

Automatisierungstechnik nach internationaler Norm programmieren (12)

Autor: Dr. Ulrich Becker
Fachzentrum Automatisierungstechnik und vernetzte Systeme im BTZ Rohr-Kloster
Mail: Ulrich.Becker@BTZ-Rohr.de

Integrierte Visualisierung

In Folge 11 haben wir die IEC-gerechte Verarbeitung analoger Signale mit dem Programmiersystem CoDeSYS behandelt. Analogwerte und Listeneinträge wurde wie gewohnt simuliert und im Online-Modus beobachtet. Bereits hier wäre eine einfache Visualisierung der Werte für Inbetriebnahme und Verständnis der Zusammenhänge vorteilhaft gewesen. Deshalb wird in dieser Folge die im Programmiersystemen integrierte Visualisierung vorgestellt.

Visualisierung in der Automatisierungstechnik

Panels und Software für die Visualisierung technischer Anlagen und Vorgänge haben längst im großen Stil klassische Bedienpulte mit Tastern und Meldeleuchten verdrängt. Mit Touch Panels werden heute Befehle, Daten und Rezepte in Steuerungen eingegeben sowie Daten, Zustände oder Meldungen angezeigt und quittiert. Die „Mensch - Maschine – Interface“ sind zu einem eigenständigen Zweig der Automatisierungstechnik mit unterschiedlichen Ausprägungen geworden. Zumeist sind die Geräte gewöhnliche Teilnehmer der dezentralen vernetzten Systeme. Mitunter übernehmen „Panel PC“ aber bereits die gesamte Steuerung von Anlagen.

Methoden der Visualisierung

CoDeSys benennt je nach Einsatzfall vier Typen von Visualisierungen (siehe www.3s-software.com):

Integrierte Visualisierung:

Für diese Art Visualisierung von Zuständen und Daten der Steuerung benötigt man zunächst kein zusätzliches Tool. Im Programmiersystem CoDeSys integriert wird bereits ein Visualisierungs-Editor angeboten. Damit kann man beim Schreiben des Programms auf der gleichen PC-Oberfläche und unter Nutzung der gleichen Variablen sofort Visualisierungsmasken entwickeln und diese zumindest im Simulationsmodus auch ohne angeschlossene Hardware sofort nutzen.

HMI

Visualisierung:

Mit dem Programm CoDeSys HMI können die gleichen Visualisierungsmasken auf dem PC angezeigt werden, ohne dass das eigentliche Programm dabei zugänglich ist. Das ist von Bedeutung, wenn auf einem PC visualisiert wird, aber Bediener von Anlagen keine Rechte für Eingriffe in das Programm erhalten sollen. Vorteilhaft ist, daß das Tool über die gleiche Programmierschnittstelle kommuniziert und weitere Software für OPC oder DDE nicht erforderlich ist.

Target-Visualisierung: Anders liegen die Probleme bei der Visualisierung mittels Display oder Panel, genannt Target-Visualisierung. Die Visualisierungsinformationen müssen hier mit einem Codegenerator aus dem Programmiersystem entnommen und für das entsprechende Zielsystem (Target) übersetzt werden. Bei der Auswahl des Panels ist darauf zu achten, dass dieses mit einem Code nach IEC 61131-3 zu betreiben ist, was bestimmte Treiber-Software erfordert. Diese muss der Hersteller der Geräte anbieten. Von Interesse ist auch die Anbindung des Panels an die Steuerung. Hier sollte sich eine Schnittstelle für die Protokolle Ethernet TCP/IP/UDP und Modbus durchsetzen.

Web-Visualisierung: Schließlich ist heute die Web - Visualisierung in aller Munde. Damit die Visualisierungsdaten auf Web-Browsern online zur Verfügung stehen, muss aus den Daten eine XML-Beschreibung generiert werden. Diese wird zusammen mit einer Java-Anwendung (Applet) auf der Steuerung abgelegt und über das Protokoll TCP/IP auf Browsern unterschiedlichster Plattformen dargestellt. CoDeSys generiert auch den für Web-Visualisierung erforderlichen Code.

Die integrierte Visualisierung im System CoDeSys

Die in CoDeSys integrierte Visualisierung bietet manche Vorteile: Für Service- und Diagnosezwecke direkt in Online-Verbindung mit der Steuerung steht die Visualisierung sofort und ohne Einsatz eines weiteren Tools zur Verfügung. Für Weiterbildung und Training ist dies von unschätzbarem Vorteil. Die so erstellten Masken können ohne Änderungen in den anderen genannten Ablauf-Varianten eingesetzt und vor ihrem Einsatz zunächst getestet werden.

