Parameter Set [DIAL-LOCATION] .....	55
6.2.4	Setting of the Parameter Set [RASMODEM].....	55
6.2.5	Setting of the Parameter Set [ISP-PPP-DIAL] .....	56
6.2.6	Setting of the Parameter Set [ISP-PPP-TCPIP].....	56
6.2.7	Setting of the Parameter Set [ISP-PPP-SECURITY] .....	57
6.2.8	Setting of the Parameter Set [ISP-PPP].....	57
6.2.9	Setting of the Parameter Set [ISP-PPP-PARAMS] .....	57
6.2.10	Setting of the Parameter Set [VPN-PPTP].....	59
6.2.11	Setting of the Parameter Set [VPN-PPTP-TCPIP].....	59
6.2.12	Setting of the Parameter Set [VPN-PPTP-SECURITY] .....	60
6.2.13	Setting of the Parameter Set [VPN-PPPT-PARAMS] .....	60
6.2.14	Setting of the Parameter Set [UNIMODEM-INIT].....	61
6.2.15	Configuration Example .....	61
6.3	Enable the Service Routines and Interfaces .....	62
6.3.1	Definition von Shared Memory .....	63
6.3.2	Modbus RS232.....	63
6.3.3	Modbus TCP Server.....	64
6.3.4	Service Routines .....	64
6.4	IEC 60870-5-101 or -104 Communication .....	65
6.4.1	General.....	65
6.4.2	Parameter File for Communication Configuration IEC 60870-5-104 (SVC_COMn) .....	65
6.4.3	Parameter File for the Communication Configuration IEC 60870-5-101 (SVC_COMn.INI) .....	66
6.4.3.1	Explanation of the Individual Parameters in Leased Line Operation .....	67
6.4.3.2	Explanation of the Individual Parameters in ADS Operation .....	68
6.4.4	Parameter File for Processing the Data Point Definition (SVC_RTUn.INI).....	69
6.5	Configuration of the Initialisation File for Modbus RS232-Slave or TCP-Server .....	77
6.5.1	Explanation of the Individual Parameters.....	78

---

---

<b>7</b>	<b>Firmware Library.....</b>	<b>81</b>
7.1	General.....	81
7.2	Library ALUX20_OHP4_SYS.....	82
7.2.1	Reading and Setting the ALU320 System Time in the User Program.....	82
7.2.2	Status of the ALU320 Read System Time.....	83
7.2.3	Status of the ALU320 Set the System Time to Invalid.....	84
7.2.4	Status of the ALU320 System Time with GPS Synchronisation.....	84
7.2.5	Status of the ALU320 System Time with SNTP Synchronisation.....	85
7.2.6	System Clock.....	85
7.2.7	Status of the GPRS Connection.....	86
7.2.8	System Tick.....	87
7.2.9	Modbus Server.....	87
<b>8</b>	<b>Tips and Tricks.....</b>	<b>88</b>
8.1	Updating an Existing Multiprog Project for Online Change.....	88
8.2	Set the Time and Diagnosis via Telnet.....	88
8.3	ALU Status Information via Telnet.....	90
<b>Part III</b>	<b>Module Description.....</b>	<b>93</b>
<b>1</b>	<b>Mounting Dimensions.....</b>	<b>94</b>
<b>2</b>	<b>Module Description ALU 320.....</b>	<b>95</b>
2.1	Interfaces.....	95
2.1.1	Interface Allocation of the Screw Terminals:.....	96
2.1.1.1	Power Supply.....	96
2.1.1.2	Interface for an Optional GPS Time-of-Day Receiver.....	96
2.1.1.3	CAN Interface (in Preparation).....	96
2.1.2	Serial RS232 Interface:.....	97
2.1.3	RS485 Interface on theALU320-485:.....	97
2.1.4	Compact Flash Memory Cards.....	97
2.1.5	ALU 320 Display Elements.....	98
2.1.6	ALU320 Operator Elements.....	99
2.2	Technical Data ALU 320.....	100



---

## Part I General

### 1 Validity Note

This user manual applies for the ALU 320x from firmware version 3.16 and MP@Plus version 2.8.

The current firmware version is available for download in the customer area at [www.ohp.de](http://www.ohp.de)

It is possible that some features will not be supported by older firmware versions. You can find a breakdown of the firmware versions and their new associated features in **Part II Configuration and Programming**, chapter 1.

### 2 Introduction

The automation and telecontrol system @120 is understood as a further development of the proven A120/Compact automation and telecontrol equipment and secures the future availability of the A120/Compact range as a powerful automation platform.

The automation and telecontrol system @120 can, apart from equipping new plant, be used for migration purposes thereby securing your investment on existing plant, because the @120 system provides a seamless transition using the I/O periphery of the A120/Compact range.

#### 2.1 System-Neutral Automation Platform

The @120 system has been conceived as a system neutral automation platform, which has where possible dispensed with the use of manufacturer-specific solutions. Based on the most modern embedded industrial PC technology together with the standard real-time operating system Windows CE it supports all standard interfaces e.g. Ethernet, RS232, USB, VGA, and CF card.

Using the manufacturer independent IEC 61131-3 programming software MULTIPROG we have implemented a powerful and mature programming tool, which is been successfully used by more than 70 renowned PLC companies.

By means of the standardised communication protocols such as IEC 60870-5-101/104, or the well-known Modbus TCP protocol and also the optional OPC server the system can be integrated into a wealth of automation and telecontrol environments.

Using the integrated Ethernet or CAN interfaces, I/O expansion can be connected using for example the @120 system's own DEA 300 or proprietary I/O systems e.g. from Phoenix Contact. Via these I/O expansions up to 1216 I/O points can be processed.

#### 2.2 Migration from AEG/Schneider Automation and Telecontrol System

The A120/Compact units provide a proven automation and telecontrol platform, in which considerable time and financial investment has made. Using the @120, OHP has developed a strategy whereby at minimal expenditure in cost and time the units can be migrated into a new automation and telecontrol system, which has a long-term future availability.

The attractive advantage of this strategy is the cost saving by using the existing periphery such as I/O modules and KOS modules and retaining their existing wiring. Only exchanges that are absolutely necessary must be made. Normally this is only the ALU and the primary rack (DTA200 -> DTA300).

By using these measures the new standardised IEC 60870-5-101/-104 procedures and a modern IEC 61131-3 programming language become available. By continued use of existing KOS modules an elegant upgrade strategy from 1/F and 1/W procedures to standardised IEC procedures can be achieved.

---

### 3 System Requirements

#### 3.1 System Requirements - Programming Software

##### Hardware

For the optimum performance when using the programming system Multiprog, the following hardware requirements must be met:

Unit/Module	Minimum	Recommendation
IBM compatible PC with Pentium processor	Pentium II 350 MHz	Pentium III 500 MHz
User memory	64 MB	128 MB
Hard disk	250 MB free memory	
CD-ROM Drive	Necessary	
VGA Monitor colour setting Resolution	256 colours 800 x 600	True Colour 1024 x 768
Ethernet interface	Necessary	

##### Software

MP@Plus requires one of the following operating systems:

- Microsoft Windows 2000
- Microsoft Windows XP

### 3.2 System Requirements - Controller

- As primary rack the DTA 300 must be used.  
When upgrading an existing A120/Compact station, then the DTA 200 must be exchanged.
- The first slot is exclusively reserved for the ALU 320.
- As expansion rack a DTA 301, DTA 201 or DTA 202 can be used.
- The standard I/O modules of the A120/Compact are supported (see table).
- The OHP I/O modules of the 300-series can be used (see table).
- Via ModbusTCP Client external I/O components and system extensions can be connected e.g. from Phoenix Contact, ABB, etc.

#### 3.2.1 @120 System Components and I/O Comparison List

The following list includes the system components for the automation and telecontrol system @120:

In particular are the I/O modules, which have terminal compatibility to the Schneider I/O modules.

Not all the OHP I/O modules included in the table are currently available. For these I/O modules OHP will deliver a comparative I/O module from Schneider Electric. I/O modules, which do not appear in the list, can be eventually delivered, respectively integrated on request.

Designation	Schneider Comparison	Description	
		<b>CPU, Racks</b>	
ALU320		CPU for @120 system	
DTA300		Primary rack for CPU and 4 I/O modules	
DTA301	DTA201	Expansion rack for 5 I/O modules	
MP@Plus		Multiprog IEC 61131-3 programming software for the @120 System. IL, FBD, LD, ST and SFC. Cross translation IL-LD-FBD, Offline simulation, online change, watch window, debugging functions, project comparison, cross-reference.	
		<b>Digital Inputs</b>	<b>Input Voltage</b>
DEP314	DEP214	16 Digital inputs	12V to 60V DC
DEP316	DEP216	16 Digital inputs	24V DC
DEP332		32 Digital inputs	24V DC
		<b>Digital Outputs</b>	<b>Output Voltage</b>
DAP316	DAP216N	16 Digital outputs	24V DC
DAP332		32 Digital outputs (hardware in preparation)	24V DC

DAP308	DAP208	8 Relay outputs	24...110V DC or 24...230V AC
		<b>Digital Combination Modules</b>	<b>Input Voltage</b> <b>Output Voltage</b>
DAP312	DAP212	8 Digital Inputs 4 Relay outputs	24V DC 24...110V DC or 24...230V AC
DAP320	DAP220	8 Digital inputs 8 Digital outputs	24V DC 20...30V DC
		<b>Analogue Inputs</b>	<b>Input Signal</b>
ADU304	ADU204	4 Analogue inputs, 12Bit + VZ	+/- 500mV, PT100
ADU306	ADU206	4 Analogue inputs, potentially isolated, 11Bit + sign	+/- 10V, +/- 20mA
ADU308		8 Analogue inputs, 12Bit + VZ	+/- 10V, +/- 20mA
ADU314	ADU 214	8 Analogue inputs, 15Bit + VZ	+/- 10V, +/- 20mA temperature and resistance measure- ment
		<b>Analogue Outputs</b>	<b>Output Signal</b>
DAU302	DAU202	2 Analogue outputs, potentially isolated, 11Bit + sign	+/- 10V, +/- 20mA
DAU304	DAU204	4 Analogue outputs, potentially isolated, 11Bit + sign	+/- 1,5V, 10V, 20mA, 4...20mA
DAU308	DAU208	8 Analogue outputs, potentially isolated, 11Bit + sign	+/- 10V
		<b>Interface Modules</b>	<b>Interface</b>
KOS202A		Networking module for telecontrol protocol Modnet 1/F.	1 x RS232
KOS203		Networking module for telecontrol protocol Modnet 1/W.	1 x RS232
KOS260		Networking module for telecontrol protocol IEC 60870-5-101	2 x RS232

---

## 4 Communication Interfaces of the @120 System

The new automation and telecontrol system @120 includes networking interfaces, which can either be operated directly via the ALU 320 or by using a KOS interface module.

### 4.1 Communication Procedures on the ALU 320:

- IEC 60870-5-101
- IEC 60870-5-101      **ADS** for **A**utomatic **d**ial **s**ervice; German: **AWD** automatische **W**ähldienst
- IEC 60870-5-104
- ModbusTCP Client
- Modbus RS232 Slave

The IEC procedures correspond to the compatibility list @120 for IEC 60870-5-101 and IEC 60870-5-104.

### 4.2 Communication Procedures with KOS Networking Modules

- Modnet 1/F (SEAB-1F)      -      KOS 202A
- Modnet 1/W      -      KOS 203
- Modnet 1/W AWD      -      KOS 203
- IEC 60870-5-101      -      KOS 260
- IEC 60870-5-101 ACE      -      KOS 260



**Note:** Existing KOS 201 stations will be supported.

### 4.3 Modem Modules for Telecontrol Procedures

The following modem connections are available for the @120 system when using the communication procedures IEC 60870-5-101 – AWD and IEC 60870-5-104:

- UEM001 - FSK dedicated line modem, 600/1200 Baud, CCITT grid.  
Insertable pcb for KOS 202, KOS 203 or KOS 260
- UEM201 - FSK leased line modem, such as UEM 001, however in a housing for mounting on the DTA300/301
- UEM202 - FSK leased line modem, such as UEM 001, however for DIN-rail mounting
- UEM300 / UEM301 – analogue modem for switched line operation
- UEM302 – ISDN terminal for switched line operation
- UEM303 / UEM 304 - Dual band GSM Modem
- UEM306 / UEM 307 - GPRS Modem

---

## 4.4 Networking via Modbus

### 4.4.1 ModbusTCP Client

The ALU 320 is equipped with a ModbusTCP Client interface. Other devices can be connected via their external I/O modules or controllers, which have a ModbusTCP Server interface.

**Remark:** The Client functionality is equivalent to a Master in a serial Modbus connection.



For additional information see **Part II Configuration and Programming**, chapter 4.4.

### 4.4.2 Modbus RS232 Slave / Modbus TCP-Server

The two serial interfaces of the ALU 320 can be configured as Modbus slave interfaces. The Modbus TCP server is connected via the Ethernet interface.

Via these interfaces any number of Modbus master / Modbus clients can request data from the station. Except for the selection of the interface (SERIAL or TCP\_IP) the configuration is identical.

The ALU 320 supports the following function codes:

FC 1	read coils
FC 2	read discrete inputs
FC 3	read holding register
FC 4	read input register
FC 5	write single coil
FC 6	write single register
FC 15	write multiple coils
FC 16	write multiple register
FC 23	read/write multiple register

The data will be made available in the shared memory area. The configuration is made using INI files, which are stored on the CF card.

**Beware:** The FC 2 and FC 4 functions do not directly affect the inputs of the hardware. The inputs must also be depicted here in the shared memory area.

**Beware:** The Modbus TCP server supports a maximum of 8 client connections. It should be noted here that the Modbus client of an @120 or @250 each creates one connection for reading and one for writing data. The number of the transactions (simultaneously transmitted inquires in asynchronous operation) is limited to 4.



For additional information see **Part II Configuration and Programming**, chapters 4.5, 6.3 and 6.5.

## **PART II Configuration and Programming**

The configuration and programming will be explained using an example project, which will be created using the presentation "ALU 320\_KOS202".

This presentation can be downloaded from the OHP Automation Systems GmbH website.

The presentation includes the following examples:

- The I/O configuration for an input module DEP 316 and an output module DAP 316
- Data type declaration of a 128 Byte arrays, that can be used for the KOS 201/202
- A function block for creating a rotating 16 Bit counter value for digital inputs
- A program "Main" in FBD, in which the module failure registers on a DEP 316 and DAP 316, are combined into a group fault.
- The global variables for status information of the PAB slots

---

## 1 New Features of the ALU 320 Firmware

The current firmware version can be determined via the Multiprog menu "Online -> Project Control -> Info".



**Note:** Because the drivers for the ALU 320 and the ALU 5xx are partially compiled from the same source program, the versions index can increase, also when the respective ALU driver is not affected by these changes. For this reason a version jump can occur.

### Version 1.40

An additional functionality has been included in the IEC 60870-5-101 and ...104 drivers. Initial values can be defined for set values and bit strings in the command direction. These initial values can be processed in two ways:

Modus 1:

During a cold restart the configured initial value written once to the corresponding %IW respectively %ID variable. As soon as a telegram is received from the host system, the value in the telegram will be written to the corresponding variable until a new value is received from the host.

Modus 2:

During a cold restart the configured initial value will be written to the corresponding %IW respectively %ID variable.

If a telegram is received from the host system, the values in the telegram will be written to the corresponding %IW respectively %ID variables and then subsequently overwritten by the configured default value. According to the length of the user program, the telegram value can be present for 1 ... 5 program cycles before it is overwritten by the initial value.

In this way the user program realises if a new telegram has been sent from the host, also when the value in the telegram has not changed.

Likewise a programmable selection between Modbus 1 and 2 can be made.

Programming example:

```
//: Bitstring C_BO_NA
1:VAR_CBO_NA_INIT= 0x0000FFFF
1:VAR_CBO_NA_CMD= TRUE // TRUE = Mode 2, initial value after each telegram delivery.
```

```
//: Sollwert C_SE_NA
1:VAR_CSE_NA_INIT= 0xFFFF
1:VAR_CSE_NA_CMD= FALSE // FALSE = Mode 1, initial value only on cold restart.
```

### Version 1.54

Time of day synchronisation via GPS, IEC...101/104 time telegram and user program.

New parameter for SVC\_RTUn.INI

*RTU\_TIME\_SET*: 0, 1 or 2.

0 = time can be set via IEC (default).

1 = time can never be set via IEC.

2 = time can only be set via IEC, when the current time IV (IV = invalid) is invalid.

The following parameter is only valid when *RTU\_TIME\_SET* is unequal to NULL!

*RTU\_TIME\_ACK\_NEG*: FALSE or TRUE.

FALSE = IEC time telegram is confirmed with a positive response (default).

---

TRUE = IEC time telegram will be confirmed with a negative response.

The following parameter is only valid when RTU\_TIME\_SET = 2!

*RTU\_TIME\_SYNC\_ENABLE*: input in minutes (1...65535).

Delta time entry in minutes since the last clock setting. After expiry the clock can be set via an IEC telegram.

**Version 1.61**

When an ADU is configured but not inserted; a measured value of "0" is displayed (earlier 32767).

**Version 1.62**

Driver for ADU204/205 modules modified.

**Version 1.70**

The name of the INI file has been changed, so that it can be created with the new user interface via Multiprog.

Old name	New name from version 1.70
PCOS_A320.ini	PCOSA320.ini
Slot_Device.ini	SlotDev.ini
SVC_COMn.ini *)	SVCCOMn.ini *)
SVC_RTUn.ini *)	SVCRTUn.ini *)
Modbus_Server.ini	MBSRV.ini

\*) n stands for the corresponding service routine (1...4)

**Beware:** When a new firmware from version 1.70 onwards is used and you want to use an INI file from an older firmware version, then you must rename it before saving to the CF card. The content of the files is identical.

**Version 1.74**

Now both COM interfaces can be used simultaneously for IEC 60870-5-101 communication.

Now two service routines can be defined for communication using IEC 60870-5-104.



For additional information see Part II **Configuration and Programming**, Chapter 6.4.2.

**Version 1.80**

Firmware block for RS232 (SCOM) communication introduced.

With block telegrams IEC...101/104 now only 127 objects of type single or double message will be blocked.

Limits for the transfer interface extended.

IEC101

Maximum 2096 objects in the monitoring and command directions in the SVCRTUn.INI file.

Maximum of 512 bytes in the monitoring direction und 1024 bytes in the command direction for the I/O configuration.

IEC104

Maximum 6000 objects for service 1 and 2 and max. 2096 objects for all further service routines in the SVCRTUn.INI file.

Maximum of 512 bytes in the monitoring direction und 1024 bytes in the command direction for the I/O configuration.

---

### Version 1.81 - 1.86

Incorporation of the driver for the integrated GPRS modem and diverse modifications.  
Configuration of the interface via the ALURAS.ini file.

Version 1.81: Summer/winter time management per time telegram IEC104/101 revised.

### Version 1.87

- IEC101 AWD (Automatic Dialup Service) operation: Routine for sending the passwords changed. The characters will now be sent quicker in succession.
- Modbus TCP for more request-enables.
- System shared memory extended to max. 32KByte!
- Modbus shared memory extended to max. 32KByte!
- PAB520.dLL - BIK114 driver for SEAB extended.
- SSDKOSBIKA.dll and CDLSSDKOSBIKA.dll for SEAB-FWB extended.
- SEAB firmware library for Multiprog and ProConOS included.

### Version 1.89

- SSDKOSBIKA.dll and CDLSSDKOSBIKA.dll for SEAB-FWB extended.
- IEC101/104: During under voltage, ring buffer data will be written to the CF card and during warm restart reloaded.
- IEC101/104: Handling of the OV bit (over range) for measured values revised.
- IEC101: Object No. now extended to 3 octets.

### Version 1.90

- IEC101/104 ADU Measured value handling changed to 2 complement and qualifier-OV-trigger transmission revised.
- IEC101 AWD (Automatic Dialup Service) operation:
  - AT\_INIT string extended to 28 characters.
  - 2 further AT\_INIT strings for SVCCOMn.ini allowed(AT\_INIT2 and AT\_INIT3)
- SEAB-FWB - Start/STOP problem after AWP download solved.

### Version 1.91

- A520/A320 Modbus Server TCP/IP communication for multiple transactions optimised.
- IEC101 AWD LED control integrated:  
The function is only designed for one internal MODEM (COM3).  
Setting in SVCCOM?.INI under section AWD101-MODE.  
MODEM\_LED=TRUE/FALSE.

Impact by TRUE:

LED ON - Data connection ("CARRIER") available - Password (Rx/Tx) and IEC communication.

LED OFF – no connection

LED blinks(slow) – connection establishment UST->LR.

LED blinks(fast) - connection establishment LR (Ring)->UST.

LED blinks twice (500ms) then pause ... Modem Initialisation problem.

LED blinks 3 times (300ms) then pause ... Modem no longer ready.

- With IEC101-AWD (Automatic Dialup Service) complete the sending of the password with "CR" (default: TRUE)  
Deactivate this functionality under section AWD101-MODE with DIAL\_PW\_CR=FALSE

### Version 1.94

- VPN - GPRS modem HW-RESET for internal modem implemented.  
This functionality is now implemented as a standard setting and can be switched off in ALURAS.ini section AXRASVPN-PARAMS as follows: HwReset3Att=FALSE
- With Telnet connection – Welcome text with version identification implemented.

---

**Version 3.00** (BETA project version)

- Changeover to PROCONOS4 based on version 1.94.  
(Requirement for online change)
- BIK unit driver for A520/A320 modified

**Version 3.10** (BETA project version)

- Codebasis for SEAB and DEA operational modes from A520/A320 adapted.
- AWD (Automatic Dial-up Service) operation:  
Procedure InitString with Motorola modem corrected.  
Delay time for redialling corrected.

**Version 3.12** (BETA project version)

- Analysis of the ALUX20.ini for SNTP parameter "Server2" amended.  
Important: The parameter "SERVER2" must always be entered after "SERVER"!
- Date/time monitoring for SNTP service extended.
- Diag\_TSync.exe  
Diagnosis for date/time service SNTP client (NTP0) etc. SNTP - 2 Server address enable.
- SNTP service (client).  
Second server address extended. The server address extended to IP address base (xxx.xxx.xxx.xxx),  
previously only the host name was possible.
- UDPCFG server.  
Actualisation of the TCP/IP address. Time setting using broadcast allowed!

