

## **Control Techniques Unidrive**

This manual presents installation and handling of the driver Unidrive to the terminals in the E-series.

The functionality in the E-terminals and in MAC Programmer+ are described in the E-manual.

© G & L Beijer Electronics AB 2000, MA00402A, 2000-02

G & L Beijer Electronics AB reserves the right to change information in this manual without prior notice. All examples in this manual are used solely to promote understanding of how the program works and its operation. G & L Beijer Electronics AB take no responsibility if these examples are used in real applications.

# Content

<b>1 Introduction</b> .....	3
<b>2 Install and update driver</b> .....	4
2.1 Installation of driver using Internet.....	4
2.2 Installation of driver from disk.....	4
<b>3 Connecting the terminal to the Unidrive inverter</b> .....	5
3.1 Settings in the MAC Programmer+ .....	5
3.2 Settings in the Unidrive inverter .....	9
3.3 Connecting the terminal to the Unidrive inverter .....	9
<b>4 Addressing</b> .....	11
4.1 Menu parameters .....	11
4.2 Internal registers .....	11
<b>5 Efficient communication</b> .....	12
5.1 Signals affecting the communication time.....	12
5.2 How to make the communication more efficient .....	13
<b>6 Drawings</b> .....	14

# **1 Introduction**

This manual describes how the Undrive inverter is connected to the terminals in the E-series. For information about the Undrive inverter we refer to the manual for current inverter.

The terminals support the inverters in the Unidrive series with serial interface i. e. UD 71.

## 2 Install and update driver

When installing MAC Programmer+ the drivers available at the time of release are installed too. A new driver can be added into MAC Programmer+ either with MAC Programmer+ using an Internet connection or from diskette. A driver can be updated to a newer version in the same ways.

### 2.1 Installation of driver using Internet

To update available drivers to the latest version or to install new drivers you can use the function Update terminal drivers, from Internet in the File menu in MAC Programmer+. All projects must be closed before this function is used and the computer must be able to make an Internet connection. You don't need a browser. When the connection is established a list is shown with all drivers that can be downloaded from Internet to the computer. The list shows the version number of available drivers and the version number of installed drivers. Mark the driver/ drivers you want to install in the MAC Programmer+. The function Mark Newer will mark all drivers that are available in a newer version then the one installed and the drivers not installed. Then you select Download. Each drivers is approximately 500 kb and it is ready to use when the download is ready.

### 2.2 Installation of driver from disk

To update available drivers to the latest version or to install new drivers you can use the function Update terminal drivers, from Disk in the File menu in MAC Programmer+. All projects must be closed before this function is used. Select the folder with the new driver and choose to open the mpd-file. A list is shown with all drivers that can be installed showing the version number of available drivers and the version number of installed drivers. Mark the driver/ drivers you want to install in the MAC Programmer+. The function Mark Newer will mark all drivers that are available in a newer version then the one installed and the drivers not installed. Then you select Install.

How to select the Unidrive driver in the project and how to transfer it to the terminal are described in *chapter 3*.

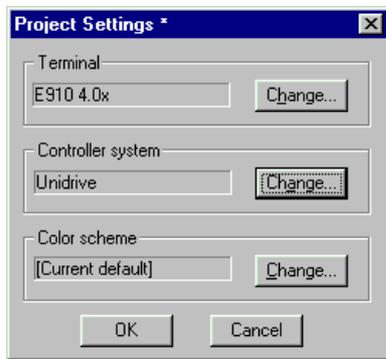
## 3 Connecting the terminal to the Unidrive inverter

### 3.1 Settings in the MAC Programmer+

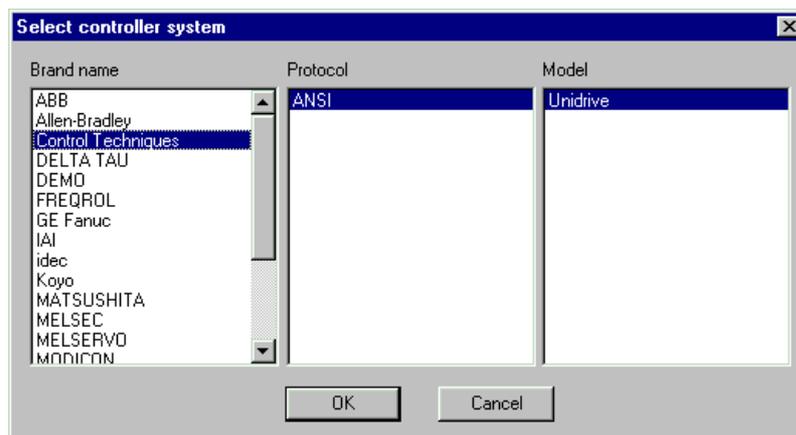
For communication with Unidrive inverter the following settings must be made in the programming tool MAC Programmer+.