Grundsätzlich vorteilhaft ist, daß für die Erstellung der Masken keine speziellen Listen für Wandlung und Zwischenspeicherung des Code benötigt wird. Die Visualisierung nutzt direkt die Variablen aus der Steuerung. Damit entfällt die Konfiguration einer OPC- oder DDE-Schicht, und der Aufwand für die Realisierung der Visualisierung wird erheblich reduziert. Selbstverständlich kann man von einer kostenfreien integrierten Visualisierung nicht die Leistungen großer Visualisierungstools der Simatic-Welt wie ProTool oder WinCCflexible erwarten. Trotzdem ist man überrascht, welche Möglichkeiten die integrierte Visualisierung bietet.

Zur Gestaltung der Masken stehen u.a. die Elemente Linie und Linienzug, Rechteck, Kreis und Ellipse, Kurvenzug, abgerundetes Rechteck zur Verfügung. Diese können farblich gestaltet werden. Textanzeigen, die Verlinkung auf andere Masken bis hin zu Vektorgrafiken ergänzen die Möglichkeiten. Komplexe Elemente wie Zeigerinstrument, Balkenanzeige, Histogramm, Trendanzeige, TabellenControl und Alarmbehandlung können ohne Aufwände in Masken eingebaut werden. Weiter sind Animationen möglich. Dazu gehören Farbumschlag, Sichtbarwerden von Texten und Elementen, Verschiebung und Rotation. Zur Eingabe von Steuerbefehlen kann man Boolesche Variablen tasten oder togglen, digitale Variablen schreiben oder auch einen Text über PC Tastatur eingeben. Mit neueren Versionen des Tools wurde die integrierte Visualisierung fortschreitend komfortabler, und dieser Trend wird sich auch fortsetzen.

Ein erste Visualisierung

Das parallele Wirken von Hardware und Visualisierung soll zuerst an einer sehr einfachen Applikation verständlich werden. Am Trainingsrack wurde die Anlage mit Ein- und Aus-Tastern betriebsbereit geschaltet. Eine LED signalisierte den Zustand „Anlage ein“. Nun soll eine Visualisierung erstellt werden, auf der parallel dazu mit einem (virtuellen) Toggle-Taster die Anlage gleichfalls ein- oder ausgeschaltet werden kann. Die Visualisierung soll weiter eine LED zur Anzeige des Schaltzustandes erhalten. Zusammen mit einem Farbumschlag bei „Anlage ein“ soll weiterhin ein Textfeld „Anlage eingeschaltet“ erscheinen.

Zur Lösung dieser Aufgabe muß zunächst eine neue Visualisierung im Projekt angelegt werden (**Bild 64**). Visualisierungen gehören neben Bausteinen, Datentypen und Ressourcen zu den grundlegenden Elementen eines Projektes. Rechter Mausklick auf Visualisierungen und -> *Objekt einfügen* sowie Eintrag eines Namens öffnet das Fenster der Visualisierung. Damit wird klar, dass ein Projekt mehrere Visualisierungen enthalten kann.

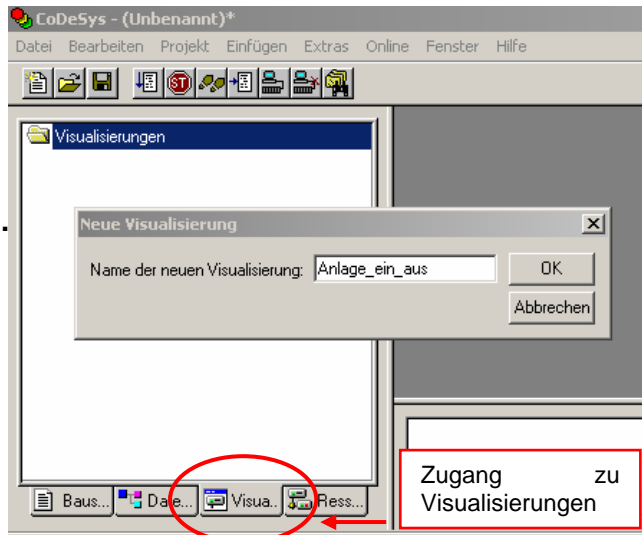


Bild 64: Anlegen einer Visualisierung im System CoDeSys

Bild 65 zeigt die sich daraus ergebende Visualisierungsoberfläche, in welcher bereits ein Kreis als LED und ein weiterer als Toogletaster angelegt wurde. Markieren und Doppelklick auf die Elemente öffnet die Fenster zum Parametrieren derselben. Als Beispiel ist im Bild oben das Fenster zur Gestaltung eines Textes zu sehen. Mit ähnlichen Fenstern kann die Farbgebung mit dem Umschlagen der Farbe grün auf Alarmfarbe rot und das Erscheinen der Textausgabe „Anlage eingeschaltet“ parametrieren werden.

