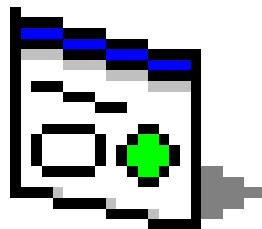


ViMod Version

0.3.X Beta



Back to Basic



Inhaltsverzeichnis

| | |
|---|----|
| Neues Projekt erstellen | 5 |
| Benutzer anlegen | 6 |
| Projektdateien | 7 |
| Projekt laden | 8 |
| Übersicht | 9 |
| Einstellungen | 10 |
| Verbindungseinstellung..... | 11 |
| TCP/IP Verbindung | 11 |
| Serielle Verbindung | 11 |
| Siemens Logo! 8 FS4 Modbus Server | 12 |
| Testmodus | 12 |
| Maximale Verbindungen | 12 |
| Default Verbindungseinstellung..... | 12 |
| IO-Check | 13 |
| Watchdog | 14 |
| Projekt automatisch laden/ausführen | 15 |
| Update..... | 16 |
| Logs..... | 17 |
| Forms..... | 18 |
| Steuerelemente einfügen..... | 20 |
| Allgemeine Eigenschaften | 21 |
| Script-Editor..... | 22 |
| Scripte..... | 22 |
| Globale Variablen | 23 |
| Timer-Funktion | 23 |
| Elemente | 23 |
| Allgemeine Syntax | 24 |
| Steuerelemente Eigenschaften | 25 |
| Analog-Meter | 26 |
| PopUp Fenster | 27 |
| Modbus-Funktionen | 28 |
| Fast Modbus TCP(BK9050) | 29 |
| Spezial Adressen..... | 29 |
| Zykluszeit | 30 |

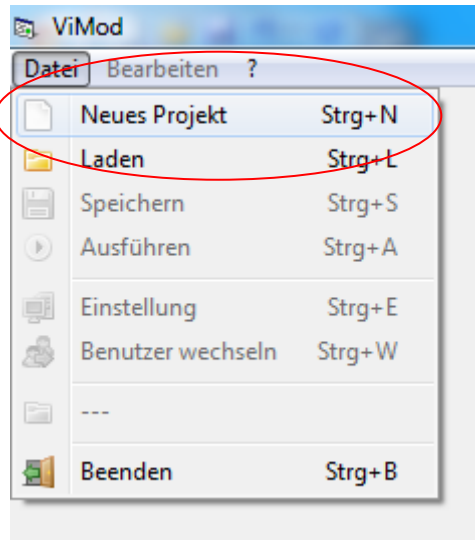


| | |
|--|----|
| I/O-Liste | 31 |
| Liste erstellen | 32 |
| Offset | 33 |
| Mapping | 34 |
| Siemens Logo! | 35 |
| Modbus-Server konfigurieren 8 FS4..... | 35 |
| Modbus-Server konfigurieren 8.3 | 36 |
| Eingänge/Ausgänge/Merker | 36 |
| Logo! Registeradressen | 37 |
| VM-Adressen | 38 |
| Spezielle VM-Adressen | 39 |
| Logo! Beispiel | 39 |
| Aufgaben | 40 |
| S7-Verbindung | 41 |
| S7-Konfigurieren..... | 41 |
| TSAP | 42 |
| S7 Telegramme erstellen..... | 43 |
| Information im Script-Editor | 44 |
| ViMod S7 Error-Codes | 45 |
| Modbus RTU (Remote Terminal Unit)..... | 46 |
| Neues Telegramm erstellen: | 46 |
| Informationen im Script Editor verarbeiten | 47 |
| Debugger | 48 |
| Datenbank | 49 |
| Serielle Schnittstelle | 50 |
| Benutzerkonten..... | 51 |
| Registrieren | 52 |
| FAQ..... | 53 |
| Bekannte Probleme | 54 |
| Versionsverlauf..... | 55 |
| v0.3.5 Beta..... | 55 |
| v0.3.4 Beta..... | 55 |
| v0.3.3 Beta..... | 55 |
| v0.3.2 Beta..... | 55 |
| v0.3.1 Beta..... | 55 |

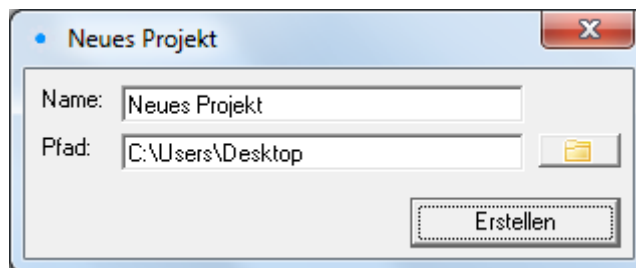


| | |
|-------------------|----|
| v0.3.0 Beta..... | 55 |
| v0.2.9 Beta..... | 55 |
| v0.2.8 Beta..... | 55 |
| v0.2.7 Beta..... | 55 |
| v0.2.6 Beta..... | 55 |
| v0.2.5 Beta..... | 56 |
| v0.2.4 Beta..... | 56 |
| v0.2.3 Beta..... | 56 |
| v0.2.2 Beta..... | 56 |
| v0.2.1 Beta..... | 56 |
| v0.2.0 Beta..... | 56 |
| v0.1.22 Beta..... | 56 |
| v0.1.21 Beta..... | 57 |
| v0.1.20 Beta..... | 57 |
| v0.1.19 Beta..... | 57 |
| v0.1.18 Beta..... | 57 |
| v0.1.16 Beta..... | 57 |
| v0.1.15 Beta..... | 57 |
| v0.1.10 Beta..... | 58 |

Neues Projekt erstellen



Um ein neues Projekt zu erstellen gehen Sie auf Datei --> Neues Projekt oder drücken Strg+N
Vergeben Sie einen Projektnamen und geben Sie an, wo der Projektordner erstellt werden soll. Der Projektnamen kann später noch geändert werden. Klicken Sie dann auf "Erstellen".



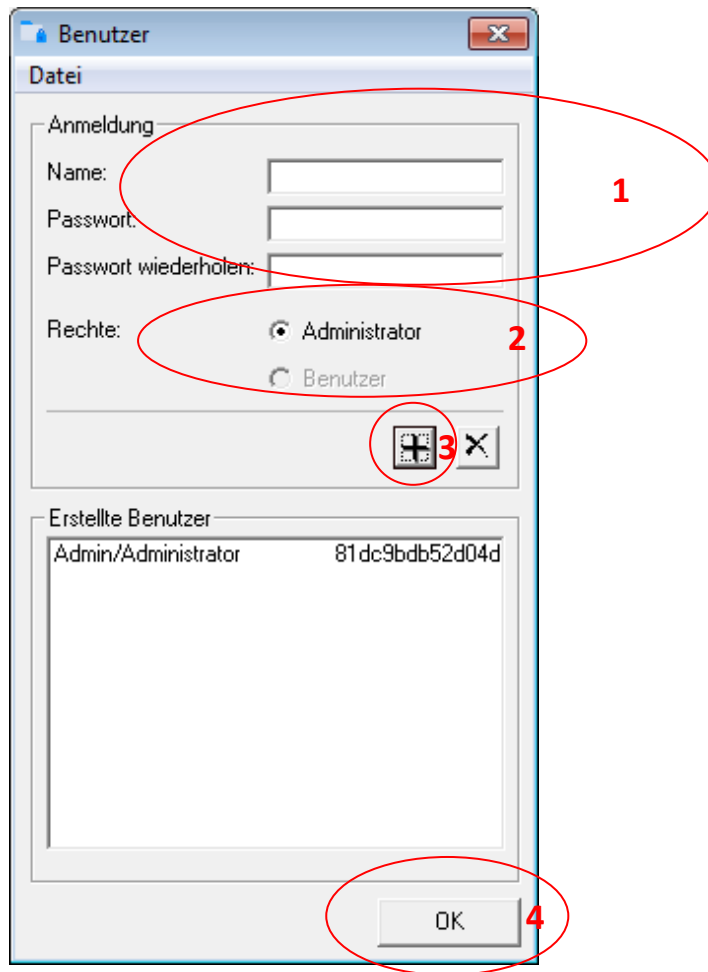
Es muss zu jedem Projekt mindestens ein Administrator erstellt werden.
Geben Sie einen Namen ein und vergeben ein Passwort. Stellen Sie die Rechte für den jeweiligen Benutzer ein und klicken dann auf "OK".
Später können sie unter Einfügen --> Benutzer Konten die Konten verwalten.

Achtung!!

Groß- und Kleinschreibung wird bei der Anmeldung berücksichtigt.

| Rechte | Beschreibung |
|---------------|--|
| Benutzer | Als Benutzer können Sie das fertige Projekt nur ausführen. Es können keine Einstellungen/Änderungen am Projekt vorgenommen werden. |
| Administrator | Als Administrator können Sie alle Einstellungen/Änderungen am Projekt vornehmen. |

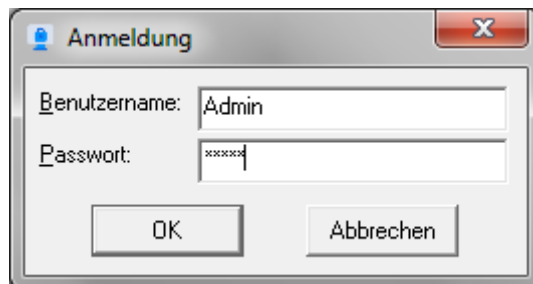
Benutzer anlegen



Nach dem Erstellen, werden Sie gefragt ob Sie ein "Admin Kontakt" hinzufügen wollen. Dieser wird dann unter "About" und/oder bei Fehlern angezeigt. Später können Sie die Einstellung unter Bearbeiten --> Benutzer --> Datei --> Admin Kontakt ändern.

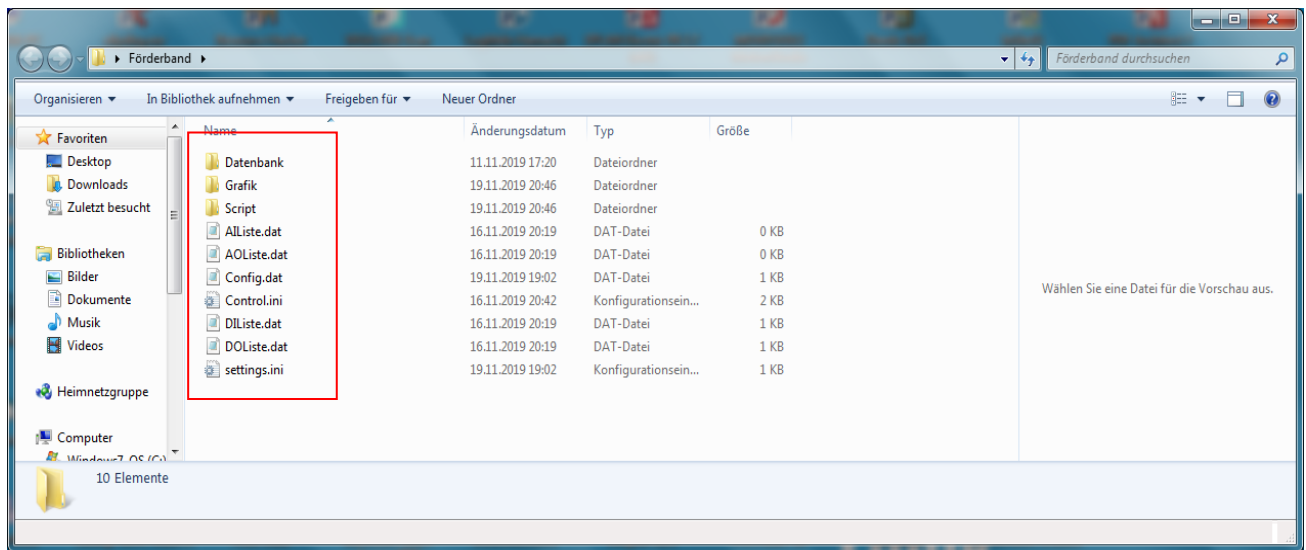
[Benutzerkonten](#)

Melden Sie sich mit dem eben erstellten Konto an.





Projektdateien



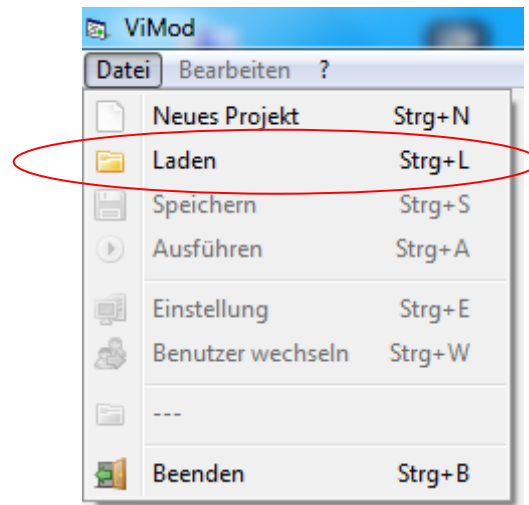
Achtung!!

Diese Dateien dürfen nicht manuell verändert werden!

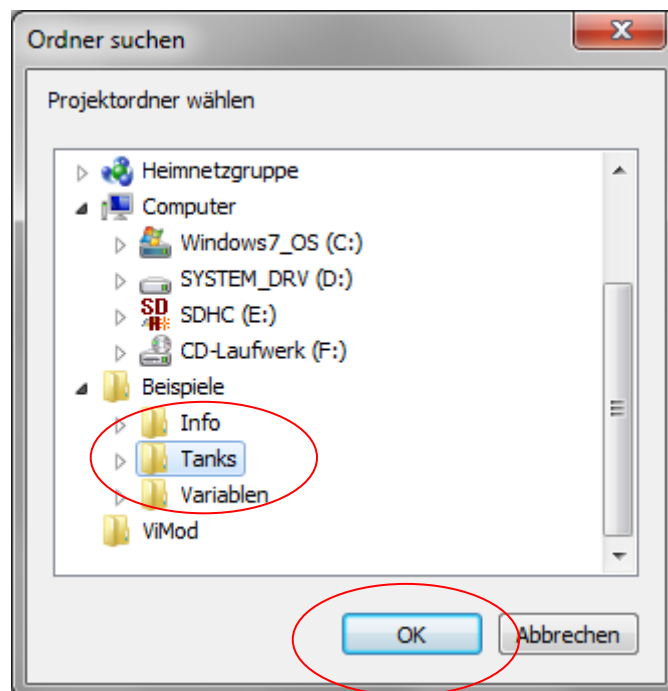
Es dürfen auch keine weiteren Dateien im Projektordner erstellt werden!

Nur die Datenbank kann manuell verändert werden. Weitere Ordner für Dateien können erstellt werden.

Projekt laden

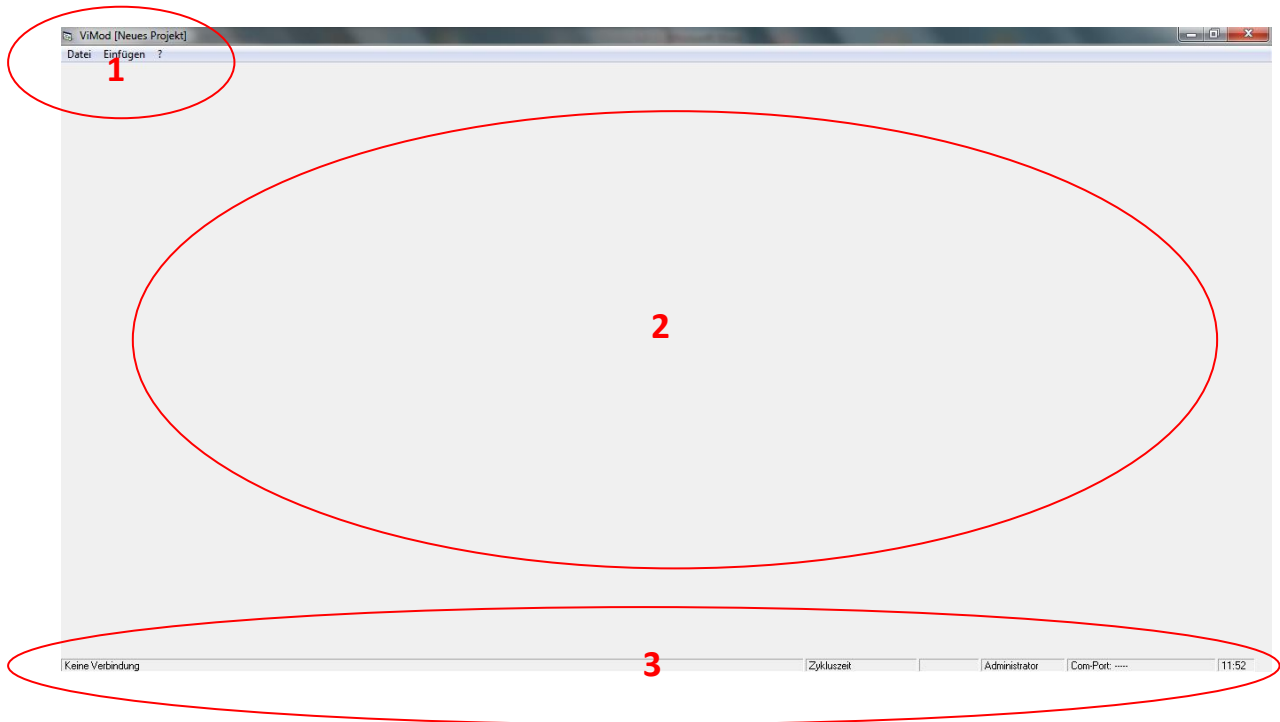


Um ein Projekt zu laden gehen Sie auf Datei --> Laden oder drücken Strg+L
Wählen Sie einen Projektordner aus und klicken Sie auf "OK"



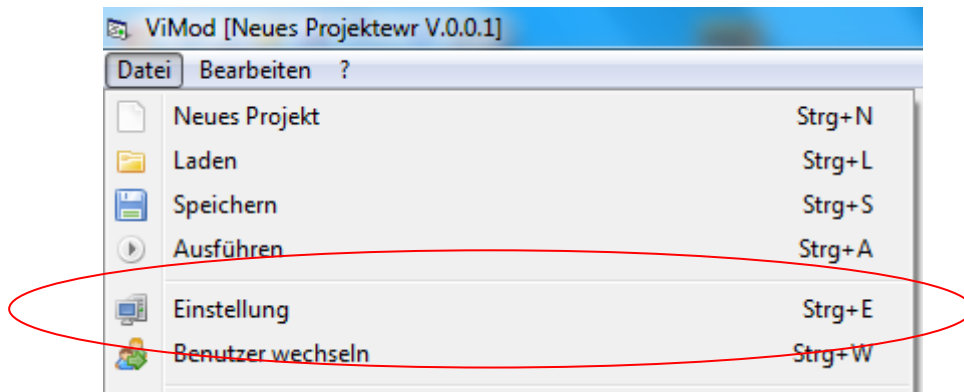


Übersicht



- (1) Im oberen Bereich finden Sie alles, was Sie für die Projekterstellung und Bearbeitung benötigen.
- (2) Im mittleren Bereich können Label, Button, Bilder etc. eingefügt werden.
- (3) Im unteren Bereich werden Informationen/Status angezeigt.

