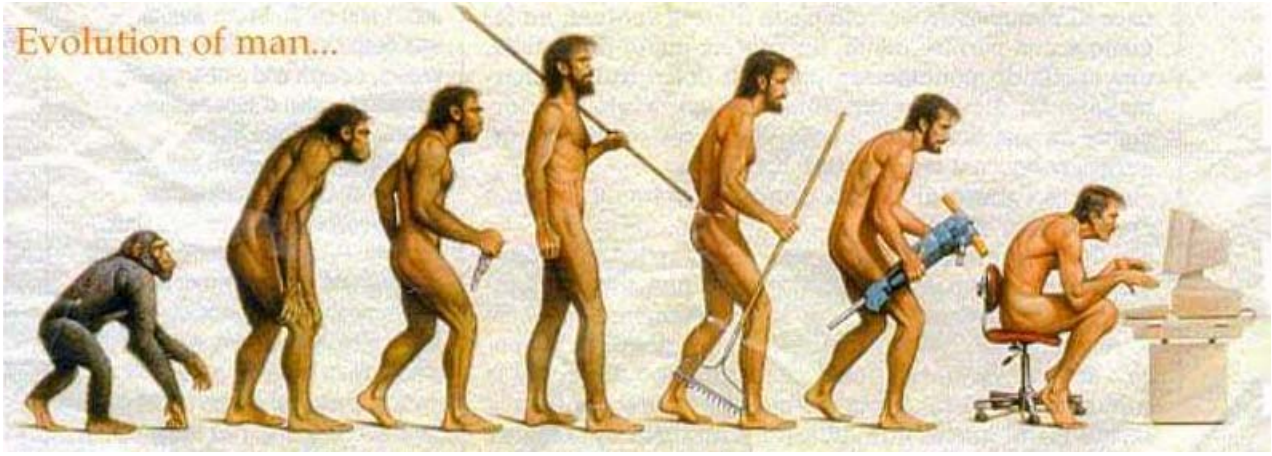
# Intelligent Terminals PCD7.D7xx

For PCD8.D81W starting from Version 4.10




## Table of content

Intelligent Terminals PCD7.D7xx ..... 1
Table of content ..... 1
1) Introduction ..... 2
2) Hardware overview ..... 3
3) Network topology ..... 4
4) Programming and connection cables ..... 7
5) Create the first application ..... 13
6) FAQ: Frequently asked questions ..... 30
7) Open an existing .mdb project ..... 32
8) Define the structure of an application ..... 33
8.1) Terminals with function keys ..... 33
8.2 Touchscreen Terminals ..... 34
9) Exchange areas ..... 35
9.1) Alarms and information messages ..... 38
9.2) Status area ..... 40
9.3) Command area ..... 41
10) Data storage and backup ..... 42
11) FBoxes for PG4 and PG5 ..... 43

# 1) Introduction

The new generation of intelligent industrial terminals offers the following advantages over the former PCD7.D8xx terminals:

- easy to use programming tool for Windows ’95 , ’98 et NT
- touchscreens available
- Profibus-DP slave connection possible (but not recommended)
- at least 4 project languages available
- random sequences and not only start/stop sequences for terminals with function keys
- individual password protection for each field or key with 10 levels of protection
- alarms can be collected from several PLC‘s
- drivers for all common PLC manufacturers
- gateway functionality possible with terminals having two serial ports (for example Siemens S7 <> SAIA PCD)
- access as well to inputs, outputs, flags/merker
- key presses can activate macros

The „old“ PCD7.D8xx terminals will still be available for a couple of years, but for new applications the new range should be used. Projects created with the DOS tool unfortunately can’t be transferred to the windows tool, so the dialogue has to be recreated from scratch.

# 2) Hardware overview

| PCD7. | ESA-designation | Function keys | Text (Charakter) | Graphics (Pixels) | Colors | Languages |
|---|---|---|---|---|---|---|
| **D790** | VT050W00000 | 5 | 2 x 20 | - | 1 | 4 |
| **D795** | VT060W00000 | 4 | 4 x 20 | - | 1 | 4 |
| **D700**<br>**D706**<br>**D709** | VT150W00000<br>VT150W000DP<br>VT100MT1000 | 5 (+20, ***) | 4 x 20 | - | 1 | 6 |
| **D710** | VT170WA0000 | 12 | 4 x 20 | - | 1 | 8 |
| **D720**<br>**D725** | VT190WA0000<br>VT190WAP000 | 24 | 4 x 40 | - | 1 | 8 |
| **D730**<br>**D735** | VT300WA0000<br>VT300WAP000 | 24 | 8 x 40,<br>4 x 20,<br>2 x 10 | 240 x 64 | 1 | 8 |
| **D740**<br>**D745** | VT310WA0000<br>VT310WAP000 | 21 | 16 x 40,<br>8 x 20,<br>4 x 10 | 240 x 128 | 1 | 8 |
| **D750**<br>**D755** | VT320WA0000<br>VT320WAP000 | 28 | 16 x 40,<br>8 x 20,<br>4 x 10 | 320 x 240 | 16 | 8 |
| **D760**<br>**D765**<br>**D766** | VT555WA000<br>VT555WAP00<br>VT555WAP0DP | Touchscreen | 16 x 40,<br>8 x 20,<br>4 x 10 | 240 x 128 | 1 | 8 |
| **D770**<br>**D775** | VT565WA0000<br>VT565WAP000 | Touchscreen | 16 x 40,<br>8 x 20,<br>4 x 10 | 320 x 240 | 16 | 8 |
| **D785**<br>**D788** | VT585WAPS00<br>VT585WAPT00 | Touchscreen | 30 x 80,<br>16 x 32,<br>32 x 64 | 640 x 480 | 16 | 8 |

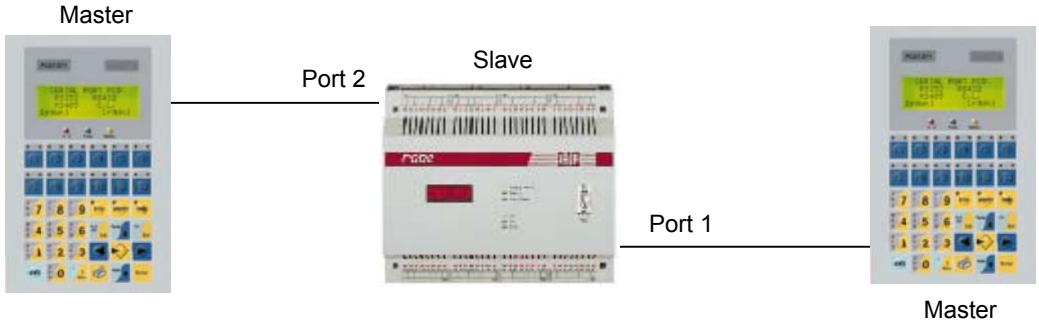
*** PCD7.D70X extendable with a PCD7.D709 for 20 extra function keys.

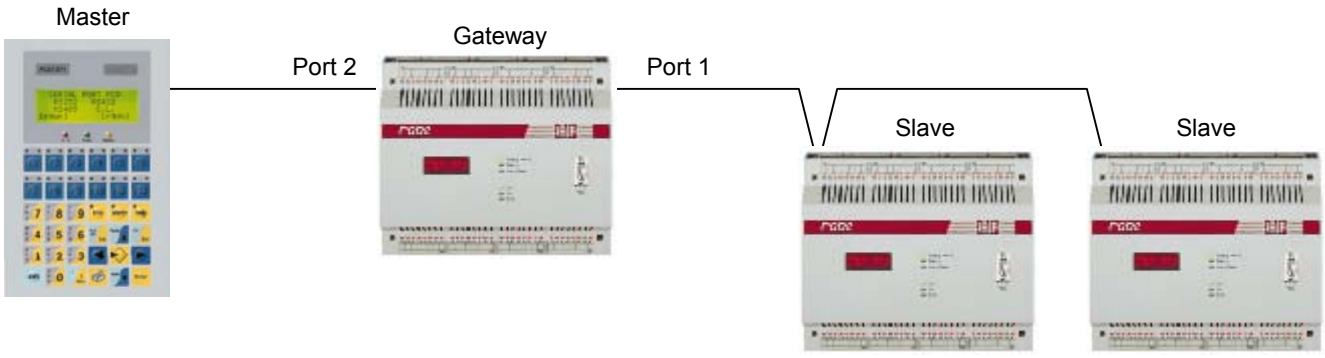
# 3) Network topology

Mode D (RS 232, RS 485, RS 422, TTY 20 mA) point to point connection


S-Bus (RS 232, RS 485, RS 422, TTY 20 mA)

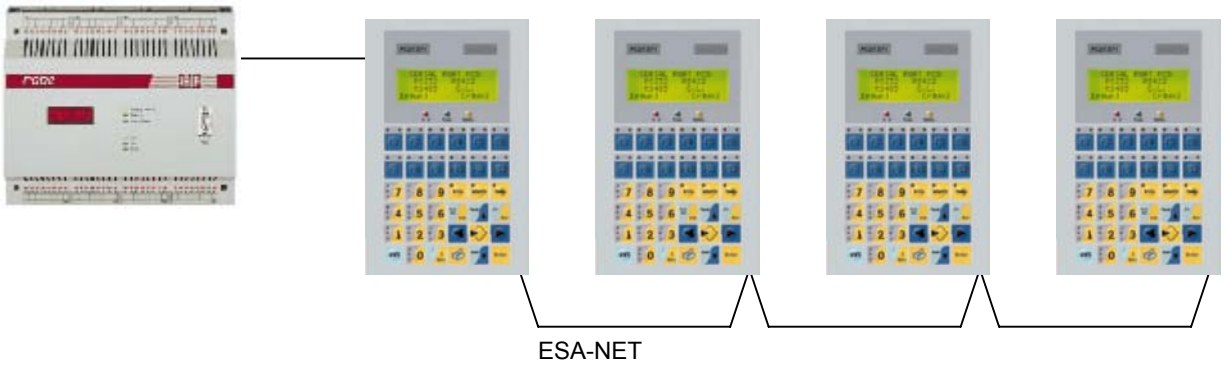






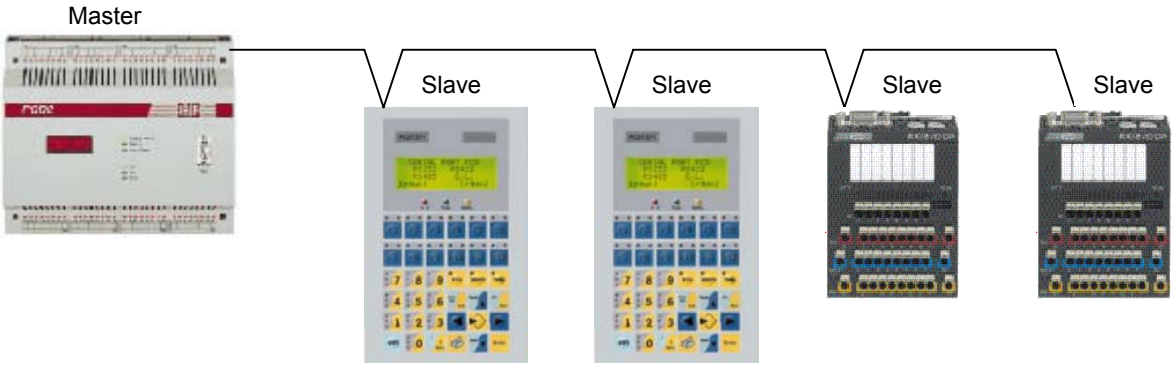

MPI (xx7, RS 485)

ESA-NET with Mode D, S-Bus or MPI




Remark: The ESA-NET server terminal (connected to the PLC) must have two ports. A good choice are the PCD7.D760 touchscreen terminals for ESA-NET networks as they can be client and server and are not very expensive.

Profibus-DP

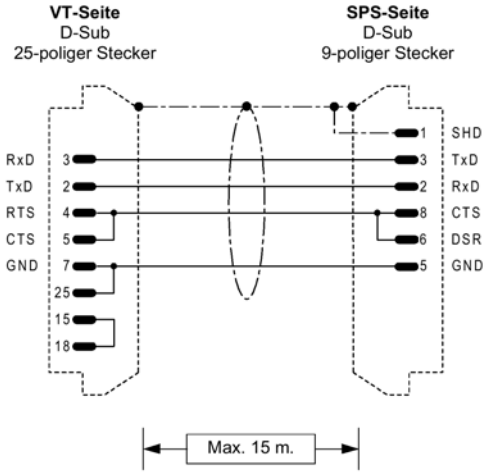



The PCD contains a driver per slave terminal which leads to long cycle times which means long reaction times on the whole DP network (up to 16 ms per terminal). Therefore the use of terminals as DP slaves is not recommended.

# 4) Programming and connection cables

The programming cable contained in the PCD8.D81W progamming package can be used to program the whole range of PCD7.D7xx terminals.

Choose an S-Bus connection for all new applications since this is more reliable.



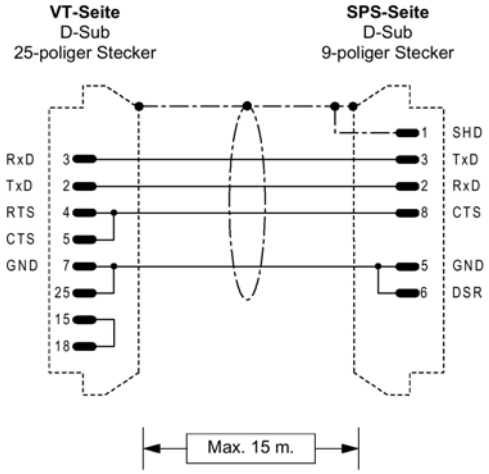

**RS 232 on PGU Port (Mode D), not recommended**

| PCD1 | PCD2 | PCD4 | PCD6 | xx7 |
|---|---|---|---|---|
| - | ✓ (1) | ✓ | ✓ | - |

(1): not suitable for PCD2.M150 and PCDx.M170

(without assignation of the PGU port, configuration of port 0 as S-Bus PGU port)

Selected PLC type: PCD




**RS 232 auf PGU Port (SBus)**

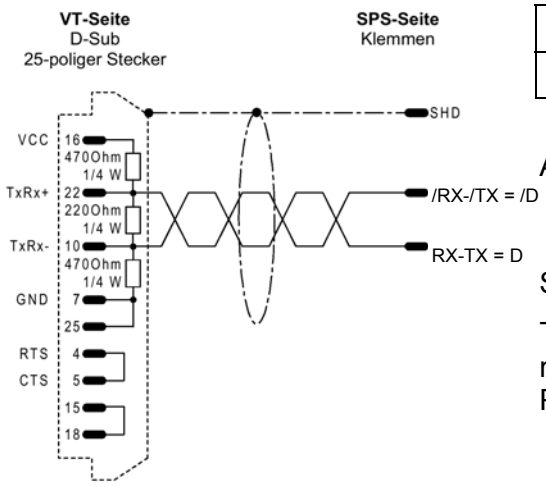
| PCD1 | PCD2 | PCD4 | PCD6 | xx7 |
|---|---|---|---|---|
| ✓ | ✓ | ✓ | ✓ | - |

(without assignation of the PGU port, configuration of port 0 as S-Bus PGU port)

Selected PLC type: S-Bus

**RS 485 (S-Bus)**

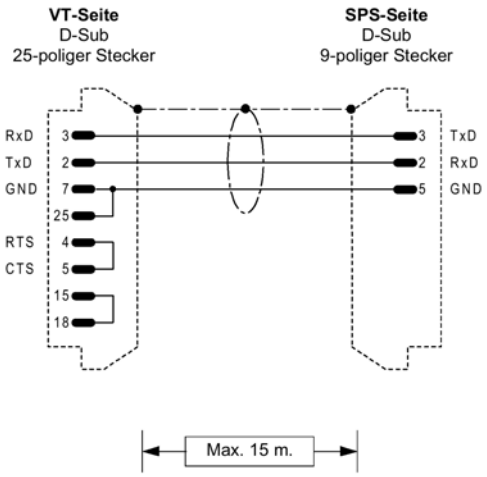
| PCD1 | PCD2 | PCD4 | PCD6 | xx7 |
|---|---|---|---|---|
| ✓ | ✓ | ✓ | ✓ | - |




Assignation of the serial port: SASI S-Bus with the relevant port number, parity mode as slave, other parameters default values.

Selected PLC type: S-Bus

Termination resistances only if terminal station in the S-Bus network. It is better to use 330 / 150 / 330 Ohm as in the PCD.