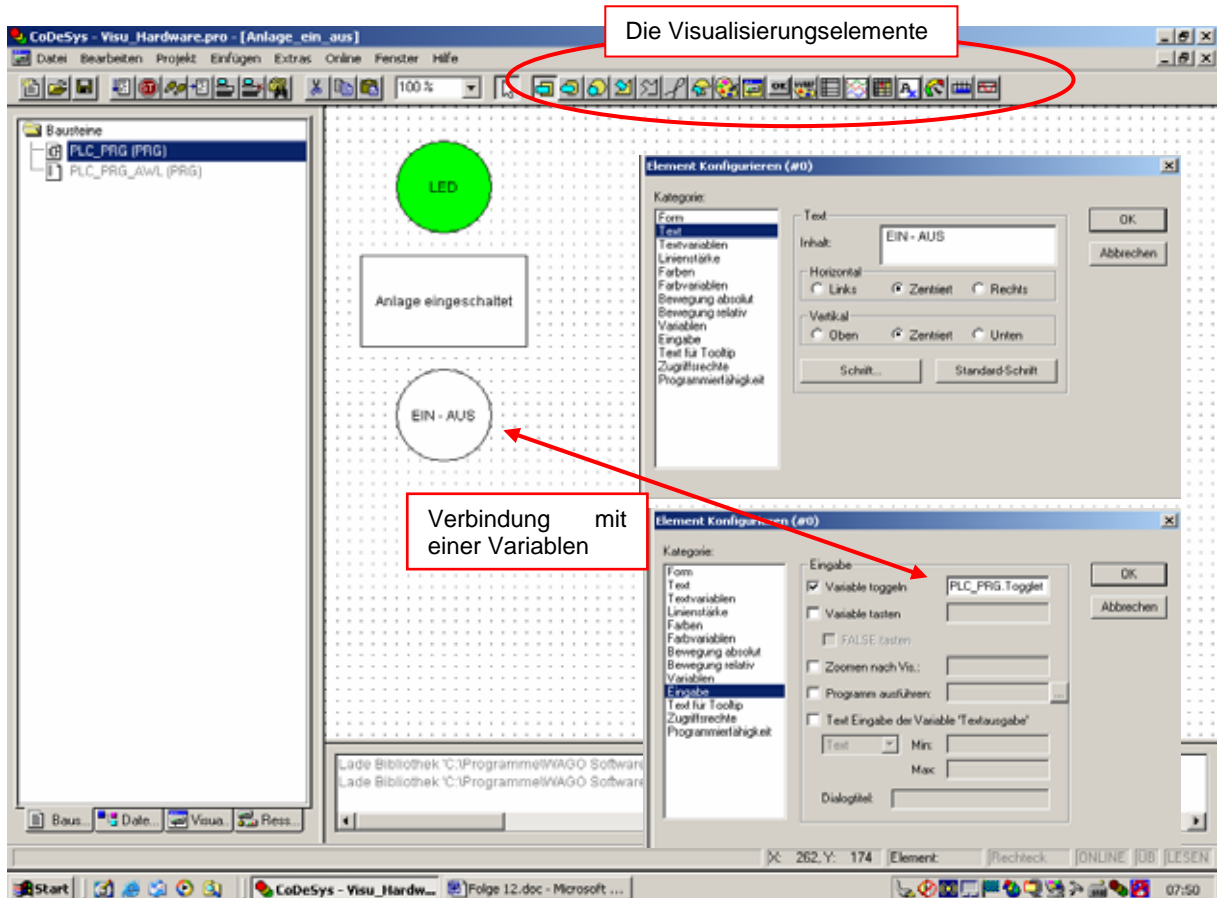


Bild 65: Arbeitsoberfläche mit Visualisierungselementen. Rechts zwei ausgewählte Fenster zum Parametrieren des Toogletasters

Entscheidend ist nun die korrekte Verbindung der Visualisierungselemente mit Variablen des zugehörigen Programms. **Bild 66** zeigt das Zusammenwirken von Programm und Visualisierung im Online-Modus. Der Einfachheit halber wurde dieses Programm in der POE PLC_PRG geschrieben und es wurden dort auch alle Variablen vereinbart.