#### Driver selection

Choosing **New** in the **File** menu creates a new project and the dialog **Project Settings** is shown. In an existing project, the dialog is shown by selecting **Project Settings** in the **File** menu.

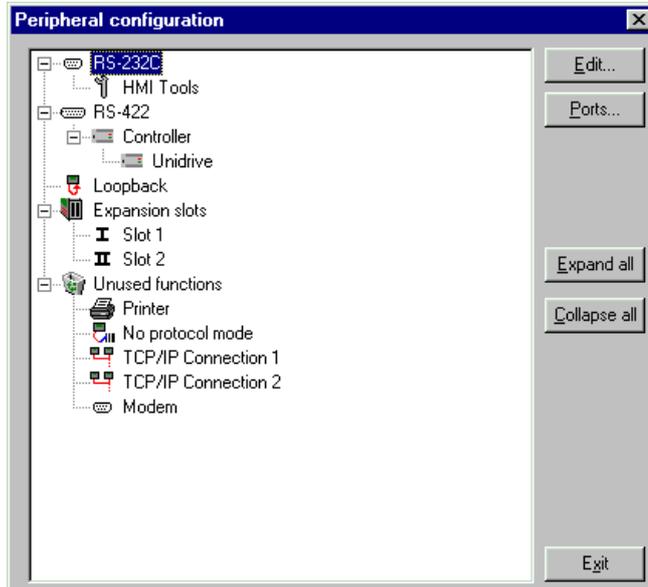


Press **Change...** under **Controller system** to get the choice list of available drivers. Choose Brand name, Protocol and Model and then press **OK**. Press **OK** again to confirm the project settings.



## Communication setup

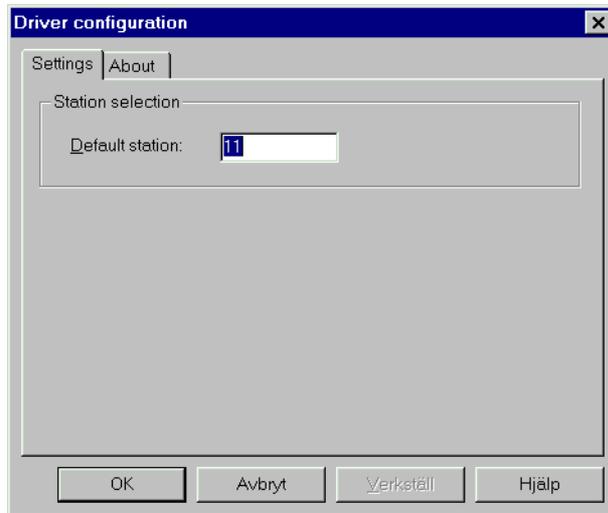
The settings for the communication between the terminal and the inverter are done under **Peripherals** in the **Setup** menu. To change which port the inverter is connected to, mark Controller and hold left mouse button down and drag to move it to another communication port. Mark the selected communication port and press **Edit** to change the other communication settings.



The settings should be:

Parameter	Description
Port	RS-422
Baudrate	4800 – 19200
Data bits	7
Stop bits	1
Parity	even

