

Connection of the operator controls of a handheld terminal to a Siemens PLC via S7 protocol

(sg, October 2020)

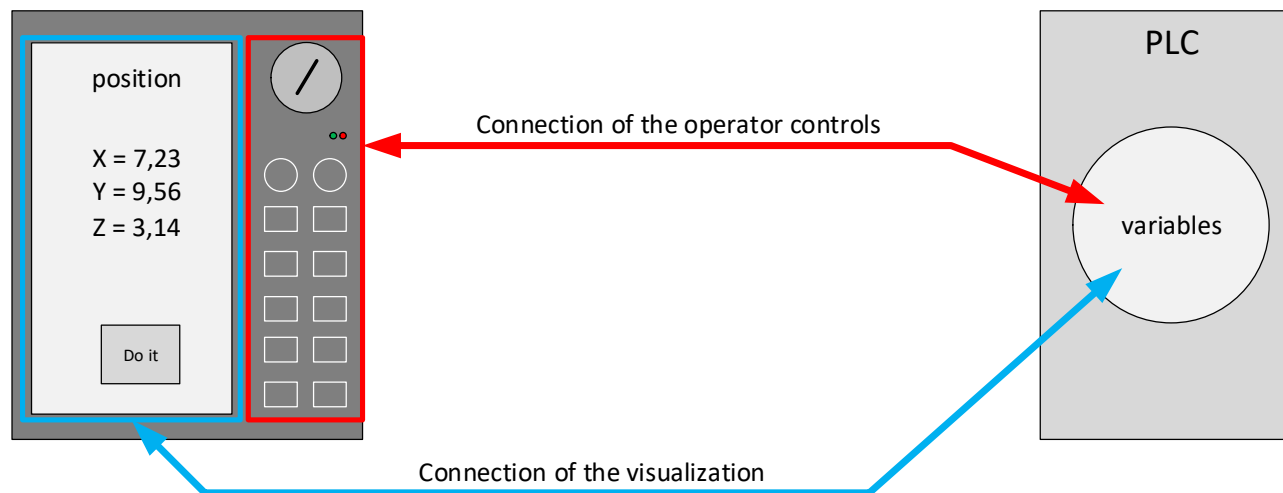
Test mit MachineSoftware V 1.4.2 (aritfactory products/KEBA/Windows-Production)

KEBA[®]

Automation by innovation.

Communication channels between handheld terminal and a PLC

- Connection of the operator controls (= installation elements)
- Connection of the visualization
- Both work on the variables of the PLC
- Connections can use the same or different protocols



Training

For connecting the operator controls to a Siemens PLC via S7 protocol

Preconditions

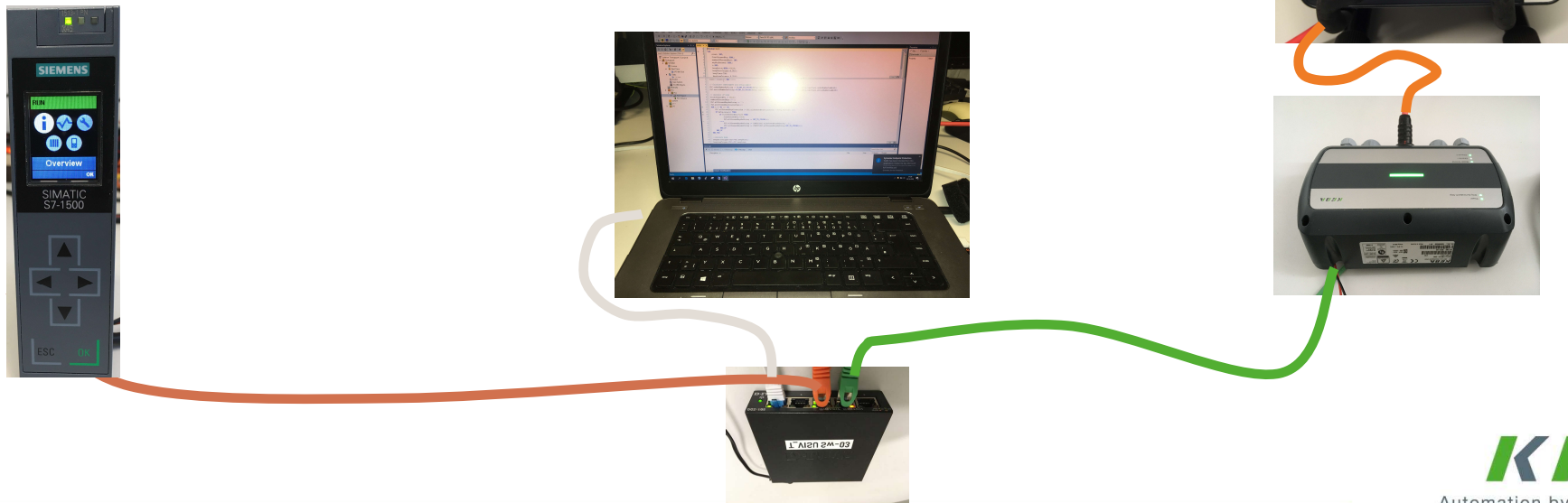
- Basic knowledge of Siemens PLCs and TIA Portal as IDE
- Basic knowledge of KEBA handheld terminals

