

# Connection of the operator controls of a handheld terminal to a Beckhoff PLC via ADS

(sg, October 2020)

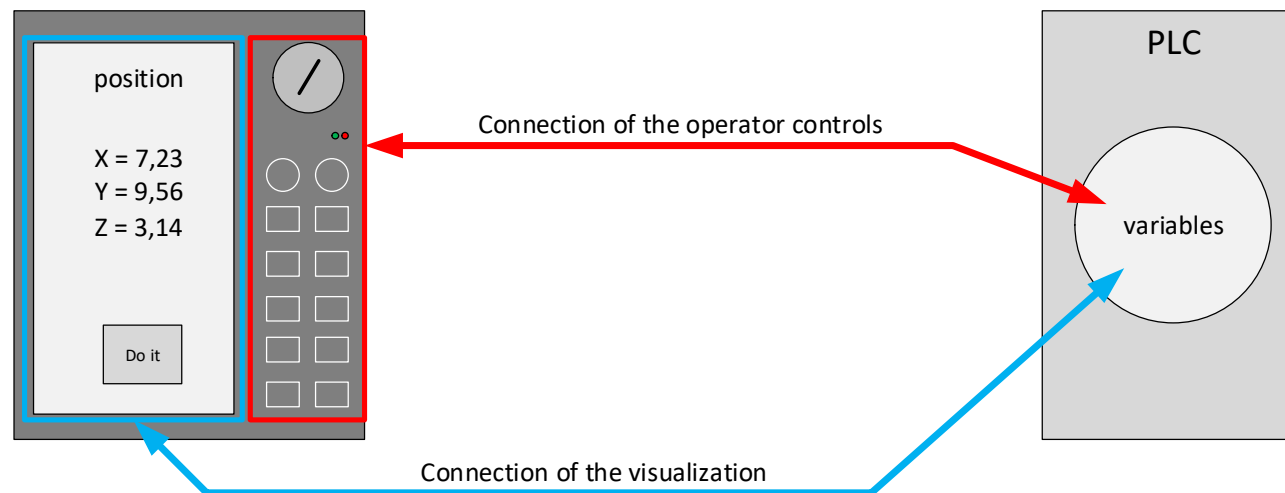
Test with MachineSoftware V 1.4.2 (arifactory products/KEBA/Windows-Production)

**KEBA**<sup>®</sup>

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 Beckhoff PLC via ADS

## Preconditions

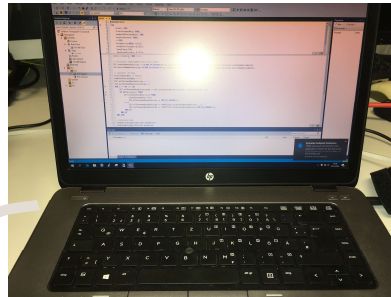
- Basic knowledge of Beckhoff PLCs and Twincat3 as IDE
- Basic knowledge of KEBA handheld terminals

**KEBA**<sup>®</sup>

Automation by innovation.

# Structure

- Beckhoff PLC
- Handheld terminal with Windows
- Developer PC with Twincat3 IDE
- All three devices must be connected and available in the network via Ethernet

