Connecting UniOP to the EBERLE PLS 514 PLC

1. Introduction

UniOP can be connected to the EBERLE PLS 514 PLC using an EBERLE K43 Communication Module. The panel and the PLC will communicate using the 3964R protocol and the RK512 command interpreter. The connection can be either RS-232 (V24) or Current Loop 20 mA (TTY).
The K43 Communication Module requires a specific, and rather complex, setup for proper operation. The necessary information is available in the EBERLE technical literature and will be only referred to in this technical note.

2. Communication Setup

This chapter describes how to set-up a demonstration system for the communication between the EBERLE PLC and the UniOP.

2.1 UniOP Communication Setup

The K43 module supports communication using the 3964R protocol and the RK512 Command Interpreter.
In the UniOP Designer you will have to choose the protocol 'Siemens 3964R' from the list of available communication drivers.
The communication parameters should be left unchanged to the default values of 9600 bauds, Even parity, 1 stop bit. Communication parameters should be changed only if the K43 communication configuration is different from the default one.
The EBERLE PLC has a particular and specific addressing mode, the K43 module however emulates 4 Data Blocks according to the Simatic S5 format. You can access from UniOP only those 4 DB and nothing else. It is <u>not possible</u> to access directly Merkers, Inputs, Outputs or any other data type.
For the connection to the K43 in RS-232, the cable CA109 should be used. Max length of the cable is 12 meters.

2.2 PLC Communication Setup

The K43 Communication Module must be properly initialized and configured for correct operation.

The configuration information is detailed described in the manual 'Handbuch K41, K43' (German version, part number: 050 046 000 200; version: 0103-9524).

The same manual contains also the detailed description of a Demo Test program, which is the simplest way to get a system up and running with a UniOP panel. The PLC Demo program is available in the diskette ESM 4036 K41, K43 Demodiskette (in German).

There are a few important things to point out for a successful installation.

?? you must initialize the K43 for operation. Note that addressing of the K43 in the PLC program depends on the physical position of the module in the rack. The protocol 3964R with RK512 interpreter must be selected.

?

?? the type of communication RS-232 or Current Loop 20 mA must be selected; the two standards are commonly referred respectively as V24 and TTY in the EBERLE literature.

?? the proper communication parameters must be selected. The suggested default values are 9600 bauds, Even parity and 1 stop bit.

?

?? the emulated Data Blocks must be configured. The suggested configuration is to have 4 Data Blocks with numbers 20, 21, 22 and 23 (Hexadecimal). Length of each block is 256 words (DW). Note that UniOP is using decimal notation for DB numbers instead; you will have to use DB numbers 32, 33, 34 and 35 in the UniOP Designer. An example is listed below:

The initialization program for the K43 module must be included in the user's PLC program to add to the final application the capability to support the K43 module.

