

BECKHOFF

Letzte Änderung: 13.07.2017

PLCStructure von der HMI einlesen und beschreiben (Maschinendaten, Aggregatedaten, etc)

TwinCAT Main application frame - [Form1]

BECKHOFF Message Text TwinCAT 01.06.2010
PLC 15:13:24

PLC File new

RezeptDaten
Masse

Description	Value	ActPlcValue	Min	Max	Unit
JobName	Auftrag1	Auftrag1			
BauteilInfoText	lange Teil	lange Teil			
Laenge	234.56	234.56			mm
Breite	123.4	123.4			mm
Dicke	18	18			mm
Betriebsart	7	7			
Dicke	4564	4564			mm
VeloAblegeRt	46	46			m/min
VeloDosierRt	6	6			m/min
VeloAuslaufRt	3	3			m/min
LangsamesAufnehmen	5	5			1/0
AblegenAufRollen	4	4			1/0
Vakuumreduzierung	4	4			1/0
Ansaugzeit	23	23			ms
DatenAktiv	2	2			1/0

Administrator Level: Administrator - Application started



F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 F11 F12

PLC Struktur lesen Laden Speichern unter Import Übertrage zur SPS Speichern zurück

- Das Formular zur Verwaltung von „SPS Strukturen“ befindet sich im Modul „Beckhoff.App.PlcStructure.dll“ im Ordner „Plugins“ der Applikation. Dieses Formular kann über den Menumanager (auch mehrfach) eingebunden werden (vgl. Dokumentation Menumanager).



- Die Settings werden pro Instanz von „FormPlcStructure“ automatisch beim ersten Aufruf des Formulars generiert. Sie befinden sich in den Settings in PlcStructure.Instanzname.

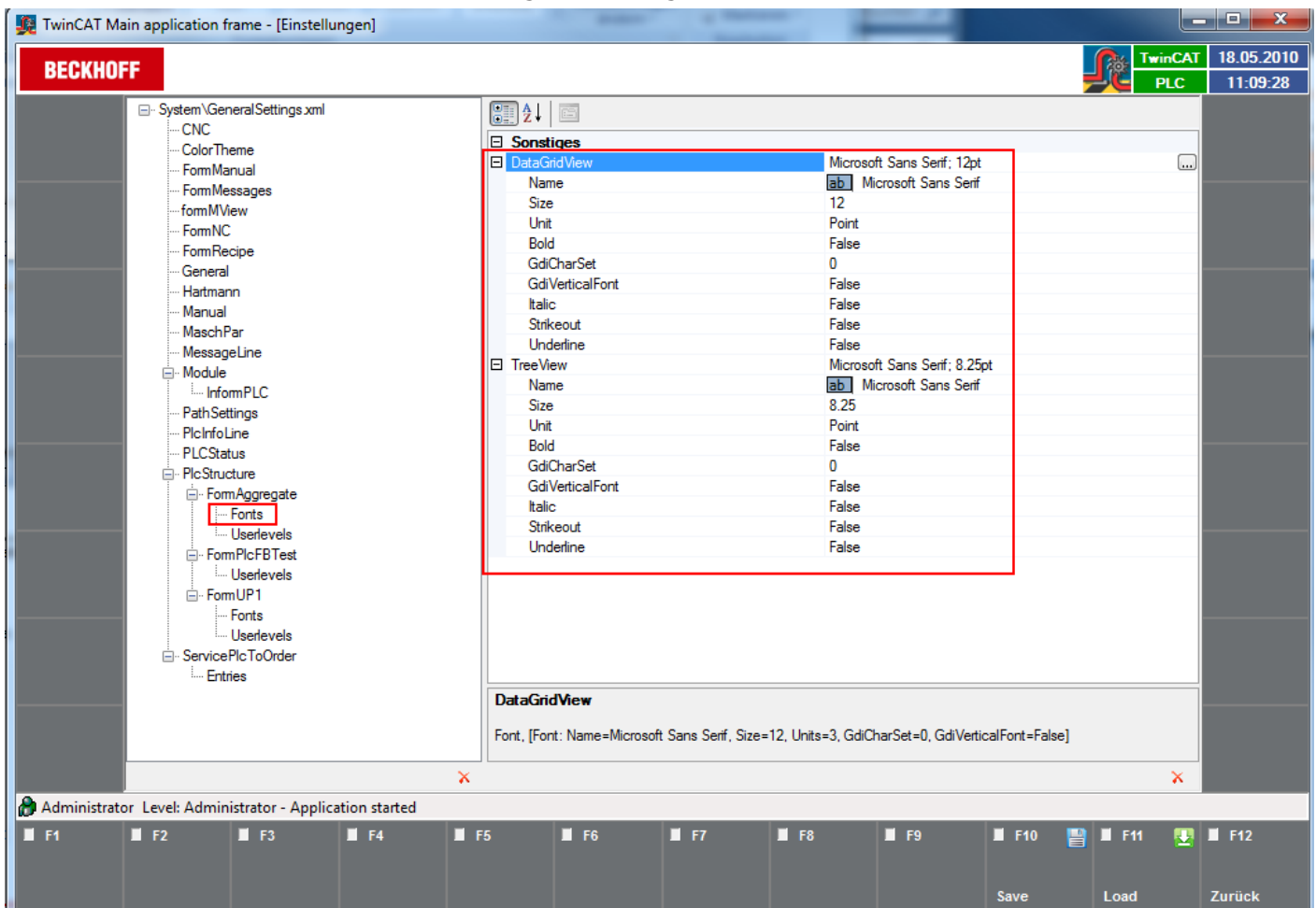
AutoWriteOnVariableChange	True
AutoWritePlcAtStartup	False
BackColorEdit_NOT_Allowed	 211; 211; 211
BackColorEditAllowed	 255; 255; 224
BitMaskElement	2
EndingOfFilename	Rez
EndingOfSelctableFilesCombobox	Rez
Folder	.\Rezeptdaten
FolderPictures	.\Bitmap\PlcStruct
InformPLCStartWriteVar	GvlApp.HMI.Rezept.StartWriteFromHMI
InformPLCWrittenFilenameVar	GvlApp.HMI.Rezept.ActFilename
InformPLCWrittenVar	GvlApp.HMI.Rezept.NewData
InputValueInvisiblePlcVar	GvlApp.HMI.Rezept.InputValueInvisible
LastUsedFile	.\Rezeptdaten\new.Rez
OpenCloseSpecialDialogs	True
PlcValueVisible	True
PrintLogo	Customer\Logo.bmp
Read Always (independent from @1)	False
ShowSpecialControlForEditValues	True
TreeAutoExpand	True
TreeViewRootPLCVar	GvlApp.HMI.Rezept
TriggerReadFromPlc_BoolVariable	GvlApp.HMI.Rezept.TriggerReadFromPLC
TriggerWriteToPlc_BoolVariable	GvlApp.HMI.Rezept.TriggerWriteToPLC
Visible_InputValue	True
Visible_Max	True
Visible_Min	True
Visible_Unit	True

- AutoWriteOnVariableChange**
 True = Bei jeder Änderung eines Eintrags in der Oberfläche, wird diese sofort zur SPS übertragen und im aktuellen File gespeichert
- AutoWritePlcAtStartup:**
 True = die komplette Struktur wird beim ersten Laden des Formulars oder beim Neustart der SPS automatisch zur SPS geschrieben.
 False = zur SPS wird nur auf Kommando geschrieben.
- AutoXMLBackup:**
 True = Bei jedem Speichern der Daten auf dem Datenträger wird zusätzlich automatisch eine XML Datei mit den Daten erzeugt und im Ordner „AutoXMLBackupFolder“ abgelegt. Dabei wird der Dateiname aus Datum, Uhrzeit, Benutzername und Benutzerlevel gebildet.
- AutoXMLBackupFolder:**
 Ordner in dem die automatisch generierten XML Dateien abgelegt werden.
- BackColor...** Farbeinstellung für das Datagrid, in dem die Daten angezeigt werden.
- EndingOfFilename:** Dateinamensendung der Datenfiles.