**Version 3.16**

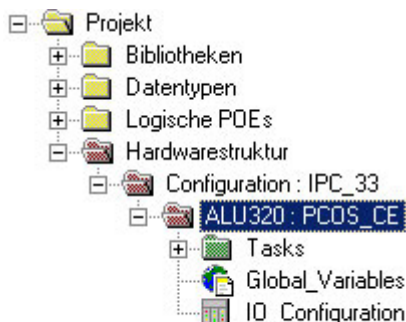
- BIK DEA connection due to BKF (Interbus-S) module modified.
- Official release version of the services from 3.0

## 2 Creating a New Project with Multiprog

When creating a new project proceed as follows:

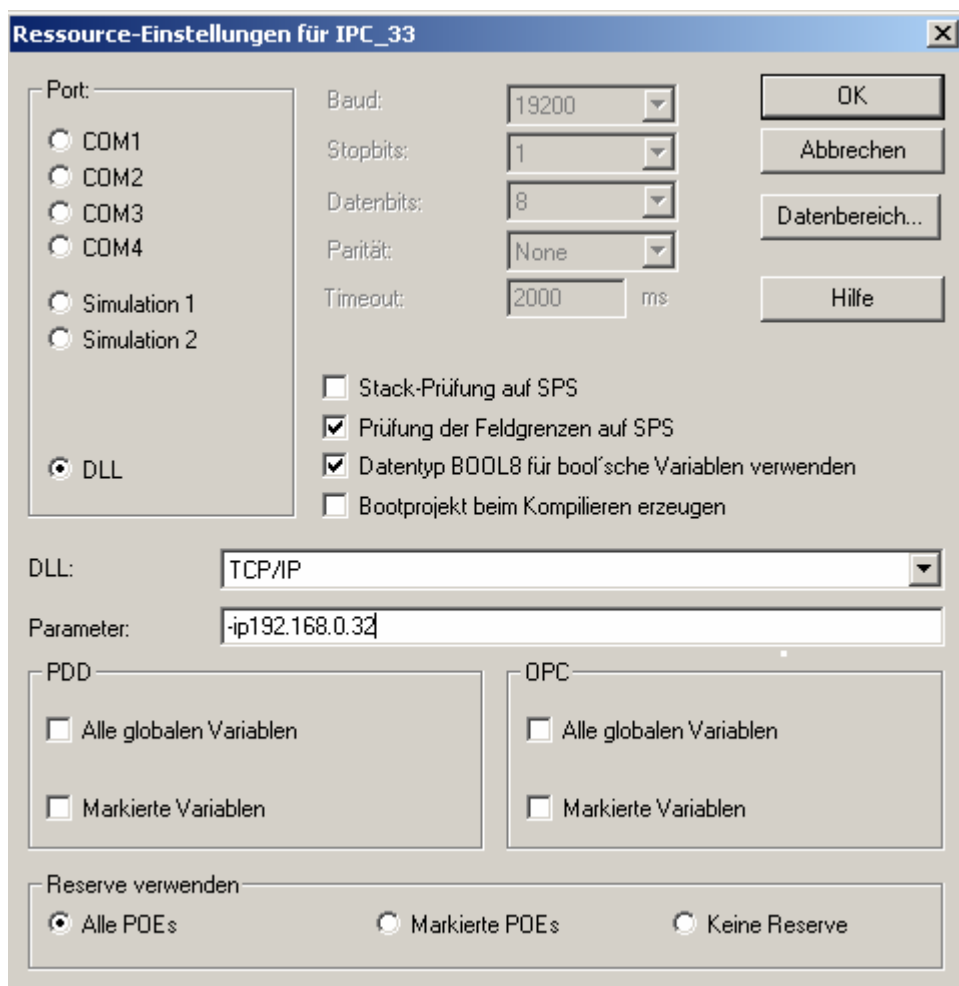
- Step 1 In the File menu select "New Project..."
- Step 2 Double click the presentation "ALU 320\_KOS202".
- Step 3 If necessary, change the IP-address for connecting to the ALU 320 (Part II, chapter 3).
- Step 4 Save the project with its project name using the File menu "Save project as...".
- Step 5 Compile the new project (F9 or in Build -> Make). By this means the data types and function blocks will be made available in the respective windows.
- Step 6 Proceed with the I/O definition (Part II, chapter 4).
- Step 7 Create a Logical Program Unit (POE).
- Step 8 Enter the POEs in the task execution list.
- Step 9 Generate the program code (F9 or in Build -> Make).
- Step 10 Load and start the program code in the ALU 320 using "Online -> Project control".

### 3 Change the IP Address for Connection to the ALU 320



Select the resource, in the example ALU 320:PCOS\_CE in the hardware configuration, using the right-hand mouse button. Using the pop-up-window select the "Settings..." dialogue.

Change the IP-Address in the Parameter box.

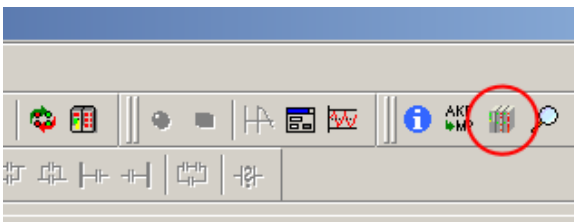


## 4 Definition of the Inputs and Outputs

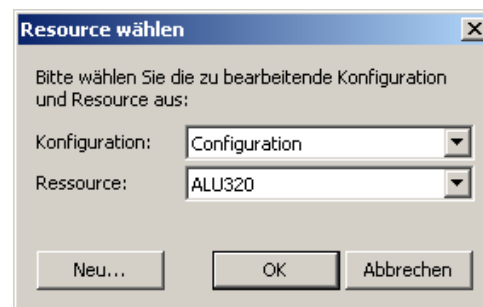
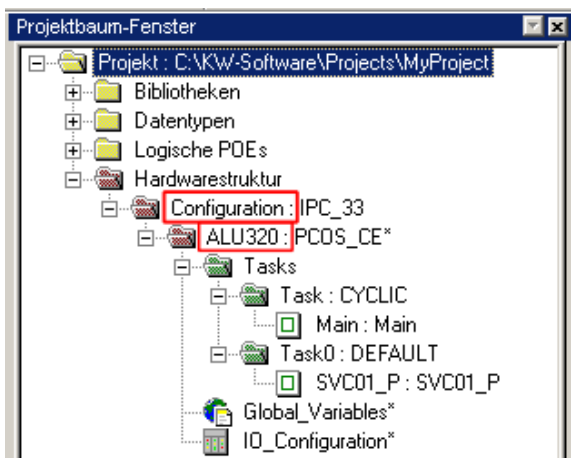
### 4.1 Inputs and Outputs with the OHP Equipment List Editor

Provided you have installed Multiprog from the OHP installation CD including the OHP add-ons, then the new PLC Configurator is available. If this is not the case, please proceed using the description in Part II, Chapter 4.2

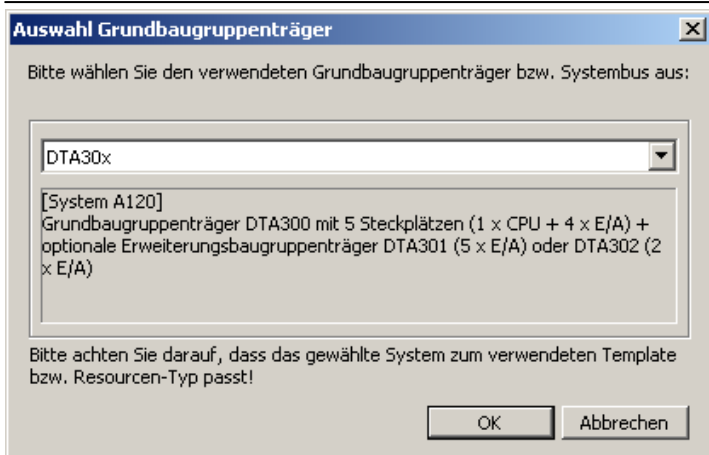
With the help of the new PLC Configuration editor the user can quickly and conveniently create the I/O configuration.



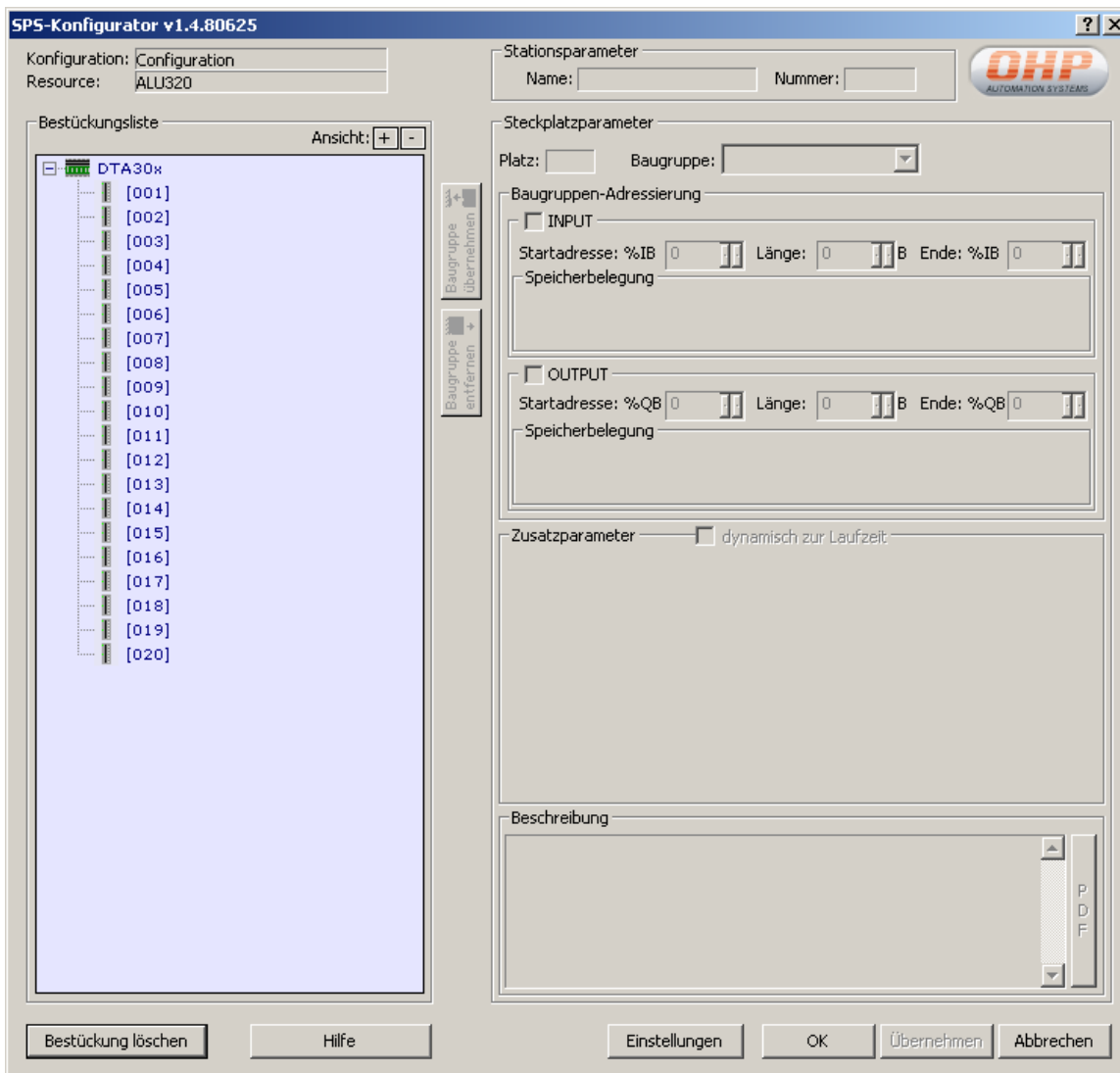
One click on the selection button in the menu bar and the “PLC Configuration” dialogue will be opened for the PLC configuration. Because the PLC is represented in the Multiprog project tree by one configuration node and one subordinate node, the selection is carried out under the corresponding designated node.

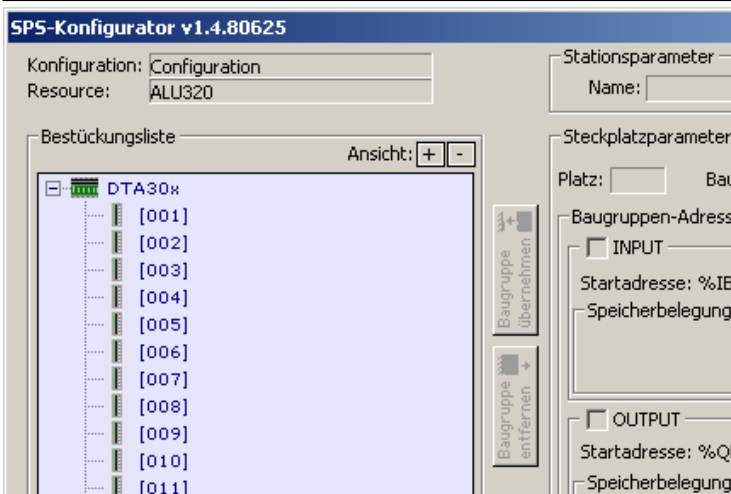


After selecting the configuration and the resource and clicking the „OK“ button the existing equipment list will be opened. When no equipment list exists, then the dialogue “select basic rack” will appear.

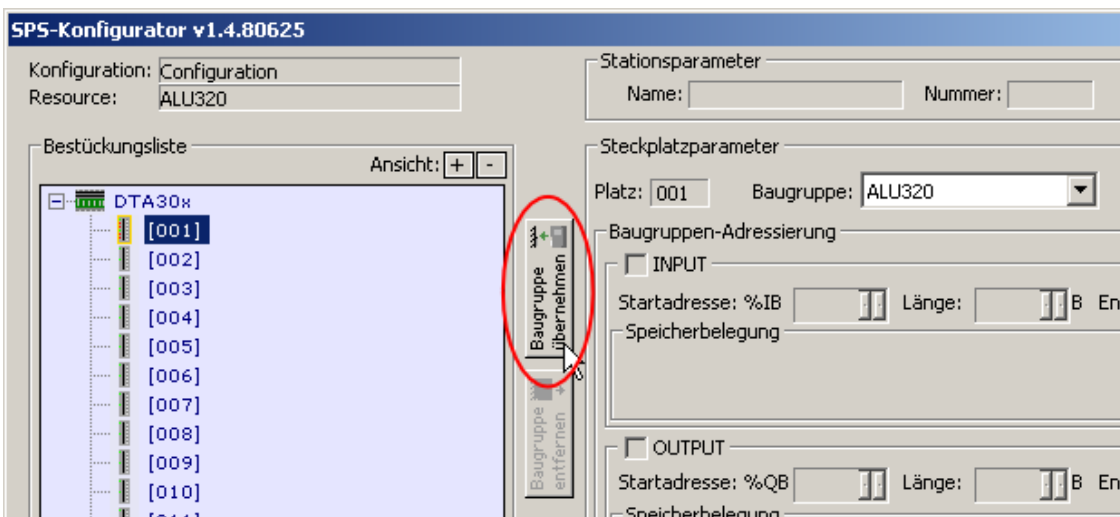


For the ALU320 resource the DTA30x basic rack will be automatically offered and after confirming by clicking on the “OK” button it will be entered in the equipment list.





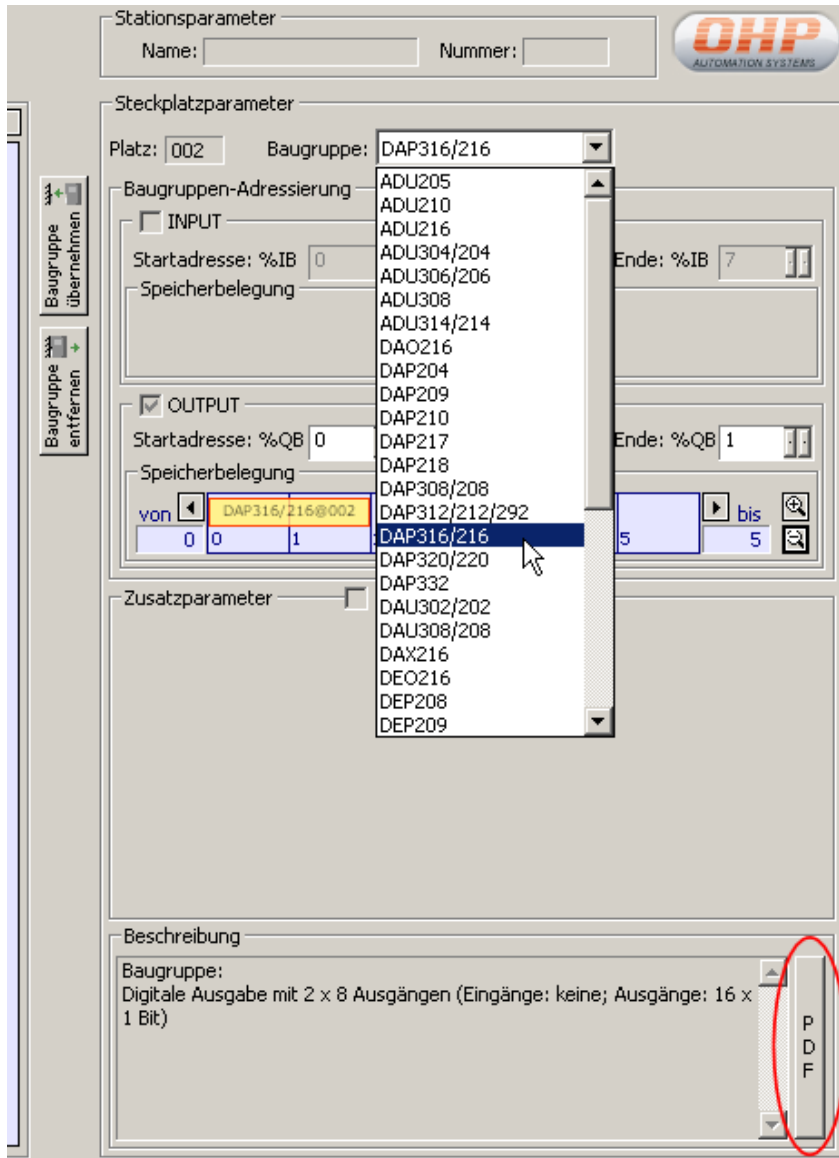
In the first slot (SP:001) of the DTA300 only the ALU320 can be inserted, which is automatically entered in the module field.



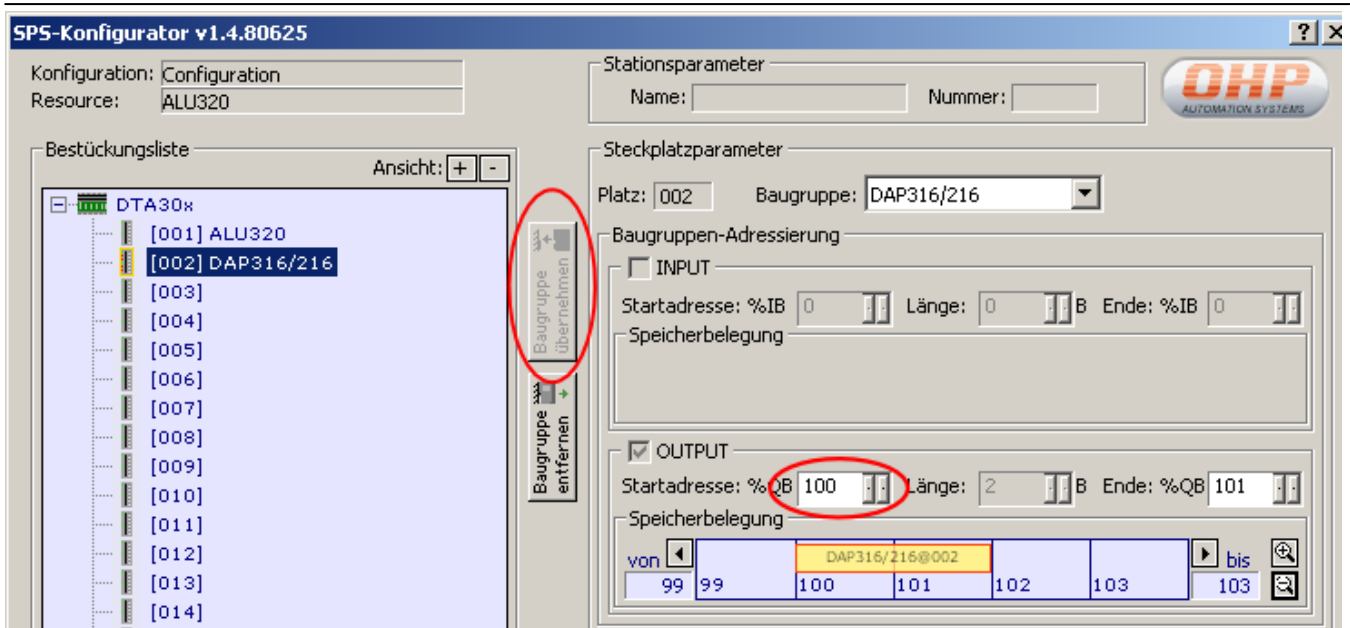
After clicking on “Accept Module” the ALU320 will be entered in the slot [001] and then the next slot can be selected using the mouse.

#### 4.1.1 Definition Standard I/O Module

Digital output module:



The desired module must be selected (here a digital output module with 2 \* 8 outputs – DAP216 or DAP316). The description field includes a short description of the selected module. By clicking the “PDF” button a detailed description of the OHP modules is loaded. For this, the “Acrobat Reader” program (at least version 5.0) must be available on the programming unit.



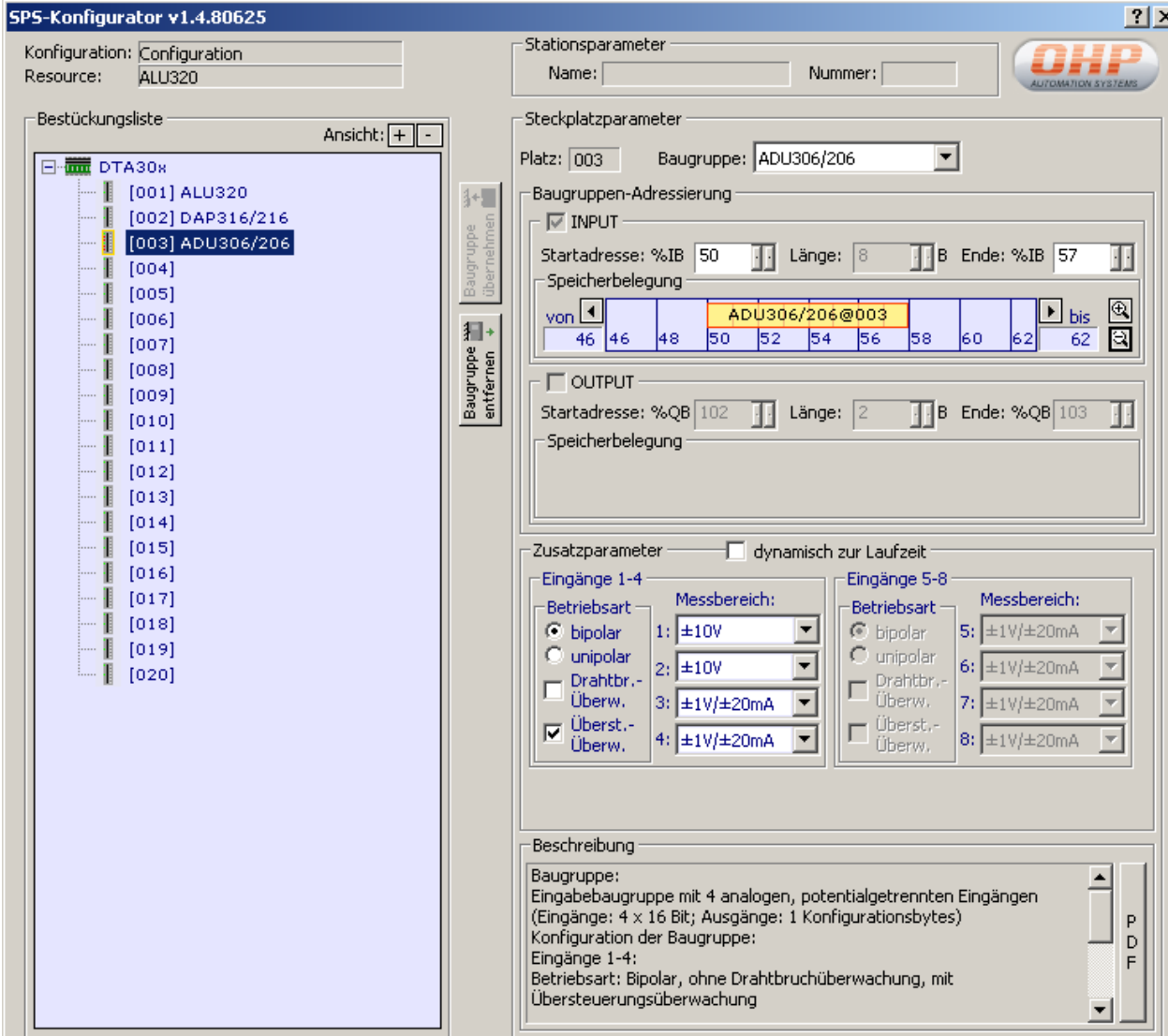
The memory area for the output module must be defined in the “Start Address” field, in the example the value 100 defines where the required 2 bytes (2 \* 8 bits) for the output module begin i.e. at output byte 100 (%QB100) and go to output byte 101 (%QB101). The occupied memory area is also marked in the “Memory Occupancy” display window with a yellow bar, which covers the used address bytes.

After clicking on “Accept Module” the module will be entered in the slot 2 of the rack.



**Note:** A subsequent change e.g. to the start address must always be confirmed by clicking on the “Accept Module” button before the change is realised.