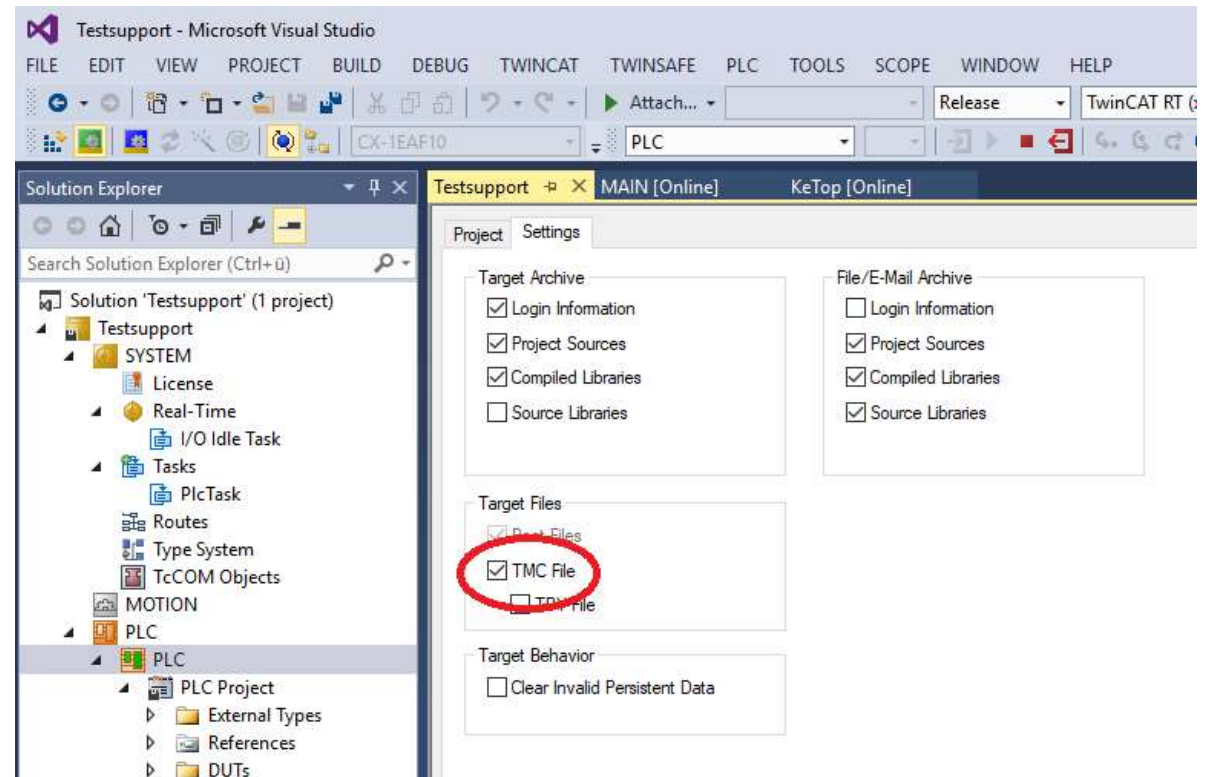


# Preconditions on the PLC

Link between PLC and handheld terminal is done via named variables

Therefore the symbolic names must be published and stored on the controller

Is activated by a tick at „TMC File“ in the settings of the PLC project

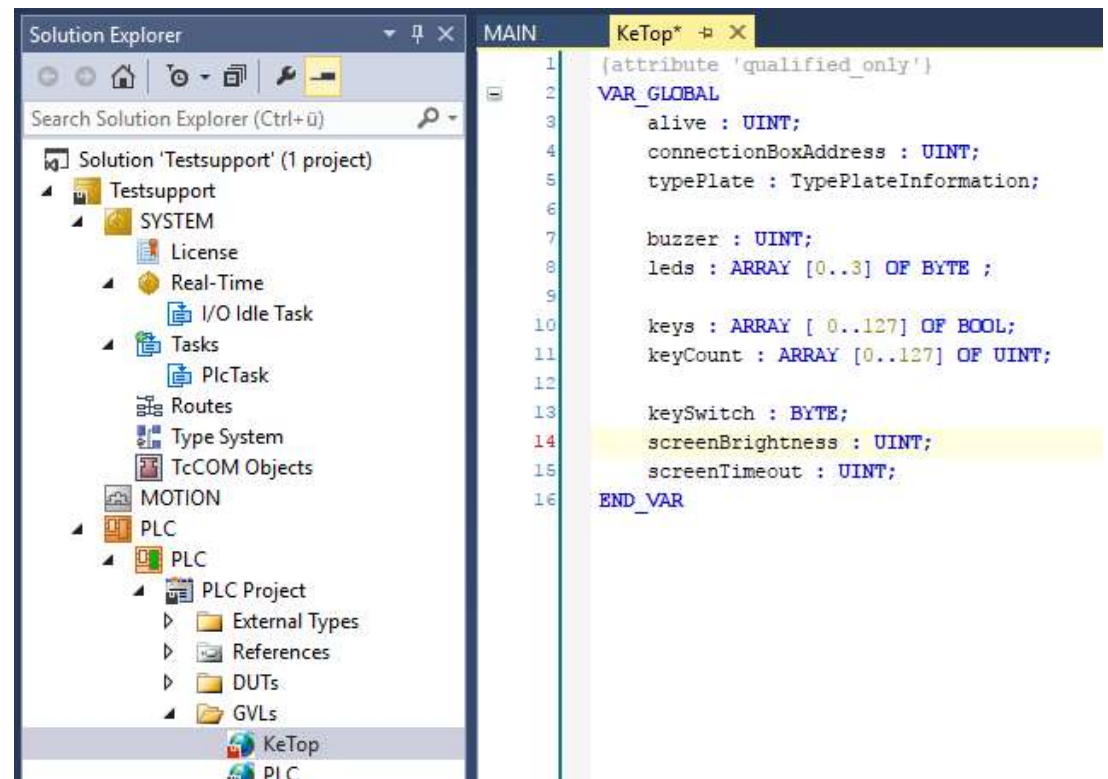


# PLC project

Create variables in a program or in a GVL

