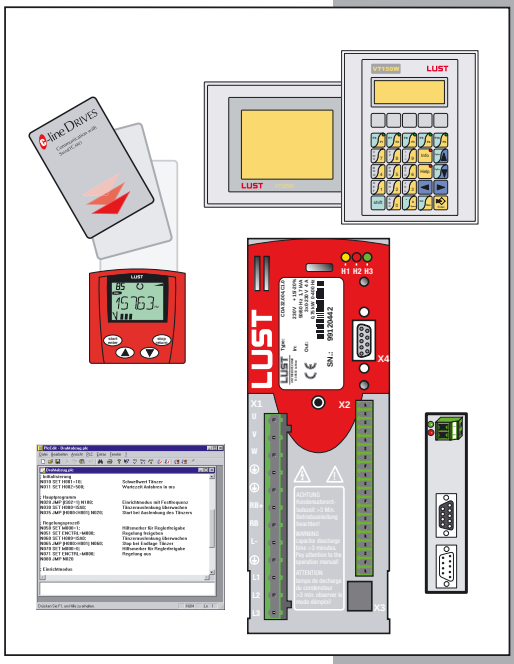
EN

CDA3000-PLC

System manual



Sub-automation unit with c-line DRIVES

LUST

LUST

System manual CDA3000-PLC

Id.-No.: 0840.12B.1

Stand: 05/.2003

Valid from CDA3000-software version V700.10

LUST

How to use this guide

Pictographs

➢ **Attention!** Faulty operation can cause damage or malfunction of the drive system.

➢ Here you will find further comparable applications.

➢ The symbol stands for **hurdles and barriers**, which have to be cleared during realization of the concept.

➢ **Note:** What you have to know, information in short form.

1 Introduction

Due to the increased pressure to reduce costs a new slogan is born in the automation industry. Sub-automation unit - a catchword, which will be used more and more. What is the difference between this solution and others, what are the advantages?

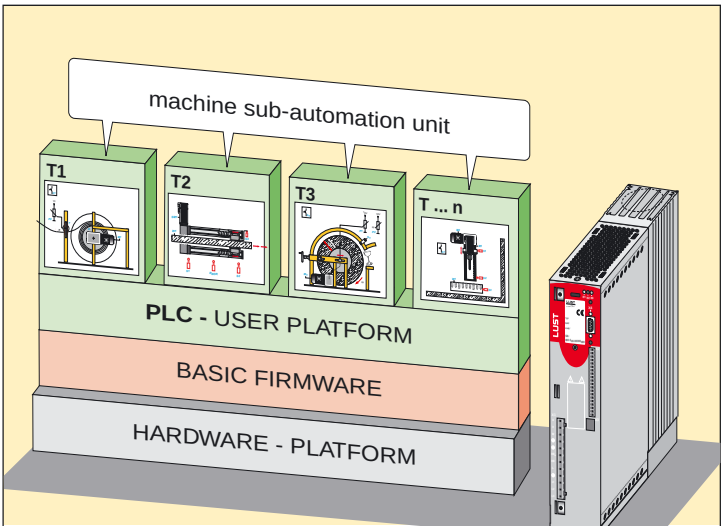
This manual clears these and other questions. It is made for all users of inverters and servocontrollers as well as for people technically interested, looking for new solutions. Salespeople, specialists and executives will get an impression of new economic solutions.

The manual includes the following five ranges:

- Chapter 1: General introduction
- Chapter 2: System survey
- Chapter 3: Sub-automation units
- Chapter 4: Commissioning of PLC-systems
- Chapter 4: Order data of system components

1.1 Thinking in machine sub-automation units

Thinking in machine sub-automation units requires in the drive units a free-programmable core inside, named PLC. Because the contribution to the solution of sub-automation units depends more and more on the flexibility of the PLC-user plattform within the drive system. Finally, only the software offers the „intelliegence“ and so the method to get over the motion solution.

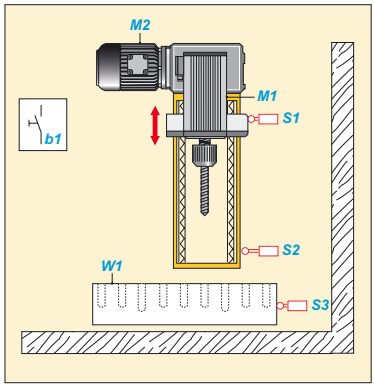


picture 1.1 PLC-user plattform

Seeing driving problems from a process oriented standpoint, always the complete system has to be taken into consideration. Large complex machines and instruments have to be disassembled in small clear sub-automation units. In machine sub-automation units, doing a job automatically and inevitably following to a set procedure. Normally such sub-automation units reflect the mechanical and electrical construction unit.

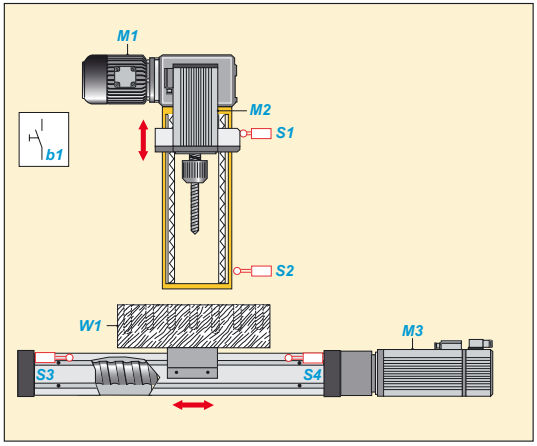
Please note at realization of new solutions that the procedure of sub-automation units is enclosed. That means that a changing of conditions cannot be made directly via an other sub-automation unit, it can only adjusted via a „clear defined interface“.

Special at this solution is that the sub-automation unit can be used in other machine types, requiring the same sub-automation unit, without changing the program.



picture 1.2 sub-automation unit - drill unit

Coupling of sub-automation units to one „function unit“ allows the co-ordination of complex procedures. In the function unit one of the sub-automation units manages the coordination of the whole function unit as well. However, the function of other sub-automation units will be required via the already mentioned „clear defined interface“.



picture 1.3 function unit - drilling appliance

The complete automatic process is clearly structured resp. is divided in closed machine sub-automation units. Sub-automation units, allowing unitizing of machine types and having a high reusability. Serial commissioning can be made without special knowledge or PC, simply via SMART-CARD - plug-in, load and run.

1.2 Typical machine sub-automation units

The already cleared machine sub-automation units can typically divided in three groups:

I/O-oriented sub-automation unit

Motion solutions, processes are mainly defined by means of I/O-signals. Typical applications are: advance unit for drilling or sinking, belt and carriage drives, drives for doors and gates as well as pump stations with float switches.

Time-controlled sub-automation unit

Motion solutions, processes are mainly time-controlled. Typical applications are: melting and mixing plants for paints and other materials, different centrifuges as well as mills and shredder.

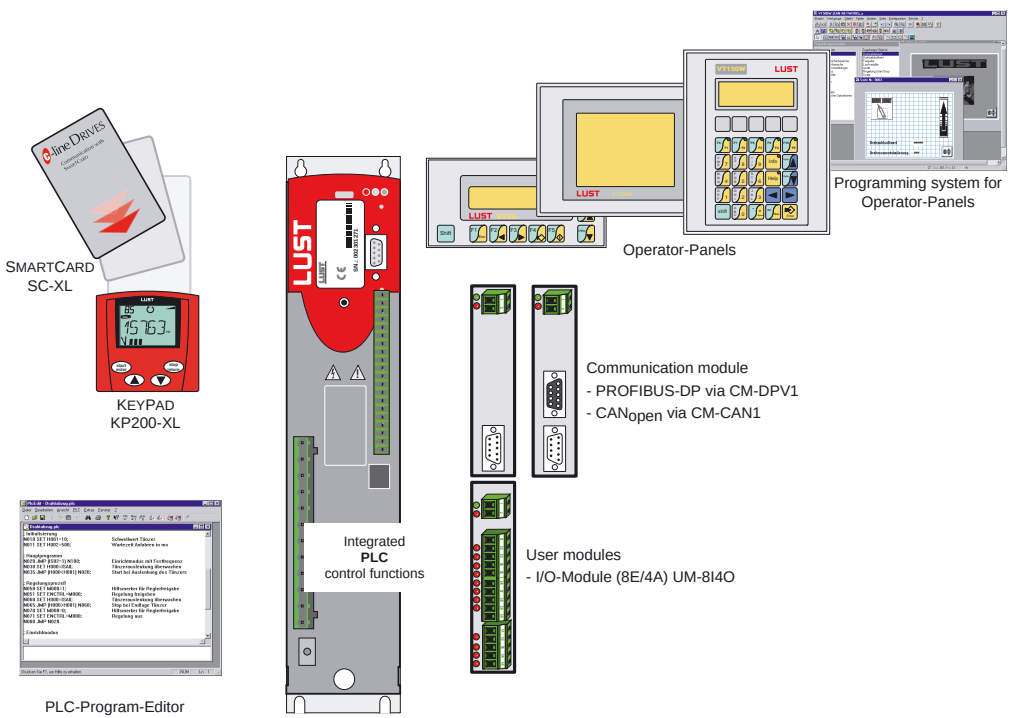
Controlled sub-automation unit

Motion solutions, for controlling process variables like torque, traction, pressure, temperature or position. Typical applications are wobbler and dancer control for winders, block-protection for shredder, simple positioning jobs for drives, rotating, as well as drives for doors and gates and the classic pressure, temperature and flow control.

2 System survey

The system consists of the following components:

- Drive CDA3000 with accessories and option modules
- Software package CDA3000-PLC
- Different operator panels with accessories



picture 2.1CDA3000-PLC System survey

2.1 Software package CDA3000, PLC

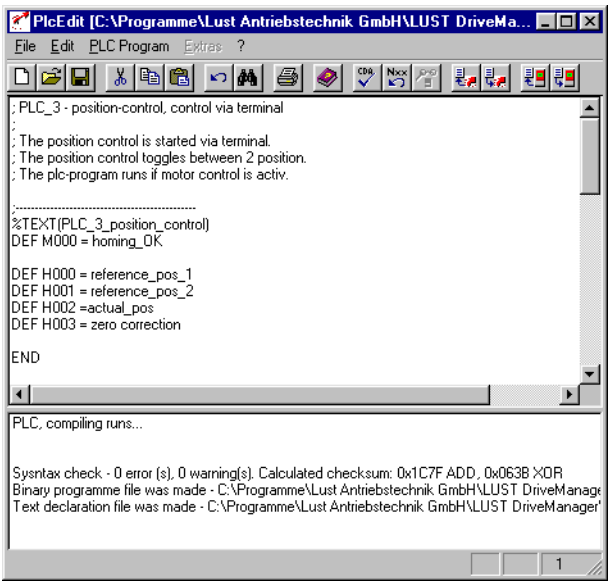
The software package CDA3000-PLC consists of three parts:

- PLC-program editor on CD-ROM. The PLC-editor is additionally to the DRIVEMANAGER.
  On CD-ROM you will find this system manual as well as PLC and device standard software. Furthermore it incl. additional product documentation (user manual CDA3000, communication modules)
- PLC-device firmware V700.xx for CDA3000
- DRIVEMANAGER V3.20 (or higher version) on CD-ROM.

2.1.1 PLC-Editor

The PLC-program editor will be delivered on a separate CD-ROM as installation version. In German and English available.

The PLC-editor is part of the DRIVEMANAGER and so it can only be used with DRIVEMANAGER. This presumes that a DRIVEMANAGER with version V3.20 and higher is installed on the PC.



picture 2.2 PLC-editor

The PLC-editor is only necessary for the engineering resp. first commissioning, a serial commissioning of the drive controller will be made via DRIVEMANAGER-data set or SMARTCARD.

The PLC-program editor offers the following functions:

- Program production
  - Editor for program production
  - Generating a text declaration file <project name>.txt for variables to show application specific texts in the DRIVEMANAGER.
  - Syntax check of command code
  - Renumbering of line numbers
- Program-Handling
  - Load/Save/Print/New production of programs
  - Load/Save a program from/in the connected drive
    Load/Save a program from/in the DRIVEMANAGER data set
- Online-Help for PLC-Editor and for command syntax with examples

All PLC-functions are selectable via function switches (buttons)

New program
Open program as file
Save program as file *.plc
Cut text
Copy text
Paste text
Undo
Search/Replace
Print program
Online help
Program syntax check
Renumbering of line numbers
Load program from data set
Save program in data set
Load program from device
Save program in device

PLC-Program process

A program is divided in two parts:

1. Text declaration for used variables, flags, counter and timer
2. PLC-program

The **Text declaration** is for marking of used variables, counters and timers in the PLC-program with the application specific function. A text file will be generated out of the text declaration, evaluated in the DRIVEMANAGER, shows the parameters with application specific texts.