Einstellungen



Gehen Sie auf Datei --> Einstellungen



Verbindungseinstellung

The screenshot shows the 'Einstellung' (Settings) window with the 'S7 Verbindung' tab selected. The 'Device Verbindung' section has 'Modbus TCP/IP' and 'IO-Check' checked. The 'TCP/IP-Verbindung' section has 'IP-Adresse' set to 172.16.17.1, 'Port' set to 502, and 'Device Adresse' set to 0. The 'COM-Port' section has 'Com-Port' set to 1, 'Baudrate' set to 9600, 'Datenbits' set to 8, 'Parität' set to Keine, and 'Stoppsbit' set to 1. An 'OK' button is located at the bottom right of the window.

TCP/IP Verbindung

Geben Sie die IP-Adresse/Port sowie die Device Adresse der Steuerung ein. Sie können auch nach der Steuerung im Netzwerk suchen.

Achtung!!

Es wird nur die Submask: 255.255.255.0 durchsucht.

Durch einen Doppelklick auf eine gefundene Steuerung werden die Einstellungen automatisch eingetragen.

Serielle Verbindung

Ab Version 0.2.x kann man sich auch mit einer seriellen Verbindung mit dem Device verbinden. (Modbus RTU über RS485)

Achtung!!

ViMod zeigt nur COM-Ports bis 16 an. Dies gilt auch für Geräte, die eingebunden werden. Nehmen Sie, falls notwendig, die richtigen Einstellungen im Windows-Gerätemanager vor.



Siemens Logo! 8 FS4 Modbus Server

Aktivieren Sie "Siemens Logo! 8 FS4 Modbus Server", wenn Sie eine Logo! (8 FS4 oder 8.3) als Device haben.

Mehr Info [Siemens Logo!](#)

Testmodus

Ist der Testmodus aktiviert, wird keine Verbindung zum Device aufgebaut. Es wird nur das Script abgearbeitet. Läuft das Programm im Testmodus, wird dies oben links angezeigt. Im Testmodus ist eine Serielle Verbindung über den Com-Port möglich.

Achtung!!

Alle Eingänge/Ausgängen haben den Wert "0" bzw. "False"

Maximale Verbindungen

- Eine TCP/IP Verbindung zu einem Modbus-Device **oder**
- Eine TCP/IP Verbindung zu einem S7-Device (S7-Verbindung) **und**
- Eine RTU Verbindung zu einem Modbus-Device **und**
- Eine Serielle Verbindung zu einem Device z.B. Scanner, Waage

Default Verbindungseinstellung

- TCP/IP Verbindung
- IP-Adresse : 172.16.17.1
- Watchdog deaktiviert
- Offsets: 0,0,0,0

IO-Check

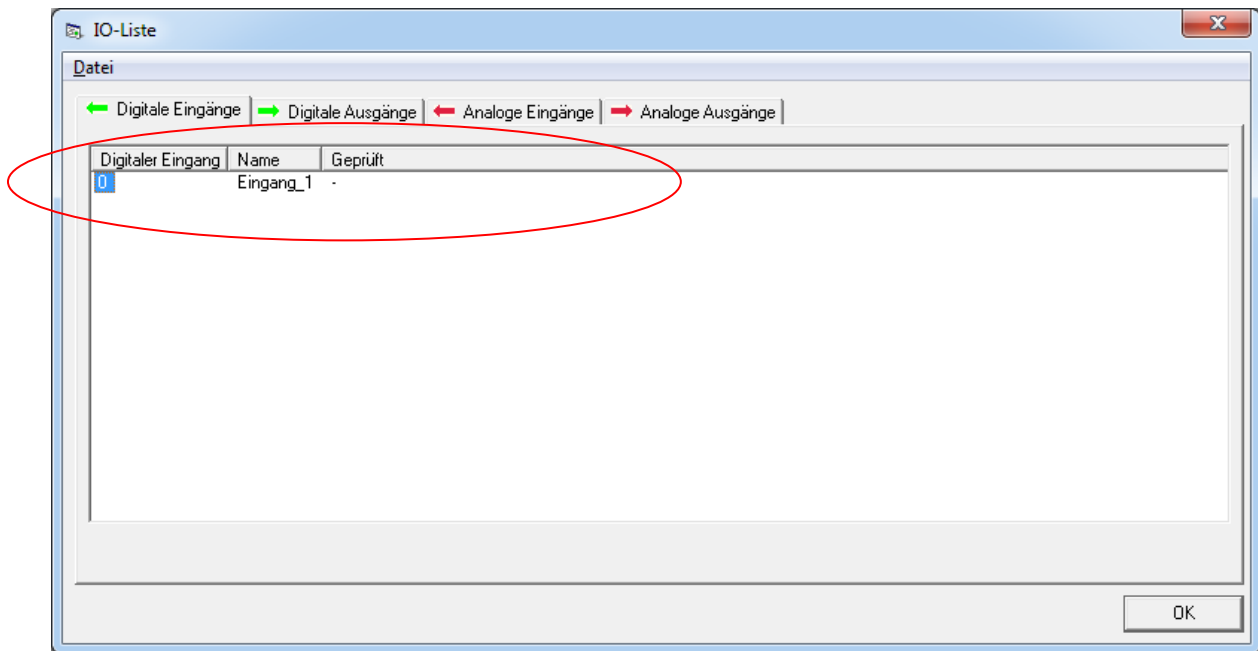
Aktivieren Sie "IO-Check" um das Programm im "IO-Check Modus" zu starten. Starten Sie das Projekt über "Datei" --> "Ausführen" Der Modus wird oben links in der Ecke angezeigt.

Achtung!!

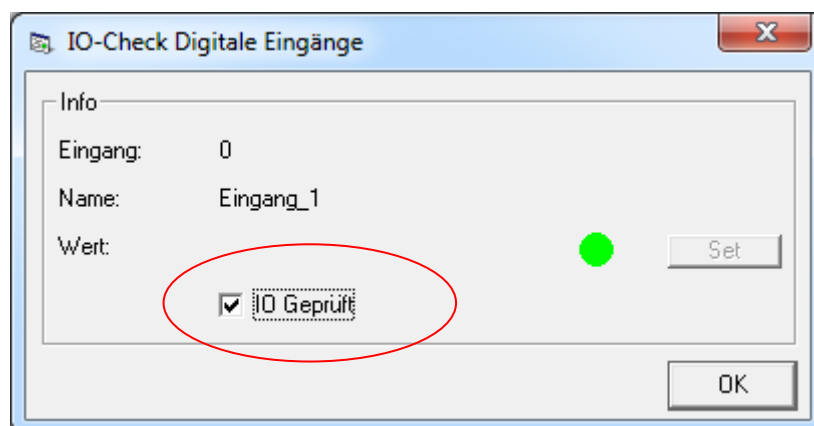
Wird das Projekt oder ViMod geschlossen wird der "IO-Check Modus" deaktiviert.

In diesem Modus werden die Skripte nicht abgearbeitet.

Ist "IO-Check Modus" und "Testmodus" aktiviert, ist der "Testmodus" priorisiert



Wählen Sie den Eingang/Ausgang durch einen Doppelklick aus.

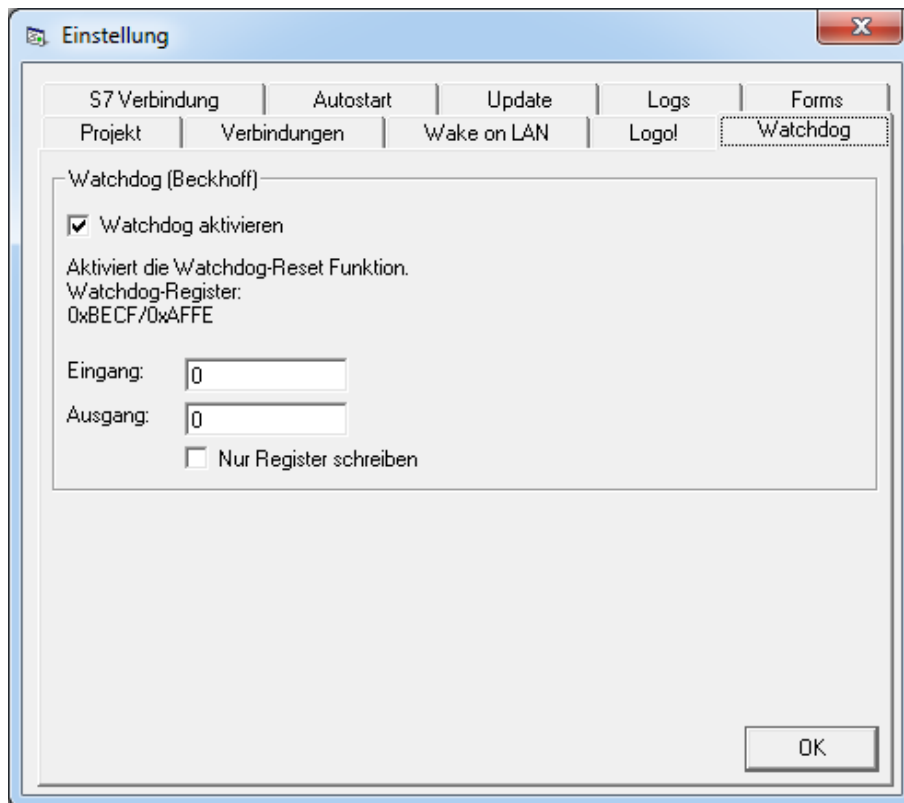


Achtung!!

In diesem Modus ist die Aktualisierungszeit reduziert.

Bestätigen Sie, dass der IO geprüft ist. In der IO-Liste wird der User und ein Zeitstempel hinzugefügt.

Watchdog



Aktivieren Sie "Watchdog aktivieren", wenn Sie eine Beckhoff-Steuerung haben z.B. BK9050. Beim Start wird dann automatisch ein Watchdog-Reset durchgeführt. Verdrahten Sie den definierten Watchdog-Ausgang auf den definierten Watchdog-Eingang. Dadurch kann die Software feststellen, ob der Watchdog-Ausgang wirklich gesetzt wurde. Außerdem haben Sie eine visuelle Bestätigung am Buskoppler. Haben Sie keine Verdrahtung, aktivieren Sie "Nur Register schreiben".
 Wenn "Nur Register schreiben" ausgewählt ist, wird der Eingang/Ausgang automatisch auf Null gesetzt.

Achtung!!

Ist die Watchdog-Funktion in den Einstellungen aktiviert, aber kein Watchdog Eingang/Ausgang definiert, kommt es während der Laufzeit zu einem Fehler. Dies gilt nur, wenn "Nur Register schreiben" deaktiviert ist.

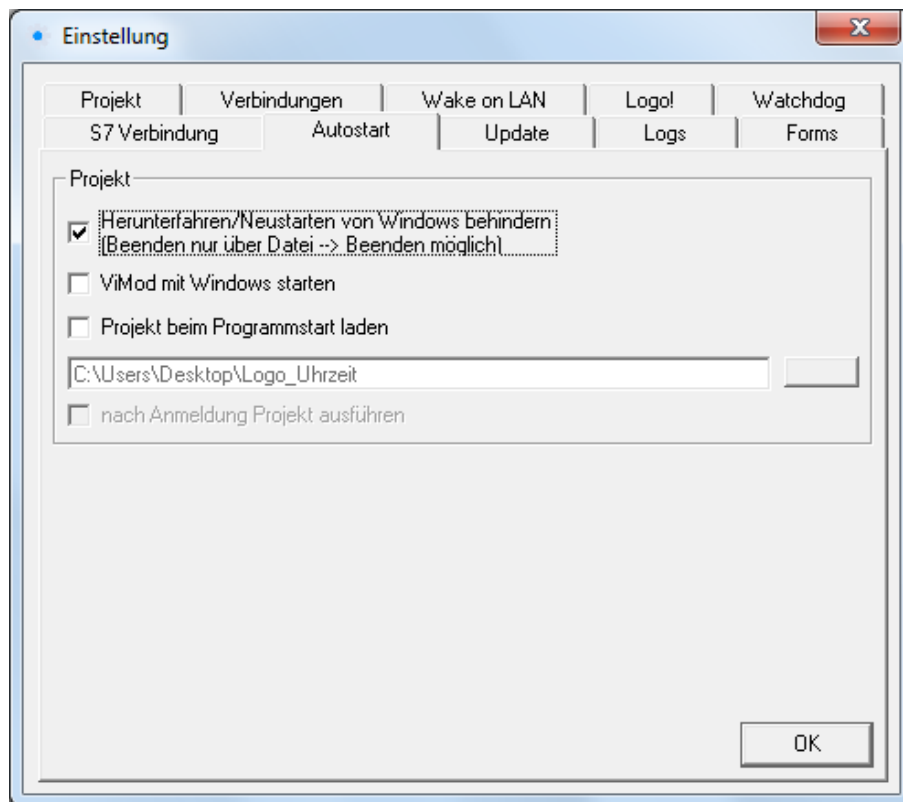
Achtung!!

Sollten Sie eine Logo! verwenden, stellen Sie sicher, dass, wenn die Verbindung zu ViMod unterbrochen wird, die Ausgänge abgeschaltet werden. Ein Watchdog-Beispiel finden Sie unter:

<https://nicos-alltag.blogspot.com/>



Projekt automatisch laden/ausführen



"Herunterfahren/Neustarten von Windows behindern (Beenden nur über Datei --> Beenden möglich)"

- Der Benutzer muss aktiv ViMod beenden bevor der PC herunterfährt bzw. neustartet. ViMod läuft bei der Meldung ("Dieses Programm verhindert, dass Windows heruntergefahren/ neu gestartet wird") im Hintergrund weiter.

Achtung!!

