



# Axeda Supervisor™

The complete Internet-based solution for control and information

## WizPLC 3.0

## Erste Schritte

Dezember 2004



Axeda Supervisor™

**Axeda Systems Inc.**  
**Global Headquarters**  
21 Oxford Road  
Mansfield, MA 02048  
USA  
tel +1 (508) 337-9200  
fax +1 (508) 337-9201

**Axeda Systems SAS**  
**Headquarters Europe & Asia**  
Parc Technologique de Lyon  
12 allée Irène Joliot-Curie  
F-69791 Saint-Priest Cedex  
France  
tel +33 (0)4 72 47 98 98  
fax +33 (0)4 72 47 98 99

**Axeda Systems Ltd**  
The Gate Hotel  
Scotland Gate  
Northumberland  
NE62 5SS  
UK  
tel +44 (0)845 606-6120  
fax +44 (0)845 606-6121

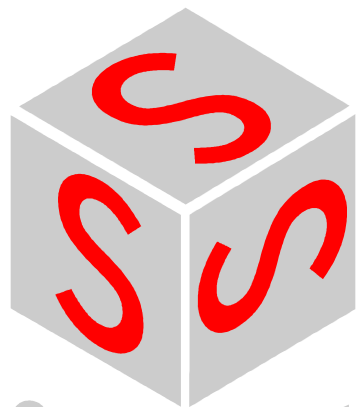
**Axeda Systems BV**  
Concordiaweg 149-151  
Postbus 351  
NL-4200 AJ Gorinchem  
Nederland  
tel +31 (0)183 646 303  
fax +31 (0)183 621 601

[www.axedasupervisor.com](http://www.axedasupervisor.com)



## Erste Schritte mit CoDeSys

Stand: 05.03.2004



S m a r t  
Software  
Solutions



## **INHALT**

<b>1</b>	<b>START VON CODESYS</b>	<b>4</b>
<b>2</b>	<b>WIR SCHREIBEN DAS ERSTE PROGRAMM</b>	<b>4</b>
<b>3</b>	<b>EINE VISUALISIERUNG DAZU</b>	<b>8</b>
<b>4</b>	<b>ZIELSYSTEM STARTEN</b>	<b>10</b>
<b>5</b>	<b>EINSTELLUNGEN ZUM VERBINDUNGSaufbau</b>	<b>10</b>
<b>6</b>	<b>STARTEN DES PROJEKTES</b>	<b>10</b>
<b>7</b>	<b>WEITERES VORGEHEN</b>	<b>11</b>



## 1 Start von CoDeSys

Starten Sie das Programmiersystem **CoDeSys** über **Start -> Programme -> 3S Software -> CoDeSys V2.3 -> CoDeSys V2.3**

## 2 Wir schreiben das erste Programm

- **Aufgabe:**

Es soll eine laufende Maschine von einem Maschinenführer beobachtet werden. Der korrekte Verlauf muss in einem bestimmten Zeitabstand immer wieder bestätigt werden. Erfolgt keine Bestätigung, wird zuerst eine Warnung ausgegeben und kurze Zeit später die Maschine gestoppt. Die Maschine bewegt einen Arm im Rechteck und die vollständigen Umläufe sollen gezählt werden.

- **Anlegen eines neuen Projektes**

Aller Anfang ist leicht. Legen Sie ein neues Projekt an mit dem Befehl **Datei -> Neu**.

- **Zielsystem Einstellungen**

Wählen Sie für unser erstes Programm bitte in der Combobox die Ihrem Zielsystem entsprechende Konfiguration aus, z.B. CoDeSys SP for Windows NT Realtime

- **Der Baustein PLC\_PRG**

Wählen Sie im Dialog **Neuer Baustein** die **Sprache des Bausteins FUP** (Funktionsplan) und behalten Sie die Einstellungen **Bausteintyp Programm** und **Name PLC\_PRG** bei. PLC\_PRG ist ein besonderer Baustein, der im laufenden Betrieb zyklisch vom Laufzeitsystem aufgerufen und abgearbeitet wird.