The text declaration starts with the designator, obtains the the project name of the text declaration file (for details see „PLC-program data“).

```
%TEXT (project name)     ; Start text declaration
```

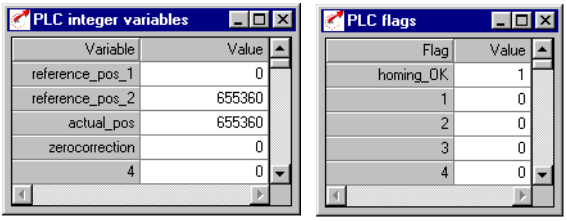
An assignation of the parameter texts follows:

```
DEF M000 = setpoint_OK
DEF H000 = reference_pos_1
DEF H001 = reference_pos_2
DEF H002 = actual_pos
DEF H003 = Zero correction
```

End of the text declaration always with:

```
END
```

The text declaration is optional. Non-declared PLC-parameters will not be saved in the text file resp. displayed with its number in the DRIVEMANAGER.



picture 2.3 Indication of PLC-parameters with application specific texts

The **PLC-program** attach to the text declaration. It contains a program head, the true program part and the program end.

The program head consists of a line, includes the program number (here only %P00 possible):

```
%P00
```

The lines of the program part are called command lines. The number of the sets, which can be saved in the CDA3000, is limited to 254 (N001 ... N254). Each command line consists of line number, command and operand. Via semicolon a command can be inserted.

```
N030 SET M000 = 0;        Setpoint not defined
```

Program ends always with line (without line number):

```
END
```

For program examples see chapter 3 or already installed DRIVEMANAGER-directory„..\userdata\PLC“.

Program check and processing

For **quickstart** resp. **new producing** of a PLC-program call for syntax check with blank text field. Now the PLC-editor offers the producing of a program body.

The **syntax check** checks if there are any errors in the command codes of the actual program. It will be executed automatically at saving of the program in the drive or manually via the appropriate button. The result of the check will be indicated in the status rail. In case of any error message, it is possible via double-click on the appropriate message to jump directly to the faulty program line.

The **renumbering** of line number ease inserting a program set. At renumbering the first line gets the number N010, all other will be incremented by steps of 10 (N020, N030, ...). If it is not possible to show a program in the stated line ranges (001-254), the steps will be reduced automatically.

PLC-Program files

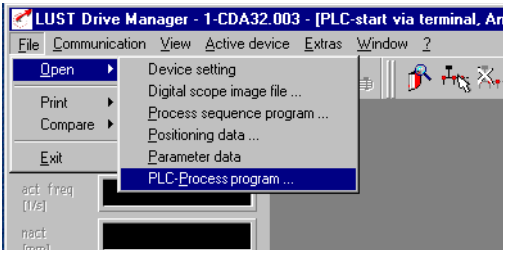
The **program content** will be stored in two files:

1. **Program file** *.plc
   This file consists the PLC- program as well as text declaration and so the complete program information. In case of transfering the PLC-program it is sufficient to copy the file.
2. **Text declaration file** <Project name>.txt
   The file will be used from the DRIVEMANAGER to indicate the application specific parameter names.
   It will automatically generated upon succesful program download in the drive controller or in a data set of the text declaration of the program file. The file <Project name>.txt will be copied in the DRIVEMANAGER file „LUST DriveManager\firmdata\<Project name>.txt“. This file is only available on the PC, generated this program resp. loaded the source code in the drive controller. However it can be copied to other PCs.

To reproduce all program information resp. data each program has to be saved as file *.plc.
The comment lines of the process program and the text declaration are not saved in the drive or in device data sets, that means a read-back is not possible.

There are 3 ways to open an available PLC-program:

- Double-click on file *.plc. Then DRIVEMANAGER opens and it starts the PLC-editor and opens the program.
- Opening via DRIVEMANAGER menu „File/Open/PLC-process program ...“



picture 2.4 Open PLC-program via DRIVEMANAGER

- Opening via already started PLC-editor

PLC-command syntax

.

Command	Operand		Note
Jump command			
JMP		Ny/END	Absolute jump
	(Ippi = 0/1)	Ny/END	Condition of input
	(Oppi = 0/1)	Ny/END	Condition of output
	(Mxxx = 0/1, = != Myyy)	Ny/END	Condition of flag
	(Mxxx & | ^ Ippi)	Ny/END	Logical link flag-input
	(Mxxx & | ^ Ippi)	Ny/END	Logical link flag-output
	(Cxx = != 0 .. 255)	Ny/END	Counter status
	(Zxx = != 0)	Ny/END	Timer status
	(Hxxx = != 0, = != < <= > >= Hyyy)	Ny/END	Size of integer variables
	(Fxxx = != 0.0, = != < <= > >= Fyyy)	Ny/END	Size of floating point variables
Set commands			
SET	Oppi = 0/1, Mxxx		Set output diretly or with flag
	Mxxx = 0/1, Ippi, Oppi, Myyy, M[Cxx]		Set flag
	Mxxx = Hyyy		Set flag (LSB of Hyyy)
	M[Cxx] = Mxxx		Set flag (indicated)
	Mxxx & | ^ Myyy		Logical link of flag
	Cxx = d, Cyy, Hyyy		Set counter
	Cxx + - d, Hyyy		Calculate counter
	Zxx = d, Hyyy		Set timer
	Hxxx = z, Hyyy, H[Cyy], Fyyy, Cyy, Zyy, PARA[n, i], PARA[n], PARA[Hxxx,Hyyy], PARA[Hxxx]		Set variable At processes with floating point variables the value range will be limited automatically. No error message follows.
	H[Cxx] = Hyyy		Set variable (indicated)
	Hxxx & | ^ Hyyy		Logical link of variables
	Hxxx << >> b, Hyyy		Move variable
	Hxxx + - * : % z, Hyyy		Calculate variable
	Hxxx = ABS Hyyy		Built variable amount
	Fxxx = f, Hyyy, F[Cyy], Fyyy, PARA[n, i], PARA[n], PARA[Hxxx,Hyyy], PARA[Hxxx]		Set floating point variable. At processes with floating point variables the value range will be limited automatically. No error message follows.
	F[Cxx] = Fyyy		Set floating point variable (indicated)
	Fxxx + - * : f, Fyyy		Calculate floating point variable
	Fxxx = ROUND Fyyy		Round floating point variable
	Fxxx = ABS Fyyy		Built floating point amount

Command	Operand	Note
SET	PARA[n, i], PARA[n], PARA[Hxxx,Hyyy], PARA[Hxxx] = Hyyy, Fyyy	Set parameter
	BRKPT = 0	Deactivate brakepoint logic
	BRKPT = 1	Activate brakepoint logic
	Hxxx = OUTPUT, INPUT	Read variables with image of output resp. input
	Hxxx, Fxxx = ACTFRQ	Read actual frequency [Hz]
	Hxxx, Fxxx = ACTSPEED	Read actual speed [rpm]
	Hxxx, Fxxx = ACTPOS	Read actual position [Incr]
	Hxxx, Fxxx = REFPOS	Read reference position [Incr]
	Hxxx, Fxxx = ACTTORQUE	Read actual torque [Nm]
	Hxxx, Fxxx = ACTCURRENT	Read actual current (effective) [A]
	Hxxx = OSA0	Read analog output value 0 - 1023 = 0V - 10V
	Hxxx = ISA0	Read analog input ISA0 -512 - 511 = -10V - +10V
	Hxxx = ISA1	Read analog input ISA1 0 - 1023 = 0V - +10V
	OUTPUT = Hyyy	Set output image Only outputs with set function selector will be set FOxxx=PLC.
	OSA0 = Hyyy	Set analog output 0 - 1023 = 0V - 10V Condition: function selector 200-FOSA0 = PLC
	REFFRQ = Hyyy, Fyyy	Write frequency reference to FOPT1 Reference is only active if reference selector is set to 280-RSSL1=FOPT1 or 281-RSSL2=FOPT1.
	INPOSWND = Hxxx	Set reference reached window [incr] (Default 100 incr.)
	INPOSTIME = Hxxx	Set reference reached time [ms] (Default 20 ms)
	Mxxx = STA_ERR	Read error status (1 -> error)
	Mxxx = STA_WRN	Read warning-status (1 -> warning)
	Mxxx = STA_ERR_WRN	Read error/warning status (1 -> error/warning)
	Mxxx = STA_ROT_R	Motor turns right (1)
	Mxxx = STA_ROT_L	Motor turns left (1)
	Mxxx = STA_ROT_0	Motor standstill (1)

Command	Operand	Note
SET	Mxxx = STA_ACTIV	Motor/Control active (1)
	Mxxx = STA_LIMIT	Limitation (1)
	Mxxx = STA_REF	Reference reached (1)
	Mxxx = STA_BRAKE	Condition brake (1 -> Brake active)
	Mxxx = STA_OFF	Condition de-energized (1)
	Mxxx = STA_C_RDY	Condition control ready (1)
	Mxxx = STA_WUV	Warning undervoltage (1)
	Mxxx = STA_WOV	Warning overvoltage (1)
	Mxxx = STA_ WIIT	Warning I^2*t (1)
	Mxxx = STA_WOTM	Warning overtemperature motor (1)
	Mxxx = STA_WOTI	Warning heat sink temperature (1)
	Mxxx = STA_WOTD	Warning interior temperature (1)
	Mxxx = STA_WIS	Warning apparent current - limit (1)
	Mxxx = STA_WFOUT	Warning output frequency - limit (1)
	Mxxx = STA_WFDIG	Warning reference of master faulty (1)
	Mxxx = STA_ WIT	Warning I*t (1)
	Mxxx = STA_ WTQ	Warning torque (only FOR and SFC) (1)
	ENCTRL = 0/1; Myyy	Stop/start control (Mxxx = 0/1) Execution of function only if control location CLSEL=PLC is set.
	INV = 0/1; Myyy	Invert reference
	PCTRL = 0/1, Myyy	Activate position controller
	BRAKE = 0/1, Myyy	Brake motor to 0 Hz (1, Mxxx = 1)
	ERR = 0/1, Myyy	Release error switch-off (1, Mxxx = 1)
Sub program call		
CALL	Ny	Sub program call to line Ny Max. nesting depth: 255
RET		Reset to line of sub program call
Wait commands		
WAIT	d, Hxxx	Waiting time in ms (0 ... 4.294.967.295 ms)
WAIT	PAR	Wait until parameter is written.
Go commands (only for positioning control)		
GO	W A Hxxx	Go **absolutely** by value Hxxx and wait for program processing until target position is reached.

Command	Operand	Note
GO	W R Hxxx	Go **relative** by value Hxxx and wait for program processing until target position is reached.
	A Hxxx	Go **absolutely** by value of Hxxx (program processing continuous)
	A Hxxx	Go **relatively** by value of Hxxx (program processing continuous)
	0	Referencing to 0 (reference position =actual positon=0)
	0+Hxxx	Referencing to Hxxx (reference position=actual position=0)
Further commands		
NOP		Instruction without function
INV	Oppi, Mxxx, Hxxx	Inverting
END		Finish program, all following lines will be ignored. Do not enter a line number.
BRKPT		SET BRKPT=1: Sets a breakpoint in the program line SET BRKPT=0: no function

Syntax of command index:

Cxx, Cyy	Counter index 00-10
Hxxx, Hyyy	Integer-Variable index 000-127
Fxxx, Fyyy	Floating-Point-Variable indes 000-127
Zxx, Zyy	Timer index 00-10
Ny	line number 001-254
PARA[n, i]	Parameter number n 000-999 Parameter index i 000-255
Mxxx, Myyy	Flag index 000-255
Ippi, Oppi	Inputs ppi = A00, A01 S00-S03, E00-E07
Oppi	Outputs ppi = S00-S02, E00-E03

b	Value 1-32
d	Status of a counter 0 ... 255 (8 bit)
d	Status of a timer 0 ... 4.294.967.295 (32 bit)
f	Floating point value (32 bit)
z	Integer value ±2147483648 (32 bit)

2.1.2 PLC-Firmware

PLC-functionality

Firmware CDA3000-PLC with software version number V700.xx is suitable for the frequency inverter CDA3000. It contains a routine for sequential processing of a user-programmed procedure.

Number of programs in drive memory:	1
Number of commandlines per program:	254
Process time per command line:	1 ms

The process program allows:

- Start of motor control
- Reference pre-set for motor control
- Set/Read analog and digital inputs/oputputs
- Read/Write parameters
- Mathematical operations (+,-,*, :, modulo, abs, round)
- Timer and counter functions
- Simple positioning control with frequency acceleration ramps over the reference structure.

Basic software

Basis of the PLC-software is the standard software V3.50-01 of CDA3000. The defined functions of V3.50-01 (s. CDA3000 application manual 07/2002) are applicable in the PLC-execution up to the following exceptions resp. changes:

- No 4 USER-data sets
  The device firmware includes one data set only. So a switching between them is not possible. The known pre-set solutions with USER data set switch-over are not applicable. Characteristic data set switching is available.
- Integrated CAN_{open}-protocol
  The system bus will be extended by the CAN_{open}-protocol. Operation with CAN_{LUST}-protocoll is also possible.
  So the CAN_{open}-link via communication module CM-CAN1 can be realised. The requested CAN-protocol can be set via parameter 654-PRSEL (CLUST/COPEN).
  Communication module CM-CAN2 (CAN_{open} with setting the address via coding switch) and CM-DPV1 (Profibus DP-V1) can be used alternatively.
- Motor identification not applicable
  The automatic identification of asynchronous motors are not part of the PLC-firmware. Motor identification can be executed with the standard firmware V3.50-01. The identified data set can be used in the PLC-firmware.