ViMod kann nur über "Datei" --> "Beenden" beendet werden. Nicht über den Schließen Dialog von Windows

"ViMod mit Windows starten"

- ViMod startet automatisch mit Windows

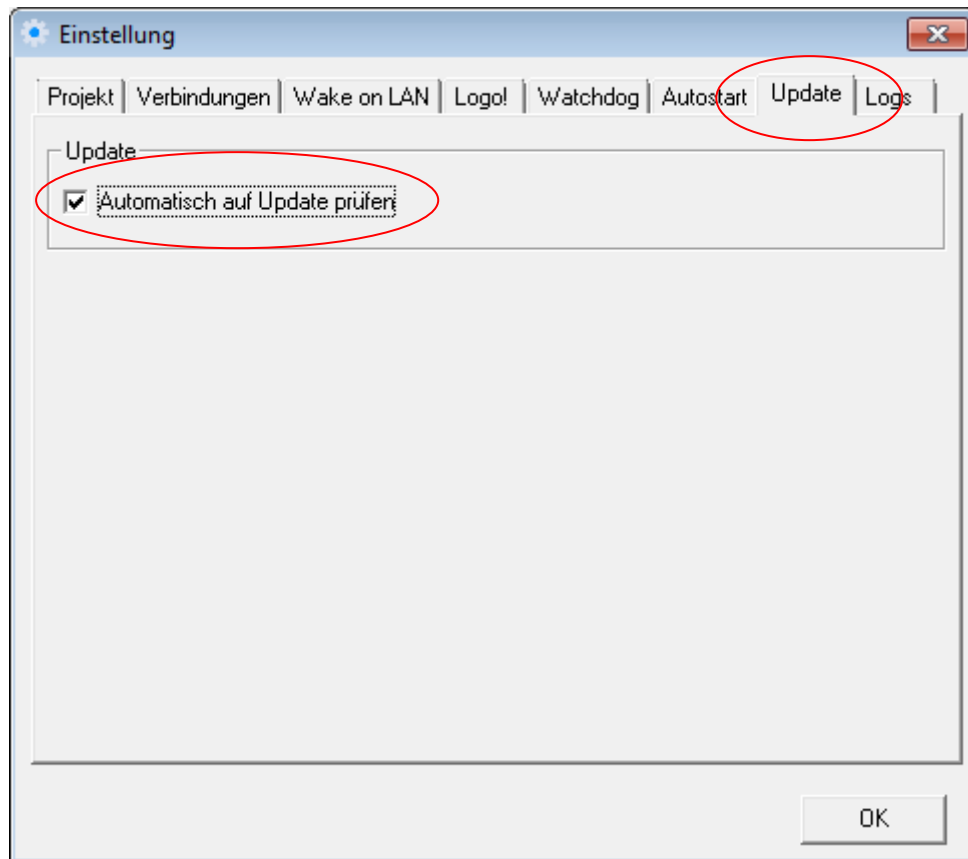
"Projekt beim Programmstart laden"

- Das Projekt wird beim Start von ViMod automatisch geladen

"nach Anmeldung Projekt ausführen"

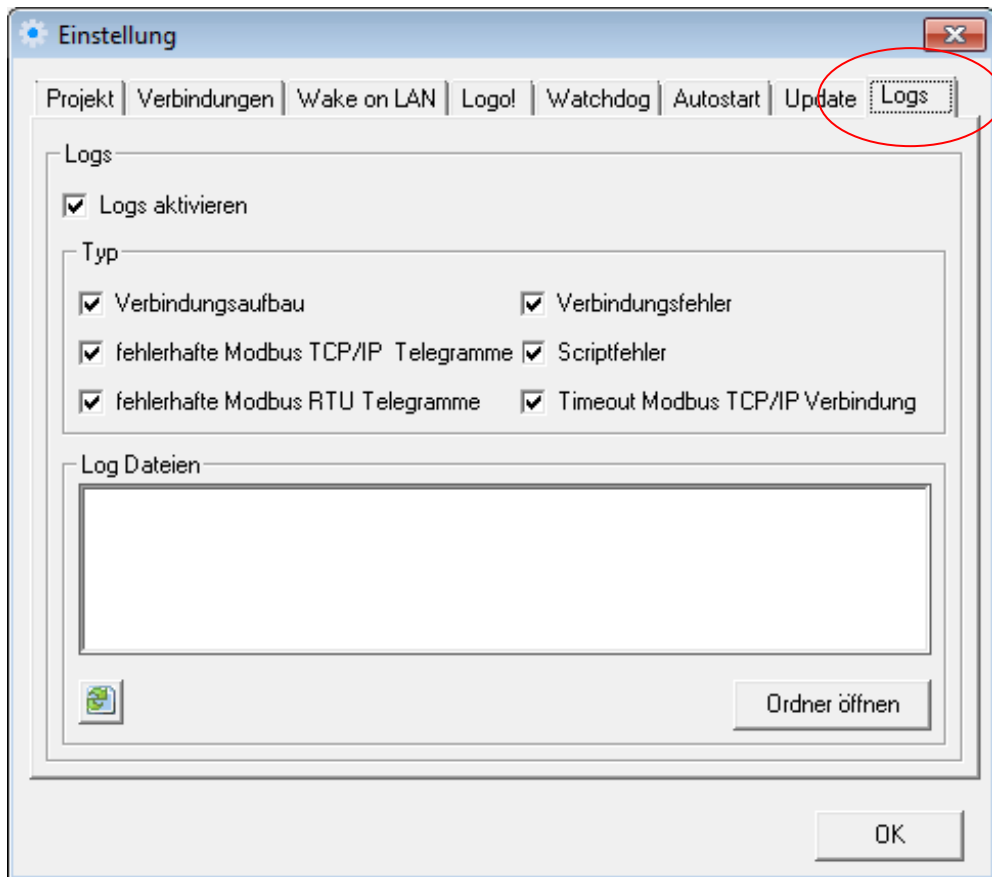
- Nach der Anmeldung wird das Projekt zeitverzögert automatisch ausgeführt. Dies wird in der Titelleiste angezeigt

Update



Wenn "Automatisch auf Update prüfen" aktiviert ist, wird beim Programmstart im Hintergrund geprüft ob eine neue Version verfügbar ist. Benutzen Sie bitte immer die aktuellste Version.

Logs



Diese Funktion dient dazu um das Projekt auf Fehler zu testen.

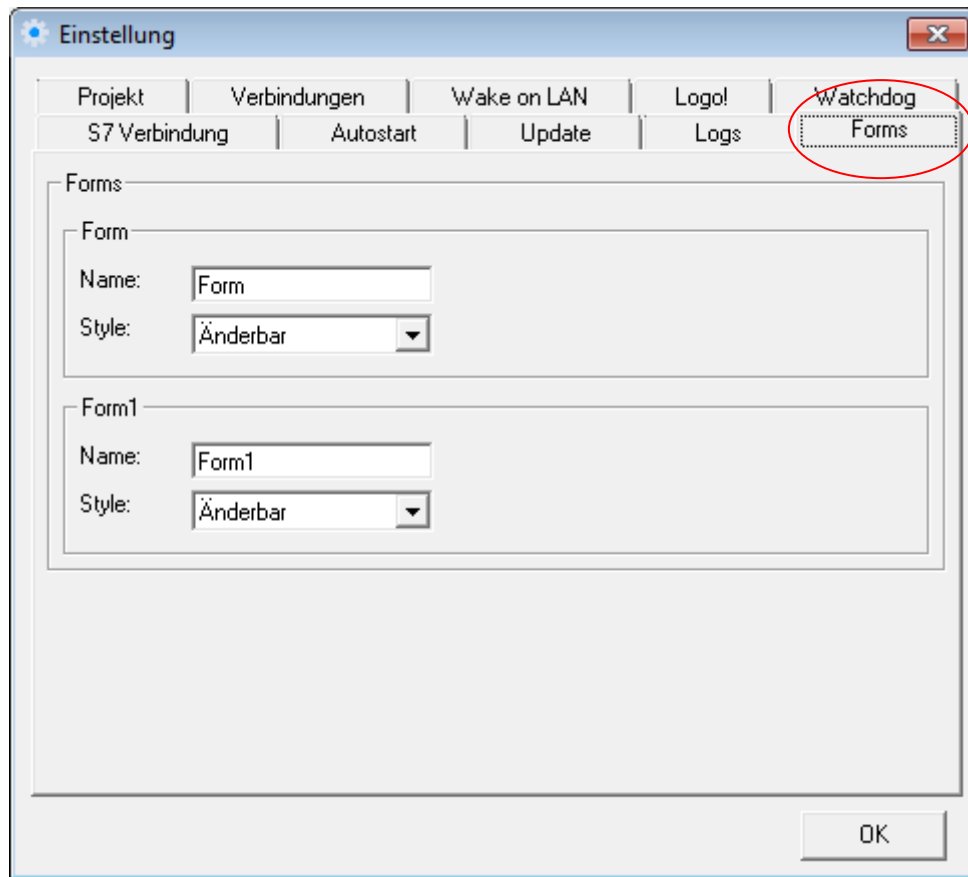
Die Logs werden beim ausführen des Projektes aktiviert und schreiben sie beim beenden des Projektes in die Log-Datei.

Sind die Logs aktiviert kann über den Befehl "Print_Log" im Script Editor in die Log-Datei geschrieben werden.

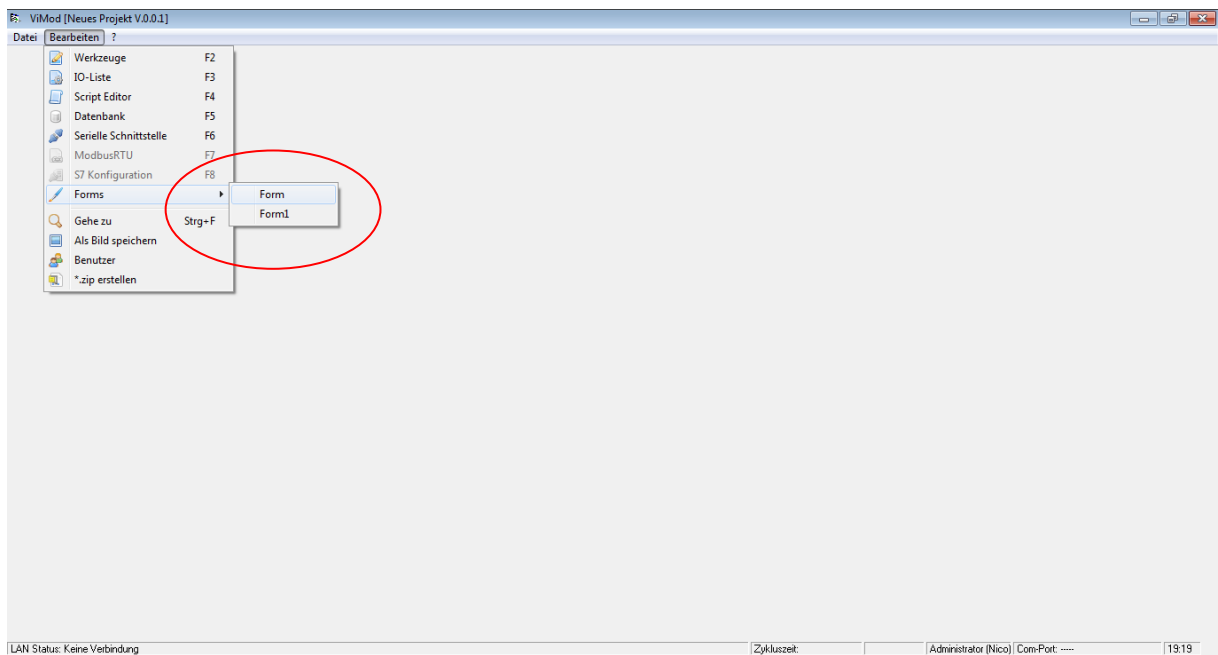
Beispiel:

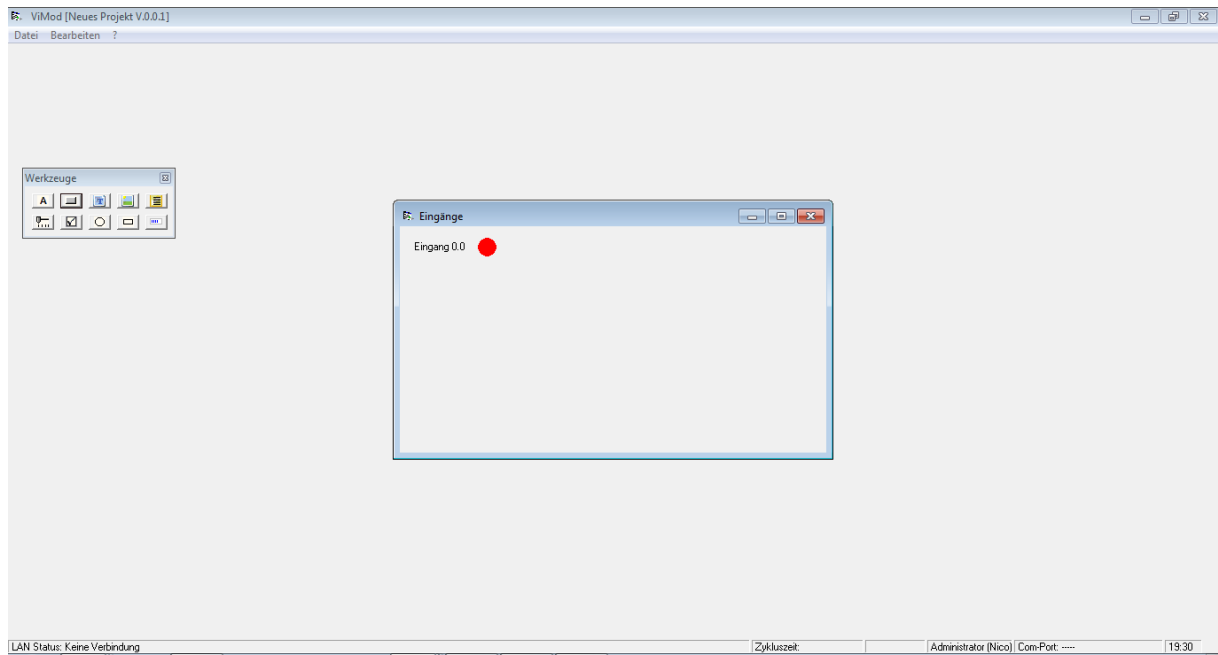
```
Print_Log = "Hallo"
```

Forms



Unter Einstellung --> Forms können zwei weitere Forms konfiguriert werden. Wie bei der Hauptform können die Steuerelemente frei platziert und angepasst werden.



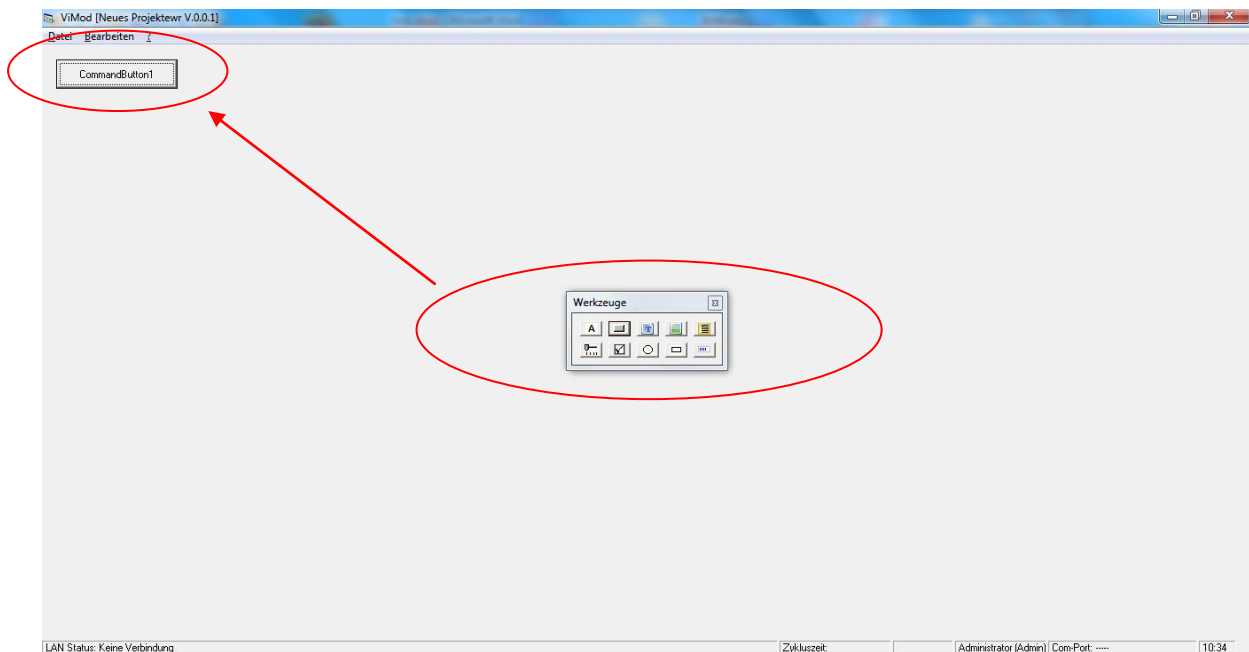


| Funktion | Beschreibung |
|--|--|
| Open_Form Beispiel: Open_Form = "Eingänge" | Öffnet die Form mit dem Namen "Eingänge" |
| Close_Form Beispiel: Close_Form = "Eingänge" | Schließt die Form mit dem Namen "Eingänge" |

Mehr unter [Steuerelemente](#)



Steuerelemente einfügen

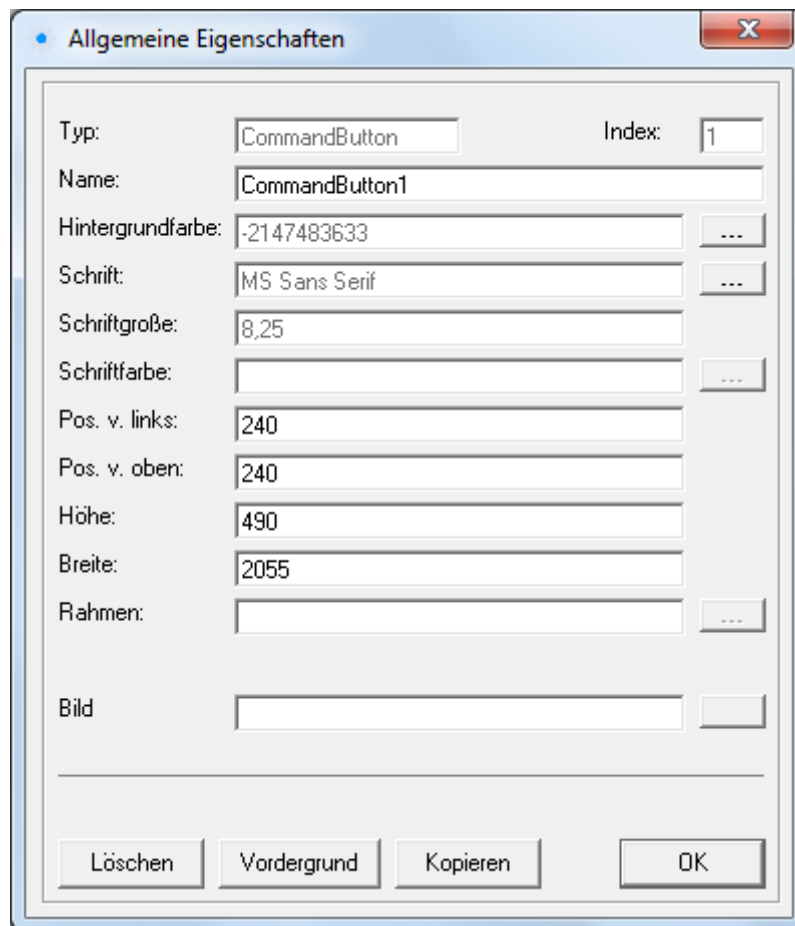


Unter Einfügen --> Werkzeuge können Sie verschiedene Steuerelemente hinzufügen.
Klicken Sie auf das Steuerelement, das Sie hinzufügen wollen. Es wird dann links oben hinzugefügt.