KEBA[®]

Automation by innovation.

Structure

- Siemens PLC S7-300/400 or S7-1200/1500
- Handheld terminal with Windows
- Developer PC with TIA Portal IDE
- All three devices must be connected and available in the network via Ethernet



PLC project

Create variables in a data block

Variables are addressed in the handheld terminal via DB number and offset

Keys or leds can also be used as single variables

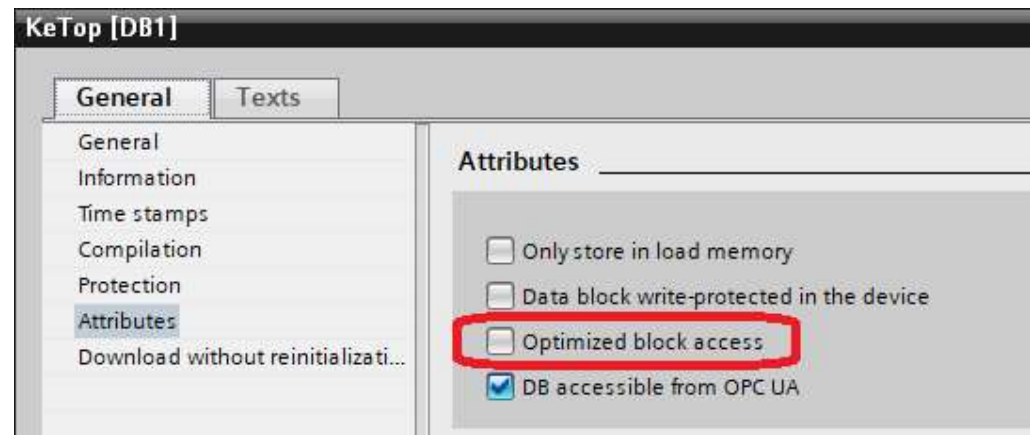
Example here a T150 with a key switch

	Name	Data type	Offset
1	Static		
2	alive	UInt	0.0
3	connectionBoxAddress	UInt	2.0
4	typePlate	Array[0..5] of UInt	4.0
5	buzzer	UInt	16.0
6	leds	Array[0..15] of Byte	18.0
7	keys	Array[0..127] of Bool	34.0
8	keyCount	Array[0..127] of W...	50.0
9	keySwitch	Byte	306.0
10	screenBrightness	UInt	308.0
11	screenTimeOut	UInt	310.0