Keys or leds can also be used as single variables

*Example here a T150 with a key switch*

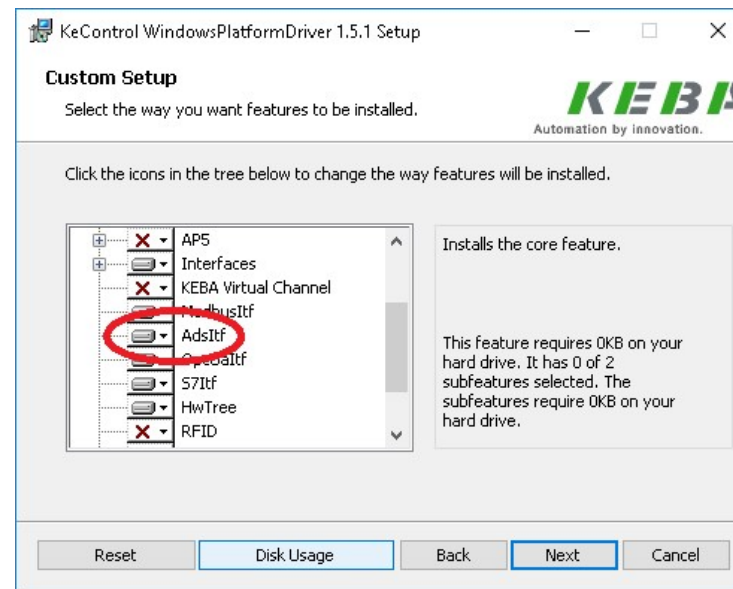


# Preconditions handheld terminal

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

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

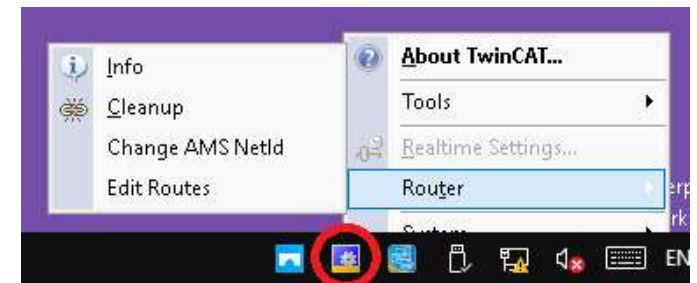
Only one interface should be used at a time.