- **Deklaration des Bestätigungsschalters**

Beginnen wir mit dem Bestätigungsschalter. Markieren Sie die drei Fragezeichen **???** im ersten Netzwerk und tippen Sie den Namen des Schalters **Observer** ein. Betätigen Sie nun die Pfeiltaste nach rechts.

Im erscheinenden Dialog zur **Variablendeklaration** ist der **Name Observer** und der **Typ BOOL** aufgeführt. Ändern Sie die **Klasse** auf **VAR\_GLOBAL** (für globale Variablen). Mit **OK** wird die Variable im Objekt für die globalen Variablen eingetragen. Diese Objekt befindet sich im Object Organizer unter Ressourcen \ Globale Variablen \ Globale\_Variablen.

Dort erscheint folgender Text:

```
VAR_GLOBAL
    Observer: BOOL;
END_VAR
```

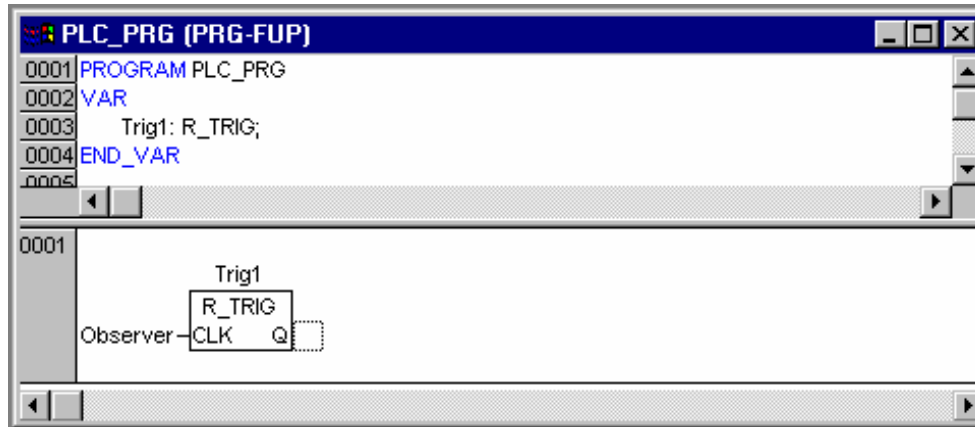
- **Steigende Flanke des Bestätigungsschalters**

Alles ist in Ordnung, wenn der Bediener da ist und den korrekten Ablauf bestätigt. D.h. wenn der Schalter vom Zustand Aus (FALSE) auf An (TRUE) wechselt, genannt steigende Flanke, oder von An auf Aus, genannt fallende Flanke. Wir beginnen damit, die steigende Flanke festzustellen, d.h. ob der Schalter von FALSE auf TRUE wechselt.

Dazu kehren wir zum Baustein PLC\_PRG zurück. Markieren Sie die Stelle hinter der Variablen Observer, so daß eine quadratische Markierung erscheint. Führen Sie über das Kontextmenü (rechte Maus) den Befehl **Baustein** aus.

Es wird standardmäßig der Baustein **AND** eingefügt, der Name ist markiert und kann nun durch den gewünschten Baustein ersetzt werden. Rufen Sie dazu die Eingabehilfe mittels Taste F2 auf. Sie erhalten einen Dialog, in dem Sie aus den verfügbaren Bausteinen auswählen können. Wählen Sie für unser Beispiel zunächst die Kategorie **Standard-Funktionsblöcke** und dort aus der Bibliothek Standard.lib den Trigger **R\_TRIG**.

Von R\_TRIG ist eine Instanz (Datenkopie) anzulegen. Schreiben Sie anstelle der drei Fragezeichen **???**, die über dem eingefügten Baustein erscheinen, den Instanznamen **Trig1** und klicken Sie mit der linken Maustaste hinter den Funktionsblock oder betätigen Sie die rechte Pfeiltaste. Daraufhin erscheint der **Variablendeklarationsdialog**, in dem die **Klasse VAR** (für lokale Variablen), der **Name Trig1** und der **Typ R\_TRIG** bereits eingetragen sind. Wird der Dialog mit **OK** geschlossen, wird die Instanz in den Deklarationsteil des Bausteins eingetragen.