PLC project – DB properties

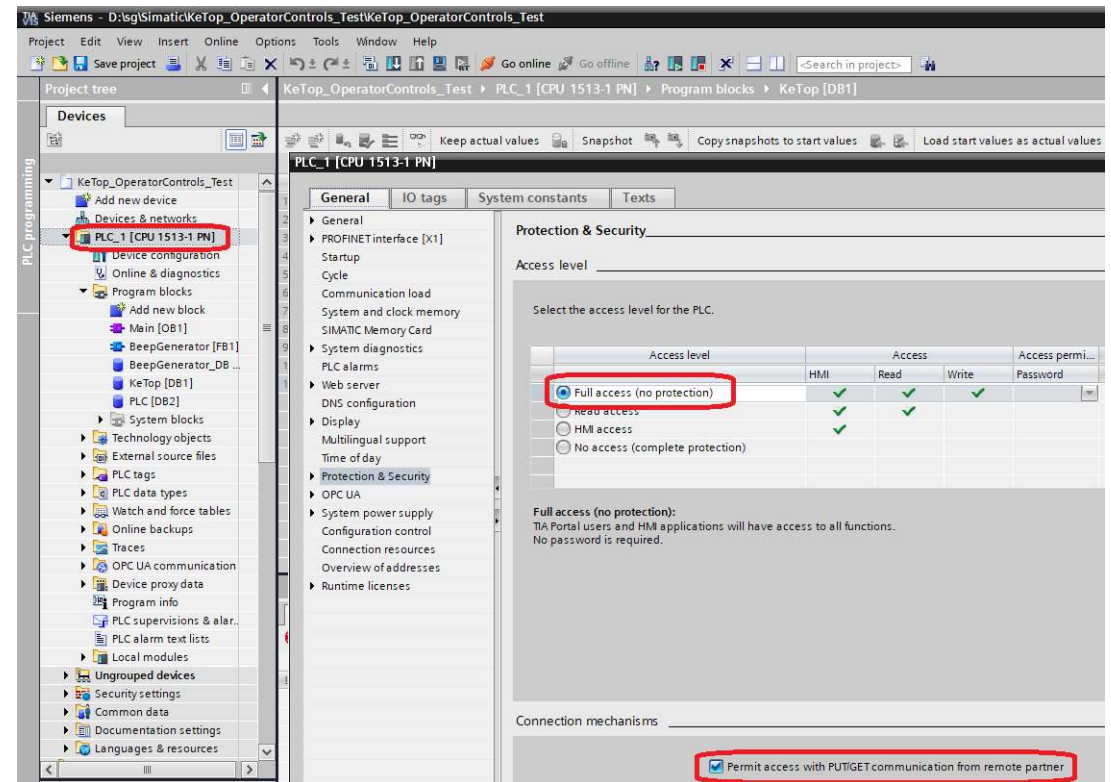
„Optimized block access“ has to be deactivated

HINT: If no offsets are displayed in the data block, either the column is switched invisible or the optimized block access is activated. The column is filled with valid values only after successful compilation.



PLC project – PLC protection & security

- Access to the PLC must be allowed, since the variables must be written to from outside.
- In addition, access via so-called PUT/GET communication must be allowed from outside, since the interface uses exactly this mechanism.

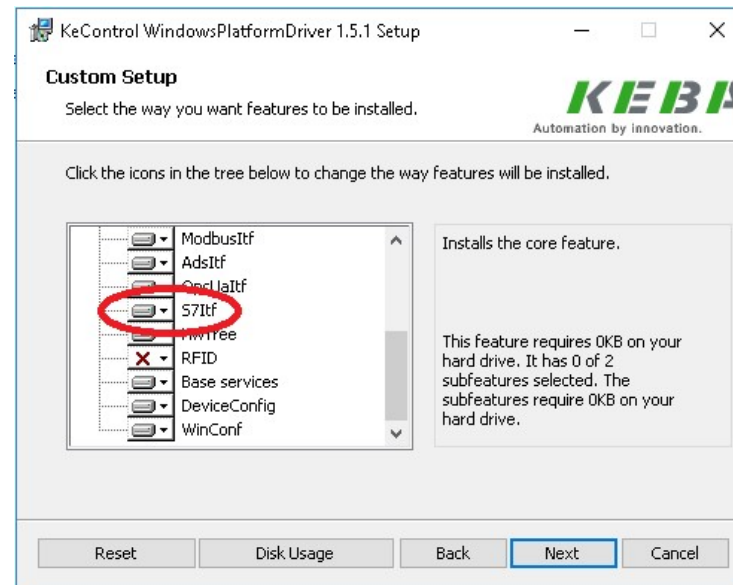


Preconditions handheld terminal

S7 interface must be installed on the handheld terminal. The installation takes place in the WindowsPlatformDriverSetup.