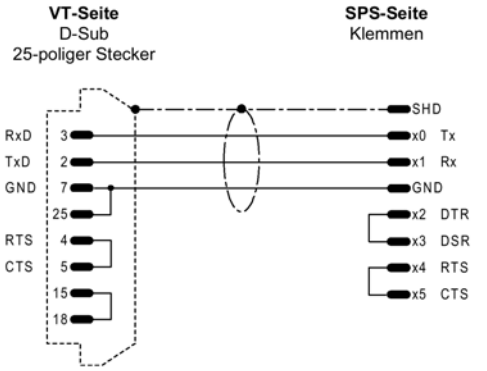

**RS 232 on PGU Port (S-Bus or Mode D)**

| PCD1 | PCD2 | PCD4 | PCD6 | xx7 |
|---|---|---|---|---|
| (✓) | ✓ | ✓ | ✓ | - |

Assignation of the serial port:

S-Bus (point to point connection): SASI S-Bus with port number 0, parity mode as slave, other parameters default values. Selected PLC type: S-Bus

ModeD: (not recommended, not for PCD1 and PCD2.M150 and PCDx.M170) SASI S-Bus with port 0, Mode D slave, other parameters default values. Selected PLC type: PCD



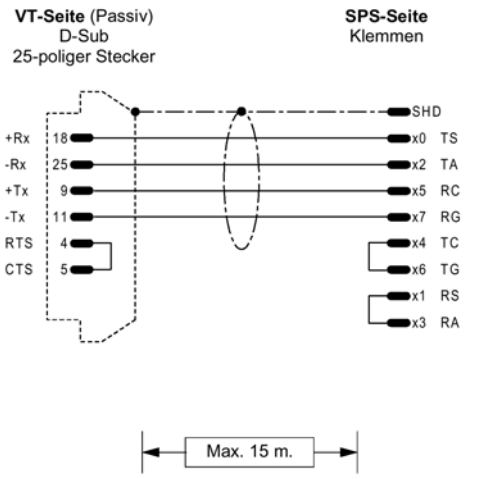

**RS 232 (S-Bus or Mode D)**

| PCD1 | PCD2 | PCD4 | PCD6 | xx7 |
|---|---|---|---|---|
| - | - | ✓ | - | - |

Assignation of the serial port:

S-Bus (point to point connection): SASI S-Bus with the relevant port number, parity mode as slave, other parameters default values. Selected PLC type: S-Bus

ModeD: (not recommended) SASI S-Bus with the relevant port number, Mode D slave, other parameters default values. Selected PLC type: PCD



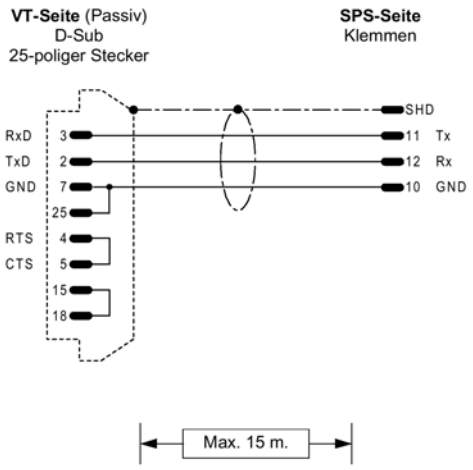

**Current Loop (S-Bus or Mode D)**

| PCD1 | PCD2 | PCD4 | PCD6 | xx7 |
|---|---|---|---|---|
| - | - | ✓ | - | - |

Assignation of the serial port:

S-Bus (point to point connection): SASI S-Bus with the relevant port number, parity mode as slave, other parameters default values. Selected PLC type: S-Bus

ModeD: (not recommended) SASI S-Bus with the relevant port number, Mode D slave, other parameters default values. Selected PLC type: PCD



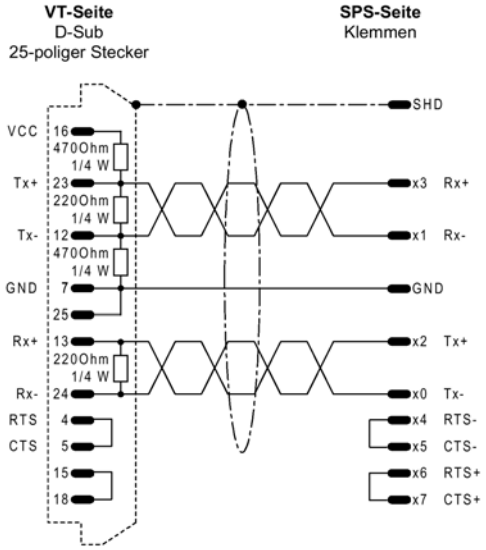

**RS 232 on PCD7.F120 (S-Bus or Mode D)**

| PCD1 | PCD2 | PCD4 | PCD6 | xx7 |
|---|---|---|---|---|
| ✓ | ✓ | - | - | - |

Assignation of the serial port:

S-Bus (point to point connection): SASI S-Bus with port number 1, parity mode as slave, other parameters default values. Selected PLC type: S-Bus

ModeD: (not recommended) SASI S-Bus with port 1, Mode D slave, other parameters default values. Selected PLC type: PCD



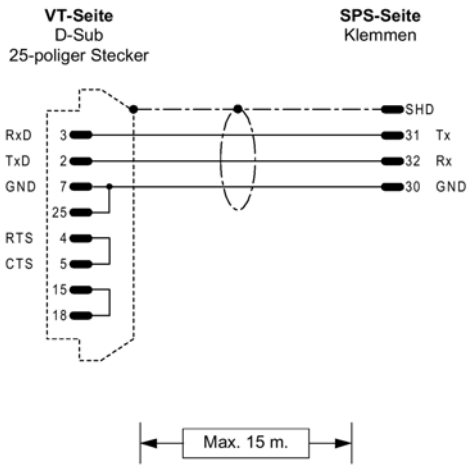

**RS 422 (S-Bus or Mode D)**

| PCD1 | PCD2 | PCD4 | PCD6 | xx7 |
|---|---|---|---|---|
| - | - | ✓ | - | - |

Assignation of the serial port:

S-Bus (point to point connection): SASI S-Bus with the relevant port number, parity mode as slave, other parameters default values. Selected PLC type: S-Bus

ModeD: (not recommended) SASI S-Bus with the relevant port number, Mode D slave, other parameters default values. Selected PLC type: PCD



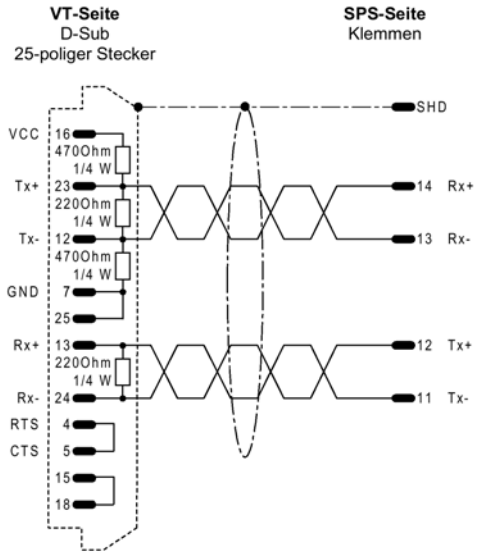

**RS 232 on PCD2.F520 (S-Bus or Mode D)**

| PCD1 | PCD2 | PCD4 | PCD6 | xx7 |
|---|---|---|---|---|
| - | ✓ | - | - | - |

Assignation of the serial port:

S-Bus (point to point connection): SASI S-Bus with port number 2, parity mode as slave, other parameters default values. Selected PLC type: S-Bus

ModeD: (not recommended) SASI S-Bus with port 2, Mode D slave, other parameters default values. Selected PLC type: PCD



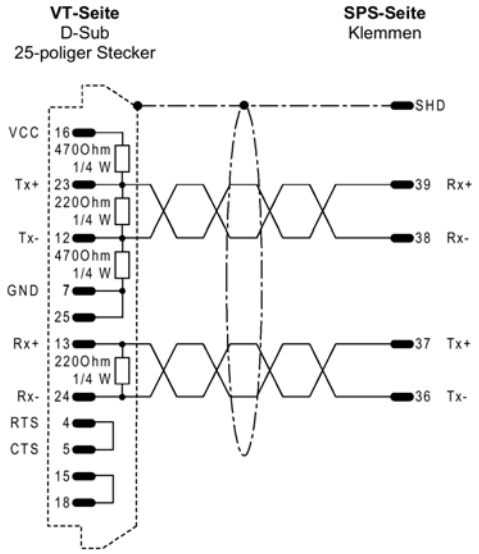

**RS 422 on PCD7.F110 (S-Bus or Mode D)**

| PCD1 | PCD2 | PCD4 | PCD6 | xx7 |
|---|---|---|---|---|
| ✓ | ✓ | - | - | - |

Assignation of the serial port:

S-Bus (point to point connection): SASI S-Bus with port number 1, parity mode as slave, other parameters default values. Selected PLC type: S-Bus

ModeD: (not recommended) SASI S-Bus with port 1, Mode D slave, other parameters default values. Selected PLC type: PCD



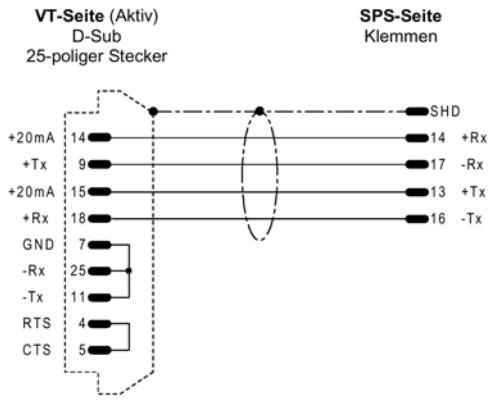

**RS 422 on PCD2.F520 (S-Bus or Mode D)**

| PCD1 | PCD2 | PCD4 | PCD6 | xx7 |
|---|---|---|---|---|
| - | ✓ | - | - | - |

Assignation of the serial port:

S-Bus (point to point connection): SASI S-Bus with port number 3, parity mode as slave, other parameters default values. Selected PLC type: S-Bus

ModeD: (not recommended) SASI S-Bus with port 3, Mode D slave, other parameters default values. Selected PLC type: PCD



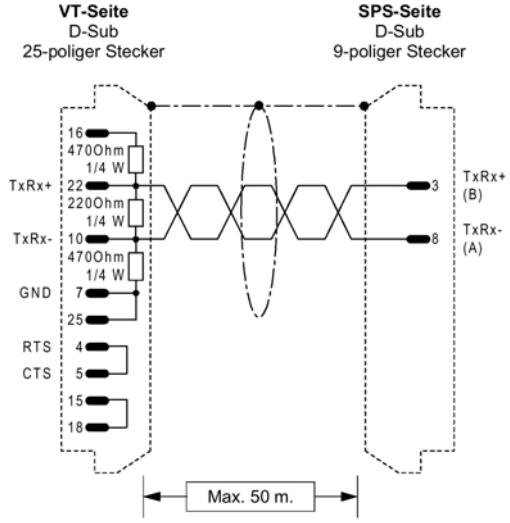

**Current loop auf PCD7.F130 (S-Bus or Mode D)**

| PCD1 | PCD2 | PCD4 | PCD6 | xx7 |
|---|---|---|---|---|
| ✓ | ✓ | - | - | - |

Assignation of the serial port:

S-Bus (point to point connection): SASI S-Bus with port number 1, parity mode as slave, other parameters default values. Selected PLC type: S-Bus

ModeD: (not recommended) SASI S-Bus with port 1, Mode D slave, other parameters default values. Selected PLC type: PCD

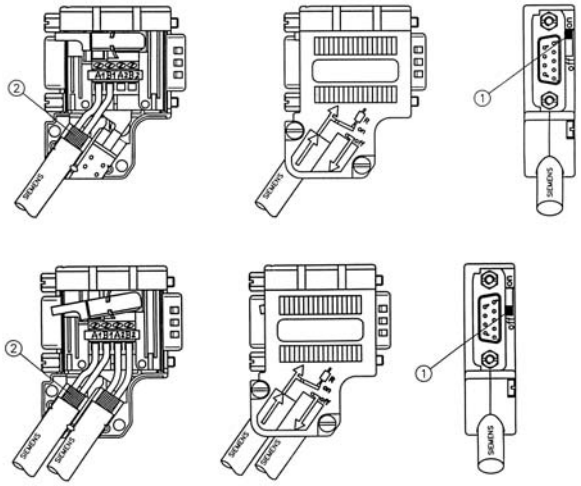



## RS 485 on MPI-Interface

| PCD1 | PCD2 | PCD4 | PCD6 | xx7 |
|---|---|---|---|---|
| - | - | - | - | ✓ |

Type of PLC selected: PCD1/2 xx7

Termination resistances only if terminal station of the MPI network. Make sure the bus is terminated the same way on both ends of the network. Adapt the resistors accordingly if necessary.

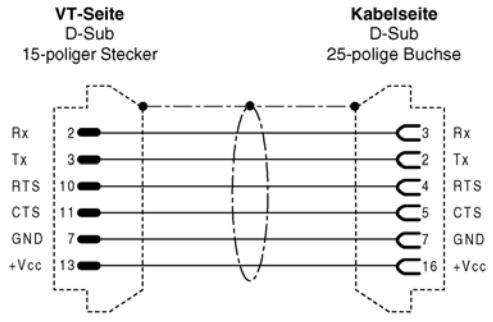


**Useful accessory for MPI:**

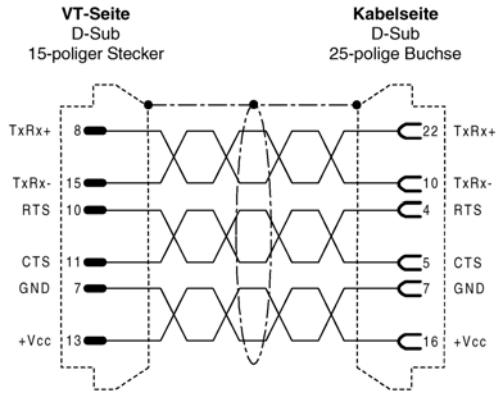
Bus connector with PG plug for the simultaneous connection of an MPI network and a programming unit on an MPI interface of an S7 or xx7 PLC.

The connector can be used for intermediate stations (set the switch to off, without termination resistances) and terminal stations (set the switch to on, with termination resistances).

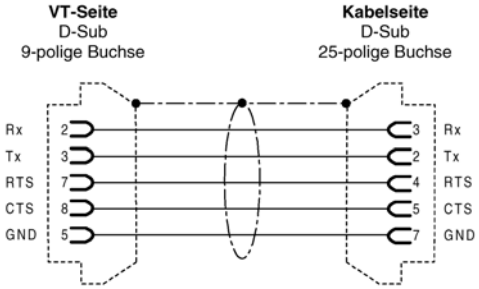
Can be ordered at Siemens,
order number 6ES7 972-0BB40-0XA0




**Adapter 15 to 25 poles RS 232**
**(ASP to existing cables for MSP)**




**Adapter 15 to 25 poles RS 485**
**(ASP to existing cables for MSP)**

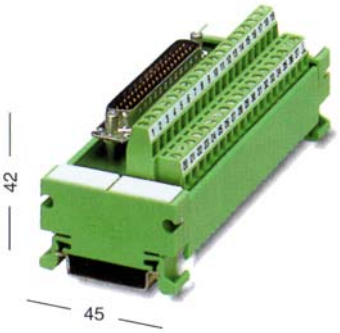



**Adapter 9 to 25 poles RS 232**
**(ASP to existing cables for MSP)**

### Alternative to soldered connection cables

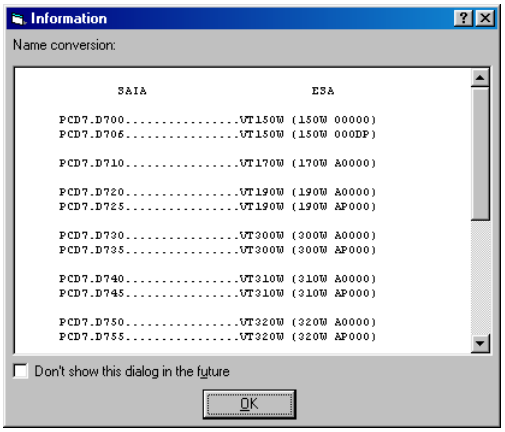
Who doesn't want to create soldered cables can use a shielded 25-poled D-Sub 1:1 cable (maximum length 1 m for RS 485) to connect the terminal to a screw terminal adapter. On the adapter you can add bridges and termination resistors as desired. One possible supplier is Phoenix (price per module about CHF 45.- in Switzerland).