Analogue input module:

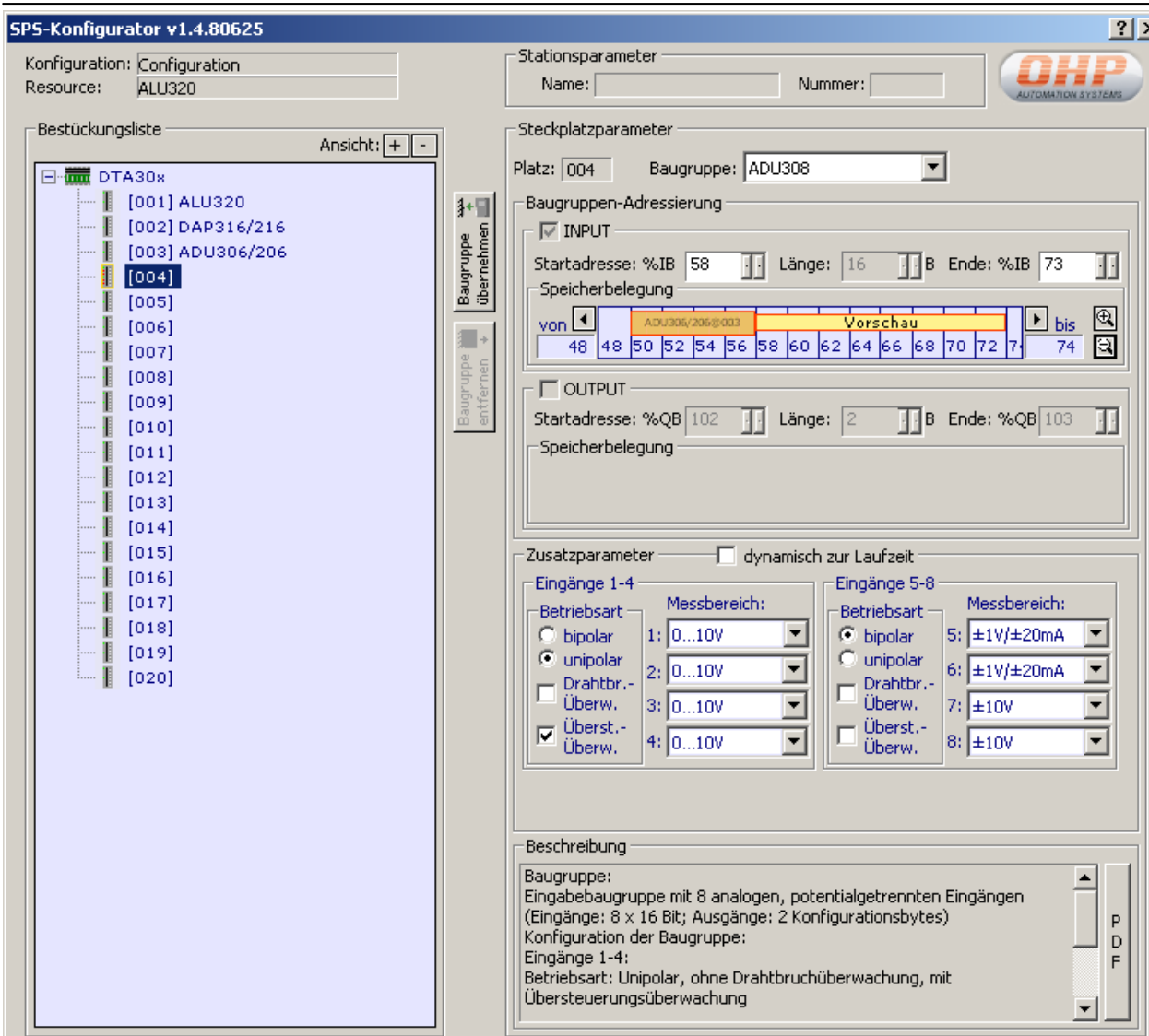


The screenshot shows the SPS-Konfigurator v1.4.80625 software interface. On the left, a 'Bestückungsliste' (Bill of Materials) lists components: DTA30x, [001] ALU320, [002] DAP316/216, [003] ADU306/206 (highlighted), and [004] through [020]. The main configuration area is divided into several sections:

- Konfiguration:** Configuration, Resource: ALU320
- Stationsparameter:** Name: , Nummer:
- Steckplatzparameter:** Platz: 003, Baugruppe: ADU306/206
- Baugruppen-Adressierung:**
  - INPUT: Startadresse: %IB 50, Länge: 8, B Ende: %IB 57
  - Speicherbelegung: von 46 to 62, with ADU306/206@003 highlighted in the 50-57 range.
  - OUTPUT: Startadresse: %QB 102, Länge: 2, B Ende: %QB 103
- Zusatzparameter:**  dynamisch zur Laufzeit
- Eingänge 1-4:**
  - Betriebsart: bipolar (selected), unipolar, Drahtbr.-Überw., Überst.-Überw. (checked)
  - Messbereich: 1: ±10V, 2: ±10V, 3: ±1V/±20mA, 4: ±1V/±20mA
- Eingänge 5-8:**
  - Betriebsart: bipolar (selected), unipolar, Drahtbr.-Überw., Überst.-Überw.
  - Messbereich: 5: ±1V/±20mA, 6: ±1V/±20mA, 7: ±1V/±20mA, 8: ±1V/±20mA
- Beschreibung:** Baugruppe: Eingabebaugruppe mit 4 analogen, potentialgetrennten Eingängen (Eingänge: 4 x 16 Bit; Ausgänge: 1 Konfigurationsbytes) Konfiguration der Baugruppe: Eingänge 1-4: Betriebsart: Bipolar, ohne Drahtbruchüberwachung, mit Übersteuerungsüberwachung

### ADU306

The example describes the insertion of an ADU206/306 analogue module with 4 potentially isolated analogue inputs in slot 3. It requires 8 bytes (4 \* 16 bits) for the input module beginning at input byte 50 (%IB50) and going to input byte 57 (%IB57) of the input memory. The operating mode of all four channels is bipolar, with channels 1 and 2 in the measurement range  $\pm 10V$  and channels 3 and 4 in  $\pm 1V/\pm 20mA$ .



## ADU308

Here the example describes the insertion of an analogue input module ADU308 with 8 potentially isolated analogue inputs in slot 4.

The system reserves the required 16 Bytes (8 \* 16 Bits) for the input module beginning at input byte 58 (%IB58) up to input byte 73 (%IB73) of the input memory area, directly after the ADU306 already inserted in slot 3. A yellow bar with the description "preview" designates this; the orange bar with the description ADU306@003 shows the already occupied memory area.

The channels 1 to 4 are unipolar with a measurement range of 0...10V and channels 5 to 8 are bipolar with measurement ranges  $\pm 1V/\pm 20mA$  (channels 5 and 6) and  $\pm 10V$  (channels 7 and 8).



**Note:** Optionally for the ADU306 one output byte and for the ADU308 two output bytes (%QB) can be configured. The measurement range can be defined via these bytes [here output byte 10 (%QB10) and output byte 11 (%QB11)] or equally using the parameters 3 and 4 (see also the example in Part II, Chapter 0). By using the output byte the measurement range can be dynamically changed. Settings using the parameters require the PLC program to be reloaded.

Steckplatzparameter

Platz: 004 Baugruppe: ADU308

Baugruppen-Adressierung

INPUT

Startadresse: %IB 58 Länge: 16 B Ende: %IB 73

Speicherbelegung

von 48 48 50 52 54 56 58 60 62 64 66 68 70 72 74 bis 74

OUTPUT

Startadresse: %QB 10 Länge: 2 B Ende: %QB 11

Speicherbelegung

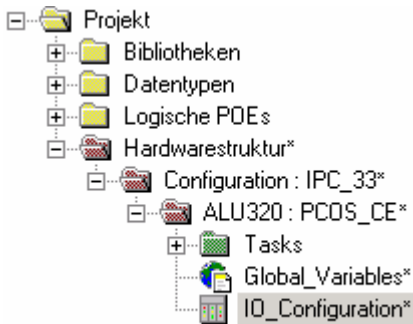
von 6 6 7 8 9 10 11 12 13 14 15 16 bis 16

Zusatzparameter  dynamisch zur Laufzeit

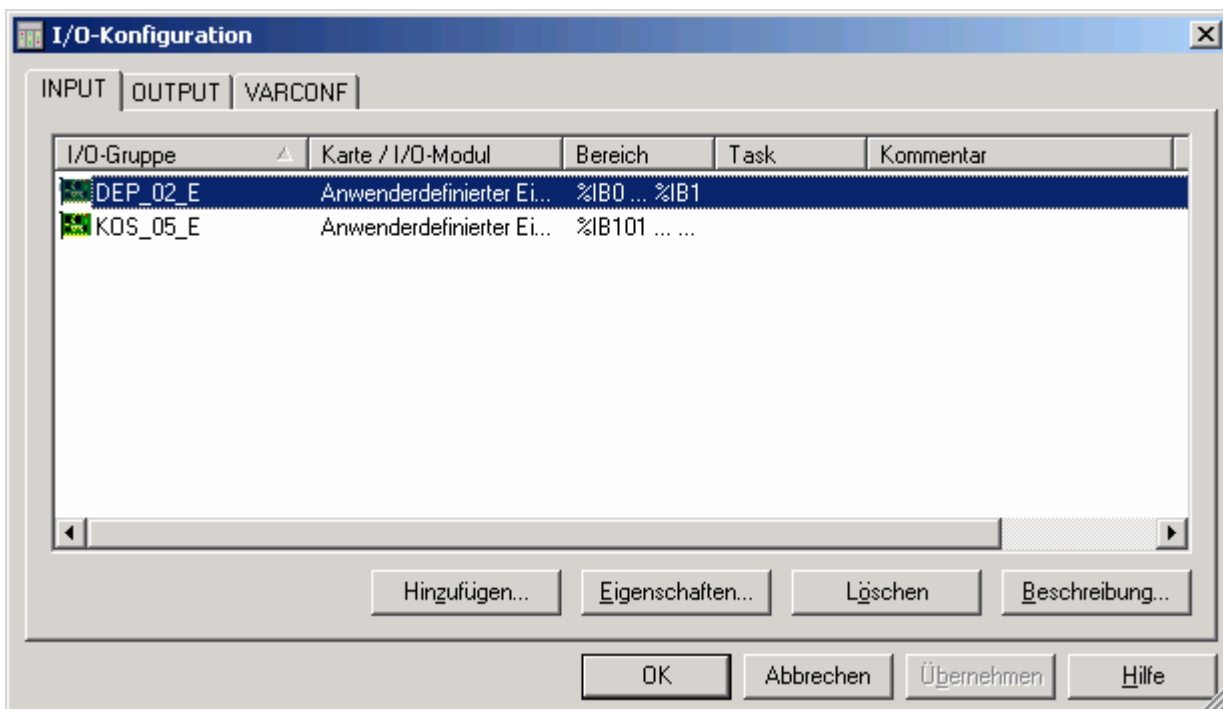
Eingänge 1-4		Eingänge 5-8	
Betriebsart	Messbereich:	Betriebsart	Messbereich:
<input type="radio"/> bipolar	1: D...10V	<input checked="" type="radio"/> bipolar	5: ±1V/±20mA
<input checked="" type="radio"/> unipolar	2: D...10V	<input type="radio"/> unipolar	6: ±1V/±20mA
<input type="checkbox"/> Drahtbr.-Überw.	3: D...10V	<input type="checkbox"/> Drahtbr.-Überw.	7: ±10V
<input checked="" type="checkbox"/> Überst.-Überw.	4: D...10V	<input type="checkbox"/> Überst.-Überw.	8: ±10V

## 4.2 Inputs and Outputs with the Multiprog I/O Configurator

Using the I/O-Configuration the interfaces to the inputs and outputs, the special modules and the ModbusTCP are defined.



The I/O-Configuration is an under menu in the hardware structure and can be opened by double-clicking or via the pop-up window (selected using the right-hand mouse button).



**Caution:** The given name of each module must be unambiguous. This also applies to mixed modules such as DAP 212 or the KOS xxx. For each of these, an entry must be made in both the INPUT and OUTPUT. It is not permitted to assign the same name. A maximum of 8 characters are allowed.

Example: DEP\_02\_E for the inputs of the DEP 316 in slot 2  
KOS\_05\_E and KOS\_05\_A for the input and output bytes of the KOS in slot 5.

The length of logical address entries is dependant on the module and can be taken from the following table.

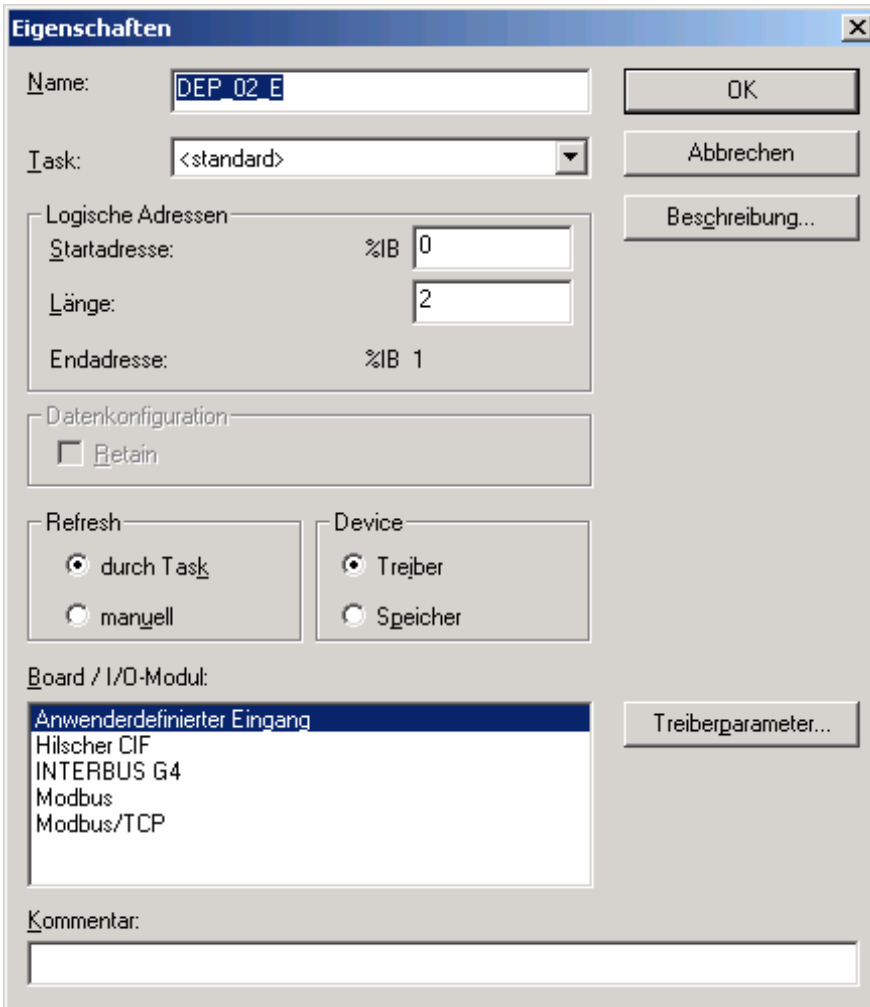
## Module Overview

Module	Ident-code	Number of Bytes		Comments
		Input	Output	
				Mixed Digital Inputs and Outputs
DAP212 / DAP292 / DAP 312	8	1	1	8 I + 4 O
DAP220 / DAP 320	9	1	1	8 I + 4 O
				Digital Inputs
DEP216/DEP316 DEP296/DEP297	12	2		2 x 8 I
DEO216	13	2		2 x 8 I
DEP220	14	2		2 x 8 I
DEP208	15	1		1 x 8 I
DEP210	16	1		1 x 8 I
DEP211	17	1		1 x 8 I
DEP218	18	2		2 x 8 I
DEP209	19	1		1 x 8 I
DEP214 / DEP 314	20	2		2 x 8 I
DEX216	44	2		
DEP217	45	2		2 x 8 I
DEP215	46	2		2 x 8 I
DEP332	70	4		4 x 8 I
				Digital Outputs
DAP204	23		1	4 O
DAP208 / DAP 308	24		1	8 O
DAP216 / DAP 316	25		2	2 x 8 O
DAP209	26		1	8 O
DAO216	27		2	2 x 8 O
DAP210	28		1	8 O
DAP218	29		2	2 x 8 O
DAX216	41		2	2 x 8 O
DAP217	42		2	2 x 8 O

DAP332	71		4	4 X 8 O	in preparation
				Analogue Inputs (ADU)	
ADU204 / ADU 304	32	8		4 x AI	
ADU205	34	8		4 x AI	
ADU206 / ADU 306	35	8	1 *)	4 x AI	
ADU308	38	16	2 *)	8 x AI	
ADU 214 / ADU 314	36	16	8 *)	8 x AI	
ADU 210	56	8	2 *)	4 x AI	
ADU 216	59	8		4 x AI	
				Analogue Outputs (DAU)	
DAU202 / DAU 302	48		4	2 x AO	
DAU208 / DAU 308	50		16	8 x AO	
				Communication Modules	
KOS201 / KOS202A	90	128	128	For substations and central stations	
KOS201 / KOS202A	92	32	32	For sub-masters	
KOS 203	93	128	128	For substations	
KOS 260	91	128	128	For primary racks	
KOS 260	95	48	48	For secondary racks	

\*) The output byte (%QB) can be used for parameter assignment.

#### 4.2.1 Definition Standard I/O Module



The screenshot shows a dialog box titled "Eigenschaften" with a close button (X) in the top right corner. The dialog is organized into several sections:

- Name:** A text input field containing "DEP\_02\_E".
- Task:** A dropdown menu showing "<standard>".
- Logische Adressen:** A section containing three input fields: "Startadresse:" with value "0", "Länge:" with value "2", and "Endadresse:" with value "1". Each field has a "%IB" label to its left.
- Datenkonfiguration:** A section with a checkbox labeled "Retain" which is currently unchecked.
- Refresh:** A section with two radio buttons: "durch Task" (selected) and "manuell".
- Device:** A section with two radio buttons: "Treiber" (selected) and "Speicher".
- Board / I/O-Modul:** A list box containing the following items: "Anwenderdefinierter Eingang" (highlighted in blue), "Hilscher CIF", "INTERBUS G4", "Modbus", and "Modbus/TCP".
- Kommentar:** A large empty text area at the bottom.

On the right side of the dialog, there are three buttons: "OK", "Abbrechen", and "Beschreibung...". Below the list box, there is a button labeled "Treiberparameter...".

For boards / I/O modules the setting "User Defined Input/Output" should always be selected. Subsequently select the button "Driver Parameter..." to open the menu for defining the driver details.

Treibername: PAB320  
Parameter 1: 2  
Parameter 2: 12  
Parameter 3: 0  
Parameter 4: 0  
Datentyp: BYTE

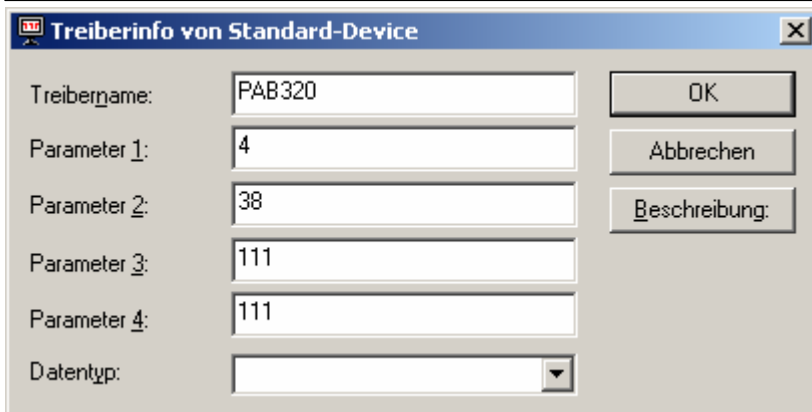
### Entering Driver Parameters

- As driver name always enter **PAB320**.
- In Parameter 1 enter the slot number of the module.
- In Parameter 2 enter the Identcode of the module.
- For certain modules Parameters 3 and 4 contain some module specific parameters e.g. some ADUs have their measurement range defined as a decimal value.

### Measurement Range Selection ADUs

Treibername: PAB320  
Parameter 1: 4  
Parameter 2: 35  
Parameter 3: 111  
Parameter 4:  
Datentyp:

**ADU 206/306**



### ADU 308

The measured values for the second group of 4 are defined in Parameter 4.

The initial state 0 means:

- All 4 inputs to measurement range  $\pm 1$  V or  $\pm 20$ mA according to the wiring of the connections or in the case of the ADU 308 according to the DIP-switch settings.
- No monitoring during overloading
- No monitoring of wire break
- Bi-polar operation

Deviating from the basic settings the following individually pre-selections are possible. These settings result from the addition of the values:

1	(Bit $2^0$ )	Input 1 to measurement range $\pm 10$ V
2	(Bit $2^1$ )	Input 2 to measurement range $\pm 10$ V
4	(Bit $2^2$ )	Input 3 to measurement range $\pm 10$ V
8	(Bit $2^3$ )	Input 4 to measurement range $\pm 10$ V
16	(Bit $2^4$ )	Unipolar operation, resolution 12 Bit without sign, cannot be combined with wire break monitoring, with outputs a conversion of the digital values is necessary.
32	(Bit $2^5$ )	All 4 inputs to measurement range 4 ... 20mA when using jumpers on the inputs with simultaneous wire-break monitoring with currents $< 2.08$ mA
47	(Bit $2^0 - 2^3, 2^5$ )	or all 4 inputs to measurement range 2 ... 10V, no jumpers on the inputs
64	(Bit $2^6$ )	Monitoring of measured values greater than the nominal value + tolerance (overload) in all 4 inputs.
128	(Bit $2^7$ )	Without meaning, setting remains 0



**Note:** You can choose to configure one output byte (%QB) for the ADU 306 and two for the ADU 308. The measurement range can then be entered in these bytes exactly the same as Parameters 3 or 4. Using the output bytes the measurement range can be dynamically changed in the user program. But when using the Parameters, they must be downloaded to the PLC.

Treibername:	PAB320	OK
Parameter 1:	4	Abbrechen
Parameter 2:	56	Beschreibung:
Parameter 3:	9	
Parameter 4:	11	
Datentyp:		

## ADU 210

2 channels are determined at the same time.

**Caution:** As opposed to programming with AKF or Concept, using Multiprog the channels cannot be individually configured.

Channels 1 and 2 are configured together via parameter 3 and channels 3 and 4 via parameter 4.

In the example, channels 1 and 2 are set to 2...10 V with 0 limit. Channels 3 and 4 are set to 0...10 V with 0 limit.

The codes for setting the measurement ranges can be found in the ADU module description.



**Note:** You can choose to configure two output bytes (%QB) for the ADU 210. The measurement range can then be entered in these bytes exactly the same as parameters 3 respectively 4. Using the output bytes the measurement range can be dynamically changed in the user program. But when using the Parameters, they must be downloaded to the PLC.

Treibername:	PAB320	OK
Parameter 1:	5	Abbrechen
Parameter 2:	36	Beschreibung:
Parameter 3:	1	
Parameter 4:	2	
Datentyp:		

With ADU 214 / ADU 314 the measurement ranges of channels 1 to 4 are set using parameter 3 and channels 5 to 8 via parameter 4. In the example channels 1 to 4 are set to 0...10 V and channels 5 to 8 to 0...5 V.