Es stehen folgende Steuerelemente zur Verfügung:

- Label
- Button
- Text Box
- Bild
- Check Box
- Kreis
- Rechteck
- Slider
- ProgressBar
- List

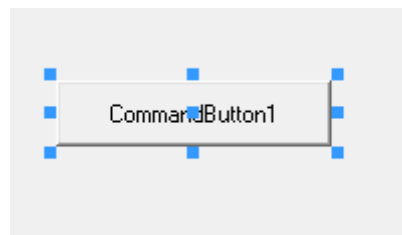
Allgemeine Eigenschaften



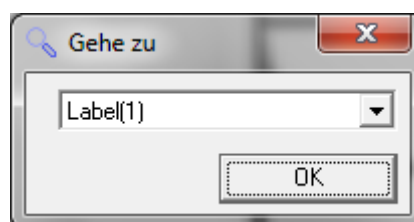
The dialog box 'Allgemeine Eigenschaften' contains the following fields and controls:

- Typ:** CommandButton
- Index:** 1
- Name:** CommandButton1
- Hintergrundfarbe:** -2147483633 (with a color selection button)
- Schrift:** MS Sans Serif (with a font selection button)
- Schriftgröße:** 8,25
- Schriftfarbe:** (with a color selection button)
- Pos. v. links:** 240
- Pos. v. oben:** 240
- Höhe:** 490
- Breite:** 2055
- Rahmen:** (with a selection button)
- Bild:** (with a selection button)
- Buttons at the bottom:** Löschen, Vordergrund, Kopieren, OK

Klicken Sie mit der linken Maustaste auf das hinzugefügte Element. Hier können Sie Einstellungen zu dem Element vornehmen. Diese Einstellungen können auch mit dem Script-Editor in der Laufzeit geändert werden.



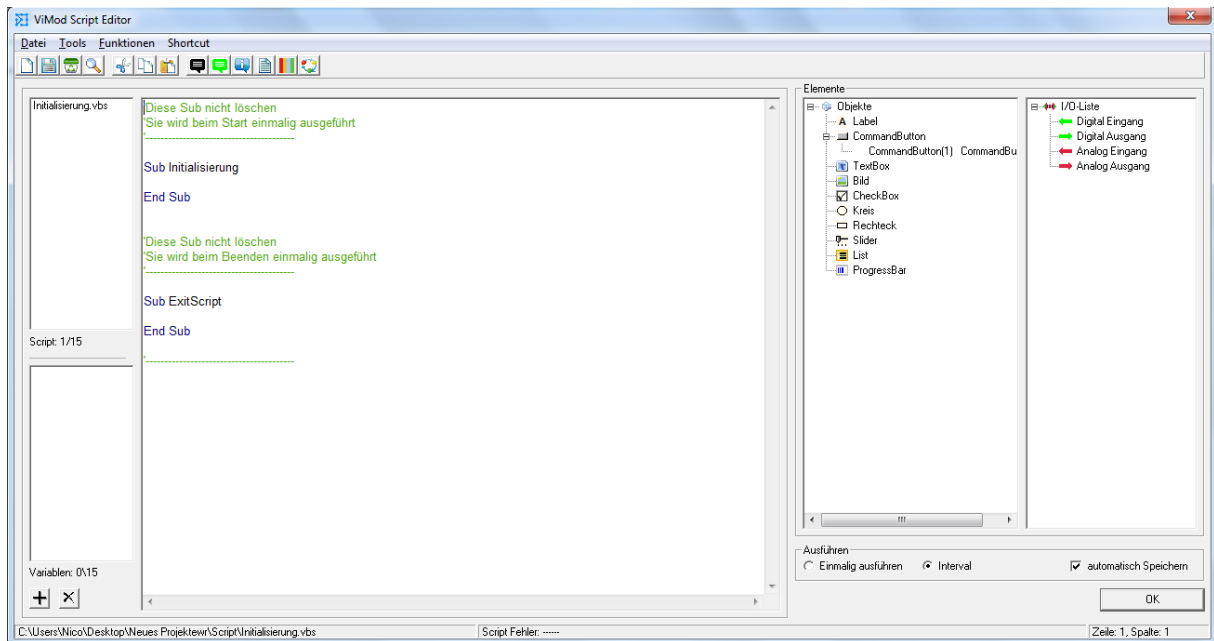
Klicken Sie mit der rechten Maustaste auf das hinzugefügte Element, um seine Position oder Größe zu verändern.



Unter "Einfügen" --> "Gehe zu" können Sie zu den erstellten Steuerelementen springen.



Script-Editor



Scripte

Mit dem Script-Editor können Sie in der VBS-Sprache Scripte erstellen. Diese Scripte laufen im Hintergrund ab.

Ist die Option "Einmalig ausführen" gewählt, werden alle Scripte einmal ausgeführt.

Bei der Option "Interval" werden Sie zyklisch ausgeführt.

Die Sub "Initialisierung" wird beim Start immer einmalig ausgeführt. Hier können sich z.B. Labeleinträge löschen oder eine Initialisierung einer Datenbank vornehmen lassen.

Die Sub "ExitScript" wird beim Stoppen des Skripts ausgeführt.

Achtung!!

In der Sub "ExitScript" können keine Eingänge/Ausgänge etc. gelesen oder geschrieben werden!!

Es können maximal 15 Scripte erstellt werden.



Globale Variablen

Sie können maximal 15 globale Variablen deklarieren. Diese Variablen haben dann in allen Scripten ihre Gültigkeit. Sie besitzen den Datentyp Variant.

Globalen Variablen im Script einen Wert zuweisen:

```
Call Set_Global_Variable("Variablen_Name", "Wert")
```

Wert aus einer Variablen lesen:

```
Get_Global_Variable("Variablen_Name")
```

Timer-Funktion

Es stehen 5 Timer zur Verfügung; Index 1 bis 5. Die Timer zählen eine Variable in einem Intervall von 1000 (1 Sek.) hoch.

| Funktion | Beschreibung |
|--|---|
| Timer.Start() Beispiel: Timer.Start(1) = True | Startet bzw. beendet den Timer True = Timer Start False = Timer Stopp |
| Timer_Reset() Beispiel: Timer_Reset(1) | Setzt die Timer Variable auf 0 |
| Timer_Counter() Beispiel: Label.Caption(1) = Timer_Counter(1) | Gibt den Timer Counter als Long zurück |

Elemente

Objekte: Hier finden Sie eine Auflistung aller Steuerelemente, die Sie erstellt haben.

I/O-Liste: Hier finden Sie eine Auflistung aller I/O's, die Sie erstellt haben.



Allgemeine Syntax

Steuerelement ansprechen:

Beispiel:

```
Label.Caption(1) = Time
```

Schreibt die aktuelle Zeit in das Label mit dem Index 1.

Beispiel:

```
Kreis.FillColor(1) = VBGreen  
Kreis.BorderColor(1) = VBGreen
```

Färbt den Rahmen und den Inhalt vom Kreis mit dem Index 1 Grün.

Beispiel:

```
Sub Command_Click1()  
Msgbox "Hallo"  
End Sub
```

Wird der Command-Button mit dem Index 1 angeklickt, erscheint eine Message-Box mit dem Inhalt "HALLO". Die VBS Syntax kennt keine Groß-Kleinschreibung.

Beispiel:

```
Sub Command_Click1()  
Msgbox "H" & LCase("allo")  
End Sub
```

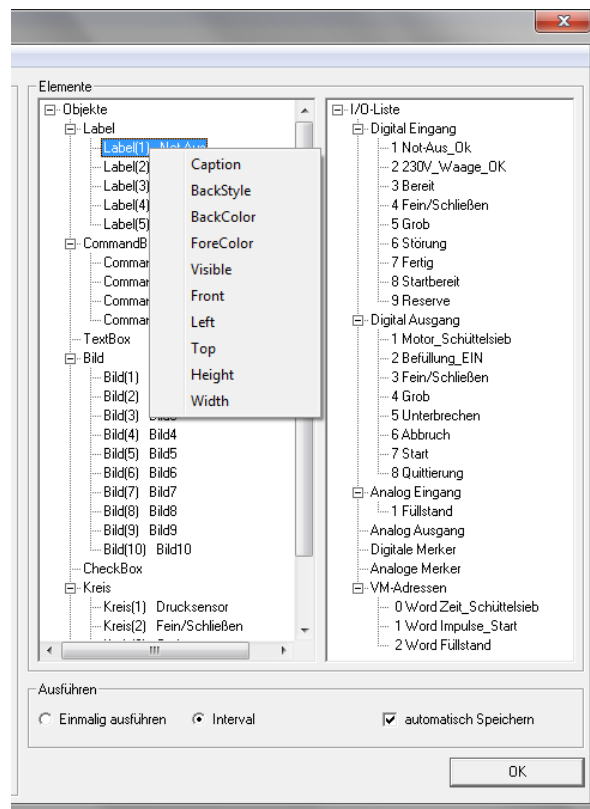
"Hallo" wird "richtig" angezeigt

Beispiel:

```
Sub Command_Click1()  
Msgbox Text.Text(1)  
End Sub
```

Wird der Command-Button mit dem Index 1 angeklickt, erscheint eine Message-Box mit dem Inhalt der TextBox mit dem Index 1.

Steuerelemente Eigenschaften



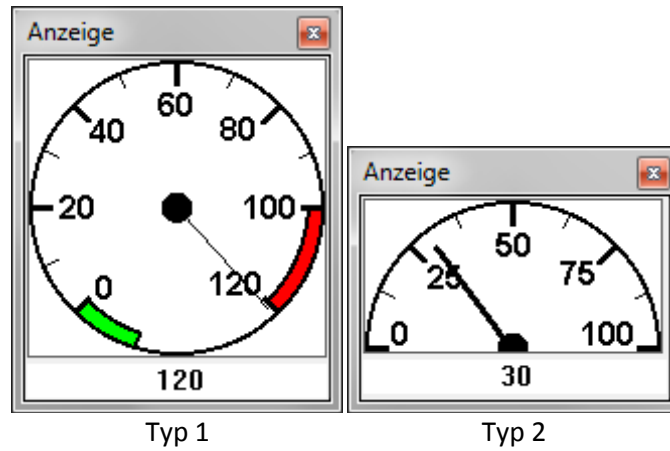
Klicken Sie unter "Elemente" --> "Objekte" mit der rechten Maustaste auf ein Objekt. Es werden dann alle Eigenschaften angezeigt, die verändert bzw. gesetzt werden können. Die Auswahl wird im Script an der Cursor-Position eingefügt.

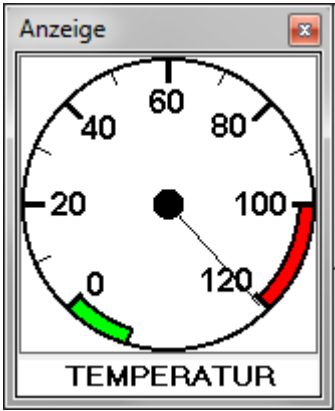
Weitere Beispiele auf <https://nicos-alltag.blogspot.com/>



Analog-Meter

Es stehen zwei Analog-Meter zur Verfügung .



| Funktion | Beschreibung |
|--|---|
| <p><code>Call Set_AnalogMeter(Typ, Bezeichnung, Wert)</code></p> <p>Beispiel: <code>Call Set_AnalogMeter(1, "Temperatur", "120")</code></p>  | <p>Erzeugt ein Analog-Meter mit den übergebenen Parametern.</p> |



PopUp Fenster

Mit der PopUp Funktion können Sie ein PopUp Fenster unten rechts am Bildschirm erzeugen.



| Funktion | Beschreibung |
|--|--|
| Auto_Close Beispiel: Auto_Close = True | Wenn Auto_Close auf True ist, muss der Benutzer das PopUp bestätigen. Es schließt sich nicht von selbst |
| PopUp_Titel Beispiel: PopUp_Titel = "Meldung" | Legt den Titel des PopUp Fenster fest |
| PopUp_Nachricht Beispiel: PopUp_Nachricht = "Nachricht" | Legt die Nachricht des PopUp Fenster fest |
| Open_Popup Beispiel: Open_Popup = True | Öffnet das PopUp Fenster |
| PopUp_Click Beispiel: If PopUp_Click = True then Kreis.FillColor(1) = VBGreen Kreis.BorderColor(1) = VBGreen else Kreis.FillColor(1) = VBRed Kreis.BorderColor(1) = VBRed End If | Prüft ob das PopUp Fenster vom Benutzer geschlossen/ quittiert wurde. Ablauf: --> PopUp_Click = False Benutzer Klickt auf das PopUp Fenster --> PopUp_Click = True PopUp Fenster verschwindet --> PopUp_Click = False |
| PopUp_Style Beispiel: PopUp_Style = 1 | Legt den Style fest 0 = Default 1 = Blau 2 = Grün 3 = Silber |



Modbus-Funktionen

Es werden automatisch folgende Eingänge/Ausgänge eingelesen:

| Discrete Input Digitale Eingänge | Coils Digitale Ausgänge | Input Registers Analoge Eingänge | Holding Registers Analoge Ausgänge |
|--|-----------------------------------|--|--|
| Von 0 bis 254 | Von 0 bis 254 | Von 0 bis 119 | Von 0 bis 119 |
| FC* 2 | FC* 1/15 | FC* 4 | FC* 6/16 |

Achtung!!

Bei Simulationsprogrammen, wie z.B. "Easy Modbus Server Simulator", kann es zu Fehlern beim Setzen der Ausgänge kommen.

| Funktion | Beschreibung |
|---|---|
| Get_Digital_Input() Beispiel: If Get_Digital_Input(0) = True then MsgBox "Eingang 0 ist gesetzt" End If | Liest den Zustand des ersten digitalen Eingangs ein. Gibt einen True- oder False-Wert zurück. |
| Get_Digital_Output() Beispiel: If Get_Digital_Output(0) = True then MsgBox "Ausgang 0 ist gesetzt" End If | Liest den Zustand des ersten digitalen Ausgangs ein. Gibt einen True- oder False-Wert zurück. |
| Set_Digital_Output() Beispiel: Set_Digital_Output(0) = True | Setzt den ersten digitalen Ausgang True = 1 False = 0 |
| Get_Analog_Input() Beispiel: If Get_Analog_Input(0) > 100 then MsgBox "Der Wert ist > 100" End If | Gibt den Zahlenwert des ersten analogen Eingangs zurück. |
| Get_Analog_Output() Beispiel: If Get_Analog_Output(0) > 100 then MsgBox "Der Wert ist > 100" End If | Gibt den Zahlenwert des ersten analogen Ausgangs zurück. |
| Set_Analog_Output() Beispiel: Set_Analog_Output(0) = 255 | Setzt den ersten analogen Ausgang auf Wert X. |

* Modbus Funktionscode



Fast Modbus TCP(BK9050)

Um die Funktion "Fast Modbus TCP" mit der KS2000 zu aktivieren gehen Sie wie folgt vor:

- Mit dem Busankoppler verbinden
- Online --> Koppler --> Dienste auswählen
- "Reset Schreibschutz"
- In der Tabelle 100, Register 29 auf 1 setzen
- Online --> Koppler --> Dienste auswählen
- "Schreibschutz setzen"
- "Reset"

Die Funktion "Fast Modbus TCP" wird nur für kleine Netzwerke empfohlen.

Spezial Adressen

Mit der Funktion `Set_SpecialAddress` bzw. `Get_SpecialAddress` können Analoge Register (Holding Register) beschrieben bzw. gelesen werden. Es stehen 5 (0-4) frei wählbare Adressen zur Verfügung.

Register schreiben:

```
Set_SpecialAddress (Index, Adresse (Hex) = Daten
```

Register lesen:

```
Get_SpecialAddress (Index, Adresse (Hex) )
```

Achtung!!