Image below: UM45-D25SUB/S Order number 2962793 (2 levels of screw terminals)
Alternative: FLKMS-D 25 SUB/S Order number 2281720(3 levels of screw terminals)

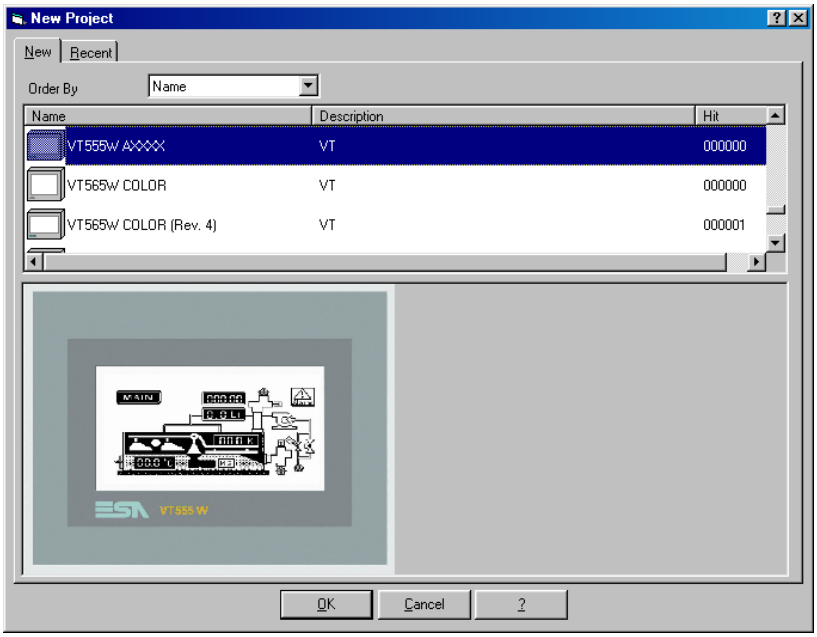


# 5) Create the first application

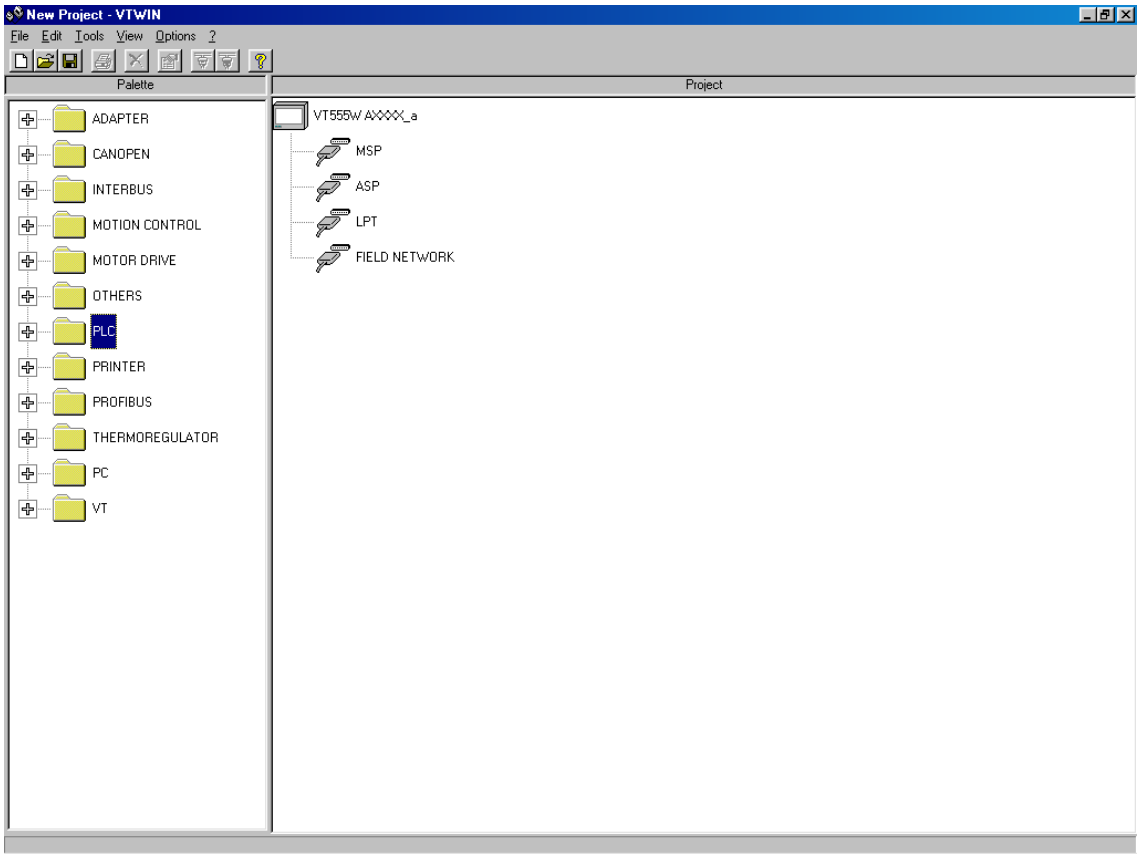
**Create a new project:** Run the PCD8.D81W software, the first screen is for the equivalence of SAIA and ESA designations, klick OK:

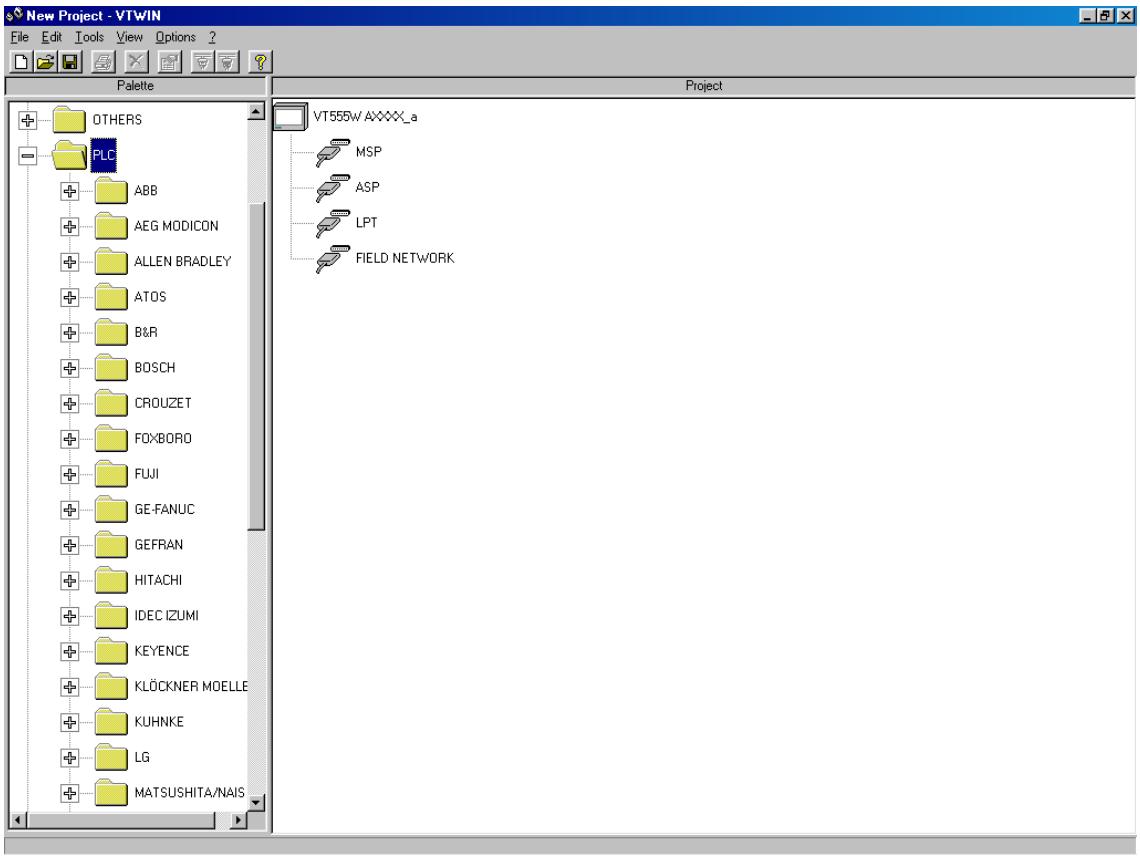


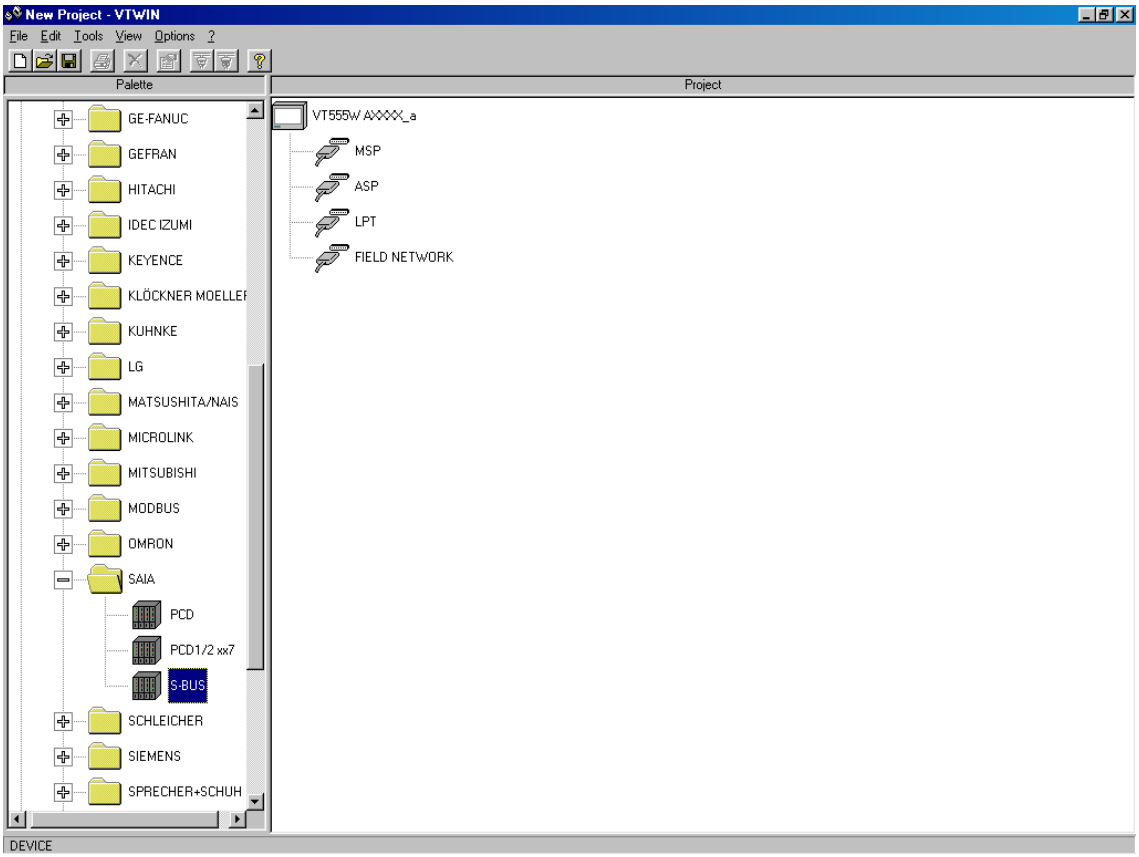

Select the terminal to program (or select the project under recent if you continue an existing project)
A PCD7.D760 which is equivalent to a VT555W AXXXX is selected for this example, confirm with OK.



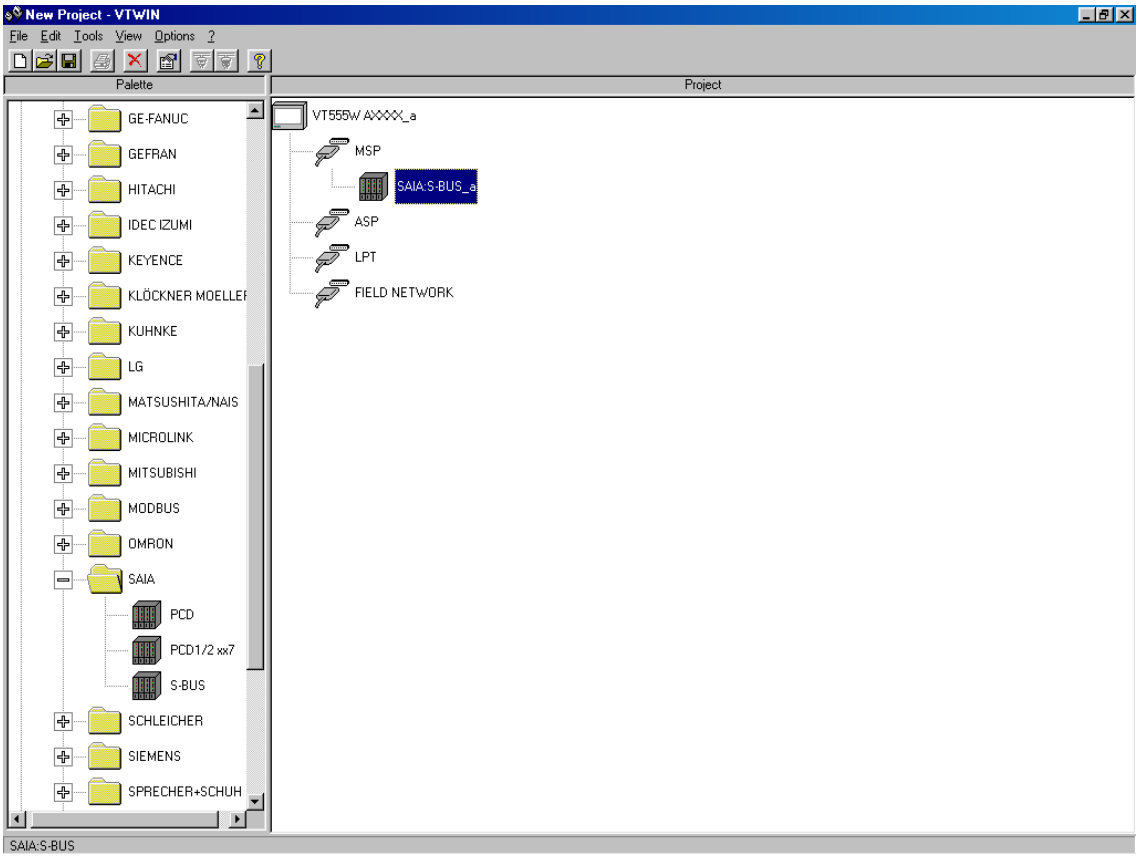
The terminal you selected is shown as follows and you are able to select the PLC the terminal is connected to within the list at the left hand side.





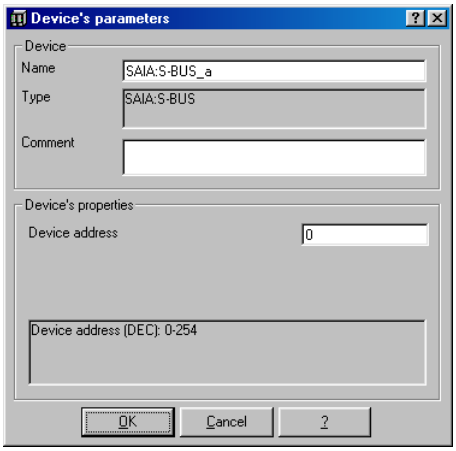

Select the S-Bus Driver with the left mouse button, keep the button pressed and drag the driver to the MSP port of the terminal:



The program is checking if a driver/unit can be connected to a port of the terminal. If this is not allowed, the connector doesn't get blue.

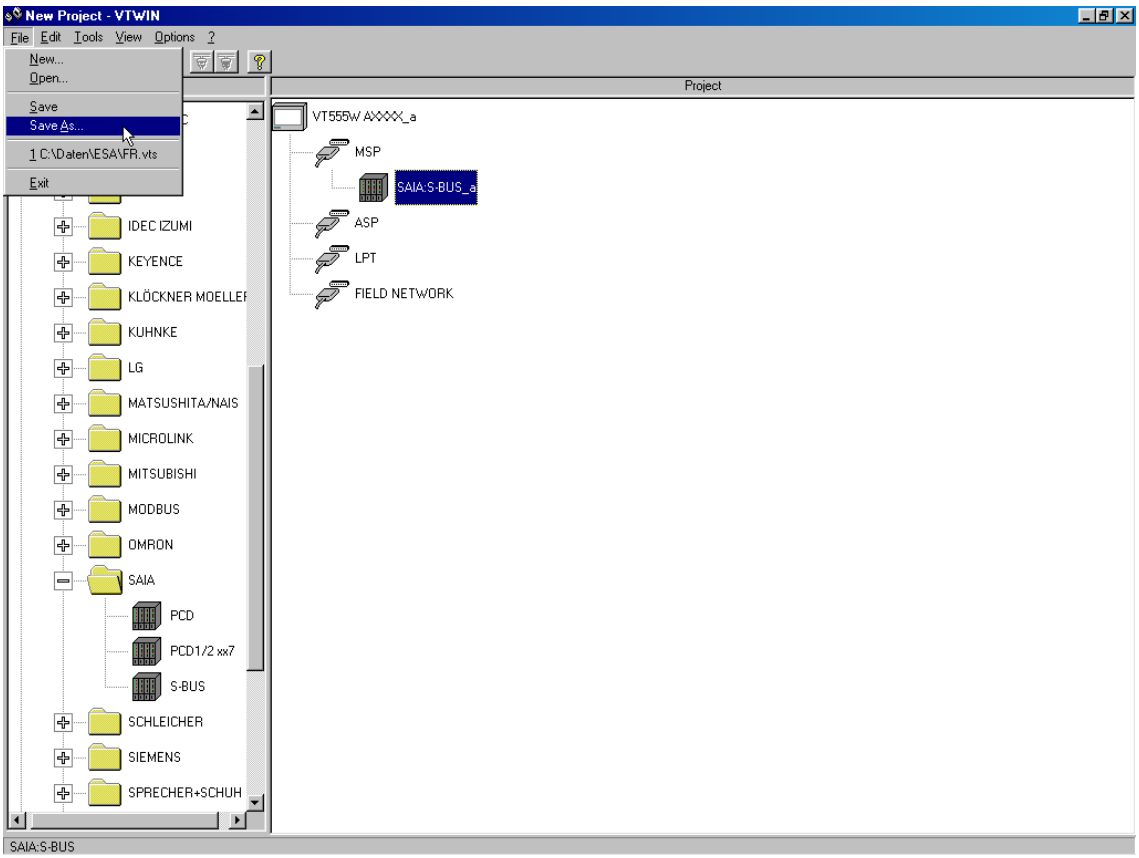
Always use the S-Bus driver and not the so called PCD driver since the PCD driver can lead to communication interruptions. (??? on the display instead of values).