PLC-Parameters

All PLC-control commands are mapped via parameters. They are subject to the already known data set handling of CDA3000. The parameters can be processed via DRIVEMANAGER in the PLC-function window (s. picture 2.5).

New ist, that the whole PLC-program is saved as machine code in two parameters. These parameters are included in the device data set and can be load resp. saved during the data set handling via DRIVEMANAGER or at serial commissioning via SMARTCARD SC-XL.

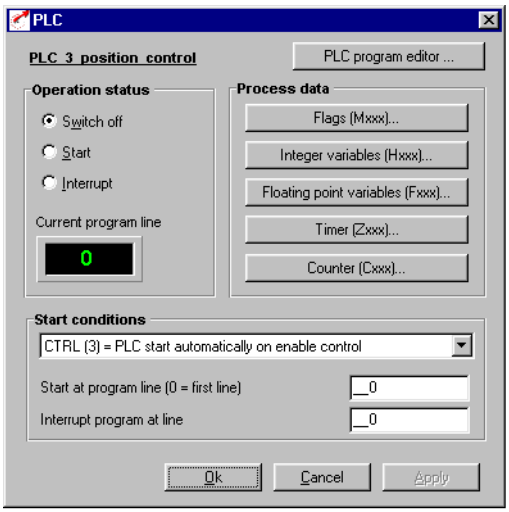
Para-meters	Description	
452-PLCCT	**Starting conditions of the process control** Parameter PLCCT defines starting position of the process program.	
	TERM(0)	PLC-start via terminal Function selector of an input has to be set to Fixxx = PLCGO. (0 -> Stop program, 1 -> Start program)
	PARA(1)	PLC-start via parameter Manual changing of operating status PLCST
	AUTO(2)	PLC-start automatically on power on, parameter operating status has to be set to GO
	CTRL(3)	PLC-start simultaneous on enable control PLC-stop simultaneous on disable control
450-PLCST	**Operation status of process control** This parameter allows start/stop (depending on parameter 452-PLCCT=PARA) resp. shows the actual operating status of the process program.	
	OFF(0)	PLC-switch-off/switched-off process program
	GO(1)	PLC-start/process program processing
	BRKPT(2)	PLC-Interruption of process program With GO the operation continuous. Independent from the control location it is possible to interrupt (BRKPT) resp. finish (OFF) the process program via parameter at every time. Restart process program from line of interruption with GO, if the condition of the control loacation is still given (e.g. terminal is still set). In case of a changed condition, set parameter to OFF.
451-PLCPL	**Current program line.** Shows the current processed program line. The line number can be seen in the digital oscilloscope, too.	
454-PLCSN	**Start at program line (0 = first program line).** Program processing starts at the line, mentioned in PLCSN. It is convenient, if there are different independent routines are available in one program.	

list 2.1 Parameters of PLC

Para-meters	Description
455-PLCBN	**Interrupt program at line x (Breakpoint).** Program will be interrupted at line, mentioned in PLCBN; parameter 450-PLCST is going to status BRKPT. Restart of program with 450-PLCST=GO(1)
461-PLC_M	**Flag (0/1)** Access to process program M000...M255
460-PLC_H	**Integer variables (32 bit)** Value range from 2^{-31} to 2^{31}. In case of a link with floating point variables or parameters the part after the decimal place will not be considered. No rounding will be made. Access to process program H000...H127
465-PLC_F	**Floating point variables** Access to process program F000...F127 Value range: -3,37x10^{38} to 3,37x10^{38}
462-PLC_Z	**Timer (32 bit)** Time basis 1 ms Access to process program Z00...Z11 Timer will be set to a value and returns to 0.
463-PLC_C	**Counter for indicated addressing (8 bit)** Access to process program C00...C10
466-PLC_I	**Code of digital and analog inputs (bit coded)** The code is also readable in the program as special variable INPUT. ISD00-ISD03 Bit 0 - Bit 3 IED00-IED07 Bit 4 - Bit 11 ISA00 - ISA01 Bit 12 - Bit 13
464-PLC_O	**Code of digital outputs (bit coded)** The code is also writable in the program as special variable OUTPUT. OSD00-OSD02 Bit 0 - Bit 2 OED00-OED03 Bit 4 - Bit 6 In order to set outputs out of the program the corresponding function selector has to be set to FOxxx = PLC.
468-PLCPJ	**Name of PLC-program (project name)** Project name will be defined during building the process program (text declaration). The name describes directly the text declaration file (project name.txt) (max. 32 figures without special characters, blanks will be ignored)

list 2.1 Parameters of PLC

An easy setting of the a.m. parameters allows the PLC-function window (expanded main window -> PLC or via „basic settings/PLC“ according to selected PLC-preset-solution):



picture 2.5 DRIVEMANAGER - PLC-function window

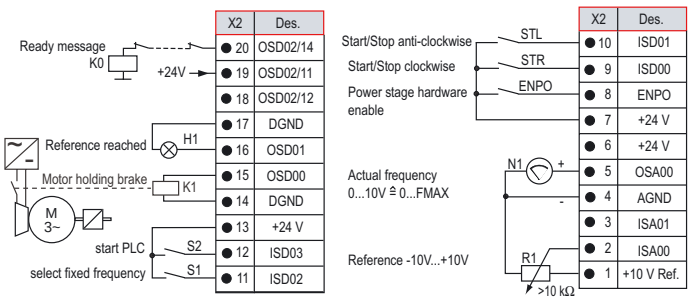
Pre-set solutions

Special pre-set solutions are included in the fimware to optimize parameter setting of PLC-applications:

PLC_1 - PLC-start via terminal, analog reference & fixed frequency

Function:

- Analog speed pre-set for two rotation directions via ISA0
- Switching-over to fixed frequency via input ISD02
- Start PLC-program via ISD03
- Motor control via inputs ISD00/01
- All further I/O's are available at the PLC-process program.



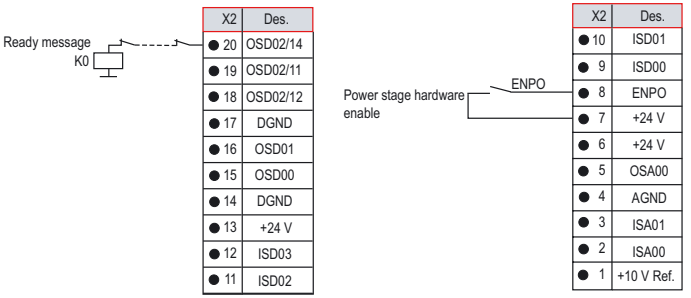
picture 2.6 Terminal assignment PLC_1

PLC_2 - PLC-autostart, motor- & I/O-control via PLC

Function:

- Motor control via PLC
- PLC-process program starts automatically on „power-on"

- All further I/O's are available at the PLC-process program.



picture 2.7 Terminal assignment PLC_2

Motor control will be controlled from the PLC-program. Independent from the selected control mode VFC, SFC or FOR.

Start motor control:

```
N010 SET ENCTRL=0          ; Switch-off motor control
N020 SET ENCTRL=1          ; Switch-on motor control
```

Condition: Control location selector 260-CLSEL=PLC (control via process program). Is already adjusted at pre-set PLC_2 .

Reference assingment:

```
N030 SET REFFRQ=H000       ; Frequency reference
                           ; (Value without decimal place)
N040 SET REFFRQ=F010       ; Frequency reference
                           ; (Value as floating point number)
```

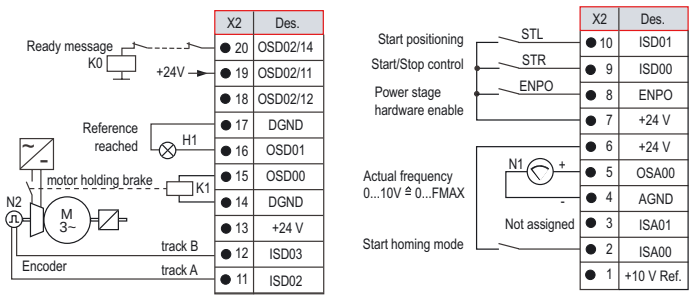
Condition: Parameter of reference selector must be set to FOPT1 (280-RSSL1=FOPT1 oder 281-RSSL2=FOPT1). That means reference channel FOPT1 will be used via PLC-program at PLC-software for reference set. At pre-setting PLC_2 (280-RSSL1=FOPT1) it is already adjusted .

PLC_3 PLC-position control

Function:

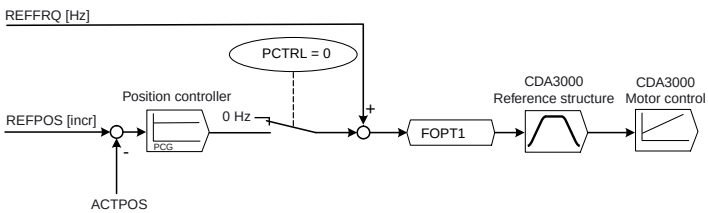
- Position control with PLC-program and control via inputs ISD01 (start positioning - reference 2) and ISA01 (start homing mode)
- Control mode FOR
- Encoder evaluation via ISD02/03
- Motor control via input ISD00
- All further I/O's are available at the PLC-process program.

For the complete function you have to download the additional PLC-program „PLC_3_position control.plc“. See DRIVEMANAGER-directory „..\user-data\PLC“ ab.



picture 2.8 Terminal assignment PLC_3

In control mode FOR control of rotor position is possible, too. The position control is suitable for slow changing processes. Therefore a P-controller will be connected in series to the reference path in the known CDA3000 reference structure.



picture 2.9 Position control with PLC-process program

Switch-on/off the position controller via the following commands:

```
N010 SET PCTRL = 1       ; Activate position controller
N020 SET PCTRL = 0       ; Deactivate position controller
```

The P-gain 819-PCG of the position controller is adjustable by the DRIVEMANAGER (Function window „basic setting“) or via parameter access in the process program.

Functions like homing mode, jog mode and cam switch are not part of the firmware. However, they can be realised with a PLC-program (sub-program).

A position profile generator is not included. That means the position controller sends a step signal to the reference structure resp. speed controller. Via suitable parameter setting of the acceleration ramps of the reference structures it can be straightened. An overswinging in the target position is possible.

The pre-set solution PLC_3 will also be described in chapter 3.6. as example.

PLC-error messages

The process control generates different error messages, collected in list 2.2.

Error	Description
E-PLC 210	Error actuated via PLC (SET ERR = 1, Mxxx mit Mxxx = 1).
E-PLC 211	Error at sub-program calls / reset via CALL / RET. Stack underflow: unexpected RET w/o previous CALL. Stack overflow: max. nesting scopes (250 CALLs) reached.
E-PLC 212	Error at writing parameters (buffer full). Writing out of the interrupt via a buffer with max. 30 entries, whereas the buffer itself will be handled in the main loop. Receiving this message means that the buffer limit is reached, that means the main loop could not process all parameter assignments. Command WAIT PAR effects in stopping the program processing as long as all parameters are written and the buffer is empty. In case of too many parameter accesses (more than 30 continuous parameter accesses) or during backup of parameter writing access at the further program processing an WAIT PAR should be effected intermediately.
E-PLC 213	Error at writing parameters. Parameter does not exist, no field parameter, value range default, value not writable, etc.
E-PLC 214	Error at reading parameters. Parameter does not exist or no field parameter..
E-PLC 215	Internal error: No code available or program instructions not executable.

list 2.2 *Error messages of PLC-process control*

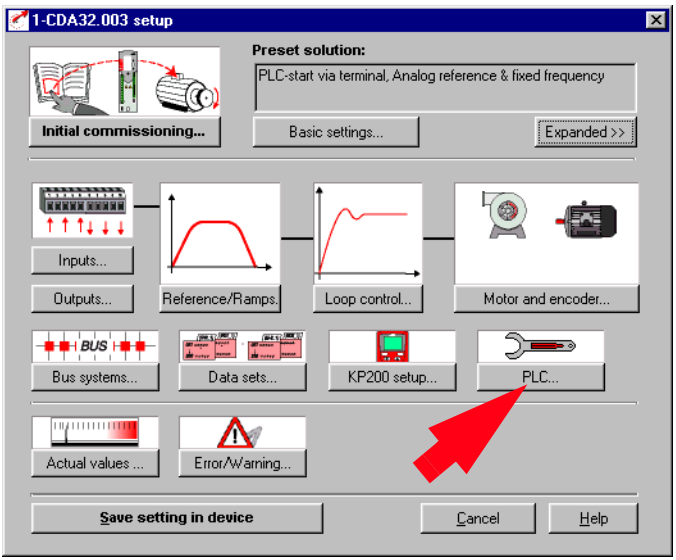
Error	Description
E-PLC 216	Internal error: No code available, program instructions not executable or jump to unused address. Error occurs during loading of process program, if there is still a process program is running in the controller and the new program contains other line numbers. If not necessary, switch-off PLC during loading of a program.
E-PLC 217	At dividing in process program a division by zero occured.
E-PLC 218	PLC is not available in this software version.
E-PLC 219	Motor identification is not available in this software version.
E-PLC 220	Error at floatingpoint operation in the process control. The process control is in waiting condition ans shows the faulty program line. Check the terms of the interruption (value ranges) at floating point operations. If necessary correct the process program resp. the faulty program line. **Please note:** At floating point calculations value range contaminations (0... 3.37E+38) can occur. May be at a comparison of two floating point variables the terms of interruption cannot be reached. Pay attention at programming that there are clear and plausible value ranges.
E-PLC 221	Cycle time of process control is exceeded, that means the program procesing takes more time as scheduled.

list 2.2 Error messages of PLC-process control

2.1.3 DRIVEMANAGER

PLC-software as well as PLC-editor can be operated resp. parameters can be set with each DRIVEMANAGER-version from V3.20. With the DRIVEMANAGER a serial commissioning of the CDA3000-PLC is possible with only one device data set.