Im Bild 65 rechts unten wurde festgelegt, daß das Tasterelement in der Visualisierung die Variable „*PLC_PRG.Toggleaster*“ toggelt, d.h. bei jedem Mausklick auf das Element zwischen FALSE und TRUE umschaltet. Die Einfügetaste F2 hilft auch hier bei der Auswahl der richtigen Variablen. Es muß stets der vollständige Pfad der Bausteine geschrieben werden. Beispielsweise ist „*.Var_1*“ eine globale Variable und „*PLC_PRG.Baustein1.Var_2*“ eine im Baustein1 vereinbarte Variable, wobei der Baustein1 im Hauptprogramm PLC_PRG aufgerufen wird.

Bisher erfolgte das Einschalten mit Tastern am Trainingsrack. Ihnen waren die direkt adressierten Variablen „*T_Ein*“ und „*T_Aus*“ zugeordnet. Über eine digitale Eingangsbusklemme mit den Adressen %IX2.0 und %IX2.1 bewirken sie das Schalten des RS-Flip-Flops mit Namen „*EIN_AUS-Schaltung*“ (Bild 66). Zu beachten ist, daß nach Vorschrift des VDE das Ausschaltung durch Unterbrechung eines Ruhestromes erfolgen muss und drahtbruchsicher zu programmieren ist. Deshalb ist im Ruhezustand an RESET1 – Eingang ein TRUE-Signal zu legen.

Soll nun der Taster der Visualisierung ebenfalls das Setzen und Rücksetzen des Flip-Flop's bewirken, so muß die zugeordnete Variable „*Toggleaster*“ für das parallele Wirken über ODER-Logik in das Programm eingebunden werden.