As the terminal is S-Bus master you could add more drivers/units to the MSP. The S-Bus addresses of the relevant stations can be selected by double clicking on the PLC symbol:

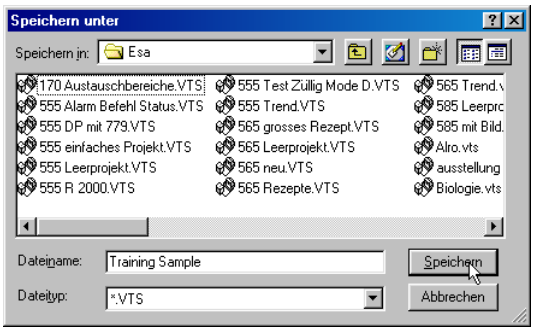



This choice is very important since the addresses have to match with the confuguration of the PCD or no communication will be possible.

Save the application at this state to define it's name:



Introduce a file name and select the folder where to store the project, then press Save:




It is important to know that the program consists of two major parts:

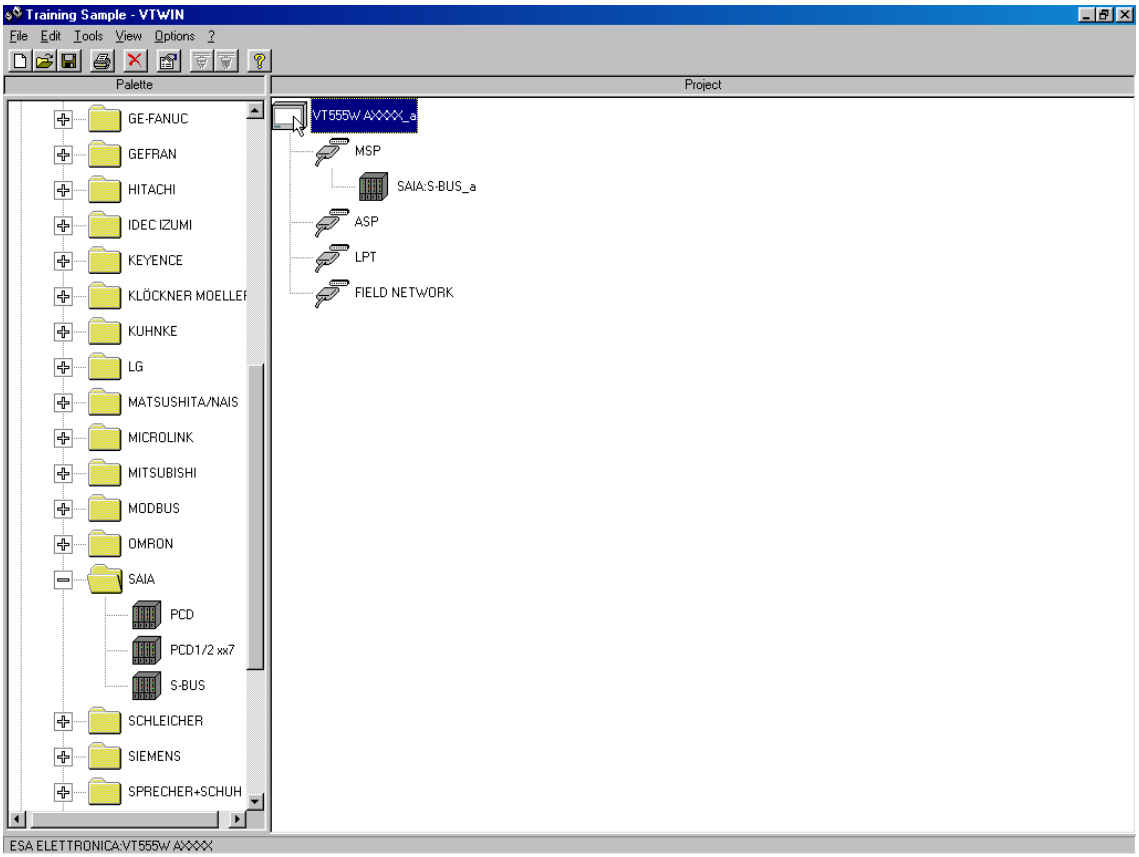
- the network configuration part where we have been until now
- the terminal dialogue creation part where we will change to just after this

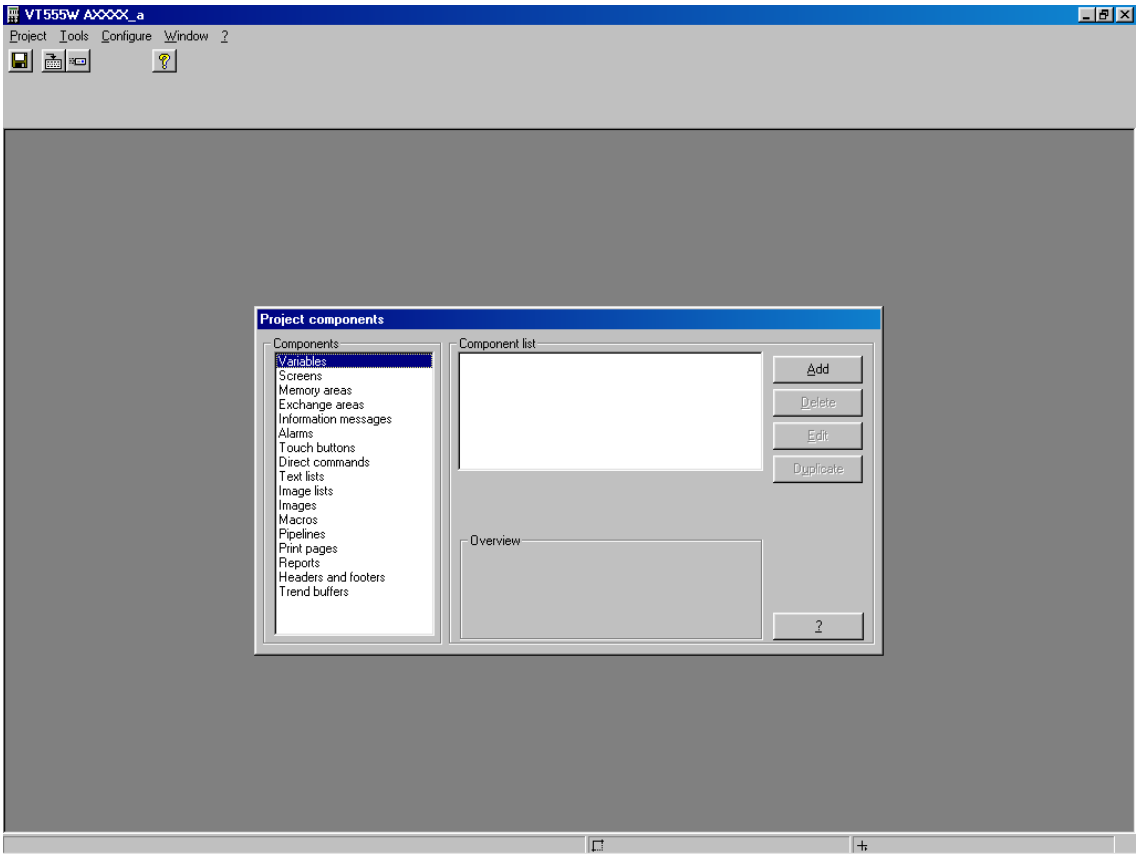
Regarding the saving of applications there is a property of the current versions you must know:

- **the project is only really saved on disk when you save in the network configuration part**. If you store in the dialogue creation part it seems that it stores as well, but this is not really the case. You must come back to the network configuration part regularly to store there to avoid losses.

Crashes are rare with this software, but according to our experience it is better to **work locally** on the harddisk and to make backups on the network. Most of the crashes we have seen happened to users who worked on projects that were stored on a network server.

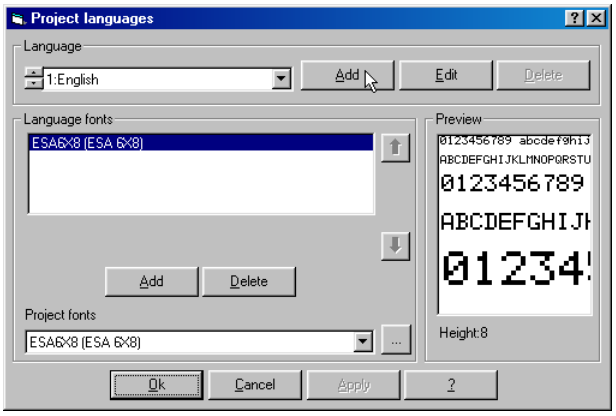
Open the terminal dialogue creation part of the software by double clicking the terminal symbol:






You can see a screen with the project components. The next step will be to define new components, which means variables, screens and so on.

The terminals support several user languages. Define what languages you want to use by choosing the menu configure, project languages and fonts:



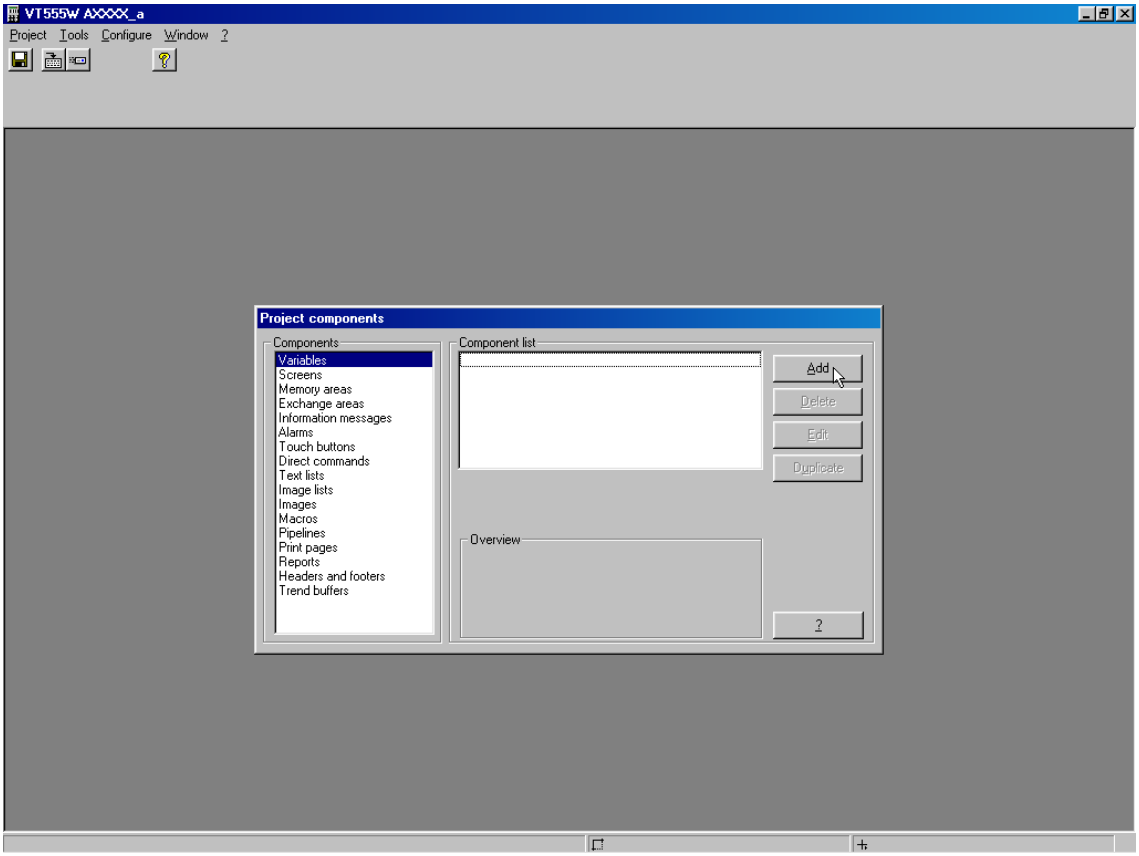

Add new languages as required. Defining the languages at the beginning allows to introduce the texts in all the languages at a time.

In the same window the font to be used for the language can be defined. If you select true type fonts you need to be aware of the following facts:

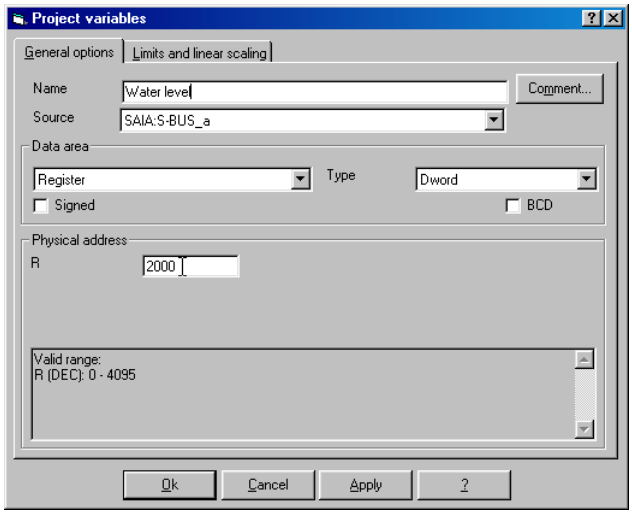
- using true type fonts can lead to problems with the maintenance since it is not said that the PC of the service engineer who would like to do a modification has the appropriate fonts installed. Use common fonts (Arial...) and store the fonts used together with the project. For copyright reasons the fonts can't be integrated into the projects automatically.

- if you select true type fonts for one language, this has as well to be done for the other languages (no mixing possible)

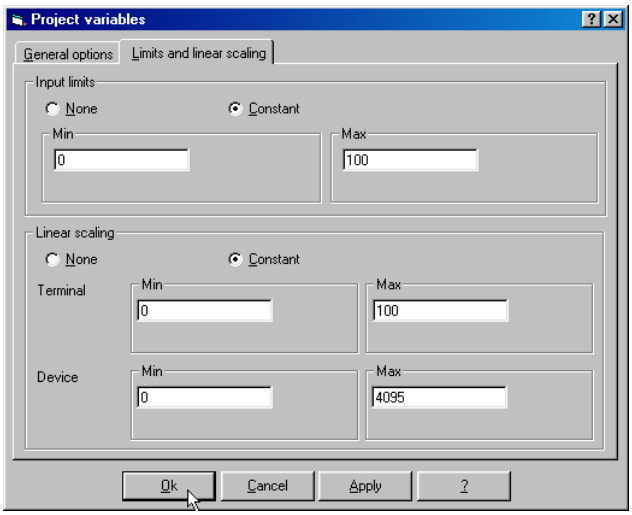
Add a couple of variables by choosing the variables component and pressing the Add button:




Define the name, the media and address of the variable:



Define limits and a linear scaling if required, confirm with OK:




The figures given above lead to the following result:

- Entries of the user are limited to the range 0 to 100
- Values in the register between 0 to 4095 lead to a value displayed of 0 to 100 (of course this scaling can as well be done in the PLC if preferred)

Please note that at this state we don't care about a decimal point that will may be be defined when the variable is displayed. We are interested in the integer value for the moment.