Condition is an installed DRIVEMANAGER for working with PLC-functionality resp. PLC-editor, because it is integrated. Parallel working with the DRIVEMANAGER and the PLC-editor functions is possible.



picture 2.10 DRIVEMANAGER-main window for CDA3000

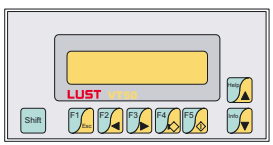
2.2 Operator panels for c-line DRIVES

With the graduated raw of operator panels high-quality, efficient, multiple use products are available for the system. They guarantee a high-operating comfort and a high-functionality and are equipped with an attractive design, too.

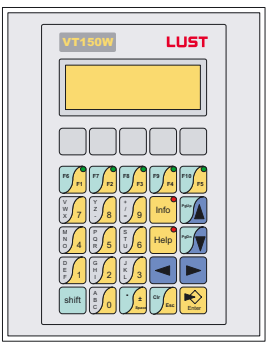
- Presentation of texts, picture, bargraphs, bitmap-pictures and inciting graphics*
- Recipe management*
- Saving of files to avoid unautorised access via passwords
- Indication of system messages
- Indication of warning messages*
- Communication via bus systems LUSTBUS (RS232) or CAN_{open}*
- Free-programmable function buttons
- Individual possibilities of labelling via slide-in strip*
- Clearly-presented multilingual programming software

* however not available at all types

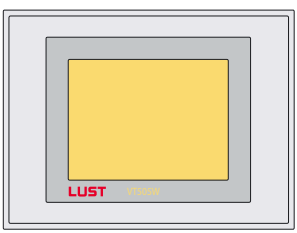
Four different operator panels are available:



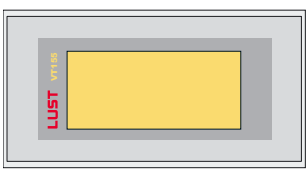
VT050
Text display,
2x20 characters, 8 buttons



VT150W
Text display,
4x20 characters, 25 buttons



VT505W
Touch-Screen, 4 blue levels,
320x240 pixel (5,6“)



VT155W
Touch-Screen, 4 grey levels,
240x128 pixel

Further displays as well as customized operator panels are available upon request.

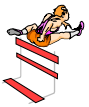
operator panels can be used for operating and parameter setting of drive controllers of c-line Drives:

- CDA3000 (Standard, HF, PLC)
- CDD3000
- CTC3000

Interfaces

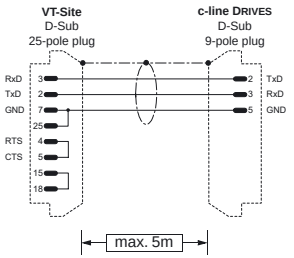
The operator panels are available with serial LUSTBUS Interface RS232 or with a CAN_{open} interface. At some panels we differ between programming and communicating interface.

At communication interface **RS232** (LustBus-interface) a Peer-to-Peer communication between axis controller and operator panel is possible. This connection is suited for single axis drives. Programming of the panel is also done via RS232-interface.



When connecting the operator panel via serial RS232-interface the service interface (e.g. for PC-tool DriveManager) is not longer resp. can only be used with disconnected operator panel.

Assignment of the connecting cable is shown in picture 2.11.

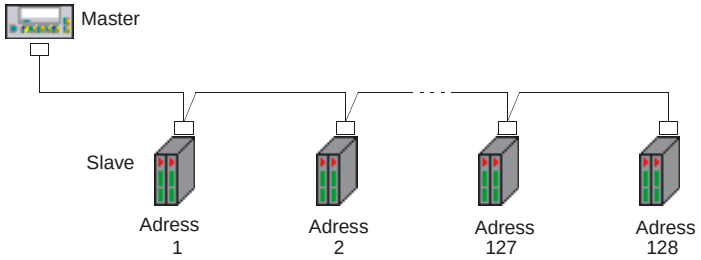


picture 2.11 RS232-connection of operator panels

For the connection there are cord sets in length of 3 m (OPK-RS03) and 5 m (OPK-RS05) at disposal.

With an integrated communication interface **CAN_{open}** in the operator panel an autonomous network operation with one operator panel and the drive controllers of c-line-DRIVES is possible. The operator panel set the parameters for all controllers, connected to the bus. For programming the panel a serial RS232-interface is available.

The CAN_{open}-interface of operator panels has a master slave structure (s. picture 2.12).



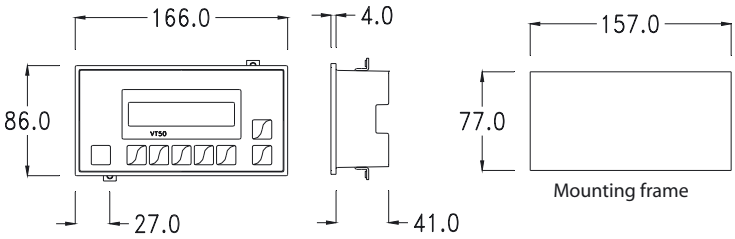
picture 2.12 CAN_{open}-network with operator panel

It is the job of the operator panels (Master) to initialise and to reconfigure the drive controllers (Slaves); furthermore it is necessary for the control of the communication status of the devices in the network. For communication the slave-devices have to be initialized (operating status) and parameters have to be set from the master-device.

Technical data VT050

Article code of panel: **VT050 00000N** **VT050 000CNN**	**Characteristics of panel:**		
Display			
Type	Text LCD	●	●
Background lighting	LED	●	●
Lines and characters	2 x 20	●	●
Representional format [mm]	73,5 x 11,5	●	●
Character matrix in text mode [Pixel]	5 x 7	●	●
Size of characters [mm]	3,2 x 5,5	●	●
Contrast adjustment	Trim pot	●	●
Character sets	ASCII, Katakana	●	●
Keyboard			
System-/ Function-/Alphanumerical keys	8 / 5 / -	●	●
User memory			
Project (Flash EPROM) [kB]	256 kB	●	●
Interfaces			

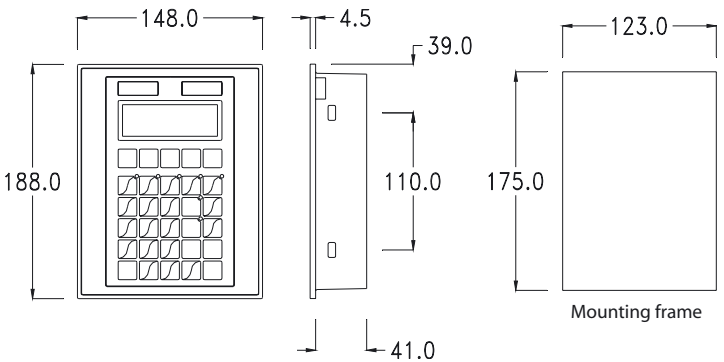
Article code of panel: VT050 00000N VT050 000CNN	Characteristics of panel:		
Serial port MSP (25 pin female)	RS232/RS422/RS485/TTY 20 mA		●
Serial port ASP-8 (pin female)	RS232	●	
Networks			
Integrated	CAN_{open}	●	
Characteristics			
Project-languages	4	●	●
Password-level /Bit-Password	- / 8 bit	●	●
Pages/pages-help	127 / 127	●	●
Variables per page	8	●	●
Variable form	DEC, HEX, BIN, BCD, ASCII, Floating Point	●	●
Dynamic texts	Depending on size of project memory	●	●
ISA-alarms/information messages	- / 128	●	●
Alarm buffer	-		
Recipes	-		
Dimensions			
Outer W x H x D [mm]	166 x 86 x 41		
Mounting frame W x H [mm]	157 x 77		
Technical data			
Power supply	24 Vdc (18 .. 32 Vdc)		
Power absorbed at 24 Vdc	5 W		
Protection fuse	5 x 20 mm - 315 mA (Fine wire fuse type F)		
Protection level	IP65 (Front)		
Working temperature	0 .. +50°C		
Storage and transportion temperature	-20°C .. +60°C		
Humidity	<85%		
Weight	0,5 kg		
Certifications and approvals	CE, RINA (UL upon request)		



Technical data VT150W

Article code of panel: VT150W 00000N VT150W 000CNN	Characteristics of panel:		
Display			
Type	Text LCD	●	●
Background lighting	LED	●	●
Lines and characters	4 x 20	●	●
Representional format [mm]	70,4 x 20,8	●	●
Character matrix in text mode [Pixel]	5 x 7	●	●
Size of characters [mm]	2,95 x 4,75	●	●
Contrast adjustment	Trim pot	●	●
Character sets	ASCII, Katakana	●	●
Keyboard			
System-/ Function-/Alphanumerc keys	9 / 5 / 11	●	●
LEDs for function/operation buttons	5 / 2	●	●
User memory			
Project (Flash EPROM) [kB]	256 kB	●	●
Interfaces			
Serial port MSP (25 pin female)	RS232/RS422/RS485/TTY 20 mA		●
Serial port ASP-8 (pin female)	RS232	●	
Networks			
Integrated	CAN_{open}	●	
Characteristics			
Project-languages	6	●	●
Password level/Bit-password	10 / 8 bit	●	●
Pages/pages help	1024 / 1024	●	●
Variables per page	16	●	●
Variable form	DEC, HEX, BIN, BCD, ASCII, Floating Point	●	●

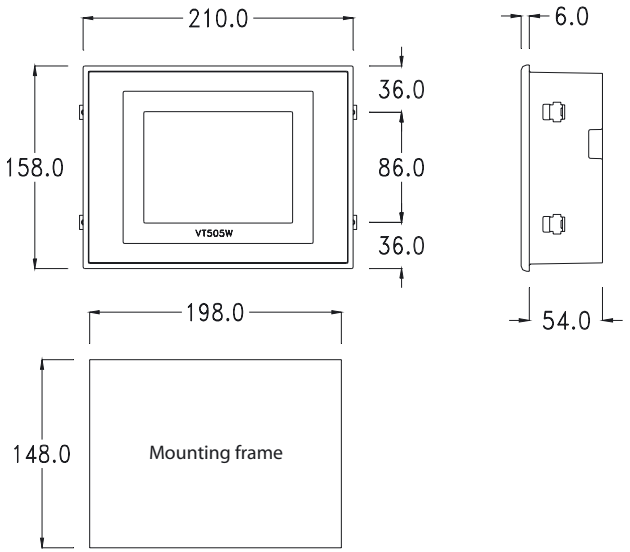
Article code of panel: VT150W 00000N VT150W 000CNN	Characteristics of panel:		
Dynamic texts	Depending on size of project memory	●	●
ISA-Alarms/Information messages	- / 1024	●	●
Alarm buffer	-		
Recipes	-		
Dimensions			
Outer W x H x D [mm]	148 x 188 x 41		
Mounting frame W x H [mm]	123 x 175		
Technical data			
Ppower supply	24 Vdc (18 .. 32 Vdc)		
Power absorbed at 24 Vdc	15 W		
Protection fuse	5 x 20 mm - 800 mA (Fine wire fuse type F)		
Protection level	IP65 (Front)		
Working temperature	0 .. +50°C		
Storage and transportion temperature	-20°C .. +60°C		
Humidity	<85%		
Weight	0,7 kg		
Certificates and approvals	CE		