## Preconditions handheld terminal

For communication with ADS the ADS driver (*TC31-ADS-Setup.3.1.xx.yy*) from Beckhoff should be installed. This can be downloaded free of charge from the Beckhoff website as component TC1000 after registration.

If an additional ADS client is used (e.g. visualization on the handheld terminal), the ADS driver from Beckhoff must be used.



*The recommendation is to always use the ADS driver.*

After the installation the route to the PLC can be defined.



## Preconditions handheld terminal

A route must be added to the PLC. By entering the IP address, the PLC is searched for in the network.

Important:

- The IP address must be used as address info.
- Route must be set for both directions as Static.

Add Route Dialog

Enter Host Name / IP: 192.168.214.133 Refresh Status Broadcast Search

Host Name	Connected	Address	AMS NetId	TwinCAT	OS Version	Comment
CX-1EAF10	X	192.168.21...	192.168.101.1...	3.1.4020	Windows 7	

Route Name (Target): CX-1EAF10 Route Name (Remote): KEBA-PC

AmsNetId: 192.168.101.120.1.1

Transport Type: TCP\_IP

Address Info: 192.168.214.133

Host Name  IP Address

Connection Timeout (s): 5

Target Route:  Project  Static  Temporary

Remote Route:  None  Static  Temporary

Add Route Close

# Configuration handheld terminal

```
adsitf.cfg - Editor
File Edit Format View Help
; #####
; ## ADS Master (Client) configuration section ##
; #####

[ADS]
enable = true ; enable ADS communication between client and AD

[ADS.Connection]
remoteAdsNetId = 192.168.101.120.1.1 ; AdsNetId of controller. More info about AdsNet
remoteIPv4 = 192.168.214.133 ; IP-address of controller
remoteAmsPort = 851 ; AMS Port of controller. More info: https://inf

[ADS.Mapping]
alive = "KeTop.alive" ; Keep-alive Counter, 1 x per second
connectionboxaddress = "KeTop.connectionBoxAddress" ; connectionboxaddress 1 x at start
typeplt:0 = "KeTop.typePlate.orderNumberLowWord" ; TypePlate Order number low word
typeplt:1 = "KeTop.typePlate.orderNumberHighWord" ; TypePlate Order number high word
typeplt:2 = "KeTop.typePlate.revision" ; TypePlate Revision
typeplt:3 = "KeTop.typePlate.variant" ; TypePlate Variant
typeplt:4 = "KeTop.typePlate.serialNumberLowWord" ; TypePlate Serial number low word
typeplt:5 = "KeTop.typePlate.serialNumberHighWord" ; TypePlate Serial number high word

buzzer = "KeTop.buzzer" ; Buzzer beep duration in ms
leds:0 = "KeTop.leds[0]" ; LED value: 0-off, 1-on, 2-blink
leds:1 = "KeTop.leds[1]"

key:0 = "KeTop.keys[0]" ; Key event mapping, max. 128 keys
key:1 = "KeTop.keys[1]"

keycount:0 = "KeTop.keyCount[0]" ; Key event count mapping, max. 128 key-counts
keycount:1 = "KeTop.keyCount[1]"

keyswitch = "KeTop.keySwitch" ; Key-switch position: 0, 1 or 2
screenbrightness = "KeTop.screenBrightness" ; brightness of screen 0 - 100
screensavertimeout = "KeTop.screenTimeout" ; timeout of screensaver in s
```

- Fill ADSitf.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

- ```
[ADS]
enable = true
```
- Enables the communication via ADS
- ```
[ADS.Connection]
remoteAdsNetId = 192.168.101.120.1.1
remoteIPv4 = 192.168.214.133
remoteAmsPort = 851
```
- AdsNetId of the PLC
  - IP Address of the PLC
  - AmsPort of the PLC
    - 801 for TC2
    - 851 for TC3