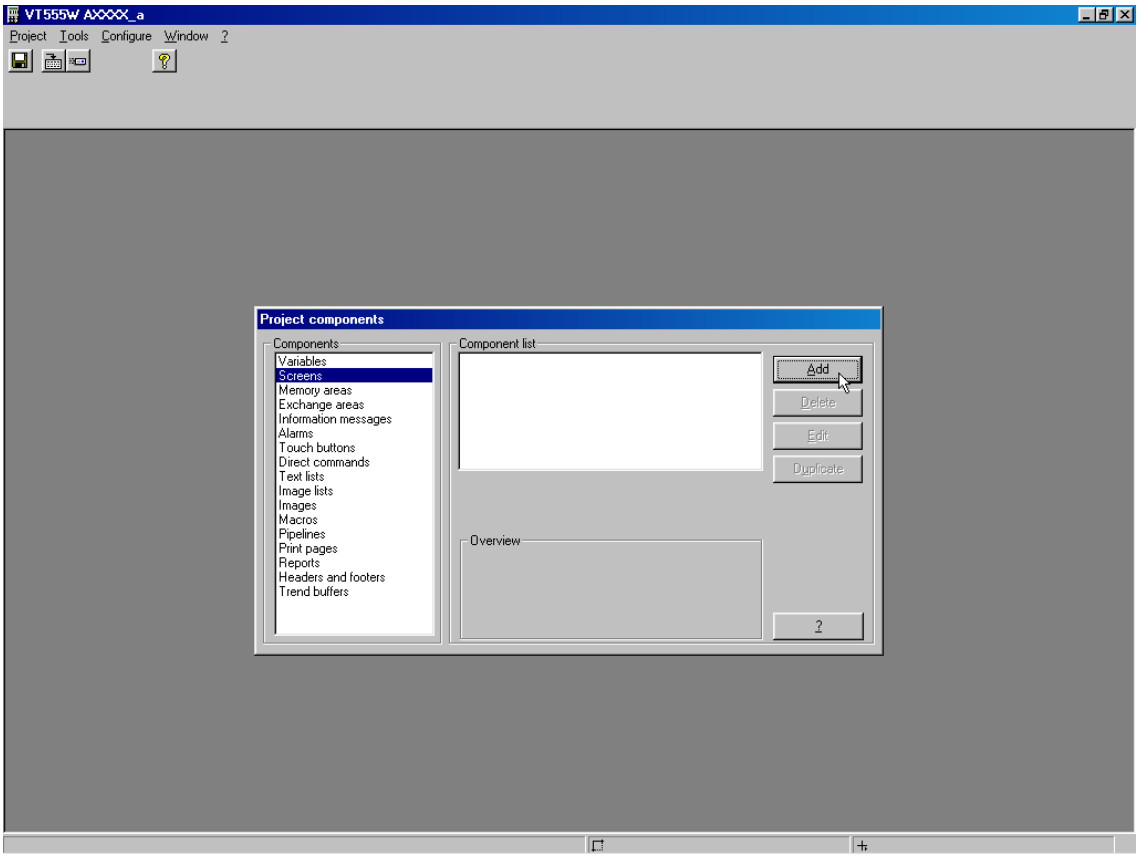
You can continue like this for the other variables. Variables can be added to the project at any time, but it is simpler to define them at the beginning.

Unfortunately there is for the moment no way to import ressources from PG5, which means

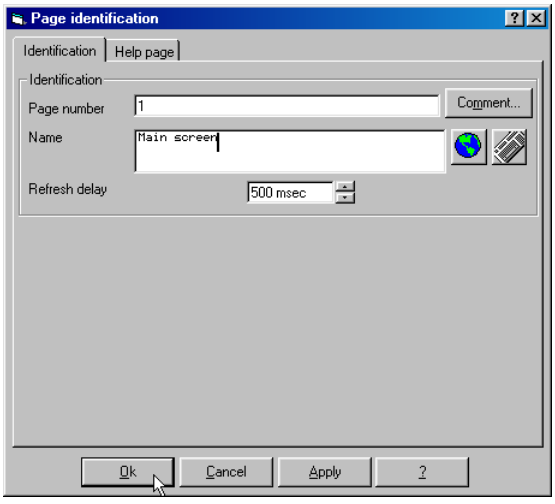
- quite a lot of work to redefine the variables that are used on the display, a source of errors
- absolute adresses have to be used
- the symbol names are not automatically the same in PG5 and VTWIN

Hopefully this will be improved with future PG5 versions.

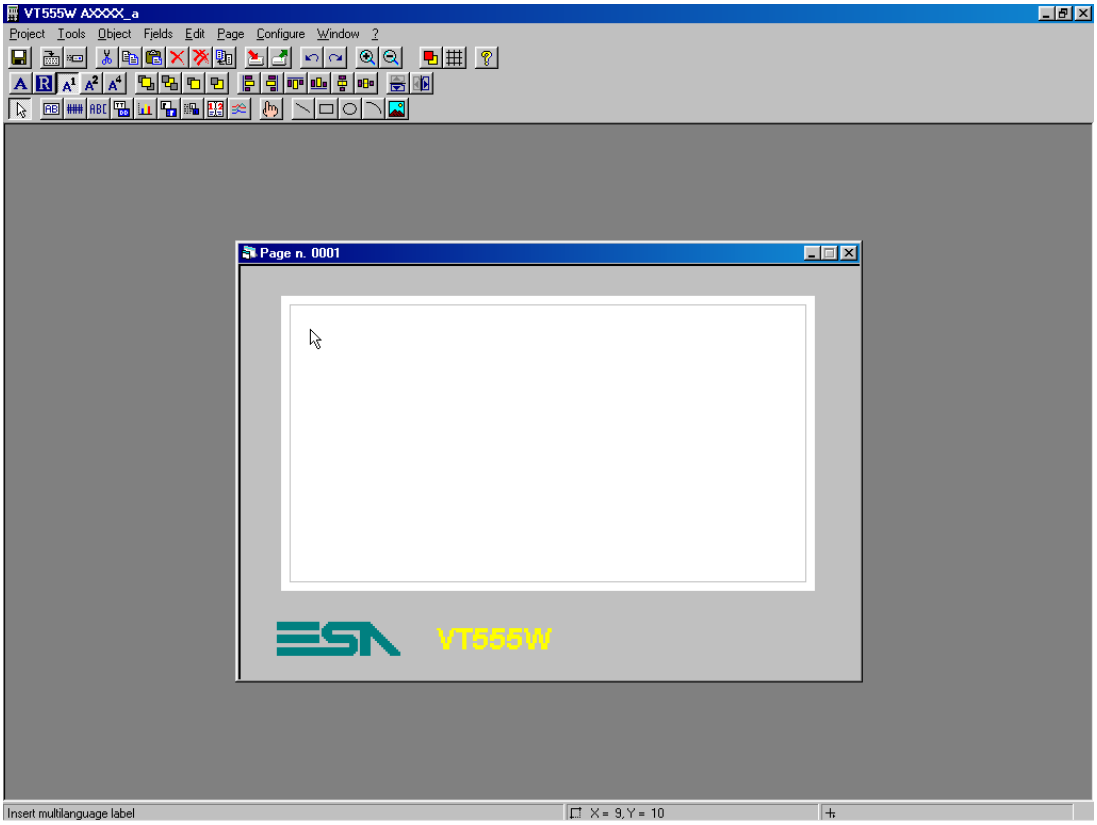
Create a new page by selecting the components screens and clicking on Add:



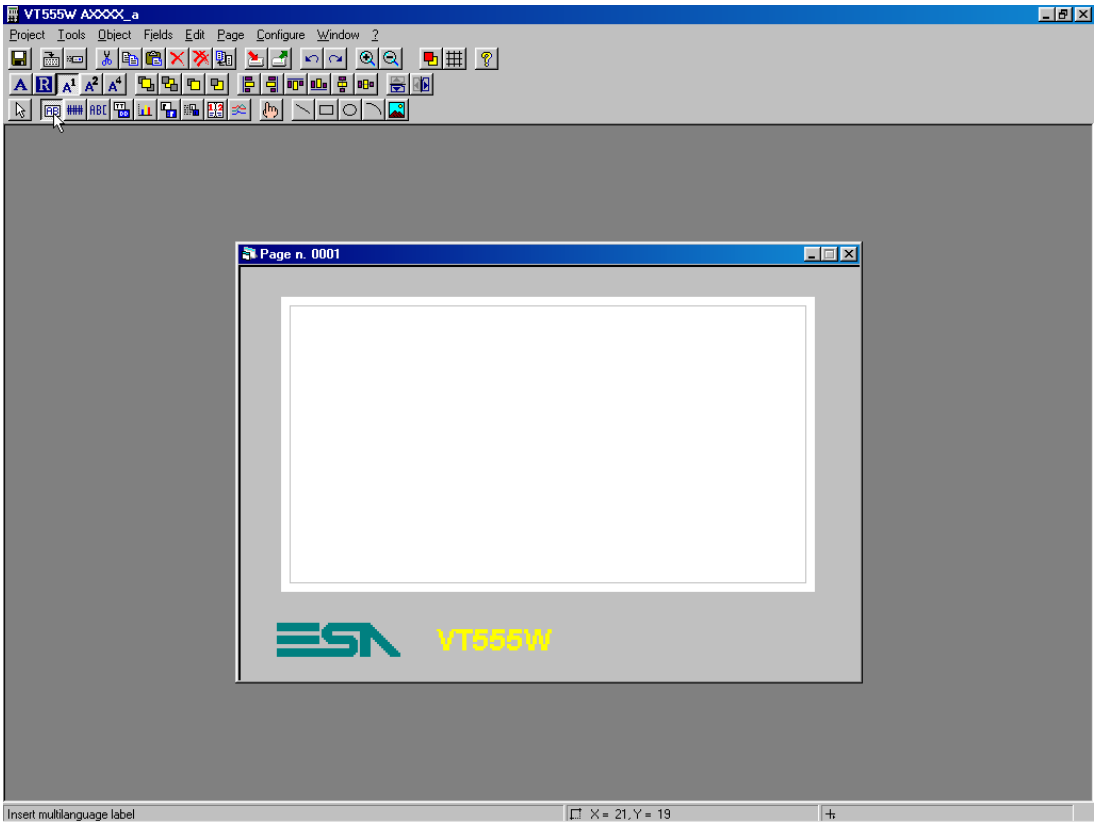

Define the page and a name, confirm with OK:



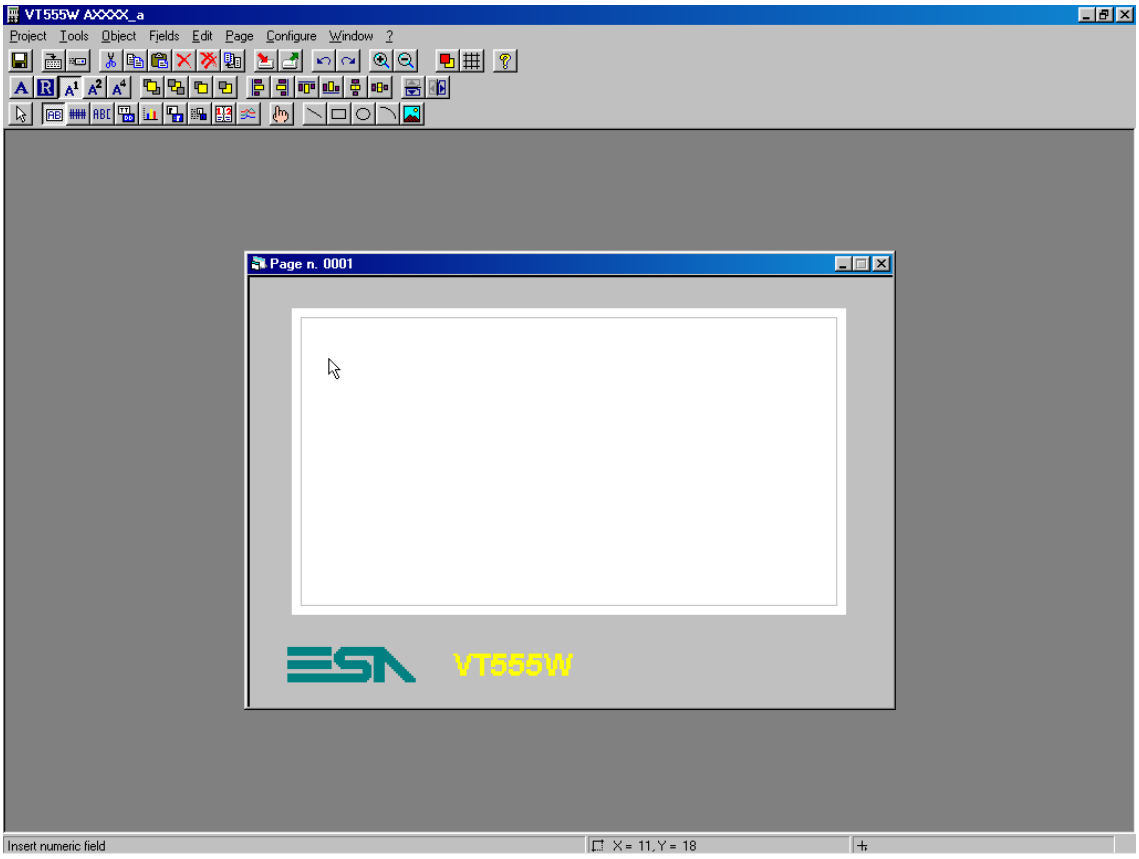
Now you can see the page editor with a graphical representation of the terminal and the available tools:




Place a multilanguage field by selecting the relevant icon,



then position the cursor on the white face of the terminal and click left:



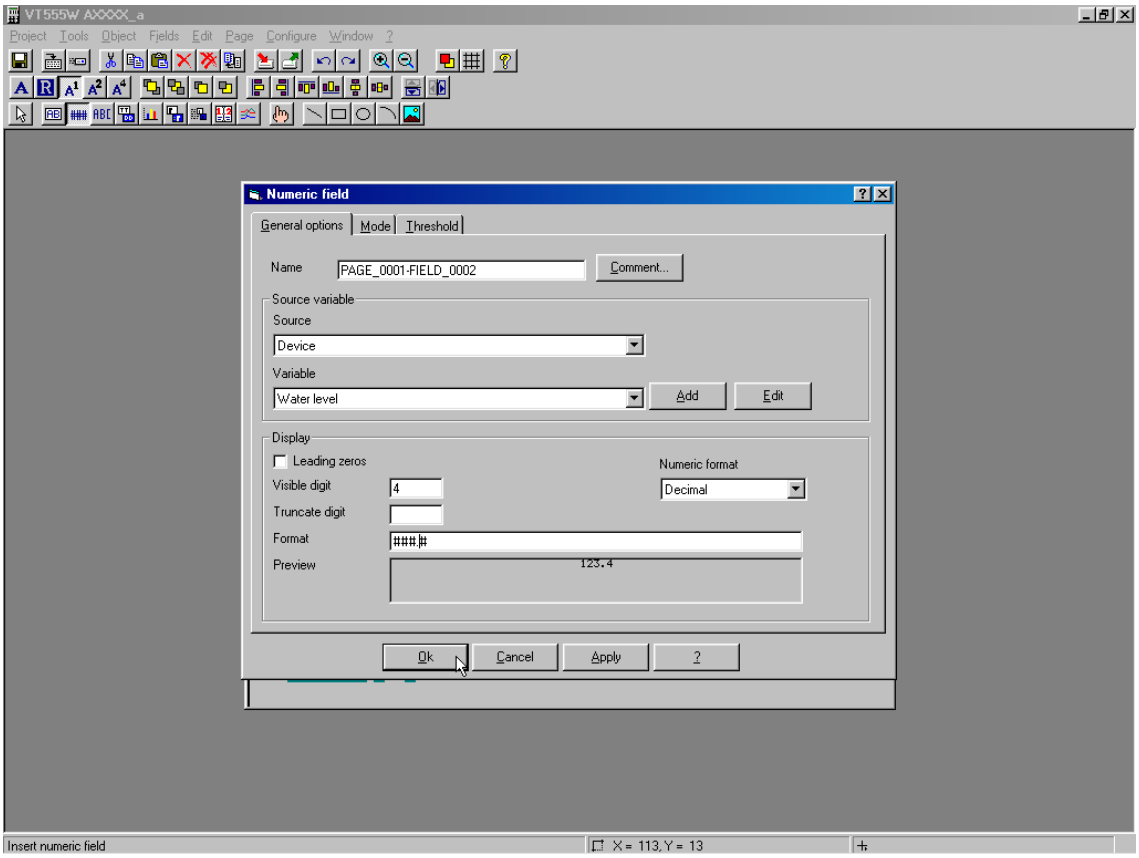

Placing new objects at locations where there is enough space at the right hand side is always a good idea. If you don't do this you can end up in a situation where you are not able to introduce the full length of the text since there is no more space left. After placing an object it can easily be moved to the desired location.

Introduce the text in the standard (first language) and press enter:

You now see a window which allows to introduce the translations:

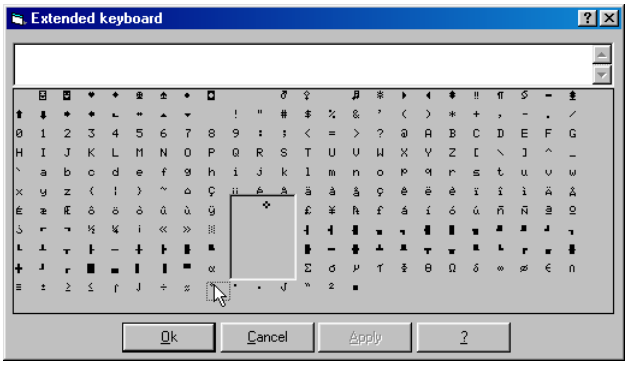

If you followed the same example you could see that the translations cannot be longer than the standard text (which is not very nice). What you can do is to cancel, add spaces to the original text and come back to this window pressing enter.