Werden die Daten z.B. über einen Button abgefragt, sind sie erst im nächsten Script Zyklus verfügbar.



Zykluszeit

Es werden pro Zyklus immer eine bestimmte Anzahl von Eingängen/Ausgängen/Merker usw. eingelesen bzw. geschrieben.

- [Modbus \(BK9050\)](#)
- [Siemens Logo!](#)
- [S7-Verbindung](#)

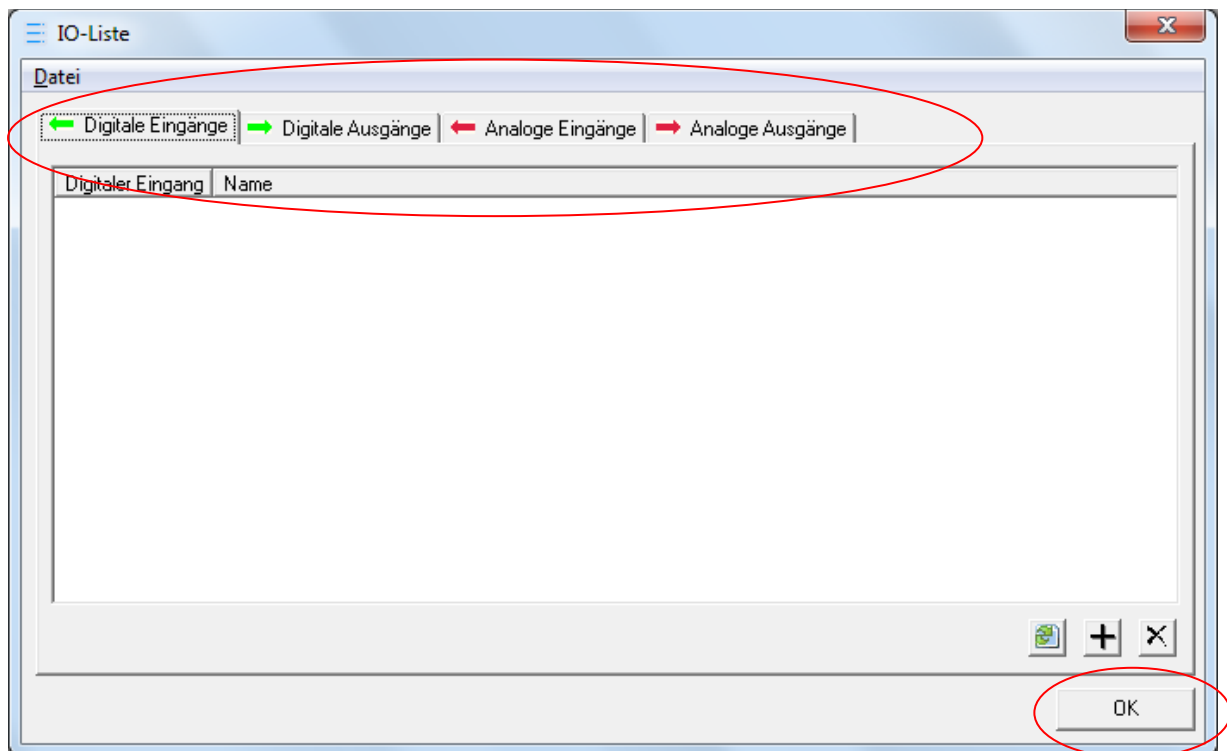
Unabhängig ob diese in der [I/O Liste](#) definiert sind oder nicht. Sie müssen für die Verarbeitung im Script Editor in der I/O Liste definiert sein. Dadurch wird eine fast gleichbleibende Zykluszeit erreicht. Die Zykluszeit setzt sich wie folgt zusammen.

- Start der Zeitmessung
- Eingänge/Ausgänge lesen/schreiben
- Benutzerdefinierte Adressen/DB lesen/schreiben
- Script abarbeiten
- Ende der Zeitmessung

Sobald ViMod eine Antwort bzw. Bestätigung von einem Device bekommt wird das nächste Telegramm gesendet. Das heißt die Zykluszeit ist noch von folgenden Dingen abhängig.

- Geschwindigkeit der Netzwerkverbindung
- Reaktionszeit des Device
- Dauer, die der Computer braucht um das Script abzuarbeiten
- In der Visualisierung anzeigen

I/O-Liste



Wählen Sie den Reiter, für den Sie einen Eingang/Ausgang hinzufügen möchten, aus.
Die Ein- und Ausgänge werden im Script-Editor hinzugefügt. Sie können die Nummer verwenden oder den Namen.

Ist unter "Einstellung" --> "Siemens Logo! 8 FS4 Modbus Server" aktiviert, ist der Reiter um Digitale Merker, Analoge Merker und VM-Adressen erweitert.

S7 und Modbus(RTU) Telegramme werden gesondert unter "Bearbeiten" --> "S7 Konfiguration"/
"ModbusRTU" erstellt.

Beispiel:

Eingang 0 = "Schalter_1"

I/O Name:

```
If Get_Digital_Input(Schalter_1) = True then
Label.Caption(1) = "Eingang 0 ist gesetzt"
End if
```

I/O Nummer:

```
If Get_Digital_Input(0) = True then
Label.Caption(1) = "Eingang 0 ist gesetzt"
End if
```

[Beispiel Logo!](#)



Liste erstellen

Unter Datei --> Liste erstellen können IO-Listen automatisch erstellt werden.

Beispiel:

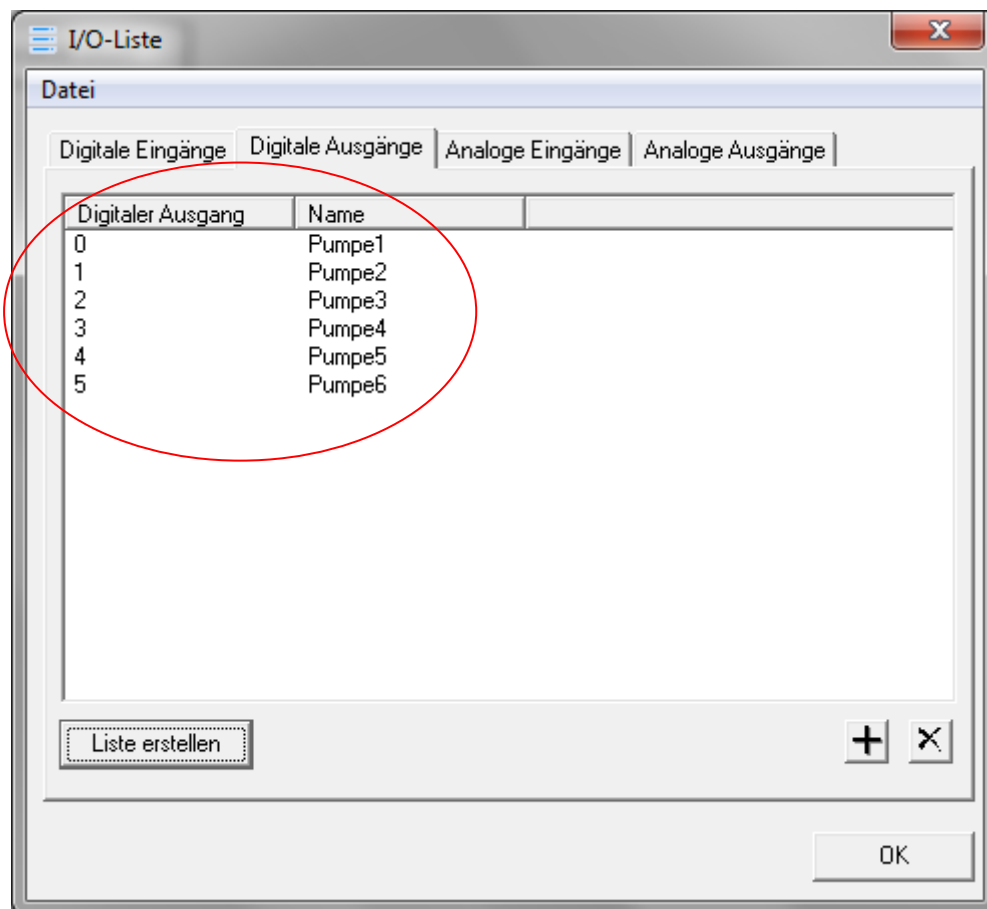
Ab Ausgang: 0

Bis Ausgang: 5

Step: 1

?? Start bei: 1

Erstellt folgende Liste:





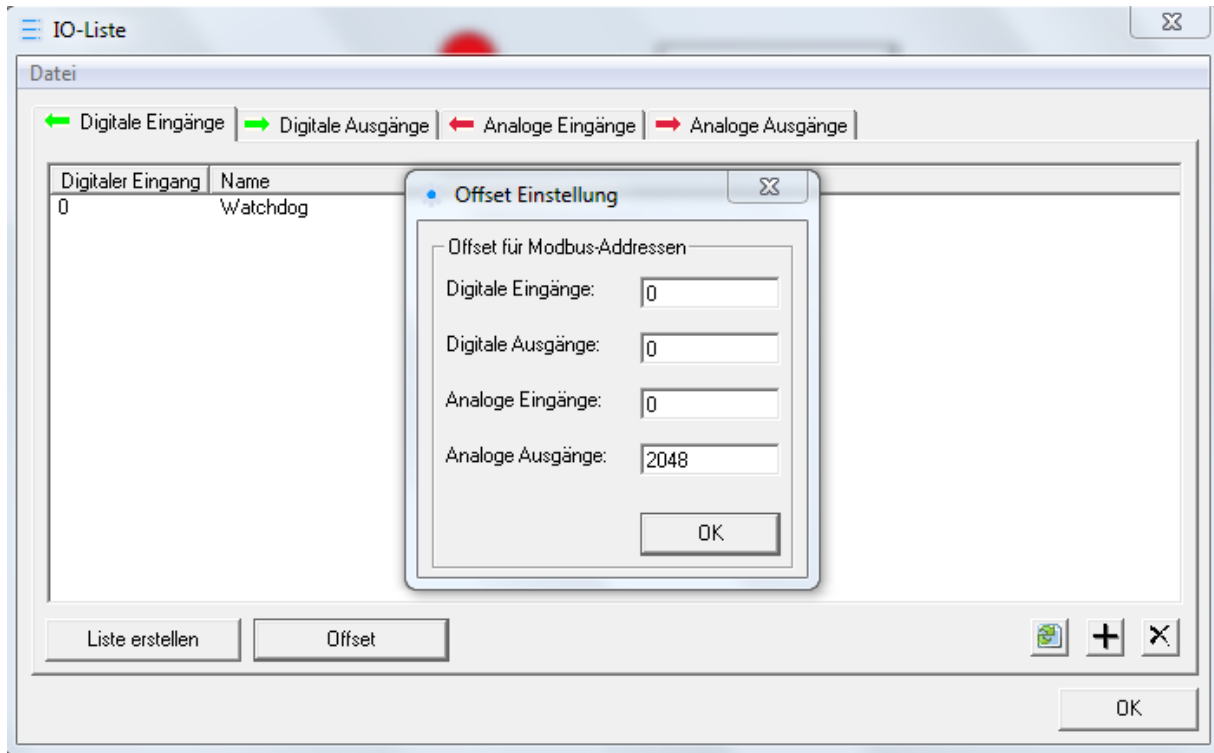
Offset

Im Dialog "IO-Liste" unter "Offset" können Sie einen Offset für die IOs definieren.

Der Offset wird Dezimal angegeben.

Um z.B. im Prozessabbild der BK9050 die Analogen Ausgänge zu schreiben (Holding Registers) geben Sie einen Offset von 2048 (Hex 0800) ein.

Ist eine Logo! ausgewählt sind die Offset fest vergeben.

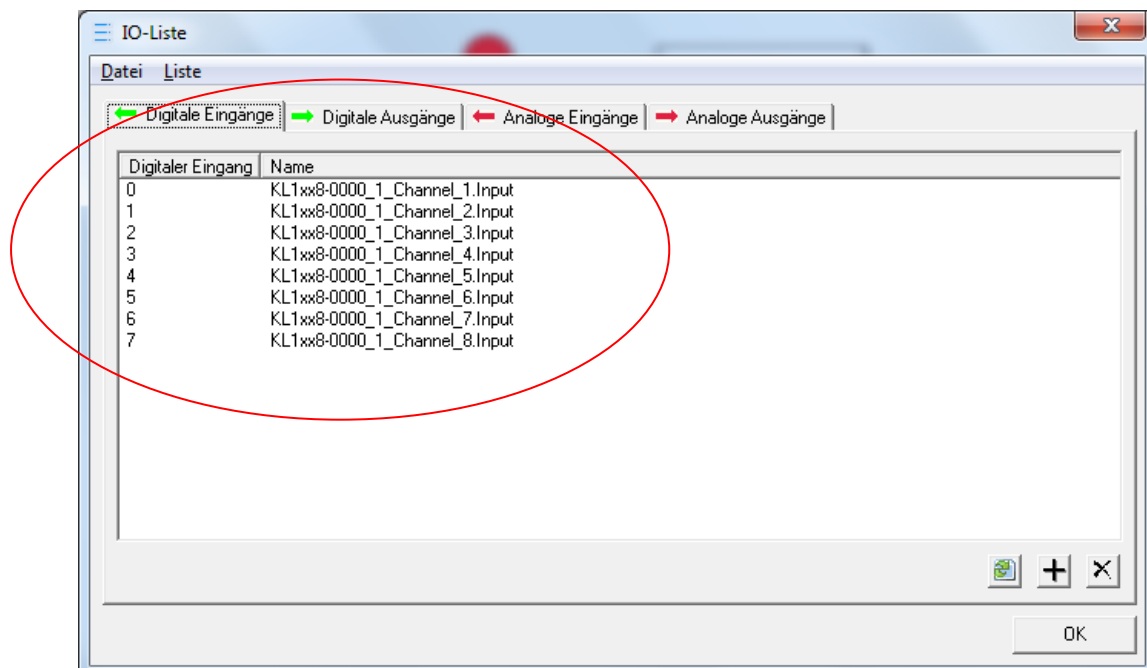
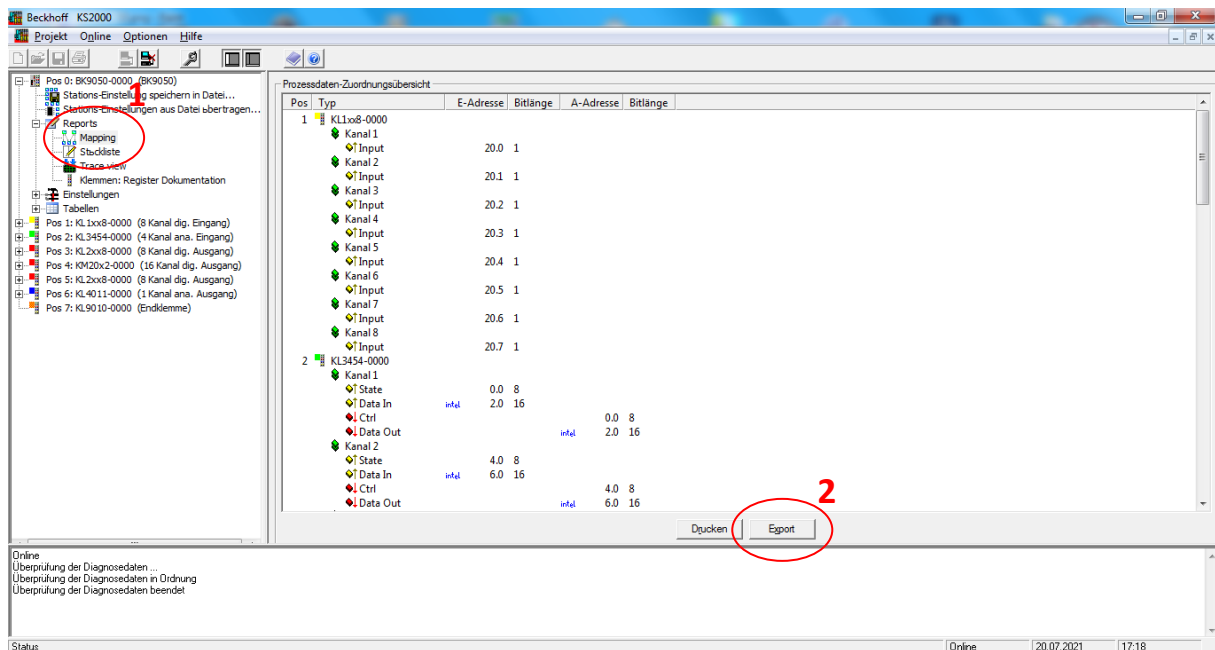




Mapping

Um das Mapping in ViMod importieren zu können, gehen Sie wie folgt vor:

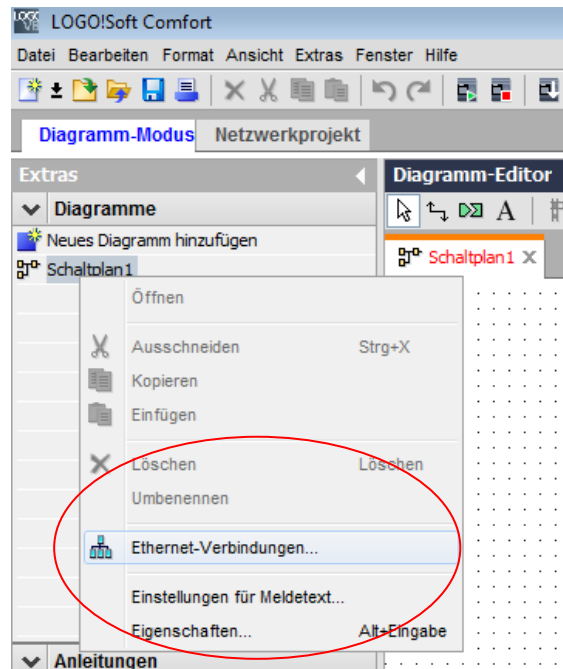
- Stellen Sie eine Verbindung mit der Steuerung her
- Öffnen Sie den Reiter "Reports" --> "Mapping "
- Wählen Sie Export aus
- Speichern Sie die Datei an einem beliebigen Ort auf Ihrer Festplatte
- Starten Sie ViMod und gehen über "Bearbeiten" --> "IO-Liste" --> "Datei" auf "*.xml importieren"
- Wählen Sie die exportierte *.xml Datei aus
- ViMod erstellt jetzt automatisch die IO-Liste aus der Mapping Datei



Siemens Logo!