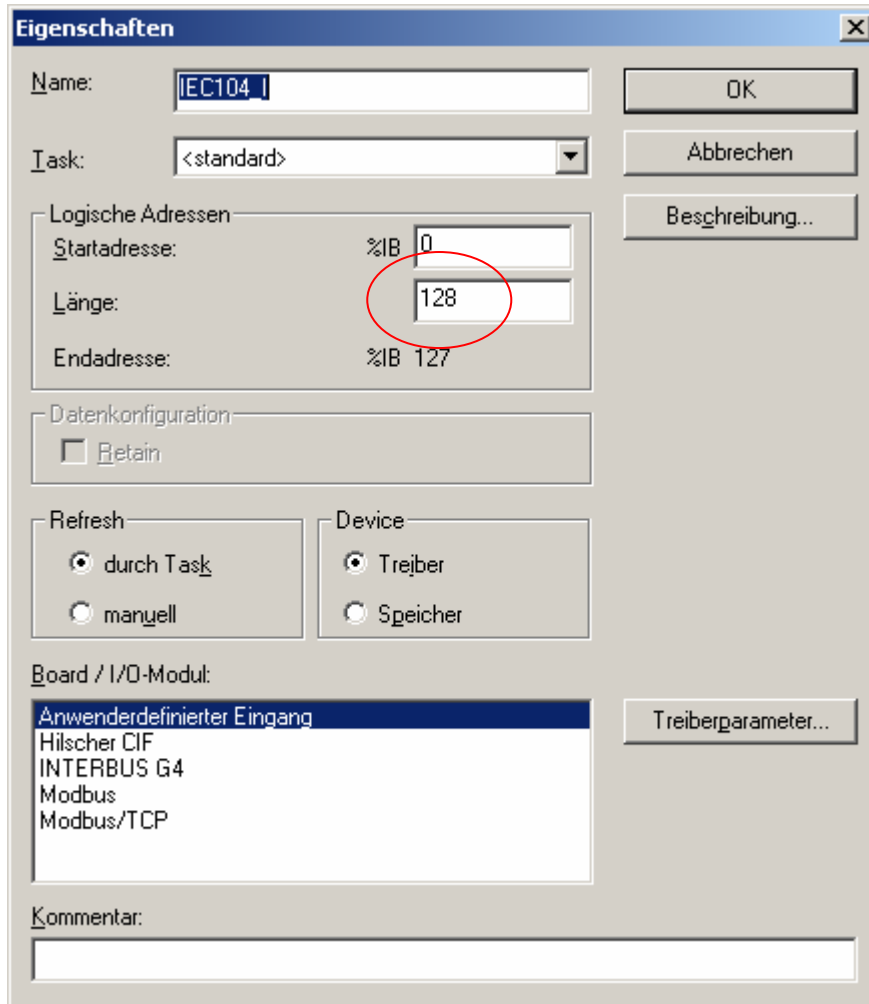
The codes for setting the measurement range can be found in the ADU module description.



**Note:** You can choose to configure the measurement range of each individual input of the ADU 214 / 314 with 8 output bytes (%QB). Using the output bytes the measurement range can be dynamically changed in the user program. But when using the Parameters, they must be downloaded to the PLC.

### 4.3 I/O Configuration for IEC 60870-5-101 or IEC 60870-5-104

The area for transferring data to the communications interface of the ALU 320 is defined in a virtual I/O module. It is possible to configure more than one of these modules.



Length of input max. 256  
Length of output max. 512

The user can define the name. It can have up to 8 characters and must be unambiguous. It is recommended that the transmit and receive directions are assigned with O or A for the outputs and I or E for the inputs.

Afterwards, using the **Driver Parameter** of the **User Defined Input/Output** you can assign the service routines for IEC 60870-5-101 or 104.

Using the appropriate configuration (file PCOS\_A320.INI) the ALU 320 can simultaneously operate a number of Service Routines.

Example:

Service1 = IEC 60870-5-104 (Driver = SVC104SA, TCP/IP interface)  
Service2 = IEC 60870-5-101 (Driver = SVC101SA, serial interface COM 1)  
Service3 = IEC 60870-5-101 (Driver = SVC101SA, serial interface COM 2)

Treibername: SVC104SA  
Parameter 1: 1  
Parameter 2: 0  
Parameter 3: 0  
Parameter 4: 0  
Datentyp: [dropdown]

Buttons: OK, Abbrechen, Beschreibung:

SVC104SA is the service for the IEC 60870-5-104 protocol.

The Service Number in Parameter 1 must be identical with the number in the files SVC\_COM? and SVC\_RTU?.

See also chapter 6

Treibername: SVC101SA  
Parameter 1: 2  
Parameter 2: 0  
Parameter 3: 0  
Parameter 4: 0  
Datentyp: [dropdown]

Buttons: OK, Abbrechen, Beschreibung:

SVC101SA is the Service for the IEC 60870-5-101 protocol.

The Service Number in Parameter 1 must be identical with the number in the files SVC\_COMn.INI and SVC\_RTUn.INI.

See also chapter 6

**Caution:**

It is imperative that the Service Routines configured in the I/O Configuration are enabled in the PCOS-A320 prior to the project being downloaded to the ALU 320. If this is not done, the download will be aborted with an error.



For additional information see chapters 0 and 0



**Note:** Commencing firmware version 1.70 different filenames are used (PCOSA320.INI, SVCCOMn.INI and SVCRTUn.INI)

#### 4.4 I/O Configuration for Modbus TCP Client

The driver will be displayed in the window "Board / I/O-Module", when a new group is inserted in the I/O configuration.

The configuration is described in the INPUT Dialogue of the I/O Configuration.

The screenshot shows the 'Eigenschaften' dialog box for configuring a Modbus TCP Client. The 'Name' field is set to 'MOD'. The 'Task' is set to 'ZYK\_1s'. Under 'Logische Adressen', the start address is 0, the length is 200, and the end address is 199. The 'Datenkonfiguration' section has the 'Retain' checkbox unchecked. In the 'Refresh' section, 'durch Task' is selected. In the 'Device' section, 'Treiber' is selected. The 'Board / I/O-Modul' list shows 'Modbus/TCP' selected. There is a 'Kommentar' field at the bottom.

#### Task:

To achieve fixed scanning times for Modbus you must define a cyclic task for calling the driver. If however you have no requirement for cyclical program calls in your application, you can create an empty "dummy" program for this purpose.

#### Logical Addresses:

In this scan dialogue enter the first IB address, from where the transmitted data will be stored.

#### Length:

The length of the scanned data is given as a number of bytes. In the example above 200 bytes = 100 WORD or registers will be scanned from the connected station.

Using the "Driver Parameter..." it is defined which registers from the connected station will be scanned.

Driver information of device MODBUS/TCP

Drivename: MODTCP

Device IP Address: 192 . 168 . 0 . 64

Device Modbus Address: 1000

Response Timeout: 1000 ms

Datatype: WORD

Function code: ReadHoldingRegister (FC3)

0 Based Device  
 1 Based Device

Synchronous Update

Successive Transactions  
 Simultaneous Transactions

OK  
Cancel  
Description

### Device IP Address:

Enter the IP address of the connected station in the field device IP address.

### Device Modbus Address:

Enter the address of the first register, which should be scanned in the "Device Modbus Address" field.

In the example configuration, 100 registers will be scanned from the connected station commencing register 1000 (401000, MW 1000 etc.) and stored in Multiprog in the area IB0 ... IB199.

### Function Code:

Here you can define whether you want to scan the connected station for Registers (WORD), Coils (BIT) or direct from the I/O.

The selected function code must be supported by the connected station.

Via the **Based Device**, value 0 or 1, you define your method of numbering i.e. whether the Register in the connected station begins with 0 or 1.

By defining a number of I/O groups you can interrogate various areas of the connected stations. The areas can be interrogated either successively (**Successive Transaction**) or in parallel (**Simultaneous Transaction**).

Successively means that the interrogation telegram is only sent when the response from the previous interrogation is received or the configured **Response Timeout** has elapsed.

Parallel means that a number of interrogation telegrams are sent one after the other irrespective of if a response has been received or not. The data transmission is therefore asynchronous.

For the configuration of outputs via Modbus/TCP you must use the same method as for the inputs. Use the OUTPUT dialog in the I/O configuration.

---

#### 4.5 Configuration of the Shared-Memory-Area for Modbus RS232-Slave / TCP-Server

The ALU 320 provides a total area of 16kB (MB3.0000 to MB3.16383) for Shared Memory. This area or part of it can be reserved for the Modbus RS232 Slave or TCP-Server interfaces.



For additional information see chapter 6.5.



**Note:** Commencing firmware version 1.70 different filenames are used (PCOSA320.INI and MBSRV.INI)

### 5 System Markers

For the maximum number of 20 slots in an @120 (DTA 300 + 3 x DTA 301) a System Marker area consisting of Double Words is reserved for errors and status display. Status information is only displayed for those slots where I/O is configured.

With simple I/O modules only the "Module Failure" and "Module Fault" System Markers are used. For expert modules (e.g. KOS xxx) and complex I/O modules (e.g. ADU 206) and 4 status bytes are additionally available.

The System Marker area has a fixed structure. Marker Bits and Marker Bytes for slots that are not configured have the value 0.



**Note** for programmers who are converting from the A120 system to the @120.  
The System Marker Double Word "Module Failure" %MD1.3000 corresponds to SM31 ... SM48 in AKF12.  
The System Marker Double Word "Module Fault" %MD1.3004 corresponds to SM2.1, SM3.1 etc. in AKF12.

The System Marker can be addressed as Double Words, Bytes or Bits.

To generate a group fault during module failure you can for example check if %MD1.3000 is not equal to ZERO.  
For checking individual slots it makes sense to interrogate in the same manner i.e. the Marker Bits %MX1.3000.1, %MX1.3000.2 etc.

The definitions can be found in the following table:

## 5.1 System Markers - Service

%MD1.2952	%MB1.2952 %MB1.2953 %MB1.2954 %MB1.2955	Status Byte DCF reception (in preparation)
%MD1.2956	%MB1.2956 %MB1.2957 %MB1.2958 %MB1.2959	Status byte service group marker Status byte service group marker Status byte service group marker Status byte service group marker
%MD1.2960	%MB1.2960 %MB1.2961 %MB1.2962 %MB1.2963	Status byte 1 Service 1 Status byte 2 Service 1 Status byte 3 Service 1 Status byte 4 Service 1
.	.	.
.	.	.
.	.	.
%MD1.2992	%MB1.2992 %MB1.2993 %MB1.2994 %MB1.2995	Status byte 1 Service 9 Status byte 2 Service 9 Status byte 3 Service 9 Status byte 4 Service 9

### Status byte Service Group Marker MD1.2956:

Bit position corresponds to the group error of a system service SVC0 to SVC9 (SVC0 is always reserved for an internal system service).

SVC1 to SVC9 can currently be configured for IEC 60870-5-101 or IEC 60870-5-104 communication.

Group error-Bit0 of MD1.2956 reserved  
Group error -Bit1 of MD1.2956 for Service 1  
Group error -Bit2 of MD1.2956 for Service 2  
etc.

### Status Byte1 Service n:

Bit 0	Not defined
Bit 1	1 = Participant has no connection, 0 = connection OK (cyclical Timeout)
Bit 2	1 = Bus no connection, 0 = Bus Ok (Bus Timeout)
Bit 3...7	Not defined

The status bytes 2...4 are not used at the moment.

## 5.2 System Markers - Slots Occupied in the I/O Configuration

<b>%MD1.2996</b>	%MB1.2996	%MX1.2996.0	ALU (slot 1) - reserved	Marker for slots 1 to 20  0 ... not configured 1 ... occupied
		%MX1.2996.1	I/O card (slot 2)	
		%MX1.2996.2	I/O card (slot 3)	
		%MX1.2996.3	I/O card (slot 4)	
		%MX1.2996.4	I/O card (slot 5)	
		%MX1.2996.5	I/O card (slot 6)	
		%MX1.2996.6	I/O card (slot 7)	
		%MX1.2996.7	I/O card (slot 8)	
	%MB1.2997	%MX1.2997.0	I/O card (slot 9)	
		%MX1.2997.1	I/O card (slot 10)	
		%MX1.2997.2	I/O card (slot 11)	
		%MX1.2997.3	I/O card (slot 12)	
		%MX1.2997.4	I/O card (slot 13)	
		%MX1.2997.5	I/O card (slot 14)	
		%MX1.2997.6	I/O card (slot 15)	
	%MB1.2998	%MX1.2997.7	I/O card (slot 16)	
%MX1.2998.0		I/O card (slot 17)		
%MX1.2998.1		I/O card (slot 18)		
%MX1.2998.2		I/O card (slot 19)		
%MX1.2998.3		I/O card (slot 20)		
%MX1.2998.4		Free (0)		
.				
.				
%MX1.2998.7	Free (0)			
%MB1.2999	%MX1.2999.0	Reserved		
	.			
	.			
	%MX1.2999.7			

These Markers display, whether a slot is occupied in the I/O configuration. When configured, the corresponding bit is set to 1.

### 5.3 System Markers – Module Failure

%MD1.3000	%MB1.3000	%MX1.3000.0	ALU (slot 1) - reserved	Markers for modules 1 to 20  0 OK 1 <b>Failure</b> — only configured modules
		%MX1.3000.1	I/O module (slot 2)	
		%MX1.3000.2	I/O module (slot 3)	
		%MX1.3000.3	I/O module (slot 4)	
		%MX1.3000.4	I/O module (slot 5)	
		%MX1.3000.5	I/O module (slot 6)	
		%MX1.3000.6	I/O module (slot 7)	
		%MX1.3000.7	I/O module (slot 8)	
	%MB1.3001	%MX1.3001.0	I/O module (slot 9)	
		%MX1.3001.1	I/O module (slot 10)	
		%MX1.3001.2	I/O module (slot 11)	
		%MX1.3001.3	I/O module (slot 12)	
		%MX1.3001.4	I/O module (slot 13)	
%MX1.3001.5		I/O module (slot 14)		
%MX1.3001.6		I/O module (slot 15)		
%MB1.3002	%MX1.3002.0	I/O module (slot 17)		
	%MX1.3002.1	I/O module (slot 18)		
	%MX1.3002.2	I/O module (slot 19)		
	%MX1.3002.3	I/O module (slot 20)		
	%MX1.3002.4	Free (0)		
	.			
%MX1.3002.7	Free (0)			
%MB1.3003	%MX1.3003.0	Reserved		
	.			
	%MX1.3003.7			

If the slot is defined in the I/O configuration and is not physically occupied or the module is defective, this module will be reported as failed.

#### 5.4 System Markers – Module Fault

<b>%MD1.3004</b>	%MB1.3004	%MX1.3004.0 %MX1.3004.1 %MX1.3004.2 %MX1.3004.3 %MX1.3004.4 %MX1.3004.5 %MX1.3004.6 %MX1.3004.7	ALU (slot 1) - reserved I/O module (slot 2) I/O module (slot 3) I/O module (slot 4) I/O module (slot 5) I/O module (slot 6) I/O module (slot 7) I/O module (slot 8)	Markers for modules 1 to 20  0 ... OK 1 ... <b>Error</b>
	%MB1.3005	%MX1.3005.0 %MX1.3005.1 %MX1.3005.2 %MX1.3005.3 %MX1.3005.4 %MX1.3005.5 %MX1.3005.6 %MX1.3005.7	I/O module (slot 9) I/O module (slot 10) I/O module (slot 11) I/O module (slot 12) I/O module (slot 13) I/O module (slot 14) I/O module (slot 15) I/O module (slot 16)	
	%MB1.3006	%MX1.3006.0 %MX1.3006.1 %MX1.3006.2 %MX1.3006.3 %MX1.3006.4 . . %MX1.3006.7	I/O module (slot 17) I/O module (slot 18) I/O module (slot 19) I/O module (slot 20) Free (0)  Free (0)	
	%MB1.3007	%MX1.3007.0 . . %MX1.3007.7	Reserved	

A number of modules e.g. KOS xxx or ADU 206/306, etc. can report failures. The system marker “Module Failure” is used as a group marker for these failures. The exact reason for the failure will be defined via the status byte (see chapter 5.5).

## 5.5 System Markers – Status Bytes

%MD1.3008	%MB1.3008	Status byte I/O module – slot 1
	%MB1.3009	Status byte I/O module – slot 1
	%MB1.3010	Status byte I/O module – slot 1
	%MB1.3011	Status byte I/O module – slot 1
%MD1.3012	%MB1.3012	Status byte I/O module – slot 2
	%MB1.3013	Status byte I/O module – slot 2
	%MB1.3014	Status byte I/O module – slot 2
	%MB1.3015	Status byte I/O module – slot 2
.	.	.
.	.	.
.	.	.
%MD1.3084	%MB1.3084	Status byte I/O module – slot 20
	%MB1.3085	Status byte I/O module – slot 20
	%MB1.3086	Status byte I/O module – slot 20
	%MB1.3087	Status byte I/O module – slot 20

### 5.5.1 Definition - Status Bytes KOS201

1<sup>st</sup> Status Byte

Bit 0	Slot failure KOS
Bit 1	SEAB communication faulty ACE: number of dialling attempts unsuccessful
Bit 2	Minute impulse missing
Bit 3	Time missing
Bit 4	No parameter
Bit 5	
Bit 6	Battery error
Bit 7	Group error Bit 0 to 4

2<sup>nd</sup> Status byte – not used

3<sup>rd</sup> Status byte – not used

4<sup>th</sup> Status byte

Bit 0	KOS output 1=busy, 0= OK
Bit 1	
Bit 2	
Bit 3	
Bit 4	KOS input 1=busy, 0= OK
Bit 5	
Bit 6	
Bit 7	

### 5.5.2 Time-of-Day Transfer via KOS 201/202

The KOS transfers the time in 8 consecutive bytes, provided that it is synchronised. The synchronisation can be made via a SEAB-1F time telegram or a DCF radio signal. The time transfer can in turn be used to synchronise the CPU clock.

The 8 bytes are defined as follows:

KOS in a substation	KOS in a sub-master	Description	Coding
121	17	Year	BCD
122	18	Month	BCD
123	19	Day of the week	BCD
124	20	Day	BCD
125	21	Hour	BCD
126	22	Minute	BCD
127	23	Special sign	1. Bit = Switch to Reserve 2. Bit = Announcement of ST/WT switch-over 3. Bit = Summer time 4. Bit = Winter time 5. Bit = Switching second
128	24	Second	BCD

The entry of the byte can always be seen relative to the configured %IB area in the I/O configuration. In the example template this corresponds to the area %IB121 ... %IB128.

### 5.5.3 Definition – Status Bytes ADU 206/306/308

1<sup>st</sup> Status byte

Bit 0	Overdriving respectively wire-break with currents <2.08mA on Input 1
Bit 1	Overdriving respectively wire-break with currents <2.08mA on Input 2
Bit 2	Overdriving respectively wire-break with currents <2.08mA on Input 3
Bit 3	Overdriving respectively wire-break with currents <2.08mA on Input 4
Bit 4	Operational mode unipolar
Bit 5	2 ... 10V / 4 ... 20mA on inputs 1 ... 4
Bit 6	1= external 24 volt supply missing
Bit 7	<u>ADU not ready, reason:</u> Overdriving respectively wire-break on one of the four inputs or the processor monitoring has responded or the U = 24 V is missing or the AS-BADU is still in the initialisation phase.

2<sup>nd</sup> Status byte

In the ADU 308 the second status byte is defined exactly the same as the first status byte however it represents the status of the measured value inputs 5 to 8.

### 5.5.4 Definition – Status Bytes ADU 210/214

To simplify the error handling in Multiprog the error byte from the ADU is distributed in two status bytes by the I/O driver.

The 1<sup>st</sup> status byte of the corresponding slot contains the error number (Bit 2<sup>0</sup> – 2<sup>3</sup> of the ADU error byte).

The 2<sup>nd</sup> status byte of the corresponding slot contains the input number (Bit 2<sup>4</sup> – 2<sup>7</sup> of the ADU error byte).

For the coding of the error number please refer to the corresponding ADU module description.

### 5.5.5 Definition – Status Bytes Modbus

1<sup>st</sup> Status byte

Bit 0	Interface: 1 = ready, 0 = error
Bit 1	Interface: 1 = running, 0 = Stopped
Bit 2	Reserved
Bit 3	Reserved
Bit 4	COM Task: 1 = ready, 0 = error
Bit 5	Reserved
Bit 6	Reserved
Bit 7	Fixed 1

---

2<sup>nd</sup> Status byte

Bit 0	Communication status: 1 = connected, 0 = not connected
Bit 1	Reserved
Bit 2	Reserved
Bit 3	Reserved
Bit 4	Reserved
Bit 5	Reserved
Bit 6	Reserved
Bit 7	Reserved

3<sup>rd</sup> Status byte

Number of the function, which caused the last error

4<sup>th</sup> Status byte

Modbus error code

1	Unknown respectively illegal function code
2	Requested address or area not known
3	Illegal reported value (dependant on reported type)

5<sup>th</sup> Status byte

Continuous error counter with rollover at 255.

6<sup>th</sup> Status byte

Continuous reply counter with rollover at 255.

The memory address and the number of status bytes are defined in the MODBUS\_SERVER.INI file. The status bytes 3 ... 6 are optional.



For additional information see chapter 6.5.1.

## 6 Configuration of the Initialisation Files

The majority of the following initialisation files can be generated via the OHP Toolbar and transferred to the ALU. The transfer is done partially by "send configuration" and partially by "FTP transfer" in the corresponding configuration menu. Modified settings are only effective after a new start of the ALU.

Editing the files using a text editor is also possible.

### 6.1 Settings in the ALU320.INI

Configuration using PLC Configuration "ALU Parameter → Edit basic settings", transfer per FTP.

The ALU320.INI file can be found in the ROOT directory of the CF card. This file contains the settings regarding the IP address and various services for access to the ALU320.

---

#ALU320: (GEN)

[DEVICE]

RASVPN= FALSE

[IP]

ADDRESS = 192.168.0.32  
NETMASK = 255.255.255.0  
DHCP = 0

[FTP]

ENABLE = 1

[TELNET]

ENABLE = 1

[TIME-GPS-NMEA]

ENABLE = FALSE  
PORT = 2  
BAUD = 4800 # should not be changed  
REFRESH = 60 # in minutes

### 6.1.1 Enabling Device Drivers

When using an ALU320-017 with integrated GPRS modem, the modem can be used to connect to a provider and if necessary to a VPN tunnel. For this purpose the entry RASVPN=TRUE must be set.

If RASVPN=TRUE is set, the ALU320 searches for the ALURAS.INI file. This file contains the settings for the GPRS connection.

### 6.1.2 Setting the IP Address on the ALU320

ADDRESS and NETMASK must always be entered. When you are operating the ALU320 via a gateway, you can extend the file as follows.

Example:

ADDRESS= 192.168.101.101  
NETMASK= 255.255.255.0  
GATEWAY= 192.168.101.1

Provided you have changed the file via an FTP access, you must cycle the ALU 320 by switching it off and on again, only then is the data adopted.

### 6.1.3 Access via FTP or TELNET

The entries [FTP] und [TELNET] make access to the ALU possible with the corresponding services. When you want to prohibit this access then set the ENABLE =0.

**Caution:** When you deactivate the FTP access, you can only modify the INI files on the ALU 320 using a card reader.

### 6.1.4 Time Synchronisation via GPS

Parameter under [TIME-GPS-NMEA]

If the parameters **ENABLE** is set to TRUE, then a time synchronisation of the ALU320 can take place via a GPS receiver.

**PORT** defines the COM interface (1 or 2) to which the receiver is connected.

---

---

**BAUD** defines the transmission rate with which the receiver transmits the data to the ALU320. This entry must not be changed.

**REFRESH** defines the time interval in minutes (1...60) after which the system clock is synchronised by the GPS receiver. During this time a number of time signals are received by the GPS receiver, however the system clock will only be updated after the set time has elapsed and is synchronised by the next valid time signal. After each successful time synchronisation the refresh timer is reset. Also see chapter 8.1.

## 6.2 Settings in the ALURAS.INI

Configuration using PLC Configuration "ALU Parameter -> Edit basic settings -> GPRS VPN Settings", transfer per FTP.