The WindowsPlatformDriverSetup can be found under „Programs and Features“.

Only one interface should be used at a time.



Configuration handheld terminal

```
s7itf.cfg - Editor
File Edit Format View Help
;
; #####
; ## S7 Master (Client) configuration section ##
; #####
[S7]
enable = true

[S7.Connection]
ip = "192.168.214.2" ; IP-address of controller
rack = 0 ; PLC rack number (0..7)
slot = 0 ; PLC slot number (0..31)
type = 2 ; 0-PG(programming console), 1-OP(Siemens HMI panel), 2-BASIC(a
connectionTimeout = 5000 ; milliseconds
receiveTimeout = 1000 ; milliseconds
sendTimeout = 1000 ; milliseconds

[S7.Mapping]
alive = "DB(1).0:WORD" ; Keep-alive Counter, 1 x per second
connectionboxaddress = "DB(1).2:WORD" ; connectionboxaddress 1 x at start
typeplt:0 = "DB(1).4:WORD" ; TypePlate Order number low word
typeplt:1 = "DB(1).6:WORD" ; TypePlate Order number high word
typeplt:2 = "DB(1).8:WORD" ; TypePlate Revision
typeplt:3 = "DB(1).10:WORD" ; TypePlate Variant
typeplt:4 = "DB(1).12:WORD" ; TypePlate Serial number low word
typeplt:5 = "DB(1).14:WORD" ; TypePlate Serial number high word

buzzer = "DB(1).16:WORD" ; Buzzer beep duration in ms
leds:0 = "DB(1).18:BYTE" ; LED value: 0-off, 1-on, 2-blink
leds:1 = "DB(1).19:BYTE"

key:0 = "DB(1).34.0:BOOL" ; Key event mapping, max. 128 keys
key:1 = "DB(1).34.1:BOOL"

keycount:0 = "DB(1).50:WORD" ; Key event count mapping, max. 128 key-counts
keycount:1 = "DB(1).52:WORD"

keyswitch = "DB(1).306:BYTE" ; Key-switch position: 0, 1 or 2
screenbrightness = "DB(1).308:WORD" ; brightness of screen 0 - 100
screensavertimeout = "DB(1).310:WORD" ; timeout of screensaver in s
```

- Fill S7itf.cfg with Notepad
- File is located under
C:\ProgramData\KEBA Automation\keview\system
- Describes communication to the PLC
- Describes the connection of operator controls to the variables of the PLC

Configuration handheld terminal / communication

```
[S7]
enable = true

[S7.Connection]
ip = "192.168.214.2"
rack = 0
slot = 0
type = 2
connectionTimeout = 5000
receiveTimeout = 1000
sendTimeout = 1000
```

- Enables the communication via S7
- IP Adresse of the PLC
- Rack, Slot and Type are S7 hardware specific parameters

	Rack	Slot	
S7 300 CPU	0	2	Always
S7 400 CPU	Not fixed	Not fixed	Follow the HW configuration
WinAC CPU	Not fixed	Not fixed	Follow the HW configuration
S7 1200 CPU	0	0	Or 0,1
S7 1500 CPU	0	0	Or 0,1
CP 343	0	0	Or follow the HW configuration
CP 443	Not fixed	Not fixed	Follow the HW configuration

Connection type:

- 0-PG (programming console)
- 1-OP (Siemens HMI panel)
- 2-BASIC (a generic data transfer connection)

Example is a S7-1500

KEBA[®]

Automation by innovation.