- **Fallende Flanke des Bestätigungsschalters**

Markieren Sie die Stelle hinter dem Funktionsblock (quadratische Markierung), führen Sie über das Kontextmenü den Befehl **Baustein** aus, ändern Sie die Bezeichnung **AND** in **OR** (logisches Oder).

Markieren Sie die Linie des zweiten Eingang des OR-Operators und fügen Sie einen **Baustein** vom Typ **F\_TRIG** für die fallende Flanke mit dem Instanznamen **Trig2** ein. Markieren Sie die drei Fragezeichen vor dem Funktionsblock Trig2, betätigen Sie die Funktionstaste <F2> (Eingabehilfe) und wählen Sie im erscheinenden Dialog in der Kategorie **Globale Variablen** die Variable **Observer** aus.

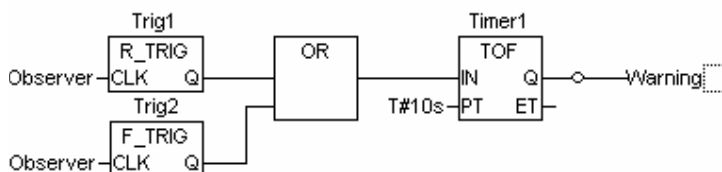
- **Zeitüberwachung, erster Teil**

Fügen Sie hinter dem OR-Operator über das Kontextmenü eine **Baustein** vom Typ **TOF** (Ausschaltverzögerung) mit dem Namen **Timer1** ein.

Ersetzen Sie die drei Fragezeichen am **PT** - Eingang mit der Zeitkonstanten **T#10s** für 10 Sekunden. (Die Zeit kann später nach erfolgreichem Test erhöht werden.)

- **Warnung ausgegeben**

Fügen Sie hinter dem Q-Ausgang des Funktionsblocks Timer1 über das Kontextmenü eine **Zuweisung** ein. Die drei Fragezeichen ersetzen Sie durch den Variablennamen **Warning**. Geben Sie im Dialog zur Variablendeklaration die **Klasse VAR\_GLOBAL** und als Typ **BOOL** an. Damit die Warnung auch richtig gemeldet wird, markieren Sie die erste Position auf der Zuweisung und führen über das Kontextmenü den Befehl **Negation** aus. Diese bewirkt eine Invertierung des booleschen Signals und wird mit einem kleinen Kreis dargestellt.



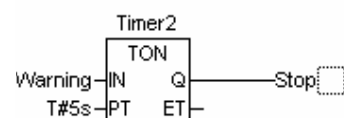
- **Stopsignal setzen nach zweiter Zeitüberschreitung**

Erzeugen Sie ein weiteres Netzwerk mit dem Menübefehl **Einfügen -> Netzwerk (danach)**.

Fügen Sie in dem zweiten Netzwerk über das Kontextmenü eine **Baustein** vom Typ **TON** (Einschaltverzögerung) mit dem Namen **Timer2** ein.