**Caution:** Most of the following parameters described below must not be changed. The relevant settings required for individual configurations are prompted by the PLC configuration in Multiprog. Please use the PLC configuration exclusively for editing the file. OHP GmbH cannot guarantee trouble-free operation due to inappropriate changes made to these settings

```
[AXRASVPN-PARAMS]
PPPDisconnect= FALSE
PPPDisconnect3Att= FALSE
RedialWTimeMS= 5000
FailRedialWTimeMS= 5000
ReConnectWTimeMS= 5000
FailReConnectWTimeMS= 5000
PPPLogoutLoginMode= 1
PPPLogoutLoginWTimeSec= 60
PPPLogoutLoginAt= 12:00:00
PPPCheckWtimeSec= 3600
HwReset3Att= TRUE

[AXRASVPN-PARAMS-T2]
VPNDisable= FALSE
DialRetry= 0
DialRetryWTimeMin= 0
VpnRetry= 0
VpnRetryWTimeMin= 0

[DIAL-LOCATION]
LOCAL= G
DISTANCE= G
INTERNATIONAL= G
AREA=
COUNTRY= 49
DIALTONE= FALSE
DISABLE-WAITING= TRUE
WAITING-CMD=
```

```
[RASMODEM]
Baudrate= 19200
Parity= NO
DataBits= 8
Stopbits= ONE
FlowCtrl= HARD
CancelTime= 120
DialWaitTime= 0
DialCmdOpt= +CGDCONT=1,"IP","internet.t-mobile"
```

```
[[ISP-PPP-DIAL]
ComPort= 3
CountryCode= 49
AreaCode= 60
LocalPhoneNumber= *99***1#
DialAsLocalCall= FALSE
```

```
[[ISP-PPP-TCPIP]
IpHeaderCompression= FALSE
SwCompression= FALSE
Slip= FALSE
SpecificIpAddr= FALSE
IpAddr= 0.0.0.0
NameServers= FALSE
ipaddrDns= 0.0.0.0
ipaddrDnsAlt= 0.0.0.0
ipaddrWins= 0.0.0.0
ipaddrWinsAlt= 0.0.0.0
```

```
[[ISP-PPP-SECURITY]
RequireDataEncryption= FALSE
PAP= FALSE
CHAP= FALSE
MsCHAP= FALSE
MsCHAP2= FALSE
```

[ISP-PPP]

UserName=t-mobile

Password=tm

Domain=

[ISP-PPP-PARAMS]

AllowSuspend= 0

AlwaysRequestDNSandWINS= 0

AlwaysSuggestIpAddr= 0

AuthMaxTries= 16

AuthMaxFailures= 3

CryptTypesSupported= 64

MaxConfigure= 16

MaxFailure= 5

MaxTerminate= 2

RestartTimer= 3

[VPN-PPTP]

Hostname= 123.123.123.123

UserName=Beispiel

Password=123456

Domain=

[VPN-PPTP-TCPIP]

IpHeaderCompression= TRUE

SwCompression= FALSE

Slip= FALSE

SpecificIpAddr= FALSE

IpAddr= 0.0.0.0

NameServers= FALSE

ipaddrDns= 0.0.0.0

ipaddrDnsAlt= 0.0.0.0

ipaddrWins= 0.0.0.0

ipaddrWinsAlt= 0.0.0.0

[VPN-PPTP-SECURITY]

RequireDataEncryption= TRUE

```
PAP= FALSE
CHAP= FALSE
MsCHAP= FALSE
MsCHAP2= TRUE

[VPN-PPTP-PARAMS]
TcpPortNumber= 1723
UdpPortNumber= 47
MaxWanEndpoints= 5
MaxTransmit= 32
InactivityIdleSeconds= 60
AlwaysEcho= 1
TunnelConfig= 0
TcpDisconnectTimeout= 30
TcpConnectTimeout= 30

[UNIMODEM-INIT]
Init1= AT<cr>
Init2= AT&FE0V1&C1&D2<cr>
Init3=
Init4=
```

### 6.2.1 Setting the Parameter Set [AXRASVPN-PARAMS]

#### **PPPODisconnect**

The GPRS provider communication disconnects and reconnects after every abortive VPN server login attempt.  
Range: TRUE / FALSE.

#### **PPPODisconnect3Att**

The GPRS provider communication disconnects and reconnects after three abortive VPN server login attempts.  
Range: TRUE / FALSE.



**Note:** Is PPPDisconnect3Att=FALSE and PPPDisconnect=TRUE, then PPPDisconnect3Att=TRUE will be automatically set.!

#### **RedialWTimeMS**

Waiting time in milliseconds between a dial-up attempt to the GPRS provider (see also the graphic of waiting time basic principles). Range: 5000 - 134217727

#### **FailRedialWTimeMS**

Waiting time in milliseconds between three dial-up attempts to the GPRS provider (see also the graphic of waiting time basic principles). Range: 5000 - 134217727

---

### **ReConnectWTimeMS**

Waiting time in milliseconds between dial-up attempts to the VPN server (see also the graphic of waiting time basic principles). Range: 5000 - 1342177727

### **FailReConnectWTimeMS**

Waiting time in milliseconds between three dial-up attempts to the VPN server (see also the graphic of waiting time basic principles). Range: 5000 - 1342177727

### **PPPLogoutLoginMode**

Mode GPRS forced disconnection. Default = 1  
0: off (no forced disconnection)  
1: Disconnection at a predetermined time  
2: Disconnection after a predetermined time interval

### **PPPLogoutLoginWTimeSec**

Duration of the GPRS forced disconnection in seconds. Default 60 seconds

### **PPPLogoutLoginAt**

Time or time interval of the GPRS forced disconnection (dependant on the parameter *PPPLogoutLoginMode*).  
Format: hh:mm:ss

### **PPPCheckWtimeSec**

Monitoring interval GPRS traffic in seconds. If during this time no data has been transmitted via the GPRS connection, the GPRS communication will be disconnected and reconnected.

### **HwReset3Att**

Determines whether a hardware reset of the modem will be carried out after a third unsuccessful GPRS dial-up attempt. Range: TRUE / FALSE. Default=TRUE.

## **6.2.2 Setting of the Parameter Set [AXRASVPN-PARAMS- T2]**

### **VPNDisable**

Determines whether a VPN tunnel should be established. Range: TRUE / FALSE. Default = FALSE.

### **DialRetry**

Number of consecutive unsuccessful GPRS dial-up attempts after the waiting time "DialRetryWTimeMin" [minutes] has elapsed. Generally, after this time the optional parameter "HwReset3Att" will be checked. Range:3...63. values lower than 3 mean off. Default = 0.

### **DialRetryWTimeMin**

Waiting time in minutes after "DialRetry" unsuccessful dial-up attempts to the GPRS provider.  
Range: 5 - 1440 min. Default = 5.

### **VpnRetry**

Number of consecutive unsuccessful login attempts to a VPN server after the waiting time "DialRetryWTimeMin" [minutes] has elapsed. Generally, after this time the optional parameter "HwReset3Att" will be checked.  
Range:3...63. values lower than 3 mean off. Default = 0.

### **VpnRetryWTimeMin**

Waiting time in minutes after "VpnRetry" unsuccessful login attempts on the VPN server  
Range: 10 - 1440 min. Default = 10.

---

### 6.2.3 Setting of the Parameter Set [DIAL-LOCATION]

#### LOCAL

External access code for local calls. Range: G, E, F. Default = G.

#### DISTANCE

External access code for long distance calls. Range: G, E, F. Default = G

#### INTERNATIONAL

External access code for international calls. Range: G, E, F. Default = G

#### AREA

Dialling code. Range: alphanumeric characters.

#### COUNTRY

International dialling code. Range: alphanumeric characters. Default =49.

#### DIALTONE

Dialling method. Default = TRUE.

TRUE: Frequency dialling (MFV),  
FALSE: Impulse dialling

#### DISABLE-WAITING

Disable the following WAITING-CMD. Range: TRUE / FALSE. Default=TRUE.

#### WAITING-CMD

LASS code (Local Area Signalling Services) or/and user defined Calling Feature Control Codes:  
e.g. \*70 ... deactivate call waiting.  
Range: Free or ASCII characters. Default = free.

### 6.2.4 Setting of the Parameter Set [RASMODEM]

#### Baudrate

Speed of the RS232 interface to GPRS modem. Setting = 19200.

#### Parity

Parity of the RS232 interface to GPRS modem: Setting = NO (no parity bit)

#### DataBits

Number of data bits RS232interface to GPRS modem. Setting = 8.

#### Stopbits

Number of stop bits RS232 interface to GPRS modem. Setting ONE (1 stop bit)

#### FlowCtrl

Data flow control RS232 interface to GPRS modem. Setting = HARD (hardware)

#### CancelTime

Time in seconds, after which the GPRS dial-up will be discontinued, if no connection is achieved.  
Setting = 120.

#### DialWaitTime

Waiting time for a dialling tone in seconds. Setting = 0 (off).

---

### DialCmdOpt

PDP context with APN (Access Point Name) of the GPRS provider.

The APN of the corresponding providers must be entered here (last field). This will be advised of this when e.g. you have signed a special contract, by which the provider establishes a VPN tunnel. Standard APN can be found on the website of your provider.

Default = +CGDCONT=1,"IP","internet.t-mobile".

## 6.2.5 Setting of the Parameter Set [ISP-PPP-DIAL]

### ComPort

Number of the GPRS serial connection on the modem. With the ALU320 with integrated modem it is always COM3.

### CountryCode

International dialling code. Range: No entry or alphanumeric characters. Default = 49 (Germany).

### AreaCode

Dialling code. Range: No entry or alphanumeric characters.

### LocalPhoneNumber

GPRS dial-up number (dependant on the GPRS provider). Setting = \*99\*\*\*1#.

### DialAsLocalCall

Range: TRUE/FALSE . Default = FLASE.

## 6.2.6 Setting of the Parameter Set [ISP-PPP-TCPIP]

### IpHeaderCompression

Activated IP header compression on the GPRS connection level. Range: TRUE/FALSE. Default= FLASE.

### SwCompression

Activated software compression on the GPRS connection level. Range: TRUE/FALSE. Default= FLASE.

### Slip

Activated *Serial Line Internet Protocol* on the GPRS connection level.  
Range: TRUE/FALSE. Default= FLASE.

### SpecificIpAddr

Determines whether a special IP address should be used,  
Range: TRUE/FALSE. Default= FLASE.

### IpAddr

Determines the IP address for the GPRS connection. If *SpecificIpAddr=FALSE* this parameter will be ignored.

### NameServers

Determines if special IP addresses should be used for the name resolution on the GPRS level.  
Range: TRUE/FALSE. Default= FLASE.

### ipaddrDns

IP address of the DNS server (will be ignored if *NameServers=FALSE*)

### ipaddrDnsAlt

---

Alternative IP address of the DNS server (will be ignored if *NameServers=FALSE*)

**ipaddrWins**

IP address of the WINS server (will be ignored if *NameServers=FALSE*)

**ipaddrWinsAlt**

Alternative IP address of the WINS server (will be ignored if *NameServers=FALSE*)

## 6.2.7 Setting of the Parameter Set [ISP-PPP-SECURITY]

### RequireDataEncryption

Determines if it is compulsory that the GPRS provider must offer data encryption. Range: TRUE/FALSE. Default= FLASE.

### PAP

Determines if the Password Authentication Protocol (PAP) will be accepted on the GRPS connection level. Range: TRUE/FALSE. Default= FLASE.

### CHAP

Determines if the Challenge Handshake Authentication Protocol (CHAP) will be accepted on the GRPS connection level. Range: TRUE/FALSE. Default= FLASE.

### MsCHAP

Determines if the Microsoft Challenge Handshake Authentication Protocol (MSCHAP) will be accepted on the GRPS connection level Range: TRUE/FALSE. Default= FLASE.

### MsCHAP2

Determines if the Microsoft Challenge Handshake Authentication Protocol Version 2.0 (MSCHAPv2) will be accepted on the GRPS connection level Range: TRUE/FALSE. Default= FLASE.

## 6.2.8 Setting of the Parameter Set [ISP-PPP]

### UserName

GPRS user name (dependant on the GPRS provider). Must not be entered for all providers. Range: Free or maximum 16 ASCII characters. Default = t-mobile

### Password

GPRS password (dependant on the GPRS provider). Must not be entered for all providers. Range: Free or maximum 16 ASCII characters. Default = tm.

### Domain

GPRS registration domain. Must not be entered for all providers. Range: Free or maximum 16 ASCII characters. Default = <free>.

## 6.2.9 Setting of the Parameter Set [ISP-PPP-PARAMS]

### AllowSuspend

Determines the interruption behaviour of the driver. To allow the automatic interruption of a PPP connection the entry must be set to 1. Range: TRUE/FALSE. Default = FALSE.

### AlwaysRequestDNSandWINS

---

Determines if the client DNS and WINS request addresses from the server, even if static addresses are entered. Range: TRUE/FALSE. Default = FALSE.

#### **AlwaysSuggestIpAddr**

If the setting is TRUE, the client will always suggest an IP address, even when a static address is not defined. If the setting is FALSE, the client requests the IP address from the server. Range: TRUE/FALSE. Default = FALSE.

#### **AuthMaxTries**

Maximum number of attempts of the Windows CE RAS server takes to authenticate the client. Range: 1...16. Default =16.

#### **AuthMaxFailures**

Determines the maximum number of unsuccessful authentication attempts, before the Windows CE RAS server disconnects the communication. Range: 1...3. Default = 3.

#### **CryptTypesSupported**

Supports encryption procedures. Default = 64.

0..encryption deactivated

32..activates 40-Bit MPPE encryption

64.. activates 128-Bit MPPE encryption

96.. activates 40-Bit and 128-Bit MPPE encryption

#### **MaxConfigure**

Determines the maximum number of Configure Request (CR) packages, that were not confirmed with a Configure ACK, Configure No Acknowledgement (NAK) or Configure-Reject, after it can be safely assumed that the remote station will not answer. Range: 1...16. Default =16.

#### **MaxFailure**

Determines the maximum number of Configure NAK packages, that have not been confirmed by a Terminate ACK, after which it can be safely assumed that the configuration does not coincide. Range: 1...5. Default =5.

#### **MaxTerminate**

Determines the maximum number of Terminate Request packages, that have not been confirmed by a Terminate ACK, after which it can be safely assumed that the remote station will not answer. Range: 1...5. Default =2.

#### **RestartTimer**

Determines the waiting time in seconds, before an unsuccessful Configure Request or Terminate Request is repeated, after which it can be safely assumed that the remote station will not answer. Range: 1...16. Default =16.

---

### 6.2.10 Setting of the Parameter Set [VPN-PPTP]

#### Hostname

IP address of the VPN servers. Range: IP address

#### UserName

VPN user name. Range: max. 16 ASCII characters

#### Password

VPN password: Range: ASCII characters

#### Domain

VPN- registry domain. Range: ASCII characters

### 6.2.11 Setting of the Parameter Set [VPN-PPTP-TCPIP]

#### IpHeaderCompression

Activates IP Header Compression on the VPN connection level. Range: TRUE/FALSE. Default= TRUE.

#### SwCompression

Activates software compression on the VPN connection level. Range: TRUE/FALSE. Default= FLASE.

#### Slip

Activates *Serial Line Internet Protocol* on the VPN connection level.

Range: TRUE/FALSE. Default= FLASE.

#### SpecificIpAddr

Determines if a special IP address should be used on the VPN connection level.

Range: TRUE/FALSE. Default= FLASE.

#### IpAddr

Determines the IP address for the VPN connection level. If *SpecificIpAddr=FALSE* this parameter will be ignored.

#### NameServers

Determines if special IP addresses should be used for the name resolution on the VPN connection level.

Range: TRUE/FALSE. Default= FLASE.

#### ipaddrDns

IP address of the DNS server on the VPN connection level. (will be ignored if *NameServers=FALSE*)

#### ipaddrDnsAlt

Alternative IP address of the DNS server on the VPN connection level. (will be ignored if *NameServers=FALSE*)

#### ipaddrWins

IP address of the WINS server on the VPN connection level. (will be ignored if *NameServers=FALSE*)

#### ipaddrWinsAlt

Alternative IP address of the WINS server on the VPN connection level. (will be ignored if *NameServers=FALSE*)

---

## 6.2.12 Setting of the Parameter Set [VPN-PPTP-SECURITY]

### **RequireDataEncryption**

Determines if it is compulsory that the VPN server must offer data encryption. Range: TRUE/FALSE. Default= TRUE.

### **PAP**

Determines if the password authentication protocol (PAP) should be accepted on the VPN connection level. Range: TRUE/FALSE. Default= FLASE.

### **CHAP**

Determines if the challenge handshake authentication protocol (CHAP) should be accepted on the VPN connection level. Range: TRUE/FALSE. Default= FLASE.

### **MsCHAP**

Determines if the Microsoft challenge handshake authentication protocol (MSCHAP) should be accepted on the VPN connection level. Range: TRUE/FALSE. Default= FLASE.

### **MsCHAP2**

Determines if the Microsoft challenge handshake authentication protocol Version 2.0 (MSCHAPv2) should be accepted on the VPN connection level. Range: TRUE/FALSE. Default= FLASE.

## 6.2.13 Setting of the Parameter Set [VPN-PPPT-PARAMS]

### **TcpPortNumber**

TCP port number of the VPN server's PPTP access. Range: 1...65535. Default 1723

### **UdpPortNumber**

UDP port number of the VPN server. Range: 1...65535. Default 47

### **MaxWanEndpoints**

Maximum number of PPTP sessions. Default 5.

### **MaxTransmit**

Maximum number remaining PPTP send packets. Default 32.

### **InactivityIdleSeconds**

Timeout interval in seconds, after which a CCP packet (control channel packet) will be sent to the VPN server if during this time no packet has been transmitted via the tunnel. Default = 60.

### **AlwaysEcho**

Range: 0/1. Default = 1 (switches the Echo-Modus on).

### **TunnelConfig**

Determines the routing configuration for the client's tunnels. The value should be set to 0.

### **TcpDisconnectTimeout**

Determines the maximum waiting time for the acknowledgement a disconnection on the TCP level. Default = 30.

### **TcpConnectTimeout**

Determines the time-out time for a TCP connection. Default = 30.

### 6.2.14 Setting of the Parameter Set [UNIMODEM-INIT]

#### Init1, Init2, Init3, Init4

The complete configuration of the internal modem is carried out using AT commands, which are sent to the modem by the operating system in the form of initialisation strings, a number of strings commands can be sent consecutively. Each string commences with AT and ends with <cr>.

AT&FE0V1&C1&D2<cr>

&F Load factory settings

E0 Deactivate echo

V1 Reporting in text form

&C1 DCD indicates, that a connection has been established and synchronised.

&D2 After a falling edge of the DTR is detected the communication will be disconnected.

### 6.2.15 Configuration Example

#### Parameter Set [AXRASVPN\_PARAMS]

```
RedialWTimeMS= 15000 // Start a new ISP dial-up attempt every 15 seconds.
FailRedialWTimeMS= 105000 // But after 3 unsuccessful ISP dial-up attempts, wait 105 seconds.
ReConnectWTimeMS= 120000 // Start a new VPN dial-up attempt every 120 seconds.
FailReConnectWTimeMS= 1800000 // But after 3 unsuccessful VPN dial-up attempts disconnect the ISP-connection and wait 30 minutes.
```

#### Parameter set [AXRASVPN\_PARAMS-T2]

```
.
DialRetry=9 // After a total of 9 unsuccessful ISP dial-up attempts...
DialRetryWTimeMin=5 // ...firstly wait for 5 minutes
VpnRetry=9 // After a total of 9 unsuccessful VPN dial-up attempts...
VpnRetryWTimeMin=480 // ... firstly wait for 480 minutes (8 hours).
```

Waiting time for the dial-up connection: Modem to GPRS – provider

Unsuccessful dial-up attempts	1	2	3	4	5	6	7	8	9	10
Time	ts	ts	tb	ts	ts	tb	ts	ts	tdr	

ts = RedialWTimeMS

tb = FailRedialWTimeMS

tdr = DialRetryWTimeMS

Waiting time for the dial-up connection: VPN client (ALU) – VPN server (router)

Unsuccessful dial-up attempts	1	2	3	4	5	6	7	8	9	10
Time	tnc	tnc	trc	tnc	tnc	trc	tnc	tnc	trc	tdr

tnc = ReConnectWTimeMS    trc = FailReConnectWTimeMS    tdr = VpnRetryWTimeMin

### 6.3 Enable the Service Routines and Interfaces

The individual service routines and interfaces are enabled using the PCOS\_A320.INI file. This file can be found in the directory PCOS\_OHP on the CF card.



**Note:** Commencing firmware version 1.70 different filenames are used (PCOSA320.INI)

In the delivered status of the CF card all entries are inactive. By removing the double forward slash // the corresponding entry is enabled.

**Caution:** Only enable the service routines and interfaces that you require for your project. According to the settings in the file, the driver files (DLLs) are loaded into memory of the ALU 320 during booting. Each enabled driver occupies memory space, even when it isn't used.

```
#=====
#
#   ALU320 PCOS Einstellungen
#   ALU320 PCOS settings
#
#=====
[PCOS-SYSTEM]
SharedMemoryAddr=1000
SharedMemorySize=600
TimeSyncDelay=0
#Memory area for Shared-Memory
#Start address ProConOs %MB3.1000
#Number of bytes in total   to %MB3.1599
#Gang reserve in hours (1...72), 0 = off
#=====
#
#   MODBUS Server Einstellungen: Speicher, Schnittstellen
#   max. 4 Schnittstellen möglich
#
#   MMODBUS server settings: memory, interface
#   max. 4 interface are possible
#
#=====
[MODBUS-SERVER]
//SharedMemoryOffset=200
//SharedMemorySize=308
//Interface1= SSERIAL
//Interface2= STCP_IP
#Memory area for Modbus
#Start address
#Number of Bytes in total
#Slave Serial
#Modbus TCP-Server
#=====
#
#   Service-Routinen für IEC 60870-5-101 oder IEC 60870-5-104
#
#   Service Routines for IEC 60870-5-101 or IEC 60870-5-104
#
```

#=====	
[SERVICE] //Service1= SVC104SA //Service2= SVC101SA	#Protocol IEC 60870-5-104 (TCP/IP) #Protocol IEC 60870-5-101 (serial)

**Example:**

By enabling Interface1 and Service2 the example project included on the CF card can be activated, e.g. create a data transfer according to IEC 60870-5-101 via COM1 and communication to a Modbus-Master via COM2 of the ALU 320.

**6.3.1 Definition von Shared Memory**

The start address of the shared memory can begin at 0 (%MB3.000). Please note always commence the Start Address using an even address (0, 2, 4, 200, 1000 etc.)

The length can up to a maximum of 16384. The length should also be an even number.

In the example the user program (AWP) is stored in the shared memory area from %MB3.1000 to %MB3.1599.

These can for example be used to define overlapping variables or structures.

Example:

```
Variable A    BYTE    %MB3.1200
Variable A1   BOOL    %MX3.1200.0
.
.
Variable A8   BOOL    %MX3.1200.7
```

The total Shared Memory area or parts thereof can be used for a Modbus connection.

**6.3.2 Modbus RS232**

If the entry for Modbus is enabled by erasing the // at Interface1, then in the example the memory area %MB3.1200 to %MB3.1507 can be written and read via the Modbus RS232 interface.

```
[MODBUS-SERVER]
SharedMemoryOffset=200
SharedMemorySize=308
```

Interface1= **SSERIAL**

You can also have the total shared memory area at your disposal for Modbus communication. In the example the following settings are possible:

```
[MODBUS-SERVER]

SharedMemoryOffset=0
SharedMemorySize=600
```

Interface1= **SSERIAL**

Via the file MODBUS\_SERVER.INI this area will be designated for coils and registers.

 For additional information see chapter 6.5.

### 6.3.3 Modbus TCP Server

If the entry for Modbus TCP Server is enabled by erasing the // at Interface2, then in the example the memory area %MB3.1200 to %MB3.1507 can be written and read via the Modbus TCP interface.

```
[MODBUS-SERVER]
```

```
SharedMemoryOffset=200  
SharedMemorySize=308
```

```
Interface2= STCP_IP
```

You can also have the total Shared Memory area at your disposal for Modbus communication. In the example the following settings are possible:

```
[MODBUS-SERVER]
```

```
SharedMemoryOffset=0  
SharedMemorySize=600
```

```
Interface2= STCP_IP
```

Via the file MODBUS\_SERVER.INI this area will be designated for coils and registers.



For additional information see chapter 6.5.

### 6.3.4 Service Routines

If communications via IEC 60870-5-101 and/or IEC 60870-5-104 are to be used, the relevant service routines must be enabled. It is possible to start several routines concurrently.

The setting

```
Service1= SVC104SA          (corresponding parameter files SVC_COM1.INI and SVC_RTU1.INI)
```

```
Service2= SVC101SA          (corresponding parameter files SVC_COM2.INI and SVC_RTU2.INI)
```

enables the concurrent transmission data via TCP/IP and a serial connection.