Configuration handheld terminal / operator controls

```
[S7.Mapping]
alive = "DB(1).0:WORD"
connectionboxaddress = "DB(1).2:WORD"
typeplt:0 = "DB(1).4:WORD"
typeplt:1 = "DB(1).6:WORD"
typeplt:2 = "DB(1).8:WORD"
typeplt:3 = "DB(1).10:WORD"
typeplt:4 = "DB(1).12:WORD"
typeplt:5 = "DB(1).14:WORD"
```

```
buzzer = "DB(1).16:WORD"
leds:0 = "DB(1).18:BYTE"
```

- Connection of the operator control „alive“ to the variable „KeTop.alive“ on the PLC (KeTop is DB1, offset of alive is 0, datatype is a two byte type = WORD)
„alive“ is an operator control that counts up its value every second and is used to check the communication
- Connection of the operator control „connectionboxaddress“ to the variable „KeTop.connectionBoxAddress“ on the PLC (KeTop is DB1, offset of connectionBoxAddress is 2, datatype is a two byte type = WORD)
- And so on ...

KEBA[®]

Automation by innovation.

Configuration handheld terminal / operator controls

key:0 = "DB(1).34.0:BOOL"
key:1 = "DB(1).34.1:BOOL"

- Connection of the operator control „key:0“ to the variable „KeTop.keys[0]“ on the PLC (KeTop ist DB1, offset of keys[0] ist 34 bit 0, datatype is a one bit type = BOOL)
- With the keys individual bits are addressed, therefore also "34.1".

Configuration handheld terminal / operator controls

Possible configuration entries

[S7.Connection]

```
ip =  
rack =  
slot =  
type =  
connectionTimeout =  
receiveTimeout =
```

[S7.Mapping]

```
typeplt:X = "address of variable"  
alive = "address of variable"  
key:X = "address of variable"  
keycount:X = "address of variable"  
buzzer = "address of variable"  
connectionboxaddress = "address of variable"  
keyswitch = "address of variable"  
enswitch = "address of variable"  
leds:X = "address of variable"  
pushbutton = "address of variable"  
pushbuttonled = "address of variable"  
selswitch = "address of variable"  
screenbrightness = "address of variable"  
rotswitch = "address of variable"  
pushswitch = "address of variable"  
pushswitchled = "address of variable"  
handwheelposition = "address of variable"  
handwheelpressed = "address of variable"
```

KEBA[®]

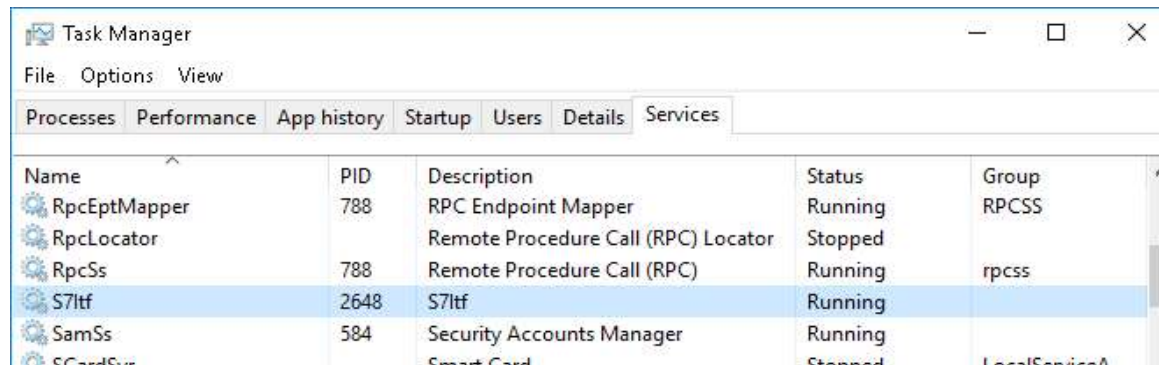
Automation by innovation.