Introduce a numeric field, by selecting the appropriate tool and placing an object on the white screen surface:



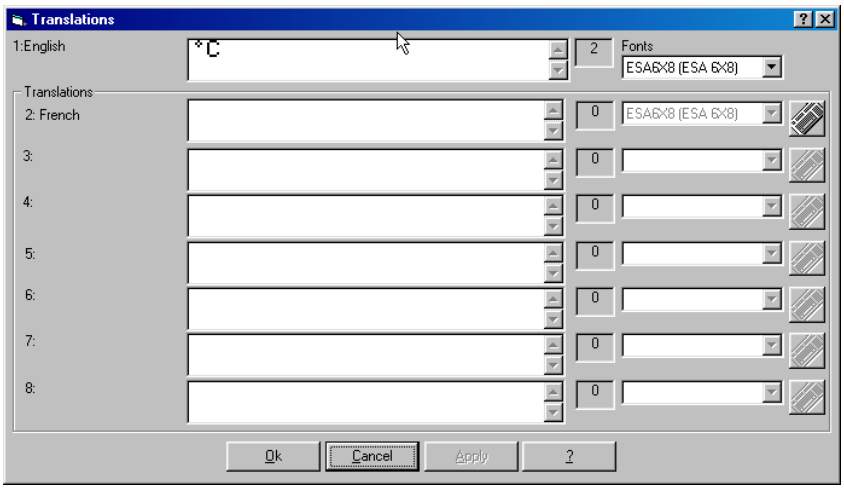

You can define how many digits you would like to display and in the format field you can add „cosmetic“ characters like decimal points.

Add a multilanguage text field for the physical unit °C. To place special characters you can **press F12** during the introduction of the text. This allows you to select characters from a table:



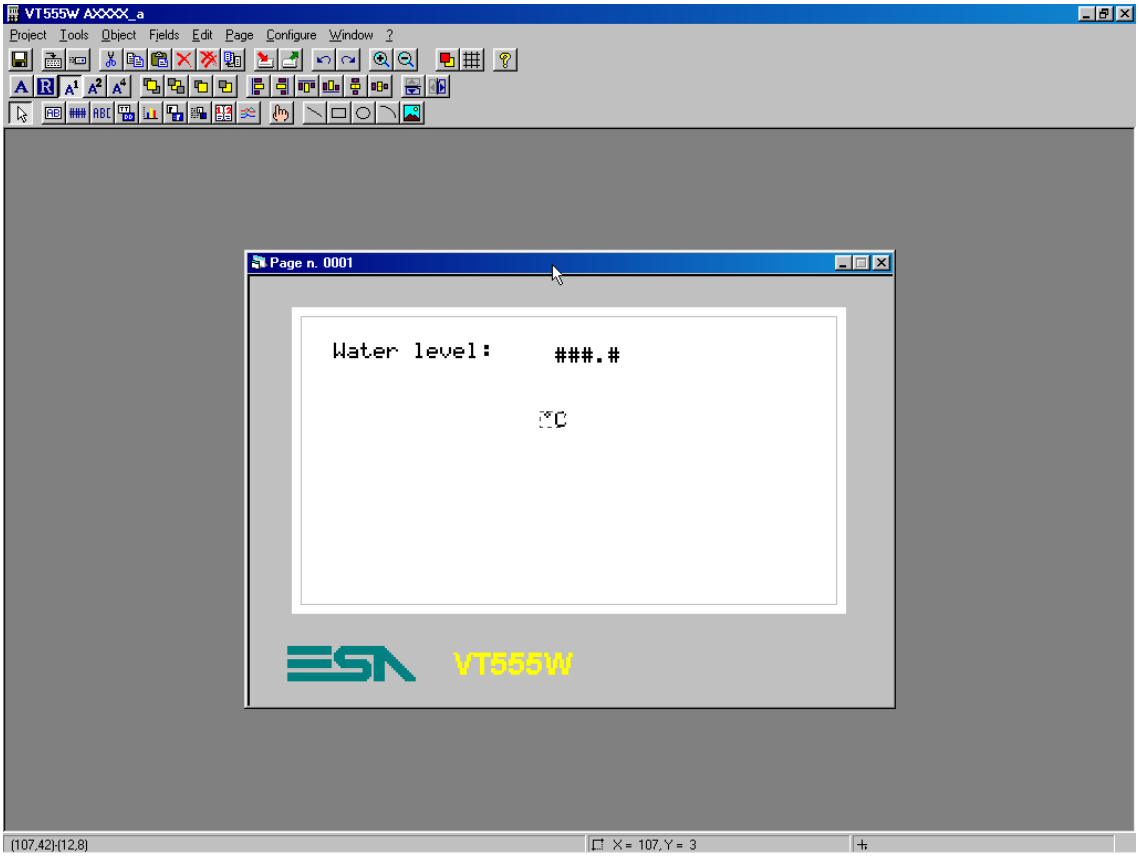

With true type fonts the characters should match with what you have on your keyboard, for the standard fonts to character position is not always the same and wrong characters can be displayed. This often seen in german and french with öäü and éàè. If you use the F12 table the right characters are displayed as well with the standard character sets.

If you don't introduce translations for the non-standard project languages, the standard text is copied,

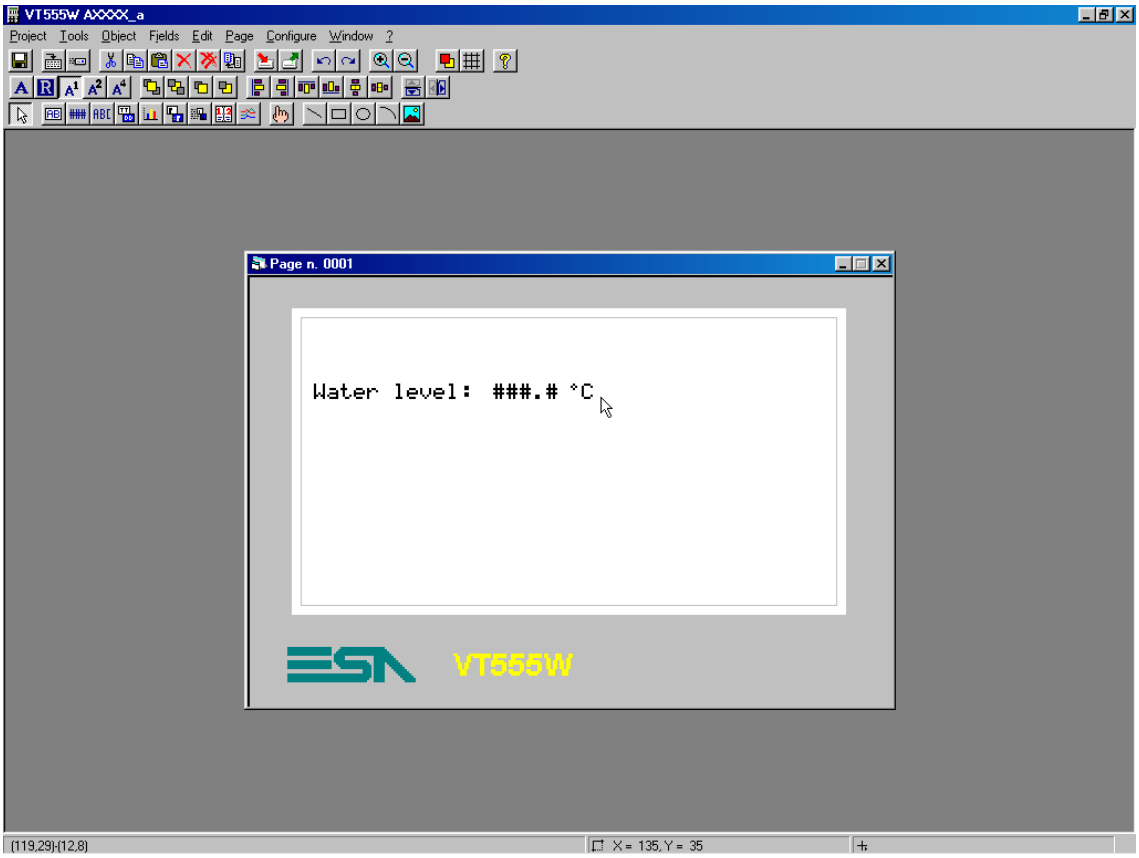



so if you confirm as shown above °C will be displayed for all project languages.

The result is as follows:

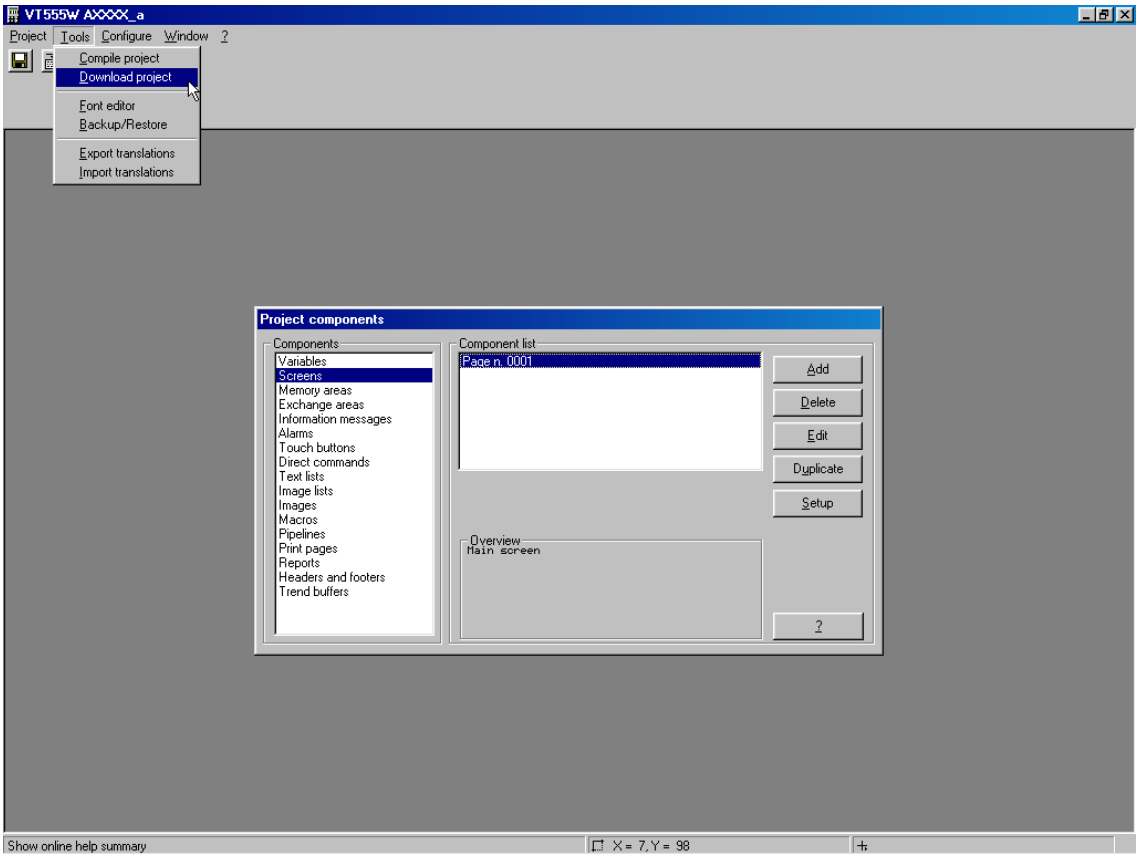



The objects can now be placed in their final position so the example looks a bit nicer:



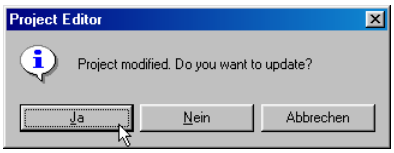

We are now ready to download our first example into the terminal.

Close the page and select in the menu tools, download project:

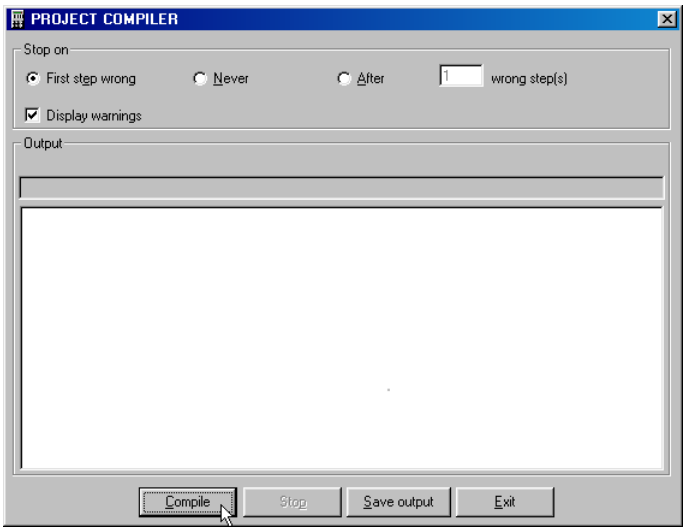


It is only worth choosing the option compile project from the same menu if you want to test whether your project is free of errors. If you choose download and the project needs to be compiled, VTWIN asks for this automatically.

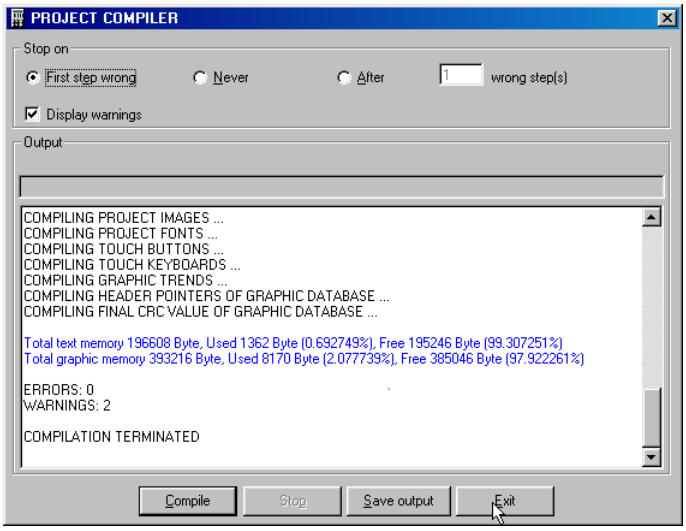
Be careful with the following message: as said before you need to go back to the network configuration area to really save your application to disk.




Start the compilation:

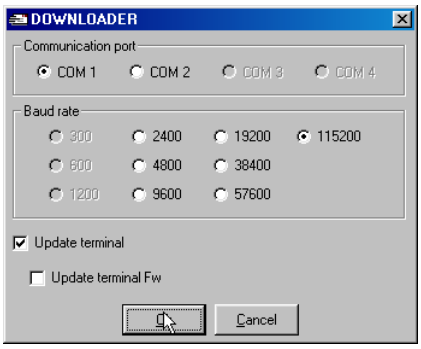



If everything is fine you shouldn‘t have errors (but warnings if you didn’t define the texts in the other project languages:




Click on Exit:

Select the port on your PC and the baudrate used for the download:




The baud rate you select has nothing to do with the baudrate used later for the communication with the PLC, so try with 115200.

Before you start the download you need to connect the programming cable and to set the terminal in the appropriate mode to receive a project:

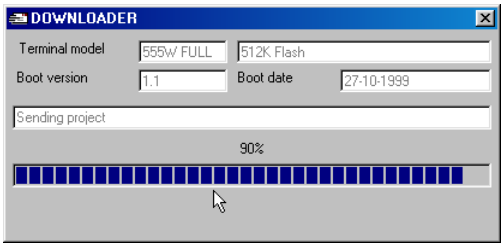
For touchscreens:

Press simultaneously on two diagonal corners of the screen, then select TRAN PAGE to get to the transfer page, select the port to be used for the download.

For terminals with function keys:

Press shift enter, wait for the message „wait for boot forced“, then select the port to be used for the download.

After this select OK to start the download:




If the fields of the downloader stay white you don’t have a connection to the terminal. Common causes:

- terminal not in the right mode
- programming cable not connected properly
- wrong port on PC used

On certain laptops you can have the problem that the download starts, but stops before it is finished. If this happens close the window, set the terminal in the reception mode again after removing power, start the download again and try step by step with lower baudrates.

Regarding the firmware the terminals are very simple to handle. If VTWIN contains a later firmware version that is available in the terminal, the new version is automatically downloaded. The first download takes longer if this is the case since before the project the new firmware is downloaded.

What you have to do to get the terminal and the PCD talking together:

- make sure the PCD port you indend to use for the link is not configured as S-Bus PGU port (the SASI we will place will be red all the time if this is the case, the full protocol is not required by the terminal)
- configure S-Bus support and select the right S-Bus address
- place a SASI S-Bus Slave FBox with the relevant port number and parity protocol
- the baud rate is 9600 baud as default, which is sufficient. As there is very little traffic on this serial line it is not worth increasing the baud rate
- create or take an appropriate connection cable and connect the PCD to the terminal

If this fails see the FAQ section of this document.

# 6) FAQ: Frequently asked questions

60% of the support questions can be covered with this document using the hints below. Another large part can be answered with the excellent online help and the manuals.