- **EndingOfSelectableFilesCombobox:** Dateiendung für die Comboboxanzeige für Dateinamen.
- **Folder:** Verzeichnis, in dem die Datenfiles abgelegt werden
- **InformPlcStartWriteVar:** Bool Variable in der SPS, die TRUE geschrieben wird, direkt bevor die Struktur geschrieben wird.
- **InformPlcWrittenVar:** Bool Variable in der SPS, die TRUE geschrieben wird, nachdem die Struktur geschrieben wurde.
- **InformPlcWrittenFilenameVar:** String Variable in der SPS, in der der Dateiname des zuletzt geschriebenen Datensatzes abgelegt wird
- **PLCValueVisible:** True = in einer weiteren Spalte werden die aktuellen SPS Variableninhalte „OnChange“ angezeigt. Die Spalte „ActPlcValue“ wird nur angezeigt und die Verbindung zu den angezeigten Variablen wird nur aufgebaut, wenn hier „TRUE“ eingestellt ist.
- **ShowSpecialControlForEditValues:** Beim Betreten eines Editierfeldes wird ein spezielles auf den Typ des Feldes abgestimmtes Control zum Ändern geladen. Dies ist für ControlPanels nützlich, die über keine Tastatur verfügen und nur mit Touch bedient werden können.
- **TreeAutoExpand:** Beim erstmaligen Anzeigen des Treeview, wird dieser komplett aufgeklappt dargestellt.
- **LastUsedFile:** Hier wird automatisch das zuletzt zur SPS geschriebene Datenfile eingetragen.
- **OpenCloseSpecialDialogs:** TRUE: Es werden spezielle Dialoge zum Laden und Speichern benutzt, in denen keine Dateien bearbeitet werden können(Löschen, Umbenennen, etc)
- **TreeViewRootPLCVar:** Wurzel der SPS Struktur, die eingelesen oder beschrieben wird.
- **TriggerReadFromPlc_BoolVariable:** Falls dieser Eintrag ungleich Leerstring ist, wird bei einer steigenden Flanke der BOOL Variablen, der Inhalt der kompletten Struktur von der SPS zur Oberfläche übertragen und überschrieben.
Falls in der SPS Variablen, die in „InformPlcWrittenFilenameVar“ hinterlegt ist ein gültiger Dateiname steht, werden die Daten darin gespeichert.
Nach dem Auslesen und eventuellem Speichern wird die Variable von der Oberfläche aus auf FALSE gesetzt.
- **TriggerWriteToPlc_BoolVariable:** Falls der Eintrag ungleich Leerstring ist, werden bei steigender Flanke alle Daten aus der aktuell geladenen Struktur zur SPS übertragen.
Falls in der Variablen „InformPlcWrittenFilenameVar“ der Dateiname einer existierenden Datei eingetragen ist, so wird der Inhalt dieser Datei an die SPS übertragen und nicht der der aktuell geladenen Struktur.

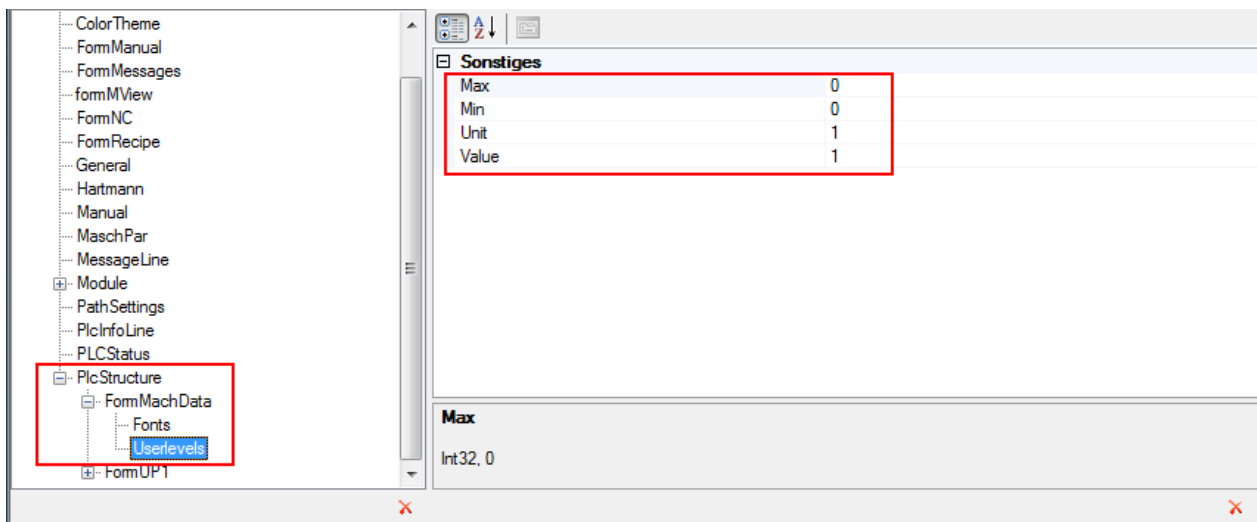
BECKHOFF

- **InputValueInvisiblePlcVar:** SPS Bool Variable, die bestimmt, ob die Werte – Eingabespalte unsichtbar geschaltet wird. Diese Variable wird „auf Änderung“ überwacht und die Spalte „Value“ kann so von der SPS aus unsichtbar/sichtbar geschaltet werden.
Visible_(Max, Min, Unit): Schaltet die Sichtbarkeit der entsprechenden Spalten.
- **BitMaskElement:** Bei Eintrag 0 werden alle mit @1 markierten Elemente beim Lesen der Struktur berücksichtigt.
Bei einem Eintrag <> 0 wird der Wert „bitweise“ mit dem Eintrag @11 von jedem Element logisch verknüpft.
Nur Elemente, bei denen sich ein Wert <> 0 ergibt, werden gelesen und zur Anzeige gebracht.
Durch dieses Verfahren kann gesteuert werden, dass nur bestimmte Teile einer Maschinendatenstruktur genutzt werden.
- In dem Untereintrag „Fonts“ können die Schriftarten der Baumdarstellung links und der Datendarstellung rechts eingestellt werden:



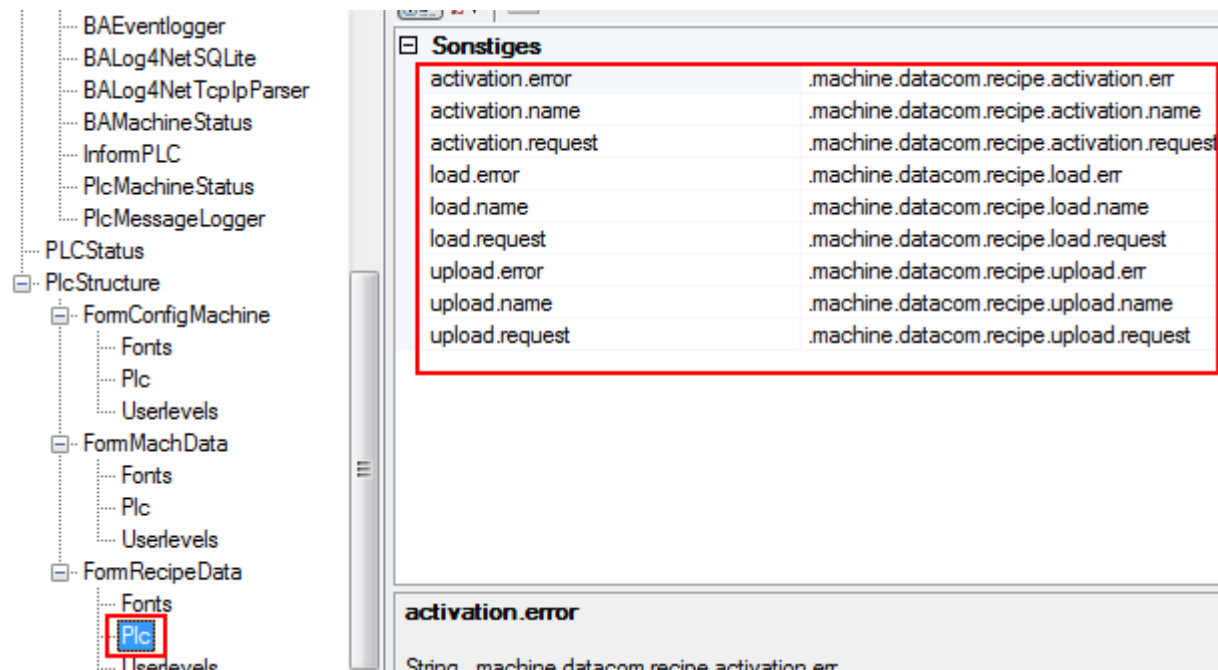
Die jeweiligen „Userlevel“ legen den benötigten Level, der zur Änderung der Inhalte im Datagridview nötig ist, fest. Es gilt folgende Zuordnung:

- 0 = Administrator
- 1 = Supervisor
- 2 = Superuser
- 3 = Standarduser
- 4 = User
- 5 = None



- In diesem Beispiel darf ein User mit Level Administrator Max, Min, Werte und Unit eingeben und ändern.
- Ein User mit Level Supervisor, darf Werte und Unit ändern, aber keine Min oder Max Einstellungen vornehmen.
- Ein User mit Level < Supervisor, darf Max, Min, und Werte nur anschauen.

- Falls im „Untereintrag PLC“ Elemente eingetragen sind, wird eine andere Schnittstelle zur PLC genutzt:



- Die HMI schreibt hier nie unaufgefordert Daten in die SPS sondern schreibt in die Variable „activation.request“ ein TRUE. Daraufhin fordert die SPS über „load.request“ die Daten der HMI an. Die HMI überträgt dann die Daten und setzt „load.request“ auf FALSE sobald sie fertig ist.



- Die Variable *.name“ fordert spezielle Daten an. Die HMI ergänzt den Namen dann automatisch mit der eingetragenen Endung und schickt diesen dann zu SPS. Auch eine Anforderung mittels „activation.request“ sollte mit einem Namen erfolgen, falls es verschiedene Datensätze gibt.
- In der Struktur „*.err“ werden Fehlermeldungen abgelegt. Sie sollte vom folgenden Typ sein:

```

FUNCTION_BLOCK FB_ERRORHDL
VAR_INPUT
    flag:                BOOL;
    id:                  DINT;
    state:               DINT;
    text:                STRING(256);
    fnc:                 STRING(64);
END_VAR
VAR_OUTPUT
END_VAR
VAR
END_VAR

```

- In der SPS können einzulesende Variablen mit einem Kommentar gekennzeichnet werden:

```

VAR
    HMILineMas           : ST_HMIMaschine; (*@1: Maschine *)
END_VAR

```

- Innerhalb des Kommentars werden folgende Tag's ausgewertet:
 - @1:Text = Name des Elements in der HMI. Nur wenn dieser Tag eingetragen ist, wird das Element von der HMI eingelesen. Es wird nur dann in darunter liegenden Strukturen nach weiteren Tag's gesucht, wenn dieser Eintrag vorhanden ist.
 - @11:Byte = Bitmaske, mit der die Elemente gefiltert werden können.
In den Settings (über **BitMaskElement**) kann eingestellt werden, welche Elemente beim Einlesen der Struktur abhängig von der Bitmaske geladen werden sollen.
 - @2:Text = Einheit (Unit) des Elements als String
 - @3:Wert = Standard MIN Wert des Elements (kann in der Oberfläche überschrieben werden)
 - @4:Wert = Standard MAX Wert des Elements (kann in der Oberfläche überschrieben werden)
 - @5:1 = hiermit wird definiert, dass es sich um eine Dateiauswahl handelt, die in der Oberfläche per Combobox auswählbar ist.

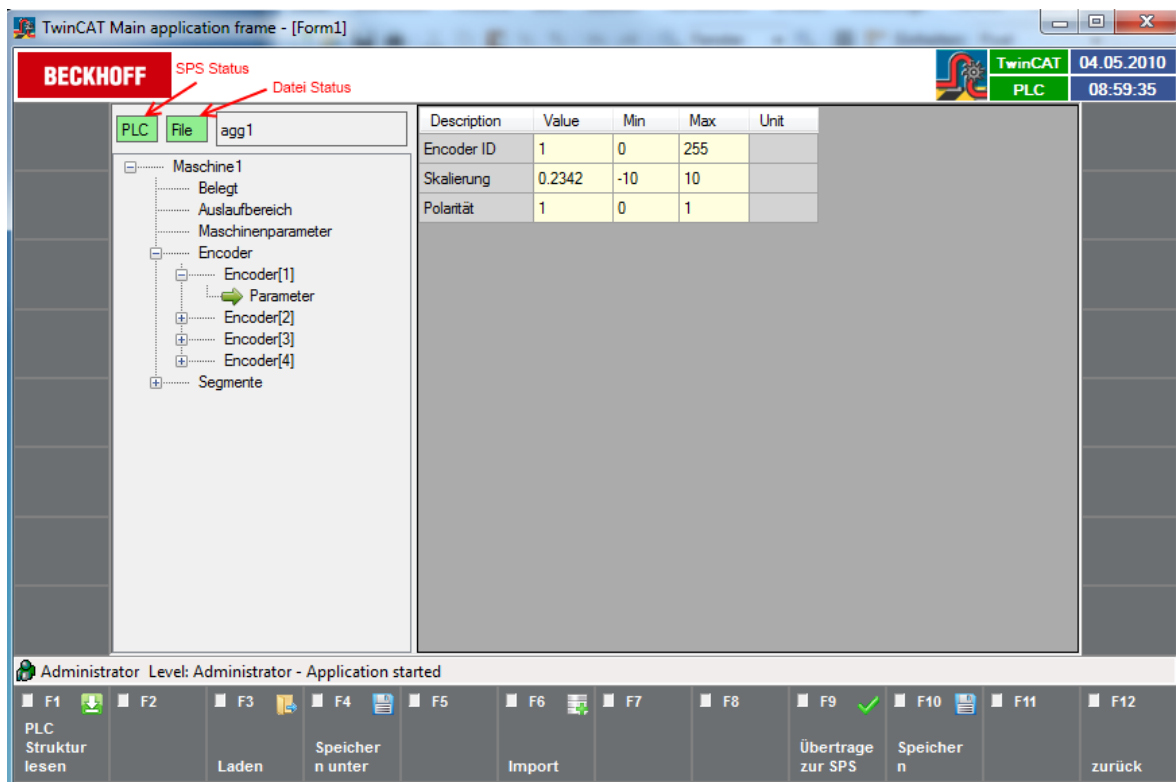


- @51: = Laufwerk für das Verzeichnis der Dateiauswahl Bsp: @51:C
 - @52: = Pfad für das Verzeichnis der Dateiauswahl Bsp: @52:\temp
 - @6:1 = hiermit wird definiert, dass es sich in der darunter liegenden Struktur um „Handfunktionen“ handelt.
- Nach dem Aufruf des Formulars stehen die Funktionen „PLC Struktur lesen“, „Laden“, „Speichern“ und „Import“ zur Verfügung.
 - PLC Struktur lesen:
List die PLC Struktur anhand der Tag's ein und stellt die eingelesene Struktur als Baum dar. Alle schon eingetragenen Variablenwerte gehen verloren und können mit der Funktion „Import“ aus einer vorher gespeicherten Datei importiert werden.
 - Laden:
Eine zuvor erstellte Struktur mit Variableninhalten wird von Festplatte geladen.
 - Speichern:
Die komplette Struktur inklusive Variableninhalten wird auf Festplatte gespeichert.
 - Import:
Aus einer früher gespeicherten Datei werden alle Variableninhalte wieder (soweit möglich) in die aktuelle Struktur importiert. Die aktuelle Struktur wird nicht verändert.