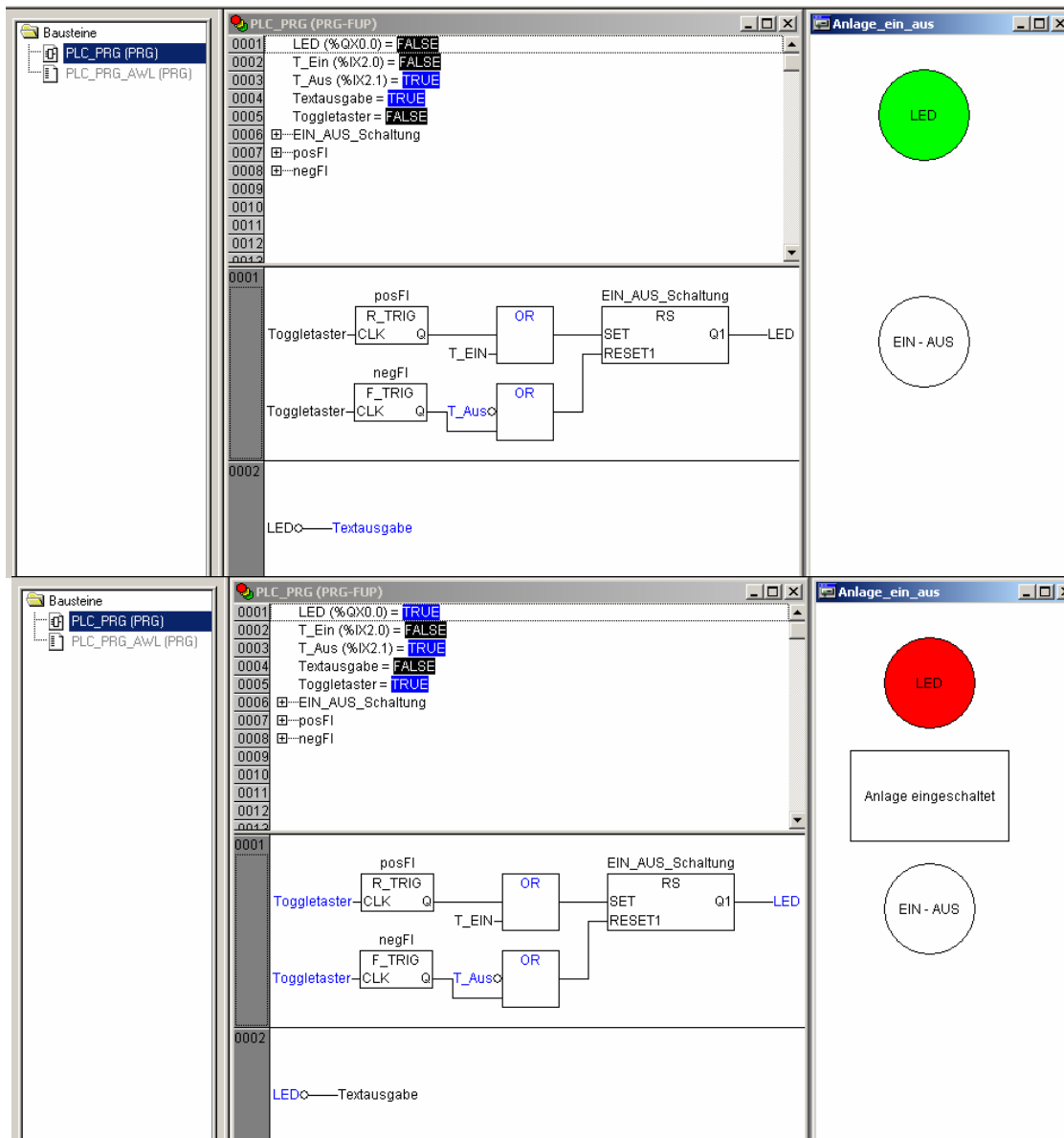


Bild 66: Das Programm mit Hardware-Tastern und virtuellem Toggletaster in grafischer Sprache, oben Zustand „Aus“, unten Zustand „Ein“ (Simulation und Online-Modus)

Der erste Mausklick schaltet die Variable „Toggeltaster“ auf TRUE und die entstehende positive Flanke setzt den RS-Speicher. Der folgende Mausklick schaltet die Variable auf FALSE und die negative Signalfanke setzt den Speicher zurück. Das Beispiel zeigt, daß über das Zusammenwirken von hardwarebestimmten Signalen und Signalen der Visualisierung stets sorgfältig nachgedacht werden muß. Dagegen ist es einfach, allein mit einer Visualisierung ein Programm zu testen, ohne daß Hardware bereits angeschlossen wurde.

Komplexe Elemente leisten mehr

Als Beispiel für komplexe Visualisierungselemente werden nachfolgend eine Balkenanzeige und ein Zeigerinstrument gestaltet. Folgende Schritte sind erforderlich

1. Anlegen einer Visualisierung mit Namen „*Balkenanzeige_Waage*“ im Projekt
2. Einfügen der Balkenanzeige bzw. des Zeigerinstrumentes
3. Gestaltung der Anzeigen: Ausrichtung, Skalierung, Farbgestaltung u.a.
4. Optional Festlegung der Parameter eines Alarms als Farbumschlag bzw. für Bereiche
5. Einbindung der Variablen, welchen den Ausschlag von Balken oder Zeiger steuert

Im **Bild 67** sind einige dieser Schritte erkennbar. In diesem Beispiel werden entsprechend der Aufgabenstellung 4 die Masse sowie der Mittelwert von zehn Messungen visualisiert. Diese Werte vom Typ INT wurden in Folge 11 in der POE „*Waage.PRG*“ aus einem analogen Meßwert ermittelt.