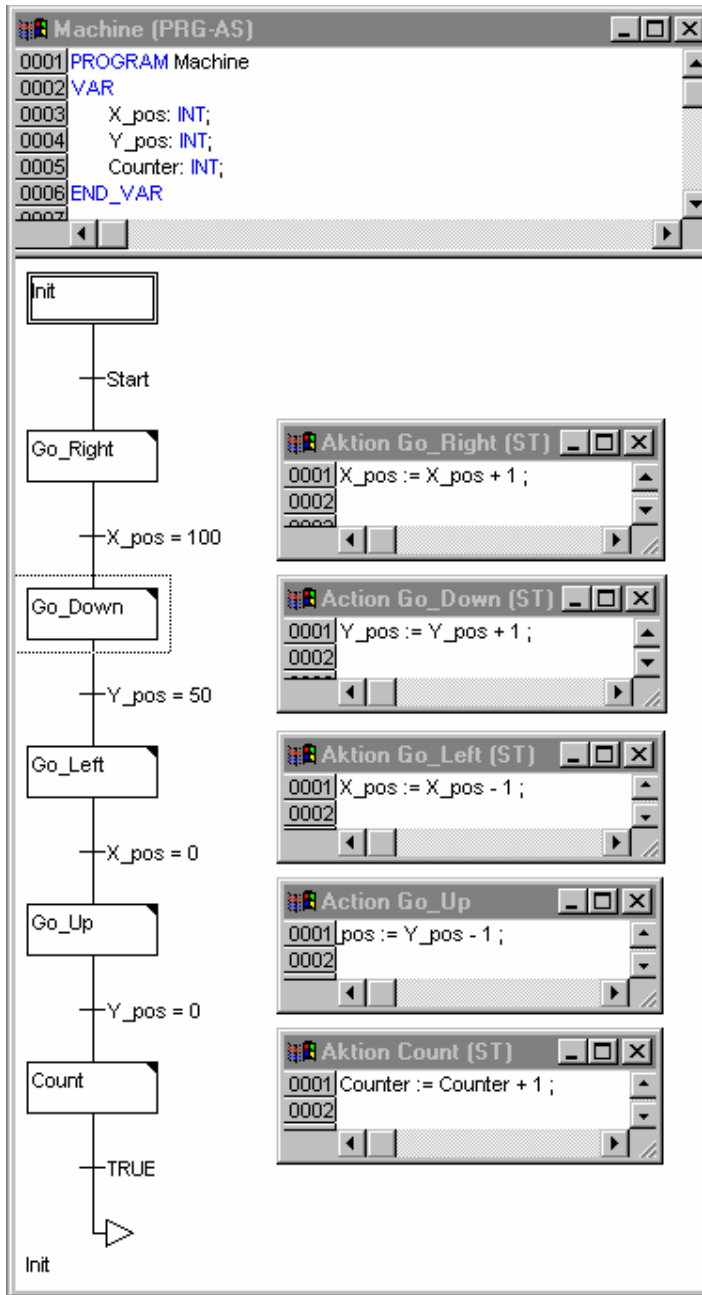
Belegen Sie den **IN**-Eingang mit der Variablen **Warning** mit Hilfe der <F2>-Taste und den **PT**-Eingang mit der Zeitkonstanten **T#5s**.

Nach dem Funktionsblock Timer2 benötigen wir wieder eine **Zuweisung**, diesmal auf die Variable **Stop** (Klasse **VAR\_GLOBAL**).