Notes on configuration

- Used addresses of variables (DB numbers and offset) of the PLC must match between project and configuration.
- Operator controls not used in the configuration or operator controls that do not exist do not need to be removed from the configuration or commented out. (For reasons of clarity, however, only elements that are actually used should be configured.)
- For a more detailed description, refer to the document „KeTop S7 protocol“. There, all names of the operator controls and their data types are listed.

Start of communication

- Download and start the PLC application
- Restart the S7 communication to accept the new configuration
 - Either restart the handheld terminal
 - Or Restart the service „S7itf“ in the Taskmanager



- After each change of the configuration, a restart of the communication must be performed, because the configuration is read at startup.

Checking the communication

- Start the debugger of the PLC
- Observe the variable alive – should change their value every second
- Set a led and write the value to the PLC
- Set the buzzer to a value of 1000 – buzzer should sound for one second

The screenshot shows the Siemens SIMATIC Manager interface. The project tree on the left indicates the current context is 'KeTop [DB1]'. The main window displays a table of variables for 'KeTop'.

Name	Data type	Offset	Start value	Monitor value
1	Static			
2	alive	0.0	0	78
3	connectionBoxAddress	2.0	0	42
4	typePlate	4.0		
5	buzzer	16.0	0	0
6	leds	18.0		
7	keys	34.0		
8	keyCount	50.0		
9	keySwitch	306.0	16#0	16#FF
10	screenBrightness	308.0	0	65280
11	screenTimeOut	310.0	0	10

Use of the variables

- In the PLC
- In the visualization

MainView1

Alive	15516
ConBoxAdr	42
Ordernumber	106366
Revision	3
Variant	0
Serialnumber	19541136

Info OpContis (P) OpContis (A)

MainView3

Alive 15703

LED 0 OFF ON BLINK

LED 1 OFF ON BLINK

LED 2 OFF ON BLINK

LED 3 OFF ON BLINK

LED 4 OFF ON BLINK

LED 5 OFF ON BLINK

BEEP short BEEP long

Info OpContis (P) OpContis (A)

Testsupport PLC_1 [CPU 1513-1 PN] Program blocks Main [OB1]

Name	Data type	Default value	Comment
Input			
Initial_Call	Bool		Initial call of this OB
Remanence	Bool		=True, if remanent data are available
Temp			
buzzerReset	Bool		
firstPressedKey	Bool		
keyCode	String		
sum	Int		
numberOfPressedKeys	Int		
i	Int		

```
IF... CASE... FOR... WHILE... OF... TO DO... DO... (*...*) REGION
1 "PLC".myCounter := ("PLC".myCounter + 1) MOD 1000;
2
3 // Calculate ordernumber and serialnumber
4 "PLC".orderNumber := "KeTop".typePlate[1] * 16#10000 + "KeTop".typePlate[0];
5 "PLC".serialNumber := "KeTop".typePlate[5] * 16#10000 + "KeTop".typePlate[4];
6 "PLC".orderNumberAsString := ULINT_TO_STRING(IN := "PLC".orderNumber);
7 "PLC".orderNumberAsString := RIGHT(IN := "PLC".orderNumberAsString, L := LEN("PLC".orderNumberAsString) - 1);
8 "PLC".serialNumberAsString := ULINT_TO_STRING(IN := "PLC".serialNumber);
9 "PLC".serialNumberAsString := RIGHT(IN := "PLC".serialNumberAsString, L := LEN("PLC".serialNumberAsString) - 1);
10
11 // analysis of keys
12 #firstPressedKey := FALSE;
13 #numberOfPressedKeys := 0;
14 "PLC".allPressedKeysAsString := '';
15 #sum:=0;
16 FOR #i := 0 TO 127 DO
17   #sum := #sum + "KeTop".keyCount[#i];
18   IF "KeTop".keys[#i] THEN
19     IF #firstPressedKey = FALSE THEN
20       #firstPressedKey := TRUE;
21       "PLC".allPressedKeysAsString := INT_TO_STRING(#i);
22     ELSE
23       "PLC".allPressedKeysAsString := CONCAT(IN1:="PLC".allPressedKeysAsString, IN2:=',');
```