The setting

```
Service1= SVC101SA          (corresponding parameter files SVC_COM1.INI and SVC_RTU1.INI)
```

```
Service2= SVC101SA          (corresponding parameter files SVC_COM2.INI and SVC_RTU2.INI)
```

enables the concurrent transmission data via both serial interfaces on the ALU 320 with IEC 60870-5-101 protocol.

**Comment:** In the delivery status of the CF card the file SVC\_COM1.INI is set for IEC 60870-5-104 protocol. When, as described above, both service routines should operate with serial protocol, copy the contents of the file SVC\_COM2.INI into SVC\_COM1.INI and change the settings according to your requirements. Pay particular attention to the fact that the settings for the ports in both files should be different.



**Note:** Commencing firmware version 1.70 different filenames are used (SVCCOMn.INI)

## 6.4 IEC 60870-5-101 or -104 Communication

### 6.4.1 General

For the individual transmission protocols on the ALU 320 up to 8 service routines can be started. For each service routine parameter data (SVC\_COM?.INI) and the definition of the PV processing (SVC\_RTU?.INI) are assigned.

The protocols IEC 60870-5-101 and IEC 60870-5-104 with slave functionality are available.

The parameter data SVC\_COM?.INI and SVC\_RTU?.INI can be found in the directory PCOS-OHP on the CF card.



**Note:** Commencing firmware version 1.70 different filenames are used (SVCCOMn.INI und SVCRTUn.INI)

**Remark:** The individual parameters included in the files on the CF card can have differing assignments to those represented here.

Change the entries according to your requirements and subsequently save the files as those on the CF card.



**Note:** If an entry is proceeded by a // then the corresponding line will not be processed. This is the same as erasing the line and results in the line not being activated. Parameters that are not required can be erased or deactivated.

The default setting for erased or deactivated parameters is 0, FALSE or OFF.

The entry TRUE corresponds to ON, the entry FALSE corresponds to OFF. Both variants of this expression can be used.

### 6.4.2 Parameter File for Communication Configuration IEC 60870-5-104 (SVC\_COMn)



**Note:** Commencing firmware version 1.70 different filenames are used (SVCCOMn.INI und SVCRTUn.INI)

[IEC-5-104]	
PORT= 2404	Port number <b>Must not be changed!</b>
KVAL= 12	Max. difference between the number of receipt and send sequences
WVAL= 8	Latest acknowledgement after receipt from w APDU in I-format
T1= 15	Time monitoring for transmitted APDU or Test-APDU
T2= 10	Time monitoring for acknowledgements, where no data was transmitted
T3= 40	Time monitoring for transmitted test frames
IP_ADDR=192.168.0.1	IP address of the remote station

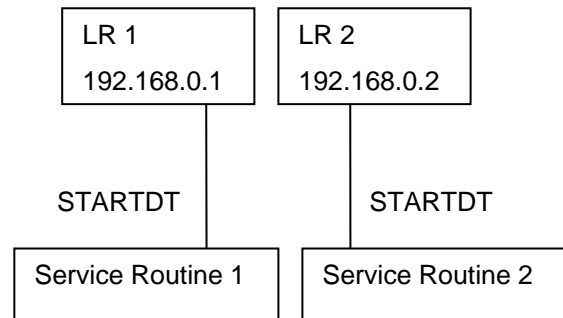
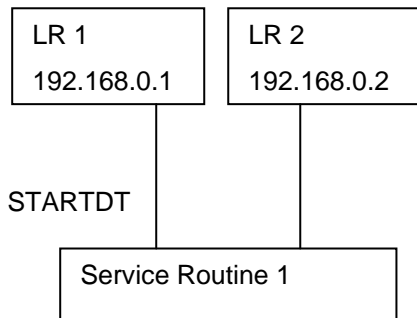
Using a service routine a maximum of two IEC 60870-5-104 connections can be created. However STARTDT can only be activated in **one** connection. In the second connection only the Testframes (TESTFR) will be confirmed.

You can enter the IP addresses of the communication partner (max. 2). Herewith is guaranteed that the ALU320 only allows a TCP/IP connection to these addresses. Connection attempts from other IP addresses will not be confirmed.

If no IP addresses are entered then a connection can be established with any IP address in the same subnet.



**Note:** If a redundant data transmission is required using two connections with activated STARTDT, then two service routines must be programmed.



Data will only be exchanged with LR1

On LR2 only TESTFR will be answered

A change can only be induced with STOPDT on LR1 and subsequently STARTDT on LR2.

Data will be exchanged with both LRs

### 6.4.3 Parameter File for the Communication Configuration IEC 60870-5-101 (SVC\_COMn.INI)

Open the file SVC\_COM.INI which you have stored on your computer using the editor.



**Note:** Commencing firmware version 1.70 modified file names are used (SVCCOMn.INI)

You will now see the following entries:

<pre> [IEC-5-101] AWD= FALSE PORT= 1 BAUDRATE= 9600 PARITY= EVEN DATA=8 STOPBIT=1 TVS2=25 TNS2=7 TVM2=40 TNM2=150 PAUSE=27           </pre>	<pre> TRUE = ADS operation, FALSE = Leased line operation Port =1 or 2 corresponds to COM interface. Transmission rate to master station Parity NO / EVEN           (with leased line always EVEN) Number of data bits        <b>Must not be changed!</b> Stop bit                   <b>Must not be changed!</b> Lead time S2 [ms]         (only relevant with leased line) Follow-up time S2 [ms]    (only relevant with leased line) Lead time monitoring M2 [ms] (only relevant with leased line) Follow-up monitoring M2 [ms] (only relevant with leased line) Pause time [ms]           </pre>
---	---

BUS_TOUT= 2 POLL_TOUT= 60 S2SIGNAL= OFF M2SIGNAL= OFF CONF_SC= OFF AVS= 3 OKTETT_AV= 2	Bus monitoring (Slave) in n*[100ms] Monitoring poll cycle (Slave) in n*[100ms] S2-Signal (RTS) switch on or off M5/M2 monitoring (DCD/CTS) switch on or off ON= Receipt confirmation with single character E5, OFF = Short telegram Address of the connection layer Octet number of the AVS (0 to 2)
[AWD101-MODE] AT_INIT= ATZ0 AT_HANGUP= ATH AT_DIAL= ATD AT_PICK_UP= ATA	!! AT instruction ASCII, max. 8 characters!! Load stored profile into modem AT instruction for hanging-up AT instruction for dialling AT instruction for picking-up
# Modem RESPONSE/Response decimal	
//RSP_OK= 0 //RSP_CONNECT= 1 //RSP_RING= 2 //RSP_NOCARRIER= 3 //RSP_ERROR= 4 //RSP_NODIAL= 5 //RSP_BUSY= 6 //RSP_NOANSWER=7 RSP_TOUT= 2	Here the customary values for modem response are given. If the modem you are using deviates from these values, then delete the double-slash „//“ before the entry and replace with your required value.  Maximum waiting time for an AT instruction acknowledgement (1 to 15 Sec.)
# Dial parameters / Connection structure	
DIAL_NUMBER= ?????????? DIAL_RETRY= 3 DIAL_TM_REDIAL= 30 DIAL_TM_RIDLE= 1 DIAL_TM_CON= 60 DIAL_TM_PW= 2  DIAL_PRI_PW= ???????? DIAL_SEK_PW= ???????? MODEM_LED=FALSE	Telephone number of the control station Re-dialling (1 to 7) Time [sec]: Pause when re-dialling (1 to 255) Time [min]: off-time after re-dialling (1 to 255) Time [sec]: waiting time for connection "Connect" (1 to 255) Time [sec]: waiting time until password is sent after establishing connection  Password primary station (master station) maximum 16 characters Password secondary station (substation) maximum 16 characters With TRUE the display for the internal COM3 will be switched on

**Comment:** The individual parameters can differ from those in this representation.

### 6.4.3.1 Explanation of the Individual Parameters in Leased Line Operation PORT

Here you declare the COM interface (1 or 2), which you want to use to connect to the IEC 60870-5-101.

---

## BAUDRATE

The baudrate can be set to values between 600 and 9600 Baud.

## PARITY, DATA, STOPBIT

When using leased-line or direct serial communication the parity should be set to EVEN. The number of data bits must always be 8 and the number of stop bits 1.

## TVS2, TNS2, TVM2, TNM2, PAUSE, S2SIGNAL, M2SIGNAL

Via these parameters you can set the lead-time and follow-up time for S2 and the monitoring time for the M2 signal as well as the pause time between the telegrams. Above all these settings are necessary when using leased-line modems (UEM 202 or 201). Using the parameter S2SIGNAL, M2SIGNAL, the setting respectively the monitoring of the corresponding signals can be switched-off.

## POLL\_TOUT

Within this monitoring time the station must be polled, otherwise the connection will be reported as being faulty and can only be reactivated using "reset the remote link".

## BUS\_TOUT

The monitoring **BUS\_TOUT** declares the monitoring time when running line operation, within which time the local station or another station must be polled. If the station runs into the timeout, it waits for an initialisation of the connection layer.

With point-to-point or ADS operation the **BUS\_TOUT** should not be set to a smaller value than the **POLL\_TOUT**.

## CONF\_SC

Via this parameter you declare how the substation should respond when no data telegram is ready for transmission. ON= receipt acknowledgement with end-character E5, OFF= short telegram (telegram of fixed length).

## AVS, OKTETT\_AVS

AVS is the address of the connection layer and the stations are polled using this. The address can be 1 or 2 octets (byte) long. The setting must be designated for each station in the system via a compatibility list.

### 6.4.3.2 Explanation of the Individual Parameters in ADS Operation

#### PARITY, DATA, STOPBIT

With most dial-up modems or ISDN TAs the transmission is with 8 data bits without the possibility of parity. Hence, set PARITY= NO. This setting must also be configured on the device at the other end.

#### AT\_INIT, AT\_INIT2, AT\_INIT3

Up to 3 initialisation sequences for the dial-up modem can be entered, each with 28 characters. It is especially important for operation with internal modems that they are not initialised using external tools.

Settings for internal modems:

AT\_INIT=ATE0X0

AT\_INIT2=ATS0=0V0

**Comment:** AT\_INIT2 and AT\_INIT3 are available as of ALU firmware version 1.90.

**RSP\_TOUT** is the monitoring time during which the modem response (CONNECT, OK etc.) should be received. The default setting is suitable for most modem types and therefore must not be changed.

If an initiated connection to a substation is not achieved within the time given in the parameter **DIAL\_TM\_CON** then the dialling attempt will be aborted.

After the waiting time **DIAL\_TM\_REDIAL** has elapsed, a new attempt will be initiated. This will be repeated until a connection is achieved or until the number of retries entered in the parameter **DIAL\_RETRY** is reached.

If a connection is not achieved after the set number of dial-up attempts a timer is initiated. The time setting is entered in the parameter **DIAL\_TM\_RIDLE**. After this time has elapsed the procedure described above will be repeated anew.

The password for the central station (**DIAL\_PRI\_PW**) and the substation (**DIAL\_SEK\_PW**) can be a maximum of 16 characters long. If fewer characters are given, it will automatically be filled with blanks. Upper and lower case characters are checked. The given passwords must also correspond with those configured in the central station.

After a successful connection is achieved (connect from the modem), the central station or the substation will send its password. The password will always be sent from the side that initiated the connection and checked by the receiving station for authenticity. If the check is successful, polling will be commenced. If unsuccessful, the connection will be terminated with ATH.

### MODEM\_LED

This switch can be set to TRUE when an ALU with integrated modem for dial-up operation is being used. If activated, the 3<sup>rd</sup> LED has the following meaning.

LED	Meaning
off	No connection
Continuous	Data connection, password OK, IEC communication
Blinking - slow	Connection establishment initiated by the ALU
Blinking - fast	Connection establishment initiated by the remote station (RING)
Blinks twice (500ms) - pause	Modem Initialisation problem. In this case, the entries in AT_INIT, AT_INI2, AT_INI3 in the SVC_COMn.INI file should be checked.
Blinks three times (300ms) - pause	Modem no longer ready.

**Comment:** The function MODEM\_LED is only available as of ALU firmware version 1.91.

### 6.4.4 Parameter File for Processing the Data Point Definition (SVC\_RTUn.INI)



**Note:** Commencing firmware version 1.70 different filenames are used (SVCRTUn.INI).

[RTU_I10X] ASDU= 3 OKT_ASDU= 2 OKT_HERK=TRUE OKT_AINF= 3	ASDU = Station address Number of octets in the address byte Reason for transmission with origin address Number of octets in the information object number
APDU_LEN=253 BLOCK_OBJEKT= TRUE BLOCK_ELEMENT= TRUE	Maximum length of the block telegram Blocking of the information objects permissible TRUE / FALSE Blocking of the information elements permissible TRUE / FALSE
IEC_CCI_TYP= 0	Counter value enquiry   enquiry   operational mode [C   D   O ]
RTU_COM_STOP=TRUE	Behaviour during a stop of the user program, TRUE= communication stopped, FALSE= continues running
EXEC_CMDNENABLE = TRUE	Instructions with quality identifier QU=0 are allowed Instruction execution time for instructions without additional definitions

EXEC_CMDNTIME = 10  EXEC_STIME= 3 EXEC_LTIME= 50	(QU=0) Shot instruction execution time (n * 100 ms) Long instruction execution time (n * 100 ms)
1:AP_NMB= 3600 1:AP_WARN= 2500	Number of telegrams in the archive for PVs with time tag (0 to 3600) Overflow warning of the archive buffer (by ADS operation leads to communication establishment).
OOFS_SP=100 OOFS_SP_TM=100 OOFS_DP=200 OOFS_DP_TM=200 OOFS_BO=300 OOFS_BO_TM=300 OOFS_ME=400 OOFS_ME_TM=400 OOFS_IT=500 OOFS_IT_TM=500	Object numbers - Offset single message (Single Point) Object numbers - Offset single message (Single Point) with time tag Object numbers - Offset double messages (Double Point) Object numbers - Offset double messages (Double Point) with time tag. Object numbers - Offset bit string Object numbers - Offset bit string with time tag Object numbers - Offset measured value Object numbers - Offset measured value with time tag Object numbers - Offset counter value Object numbers - Offset counter value with time tag
OOFS_SC= 600 OOFS_DC=700 OOFS_SE=800 OOFS_CBO=900	Object numbers – Offset single instruction (Single Command) Object numbers - Offset double instruction (Double Command) Object numbers - Offset set value standardised (Set-point command) Object numbers - Offset bit string (bit string of 32 Bit))
# Internal instructions for the variables (data processing)	
# Data version Var1	
1:VAR_XX_ABF= TRUE 1:VAR_XX_SPO= TRUE 1:VAR_XX_AWD= FALSE 1:VAR_XX_AP= 1 1:VAR_XX_GRP=1	Interrogation active, general interrogation or counter interrogation Enables spontaneous transmission 1= by ADS operation commences establishment of communication Cyclic in the archive buffer 0 = no, archive 1...3 Value will be transmitted via GI with group enquiry 1
# Internal instructions for the variant 1 only for double messages (DP)	
1:VAR_DP_TS=100	Faulty state information suppression time, 0 = no monitoring, n * 10 ms
# Internal instructions for the variant 1 only for bit string (BO)	
1:VAR_BO_NMB=32	Number of process points, 1...32 per bit string
# Internal instructions for the variant 1 only for integral totals (IT)	
1:VAR_IT_IMPS= 0	Impulse threshold 0 to 65535 (def. = 0, off)
# Internal instruction for variant 1 only for integral totals (IT) and measured values (ME)	
1:VAR_ITME_ZYK = 5	Difference and cycle time= n* minute, 0 to 3600, 0 = off
# Internal instruction for the variant 1 only for measured values (ME)	
1:VAR_ME_AZI=512	Deviance time integral, AZI = 0... 32760 / 0x7FF8

	12 bit measured value + sign, left aligned (AZI = 8 corresponds to the change of one digit on the input)
1:VAR_ME_OV=32000	Over range (OV) from 0 to 32767 (0x7FFF)
# Internal instruction for the variants (data processing) # Data variant Var2	
2:VAR_XX_ABF= TRUE 2:VAR_XX_SPO= TRUE 2:VAR_XX_AP= 0 2:VAR_XX_GRP=2	Enquiry active, general enquiry or counter enquiry Spontaneous Cyclic in the archive buffer 0 = no, archive 1...3 Value will be transmitted via GI with group enquiry 2
# Internal instruction for the variant 2 only for integral totals (IT)	
2:VAR_IT_IMPS= 20	Impulse threshold 0 to 65535 (def. = 0, off)
# Internal instruction for variant 2 only for integral totals (IT) and measured values (ME)	
2:VAR_ITME_ZYK = 0	Difference and cycle time= n* minute, 0 to 3600, 0 = off
# Internal instruction for the variant 2 only for measured values (ME)	
2:VAR_ME_AZI=256	Deviation time integral, AZI = 0... 32760 / 0x7FF8 12 bit measured value + sign, left aligned (AZI = 8 corresponds to the change of one digit on the input)
2:VAR_ME_OV=32000	Over range (OV) from 0 to 32767 (0x7FFF)
# Internal instruction for the variants (data processing) # Data variant Var3	
3:VAR_XX_ABF= TRUE 3:VAR_XX_SPO= FALSE 3:VAR_XX_AP= 1	Enquiry active, general enquiry or counter enquiry No spontaneous transmission Cyclic in the archive buffer 0 = no, archive 1...3
# Internal instruction for variant 3 only for integral totals (IT) and measured values (ME)	
3:VAR_ITME_ZYK = 5	Difference and cycle time= n* minute, 0 to 3600, 0 = off
# Internal instruction for the variant 3 only for measured values (ME)	
3:VAR_ME_AZI=0	Deviance time integral, AZI = 0... 32760 / 0x7FF8 12 bit measured value + sign, left aligned (AZI = 8 corresponds to the change of one digit on the input)
	<b>Up to 16 different processing variants can be defined</b>
# Logical operation assignment of process data in the monitoring direction	
QX0.0:TAG_SP_TM= 1;V1	Single event with time tag
QX0.0:TAG_SP= 1;V1 QX0.1:TAG_SP= 2;V1	Single event without time tag

QX0.4:TAG_DP=1;V1	Double event
QD4:TAG_BO=1;V2	Bit string 32 Bit
QW8:TAG_ME= 1;V1; QW10:TAG_ME= 2;V2; QW12:TAG_ME_TM= 3;V1; QW14:TAG_ME_TM= 4;V2;	Measured value without time tag defined as WORD  Measured value with time tag defined as WORD
QD16:TAG_IT= 1;V1 QD20:TAG_IT_TM= 2;V1 QD24:TAG_IT= 3;V2 QD28:TAG_IT_TM= 4;V2	32 bit integrated value defined as DWORD
# Logical operation assignment of process data in the control direction	
IX0.0:TAG_SC= 1;V1 IX0.1:TAG_SC= 2;V1	In the 1 <sup>st</sup> transmission byte array two single commands are entered
IW2:TAG_SE=1;V1 IW4:TAG_SE=2;V1	In the 2 <sup>nd</sup> and 3 <sup>rd</sup> bytes of the transmission array a set value are entered In the 5 <sup>th</sup> and 6 <sup>th</sup> bytes of the transmission array a set value are entered

### Explanation of the individual parameters

#### ASDU, OKT\_ASDU:

The **Application Service Data Unit** is the station address. The length is only variable with IEC 60870-5-101, when using IEC 60870-5-104 it is preset to the value 2.

#### OKT\_HERK:

Declares whether the transmission reason will be transmitted with or without the source address. In IEC 60870-5-104 the octet for the source address is always included, for the ALU 320 it is fixed to ZERO.

#### OKT\_AINF:

Address length for the information object. It can be 1, 2 or 3 bytes long. The length is only variable with IEC 60870-5-101, with IEC 60870-5-104 it's fixed with 3 bytes.

#### APDU\_LEN= 30

The maximum length of the APDU (Application Protocol Data Unit = telegram) can be set. With IEC-870-5-104 the maximum length is 253 bytes (255 less the start and length octet)

#### BLOCK\_OBJEKT

This switch determines whether the transmission of a number of PVs as blocked information objects is permissible.

The default setting= TRUE, because it enables an optimised transmission of the PVs to be achieved especially with GAs. The switch should only= FALSE when the superior system can process this type of block telegram

---

## **BLOCK\_ELEMENT**

This switch determines whether the transmission of a number of PVs as blocked information elements is permissible.

The default setting= TRUE, because it enables an optimised transmission of the PVs to be achieved especially with GAs. The switch should only= FALSE when the superior system can process this type of block telegram.

## **IEC\_CCI\_TYP= 0**

This standard enables the transmission of integral totals in one of 4 modes - the operational modes A, B, C and D. Mode A can be realised by using the parameter VAR\_ITME\_ZYK in the definition of the processing variants.

With modes B, C and D a counter enquiry instruction (CI counter interrogation) will be sent from the superior system. The ALU 320 supports only operational modes C and D as well as an OHP specific procedure (O) in which the integral totals are immediately transmitted after a restore instruction (counter interrogation command with FRZ= 1, RQT= 1...5) where the cause of transmission was 37 ... 41. This is a variation on the operational mode C (where a counter enquiry instruction FRZ=0, RQT=1...5 is not sent).

## **RTU\_COM\_STOP**

This switch determines whether the communication to the master station should be discontinued or not when the user program is stopped.

FALSE= the communication driver continues to operate and poll telegrams are answered with a short telegram or E5. When the user program is stopped all defined PVs are transmitted with invalid bit set. If the user program is restarted, the invalid bits will be reset and the PVs transmitted.

## **RTU\_TIME\_SET**

0 = Time may be set via IEC (default)

1 = Time may never be set via IEC

2 = Time may only be set via IEC, when the current time IV is invalid

## **RTU\_TIME\_ACK\_NEG**

FALSE = IEC time telegram is confirmed with a positive response (default)

TRUE IEC time telegram is confirmed with a negative response

This parameter is only valid when RTU\_TIME\_SET is not equal to NULL!

## **RTU\_TIME\_SYNC\_ENABLE**

Entry in minutes (1...65535)

Delta time designation in minutes, since the last setting of the clock. After it has elapsed the clock can be set using an IEC telegram.

This parameter is only valid when RTU\_TIME\_SET = 2!

## **EXEC\_CMDNENABLE, EXEC\_CMDNTIME:**

When the parameter EXEC\_CMDNENABLE = TRUE, then instructions with the quality identifier QU= 0 (without additional definition) will be received and then transmitted after the time set in the parameter EXEC\_CMDNTIME. If the parameter EXEC\_CMDNENABLE = FALSE, then the instructions with QU=0 will be negatively confirmed and not executed.

This setting is normally used when the instruction execution times in the station are individually chosen for each instruction. If the user program is realised in this way, then it should be noted, that the ALU 320 "Termination of Activation" (cause of transmission 10) is always transmitted after the time set in EXEC\_CMDNTIME has elapsed.

## **EXEC\_STIME, EXEC\_LTIME:**

Here the short and long instruction output times are defined. At what time the instructions are outputted, is defined via the instruction identifier in the telegram from the central station. Time base: n \* 100 ms.

---

## ?:AP\_NMB, ?:AP\_WARN:

The number of telegrams that the archive buffer should hold is defined in (AP\_NMB).

The parameter 1:AP\_WARN is only active in ADS operation. It declares at what buffer level the substation automatically initiates a connection to the superior system.

A maximum of 3000 telegrams can be stored. The number can be divided in up to 3 archives.

## OOFS\_??:

Via this parameter you can enter an offset for each data type, it will be added to the object numbers declared during input definition. Use of the offsets enables **all** object numbers of a specific data type to be shifted to another area.

In the example above the first 2 bits in first transmission byte are transmitted as single messages with the object numbers 101 and 102 because the parameter OOFS\_SP has a value of 100.

Bits 4 and 5 are transmitted as a double message with the object number 201.

Bytes 4...7 of the transmission block (defined as DWORD) are transmitted as a bit string with the object number 301.

The 4 measured values (defined as WORD), which are transmitted, commencing byte 8, will each be transmitted with and without a time tag in the object numbers 401, 402, 403 and 404.