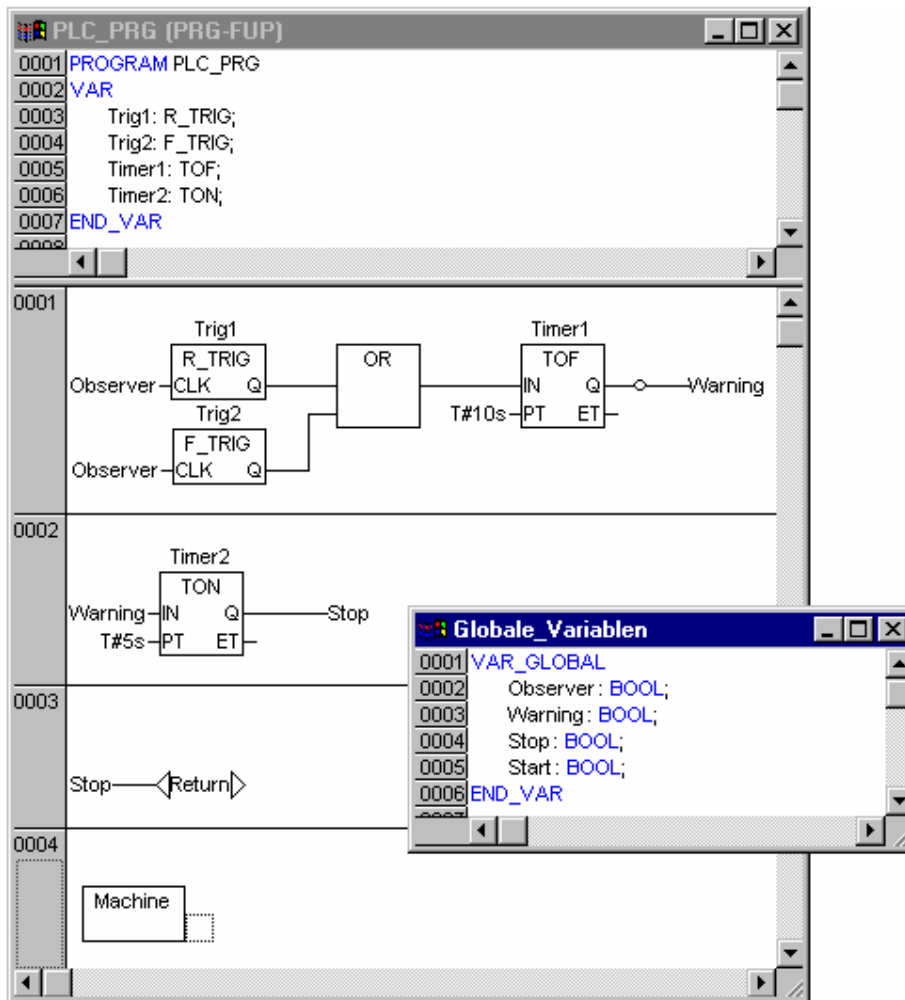




- **Baustein für Maschine einfügen**  
 Fügen Sie im Objekt Organizer (linker Bereich von CoDeSys) unter Bausteine mit dem Kontextmenü-Befehl **Objekt einfügen** einen weiteren Baustein ein mit **Namen Maschine** vom **Typ Programm** und in der **Sprache AS** (Ablaufsprache).  
 Das Anfangsdiagramm eines Bausteins in AS besteht aus einem Schritt "Init" und einer nachfolgenden Transition "Trans0" und einem Sprung zurück zu Init.
- **Ablauf der Maschine festlegen**  
 Wir benötigen für jede Phase unseres Maschinenablaufes einen Schritt. Markieren Sie die Transition **Trans0** so, daß ein gepunkteter Rahmen um Trans0 erscheint und wählen im Kontextmenü den Befehl **Schritt-Transition (danach)**. Wiederholen Sie diesen Vorgang fünfmal.  
 Wenn Sie direkt auf den Namen einer Transition oder eines Schrittes klicken, wird er blau markiert und kann geändert werden.  
 Der Schritt nach dem **Init**-Schritt soll **Go\_Right** heißen, die folgenden **Go\_Down**, **Go\_Left**, **Go\_Up** und **Count**.
- **Programmieren des ersten Schrittes**  
 Nach einem Doppelklick auf den Schritt **Go\_Right** werden Sie nach der gewünschten **Sprache** gefragt. Wählen Sie **ST** (Strukturierter Text) und es erscheint ein eigenes Fenster für die Aktion zu diesem Schritt.  
 Der Maschinenarm soll in X-Richtung verschoben werden. Die Programmzeile lautet:  
**X\_pos := X\_pos + 1 ;**  
 Schließen Sie die Zeile mit einem Return ab und deklarieren Sie die Variable **X\_pos** mit dem **Typ INT** für Integer.  
 In der linken oberen Ecke des Schrittes erscheint nun kleines Dreieck zum Zeichen, daß dieser Schritt programmiert ist.
- **Programmieren der weiteren Schritte**  
 Wiederholen Sie den Vorgang für die weiteren Schritte mit anderen Programmzeilen und deklarieren Sie dabei die Variablen **Y\_pos** und **Counter** vom **Typ INT**.  
 Im Schritt **Go\_Down**      **Y\_pos := Y\_pos + 1 ;**  
 Im Schritt **Go\_Left**      **X\_pos := X\_pos - 1 ;**  
 Im Schritt **Go\_Up**      **Y\_pos := Y\_pos - 1 ;**  
 Im Schritt **Count**      **Counter := Counter + 1 ;**
- **Transitionen festlegen**  
 Transitionen sind die Übergangsbedingungen von einer Phase in die nächste. Überschreiben Sie die erste Transition nach Init mit der Variablen **Start**. Geben Sie bei der Variablendeklaration die **Klasse VAR\_GLOBAL** und den **Typ Bool** an. Die Maschine beginnt somit erst zu laufen, wenn der Start-Schalter gedrückt ist.  
 Die zweite Transition erhält die Bedingung **X\_Pos =100**, d.h. wenn die X-Position den Wert 100 erreicht hat, beginnt die nächste Phase.  
 Die dritte erhält die Bedingung **Y\_pos =50**,  
 die vierte erhält die Bedingung **X\_pos = 0**,  
 die fünfte erhält die Bedingung **Y\_pos = 0** und  
 die sechste erhält die Bedingung **TRUE** ( Nach einmaligem Durchlauf sofort weiterschalten).