Technical data VT505W

Article code of panel: VT505W 00000N VT505W 000CNN	Characteristics of panel:		
Display			
Type	Graphic, LCD 4 blue levels STN	●	●
Touch Screen	Matrix 20 x 16	●	●
Background lighting	CCFL-tube	●	●
Lifetime [hours]	15000	●	●
Resolution	320 x 240 (5,6“)	●	●
LInes and characters	16x40 / 8x20 / 4x10	●	●
Representional format [mm]	115,2 x 86,4 (5,6“)	●	●
Character matrix in text mode [Pixel]	8x15 / 16x30 / 32 x 60	●	●
Size of characters [mm] x1 / x2 / x4	2,8x5,2 / 5,6x10,4 / 11,2x20,8	●	●
Contrast adjustment	Software	●	●
Character sets	programmable sets	●	●
User memory			
Project (Flash EPROM) [kB]	512 kB	●	●
Data storage (Flash EPROM) [kB]	16 kB	●	●
Interfaces			
Serial port MSP (25 pin female)	RS232/RS422/RS485/TTY 20 mA	●	●
Networks			
Integrated	CAN_{open}	●	
Characteristics			
Multilingual texts	4	●	●
Password-level /Bit-Password	10 / 8 bit	●	●
Pages/pages help	64 / 64	●	●
Variables per page	24	●	●
Variable form	DEC, HEX, BIN, BCD, ASCII, Floating Point	●	●
Dynamic texts / picture index	Depending on size of project memory	●	●
ISA-Alarms/Information messages	- / 256	●	●
Alarm buffer	-		
Recipes (number of/variables per recipe)	128 / 256	●	●
Bar graphs per page	24	●	●
Project pictures	BMP, JPG; TIFF, PSD, WMF, PNG, EPS, ...	●	●
Clock	Software (w/o buffer battery)	●	●

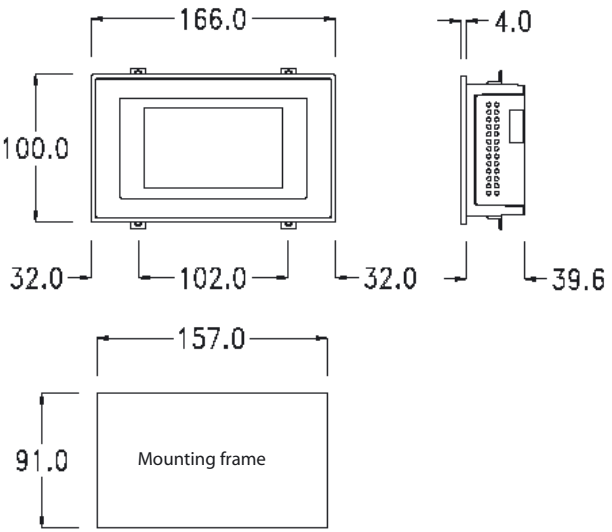
Article code of panel:	Characteristics of panel:
VT505W 00000N VT505W 000CNN	
Dimensions	
Outer W x H x D [mm]	210 x 158 x 54
Mounting frame W x H [mm]	198 x 148
Technical data	
Power supply	24 Vdc (18 .. 32 Vdc)
Power absorbed at 24 Vdc	10 W
Protection fuse	5 x 20 mm - 800 mA (Fine wire fuse type F)
Protection level	IP65 (Front)
Working temperature	0 .. +50°C
Storage and transportation temperature	-20°C .. +60°C
Humidity	<85%
Weight	1,4 kg
Certificates and approvals	CE



Technical data VT155W

Article code of panel: VT155W 00000N VT155W 000CNN	Characteristics of panel:		
Display			
Type	Graphic, LCD 4 grey levels STN	●	●
Touch Screen	Analog/20x8 (12x16 pixel)	●	●
Background lighting	LED	●	●
Resolution	240x128 pixel	●	●
Lines and characters	16x40 / 8x20 / 4x10	●	●
Representional format [mm]	94,5x54,5	●	●
Character matrix in text mode [Pixel]	6x8 / 12x16 / 24x32	●	●
Size of characters [mm] x1 / x2 / x4	2,3x5,2 / 4,6x5,8 / 9,1x11,7	●	●
Contrast adjustment	Software	●	●
Character sets	Programmable sets	●	●
User memory			
Project (Flash EPROM) [kB]	640 kB	●	●
Data storage (Flash EPROM) [kB]	16 kB + 8 kB (Alarm buffer)	●	●
Interfaces			
Serial port MSP (25 pin female)	RS232/RS422/RS485/TTY 20 mA		●
Serial port Port ASP (8 pin female)	RS232	●	●
Networks			
Integrated	CAN_{open}	●	
Characteristics			
Project-languages	4	●	●
Password-level /Bit-Password	10 / 8 bit	●	●
Pages/pages help	64 / 64	●	●
Variables per pages	32	●	●
Variable form	DEC, HEX, BIN, BCD, ASCII, Floating Point	●	●
Dynamic texts/picture index	Depending on size of project memory	●	●
ISA-Alarms/Information messages	256 / 256	●	●
Alarm buffer	220		
Recipes (number of/variables per recipe)	128 / 256	●	●
Bar graphs per page	32	●	●
Project pictures	BMP, JPG; TIFF, PSD, WMF, PNG, EPS, ...	●	●
Clock	Hardware (with Supercapacitor)	●	●

Article code of panel:	Characteristics of panel:
VT155W 00000N VT155W 000CNN	
Dimensions	
Outer W x H x D [mm]	166 x 100 x 43,6
Mounting frame W x H [mm]	157 x 91
Technical data	
Power supply	24 Vdc (18 .. 32 Vdc)
Power absorbed at 24 Vdc	10 W
Protection fuse	5 x 20 mm - 800 mA (Fine wire fuse type F)
Protection level	IP65 (Front)
Working temperature	0 .. +50°C
Storage and transportation temperature	-20°C .. +60°C
Humidity	<85%
Weight	0,5 kg
Certificates and approvals	CE (UL upon request)



Programming operator panels

Programming of operator panels by means of a graphic PC-user interface. Only one software for all operator panels. The CD-ROM software package VTWINCD includes:

- Programming software in 5 languages
- Suitable for operating system Windows 95/98/NT/2000/ME/XP
- Software- and hardware-manuals in 5 languages

For the programming of operator panels various programming cables resp. adapters are available. The following list 2.3 shows which cable is needed for a defined panel (X).

	Programming cable CVCOM 11102	Programming adapter CVCOM 25F8M
VT050 00000N	X	
VT050 000CNN	X	X
VT150W 00000N	X	
VT150W 000CNN	X	X
VT505W 00000N	X	
VT505W 000CNN	X	
VT155W 00000N	X	X (optional)
VT155W 000CNN	X	X

list 2.3 Using program cables

2.3 SmartCard / KeyPad

For serial commissioning we recommend SmartCard SC-XL. It has a large data storage, so that the PLC-program with all PLC-configuration parameters can be saved.

The SmartCard SC-XL is no replacement for the card SC. The SmartCard SC (A028.V01.1) used up to now cannot be used any longer, due to a loss of memory capacity. An error at the KeyPad is indicated when trying to write on such a card.

To support the SmartCard SC-XL the KeyPad KP200-XL will be used. The KP200-XL is compatible with the functions of the KP200 (0840.0000). The SmartCard SC (A028.V01.1) will also be supported by the KP200-XL.

Parallel to the already used KP200 we offer the KP200-XL, which has to be ordered with a separate article number.

Attention: The CDA3000-PLC-operation with KeyPad KP200 and/or SmartCard SC can cause malfunction and is not allowed.

2.4 Training

To introduce the CDA3000-PLC system we offer different trainings. Trainings methods are lectures, discussions, demonstrations and practical examples.

Training how to program CDA3000-PLC

Target:	Learning how to handle and program the CDA-PLC software
Contents:	• Chracteristics, efficiency, application limits of the PLC-function. • Making of process programs with PLC – PLC-Program editor – Survey of command sets – PLC-parameter setting • Commissioning of CDA3000-PLC • Possibilities of use and training examples
Conditions:	• Experience in working with inverter drive systems CDA3000 and DRIVEMANAGER • MS-Windows system-knowledge
Date:	1 day seminar
Time:	8.00 a.m. to 5.00 p.m.
Location:	Lust Antriebstechnik GmbH, Lahnau
Number of participants:	6-10 Persons
Instructors:	Andreas Kling, Jörg Brinkemper
Languages:	German or English

Handling and programming of operator panels

Target: Learning how to handle and program Lust-operator panels

Contents:
- Survey of operator panel components
- Characteristics and efficiency of operator panels
- Connection of operator panels with the drive controller
- Configuration and programming of operator panels
- Possibilities to use and training examples

Conditions:
- Experience in working with the inverter drive system CDA3000 and DRIVEMANAGER
- MS-Windows system-knowledges

Date:

Time:

Location:

Number of participants:

Instructors:

— in preparation —

Language: German or English

2.5 Service-know-how for your projects

Use the know-how of our qualified application engineers. Get decisive advantages to realize efficient sub-automation units with our experience. Solutions, made exactly for your requirements, your demands.

Compatibility from the beginning:

- Projectmanagement
  - Work out project order
  - Detailed project planning
  - Execution of project
  - Sorted finishing of the project and saving the knowledge
- Engineering
  - Analysis of the conceptual formulation
  - Concipate and design of the system
  - Apply the components for the system
  - Introduction of different solution opportunities
- Service of making software
  - Making of PLC- or operator panel software acc. to specification
  - Test the software in the system
  - Transfer of software incl. all documents
- Commissioning of a sub-automation unit
  - Prepare the commissioning resp. function test
  - Commissioning of the system at site
  - Making of commissioning protocol with all measuring data

Use our experience of over 30 years and together we will find a solution for your complex jobs.

3 Sub-automation units

By the mentioned examples in this chapter it deals only with programming exercises for the CDA3000-PLC. Neither the conceptional formulation nor the solutions are checked under safety specifications.

The examples show which solutions are possible with integrated sequence control and how a typical program looks like. The mentioned programs are saved in the DRIVEMANAGER sub-directory „userdata/PLC“.

You will agree that under these circumstances Lust Antriebstechnik GmbH is not responsible and cannot assume any liability for using this programs or parts of it.

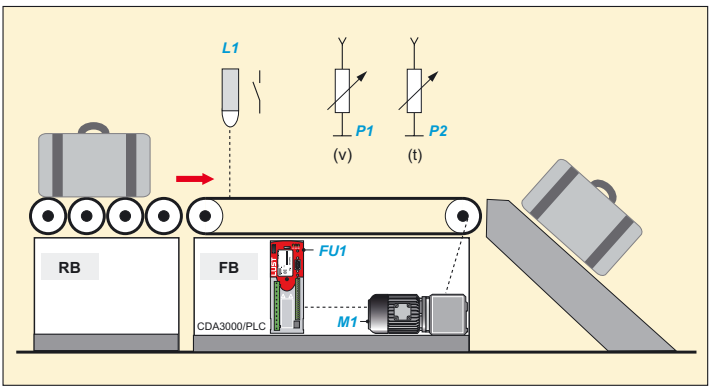
3.1 Time-controlled luggage belt-drive

Description of the function

If the luggage interrupts the light beam of the light-barrier L1, the conveyor belt FB starts. Max. conveyor speed will be pre-selected via potentiometer P1.

Cycle length of conveyor FB will be set by means of potentiometer P2. If the time set with potentiometer F2 is passed, the inverter M1, FU1 will be switched-off and the belt runs off.

Technology scheme



picture4.1 Time-controlled luggage belt-drive

Process program

```
; Process programme for CDA-PLC, example luggage belt-drive

;Initialisation
;Reference setpoint via analog input ISA0 with caling in the
device parameters

%TEXT(luggage belt)
DEF H000=max_cycle length
DEF H001=auxiliary variable
DEF H002=analog input
END

%P00
N010 SET H000=20000;      max. value cycle length in ms
N020 SET H001=H000;       auxiliary variable
N021 SET H001:1023;       resolution of analog input 10Bit

N050 JMP (IS00=0) N050;   wait for release signal
;                         of light barrier
N060 SET ENCTRL=1;        start control

;Read timer value of analog input ISA1
N070 SET H002=ISA1;       analog value in H002
N075 SET H002*H001;       Calculate resolution of analog input

;Initialise timer
N080 SET Z00=H002
N085 JMP (Z00!=0) N085;   wait for timer run-off

;Stop control
N090 SET ENCTRL = 0

N100 JMP N020;   jump back
END
```

Further typical applications are melting and mixing plants for paints and other materials as well as centrifuges, mills and shredder.

3.2 Drill feed unit

The drilling appliance consists of a drilling spindle, feed unit and conveyor unit. In the following we like to explain the sub-automation unit „drilling“ and with it spindle and feeding drive.

Basic position

The drilling unit is in basic position, if

- the feed unit is above (S1 damped)
- the drilling appliance is free (S3 not damped)
- spindle M1, FU1 is switched-off.

Description of the function

If the workpiece W1 is in the drilling appliance (S3 damped) and the start button b1 is pressed, the drilling unit receives okay for processing.