- Mit „Rechtsklick“ auf ein Bauelement oder einer Beschreibung in der Datenanzeige (Datagridview) gelangt man (falls man den Userlevel „Administrator“ hat) in ein Kontextmenu, mit dem man die Texte des angeklickten Elements in der aktuell gewählten Sprache ändern kann:
 - Globaltext ändern: Die Übersetzung des Eintrags wird global geändert, dass heißt alle anderen Elemente mit genau diesem Text werden auch entsprechend übersetzt.
 - Instanztext ändern: Hier wird nur der Text der gerade angeklickten Variablen geändert.

Description	Value	Min	Max	Unit
Encoder ID				
Skalierung				
Polarität				
Simulation aktiv				
Simulationgeschwindigkeit				

Status Anzeige „SPS“ und „Datei“:

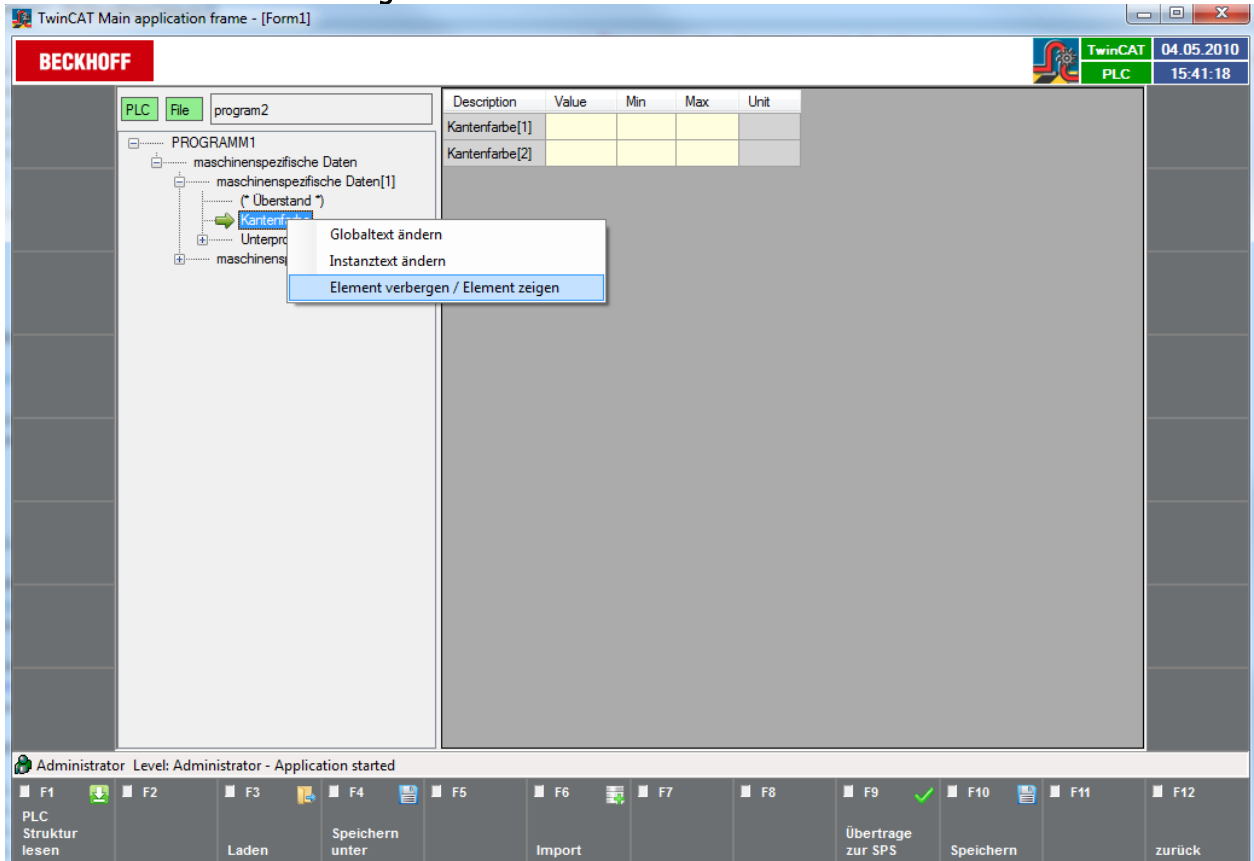


- Die Statusanzeige über dem TreeView hat folgende Bedeutung:
- Im ersten Teil wird die aktuell geladene Datei angezeigt.
- Das Feld „SPS Status“ zeigt den Status zwischen den geladenen Daten und denen in der SPS:
 - Rot: Daten sind nicht übereinstimmend oder es kann nicht 100% gewährleistet werden, dass sie übereinstimmend sind.
 - Grün: Die angezeigten Daten befinden sich genauso in der SPS.
- Das Feld „Datei Status“ zeigt den Status zwischen den geladenen Daten und denen in der Datei:
 - Rot: Daten sind nicht übereinstimmend oder es kann nicht 100% gewährleistet werden, dass sie übereinstimmend sind.
 - Grün: Die angezeigten Daten befinden sich genauso in der gespeicherten Datei.
- Ein Klick auf das jeweilige Feld synchronisiert zwischen SPS bzw. Datei und den angezeigten Daten, d.h. die angezeigten Daten werden geschrieben. Es werden nur die Daten zur SPS geschrieben, die seit dem letzten Schreiben verändert wurde. Die Funktionen „WriteToPlc“ und „Save“ entsprechen der beschriebenen Funktionalität. Sie können mit „CallMethod“ auf beliebige Tasten des Menumanagers gelegt werden.



- Die mit CallMethod anwählbare Funktion „**SaveToFileAndWritePLC**“ speichert in der zuletzt angewählten Datei und schreibt die geänderten Daten zur SPS. Die Funktion „**SaveToFileAndWriteAllPLC**“ schreibt im Gegensatz dazu ALLE Daten in die SPS.
- Die mit „CallMethod“ aufrufbare Methode „**WriteAllDataToPlc**“ schreibt alle Daten zur SPS, unabhängig davon ob sie seit dem letzten Schreiben geändert wurden.
- Mit der Methode „**ExportXML**“ kann der komplette Baum mit allen aktuell eingetragenen Werten in eine XML Datei exportiert werden.
- Mit der Methode „**ImportXML**“ kann der komplette Baum aus einer vorher exportierten XML Datei wieder importiert werden. Dabei werden nur die Werte (Value) und keine anderen Elemente importiert (Min, Max,)
- Die mit callMethod aufrufbaren Methoden „**ReadStructureFromPlcAndConvertDefaultDirectory**“ und „**ReadStructureFromPlcAndConvertDirectory**“ lesen zuerst die Struktur aus der SPS und konvertieren dann alle Dateien im entsprechenden Verzeichnis mit Hilfe der Import Funktion. Achtung! Hierbei werden die Original Dateien überschrieben.
- Mit <Ctrl><Shift> und gleichzeitigem Klick auf eine Taste, kann die Standardbelegung der F-Tasten geladen werden.