Modbus-Server konfigurieren 8 FS4

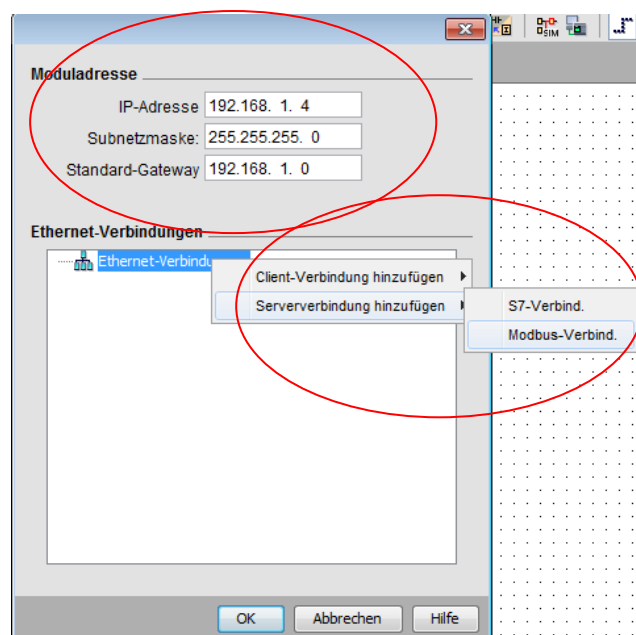
Starten Sie Logo!Soft Comfort und erstellen einen neuen Schaltplan. Klicken sie mit der rechten Maustaste auf den Schaltplan und wählen "Ethernet-Verbindungen" aus.



Tragen Sie unter "Moduladresse" die IP-Adresse, die Subnetzmaske und das Standard-Gateway ein. Danach klicken Sie mit der rechten Maustaste auf "Ethernet-Verbindungen" --> "Serververbindung hinzufügen" und dann auf "Modbus-Verbind."

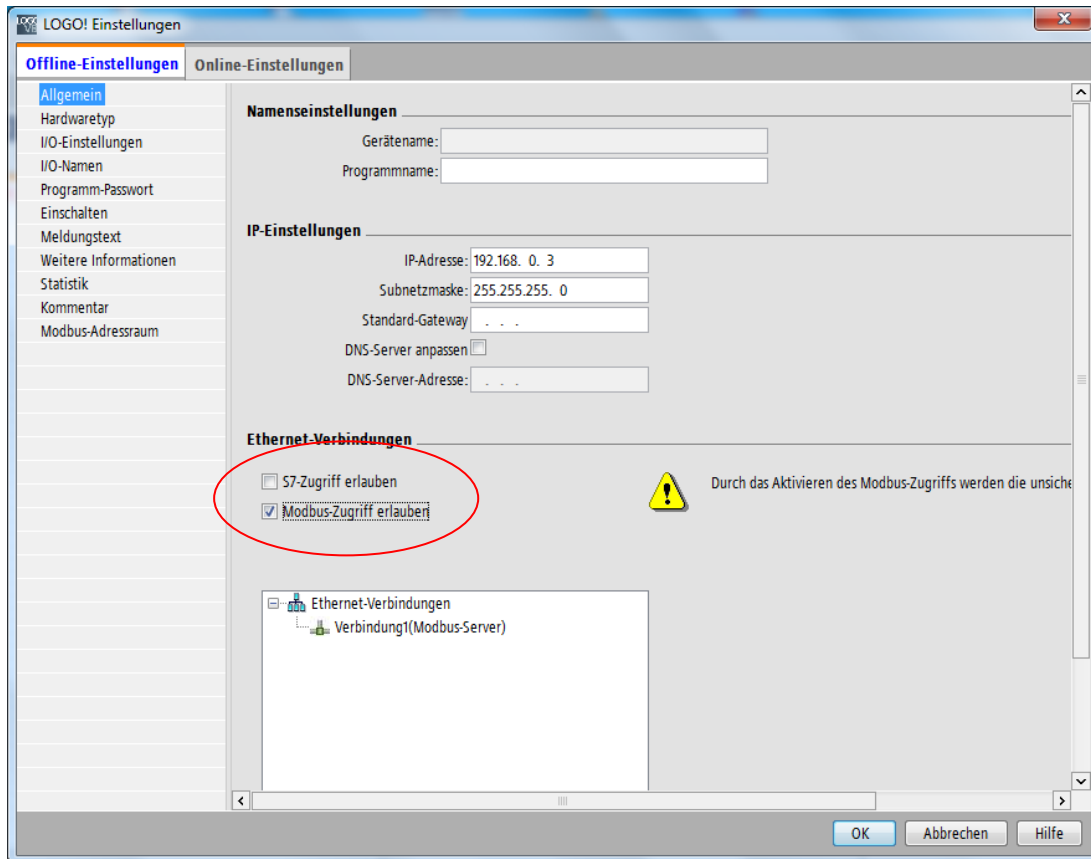
Es erscheint der Eintrag "Verbindung1(Modbus-Server) "

Durch einen Doppelklick auf den Eintrag können der Port oder nur zugelassene IP-Adressen eingestellt werden. Schließen Sie das Fenster mit "OK".



Modbus-Server konfigurieren 8.3

Aktivieren Sie zusätzlich "Modbus-Zugriff erlauben"



Eingänge/Ausgänge/Merker

Es werden automatisch folgende Eingänge/Ausgänge/Merker eingelesen:

| IO | Adresstyp | Anzahl | R/W | Script-Editor |
|-------------------|-----------|--------|-----|--|
| Digitale Eingänge | I | 1-24 | R | Get_Digital_Input |
| Digitale Ausgänge | Q | 1-20 | R/W | Get_Digital_Output Set_Digital_Output |
| Analoge Eingänge | AI | 1-8 | R | Get_Analog_Input |
| Analoge Ausgänge | AQ | 1-8 | R/W | Get_Analog_Output Set_Analog_Output |
| Digitale Merker | M | 1-64 | R/W | Get_Digital_Merker Set_Digital_Merker |
| Analoge Merker | AM | 1-64 | R/W | Get_Analog_Merker Set_Analog_Merker |
| VM-Adressen | V,VW | 0-850 | R/W | Get_VMAdresse Set_VMAdresse |

[Beispiel](#)



Logo! Registeradressen

| Typ | Modbus Typ | Anzahl | Adresse | Funktioncode |
|-------------------|------------------------|--------|------------------------------|--------------|
| Digitale Eingänge | Discrete Input | 1-24 | 0-23 | 04 |
| Digitale Ausgänge | Coil | 1-20 | 8192-8211 | 01 / 05 / 15 |
| Analoge Eingänge | Input Register | 1-8 | 0-7 | 04 |
| Analoge Ausgänge | Holding Register | 1-8 | 512-519 | 03 / 06 / 16 |
| Digitale Merker | Coil | 1-64 | 8256-8319 | 01 / 05 / 15 |
| Analoge Merker | Holding Register | 1-64 | 528-591 | 03 / 06 / 16 |
| VM-Adressen | Holding Register/Coils | 0-850 | 0-6807 (Bit) 0-425 (Word) | 03 / 06 / 16 |

VM-Adressen

VM-Adressen müssen je nach Vergabe in der Logo!Soft Comfort eingelesen werden. Die vergebenen Adressen können direkt in ViMod übernommen werden.

[illegible]

IO-Liste

Datei

← Analoge Eingänge → Analoge Ausgänge → Digitale Merker → Analoge Merker → VM-Adressen

| Modbus-Adresse | Typ | Name |
|----------------|-------|-----------------|
| 0 | Word | B006_V1 |
| 2 | Word | B006_V2 |
| 4 | DWord | B007_Zähler |
| 8 | Byte | B004_Wochentag1 |
| 9 | Word | B006_V3 |

Liste erstellen

OK



Spezielle VM-Adressen

| VM-Adresse | Beschreibung | Größe |
|------------|-------------------|--------|
| 984 | Diagnosebit-Array | 1 Byte |
| 985 | Jahr (RTC) | 1 Byte |
| 986 | Monat (RTC) | 1 Byte |
| 987 | Tag (RTC) | 1 Byte |
| 988 | Stunde (RTC) | 1 Byte |
| 989 | Minute (RTC) | 1 Byte |
| 990 | Sekunde (RTC) | 1 Byte |
| 998 | Wochentag | 1 Byte |

Logo! Beispiel

- Den Status von Eingang I1 auslesen und den Kreis bei 1 grün färben und bei 0 rot:

```
If Get_Digital_Input(Eingang_1) = True then
Kreis.FillColor(1) = VBGreen
else
Kreis.FillColor(1) = VBRed
End If
```

- Den Status von Ausgang Q1 auslesen und den Kreis bei 1 grün färben und bei 0 rot:

```
If Get_Digital_Output(1) = True then
Kreis.FillColor(2) = VBGreen
else
Kreis.FillColor(2) = VBRed
End If
```

- Den Status von Merker M1 auslesen und den Kreis bei 1 grün färben und bei 0 rot:

```
If Get_Digital_Merker(1) = True then
Kreis.FillColor(3) = VBGreen
else
Kreis.FillColor(3) = VBRed
End If
```

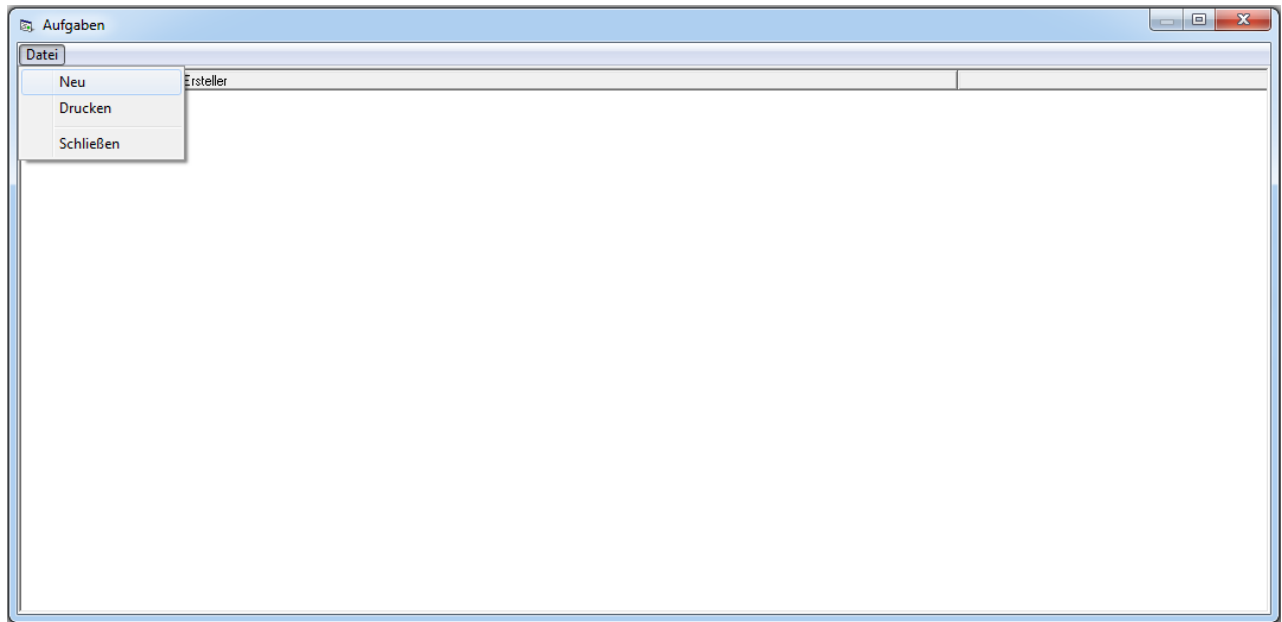
- Den Zählerwert aus der VM-Zuordnung auslesen und in ein Label schreiben:

```
Label.Caption(4) = Get_VMAdresse("Zaehler")
```

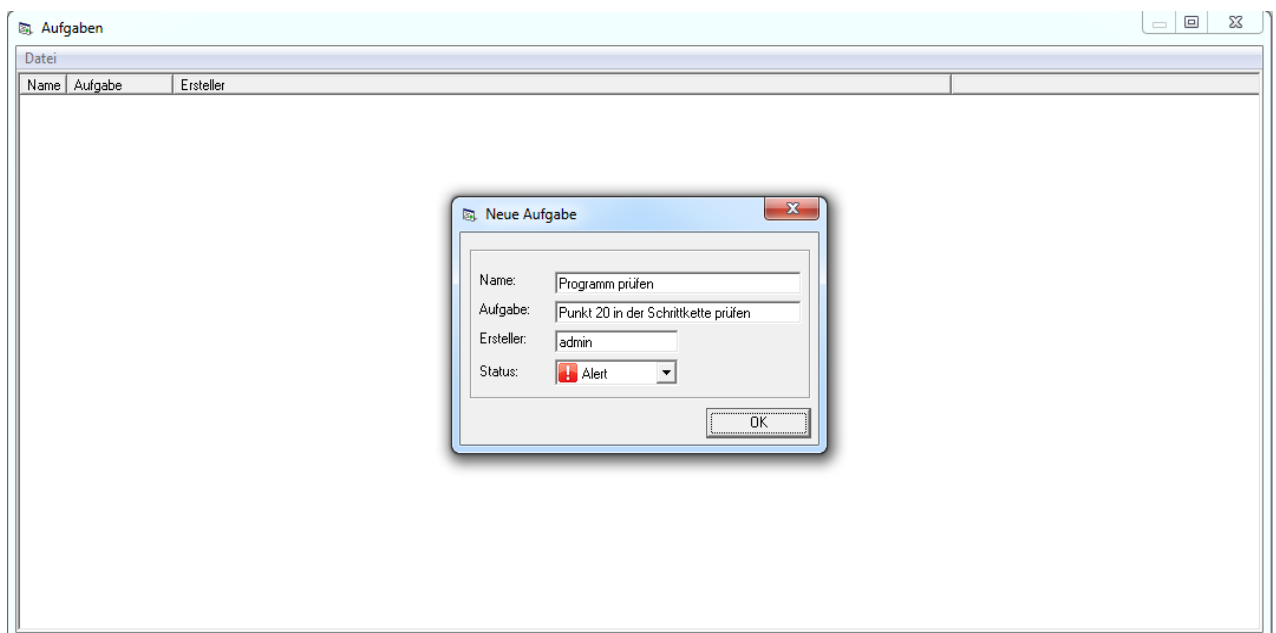


Aufgaben

Öffnen Sie unter "Bearbeiten" --> Aufgaben die Aufgabenverwaltung.
Um eine neue Aufgabe hinzufügen, gehen Sie über "Datei" --> "Neu"



Geben Sie ihre Aufgabe ein und bestätigen Sie mit "OK"



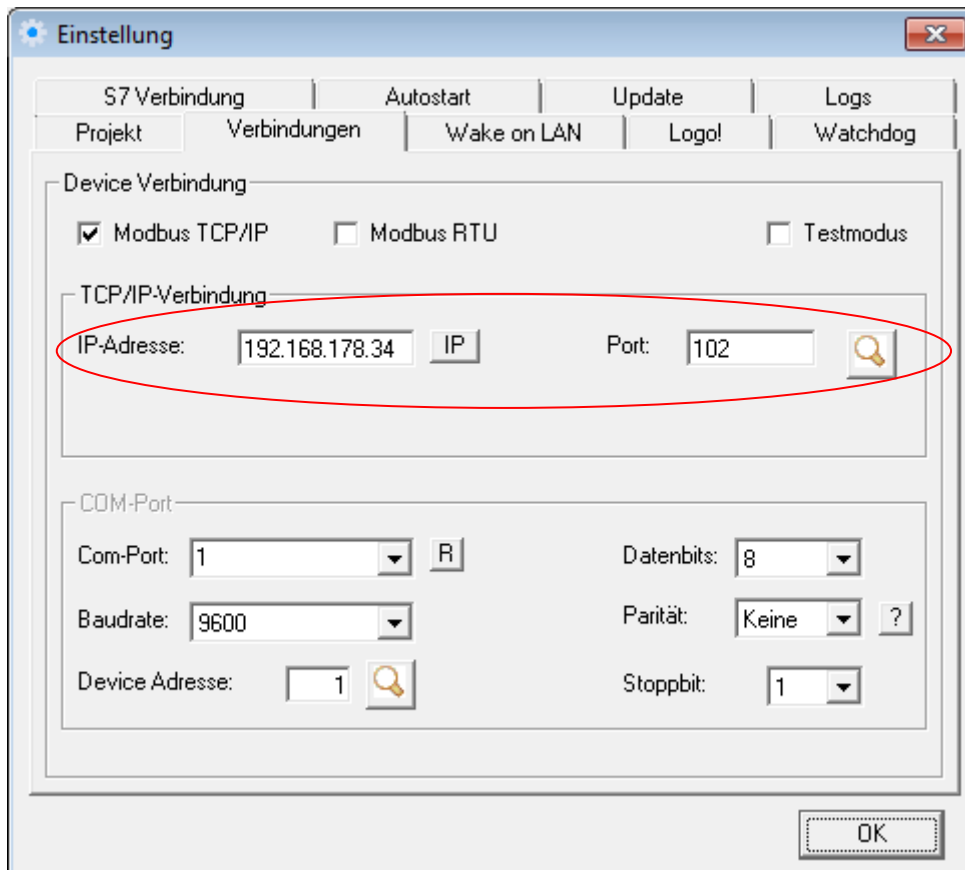
S7-Verbindung

S7-Konfigurieren

Wenn Sie eine S7-Verbindung konfigurieren wollen, gehen Sie wie folgt vor.