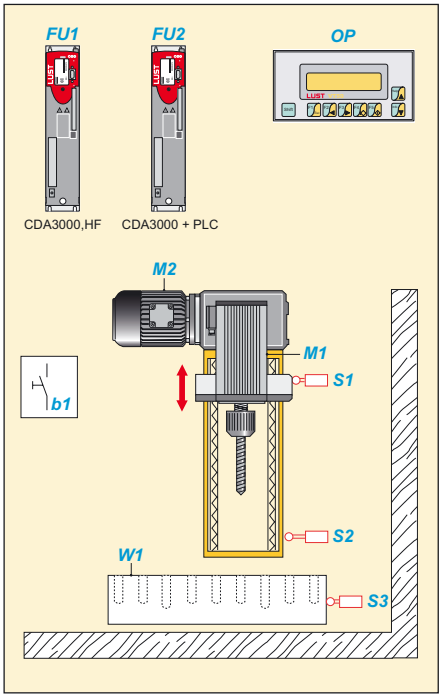
The drilling spindle M1,FU1 runs to the working speed. If drilling spindle M1,FU1 reaches the working speed, the drilling unit sinks over the feeding drive M2,FU2.

Approaching the lower sensor (S2) the point of reversal is reached. The feeding drive M2,FU2 reverses, so that the drilling spindle runs in basic position again.

Approaching the upper sensor (S1) the feeding drive M2,FU2 and the drilling spindle M1,FU1 stop automatically. The workpiece will be carried via conveyor unit and the process can start again.

Set spindle speed and feeding speed via operator panel OP.

Technology scheme



picture4.2 Drill feed unit

Process program

```
;Program example feed unit

;Inputs:
;M001=Start feed
;IS01=Pre-stop opener
;IS02=Upper limit switch opener
;IS03=Lower limit switch closer

%TEXT (feed)
DEF H000 = Reference_0
DEF H001 = Timer_1
DEF M002 = Initialisation
DEF H002 = Quick-jog frequency
DEF H003 = Slow-jog frequency
DEF H004 = Waiting time
DEF H010 = Quick-jog_positive
DEF H011 = Quick-jog_negative
DEF H012 = Slow-jog_positive
DEF H013 = Slow-jog_negative
DEF M001 = Start motion
```

```
END

; Process program for CDA-PLC
%P00
N005 SET H000=0;          Reference 0
N010 SET H001=1000;       Value for timer 1
N015 JMP (M002=1) N031;   Jump over initialising
N020 SET H002=70;         Reference feed Hz
N030 SET H003=20;         Slow-jog
N031 SET H004=200;        Waiting time working point
N032 SET M002=1

N040 SET H010=H002;       Variable quick-jog positive
N041 SET H011=H002;       Variable quick-jog negative
N042 INV H011

N050 SET H012=H003;       Variable slow-jog positive
N051 SET H013=H003
N052 INV H013

N060 JMP (M001=1) N100;   Start feed motion
N065 JMP (IS02=0) N040;   Upper limit switch reached
N070 SET REFFRQ=H010;     Move to upper limit switch
N075 JMP (IS02=1) N075;   Wait for reaching limit switch
N080 SET REFFRQ=H000;     Stop axis
N081 SET OS00=1;          Axis in upper position
N082 SET OS01=0
N085 JMP N040;            Close loop

N100 SET REFFRQ=H011;     Start feed quick-jog
N105 SET OS00=0;          Axis runs
N109 JMP (M001=0) N040
N110 JMP (IS01=1) N109;   Control contact pre-stop
N120 SET REFFRQ=H013;     Switch-over to low-jog
N129 JMP (M001=0) N040
N130 JMP (IS03=0) N129;   Wait for lower limit switch
N140 SET REFFRQ=H000;     Stop axis

N150 JMP (M001=0) N040
N151 SET REFFRQ=H012;     Move back to pre-stop
N152 JMP (M001=0) N040
N153 JMP (IS01=0) N152
N154 WAIT H004
N155 JMP N120;            Feed

N190 JMP N040
END                       ;End of program
```

Further typical applications are belt and carriage drives, lifting and tota-ting tables, drives for doors and gates and for example pumping stations with float switch.

3.3 Shredder with overload detection

Shredder (hacking machines) are used in various applications, for example in the food industry, construction industry or in offices. Frequently, problems during this process are caused by blocking of the drive.

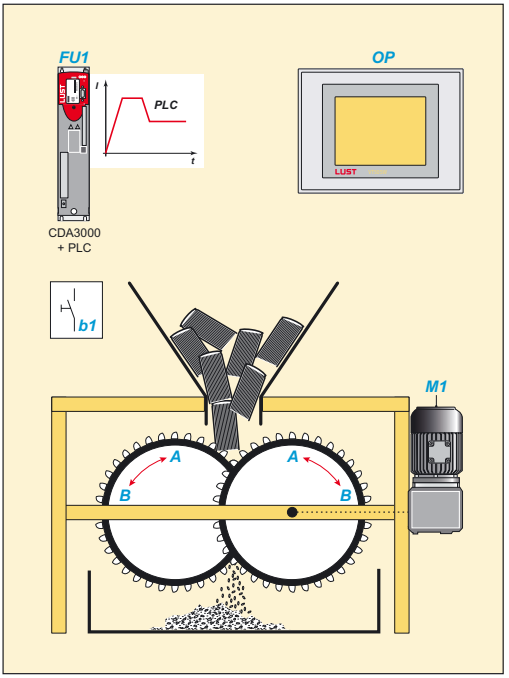
The example shows a shredder with overload detection and automatic free-running (back-off) of the roller at blocking. The user can have an effect on this behaviour by means of setting the parameters of the overload detection. Therefore the user set the time of reaction at overload, minimum time for overload and the number of free-running trials per recipe management.

Description of the function

Upon actuating the drive FU1,M1 via button b1 the rollers of the shredder rotate with an adjustable fixed frequency in forward direction A. During the process the PLC of FU1 manages the control of the motor current I up to a settable threshold. In case of exceeding the reference, that means at overloading or blocking or the roller, the drive stops as soon as the set overload time is exceeded. After a set time a backwards rotation of the roller in direction B will be initiated (free-running). The period of rotating backwards can be controlled via PLC-time as well. After the backward rotation the standard operation in direction A will be actuated again. The drive stops, if it deals with an multiple overload (quantity can be set in PLC) of the time, set in parameters.

Via the operator panel OP and by means of recipe management all timer and threshold values will be set depending on the material, shall be shreddered.

Technology scheme



picture4.3 Shredder with overload detection

Process program

```
; Process program for CDA shredder

; Inputs
; IS00 - Start control
; IS01 - Start process program

;Ausgänge
; OS00 - Warning overload
; OS01 - Reference reached
; OS02 - S-RDY

%TEXT(Shredder)
DEF H001 = Timer_overload
DEF H002 = Break time_overload
DEF H003 = Reversing time
DEF H004 = Timer_repeat
DEF H005 = Counter_reverse
DEF H006 = max_repeats
DEF H010 = Reference_process
DEF H011 = Reference reverse
DEF F000 = Actual_apparent current
DEF F001 = Threshold value
```

```
END

%P00
;Init
N005 SET H001 = 500;              Reaction time at overload (ms),
Z000
N010 SET H002 = 500;              Break time at overload (ms)
N015 SET H003 = 3000;             Reversing time (ms)
N020 SET H004 = 20000;            Timer repeats reverse
N025 SET H005 = 0;                Counter reverse trials
N030 SET H006 = 3;                Max. value counter
N035 SET H010 = 50;               Reference forward (Hz)
N040 SET H011 = -20;              Reference reverse (Hz)

N045 SET F000 = 0;                actual apparent current
N050 SET F001 = 1;                Threshold overload

;Hauptprogramm
N055 SET H005 = 0
N060 SET Z001 = H004;             Timer repeats reverse
N065 SET REFFRQ = H010;           Forward

N070 SET F000 = PARA[408];        Call for apparent current
N075 JMP (Z001 = 0) N055;         Time repeats reset
N080 JMP (F000 < F001) N070

N085 SET Z000 = H001;             Timer reaction time overload

N090 SET F000 = PARA[408];        Call for apparanet current

N095 JMP (F000 < F001) N070;      Overload gone?
N100 JMP (Z000 != 0) N090;        Timer runs off?

N105 SET REFFRQ = H000;           Stop drive
N110 WAIT H002   ;                Wait break time

N115 JMP (H005 = H006) N145;      Reverse to often

N120 SET REFFRQ = H011;           Reference reverse
N125 SET H005 + 1;                Counter reverse
N130 WAIT H003   ;                Reverse time
N135 WAIT H002   ;                Break time

N140 JMP N065    ;                Jump back to forward

N145 NOP
END                       ;       End of program
```

An other possibility to solve this applications shows the following program- The process is modified a little bit compared with the a.m. description of the function.

```
; PLC-program for shredder

;parameter
; 270-FFIX1 = Reference forward
; 271-FFIX2 = Reference reverse
; Zeile  55 Special function warning current

; inputs
; IS00 - Start forward
; IS01 - Start reverse
```

```
; IS02 - Stop
; IS03 - Fault reset

; outputs
; OS00 - c_rdy
; OS01 - Reference reached
; OS02 - Warning current limit

%TEXT(shredder)
DEF H000 = Value timer reverse
DEF H001 = Value timer repeat
DEF H002 = Max repeat
DEF H003 = Repeat
DEF M000 = STA_WIS
DEF Z000 = Timer revers
DEF Z001 = Timer repeat
DEF F000 = Reference forward
DEF F001 = Reference reverse
DEF F003 = Reference 0
END

%P00
;Init
N005 SET H000 = 5000;     Timer reverse
N006 SET H001 = 600000;   Timer repeat
N007 SET H002 = 3;        Max. count repeat
N010 SET F000 = PARA[270];Reference forward
N011 SET F001= PARA[271];Reference reverse
N012 SET F003=0

;main
N030 SET REFFRQ=F003
N035 SET OS02=0

N040 JMP (IS00=1) N050;   Start forward
N041 JMP (IS01=1) N200;   Start reverse
N043 JMP N040

N050 SET H003=1
N051 SET Z001 = H001;     Timer repeat reverse
N052 SET ENCTRL=1;        Enable control
N053 SET REFFRQ = F000;   Forward
N054 WAIT 2000;           Waiting for acceleration

N055 SET M000 = STA_WIS; Warning current?
N056 JMP (Z001=0) N050;   Timer repeat reset
N057 JMP (IS02=1) N030
N060 JMP (M000=0) N055;   No warning

N070 SET Z000=H000;       Timer reverse
N071 SET H003+1;          Count reverse
N075 JMP (H003>H002) N150;Max count reverse
N080 SET REFFRQ = F001;   Reference reverse
N085 JMP (Z000=0) N053;   Timer reverse
N090 JMP (IS02=1) N030;   Stop required
N095 SET M000 = 0
N100 JMP N085

N150 SET REFFRQ=F003;     Stop
N155 SET OS02=1
N160 JMP (IS03=0)N160;    Waiting for reset
N165 SET OS02=0
N170 JMP N030
```

```
N200 SET ENCTRL=1;        Enable control
N201 SET REFFRQ = F001;   Start reverse
N210 JMP (IS02=1) N030;   Stop required
N220 JMP N210
END                       ;End of program
```

Further typical applications are block-protection controls for mixing plants, mills and shredder.

3.4 Unwinder for wire

The unwinding process, described herein, supplies materials to a dry or wet wire drawing machine. The solution of the sub-automation unit includes the integration of the drive solution in the automatic process and the set-up mode.

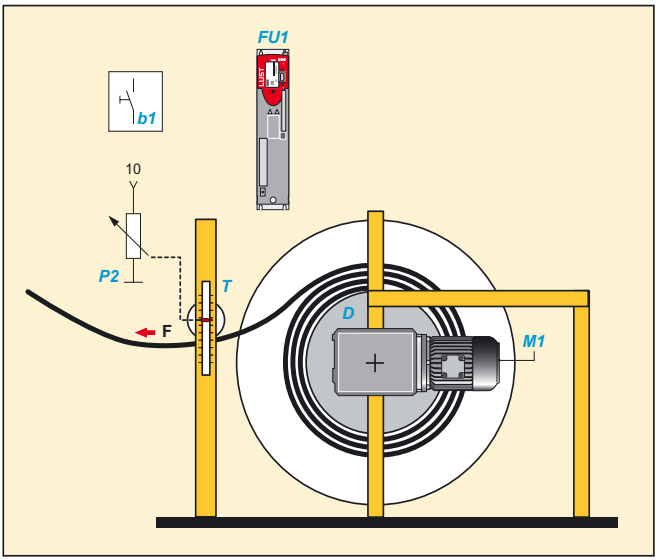
Description of the function

The stripping speed will be controlled via unit FU1, M1 resp. implemented process controller and a feedback via dancer T. For mounting a new wire roller at the mandrel D a set-up mode is implemented in order to ease the work of the operator. The set-up mode will be activated at standstill via button b1 resp. control input IS02 at the frequency inverter FU1.

During set-up mode the drive FU1, M1 runs to a fixed speed and controls the motor current. The motor current is as long under the set limit as the tapped at the shaft drive takes-over the coil. After that the drive FU1, M1 stops immediately, so that the operator can fix the coil at the driving shaft.