**Compilation error, „sensitive element is overwritten by...“ on touchscreens**

- two elements with the option edit allowed = on are one above the other or too closed together

**Download suddenly stops before it is finished**

- normally only happens on certain laptops. The problem was quite common with version 2.xx and 3.xx, but should be seldom with versions 4.xx
- cut the supply of the terminal an put it in reception mode again (see chapter 5 if you are not sure about this)
- close the download window, start the download again and try with the next lower baud rate
- failing this repeat the steps above each time with a lower baud rate

**Terminal blocked, cannot be put in the reception mode**

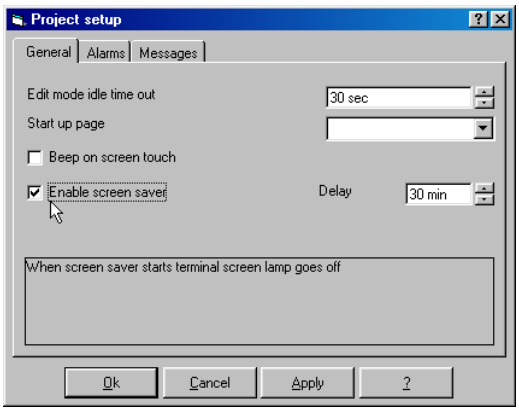
- can happen after non-complete firmware downloads
- procedure for touchscreens: find somebody to help you, cut the supply, press the two diagonal corners as usual and stay on it, apply the supply and remove the fingers when the terminal is ready to receive a firmware or project
- procedure for terminals with function keys: cut the supply, press coninuously shift + enter, apply the supply and remove the fingers when the terminal is ready to receive a firmware or project

**No communication between the terminal and the PCD (very popular question)**

- make sure your customer has this document available
- make sure the right port ASP, MSP has been selected, all the cables in the documentation are for the MSP which is available on all terminals of the series PCD7.D7xx
- make sure the S-Bus driver has been selected in the VTWIN application (if this is not the case assign the port of the PCD with Mode D, slave and the appropriate baudrate)
- if the SASI FBox is red this could be because of an S-Bus PGU configuration of the same port, a baud rate too high together with the baudrate of the other port using the same UART (for instance port 0 and 1 both 38.4k doesn’t work, see manuals PCD1+2...), or a hardware problem (port not existing, module missing)
- check the parameters of the SASI FBox. For S-Bus the parameters should be S-Bus parity slave, default 9600 baud or matching with what has been selected in the terminal project
- check the S-Bus address to match with the terminal project
- check the cablework according to the drawings in chapter 4, for S-Bus try to twist the two communication wires
- the problem should be solved at this state since upon now there was never a problem with a terminal regarding this. If you can download your application you have a big chance that the serial port is working

**Backlight getting darker and darker with the months and years**

- the fluorescent backlights of the graphical terminals are not made to be on all the time. Their life time is limited and you need to program a timeout which turns the backlight off after a while without key presses. Do this in the menu configure, project settings:

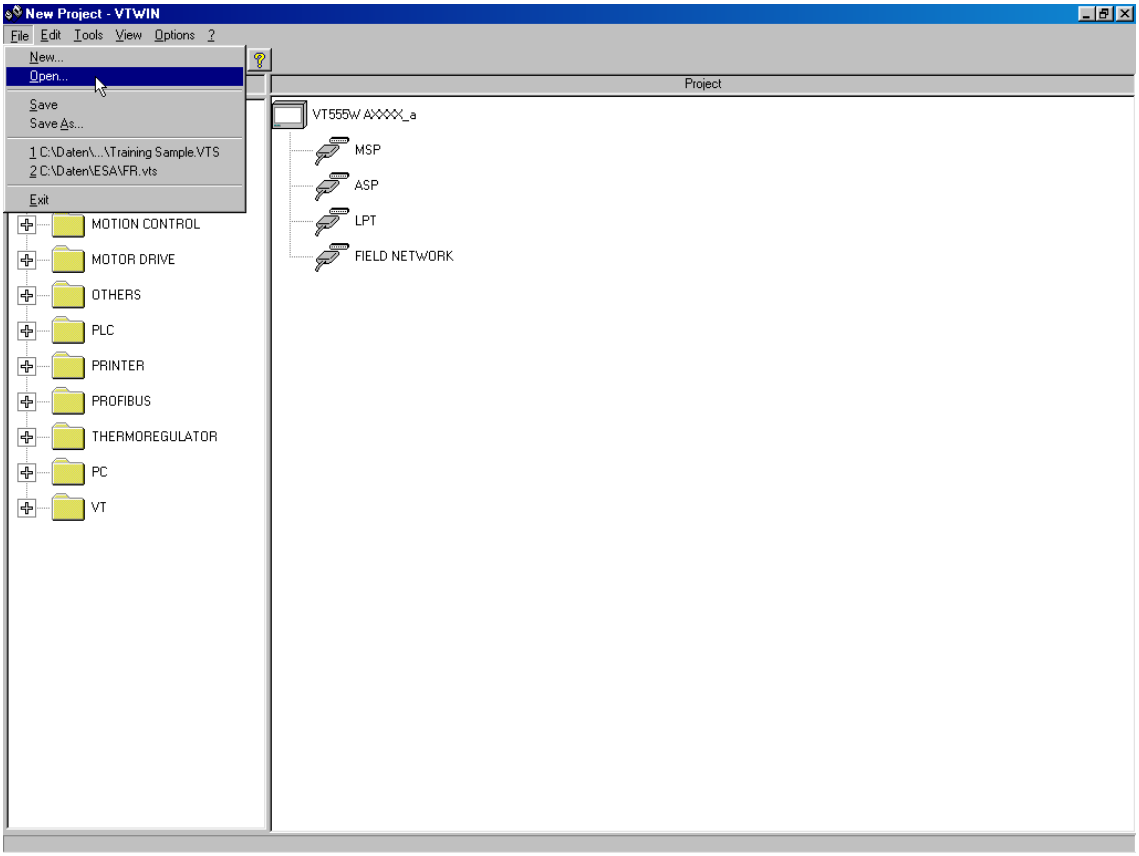


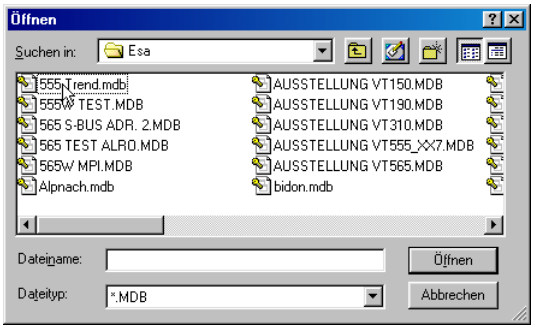

This should be done for all graphical terminals as default.

# 7) Open an existing .mdb project

The 2.xx versions of VTWIN did store their projects in the .mdb format. If you want to import such projects into version 4 you can do this in the network configuration area choosing file, open by selecting the file type .mdb.

After this the old project is converted to a .vts file (new standard for single terminal applications). The .mdb file is not deleted but no longer used. You can keep it for backup purposes.



# 8) Define the structure of an application

Before you create an application it is worth drawing a rough structure on paper. A clear structure helps the user significantly to navigate and access the information desired.

### 8.1) Terminals with function keys

These terminals allow to define sequences of pages, which simplify the work of the programmer and the user. A sequence is a collection of pages which belong together because they are about the same topic, for instance online values, maintenance parameters, clock parameters....).

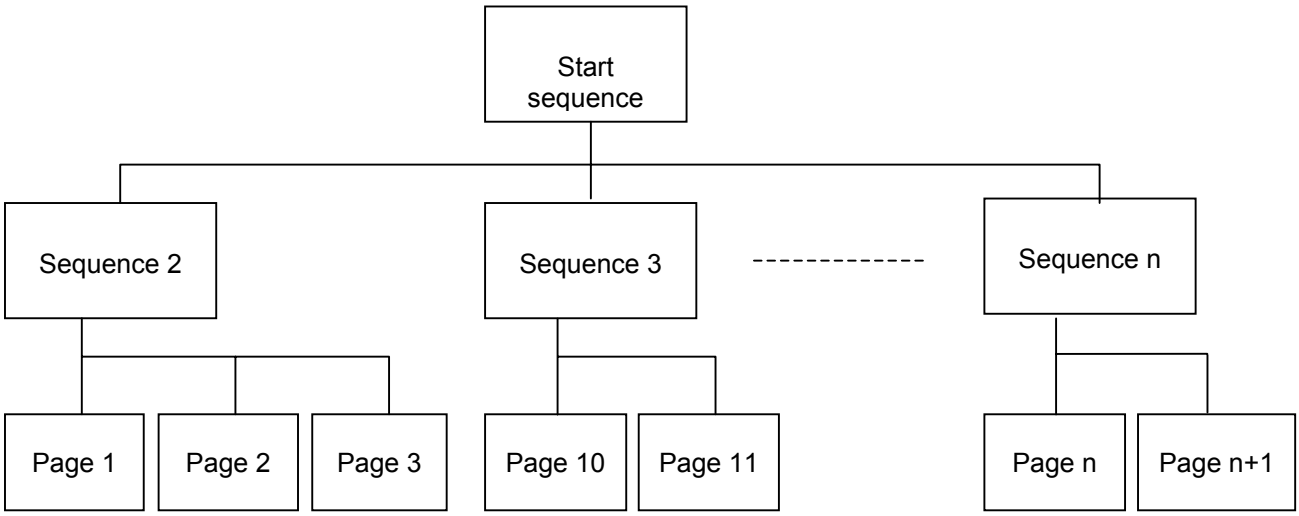
With version 4 of VTWIN sequences can consist of several pages with page numbers in a line (start / stop sequence, for instance page 1 to 10) or a random sequence (for instance page 1, 3, 5, 2, 10...).

The advantages of the use of sequences are:

- for the programmer: nothing has to be programmed to navigate between the pages of the sequence, the page can be changed using the arrow keys. No space is needed on the screen for the navigation within the sequences
- for the user: simpler to use

To change from one sequence to another a function key has to be assigned with this function. This requires that the meaning of the F-Key has to be put on a label or mentioned on the screen.

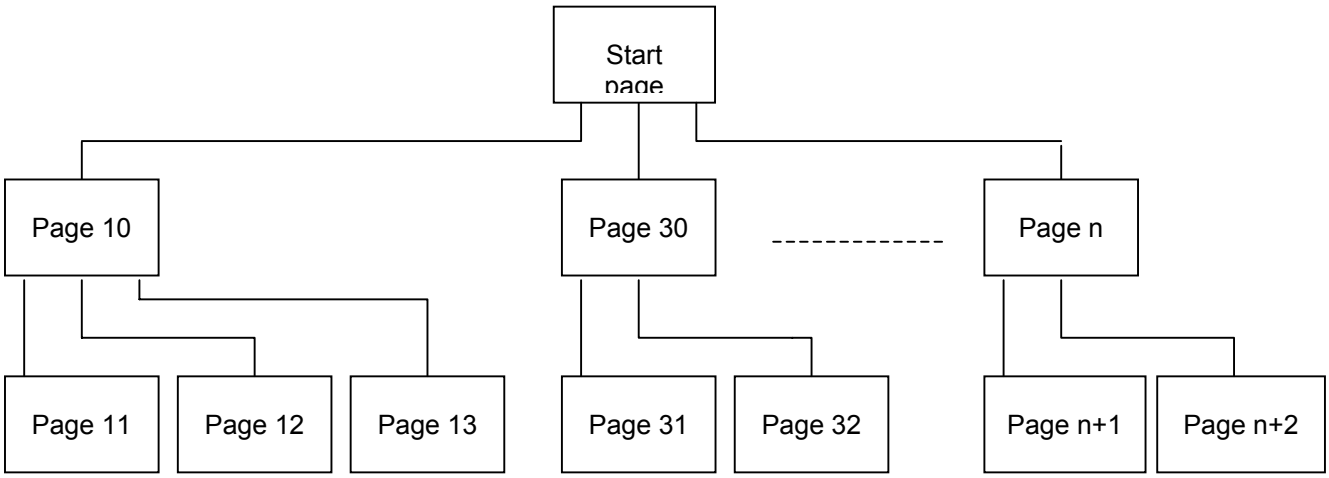
Typical structure:



### 8.2 Touchscreen Terminals

The touchscreen terminals don't allow to define sequences. The programmer decides by placing buttons on the pages what jumps are possible. It is all the same worth the pain to draw a structure on paper before starting.

Typical structure:

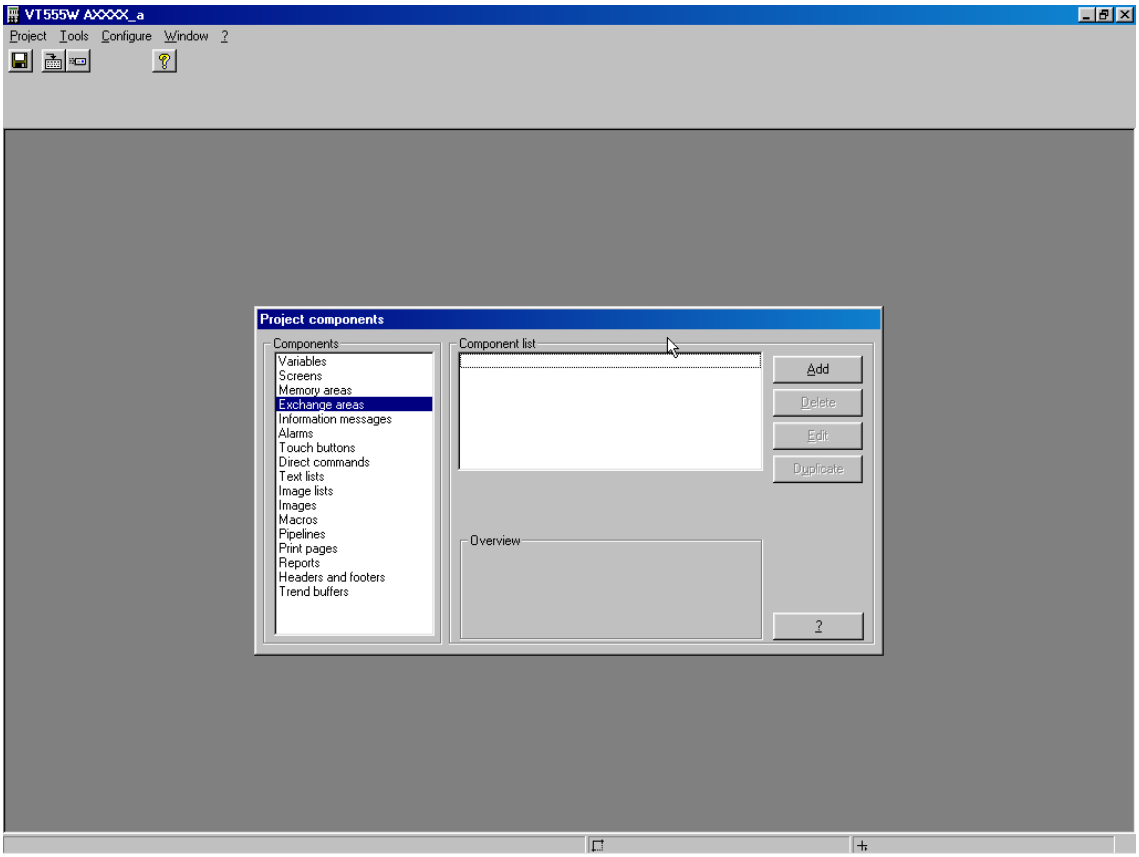


# 9) Exchange areas

For the data exchange between the PLC and the terminal there are several exchange areas available. Remember that for the access to the PLC's ressources no exchange area is necessary.