Achtung!!

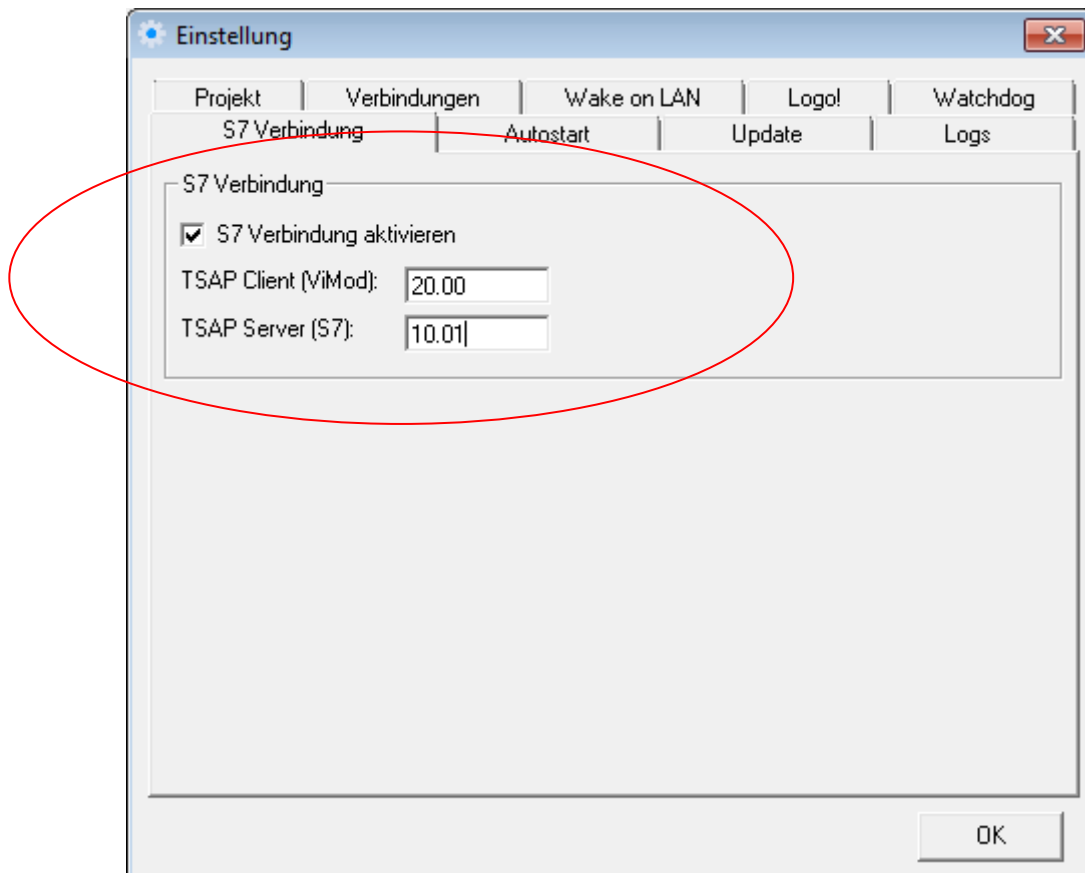
Der Zugriff muss im TIA-Portal konfiguriert sein. Die IP-Adresse und die TSAP muss bekannt sein.
ViMod stellt den Client da und die S7 den Server.



The screenshot shows the 'Einstellung' (Settings) window with the 'S7 Verbindung' tab selected. The 'Device Verbindung' section has 'Modbus TCP/IP' checked. The 'TCP/IP-Verbindung' section is highlighted with a red oval, showing the IP address '192.168.178.34' and the port '102'. The 'COM-Port' section shows 'Com-Port: 1', 'Baudrate: 9600', and 'Device Adresse: 1'. The 'OK' button is at the bottom right.

Geben Sie unter "Datei" --> "Einstellung" --> "Verbindungen" die IP-Adresse ein. Der Port ist immer "102".

TSAP

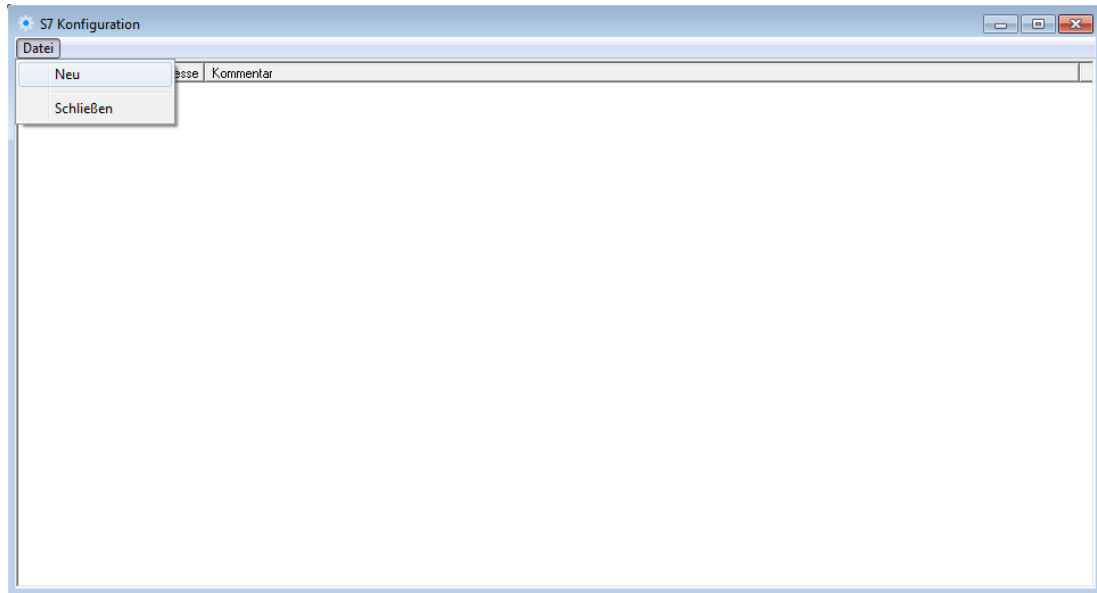


Gehen Sie unter "S7-Verbindung" und aktivieren Sie "S7 Verbindung aktivieren" und geben Sie die TSAP ein.



S7 Telegramme erstellen

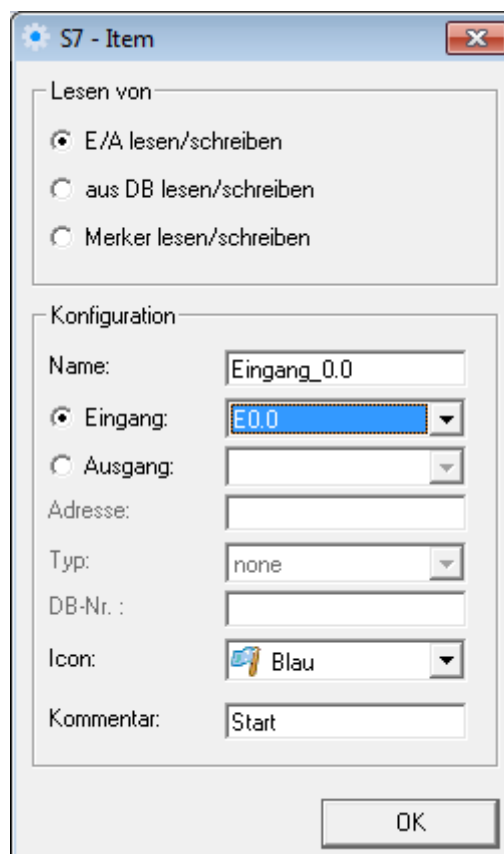
Öffnen Sie unter "Bearbeiten" --> " S7 Konfiguration". Danach erstellen Sie unter "Datei" --> "Neu" ein neues Telegramm.

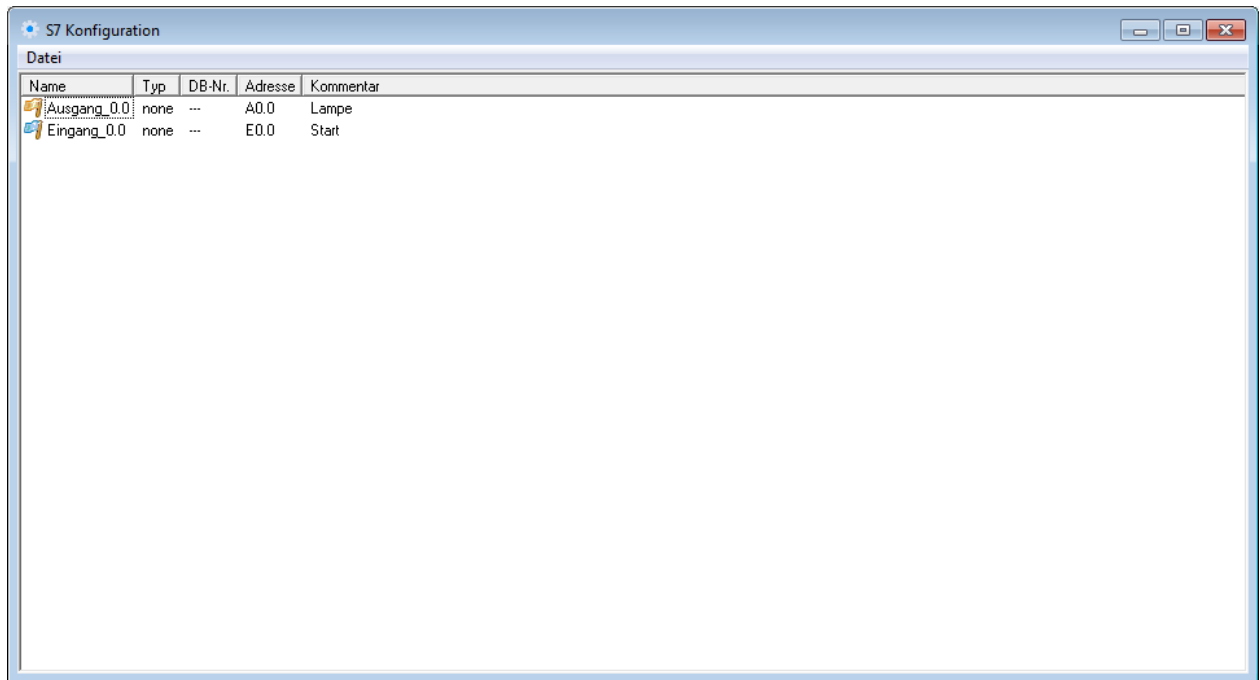


Im folgenden Beispiel. Wird der Eingang 0.0 Abgefragt.

Achtung!!

Eingänge und Ausgänge werden immer direkt abgefragt. [Zykluszeit](#)





Hier wurde jetzt noch der Ausgang 0.0 eingefügt.

Information im Script-Editor

| Befehl | Beschreibung |
|---|---|
| <code>S7_Read("Name") as Double</code> Beispiel: <code>If S7_Read("Eingang_0.0") = 1 then</code> <code>S7_Write("Ausgang_0.0") = 1</code> <code>End if</code> | Liest den aktuellen Wert von "Name" aus und gibt ihn als Typ Double wieder. |
| <code>S7_Write("Name") as Double</code> Beispiel: <code>Sub Command_Click1()</code> <code>S7_Write("Ausgang_0.0") = 0</code> <code>End Sub</code> | Schreibt den aktuellen Wert von "Name" |
| <code>S7_List(ListIndex) Return as Boolean</code> Beispiel: <code>If S7_List(1) = False then</code> <code>MsgBox "Liste konnte nicht erstellt werden"</code> <code>End if</code> | Gibt die Namen der erstellten S7 Items in einer Liste zurück. |

Weitere Beispiele auf <https://nicos-alltag.blogspot.com/>



ViMod S7 Error-Codes

| Code | Beschreibung |
|------|--|
| 1 | ungültige Telegrammlänge TPKT.Length <> S7_Temp_Data Länge |
| 2 | Protocol Data Unit Reference der Antwort entspricht nicht der Protocol Data Unit Reference der Anfrage |
| 3 | Area 0x81 (Inputs) Antwort mit falschem Funktion Code |
| 4 | Area 0x81 falscher Item Count |
| 5 | Area 0x81 Kein Success |
| 6 | Area 0x81 Falsche Transport Size |
| 7 | Area 0x81 Falsche Anzahl DatenBytes |
| 8 | Area 0x82 (Digital Outputs) Read Antwort mit falschem Funktion Code |
| 9 | Area 0x82 Read falscher Item Count |
| 10 | Area 0x82 Read Kein Success |
| 11 | Area 0x82 Read Falsche Transport Size |
| 12 | Area 0x82 Falsche Anzahl DatenBytes |
| 13 | Area 0x82 (Digital Outputs) Write Antwort mit falschem Funktion Code |
| 14 | Area 0x82 Write falscher Item Count |
| 15 | Area 0x82 Write Kein Success Item[1] |
| 16 | Area 0x82 Write Kein Success Item[2] |
| 17 | Area 0x82 Write Kein Success Item[3] |
| 18 | Area 0x82 Write Kein Success Item[4] |
| 19 | Area 0x82 Write Kein Success Item[5] |
| 20 | Area 0x82 Write Kein Success Item[6] |
| 21 | Area 0x82 Write Kein Success Item[7] |
| 22 | Area 0x82 Write Kein Success Item[8] |
| 23 | Area 255 (Items Write) Write Antwort mit falschem Funktion Code |
| 24 | Area 255 falscher Item Count |
| 25 | Area 255 Kein Success Item[i] |
| 26 | Area 256 (Items Read) Write Antwort mit falschem Funktion Code |
| 27 | Area 256 falscher Item Count |
| 28 | Area 255 Kein Success Item[i] |
| 29 | Area 256 ungültige Data Transport size |
| 30 | Area 256 ungültige Bitlänge der Daten |

Modbus RTU (Remote Terminal Unit)

Unter "Bearbeiten" ---> "ModbusRTU" können Sie Telegramme nach dem Modbus Protokoll erstellen.

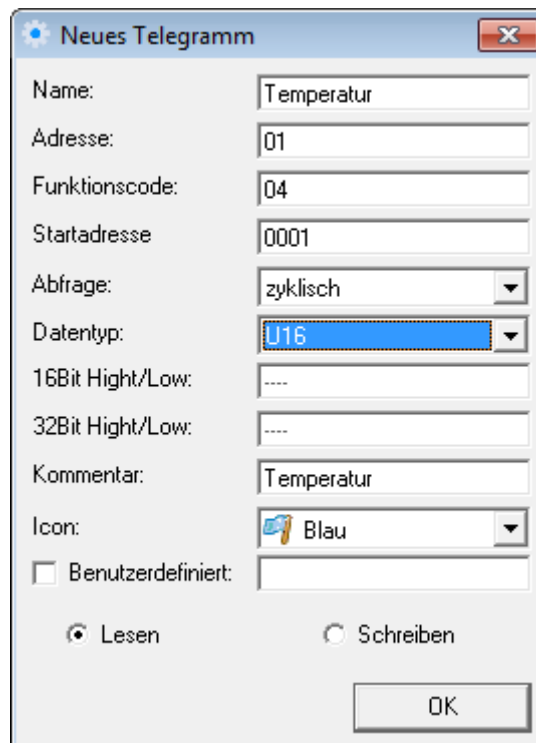
Diese werden in Hexadezimal dargestellt und versendet. Die Länge des Telegramm hängt von dem verwendeten Funktionscode ab. Die CRC (cyclic redundancy check) Checksumme wird automatisch mit dem Polynom "A001" berechnet.