- Bei Stop Abarbeitung anhalten**  
 Gehen Sie zurück zum Baustein **PLC\_PRG** und fügen Sie ein drittes Netzwerk ein. Ersetzen Sie die Fragezeichen durch die Variable **Stop** und wählen Sie danach im Kontextmenü den Befehl **Return** aus. Der Return bewirkt ein Verlassen des Bausteines PLC\_PRG, wenn Stop gesetzt ist.
- Baustein Machine aufrufen**  
 Fügen Sie ein weiteres Netzwerk ein, setzen Sie den Fokus dorthin und führen Sie im Kontextmenü den Befehl **Funktionsblock** aus. Das "AND" im Funktionsblock erscheint bereits markiert. Drücken Sie nun <F2> und wählen Sie im erscheinenden Eingabehilfe-Dialog in der **Kategorie Definierte Programme** den Baustein **Maschine**. Damit ist die Programmierung abgeschlossen.



- **Projekt Übersetzen**

Übersetzen Sie das Projekt mit dem Menübefehl **Projekt -> Alles übersetzen** oder der Funktionstaste **<F11>**.

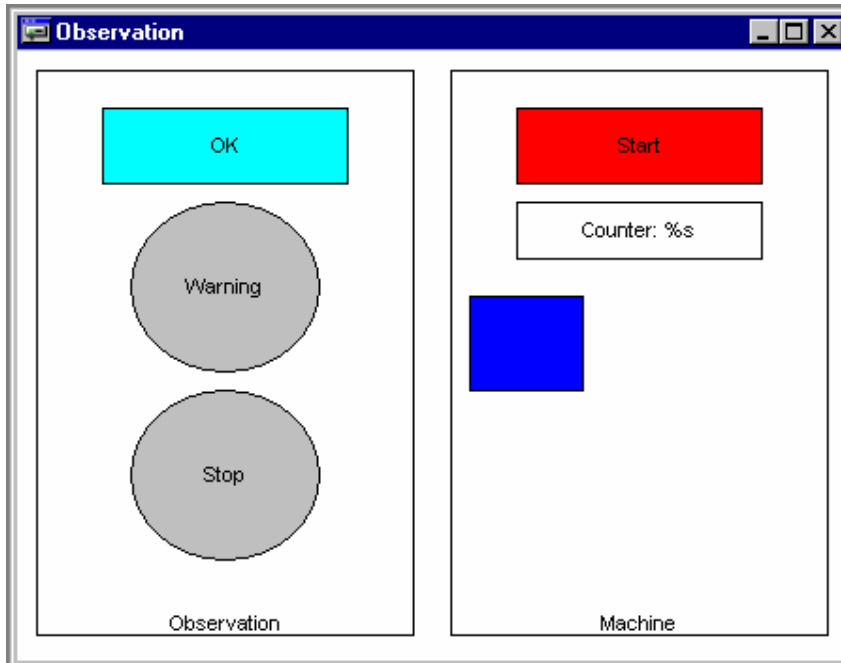
Im Meldungsfenster rechts unten sollte nach dem Übersetzungslauf „**0 Fehler**“ gemeldet werden. Wenn nicht, überprüfen Sie bitte die Richtigkeit aller Eingaben. Beachten Sie dazu auch die gemeldeten Fehler.