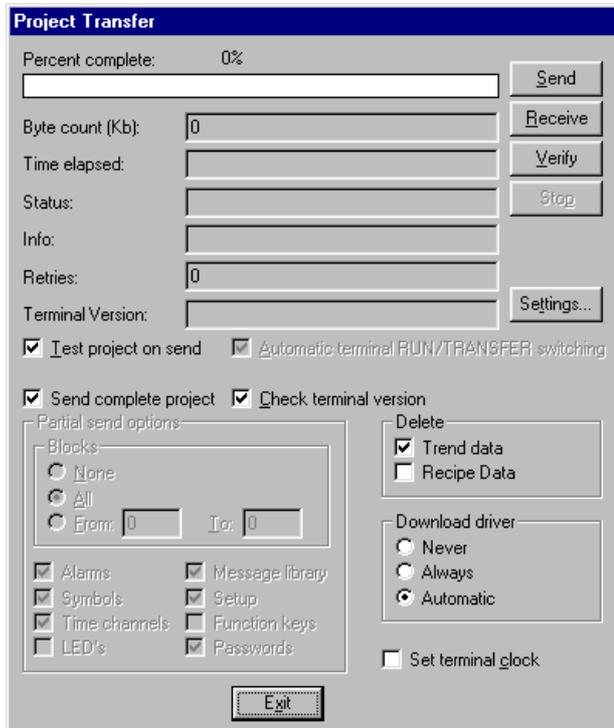
To make specific settings for the selected driver mark the driver name and press Edit.



Parameter	Description
Default station	11 – 99. Do not use 20, 30, 40, 50, 60, 70, 80, or 90.

## Transfer the driver to the terminal

The selected driver is down-loaded into the terminal when the project is transferred to the terminal. Choose Project in the Transfer menu.



There are three alternatives when the driver is down-loaded into the terminal.

Function	Description
Never	The driver is not down-loaded . The existing driver in the terminal is used.
Always	The driver is down-loaded every time the project is transferred.
Automatic	The driver is down-loaded if the driver in the terminal is not the same as the selected driver in the project. If it is the same the driver is not down-loaded.

## 3.2 Settings in the Unidrive inverter

For information about settings in the inverter we refer to the manuals for current inverter.

Parameter	Description	Default
11.23	Serial address. Corresponds to the default station number in the terminal	1.1
11.25	Serial baudrate	4800
11.26	Two wire mode 0 = 4-wire mode 1 = 2-wire mode	0

---

**Note!**

With multidrop network all inverters require unique serial addresses.

---



---

**Note!**

Above parameters are valid for the UD 71 module. For other modules with serial communication interface please refer to separate manuals!

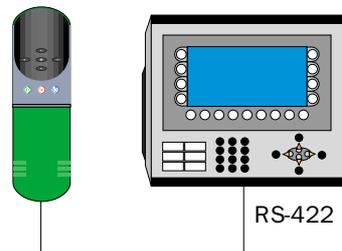
---

## 3.3 Connecting the terminal to the Unidrive inverter

The connection to the terminal can either be of the type “point to point” or “multidrop” . A serial communication interface must be used to the inverter, i.e . UD-71.

### Point to point

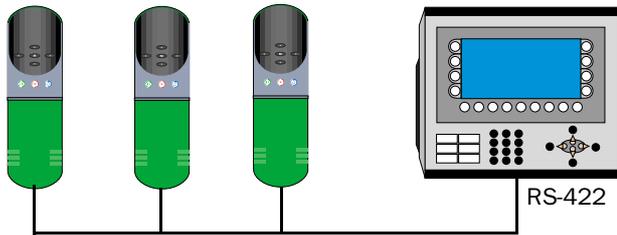
#### 4-wire mode



Cable according to drawing K-03332 in the chapter Drawings.

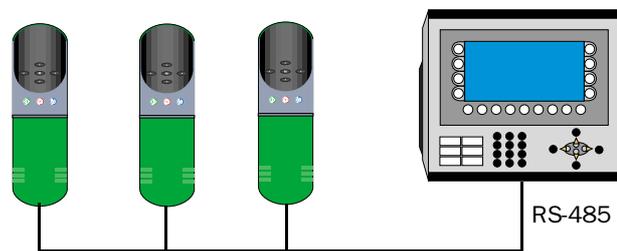
## Multidrop

### 4-wire mode



Cable according to drawing K-03331 in the chapter Drawings.

### 2-wire mode



Cable according to drawing K-03330 in the chapter Drawings.

## 4 Addressing

### 4.1 Menu parameters

The terminal can handle the following parameters according to the notation Menu.parameter.

*Example*

Addressing parameter 19 in menu 0 is done by writing 0.19.

The menus 0 to 99 can be used in the terminal.

When network addressing is used the station number is added to the address according to the notation Station number: Menu.Parameter

The default station stated in Driver configuration is used

*Example*

Addressing parameter 8 in menu 2 in the node with Station number 11 is done by writing 11:2.08.