Here is a listing of part of the demo program mentioned previously:

```
{       Definition des Kommandodigits Digit 3 der Ebene 0}
{       ---------------------------------------------------}

{       Einstellen der Datenubertragung zum K 43 auf }
{       Initialisierung-Modus}

{       Digit 0xx3}

{   Bit   3   2   1   0}
{       _________________}
{       !   !   !   !   !}
{       ! X ! X ! X ! X !}
{       !   !   !   !   !}
{       -----------------}
{         !   !   !   !}
{         !   }
{         !   !   !   0  = Kein SW-Reset }
{         !   1   !   1  = SW-Reset}
{         !   }
{         !   0          = K 43 Betriebsart passiv}
{         !   1          = K 43 sendet uber V.24 TTY}
{         0              = Initialisierung}
{         1              = Datenaustausch mit K 43 aktiv}

{       Eingabe der Konstanten 00 in das Digit 0xx3 }
{       d.h. Initialisierung des K 43 }
```

```
{               ohne SW-Reset}
{               Protokoll initialisieren gem„á Ebene 0, }
{               Digit 0...2 und Ebene 1, Digit 0...7}

LD      %K 0
=D      KOMMAND

{       Initialisierung uber Ebene 0 Digit 0,1 und 2:}
{       ---------------------------------------------}
{       Digit 0xx0}

{   Bit   3   2   1   0}
{       _________________}
{       !   !   !   !   !}
{       ! X ! X ! X ! X !}
{       !   !   !   !   !}
{       -----------------}
{         !   !   !   !}
{         !   }
{         !   0   0   0  =   110 Baud }
{         !   0   0   1  =   300 Baud }
{         !   0   1   0  =   600 Baud }
{         !   0   1   1  =  1200 Baud }
{         !   1   0   0  =  2400 Baud }
{         !   1   0   1  =  4800 Baud }
{         !   1   1   0  =  9600 Baud }
{         !   1   1   1  = 19200 Baud       }

{         0               = Datenformat 7 Bit }
{         1               = Datenformat 8 Bit }

{       Eingabe der Konstanten 14 in das Digit 0xx0 }
{       d.h. 9600 Baud, Datenformat 8 Bit}

LD      %K 14
=D      Data00


{       Digit 0xx1}
{       ----------}
{ }
{   Bit   3   2   1   0}
{       _________________}
{       !   !   !   !   !}
{       ! X ! X ! X ! X !}
{       !   !   !   !   !}
{       -----------------}
{         !   !   !   !}
{         !   !   !}
{         !   !   !   0  = kein Parity-Bit, keine Pr•fung  (none)}
{         !   !   !   1  = Parit„t wie Bit 1.1 und 1.2}
{         !   !   }
{         !   0   0   1  = Pr•fung auf ungerade Parit„t   (odd)}
{         !   1   0   1  = Pr•fung auf gerade Parit„t     (even)}
{         !   }
{         !   0   1   1  = Parity-Bit immer auf "1" gesetzt (mark)}
{         !                keine Pr•fung                  }
{         !   1   1   1  = Parity-Bit immer a. "0" gesetzt (space)}
{         !                keine Pr•fung                  }
```

```
{         0              =  1 Stop Bit}
{         1              =  2 Stop Bit}

{       Eingabe der Konstanten 05 in das Digit 0xx1 }
{       d.h. gerade Parit„t, 1 Stop Bit }
{ }
LD      %K 5
=D      Data01


{       DIGIT 0xx2}
{       ----------}

{   Bit   3   2   1   0}
{       _________________}
{       !   !   !   !   !}
{       ! X ! X ! X ! X !}
{       !   !   !   !   !}
{       -----------------}
{         !   !   !   !}
{         !   !   !}
{         !   !   !   0  = RTS/CTS eingeschaltet}
{         !   !   !   1  = RTS/CTS ausgeschaltet}
{         !   !   }
{         !   !   0      = V 24}
{         !   !   1      = TTY}
{         !   }
{         !   0          = aktives Senden mit niedriger Priorit„t}
{         !   1          = aktives Senden mit hoher Priorit„t}

{         0              = Protokoll 3964 aktiv}
{         1              = Protokoll 3964 R aktiv}

{       Eingabe der Konstanten 09 in das Digit 0xx2 }
{       d.h. Handshake XON/XOFF, V24-Schnittstelle, }
{       niedrige Priorit„t, Protokoll 3964 R aktiv.}

LD      %K 9
=D      Data02

{       Initialisierung der Ebene 1}
{       ===========================}

{       Die Initialisierung der Datenbausteine erfolgt im }
{       hexadezimalen Code. Bei der Angabe des entsprechenden}
{       Zeichens wird jede Stelle einzeln eingegeben. }
{       D.h. zwei Halbbytes ergeben ein Zeichen und m•ssen demzu-}
{       folge in zwei Digits eingetragen werden.}
{  }
{       Initialisierung des 0. Datenbausteine}
{       -------------------------------------}

{       Digit 1xx0  niederwertiges Halbbyte des 0. Datenbausteins}
{       Digit 1xx1  h"herwertiges  Halbbyte des 0. Datenbausteins}
{           }
{  -->  z.B. 1. Datenbaustein auf 32 dez. => 20 hex. }
{ }
LD      %K 0
```

```
=D      Data10
LD      %K 2
=D      Data11
```