### 3 Eine Visualisierung dazu

- **Visualisierung anlegen**

Wechseln Sie im Object Organizer links unten in die dritte Registerkarte von links mit Namen **Visualisierung**.

- Rufen Sie über das Kontextmenü des Object Organizers den Befehl **Objekt einfügen** auf. Geben Sie dem Visualisierungsobjekt einen **Namen**, z.B. **Observation**. Am Ende soll die Visualisierung so aussehen:



- **Visualisierungselement zeichnen**

Beginnen wir mit dem Bestätigungsschalter (Im Bild das Rechteck mit dem Text OK). Wählen Sie aus der Symbolleiste das Symbol für **Rechtecke** aus. Drücken Sie die linke Maustaste im Visualisierungseditor, halten Sie sie gedrückt und ziehen Sie dabei eine Rechteck auf. Hat es die gewünschte Form und Größe erreicht, lassen Sie die Maustaste los.

- **Erstes Visualisierungselement konfigurieren**

Mit Doppelklick auf das Rechteck erscheint der Konfigurationsdialog.

Schreiben Sie in der **Kategorie Text** in das Feld **Inhalt** den Text **OK**.

Wählen Sie die **Kategorie Variablen**, klicken in das Feld **Farbwechsel** und rufen die Eingabehilfe mit der Funktionstaste <F2> auf. Mit Doppelklick auf den Punkt in der rechten Hälfte des Dialogs erscheinen die globalen Variablen. Wählen Sie die Variable **Observer** aus. Im Feld steht nun **.Observer**

Anschließend wählen Sie die **Kategorie Farben** im Konfigurationsdialog und klicken auf die Schaltfläche **Innen** im Bereich **Farbe**. Wählen Sie eine Farbe (z.B. hellblau). Danach wählen Sie eine weitere Farbe mit der Schaltfläche **Innen** im Bereich **Alarmfarbe** (z.B. blau). Wählen Sie in der **Kategorie Eingabe** den Punkt **Variable toggeln** an und tragen Sie in dem Feld dahinter die Variable **Observer** wieder mit <F2> ein. Verlassen Sie den Konfigurationsdialog. Damit zeigt das Rechteck im laufenden Betrieb die Farbe hellblau, wenn die Variable **Observer** den Zustand **FALSE** hat und blau im **TRUE** - Zustand. Mit jedem Mausklick wird der Zustand von **TRUE** auf **FALSE** gesetzt bzw. umgekehrt.

- **Visualisierung ergänzen**

Zeichnen Sie einen Kreis für die Warnung und konfigurieren ihn mit folgenden Werten:

**Kategorie Text** Feld **Inhalt** mit **Warning**

**Kategorie Variablen** Feld **Farbwechsel** mit **.Warning**

**Kategorie Farben** Bereich **Farbe** über **Innen** mit Farbe grau und **Alarmfarbe** mit rot.

Kopieren Sie den erstellten Kreis mit dem Menübefehl **Bearbeiten -> Kopieren** und fügen Sie einen weiteren ein mit dem Menübefehl **Bearbeiten -> Einfügen**. Ändern Sie im neuen Kreis folgende Werte:

**Kategorie Text** Feld **Inhalt** mit neuem Eintrag **Stop**

**Kategorie Variablen** Feld **Farbwechsel** mit neuem Eintrag **.Stop**

Zeichnen Sie ein Rechteck für den Maschinenstart und konfigurieren ihn mit folgenden Werten:



**Kategorie Text** Feld **Inhalt** mit **Start**

**Kategorie Variablen** Feld **Farbwechsel** mit **.Start**

**Kategorie Eingabe** Punkt **Variable toggeln** anwählen

**Kategorie Eingabe** hinter **Variable toggeln** im Feld **.Start** eintragen

**Kategorie Farben** Bereich **Farbe** über **Innen** mit Farbe rot und **Alarmfarbe** mit grün.

Zeichnen Sie ein Rechteck für den Zähler und konfigurieren ihn mit folgenden Werten:

**Kategorie Text** Feld **Inhalt** mit **Counter : %s**

(%s dient als Platzhalter für den Variablenwert)

**Kategorie Variablen** Feld **Textausgabe** mit **Machine.Counter**

Zeichnen Sie ein Rechteck für die Maschine und konfigurieren ihn mit folgenden Werten:

**Kategorie Bewegung absolut** Feld **X-Offset** mit **Machine.X\_pos**

**Kategorie Bewegung absolut** Feld **Y-Offset** mit **Machine.Y\_pos**

**Kategorie Farben** Bereich **Farbe** über **Innen** mit Farbe blau.

Wer möchte, kann noch zwei große Rechtecke um die beiden Bereiche Beobachtung und Maschine zeichnen, in der **Kategorie Text** beim **Inhalt Observation** bzw. **Machine** eintragen und die **vertikale** Ausrichtung **unten** wählen. Legen Sie die Rechtecke mit dem Kontextmenü-Befehl **nach hinten** hinter die bestehenden Elemente.

**Die folgenden Punkte 4,5,6 können nur durchgeführt werden, wenn ein Laufzeitsystem installiert ist, das den Zielsystemeinstellungen in CoDeSys entspricht. Andernfalls haben Sie die Möglichkeit, im Simulationsmodus zu arbeiten. Springen Sie in diesem Fall direkt in Punkt 6, um das Projekt zu laden und zu starten.**

#### **4 Zielsystem starten**

Starten Sie des Zielsystems. (Beachten Sie, daß dies beispielsweise im Falle von CoDeSys SP RTE nur unter Windows NT 4.0, Windows 2000 bzw. Windows XP möglich ist. Hier erscheint daraufhin das Icon für CoDeSys SP RTE in der Statusleiste. Klicken Sie dieses bitte mit der rechten Maustaste an und betätigen Sie den Menüpunkt 'Start System'. )

#### **5 Einstellungen zum Verbindungsaufbau**

Bevor wir die Verbindung mit dem Zielsystem aufbauen können, sind einige wenige Einstellungen vorzunehmen.

Führen Sie dazu den Menübefehl **Online -> Kommunikationsparameter** aus.

Wählen Sie in dem erscheinenden Dialog die Schaltfläche **Neu**, um den Verbindungsaufbau zu Ihrem Zielsystem zu konfigurieren. Vergeben Sie in dem neuen Dialog einen geeigneten Namen für die Verbindung und selektieren Sie die entsprechende Verbindung zum Zielsystem. Für CoDeSys SP RTE **Shared memory (Kernel)**. Verlassen Sie den Dialog mit OK und beenden den Kommunikationsparameter Dialog mit OK.

#### **6 Starten des Projektes**

- Mit dem Menübefehl **Online -> Einloggen** in der Entwicklungsumgebung **CoDeSys** wird der Verbindungsaufbau zum Zielrechner hergestellt und
- mit **Online -> Start** Ihr Programm auf dem Zielrechner gestartet. (Falls Sie im Simulationsmodus arbeiten wollen, muß die Option 'Online' 'Simulation' aktiviert sein).
- Über die Visualisierung können Sie nun die Maschine starten und den Bestätigungsschalter bedienen.



## **7 Weiteres Vorgehen**

Nun liegt es an Ihnen, weitere Applikationen zu erstellen und zu verwenden. Weitere Informationen finden Sie in der Online Hilfe und im Benutzerhandbuch zu CoDeSys.  
Wir wünschen Ihnen dazu viel Erfolg!