Addressing a single bit in a 16-bit parameter is done according to the notation Menu.Parameter.Bit where Bit could be 0 – 15.

*Example*

Addressing bit 0 in parameter 10 in menu 20 is done by writing 20.10.0.

---

**Note!**

If the parameter is not available in the system or the wrong number of decimals is defined for the object an error message is shown on the terminal display as Write ERR/Read ERR XX:YY:ZZ where XX is the station number, YY the menu and ZZ the parameter in which the error occurs.

---

### 4.2 Internal registers

256 internal 16-bit registers are defined in the driver internally. These are named I1 to I256 and they are reset each time the terminal is restarted.

## **5 Efficient communication**

To make the communication between the terminal and the controller quick and efficient the following should be noted about how the signals are read and what that can be done to optimize the reading.

### **5.1 Signals affecting the communication time**

It is only signals to objects in the current block that are read continuously. Signals to objects in other blocks are not read, that is the number of blocks does not affect the communication time.

Besides the signals to objects in the current block, the terminal is continuously reading the following signals from the controller:

- Display signals
- Block print-out signals
- LED registers
- Alarm signals
- Remote acknowledge signals on alarms and alarm groups
- Login signal
- Logout signal
- Trend registers at the sample points
- Bargraph registers if using min/max indicators
- New display register
- Buzzer register
- Backlight signal
- Cursor control block
- Recipe control block
- Library index register
- Index registers
- Controller clock register if the controller clock is used in the terminal
- List erase signal
- No protocol control register
- No protocol on signal

### **Signals not affecting the communication time**

The following signals do not affect the communication time:

- Signals linked to function keys
- Time channels
- Objects in the alarm messages

## 5.2 How to make the communication more efficient

### Efficient block changes

Block changes are carried out most rapidly and efficiently through the block jump function on the function keys or through a jump object. "Display signals" in the block header should only be used when the controller is to force the presentation of another block. The "New Display" register can also be used if the controller is to change the block. This does not affect communication as much as a larger number of "Display signals".

### Use the clock of the terminal

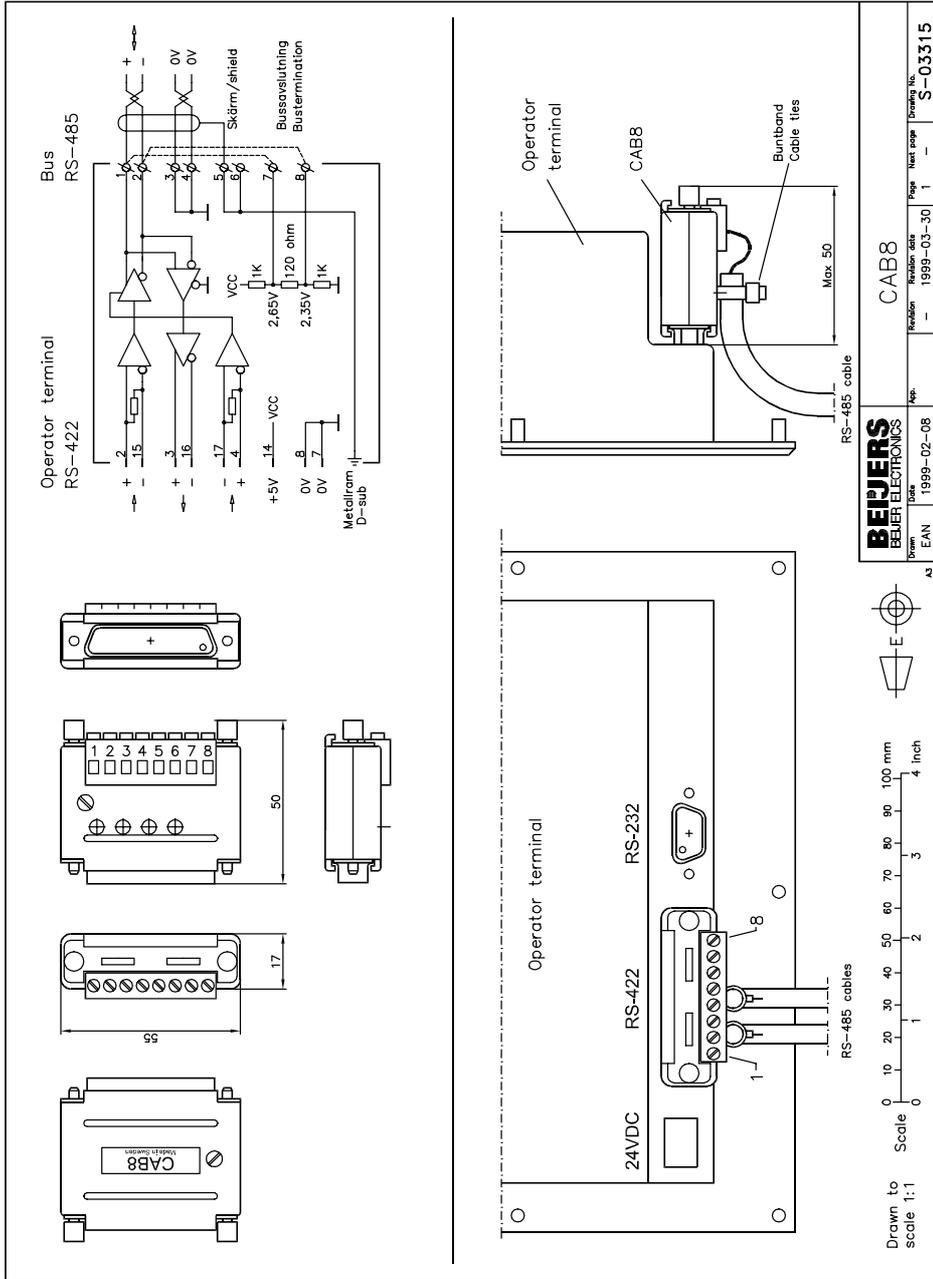
An extra load is put on communication if the clock of the controller is used since the clock register must be read up to the terminal. Downloading of the clock to the controller also creates an extra load. The interval between downloadings should therefore be as long as possible.