# Configuration handheld terminal / operator controls

[ADS.Mapping]

```
alive = "KeTop.alive"  
connectionboxaddress = "KeTop.connectionBoxAddress"  
typeplt:0 = "KeTop.typePlate.orderNumberLowWord"  
typeplt:1 = "KeTop.typePlate.orderNumberHighWord"  
typeplt:2 = "KeTop.typePlate.revision"  
typeplt:3 = "KeTop.typePlate.variant"  
typeplt:4 = "KeTop.typePlate.serialNumberLowWord"  
typeplt:5 = "KeTop.typePlate.serialNumberHighWord"  
  
buzzer = "KeTop.buzzer"  
leds:0 = "KeTop.leds[0]"
```

- Connection of the operator control „alive“ to the variable „KeTop.alive“ on the PLC  
„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
- And so on ...

# Configuration handheld terminal / operator controls

## Possible configuration entries

*[ADS.Connection]*

```
remoteAdsNetId =  
remoteIPv4 =  
remoteAmsPort =  
timeout =
```

*[ADS.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**<sup>®</sup>

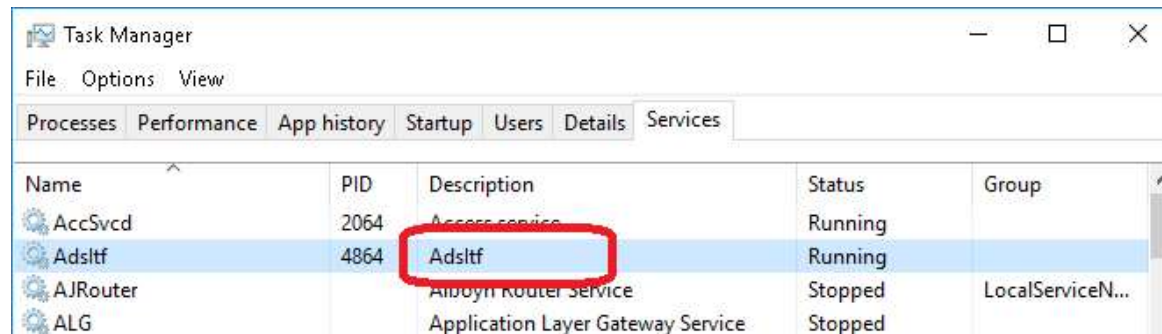
Automation by innovation.

## Notes on configuration

- Variable names including the name of the block used (program name or name of the variable list) must match the 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 ADS 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 ADS communication to accept the new configuration
  - Either restart the handheld terminal
  - Or restart the service „ADSI“ 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