The 4 integral totals (defined as DWORD), for transmission commencing byte 16, are transmitted with the object numbers 501, 502, 503 and 504.



**Note:** It is recommended that the offsets for PVs with or without time tags should be set identically and to number the inputs consecutively. In this way an optimal data transmission is achieved during a general enquiry.

## Processing variants:

Up to 8 different processing variants (1:VAR..., 2:VAR... etc.) can be defined. These are subsequently assigned to the individual inputs. There are parameters that are valid for all data types (identified with XX), others relate to specific data types (e.g. VAR\_IT\_IMPS valid only for integral totals).

### ?:VAR\_XX\_ABF

TRUE = Process variable will be transmitted during a general enquiry (global).

### ?:VAR\_XX\_GRP=1

Via this parameter a general enquiry can be undertaken as a group enquiry (group 1 ... 16).

### ?:VAR\_XX\_SPO

TRUE= Process variable is spontaneously transmitted on a change of state.

### ?:VAR\_XX\_AWD

TRUE= Initiation of a connection to the central station on a change of the process variables.

### ?:VAR\_XX\_AP

Write process variables periodically into the archive. 1= yes, 0= no.

### ?:VAR\_BO\_NMB

Via this parameter you can declare the number of consecutive bits in a transmission array that will be consolidated into a bit string. Area: 1...32.

---

### ?:VAR\_IT\_IMPS

Impulse threshold in the range 0 to 65535. On reaching the configured threshold the integral value is spontaneously transmitted (cause of transmission 3). Setting 0 means, that the integral value will only be transmitted by integral value interrogation or, if configured, periodically.

### ?:VAR\_ITME\_ZYK

Defines the time interval for the periodic transmission of integral and measured values. The values will be transmitted according to the settings in the parameters OFFS\_IT / OFF\_ME or OOFS\_IT\_TM / OOFS\_ME\_TM, with or without time tag and the cause of transmission 3 (spontaneous).

Entries in minutes in the range between 1 to 3600. Entry 0= no periodic transmission.

### ?:VAR\_ME\_AZI

Setting the deviation time integral. The measured value changes are recorded and added together according to their sign at 1 second intervals, commencing at the moment of the last transmission. If the sum of these additions reaches the AZI setting the measured value will be transmitted with the cause of transmission 3 (spontaneous). Using this method the measured value transmission will be dampened down. Slowly rising measured values will be transmitted with a time lag and on the other hand fast rising measured values transmitted more quickly.

If the setting AZI=0 and the parameter VAR\_XX\_SPO=TRUE, then every measured value change will be transmitted. Consider here that the measured value inputs generally fluctuate.

### ?: VAR\_ME\_OV

Here you can declare the limit of the over range. If the measured value exceeds the preset value, then the over range bit will be set in the quality identifier. Setting 0 means no monitoring.

Measured values are taken over 1:1 from the transmission array and entered in the telegram. It is the responsibility of the user program to ensure that the measured values are presented in the standardised format.

### Definition of the transfer block:

Syntax: reference transfer field: IEC type= object number; processing variant

The reference on the transfer field can be entered as a QX?.? (BIT), QB (BYTE), QW (WORD) or QD (DWORD). In the command direction IB?.?, IB, IW and ID are used.

The reference is **relative** and always correlates to the transfer field defined for the service routine. If, in the I/O configuration of MULTIPROG, the output for this service is for example defined as %QB501 to %QB628, then during declaration of the transfer field, QB0 will define the contents of %QB501, QB1 of %QB502, etc.

Maximum length of the transfer blocks:

Command direction	256 Byte
Reporting direction:	512 Byte

Messages are defined using QX?.?. With double messages, the first of the two consecutive bits is given as reference.

Measured values (16 bit) are defined using QW.

Bit strings and counter values (32 Bit) are defined using QD.

Commands are defined using IX?.? Whereby with double commands, the first of the two consecutive bits is given as reference.

Set values (16 bit) are defined using IW.

Bit strings (32 bit) are defined using ID.

---

Example:

QX0.0:TAG\_SP\_TE= 1;V1

QX0.0:TAG\_SP= 1;V1

The 1<sup>st</sup> bit in the transfer block will be transmitted as a **Single Point** either with or without a time tag. For transmission parameters defined in the variant 1 apply.

QD16:TAG\_IT= 1;V1

Bytes 16 to 19 of the transferred integral value are transmitted with parameters defined in the variant 2. In the example always after 5 minutes and on counter interrogation.

QD28:TAG\_IT= 4;V2

Bytes 28 to 31 of the transferred integral value are transmitted with parameters defined in the variant 2. In the example always after 5 counter pulses with time tag and on counter interrogation.

### TAGs of the IEC Type:

TAG_SP	<b>Single point</b> information
TAG_SP_TM	<b>Single point</b> information with <b>time tag</b>
TAG_DP	<b>Double point</b> information
TAG_DP_TM	<b>Double point</b> information with <b>time tag</b>
TAG_IT	<b>Integrated total</b>
TAG_IT_TM	<b>Integrated total</b> <b>time tag</b>
TAG_ME	<b>Measured value</b> normalized
TAG_ME_TM	<b>Measured value</b> normalized with <b>time tag</b>
TAG_BO	<b>Bit string of 32 Bit</b>
TAG_BO_TM	<b>Bit string of 32 Bit</b> with <b>time tag</b>
TAG_SC	<b>Single command</b>
TAG_DC	<b>Double command</b>
TAG_SE	<b>Set-point command</b> normalized value
TAG_CBO	<b>Bit string of 32 Bit</b> command direction)

### 6.5 Configuration of the Initialisation File for Modbus RS232-Slave or TCP-Server

The file MODBUS\_SERVER.INI is stored in the directory PCOS\_OHP and is used both for Modbus RS232-Slave and also for Modbus TCP-Server.



**Note:** Commencing firmware version 1.70 different filenames are used (MBSRV.INI)

<pre>#===== # MODBUS SERVER / SLAVE Konfigurationsdatei # MODBUS SERVER / SLAVE configuration file #=====  #----- # Interface 1 - SERIAL - Seriell #-----  # Communication parameter: Modbus slave serial Interface - Service 1</pre>	
<pre>[IF1-COM-MODBUS-SSERIAL] PORT= 2 BAUDRATE= 9600 PARITY= EVEN DATA=8 STOPBIT=1 PAUSE=10 BUS_TOUT= 2000 ASCII= FALSE SLAVE_ADR= 1</pre>	<pre># PORT [xx], 0 ... off  # Stop bit (1= default) # minimum waiting time [ms] between requests and responses # Bus monitoring (Slave) in n*[ms], def. (2000), range1000 to 2000 # Protocol ASCII, FALSE/TRUE; (def. FALSE -&gt; RTU)</pre>
<pre># Server parameter for Modbus slave serial interface - Service 1</pre>	
<pre>[IF1-SERV-MODBUS] REF_BASEINDEX= 0 STATUS_OFFSET= 300 STATUS_LEN= 6</pre>	<pre># 0/1 default (0)  # 2, 4 or 6 bytes long</pre>
<pre># 0x COILS - Read/Write - 1 bit access REF0_OFFSET= 0 REF0_LEN=50</pre>	<pre># REF0: Offset in byte, Modbus server # Length in byte (50 x 8 COILS)</pre>
<pre># 1x Discret Input - Read - 1 bit access REF1_OFFSET= 50 REF1_LEN=50</pre>	<pre># REF1: Offset in Byte, Modbus server # Length in Byte (50 x 8 discrete inputs)</pre>
<pre># 3x Input Register - Read - 16 bit access REF3_OFFSET= 100 REF3_LEN=100</pre>	<pre># REF3: Offset in Byte, Modbus server # Length in Byte (100 corresponds to 50 input registers)</pre>
<pre># 4x Holding Register - Read/Write - 16 bit access REF4_OFFSET= 20 REF4_LEN=100</pre>	<pre># REF4: Offset in Byte, Modbus server # Length in Byte (100 e corresponds to 50 holding registers)</pre>
<pre>#----- # Interface 2 - Modbus TCP/IP Server (MODBUS-STCP_IP) #-----  # Communication parameter: Modbus - STCP_IP interface 2</pre>	
<pre>[IF2-COM-MODBUS-STCP_IP] //IP_ADDR= 192.168.0.101 //IP_ADDR= 192.168.0.102</pre>	<pre>#IP address of the participant #IP address of the participant #IP address of the participant</pre>

//IP_ADDR= 192.168.0.103 //IP_ADDR= 192.168.0.104	#IP address of the participant
# Server parameter for Modbus TCP interface 2	
[IF2-SERV-MODBUS] REF_BASEINDEX= 0 STATUS_OFFSET= 300 STATUS_LEN= 6	# 0/1 default (0)  # 2, 4 or 6 bytes long
# 0x COILS - Read/Write - 1 bit access REF0_OFFSET= 0 REF0_LEN=50	# REF0: Offset in Byte, Modbus server # Length in byte (50 x 8 coils)
# 1x Discrete Input - Read - 1 bit access REF1_OFFSET= 50 REF1_LEN=50	# REF1: Offset in Byte, Modbus server # Length in byte (50 x 8 discrete inputs)
# 3x Input Register - Read - 16 bit access REF3_OFFSET= 100 REF3_LEN=100	# REF3: Offset in Byte, Modbus server # Length in byte (100 corresponds to 50 input registers)
# 4x Holding Register - Read/Write - 16 bit access REF4_OFFSET= 20 REF4_LEN=100	# REF4: Offset in Byte, Modbus server # Length in byte (100 corresponds to 50 holding registers)

### 6.5.1 Explanation of the Individual Parameters

With [IF1... ], [IF2... ], [IF3... ] etc. is specified in which interface the ensuing parameters refer to.

#### [IF1-COM-MODBUS-SSERIAL]

[IF-1-COM-MODBUS-SSERIAL] declares, that this setting is valid for interface 1 with a serial Modbus communication.

#### PORT

Here you declare COM interface 1 or 2, which you want to use for Modbus RS232 slave communication.

#### BAUDRATE

The baudrate can be set between 600 and 9600 baud.

#### PARITY, DATA, STOPBIT

Parity can be selected between EVEN or ODD. The number of data bits must always be 8 and the stop bits 1.

#### PAUSE

The pause time in ms between polling and receiving an answer. Via this parameter the reply telegram can be delayed, when the master is not able to immediately receive an answer.

#### BUS\_TOUT

Bus monitoring in ms. Range 1000 to 2000, default setting 2000 (2 seconds).

The slave checks, whether during the set monitoring time, bus telegrams have been transmitted. If this is not the case, the status "communication fault" is set.

#### ASCII

Switch for Modbus-ASCII or Modbus-RTU. At this time **only** Modbus-RTU is possible, setting ASCII=FALSE.

#### SLAVE\_ADR

Here you set the slave respectively the node address.

---

## [IF1-SERV-MODBUS]

The following instructions and parameters apply for the interface 1 and define the range over which Modbus can be read and written.

### REF\_BASEINDEX= 0

Here you declare, whether the Modbus master addresses the 1<sup>st</sup> coil respectively register. The default setting is 0.

### STATUS\_OFFSET, STATUS\_LEN

Using both of these parameters you define, from which MB3.???? address the Modbus status will be stored and how many bytes should be displayed for the status. The length can be set to 2, 4 or 6 bytes.



For additional information see chapter 5.5.5.

The shared memory area reserved for Modbus communication can be divided into bit and word areas. For this, both an offset and a length can be given. These areas can then be read and written to using the corresponding function codes.

In a similar way to other Modbus configurations these areas are defined as follows:

- 0 reference = Output or marker bits
- 1 references = Input bits
- 3 references = Input words
- 4 references = Output or marker words

### REF0\_OFFSET, REF0\_LEN

This area can be processed using the following function codes:

- FC 1 Read coils
- FC 5 Write single coil
- FC 15 Write multiple coils

### REF1\_OFFSET, REF1\_LEN

This area can be processed using the following function codes:

- FC 2 Read discrete inputs.

### REF3\_OFFSET, REF3\_LEN

This area can be processed using the following function codes:

- FC 4 Read input register.

### REF4\_OFFSET, REF4\_LEN

This area can be processed using the following function codes:

- FC 3 Read holding register
- FC 6 Write single register
- FC 16 Write multiple registers
- FC 23 Read/write multiple register

This area can also be configured overlapping:

With the setting

REF0\_OFFSET= 0  
REF0\_LEN=50

REF4\_OFFSET= 0  
REF4\_LEN=50

the first 50 bytes of the shared memory can be read respectively written with the function codes FC1, FC5 and FC15 as well as with function codes FC3, FC6, FC16 and FC23.

#### **[IF2-COM-MODBUS-STCP\_IP]**

[IF-2-COM-MODBUS-STCP\_IP] declares that this setting is valid for interface 2 with Modbus TCP server communication.

#### **IP\_ADDR= ??????**

In cases where the TCP connection should only be enabled for certain clients, their IP addresses can be entered here. Four connections can be simultaneously established. If no IP addresses are declared, then the first four clients have access.

#### **[IF2-SERV-MODBUS]**

The following instructions and parameters apply for the interface 2 and define the range over which Modbus can be read and written.

For a description of the base index, status and references see above.

Using the example configuration the same memory area can be read and written to via a serial Modbus connection and also a TCP/IP connection, when both interfaces are enabled in PCOS\_A320.INI.

Both connections can however be assigned with separate areas. This should be considered correspondingly in the shared memory and in the definition of the references.

---

## 7 Firmware Library

### 7.1 General

The library ALUX2X\_OHP\_??? has been developed for Multiprog version 4.6. The firmware library will be revised after the changeover to Multiprog version 4.8 and the integration of the function “Send Changes” (online change). The new library has the name ALUX2X\_OHP4\_???.

When installing OHP Add-Ons both versions will be installed thereby maintaining the compatibility for your older projects. The new project submittals will however only contain the newer version ALUX2X\_OHP4\_???.

You must exchange the library when you wish to use “send changes” with an older project. For this purpose click the *Libraries* entry in the *Project Tree* menu with the right-hand mouse button. Afterwards using *Insert – Firmware Library* the new library can be incorporated.



**Path for the FW library:** [Lw]:\Programme\OHP\Multiprog\_48\Multiprog\PLC\FW\_LIB\

After selection with the mouse, the old libraries can be deleted using the *Del* button on the keyboard.

A requirement for using online changes is that the project already uses the new resources and that the ALU320 has a firmware version 3.00 or later.



For additional information see also **Part II Configuration and Programming**, chapter 8.1.

After exchanging the libraries the project must be completely translated (Code –) and transferred, subsequently the “Send Changes” function will be available.

#### OHP has provided the following libraries:

##### For PROCONOS 3

ALUX20\_OHP\_SCOM

ALUX20\_OHP\_SEAB

ALUX20\_OHP\_SYS

##### For PROCONOS 4 (online change)

ALUX20\_OHP4\_SCOM (Functions for free communication via the RS232 interface)

ALUX20\_OHP4\_SEAB (Functions for communication via IKOS)

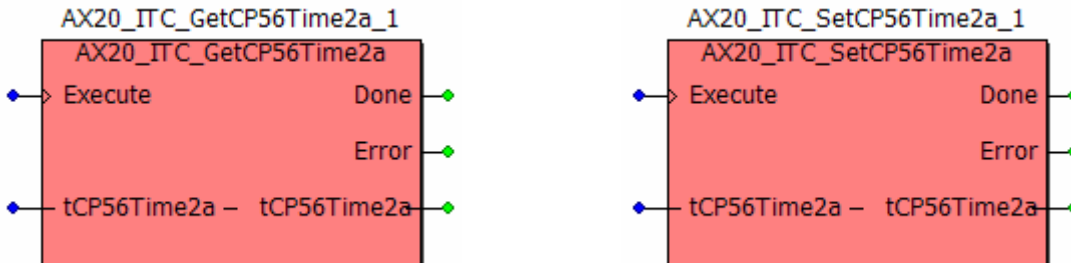
ALUX20\_OHP4\_SYS (Diverse general functions)

ALUX20\_OHP4\_SYS\_SA (Functions for fault analysis and for DEA coupling evaluation)

## 7.2 Library ALUX20\_OHP4\_SYS

### 7.2.1 Reading and Setting the ALU320 System Time in the User Program

Function blocks AX20\_ITC\_GetCP56Time2a and AX20\_ITC\_SetCP56Time2a



#### Input variable (VAR\_IN):

**Execute** : BOOL Enable flag for the function block, 1 = execute

#### Output variable (VAR\_OUT):

**Done** : BOOL 1 = Function executed

**Error** : BOOL 1 = Function faulty

#### Input/output variable (VAR\_IN\_OUT):

**tCP56Time2a** : 7 Byte System time (date/time)

The system time is displayed according to IEC 60870-5-4, CP56Time2a in 7 bytes as follows:

$2^7$		milliseconds		$2^0$	
$2^{15}$		0...59999		$2^8$	
IV	Res	$2^5$	minute 0...59	$2^0$	
S	Res	$2^4$	hours 0...23	$2^0$	
$2^7$	WT 1..7	$2^5$	$2^4$	day of the month 1...31	$2^0$
Res		$2^3$	month 1...12	$2^0$	
Res	$2^6$	year 0...99		$2^0$	

IV = invalid

IV <0> = valid

IV <1> = invalid

S = summer time

S<0> = normal time

S<1> = summer time

WT = day of the week

WT<0> = no entry

WT<1> = Monday

:

WT<7> = Sunday

In Multiprog the time can depicted for example be in an array or in a structure.

TYPE

```
ArrayTime      :      ARRAY [1..7] OF BYTE;
```

END\_TYPE

TYPE

IEC\_Time:

STRUCT

```
ms:            WORD;  
minute:       BYTE;  
hour:         BYTE;  
day:         BYTE;  
mon:         BYTE;  
year:        BYTE;
```

END\_STRUCT;

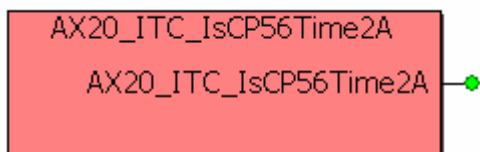
END\_TYPE

A variable of the type IEC\_Time or ArrayTime is transferred to the function block AX20\_ITC\_GetCP56Time2a to retrieve the current time or to set the time using the function block AX20\_ITC\_SetCP56Time2a -

The OHP project submittals already include the IEC\_Time structure in the data type declaration.

## 7.2.2 Status of the ALU320 Read System Time

Function AX20\_ITC\_IsCP56Time2A



**Beware:** Don't use any EN/ENO for the OHP functions.

**Output variable (VAR\_OUT):**

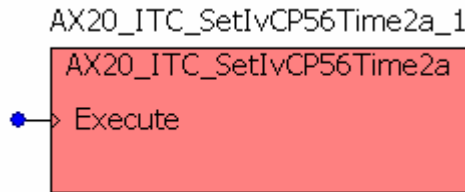
**AX20\_ITC\_IsCP56Time2A** : BOOL                      Status of the system time

0 = Time is invalid

1 = Time is valid

### 7.2.3 Status of the ALU320 Set the System Time to Invalid

Function block AX20\_ITC\_SetIvCP56Time2A



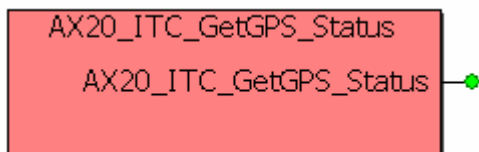
**Input variable (VAR\_IN):**

**Execute** : BOOL Enable flag for the function block, 1 = execute

If this function block is executed, the ALU system time will be set to invalid. This call can be used for example in a system task cold and/or warm start, to set the system time to invalid after a restart, until it is specifically set again (GPS, SNTP).

### 7.2.4 Status of the ALU320 System Time with GPS Synchronisation

Function AX20\_ITC\_GetCP56GPS\_Status



**Beware:** Don't use any EN/ENO for the OHP functions.

**Output variable (VAR\_OUT):**

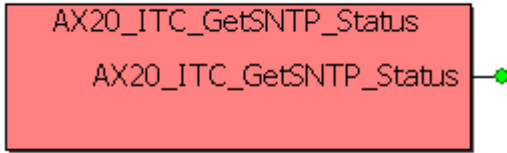
**AX20\_ITC\_GetCP56GPS\_Status** : DWORD Status of the system time with GPS synchronisation

Status in Hex:

- 16#00000001 GPS module ready
- 16#00000003 Time set
- 16#81000001 Due to a GPS warning „time invalid“ the time cannot be adopted
- 16#82000001 Error on setting the time
- 16#84000001 RMC data set error
- 16#88000001 No connection to the GPS module

### 7.2.5 Status of the ALU320 System Time with SNTP Synchronisation

Function AX20\_ITC\_GetCP56SNTP\_Status



**Beware:** Don't use any EN/ENO for the OHP functions.

**Output variable (VAR\_OUT):**

**AX20\_ITC\_GetCP56SNTP\_Status** : DWORD Status of the system time with SNTP synchronisation

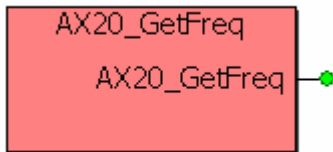
Status in Hex:

16#00000001 Connection to SNTP server enabled

16#00000003 Time set

### 7.2.6 System Clock

Function AX20\_ITC\_GetFreq



**Beware:** Don't use any EN/ENO for the OHP functions.

**Output variable (VAR\_OUT):**

**AX20\_ITC\_GetFreq** : WORD Blink pulse

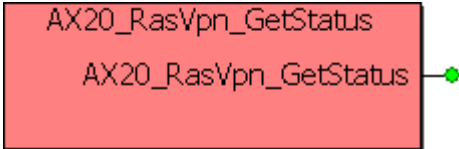


**Note:** There is a special block ILAX20\_GetFreq for the instruction list (IL) programming language. The functionality is the same, however the block has an additional execute input.

Bit	Frequency	Time Grid
X0	20 Hz	50 ms
X1	10 Hz	100 ms
X2	5 Hz	200 ms
X3	2,5 Hz	400 ms
X4	1,25 Hz	800 ms
X5		Not defined
X6		Not defined
X7		Not defined
X8	2 Hz	500 ms
X9	1 Hz	1 s
X10	0,5 Hz	2 s
X11	0,25 Hz	4 s
X12	0,125 Hz	8s
X13		Not defined
X14		Not defined
X15		Not defined

## 7.2.7 Status of the GPRS Connection

Function AX20\_RasVpn\_GetStatus



**Beware:** Don't use any EN/ENO for the OHP functions.

**Output variable (VAR\_OUT):**

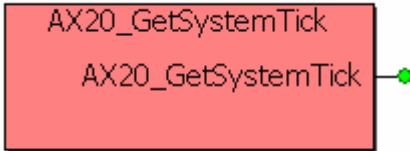
**AX20\_RasVpn\_GetStatus** : DWORD Status of the connection

This function delivers the status of the RAS and VPN connection during data transfer with the internal GPRS-modem.

Status in Hex	Meaning
16#00000001	Initialisation running
16#00000002	GPRS module ready
16#00000003	GPRS module ready – no connection
16#00000004	GPRS module ready - modem connection discontinued
16#00000005	GPRS module ready – VPN connection/login discontinued
16#00000006	GPRS module ready – VPN connection no login - PPP discontinued
16#00000008	Start modem connection
16#00000009	Modem connection ok
16#0000000A	Start VPN connection
16#0000000B	VPN tunnel exists
16#0000000C	Forced disconnection LOGOUT/LOGIN
16#0000000D	Forced disconnection after Traffic
16#0000000E	Hardware reset triggered
16#0000000F	Software reset triggered
16#00000010	Number of connection attempts to the provider reached
16#00000011	Number of connection attempts to VPN reached

### 7.2.8 System Tick

Function AX20\_GetSystemTick



**Beware:** Don't use any EN/ENO for the OHP functions.

**Output variable (VAR\_OUT):**

**AX20\_GetSystemTick** : DWORD      Time in milliseconds

This function delivers the current time in milliseconds since the ALU was started.

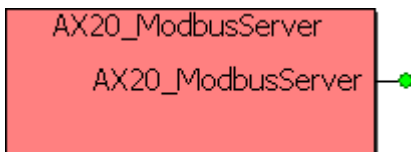
Forwards counter with transient point. Therefore the value will be set to zero after 49.71 days ( $2^{32}$  milliseconds).