### Use bit addressing in alarms

By using bit addressing and putting the addresses in consecutive order, alarm handling could be made up to 16 times faster. See 4.1.



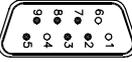






**RS-232**  
**MAC 50/90/Exxx, MTA-250/GI/Exxx**  
**MAC 10/CM, CMI0**

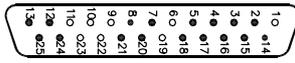
Pin no	Name	Signal direction MAC ←→ XXX
1	+TXD	→
2	-TXD	→
3	RxD	←
4	0V	
5	CTS	←
6	RTS	→
7	RTS	→
8	+5V <5mA	→
9 *		



\* E1 i E-serien  
 Not in E-series

**RS-422**  
**MAC 40+/Exxx, MTA-100/Exxx**  
**MAC 10/CM, CMI0**

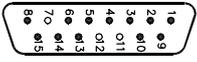
Pin no	Name	Signal direction MAC ←→ XXX
1	+TXD	→
2	-TXD	→
3	RxD	←
4	0V	
5	CTS	←
6	RTS	→
7	RTS	→
8	+5V	→
9		
10	+CTS	←
11	-CTS	←
12		
13		
14	+5V	→
15	<50mA	→
16	>200mA	→
17		
18		
19		
20		
21		
22		
23		
24		
25		



- D-sub  
 25-pin Home  
 25-pin Female
- 1) Sift 20 är anslutet till siff 21. Internt i MAC'en.  
 Pin no 20 connected to pin no 21 Internal in MAC/MTA.
  - 2) Endast i E-serien och med serie nr 9901 eller senare  
 Only for E-series and with serial no 9901 or later
  - 3) Endast E100/MAC40+ /MTA-100  
 Only for E100/MAC40+ /MTA-100

**RS-422**  
**MAC 50/90, MTA-250/GI**

Pin no	Name	Signal direction MAC ←→ XXX
1	+TXD	→
2	-TXD	→
3	RxD	←
4	0V	
5	CTS	←
6	RTS	→
7	RTS	→
8	+5V	→
9		
10	+CTS	←
11	-CTS	←
12		
13		
14		
15		



D-sub  
 15-pin Home  
 15-pin Female

Bild behandlad i 2500gr A3

<b>BEIJERS</b>		<b>MAC/MTA RS-232/RS-422</b>	
BEIJER ELECTRONICS		1993-09-07	
Order no	1993-09-07	Revision	1993-12-01
Page	1	Page	1
Product no	S-00724		