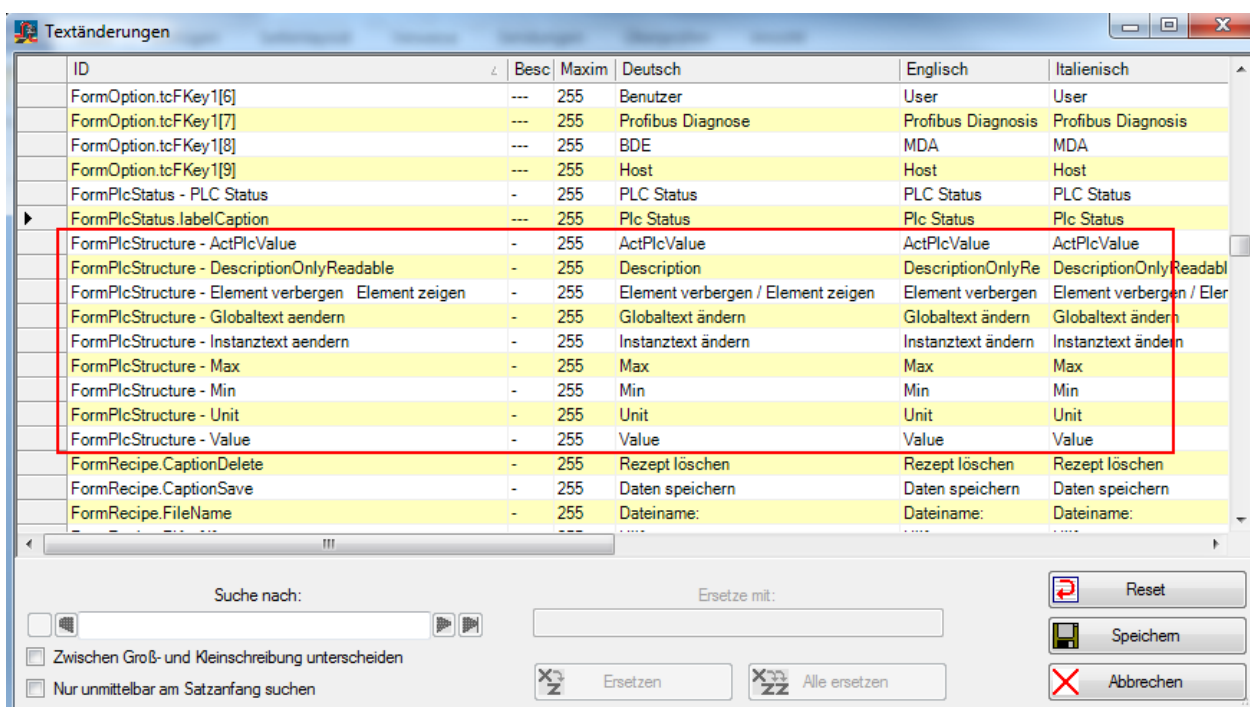
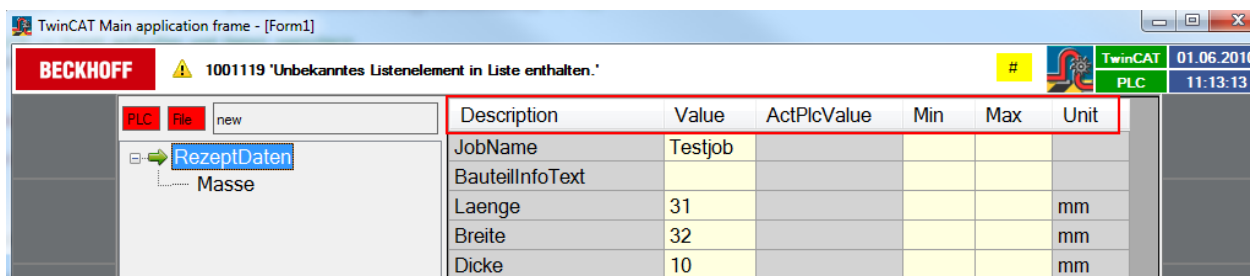
Elemente im Treeview verbergen:



- Mit Userlevel „Administrator“ besteht die Möglichkeit, einzelne Einträge in dem angezeigten Baum und in der Datenanzeige rechts zu verbergen bzw. verborgene Einträge wieder anzuzeigen. Der Status „verborgen/angezeigt“ wird im Kontextmenu des entsprechenden Eintrages verändert.
- In der Datenanzeige öffnet sich das Kontextmenu mit einem Rechtsklick auf die Beschreibung.
- Als User mit Userlevel „Administrator“ sieht man verborgene Einträge. Sie sind durch Kommentarklammern (* *) gekennzeichnet.
- Usern mit Userlevel geringer als „Administrator“ werden die verborgenen Einträge inklusive aller Subeinträge nicht mehr angezeigt.
- Das Kontextmenu kann nur von Usern mit Userlevel „Administrator“ aufgerufen werden.
- HINWEIS: Das Verbergen von Einträgen verringert nicht die Datenmenge, die gespeichert wird. Als nicht sichtbar markierte Elemente werden NICHT zur SPS übertragen.

Sprachumschaltung der Spaltenüberschriften

- Die Sprachumschaltung der Spaltenüberschriften und Kontextmenueinträge erfolgt im LanguageManager. Dort können die Spaltenüberschriften übersetzt werden. Die Indizes lauten FormPlcStructure - „Spaltenname“.



Dateiauswahl mit Hilfe einer Combobox im Datagridview

- Ein Eintrag in der SPS, der mit @5:1 gekennzeichnet ist, wird in der Oberfläche im Datagridview als Combobox zur Dateiauswahl dargestellt.
- Mit einem Rechtsklick auf den Eintrag kann man den Pfad, der in der Combobox dargestellten Dateien, einstellen.
- Das Standard Verzeichnis für die in der Combobox dargestellten Dateien wird in den Settings mit Hilfe des Eintrags „FolderComboboxItems“ festgesetzt.
- Die Endung wird in den Settings mit Hilfe des Eintrages **EndingOfSelectableFilesCombobox** festgelegt.
- In der SPS sollte für den Eintrag eine genügend große String Variable definiert sein.

PLC	File	new	Description	Value	ActPlcValue	Min	Max	Unit
			JobName					
			BauteilInfoText					
Rezepte			Unterprogramm1	25016.utab	25016.utab			
RezeptDaten1			Unterprogramm2	25016.utab	MANUAL.utab			
RezeptDaten2			Laenge	CHANNEL.utab	1			mm
			Breite	MANUAL.utab	2			mm
			Dicke		3			mm
			Betriebsart		3			
			Dicke		5			mm
			VeloAblegeRt		4			m/min
			VeloDosierRt		7			m/min
			VeloAuslaufRt		8			m/min
			LangsamesAufnehmen		4			1/0
			AblegenAufRollen		0			1/0
			Vakuumreduzierung		0			1/0
			Ansaugzeit		0			ms
			DatenAktiv		0			1/0

Handfunktionen (@6:1)

- Handfunktionen werden in der darüber liegenden Struktur mit dem Tag @6:1 definiert. Mit dem Tag @1 wird der Beschreibungstext definiert. Beispiel:

manualFuntction: ST_HMI_Manual; (*@1:Handfunktionen @6:1 *)

- Die Struktur zur Definition von Handfunktionen enthält pro Handfunktionszeile zwei Variablen vom Typ Byte. In der Definition einer Zeile haben die Tags folgende Bedeutung:

- Beispiel:

blowing: BYTE; (* @1:abblasen @2:on:1 @3:Position 1 *)

- @1:Beschreibungstext der Taste
- @2 bis @9:Tastentext:(0=tastend;1=rastend) für die Tasten 1 bis 8
- Die Tasten 1 bis 8 werden von der HMI auf die Bits der Variable geschrieben.
- Die Statusanzeige der Tasten ist über eine byte Variable, deren Name um den String „Status“ erweitert wurde, realisiert. Eine Änderung der Bits in dieser Variablen führt zu einer Farbänderung in der HMI. Beispiel:
blowingStatus: BYTE;
- Die Tastenbeschriftungen können zur Laufzeit durch die SPS mit Hilfe einer Array Variablen vom Typ String, deren Name um den String „Text“ erweitert wurde, geändert werden. Hiermit können zum Beispiel aktuelle Prozesszustände zu Anzeige gebracht werden.

Der Inhalt der Strings wird auf Änderung überwacht (OnChange). Nur String Inhalte, die ungleich einem Leerstring sind, werden berücksichtigt.

blowingText: ARRAY [0..7] OF STRING

Die Deklaration dieser STRING Variablen ist optional, das heißt es gibt keinen Fehler, wenn sie nicht definiert ist.

Die angezeigten Texte werden nicht mit Hilfe der Sprachumschaltung übersetzt.