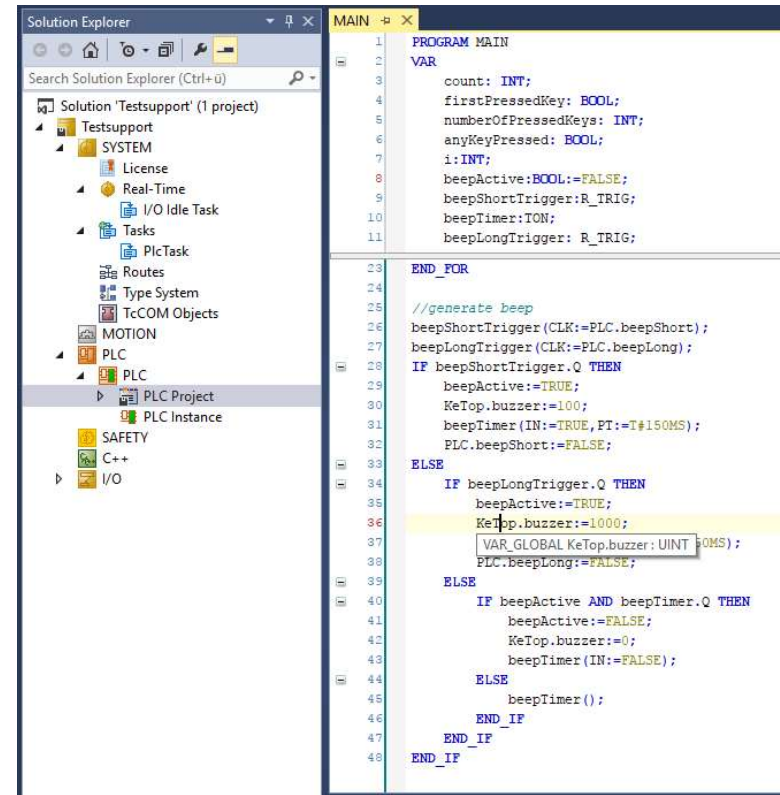
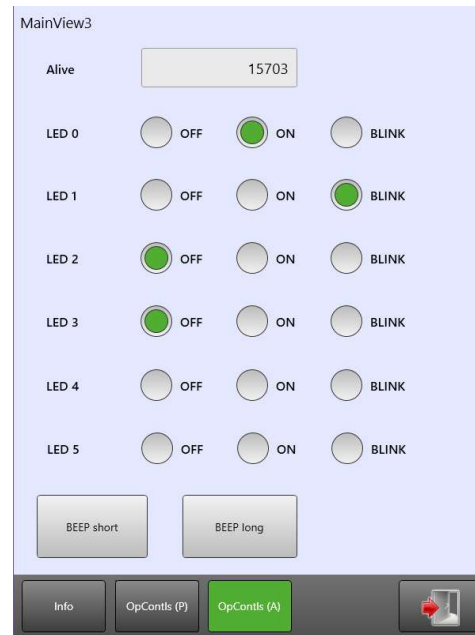
The screenshot shows the Visual Studio interface with the 'MAIN [Online]' window displaying a table of PLC variables. The table has columns for Expression, Type, Value, Prepared value, and Address. Two cells are highlighted with red boxes: the 'Value' column for the 'alive' variable (14880) and the 'Value' column for the 'leds[0]' variable (1).

Expression	Type	Value	Prepared value	Address
alive	UINT	14880		
connectionBoxAddress	UINT	42		
typePlate	TypePlateInformation			
orderNumberLowWord	UINT	40830		
orderNumberHighWord	UINT	1		
revision	UINT	3		
variant	UINT	0		
serialNumberLowWord	UINT	11408		
serialNumberHighWord	UINT	298		
buzzer	UINT	0	500	
leds	ARRAY [0..15] OF B...			
leds[0]	BYTE	2	1	
leds[1]	BYTE	2		



# Use of the variables

- In the PLC
- In the visualization



```
1 PROGRAM MAIN
2 VAR
3   count: INT;
4   firstPressedKey: BOOL;
5   numberOfPressedKeys: INT;
6   anyKeyPressed: BOOL;
7   i: INT;
8   beepActive: BOOL:=FALSE;
9   beepShortTrigger: R_TRIG;
10  beepTimer: TON;
11  beepLongTrigger: R_TRIG;
12
13
14
15
16
17
18
19
20
21
22
23 END_FOR
24
25 //generate beep
26 beepShortTrigger (CLK:=PLC.beepShort);
27 beepLongTrigger (CLK:=PLC.beepLong);
28 IF beepShortTrigger.Q THEN
29   beepActive:=TRUE;
30   KeTop.buzzer:=100;
31   beepTimer (IN:=TRUE,PT:=T#150MS);
32   PLC.beepShort:=FALSE;
33 ELSE
34   IF beepLongTrigger.Q THEN
35     beepActive:=TRUE;
36     KeTop.buzzer:=1000;
37     VAR_GLOBAL KeTop.buzzer: UINT #0MS);
38     PLC.beepLong:=FALSE;
39   ELSE
40     IF beepActive AND beepTimer.Q THEN
41       beepActive:=FALSE;
42       KeTop.buzzer:=0;
43       beepTimer (IN:=FALSE);
44     ELSE
45       beepTimer ();
46     END_IF
47   END_IF
48 END_IF
```