Via controlling the dancer position P2 in automatic operation it is possible to detect when the wire will drop-off and the drive control FU1, M1 will be activated automatically.

Technology scheme



picture4.4 Unwinder for wire

Process program

```
; Process program for CDA-PLC, unwinder
; (Winder with dancer control)
;Dancer control via process controller in firmware

%TEXT(Winder)
DEF H001=Threshold value
DEF H002=Waiting time
DEF H000=Analog value
DEF M001=Warning_current
END

%P00
;Initialising
N010 SET H001=10;         Threshold value dancer
N011 SET H002=500;        Waiting time start in ms

;Mainprogram
N020 JMP (IS02=1) N100;   Set-up mode with fixed frequency
N030 SET H000=ISA0;       Control dancer excursion
N035 JMP (H000<H001) N020;Start at excursion of dancer

;Control process
N050 SET ENCTRL=1;        Release control
N060 SET H000=ISA0;       Control dancer excursion
N065 JMP (H000>H001) N060;Stop at final position dancer
N070 SET ENCTRL=0;        Control off
N080 JMP N020
```

```
;Set-up mode
N100 SET ENCTRL=1;        Release control
N110 WAIT H002;           Waiting time start
N115 SET M001=STA_WIS;    Threshold apparent current exceeded?
N120 JMP (IS02=0) N020;   SEt-up mode interrupted
N125 JMP (M001=0) N115;   Detect load surge?
N130 SET ENCTRL=0;        Control off
N140 JMP N020

END
```

Further typical applications are warbler and dancer controls for winders.

3.5 Diameter depending speed control

This PLC-example controls the main drive of a polishing machine. It requires constant circumferential speed depending on the abrasion of the polishing disc.

Description of the function

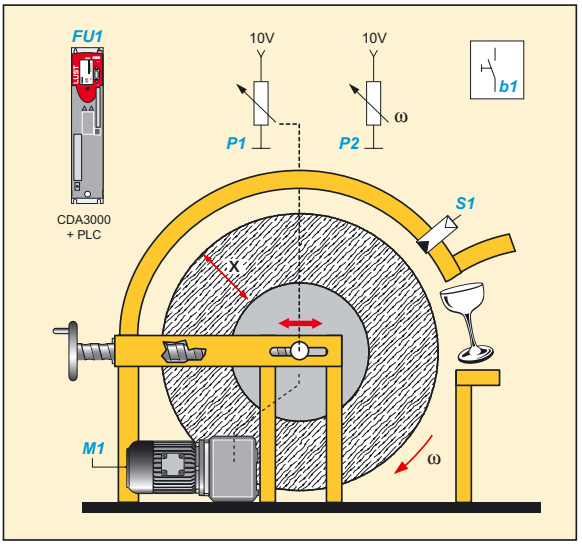
Drive FU1, M1 will be actuated with button b1. Reference of the circumferential speed ω will be transmitted to the process program via potentiometer P2.

The program works directly in application units, that means circumferential speed is unit [m/s], circumference of the disc is unit [m]. The necessary norms are fixed in the process program. In the example the customer-specific set range of 10 m/s-34,5 m/s is the norm.

By abrasion of the disc, the polishing can be misplaced via handwheel, so that an optimal distance to the working place is guaranteed. This mechanical misplacing provides at the same time a 0-10V analog signal for the misplacing range via potentiometer P1. The misplaced position is directly proportional to the diameter of the polishing disc x. Changing the diameter results in calculating the new reference frequency of the main drive M1, from the analog information at requested constant circumferential speed. In this example the diameter of the disc is between 0,55 m and 0,96 m (=> Umfang 1,73 m - 3,01 m).

Initiator S1 regulates the distance between polishing disc and protection hood. If the distance is too small a starting of the main drive is not possible resp. the running process will be interrupted.

Technology scheme



picture4.5 Diameter-depending speed control

Process program

```
; Process program for CDA-PLC polishing disc
; Information:
; Speed reference via ISA0, with the following customer-specific
; settings:
;    0V = 10m/s
;    10V = 34,5m/s   -> Delta = 24,5m/s  -> Resolution 2,45m/s/V
;
; Circumference reference via potentiometer at ISA1, with the
; following customer specific settings:
;   0V -> 0,96m => 3,01m  max. circumference
;   10V -> 0,55m => 1,73m min circumference  -> Delta = 1,28m  ->
;   Resolution 0,128m/V
;
;
%TEXT(Polishing disc)
DEF F000=Analog value0
DEF F003=Analog value1
DEF F002=Reference_m_pro_s
DEF F005=Circumference_m
DEF F007=Reference r_p_m
DEF F009=Reference_in_Hz
END

;F002=ISA0 Reference im m/s, auxiliary variables (F000-001)
;F005=ISA1 Actual value in m, auxiliary variables (F003-004)

%P00
;Reference and actual value regulation
```

```
N010 SET F000=PARA[416]; Call for analog value 0
N015 SET F000*2.45;      Scaling in m/s
N020 SET F001=F000
N021 SET F001+10;        Reference in m/s +Offset 10m/s
N022 SET F002=F001;      Save reference

N030 SET F003=PARA[417]; Call for analog value 1
N035 SET F003*0.128;     Scaling in m
N040 SET F004=3.01;      max. circumference = 3.01m
N041 SET F004-F003;      Actual value circumference in m
N042 SET F005=F004;      Actual circumference in m

;Calculating reference in m/s
N050 SET F006=F002
N055 SET F006*60;        Calculate m/min
N065 SET F006:F005;      Calculate in rpm at the disc
N070 SET F007=F006;      Save reference in F007

;Calculating speed -> frequency
N100 SET F008=F007;      Call for speed
N110 SET F008*2;         Consider gear ratio 1:2
N115 SET F008:20;        Calculate frequency of rotary field:
;                        f=n*pp/60, pp=3
N120 SET F009=F008;      Save Reference in Hz

N150 SET REFFRQ=F009;    Pre-set frequency reference

N250 JMP N010
END                      ;End of program
```

Further typical applications are winding drives with diameter control.

3.6 Simple positioning drive with PLC_3

The example refers to the pre-set solution PLC_3 (position control with PLC-process program). Workpiece positioning via feed drive. Set parameters for the both reference positions via operator panel.

Basic position

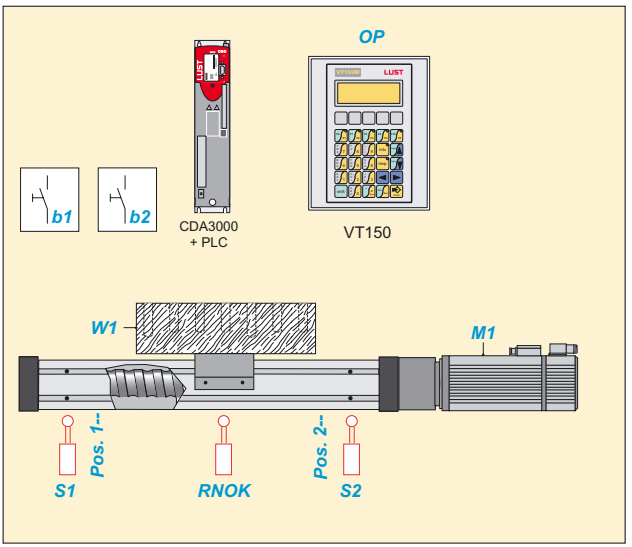
The used CDA3000, PLC has an process control for the position control. Feed drive is in basic position if

- homing mode is released via button b1 and executed. Homing mode gives the absolute position to the feed. Homing mode is not mentioned in this program example. In this program at homing mode the actual axis position will be set as homing position.
- feed carriage reaches no limit switch S1, S2.

Description of the function

If the material is on the carriage as well as reference positions 1 and 2 are set via operator panel OP homing can be start via button b1. After successful homing the carriage runs automatically in the rest position 1. Via button b2 the feed to position 2 can be actuated. Positioning of material via drive M1 resp. moving forward by the set lift length. If the position Pos. 2 is reached and button b2 is resetted carriage moves back to rest position Pos. 1.

Technology scheme



picture4.6 Feed drive with PLC-position control

Process program

```
; PLC_3 - position-control, control via terminal
;
; The position control is started via terminal.
; The position control toggles between 2 position.
; The PLC-program runs if motor control is activ.

;-----------------------------------------------
%TEXT(PLC_3_position_control)
DEF M000 = Homing_OK

DEF H000 = Reference_pos_1
DEF H001 = Reference_pos_2
DEF H002 = Actual_pos
DEF H003 = Zero correction
```

```
END

;-----------------------------------------------
%P00
; Initialisation of reference position
    ; deactivated -> reference positions should not set at every
    ; start of motor control
; N010 SET H000 = 0;
; N020 SET H001 = 655360; 10 U * 65536 Inkr.

; Homing requested?
N030 SET M000 = 0;                Zero position not defined
N040 JMP (IA00 = 0) N060;         ISA0=0: start positioning
                          ;       ISA0=1: start homing
N050 CALL N140            ;       Call sub-routine homing mode

; Main program positioning
N060 JMP (M000 = 0) N040;         Return while homing is not
;                                 finished

    ; Selection of reference position
N070 JMP (IS01 = 1) N110;         Select position 2 if  IS01=1

    ; Select position 1
N080 GO W A H000 ;                Absolut positioning with H000
;                                 and wait while reference is rea
;                                 ched
N090 SET H002 = ACTPOS;           Read actual position
N100 JMP N060             ;       Return

    ; Select position 2
N110 GO W A H001 ;                Absolute positioning with H001
;                                 and wait while reference is rea
;                                 ched
N120 SET H002 = ACTPOS;           Read actual position
N130 JMP N060             ;       Return

;-----------------------------------------------
; Sub-routine homing mode
N140 SET PCTRL = 1;               Switch on position control
N150 GO 0 + H003
N160 SET M000 = 1;                Homing OK (finished suc
;                                 cessfully)
N170 RET

END                               ; End of program
```

Typical applications are simple positioning controls for drives of doors and gates, belts, carriages and material gripper feeds, etc.

4 Commissioning of PLC-system

The following commissionings are described in this chapter:

- First commissioning of CDA3000 with PLC-firmware
- First commissioning of operator panels at the drive controller

The serial commissioning of a frequency inverter with PLC-function is identically with the standard serial commissioning. This form of the commissioning is described in the operating manual of CDA3000.

4.1 First commissioning CDA3000, PLC

For the first commissioning the mentioned devices, assembly groups or knowledges are necessary:

- Frequency inverter CDA3000 with standard software V3.50-01 or higher
- CD-ROM for DRIVEMANAGER and PLC-editor
- Serial interface cable CCD-SUB90x
- Experience with the handling of inverter drive system CDA3000 and DRIVEMANAGER

Strategy:

1. Installation of software DRIVEMANAGER V3.20 (or higher)

   For operating the PLC-editor a DRIVEMANAGER-Version from V3.20 is necessary.

   Please find information to the installation of the software in the documentation of the DRIVEMANAGER.

2. Installation of PLC-editor

   Please take care that the DRIVEMANAGER is closed before installation of the PLC-editor. The PLC-editor is an additional module for the DRIVEMANAGER, which has to be installed additionally.

3. Connect CDA3000 via serial interface (RS232) to the PC
4. Starting of the software DRIVEMANAGER

   This process searches automatically for connected inverters and description of the device will be read into.

5. Installation of PLC-software

   The installation of the firmware is only necessary, if there is not yet a PLC-software version V700.xx available at the device. Please find information to the actual firmware of the connected device in the DRIVEMANAGER main window in menu „actual values“.

Note: A motoridentification is not included in the PLC-firmware. On demand it can be executed with the help of a standard software (e.g. V3.50-01, at PLC-editor-CD-ROM in directory "Firmware\Std-Firmware"). After that the identified and at the PC stored dataset can be loaded on a device with PLC-firmware

The PLC-software and the DRIVEMANAGER-text files can be find on the PLC-editor-CD-ROM in sub-directory "Firmware\PLC-Firmware".

At DRIVEMANAGER-version smaller than V3.20-00 (z.B. V3.10 or V3.20-98) copy the DRIVEMANAGER-text files in the DRIVEMANAGER-file (C:\program files\Lust Antriebstechnik GmbH):
- 3_700_*.pit to „Lust DriveManager\language\001\“
- 3000_ERR.txt to „Lust DriveManager\language\001\“
- 3_700_*.mcw to „Lust DriveManager\firmdata\“

Install the PLC-software on the frequency inverter via menu „Extras/ Load device software“.

An succesful „Download“ effects in an automatic reset of the frequency inverter as well as a read into of the device description.