Der grundsätzliche Aufbau sieht wie folgt aus:

| | | | | |
|---------|---------------|----------|-------|-----|
| Adresse | Funktionscode | Register | Daten | CRC |
|---------|---------------|----------|-------|-----|

Neues Telegramm erstellen:

Klicken Sie auf Datei ---> Neu



In diesem Beispiel wird eine Temperatur von einem Sensor zyklisch ausgelesen.
Das Telegramm sieht wie folgt aus:

| Adresse vom Device | Funktionscode | Startadresse | Anzahl Register | CRC |
|--------------------|---------------|--------------|-----------------|------|
| 01 | 04 | 0001 | 0001 | 600A |



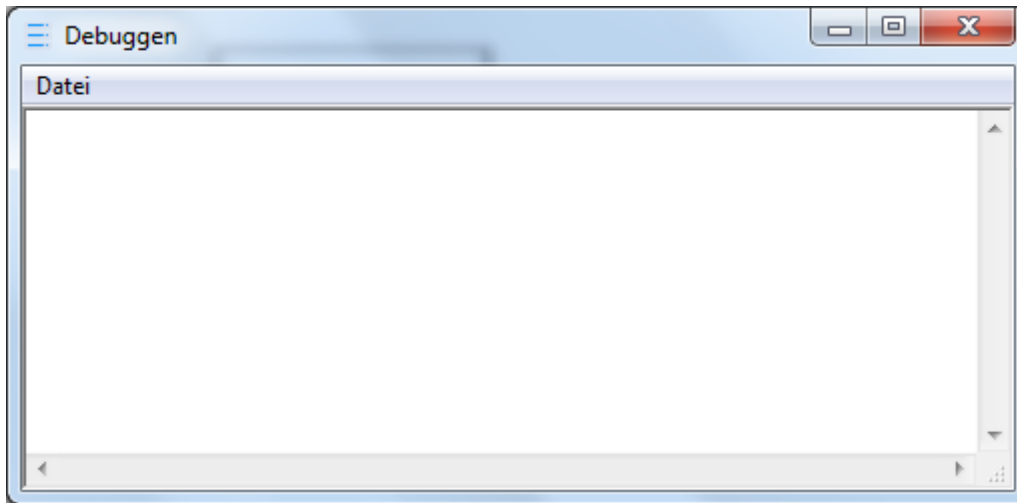
Informationen im Script Editor verarbeiten

| Funktion | Beschreibung |
|--|--|
| <code>ModbusRTU_Abfrage(WertName)</code> Beispiel: <code>ModbusRTU_Abfrage("Temperatur")</code> | Diese Funktion sendet das Telegramm das unter dem "WertName" hinterlegt ist. Die Abfrage muss in der ModbusRTU Tabelle unter "manuell" eingestellt sein. |
| <code>ModbusRTU_Read(WertName, Wert, Data_Format)</code> Beispiel: <code>Label.Caption(1) = ModbusRTU_Read("Temperatur", 4, "Dec")</code> | Diese Funktion liest ein bestimmten Hexadezimalwert aus einer Antwort aus und gibt Sie im eingestellten Format zurück.* |
| <code>ModbusRTU_Data(WertName, Data_Format)</code> Beispiel: <code>Label.Caption(1) = ModbusRTU_Data("Temperatur", "Dec")</code> | Diese Funktion gibt den gesamten Registerwert aus einer Antwort zurück (16/32Bit) und gibt Sie im eingestellten Format an.* |

*Dec: Dezimalwert, Hex: Hexadezimalwert, Bin: Binärwert

Weitere Beispiele auf <https://nicos-alltag.blogspot.com/>

Debugger



| Funktion | Beschreibung |
|--|---|
| <code>Debugger.DebugShow</code> Beispiel: <code>Debugger.DebugShow = True</code> | Zeigt das Debugger Fenster an bzw. schließt es True = sichtbar False = nicht sichtbar |
| <code>Debugger.DebugClear</code> Beispiel: <code>Debugger.DebugClear</code> | Löscht den Inhalt des Debugger Fenster |
| <code>Debugger.DebugPrint</code> Beispiel: <code>Debugger.DebugPrint = "Fehler"</code> | Schreibt etwas ins Debugger Fenster |

Achtung!!!

- Der Inhalt kann nicht über ViMod gespeichert werden, wenn das Program im Run ist.
- Wird das Debugger Fenster über den Windows schließen Dialog, oben rechts geschlossen, wird der Inhalt gelöscht.
- Der Debugger ist nur verfügbar, wenn man als Administrator angemeldet ist.



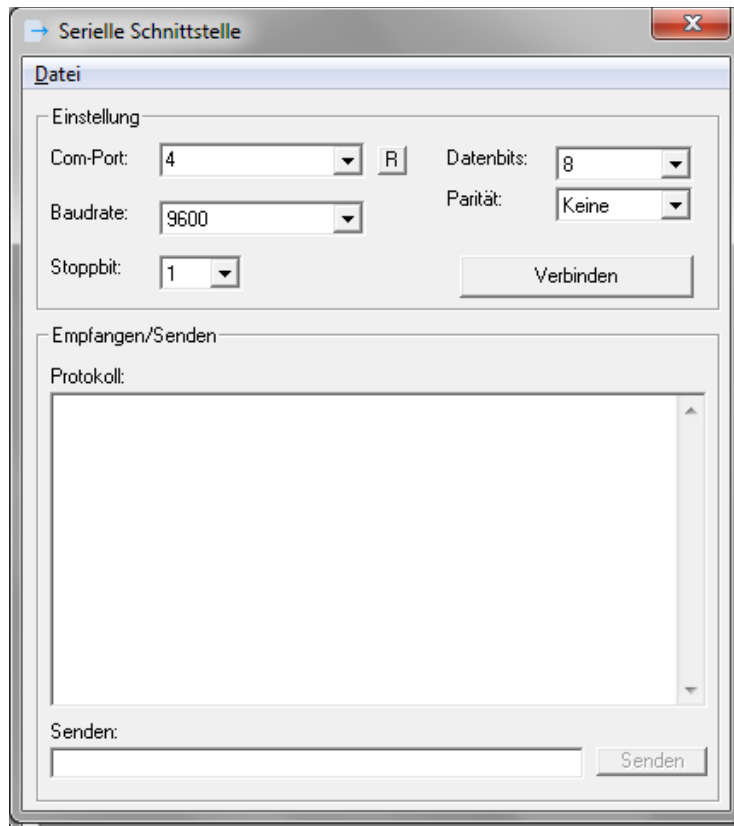
Datenbank



Mit dem Datenbank-Viewer können Sie Datenbanken erstellen und sich ihren Inhalt anzeigen lassen. ViMod kann nur Datenbanken ohne Passwort laden. Sie können mit dem Datenbank -Viewer keine Änderungen an der Datenbank vornehmen. Dies ist nur programmtechnisch über den Script-Editor möglich.



Serielle Schnittstelle

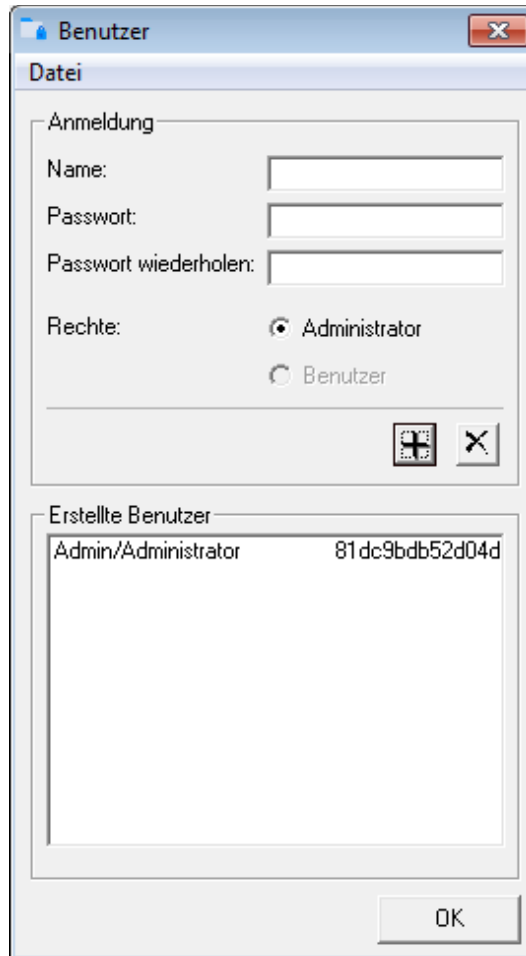


Hier können Sie eine serielle Schnittstelle konfigurieren und testen.
Über Datei --> Einstellung übernehmen werden die Einstellungen ins Projekt übernommen. Die konfigurierte Schnittstelle wird unten in der Infoleiste angezeigt.

Funktionen

| Funktion | Beschreibung |
|---|--|
| <code>Connect_COM()</code> Beispiel: <code>Connect_COM = True</code> | Öffnet die Verbindung zum Device True = Stellt die Verbindung her False= Schließt die Verbindung |
| <code>ISConnect_COM()</code> Beispiel: <code>If ISConnect_COM = True then</code> <code>Label.Caption(1) = "Verbunden"</code> <code>End if</code> | Gibt einen True-/False-Wert zurück True = Verbunden False = Nicht Verbunden |
| <code>Output_COM()</code> Beispiel: <code>Output_COM = "Test"</code> | Sendet Information über den COM-Port als String an das Device |
| <code>Input_COM()</code> Beispiel: <code>Label.Caption(1) = Input_COM</code> | Gibt die Daten von Device als String zurück |

Benutzerkonten



Benutzer

Datei

Anmeldung

Name:

Passwort:

Passwort wiederholen:

Rechte: ☒ Administrator
☐ Benutzer

Erstellte Benutzer

| | |
|---------------------|----------------|
| Admin/Administrator | 81dc9bdb52d04d |
|---------------------|----------------|

OK

Hier können Sie neue Benutzer anlegen bzw. löschen. Es muss immer mindestens ein Administratorkonto geben. Die Passwörter werden verschlüsselt bzw. als Hash gespeichert.

Achtung!!

Ein vergessenes Passwort kann man nicht wiederherstellen.

Achtung!!

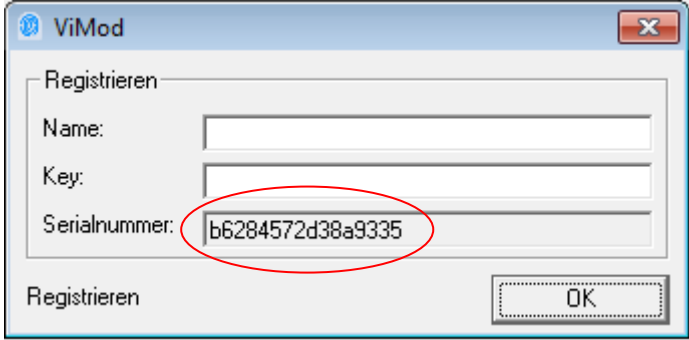
Es können maximal 5 Benutzer angelegt werden

Registrieren

Die Verwendung von ViMod ist kostenlos. Die Registrierung soll nur verhindern, dass ViMod kommerziell genutzt wird. Ohne eine Lizenz wird beim Start und in der Titelleiste "ViMod Demo" angezeigt. Dies hat keine Auswirkungen auf die Funktionen.

Senden Sie uns die individuelle Seriennummer und einen Name zu und Sie erhalten eine Lizenz-Datei. Die Seriennummer ist PC gebunden und wird aus der verwendeten Hardware berechnet. Aus der Nummer kann man keine Rückschlüsse auf die Hardware beziehen. Sie besteht aus einem Teil-Hash.

Die Seriennummer finden Sie unter "? -->About" oder "? --> Registrieren".
Die Daten werden nicht an Dritte weitergegeben.



The image shows a Windows-style dialog box titled "ViMod". It contains a section titled "Registrieren" with three input fields: "Name:", "Key:", and "Seriennummer:". The "Seriennummer:" field contains the text "b6284572d38a9335" and is circled in red. At the bottom left is a button labeled "Registrieren" and at the bottom right is a button labeled "OK".



FAQ

Q: Warum benötigt ViMod Adminrechte?

A: Um die Einstellungen in die "settings.ini" zuschreiben bzw. um die Datei zu erstellen.

Q: Welche Daten werden bei der Registrierung gespeichert?

A: Name, Key, Seriennummer und E-Mail Adresse werden offline gespeichert. Die Daten werden nicht an Dritte weitergegeben oder anderweitig verwendet.

Q: Auf welchem Betriebssystem kann ViMod verwendet werden?

A: Getestet unter Windows 7/8/10.

Q: Ist ViMod eine 32Bit oder eine 64Bit Anwendung?

A: Aus historischen Gründen, ist ViMod eine 32Bit Anwendung.



Bekannte Probleme

- ViMod braucht um sich zu konfigurieren beim ersten Start länger.
- Bei einigen Windows Versionen kann es zu grafischen Fehlern bei den Steuerelementen und im Script-Editor kommen. Dies schränkt die Funktionen aber nicht ein.



Versionsverlauf

v0.3.5 Beta

- Aufgabenverwaltung hinzugefügt

v0.3.4 Beta

- "IO-Check" hinzugefügt"

v0.3.3 Beta

- ViMod mit Windows starten
- ViMod kann verhindern, dass Windows herunterfährt/neu gestartet wird

v0.3.2 Beta

- Automatischer Projekt Start möglich
- Bugfix "Steuerelemente ändern/speichern

v0.3.1 Beta

- PopUp hinzugefügt
- Bugfix

v0.3.0 Beta

- Zwei weitere Forms können verwendet werden
- Für die Hauptform können Vorlagen gespeichert /geladen werden

v0.2.9 Beta

- Funktion "S7_List" hinzugefügt

v0.2.8 Beta

- Bugfix einlesen von DB und Merker
- Weitere Funktion im S7 Konfigurator hinzugefügt

v0.2.7 Beta

- Befehl "Output_COM angepasst
- Admin Kontakt kann hinterlegt werden
- S7 Verbindung hinzugefügt

v0.2.6 Beta

- Set_SpecialAddress bzw. Get_SpecialAddress Funktion hinzugefügt (Logo!)



v0.2.5 Beta

- Logs eingefügt
- Bug "VM-Adressen dWord" behoben

v0.2.4 Beta

- BK9050 Watchdog Einstellung überarbeitet
- Dokumentation erweitert
- KS2000 Mapping Export kann in ViMod importiert werden
- Neue Funktionen im Script-Editor hinzugefügt

v0.2.3 Beta

- Bug "TextBox kopieren" behoben
- Bug "VM-Adressen sortieren" behoben
- Einzelne IOs können bearbeitet werden
- Offset Einstellung für Modbus-Adressen
- `Set_SpecialAddress` bzw. `Get_SpecialAddress` Funktion hinzugefügt

v0.2.2 Beta

- Geschwindigkeitsoptimierung

v0.2.1 Beta

- Vollbildmodus hinzugefügt
- Bugfix

v0.2.0 Beta

- ModbusRTU hinzugefügt
- Debugger Fenster hinzugefügt
- Neue Funktionen im Script-Editor hinzugefügt
- "Sub" und "Function" werden in einer Auswahl aufgelistet

Achtung!! !Der Code im Script Editor muss eventuell angepasste werden!!!

v0.1.22 Beta

- Steuerelemente kopieren
- "ProgressBar" Steuerelement hinzugefügt
- "List" Steuerelement hinzugefügt
- "Allgemeine Eigenschaften" erweitert
- Button IO-Listen aktualisieren/neu ordnen
- Bug "Rechteck" behoben
- Bug "Slider" behoben



[v0.1.21 Beta](#)

- "Slide" Steuerelement hinzugefügt
- Neue Funktionen im Script-Editor hinzugefügt
- IO-Listen werden sortiert

[v0.1.20 Beta](#)

- Message Box Generator hinzugefügt
- Syntax Highlighting verbessert
- Bug "Liste erstellen" behoben
- Bug beim laden eines Projektes behoben

[v0.1.19 Beta](#)

- Datenbankzugriff verbessert
- Update-Funktion hinzugefügt
- Ansicht der Einstellungen geändert
- "Historie" hinzugefügt
- Projektversion hinzugefügt
- Projektbeschreibung hinzugefügt
- Benutzerkonten Bug behoben
- Bugfix

[v0.1.18 Beta](#)

- Farbcode-Editor im Script-Editor hinzugefügt
- Projekt als ZIP-Datei exportierbar
- Fehlerbehandlung im Script verbessert
- Analog-Meter hinzugefügt
- Übersicht der Funktionen im Script-Editor hinzugefügt
- PNG-Dateien werden unterstützt
- "ExitScript"-Funktion hinzugefügt
- Neue Funktionen im Script-Editor hinzugefügt
- Testmodus hinzugefügt
- Bugfix

[v0.1.16 Beta](#)

- Siemens Logo! 8 FS4 Modbus Server Unterstützung
- Bugfix

[v0.1.15 Beta](#)

- COM-Port-Verbindung implementiert (Device-Verbindung)
- Auslesen der Eingänge/Ausgänge optimiert
- IO Status eingefügt
- IO-Liste automatisch erstellen
- Bugfix



ViMod Version 0.3.X Beta

[v0.1.10 Beta](#)

First Release