Falls die Dimension des Arrays bis 8 geht[0..8], wird in „blowingText[8]“ der Text des Beschreibungstext der Taste (vgl. @1) definiert.



○ Komplette Beispielstruktur ST_HMI_Manual:

TYPE ST_HMI_Manual :

STRUCT

(* @1:Beschreibung @2: 1.Spalte Beschreibung:0=tastend, 1 =rastend @3 2.Spalte
Beschreibung:0=tastend, 1 =rastend ...*)

(* Milling *)

milling : BYTE; (* @1:Motor @2:Ein:1 @3:Rückwärts:1 @4:Arbeitspos:0 *)
millingStatus : BYTE;
millingText : ARRAY[0..7] OF STRING;

(* Blowing *)

blowing : BYTE; (* @1:abblasen @2:on:1 @3:Position 1 *)
blowingStatus : BYTE;
blowingText : ARRAY[0..7] OF STRING;

(* Axis *)

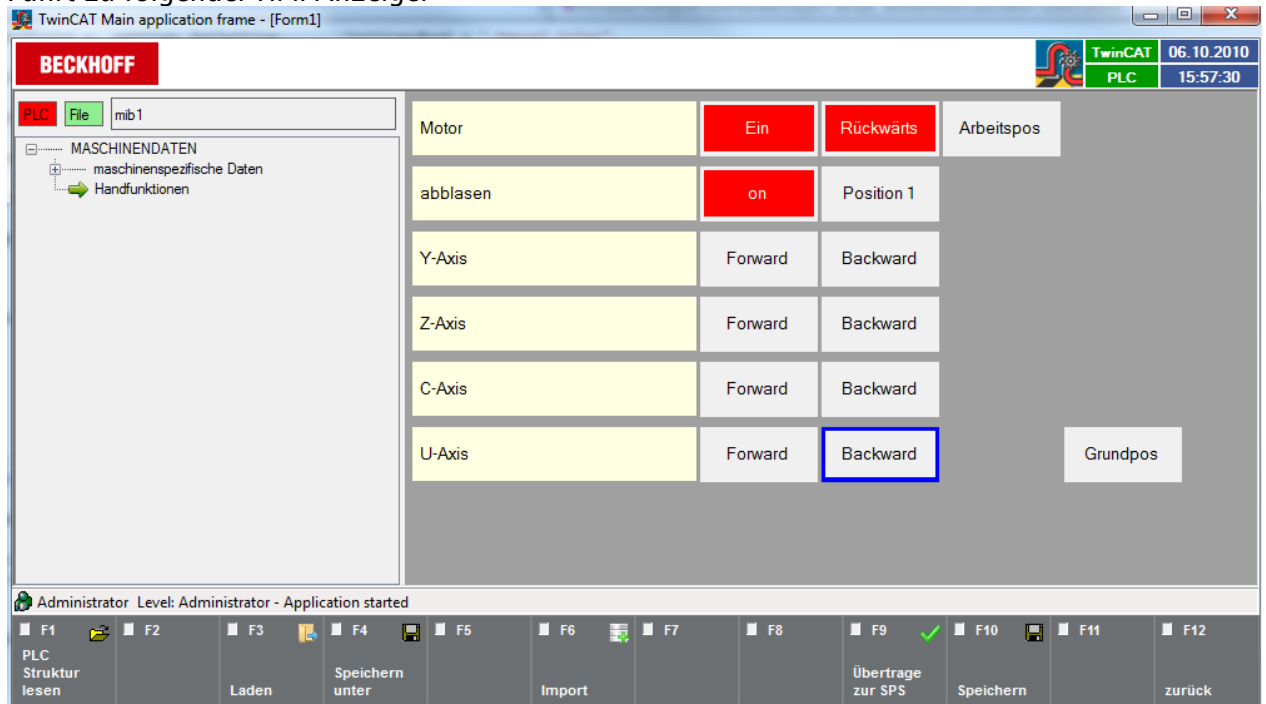
axisY : BYTE; (* @1:Y-Axis @2:Forward:0 @3:Backward:0*)
axisYStatus : BYTE;
axisZ : BYTE; (* @1:Z-Axis @2:Forward:0 @3:Backward:0*)
axisZStatus : BYTE;
axisC : BYTE; (* @1:C-Axis @2:Forward:0 @3:Backward:0*)
axisCStatus : BYTE;
axisU : BYTE; (* @1:U-Axis @2:Forward:0 @3:Backward:0 @5:Grundpos*)
axisUStatus : BYTE;

END_STRUCT

END_TYPE

BECKHOFF

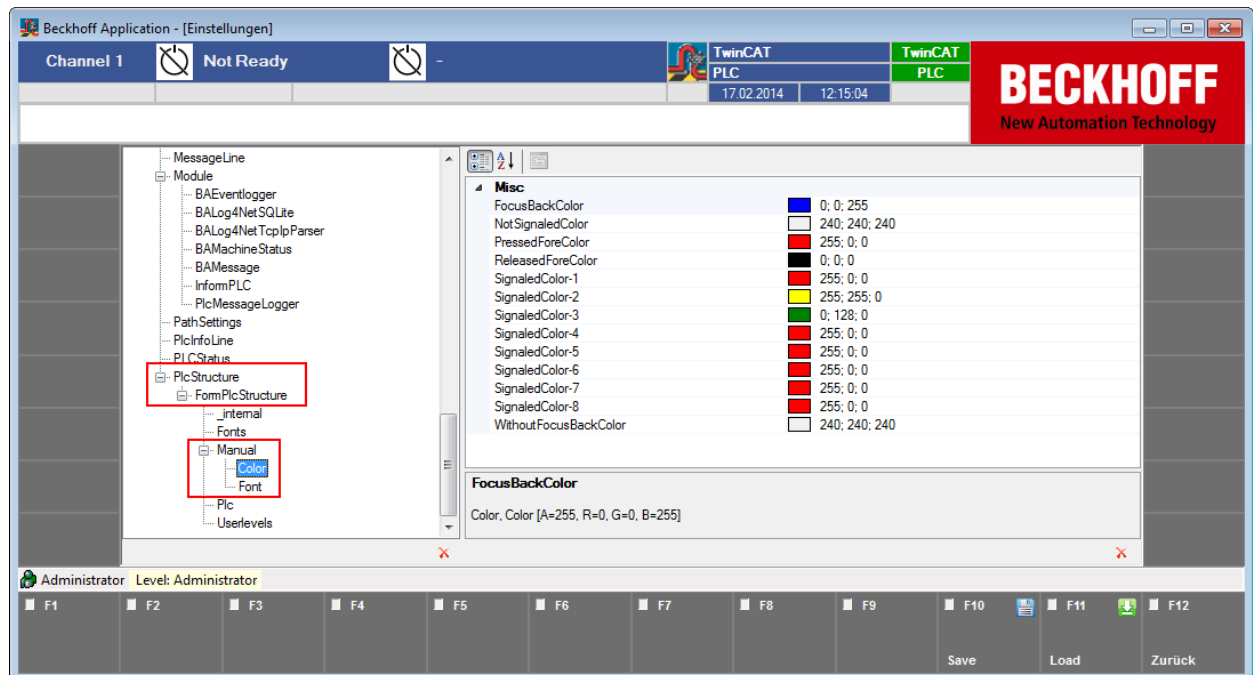
- Führt zu folgender HMI Anzeige:





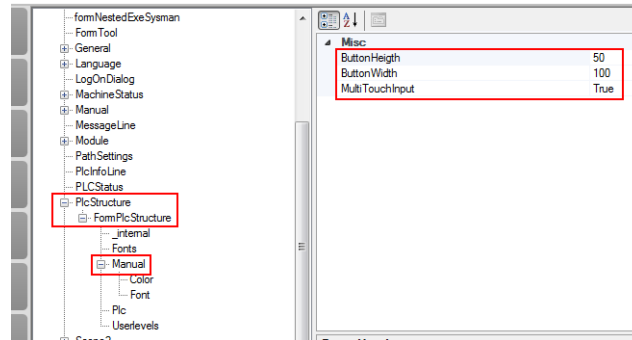
Einstellungen in den Settings für Handfunktionen

- In den Settings können die verschiedenen Farben und Schriften für jede Instanz eingestellt werden.
- Die Farbe für eine Statusanzeige kann Spaltenweise eingestellt werden. Als Standardwert ist „Rot“ eingetragen.