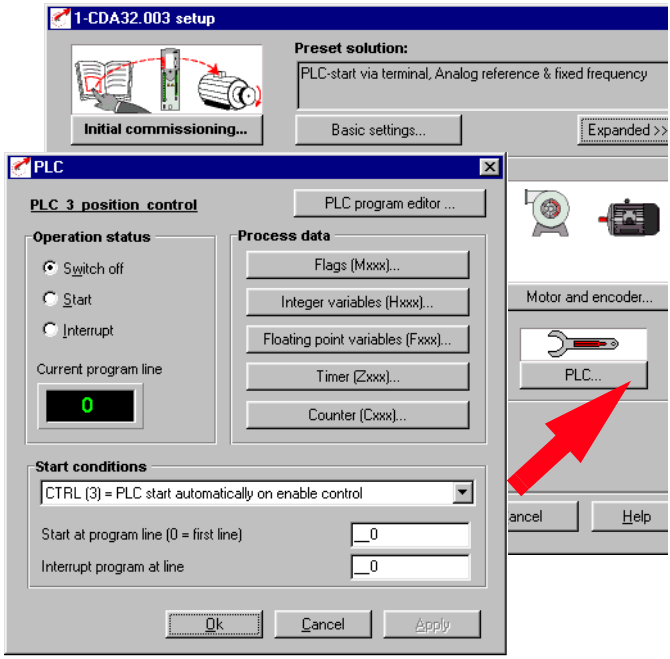
6. First commissioning

The PLC-software includes no motoridentification. If an exact setting of the motor data is necessary (e.g. at control modes SFC or FOR) start the first commissioning with loading a data set with corresponding motor adjustments.

Further steps of the first commissioning of the drive controller will be executed as described in the manual of CDA3000.

7. Programming of PLC-functions

Select „PLC...“ in the main windows of the DRIVEMANAGER. With this function mask all functions and terms of the PLC can be programmed and tested.



picture 4.7 PLC-main window

8. PLC-program-editor

With button "PLC-program-editor..." you are able to write a PLC-program or to load one in the device. Functions and command syntax can be find in the online help or in this system manual.

4.2 First commissioning of operator panels

This chapter describes basically the configuration of the communication interface between operator panel and drive. The application specific programming of each operator panel is describee in the operator panel software manual (included on VTWINCD CD-ROM).

Condition for the programming of operator panels is an installed version of the VTWIN software on your PC.

4.2.1 Configuration of RS232-interface

1. Installation of RS232-driver for LustBus-protocol

If the VTWIN software does not include the Lust-specific software driver for communication via serial RS232-interface (e. G. V4.50 or V4.64), follow the stated steps:

- Close VTWIN application
- Extract the driver package *.zip. At VTWIN-Version V4.64 the driver is stored on the CD-ROM in the directory „Lust driver“.
- Replace VTWINDEV.DBR in VTWIN-installation directory (e.g. B. C.\Program files\ESA\VTWIN)
- Copy LUSTBUS.BIN in sub-directory firmware

2. Start VTWIN

3. Select a new project.

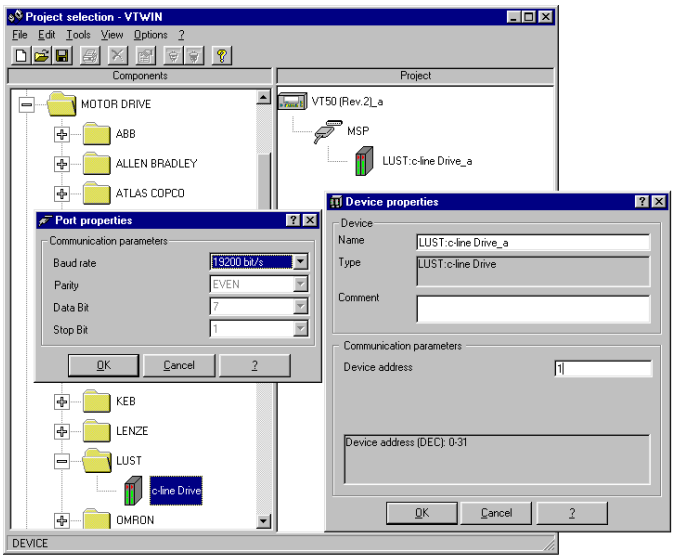
At first select type of operator panel, shall be programmed:

Type	Name VTWIN
VT050 00000N	VT50 (Rev.2)
VT150W 00000N	VT150W
VT505W 00000N	VT505W
VT155W 00000N	VT155W or VT155W (Porträt)

Selected operator panel and bus-interface will be indicated in project window.

4. Select a drive controller

Open directory „MOTOR DRIVE/LUST“ in component window. Draw the icon of drive controller „c-line Drive“ to the interface-icon (MSP or ASP) in the project window. If Lust directory is missing please install the driver as described in 1.). Now the drive is physically connected with the operator panel.



picture 4.8 Configuration of operator panel with RS232-interface

5. Setting of RS232-interface Baud rate

Open via double-click on the interface icon (MSP or ASP) the window „port properties“ and set the Baud rate to 19200 bit/s.
This Baud rate must be set in the drive, too. Therefore start the DRIVEMANAGER and open in menu „communication“ the bus-configuration. Save the settings in the drive.

6. Setting of device name and address

Open via double-click on the drive icon the window „device properties“. The device address must (always) be set to 1. Furthermore the symbolic name of the drive controller can be changed.

7. Programming of operator panels

Open via double-click on operator panel icon in project window the program application. For programming use the software manual of the operator panels (available on VTWIN CD-ROM).

4.2.2 Configuration of CAN_{open}-network

Conditions for bus-operation:

- Drive is equiped with CAN-module (CAN_{open}-protocol)
- Drive is connected with the operator panel via CAN-Bus
- Parameters of the fieldbus interface are set correctly in the connected drive (e.g. Bus-protocol, Baud rate, device addresses)

1. Start VTWIN

2. Select a new project.

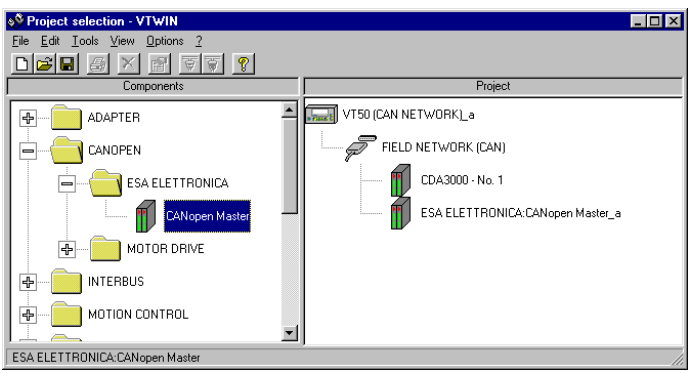
At first select type of operator panel, shall be programmed:

Type	Name VTWIN
VT050 000CNN	VT50 (CAN NETWORK)
VT150W 000CNN	VT150W (CAN NETWORK)
VT505W 000CNN	VT505W CAN
VT155W 000CNN	VT155W CAN or VT155W (Porträt) CAN

Selected operator panel and bus-interface will be indicated in project window.

3. Select drive

Open directory „CANOPEN/ESA ELLETRONICA“ in component window. Draw icon „CANopen Master“ to the interface-icon FIELD NETWORK (CAN)) in the project window. The drive controller is now physically connected to the operator panel. It is possible to install up to 127 drives of c-line to the bus. .



picture 4.9 Configuration of operator panels with CAN_{open}-network

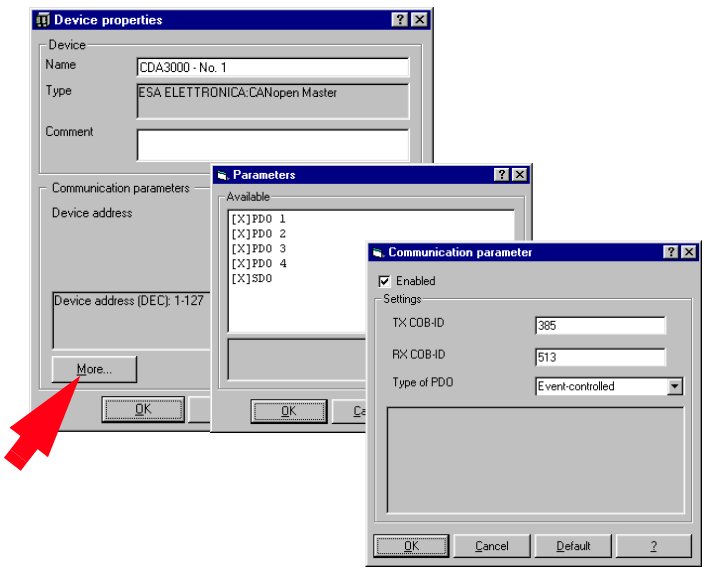
4. Setting the Baud rate of CAN_{open}-network

Open via double-click on the icon interface (FIELD NETWORK (CAN)) the window „port properties“ and set the Baud rate to the requested speed.
Please note that the adjustment of the baudrate has to be set in the drive as well.

5. Setting of device name, address and transfer parameters

Open via double-click on the icon drive controller the window „device properties“. Set the device address. Furthermore the symbolic name of the drive can be changed here.
Please note that the device address has to be set in each drive as well.

Via actuating the button „More ...“ in window „device properties“ the window opens for setting the transfer parameters resp. channels. For setting the parameters and process data channels use the user manual „communication module CAN_{open}.



picture 4.10 Configuration of drive for CAN_{open}*-network*

6. Programming the operator panels

Open via double-click on the operator panel icon in project window the program application. For programming use software manual to the operator panels (available on VTWIN-CD-ROM).

5 Order data of system components

Order codes for PLC-software package

Order designation	Short description	Article no.
PLC-Editor	CD-ROM incl. - PLC-Program editor - PLC-Training program - System manual CDA3000-PLC - Firmware CDA3000-PLC - Licence PLC-Editor	0842.V11.1
DRIVEMANAGER V3.20	DriveManager full version on CD-ROM	0842.V08.2
DRIVEMANAGER V3.20, TEST	DriveManager-180-days-testversion on CD-ROM	0842.V09.2

Order codes for user and communication modules

Order designation	Short description	Article no.
CM-CAN1	Communication module for CAN_{open}	0916.0001
CM-DPV1	Communication module for Profibus DP-V1 Supports CDA3000-PLC	0916.0003.1 0916.0000.1 (with-GSD)

Order codes for manual operating unit

Order designation	Short description	Article no.
KP200-XL	KeyPad KP200 for parameter setting and serial commissioning of CDA3000-PLC with SmartCard SC-XL	
SC-XL	SmartCard with large EEPROM-memory for saving PLC-Data	

Order codes for Operator Panel and accessories

Bestell-bezeichnung	Kurzbeschreibung	Artikelnummer
VT050 00000N	operator panel VT50 (RS232)	1002.0001.0
VT050 000CNN	operator panel VT50 (CAN_{open})	1002.0002.0
VT150W 00000N	operator panel VT150 (RS232)	1002.0003.0
VT150W 000CNN	operator panel VT150 (CAN_{open})	1002.0004.0
VT505W 00000N	operator panel VT505 (RS232)	1002.0005.0
VT505W000CNN	operator panel VT505 (CAN_{open})	1002.0006.0
VT155W 00000N	operator panel VT155 (RS232)	1002.0007.0
VT155W000CNN	operator panel VT155 (CAN_{open})	1002.0008.0
VTWINCD	Software package VTWIN to program the operator panels - CD-ROM (Software, Manuals, 5 languages)	1002.0011.0
CVCOM 11102	Programming cable PC 9-polig-MSP 25-poles necessary for all operator panels	1002.0009
CVCOM 25F8M	Programming cable adapter PC 25-poles-ASP 8-poles DIN Necessary for - VT050 000CNN - VT150W 000CNN - VT505W 000CNN - VT155W 00000N - VT155W 000CNN CVCOM 11102 must be available!	1002.0010
OPK-RS03	Serial RS232-interface cable to connect the drive controller to the operator panel. Suitable for - VT050 00000N - VT150W 00000N - VT505W 00000N - VT155W 00000N Length 3 m	in preparation
OPK-RS05	Serial RS232-interface cable to connect the drive controller to the operator panel. Suitable for - VT050 00000N - VT150W 00000N - VT505W 00000N - VT155W 00000N Länge 5 m	in preparation

Order codes for trainings and services

Order designation	Explanation	Net price
PLC-Projectmanagement	8 hours incl. post-processing without travel costs	on request
PLC-Engineering	8 hours incl. post-processing without travel costs	on request
PLC-Software producing	8 hours incl. post-processing without travel costs	on request
PLC-Commissioning	8 hours incl. post-processing without travel costs	on request
PLC-Training	1 day incl. records, snacks	on request
PLC-Training, Licence	1 day incl. records, snacks PLC-Editor-Licence	on request
OP-Training	1,5 dayse incl. records, snacks	on request
OP-Training, Licence	1,5 dayse incl. records, snacks, VTWINKIT with licence	on request

LUST
ANTRIEBSTECHNIK

Lust Antriebstechnik GmbH
Gewerbestrasse 5-9 • D-35633 Lahnau
Tel. 0 64 41 / 9 66-0 • Fax 0 64 41 / 9 66-137
Internet: http://www.lust-tec.de • e-mail: info@lust-tec.de

ID no.: 0840.12B.1 • 05/2003