- Input areas (the terminal is reading): for instance for alarms, info messages
- Status areas (the terminal is writing): for instance for the status of keys or the printer
- Command area (the terminal gets a command from the PLC): for instance set the clock, jump to a certain page

Definition of exchange areas: first step: define a memory area: select the exchange area component and click on Add:

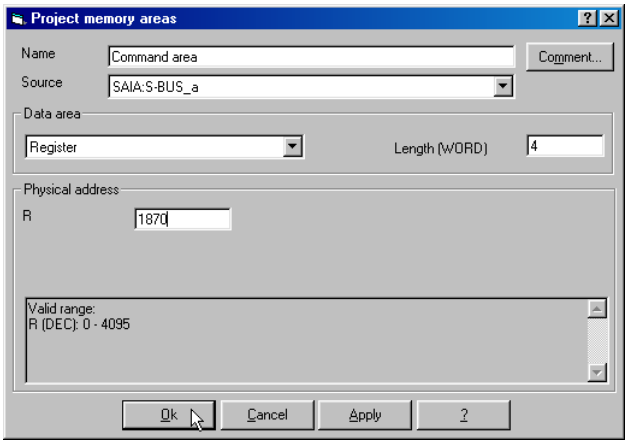



The exchange areas in a standard PCD are one or more registers, for xx7 a DB. The length of the buffer is always indicated in words (1 word = 16 bit), so two words occupy a register. Consult the table below to know what exchange area size has to be defined:

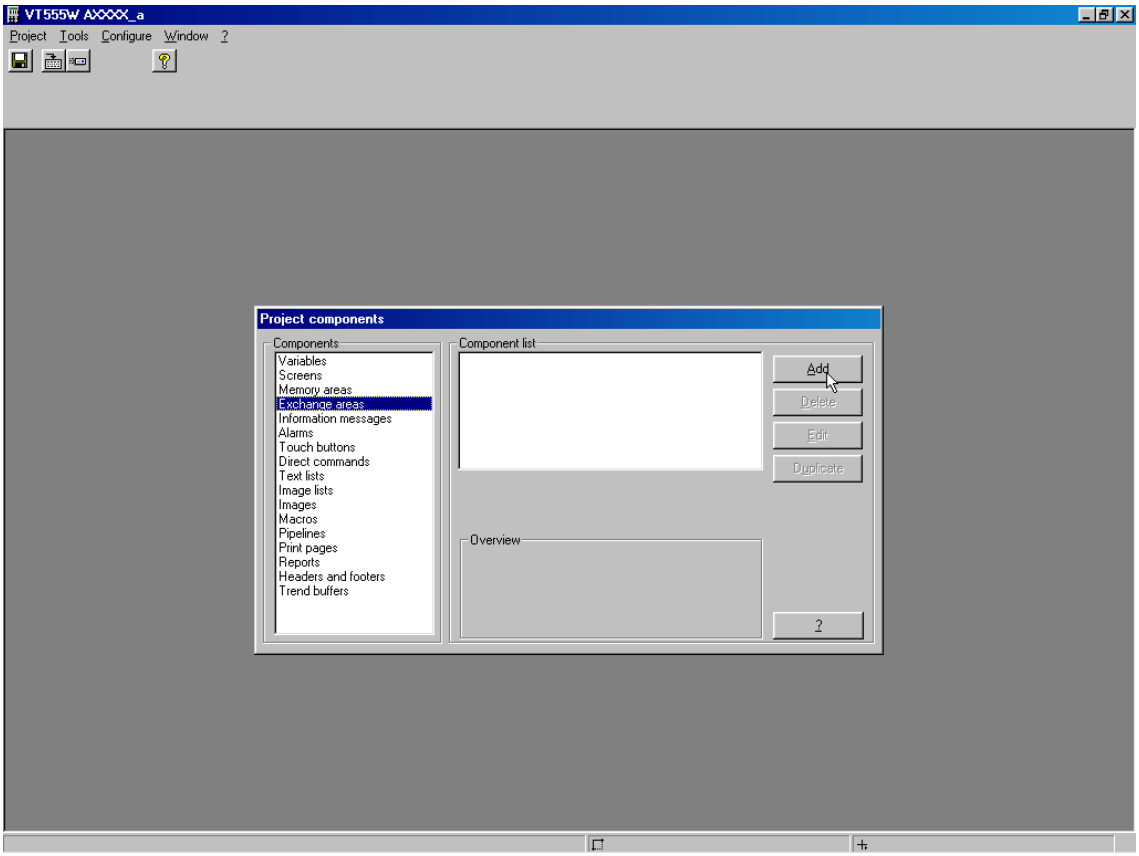
Exchange area lengths required:

| | Type | Length in words | Number of PCD registers occupied |
|---|---|---|---|
| Input areas | Alarm area | Number of alarms divided by 16 | Number of words divided by 2, round up |
| | Information message area | Number of alarms divided by 16 | Number of words divided by 2, round up |
| Status areas | Command answer area | 4 | 2 |
| | Key status area | 4 | 2 |
| | Print status area | 2 | 1 |
| | Recipe status area | 1 | 1 |
| | Terminal status area | 4 | 2 |
| | Trend status area | 1 | 1 |
| Command area | Command area | 4 | 2 |

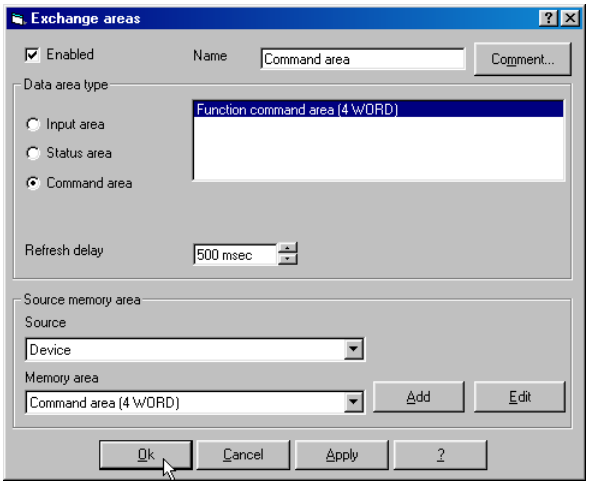
Example for a memory area designed for a command area:



2nd step: define an exchange area that uses the memory area defined before:




Select the desired exchange and memory area, confirm with enter:




Hint for alarm and information message areas: make sure all alarms are in the same area and can be read in one block.

### 9.1) Alarms and information messages

Alarms and information messages are very similar: on events there are messages made available to the user. All alarms have to be acknowledged and stay visible until the bit assigned to it goes low and the alarm is acknowledged. Information messages disappear automatically when the bit assigned to it goes back to low, whether the user has seen the message or not.

If messages are present this is shown to the user as follows:

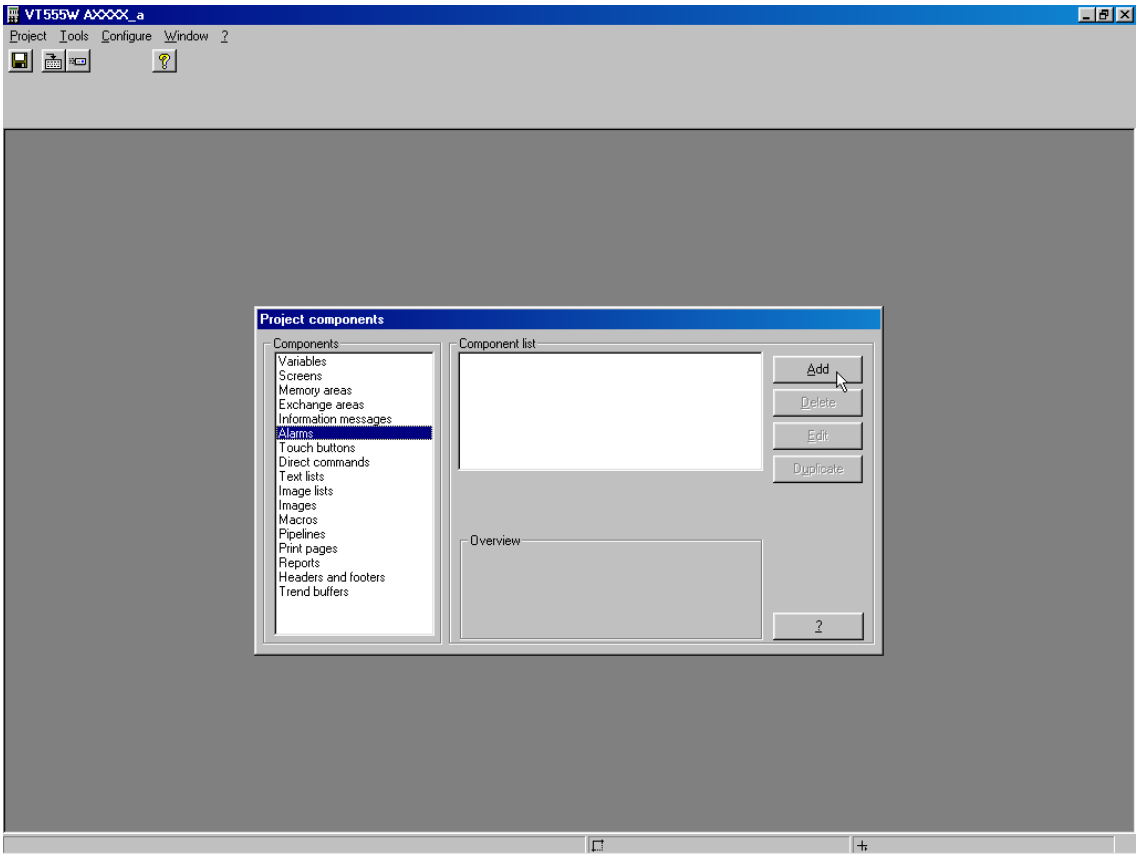
- touchscreens: a symbol is automatically displayed at a defineable position
- terminals with function keys: a LED is activated

Both typs of terminals can be configured in a way that the alarm page is automatically shown on each new alarm. This automatism is not possible for information messages.

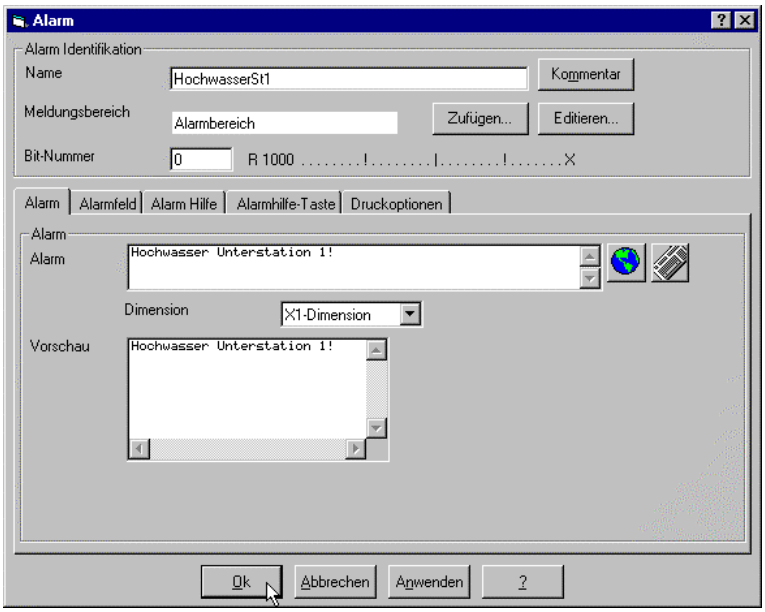
To each message a bit in the alarm exchange area is assigned. More than one bit can be activated at a time. The user can scroll through the different messages.

Define alarms: First of all an alarm exchange area has to be defined as described in the precedent chapter.

The second step is to define the messages, select the component alarms, click on Add:



Define the text and relevant bit




Alarms and messages already defined can be duplicated. Doing this VTWIN automatically assings the first free bit to the copy, so it is recommended to define the alarms starting from the least significant bit of the exchange area.

### 9.2) Status area

Status areas inform about the status of the terminal. Using these areas the PLC can for insance see whether a key is pressed on the terminal and what LED's are on.

The online help and the hardware manual explain in detail the status areas of the different terminals. As an example here the first word of the key status area:

| BIT | WORD 0 | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | Shift | VT50 | VT60 | VT150W | VT160W | VT170W | VT190W | VT300W | VT310W | VT320W | VT330W |
| 0 | | HAND SHAKE | HAND SHAKE | HAND SHAKE | HAND SHAKE | HAND SHAKE | HAND SHAKE | HAND SHAKE | HAND SHAKE | HAND SHAKE | HAND SHAKE |
| 1 | | F1 Esc | Help Esc | Clr Esc | Clr Esc | Clr Esc | Clr Esc | Clr Esc | Clr Esc | Clr Esc | Esc |
| 2 | | F2 | -- | ◄ | ◄ | ◄ | ◄ | ◄ | ◄ | ◄ | ◄ |
| 3 | | F3 | F3 | ► | ► | ► | ► | ► | ► | ► | ► |
| 4 | | F4 | F4 Enter | Enter | Enter | ► | ► | ► | ► | ► | ► |
| 5 | | F5 | F4 Enter | -- | -- | Enter | Enter | Enter | Enter | Enter | Enter |
| 6 | | Info | F2 | PgDn | PgDn | PgDn | PgDn | PgDn | PgDn | PgDn | PgDn |
| 7 | | Help | F1 | PgUp | PgUp | PgUp | PgUp | PgUp | PgUp | PgUp | PgUp |
| 8 | | F1 Esc | F1 | PgDn | PgDn | Info | Info | Info | Info | Info | Info |
| 9 | | F2 | F2 | PgUp | PgUp | Help | Help | Help | Help | Help | Help |
| 10 | | F3 | F3 | Clr Esc | Clr Esc | Alarm | Alarm | Alarm | Alarm | Hist Alarm | Hist Alarm |
| 11 | | F4 | F4 Enter | · ± Space | · ± Space | Ack All Ack | Ack All Ack | Ack All Ack | Ack All Ack | Ack All Ack | Ack All Ack |
| 12 | | F5 | -- | · ± Space | · ± Space | · ± Space | · ± Space | · ± Space | · ± Space | · ± Space | Clr |
| 13 | | Help | Help Esc | Help | Help | | | | | | |
| 14 | | Info | Help Esc | Info | Info | -- | -- | Spare | Spare | -- | Spare |
| 15 | | -- | -- | -- | -- | shift | shift | shift | shift | shift | Shift |

-- : not used

Please note that two different key status areas can be defined

- **Key status area:**
  If a key is pressed, the relevant bit is set and stays high until another key is pressed or it is deleted by the PLC. There is a handshake bit that is always set when one of the keys is pressed. The PLC can reset this bit after having treated the key press and can so detect new key presses.
- **Key status area (real time):**
  If a key is pressed, the relevant bit is high as long as the key is pressed and returns to low when the key is released. As long as any key is pressed the handshake bit is high. As the exchange area is updated in fixed intervals there is a risk of not seeing short keypesses using this status area.

### 9.3) Command area

The command area can be used by the PLC to give orders to the terminal. The following procedure is recommended:

- check whether the last command was executed (command word 0 equals to 0 if command executed, the terminal resets it after executing a command)
- write the parameters in word 1 to 3
- write the command in word 0
- optionally check whether the command has been executed
- optionally check the command answer area if relevant, not all commands create an answer

The online help and the hardware manual explain in detail what commands are available for what terminal.

The exectution of a command can take up to 1 second even when the communication line is fine, as the command area is interpreted in fixed intervals of 500 ms as default. It can as well happen that the terminal is busy with a higher priority task. The terminal can only handle one order at a time, there is no buffer. An ideal tool to ensure a proper treatment of several commands is the graphtec editor of PG4 or PG5. If the procedure given above is respected, no commands are lost.

Example: Command 16, Set the clock of the terminal with the parameters given

What the command area looks like:

| Word | Content |
|---|---|
| 0 | 16 (command to set the clock) |
| 1 | HHMM (Hours and minutes, BCD coded ) |
| 2 | SS (seconds, BCD coded) |
| 3 | Not used for this command |

# 10) Data storage and backup

To be able to modifiy a terminal project the source code is always required (same as for PCD). The upload and download features only enable to copy projects from one terminal to another without being able to modify them.

**Creating backups on a regular basis is of utmost importance!**

If you loose days of work because of a crash this is your fault, even though crashes are very rare with VTWIN.

Some hints:

- zip the projects, as they are database files the difference is impressive
- seperate the data from the programs, choose different folders

# 11) FBoxes for PG4 and PG5

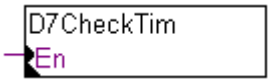
In most of the cases the PLC program does only contain an assignation of the serial port for the dialogue with the terminal.

For advanced features the following FBoxes can avoid programming in instruction list and speed up the programming since they are handy and well tested.

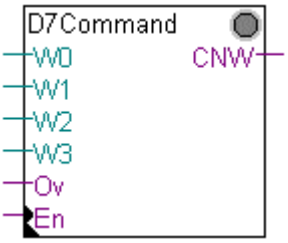
**On the other hand you must note that this is not an official FBox library from SAIA, but freeware.** You can copy and use it without problems, but we don't gurantee the maintenance of them. It could for instance happen that they are no longer updated with future PG versions.

This is only a short presentation, consult the online help of the FBoxes to know more about them:

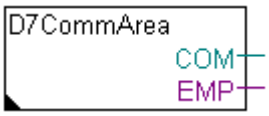
**D7CheckTim** Survey a range of registers for valid time information
(useful together with heavac clocks)




**D7Command** Write a command in the command area

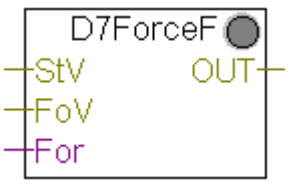



**D7CommArea** Show the number of the current command in the command area

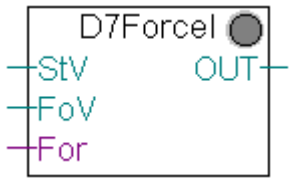



**D7ForceB** Force a binary signal

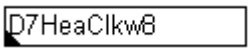

**D7ForceF** Force a floating value



**D7ForceI** Force an integer value

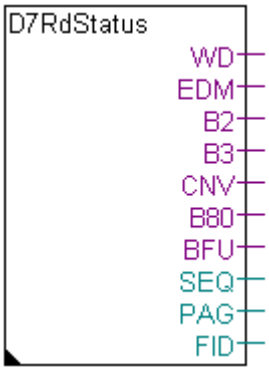



**D7HeaClkw8** Make the parameters of a HeaClkw8 FBox available to the terminal

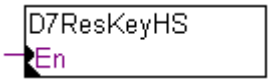



**D7Key** Show the state of a key

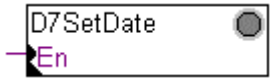

**D7RdStatus** Read the status area



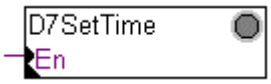

**D7ResKeyHS** Reset the handshake bit in the key status area




**D7SetDate** Write the date from the PCD to the terminal




**D7SetTime** Write the time from the PCD to the terminal




The FBoxes are installed using setup extra files in the PG4 or PG5 group of your start menu.