**Note:** There is a special block ILAX20\_GetSystemTick for the instruction list (IL) programming language. The functionality is the same however the block has an additional execute input.

### 7.2.9 Modbus Server

Function AX20\_ModbusServer



**Beware:** Don't use any EN/ENO for the OHP functions.

**Output variable (VAR\_OUT):**

**AX20\_ModbusServer** : INT      Status

Status = 0 = Modbus driver not working

Status = 1 = Modbus driver working

When the ALU uses either Modbus-RS232-Slave or Modbus-TCP-Server transmission protocol, this block should be called at least once in the default task. The block answers the interrogation from the master/client. Hence it is ensured that the contents of the individual registers originate from the same program cycle. This is especially important when values are to be calculated that have a specific relationship to one another.

If this block is not used, it can happen that the contents of a telegram originate from different program cycles because the Modbus driver processes the data asynchronously with respect to the program cycle.



**Note:** There is a special block ILAX20\_ModbusServer for the instruction list (IL) programming language. The functionality is the same however the block has an additional execute input.

## 8 Tips and Tricks

### 8.1 Updating an Existing Multiprog Project for Online Change

Precondition for online change is Multiprog 4.8 and an ALU 320 firmware later than version 3.00. The current version can always be downloaded from the OHP website.

When you want to use the online change function on older projects you must check and if necessary modify the following settings.

#### Hardware Structure:

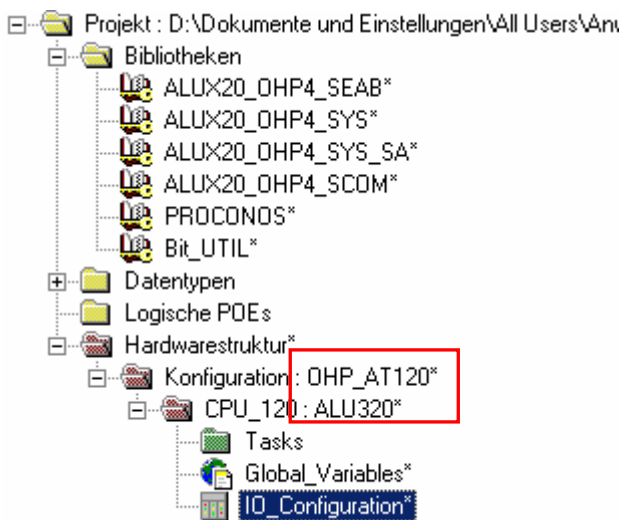
When the PLC type "IPC\_33" is configured, select the entry with the right mouse button and open the *Properties* dialogue. In the *PLC/Processor* register select "OHP\_AT120" as the *PLC Type*.

The CPU type should now be automatically set to "ALU320". If not you can change it again by reselecting and calling the properties dialogue using the right mouse button.

#### Libraries:

If you have used functions from the OHP firmware library in your project, you must also update them. How you proceed here is described in **Part II Configuration und Programming**, chapter 7.1.

Afterwards the project tree should look like this.



**Note:** Only the red framed entries are relevant and must be set exactly the same. The notation "Konfiguration" and "CPU\_120" can eventually be different in your project and must not be changed-. Firmware libraries must only be changed when you have used functions from them.

### 8.2 Set the Time and Diagnosis via Telnet

You can set the ALU system time or determine the status of the ALU system time using a Telnet connection.

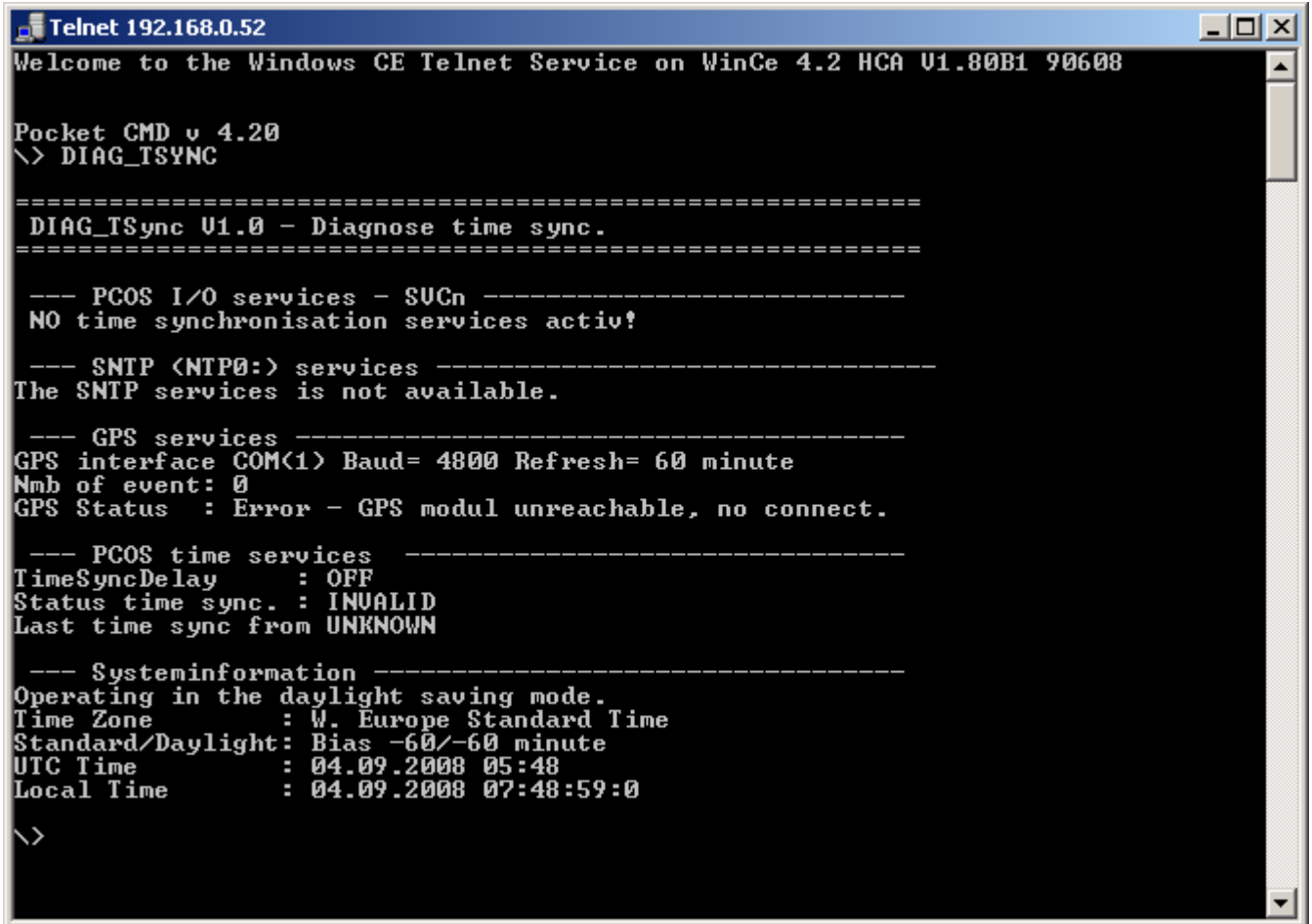
- To do this, connect the PC to the Ethernet interface on the underside of the ALU.
- Using the *Run* dialogue in the Windows start menu; enter *telnet 192.168.0.32* (if you have changed the ALU IP address, enter the modified address).

The dialogue window which subsequently appears will contain the WindowsCE version and the firmware version.

The following functions will now be available:

- Using *time* the current time will be displayed and can be changed.

- Using *date* the current date will be displayed and can be changed.
- *DIAG\_TSYNC* displays the status of the ALU system time



```
Telnet 192.168.0.52
Welcome to the Windows CE Telnet Service on WinCe 4.2 HCA U1.80B1 90608

Pocket CMD v 4.20
\> DIAG_TSYNC

=====
DIAG_TSync V1.0 - Diagnose time sync.
=====

--- PCOS I/O services - SVCn -----
NO time synchronisation services activ!

--- SNTP <NTP0:> services -----
The SNTP services is not available.

--- GPS services -----
GPS interface COM<1> Baud= 4800 Refresh= 60 minute
Nmb of event: 0
GPS Status : Error - GPS modul unreachable, no connect.

--- PCOS time services -----
TimeSyncDelay : OFF
Status time sync. : INVALID
Last time sync from UNKNOWN

--- Systeminformation -----
Operating in the daylight saving mode.
Time Zone : W. Europe Standard Time
Standard/Daylight: Bias -60/-60 minute
UTC Time : 04.09.2008 05:48
Local Time : 04.09.2008 07:48:59:0

\>
```

#### PCOS I/O services –SVCn

If the system time is set using a service routine (IEC 60870-5-101 or 104 time synchronisation), it will be displayed in this category.

#### SNTP <NTP0:> services

If the system time is set using a time server, it will be displayed here.

The function must be enabled in the ALU320.INI.

"SNTP service is not available" means that the service has not been enabled in the ALU320.INI.

#### GPS services

Displays the status of the synchronisation via the time receiver GPSRVC.

This function must be enabled in the ALU320.INI. See chapter 6.1.4.

"GPS service is not available", means that the service has not been enabled in the ALU320.INI.

### 8.3 ALU Status Information via Telnet

Using a Telnet connection you can call diverse diagnostic functions.

- To do this, connect the PC to the Ethernet interface on the underside of the ALU or use an existing GPRS connection.
- In the *Run* dialogue of the Windows start menu enter *telnet 192.168.0.32* (if you have changed the ALU IP address, enter the modified address).

The dialogue window which subsequently appears will contain the WindowsCE version and the firmware version.



**Note:** The following entries are only examples and can differ from the actual entries depending on the ALU configuration.

#### services list

Using this call you can determine which drivers and routines are running on the ALU.

Example:

HTP0:	0x00030110	HTTPD.DLL	Running
TEL0:	0x00031e90	TELNETD.Dll	Running
FTP0:	0x00036190	FTPD.Dll	Running
AUC0:	0x00036350	Storage Card\A320_SYS\AX20UDPCFG.dll	Running
S4U0:	0x0003c620	\Storage Card\PCOS_SYS\SVC_SEK_I104SUS.dll	Running
AXR1:	0x00036d90	Storage Card\A320_SYS\AXRASVPN.dll	Running
R4U1:	0x0003c7b0	Storage Card\PCOS_SYS\SVC104SA.dll	Running

#### diag\_cpu

Displays the operating system, hardware and driver versions.

```
=====
OS Version           : 4.20 Build 0 Pb-Id 3
ALU Version          : 3.23 Build 0.1 09.11.09
PLC SYSTEM           : A120
PLC ALU              : ALU 320
ProConOS Version     : 4.40
Pcos Version         : 3.16
Primary class name   : CPU
Name of the board    : XBD1
Manufacturer name    : OHP GmbH / JUMP
Date                 : 30.1.2009
Serial number        : UAD6K0200
Hardware revsion     : 0.1
Firmware revsion    : 1.5
```

---

Jida interface rev.: 1.2  
Jida boot count : 192

-----

### diag\_rasvpn /svc

Displays the RASVPN statistics.

```
RASVPN Status: (11) RASVPN service - VPN ready
RASVPN Nmb RasFct() Error: 1
Modem Nmb Reset : 7
PPP -ISP Nmb Dial: 11 (Dial-up attempts)
PPP -ISP Nmb Connect: 10 (Successful connections)
PPTP-VPN Nmb Login: 41 (VPN establishment attempts)
PPTP-VPN Nmb Connect: 30 (Successful VPN connections)
RASVPN Nmb RasFct() Error: 1 (Last occurred error code)
Modem Nmb Reset : 7 (Number of Modem resets)
```

It displays the number of ALU connections etc. that have occurred since the last boot of the ALU.

### diag\_rasvpn /stats

Displays the information regarding the RAS connection.

```
RAS - 2 Active connections
01.) 'RASPPPTP' Status='Connected'
DevName='RAS VPN Line 0' DevType='vpn'
Bytes S=10903, R=11056
Frames S=199, R=258
Compression I=100% O=107%
Duration 221:46.766 [mm:ss.mss]
Link (start): 28800 bps
02.) 'RASMODEM' Status='Connected'
DevName='Hayes Compatible on COM2:' DevType='modem'
Bytes S=62180, R=59404
Frames S=1139, R=1144
Compression I=0% O=0%
Duration 223:52.944 [mm:ss.mss]
Link (start): 28800 bps
```

---

Furthermore, Windows standard functions such as ipconfig are available.

Example of an existing GPRS connection:

**ipconfig**

Windows IP configuration

Ethernet adapter [AX887961]:

IP Address ..... : 0.0.0.0  
Subnet Mask ..... : 0.0.0.0

Ethernet adapter [SERIAL ON COM1]:

IP Address ..... : 10.209.197.137  
Subnet Mask ..... : 255.0.0.0  
Default Gateway ... : 10.209.197.137

Ethernet adapter [RAS VPN LINE 0]:

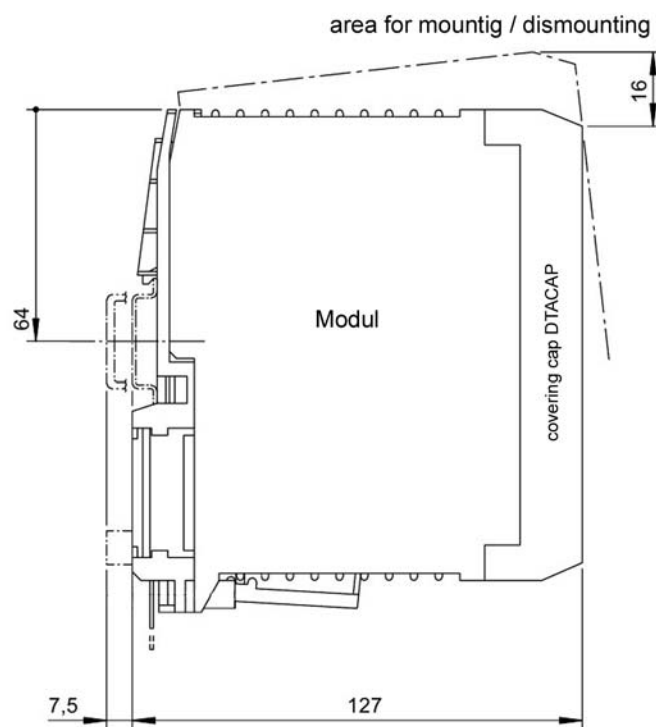
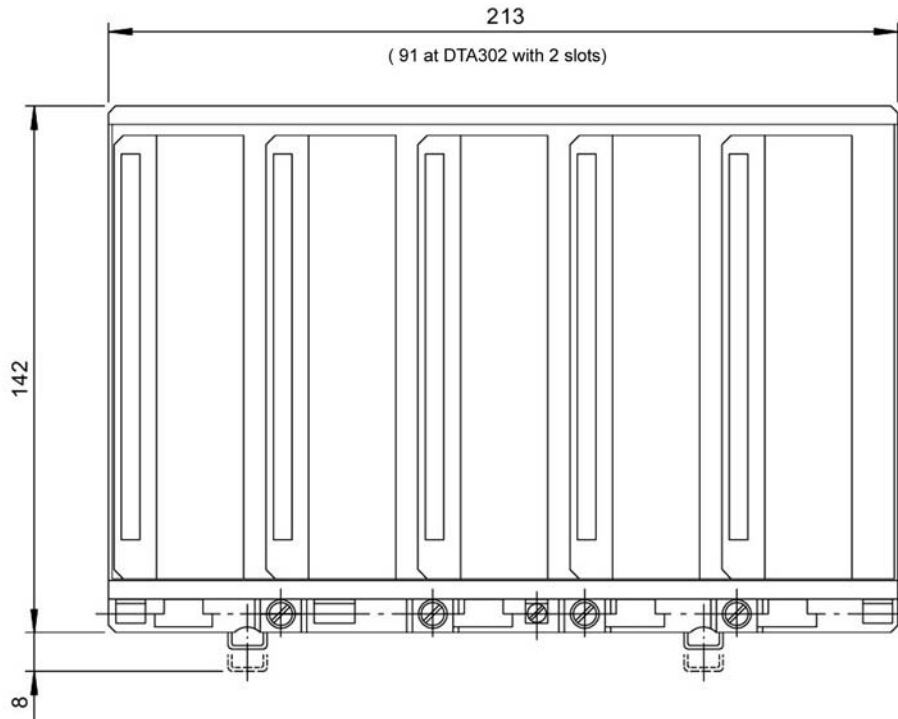
IP Address ..... : 13.137.0.116  
Subnet Mask ..... : 255.255.255.255  
Default Gateway ... : 13.137.0.116

DNS Servers..... : 10.74.83.22  
                          193.254.160.1

## Part III Module Description

## 1 Mounting Dimensions

Dimensions DTA300 and DTA301 (specifications in mm)





### 2.1.1 Interface Allocation of the Screw Terminals:

Terminal	Function	Description
1	UB	24V Operating voltage
2	UB	24V Operating voltage
3	Reserved	
4	CAN_HZ	CAN High supply *)
5	CAN_LZ	CAN Low supply *)
6	CAN_GND	CAN Ground *)
7	CAN_GND	CAN Ground *)
8	Reserved	
9	Reserved	
10	GND	Ground for operating voltage
11	GND	Ground for operating voltage

\*) in preparation

#### 2.1.1.1 Power Supply

The ALU 320 has an internal power supply with an input voltage range of 14 – 30 VDC. Over this the ALU 320 and external periphery modules are supplied. For the periphery modules connected to the PAB1 3500 mA are available.

In case of under voltage < 14 V the CPU signal memory and the telegram buffer will be written to the CF card, non volatile.

It is recommended to feed the ALU320 with a potentially isolated power supply with at least 5A-

#### 2.1.1.2 Interface for an Optional GPS Time-of-Day Receiver

The ALU 320 has a connection possibility for an optional GPS time-of-day receiver GPSRCV from OHP.

This is connected via a serial interface to the ALU 320.

#### 2.1.1.3 CAN Interface (in Preparation)

The ALU 320 includes a CAN master/slave interface. The interface module is based on a Philips chip SAJ1000- The interface has galvanic isolation using opto-couplers. The bus address and the termination are set via an 8-pole DIL switch located on the left side plate of the ALU 320.

(See ALU 320 Display Elements)

### 2.1.2 Serial RS232 Interface:

Pin-No.	Abbr.	Signal Description	DIN Abbr.	DIN Signal Description
1	DCD	Data Carrier Detect	M5	Empfangssignalpegel
2	RXD	Receive Data	D2	Empfangsdaten
3	TXD	Transmit Data	D1	Sendedaten
4	DTR	Data Terminal Ready	S1.2	DEE betriebsbereit
5	GND	Ground	E2	Betriebserde
6	+5VDC	Supply for GPS receiver		
7	RTS	Request to Send	S2	Sendeteil anschalten
8	CTS	Clear to Send	M2	Sendebereitschaft

### 2.1.3 RS485 Interface on the ALU320-485:

Pin-No.	Abbr.	Signal Description	DIN Abbr.
12	T+	Transmit Data (+)	TXD +
13	T-	Transmit Data (-)	TXD -
14	R-	Receive Data (-)	RXD -
15	R+	Receive Data (+)	RXD +
22		Ground	GND

### 2.1.4 Compact Flash Memory Cards

The ALU 320 has a memory card slot for a CF card. The card is inserted beneath the LEDs. CF cards with a maximum memory capacity of 8 GB can be used.

The CF card contains all relevant data for the ALU 320, such as operating and runtime system, user application and telegram buffer. The ALU 320 is delivered complete with an operational CF card.

**Caution:** Insert and remove the CF card only when the voltage is switched off.

### 2.1.5 ALU 320 Display Elements

The ALU 320 has 12 LEDs, which display the operating status of the device.

No.	Function	Colour	Display
1	UB OK	Green	Continuously on
2	AWP run, stop, fault	Green	Continuously on, off, blinks
3	CAN OK, error, NC	Green	
4	Reserved	Red	
5	TXD2	Green	Blinks
6	RXD2	Yellow	Blinks
7	TXD1	Green	Blinks
8	RXD1	Yellow	Blinks
9	Ethernet Link	Green	Blinks
10	Ethernet Active	Yellow	Continuously on
11	Reserved	Green	
12	Reserved	Rot	

During booting after power-on, the LEDs are as follows:

- LED 1...4 continuously on
- LED 7 blinks twice
- With connected Ethernet after ca. 10 seconds LED 9 blinks, LED 10 continuously on.
- After 30 seconds LED 3 and 4 go off. If the user program is stored on the CF card as a boot project, then this will be started. If no program is stored, then LED 2 will also go off.

Meaning of LED 3 when using an ALU with integrated modem for dial-up operation.

LED	Meaning
Off	No connection
Continuously on	Data connection, Password OK, IEC communication
Blinks slowly	Connection establishment initiated by the ALU
Blinks fast	Connection establishment initiated by the remote station (RING)
Blinks twice (500ms) - Pause	Modem initialisation problem
Blinks 3 times (300ms) - Pause	Modem no longer ready

**Remark:** This function is available commencing with the ALU firmware version 1.91.

## 2.1.6 ALU320 Operator Elements

### Reset Button

The CPU pcb has a reset button on the front side. Independent of the CPU this button has the following functions:

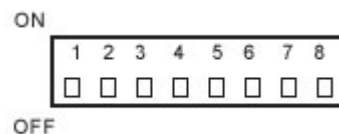
- press the button > 5 s: hardware reset
- press the button < 5 s: triggers a free interrupt

### DIL Switch

The 8 pole DIL switch is used to set the start behaviour as well as the CAN bus address and the CAN bus termination (CAN Bus in preparation).

The switch is situated on the solder-side of the pcb and can be accessed through a cut-out in the left side of the ALU 320 housing.

Switch	Function
1	CAN address (in preparation)
2	CAN address (in preparation)
3	CAN address (in preparation)
4	CAN address (in preparation)
5	Reserved
6	<b>ON = Warm start, OFF = Cold start</b>
7	ON = CAN Bus terminated
8	ON = CAN Bus terminated



## 2.2 Technical Data ALU 320

<b>Supply</b>	
Supply voltage	14V – 30V
Power loss ALU 320	< 5 W, without expansion modules
Available current for I/O modules	Max. 3500mA
<b>Serial Interfaces</b>	
Number	2
Type, speed	serial, asynchronous, V.24 level, max. 19.200 Baud
Connection	RJ45, connection front side
<b>Ethernet</b>	
Number	1
Type	IEEE802.3, 100 Base T, 100 MBit/s
Connector	RJ45, Connection underneath
<b>USB</b>	
Number	1 x USB 1.1
Type	1 Standard USB front side
<b>Slot for CF Card</b>	
Number	1
Type	Standard Compact Flash (CF Card), max. 8 Gbyte
<b>CAN Connection</b>	
Number, Type	1 CAN Master,
Connection	Connection via screw terminals, front side.
<b>CPU /Memory</b>	
Processor type	Scaleable CPU Board
Memory	32 MByte RAM / 32 MByte Flash
Telegram buffer	1 MB in Flash, no battery necessary
Operating system	Windows CE 4.2 real-time kernel
<b>Time-of-Day Handling</b>	
Type	Buffered ms exact RTC
Synchronisation	Via an optional GPS Module connected to a serial interface
<b>CPU Expansion Slot</b>	
Number, type	1 slot for modem expansion card
<b>Protection Class</b>	
Protection class (IEC 60529)	IP20
Protection class (IEC61140)	III
Isolation	EN60950, IEC 950
<b>Environmental Conditions</b>	
Ambient temperature in operation	0°C to 60°C
Storage temperature	-25°C to 85°C
Relative humidity (IEC 68-2-1-1/2)	Up to 95% no condensation
<b>Displays</b>	
9 LED displays	1 LED UB OK 2 LED run, stop, fault 3 LED internal COM interface 4 LED res. 5 LED TXD1 6 LED RXD1 7 LED TXD2 8 LED RXD2 9 LED Ethernet link 10 LED Ethernet active
<b>Mechanical Construction</b>	
Format	Standard A120/Compact I/O module 3 HE, 8TE