- Die Größe der Buttons in den Handfunktionen kann ebenfalls in den Settings eingestellt werden.

- Der Eintrag „MultiTouchInput“ erlaubt es ein MultiTouch Panel als Eingabegerät zu verwenden, bei dem die Fingereingabe aktiviert ist.



- Ist der Eintrag „MultiTouchInput“ auf TRUE eingestellt, besteht die Möglichkeit statt eines Textes ein vordefiniertes Icon aus einem Satz von 14 Icons anzuzeigen. Dazu muss statt des Textes die Buchstaben IMG gefolgt von der Nummer des Icons eingetragen werden.

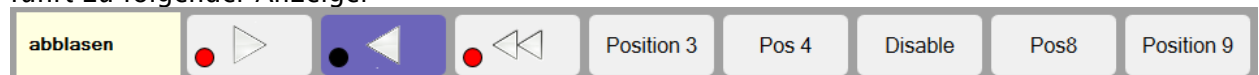
Bsp:

```
/// Blowing
```

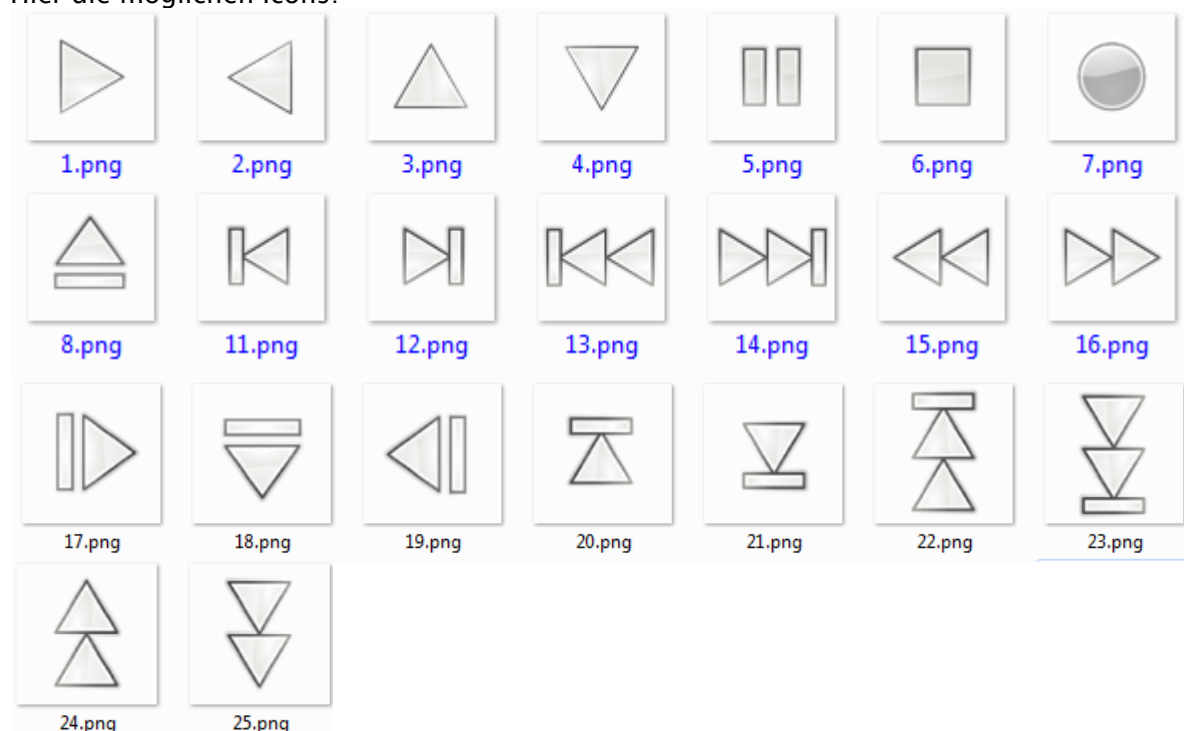
```
/// @1:abblasen @11:1 @2:IMG1:1 @3:IMG2:1 @4:IMG1 5:1 @5:Position 3 @6: Pos 4 @7: Disable @8:Pos8 @9: Position 9
```

```
blowing: BYTE;
```

führt zu folgender Anzeige:



Hier die möglichen Icons:



Texte zur Auswahl (@7:SPSVariablenArray)

Hier wird in „SPSVariablenArray“ eine SPS Variable angegeben, die ein ARRAY OF STRING enthält. Diese Einträge können dann in der HMI zur Auswahl angezeigt werden.

Von der HMI wird dann die Zahl des selektierten Eintrags in die Variable übertragen. Die Variable in der SPS muss vom Typ INT sein.

Beispiel:

```
texts:                                ARRAY[0..6] OF STRING(20) :=
'DISABLE',                            (* 0 *)
'ENABLE',                             (* 1 *)
'Mode2',                              (* 2 *)
'SuperFastMode',                      (* 3 *)
'SlowMotion',                         (* 4 *)
'FastMotion',                         (* 5 *)
'Maintenance',                       (* 6 *)
```

In der Struktur:

```
modes:                                INT; (*@1: Modes @7:.texts *)
```

Die Texte können in der Sprachdatenbank übersetzt werden. Sie werden über den Index „Instanzname von PlcStructure“ + „- arr-“ + Originaltext eingetragen:

PlcStructure.FormPlcStructure - arr-DISABLE	-	255	Deaktivieren	DISABLE	DISAI
PlcStructure.FormPlcStructure - arr-ENABLE	-	255	Aktivieren	ENABLE	ENAE
PlcStructure.FormPlcStructure - arr-FastMotion	-	255	schnelle Bewegung	FastMotion	FastM
PlcStructure.FormPlcStructure - arr-Maintenance	-	255	Wartung	Maintenance	Maint
PlcStructure.FormPlcStructure - arr-Mode2	-	255	Mode2	Mode2	Mode
PlcStructure.FormPlcStructure - arr-SlowMotion	-	255	langsame Bewegung	SlowMotion	SlowM

In der HMI werden die übersetzten Einträge dann in einer Combobox angezeigt. In die SPS wird weiterhin „nur“ der Zahlenwert übertragen.

Description	Value	ActPlcValue	Min	Max	Unit
Aktivflag	0	False			
Wert 1	0	0			
Wert 1	0	0			
Modes	schnelle Bewegung	5			

BECKHOFF**Spezielle Controls zur Eingabe (ShowSpecialControlForEditValues = TRUE)**

Werte aus Variablenarray: (@7:SPSVariablenArray)

Description	Value	ActPlcValue	Min	Max	Unit	Modes
Aktivflag	0	False				Deaktivieren
Wert 1	0	0				Aktivieren
Wert 1	0	0				Mode2
Modes	schnelle Bewegung	5				Super schnell
						langsame Bewegung
						schnelle Bewegung
						Wartung

Zahlenwerte(INTs, Reals):

Description	Value	POSITION
POSITION	377	377

Michael Balsfulland Level: Administrator

F1 F2 F3 Laden F4 Speichern unter F5 zurueck F6 Import

Bool Werte:



PLC

File

1

27.08.12

PLC

11:49:02

CONFIG

FLUXER1

FLUXER2

FLUXER3

HEIZUNG1 U

HEIZUNG2 U

HEIZUNG3 U

HEIZUNG1 O

HEIZUNG2 O

HEIZUNG3 O

TIEGEL1

TIEGEL2

Description	Value
VORHANDEN	False
TYPE	2

VORHANDEN

True

False

Michael Balsfulland

Level: Administrator

F1 read

F2

F3 Laden

F4 Speichern

F5 g

F6 Import

BECKHOFF**Bilddatei anzeigen (@8:DateiName;PosX;PosY;Width;Height)**

Beispieldeklaration in der SPS:

```
stKinematic: ST_KinPara; (* @1:Kinematic Parameter @8:test.jpg;50;100;400;350 *)
```

führt zu folgendem Aussehen:

PLC
File
kin1

KINEMATIC PARAMETER

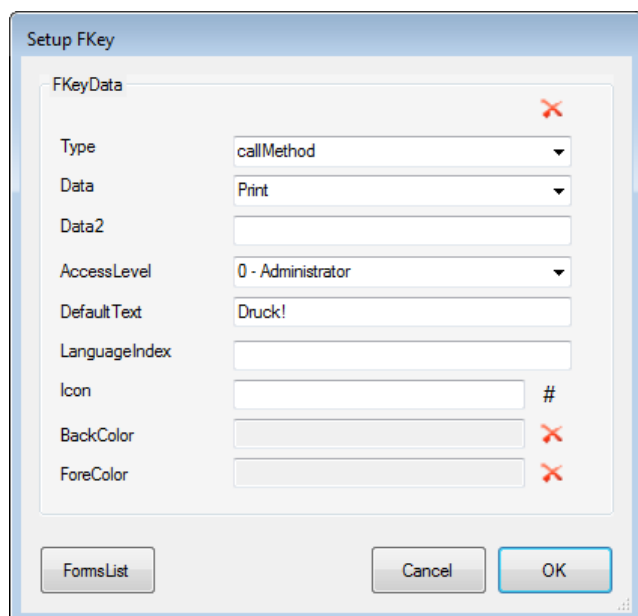
- additional Para
- more additional Para

Description	Value
HD1 Z Offset	1
HD6 B Offset	2
HD16 rotational offset B axis	3

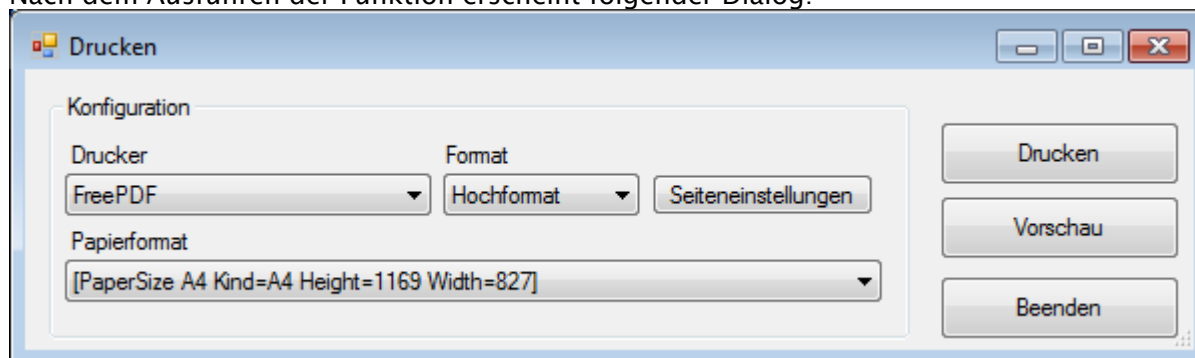
Der Suchpfad für die Bilddateien wird in den Settings mit dem Eintrag „FolderPictures“ eingestellt.

Drucken von Rezepten

Das Drucken vom aktuell angezeigten Rezept kann mit der Methode „Print“ im Menumanager über „callMethod“ für eine beliebige F Taste eingetragen werden:



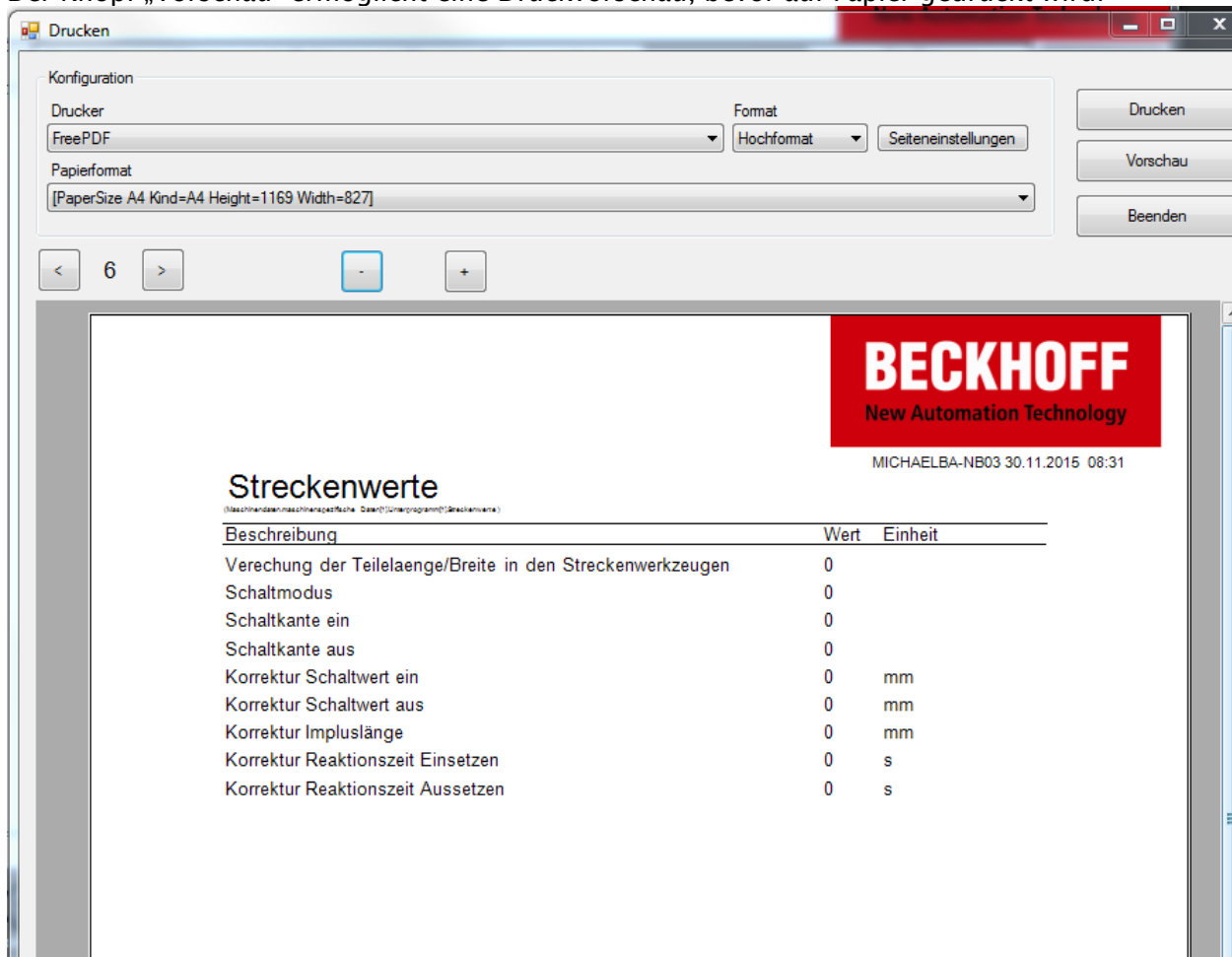
Nach dem Ausführen der Funktion erscheint folgender Dialog:



Es wird immer der Windows Standarddrucker als Voreinstellung gewählt. Die einsprechenden Einstellungen können über die Comboboxen geändert werden.

BECKHOFF

Der Knopf „Vorschau“ ermöglicht eine Druckvorschau, bevor auf Papier gedruckt wird:



Das Logo in der rechten oberen Ecke wird aus der Datei „Customer\Logo.bmp“ geladen und kann durch Überschreiben dieser Datei angepasst werden.

Rechts oben werden außerdem der Rechnername und das Druckdatum mit Uhrzeit gedruckt. Es werden nur sichtbar geschaltete Einträge des Rezepts ausgegeben.