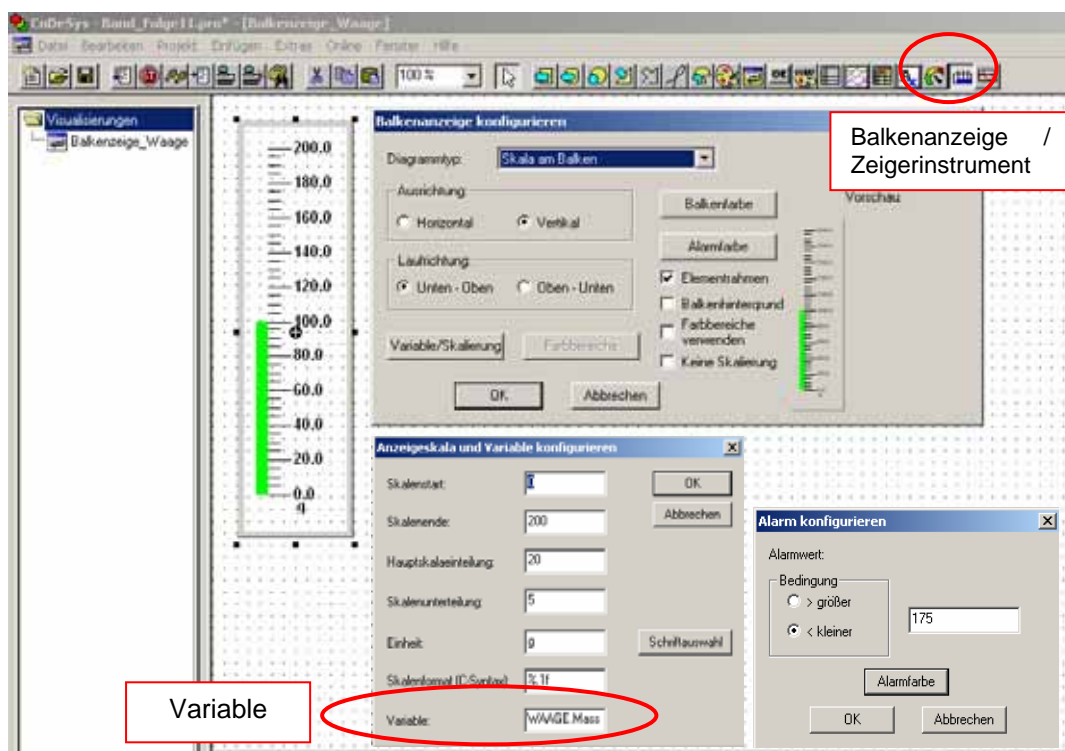


Bild 67: Auswahl von Werkzeugen zur Parametrierung von Balkenanzeigen

Im Bild 67 unten ist nun zu sehen, wie die Variable „*Waage.Masse*“ als Steuerelement des Balkens und ein Alarmwert von 175 im Wertebereich 0 .. 200g festgelegt werden. Unterschreitet die Masse diesen Wert, so erfolgt ein Farbumschlag von Balkenfarbe grün auf Alarmfarbe rot. Ähnlich ist bei der Gestaltung eines Zeigerinstrumentes vorzugehen. Hier ist die Variable „*Waage.Mittelwert*“ einzubinden. Die Farbbereiche wurden für gleiche Werte eingerichtet.

Die Funktion der Visualisierung zeigt **Bild 68** im Online-Modus für eine aktuelle Masse von 186 g. Der Mittelwert der letzten zehn Wägungen jedoch beträgt nur 174 g und liegt deshalb im roten Bereich.

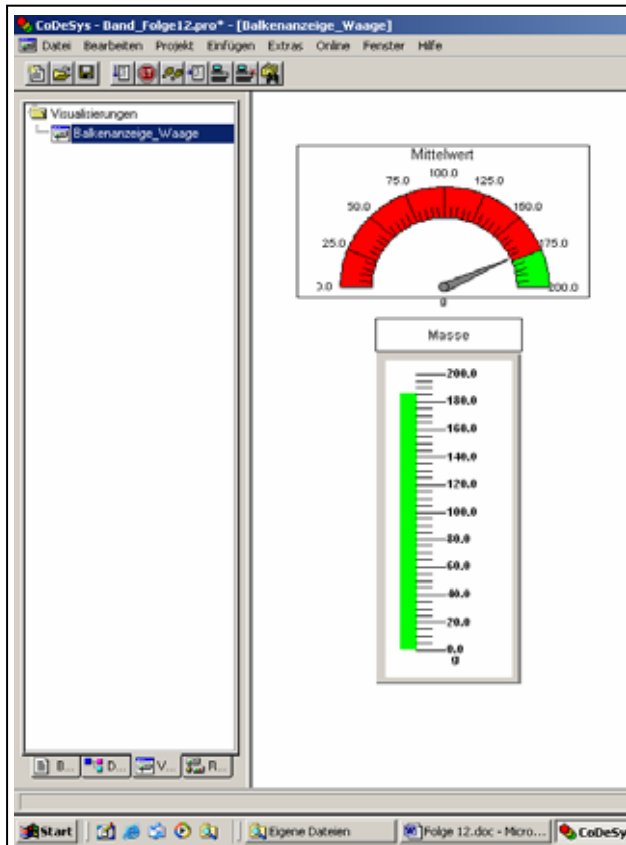


Bild 68: Online-Funktion der Visualisierung der POE „Waage“

Die integrierte Visualisierung ist besonders wertvoll für den Test von Programmen und hilfreich bei der Inbetriebnahme von Anlagen. So kann man sich zum Beispiel virtuelle Taster und Anzeigen schaffen und damit Funktionen tarten und übersichtlich kontrollieren. Ein Beispiel wäre die Inbetriebnahme einer Maschine, für die das Steuerpult noch nicht geliefert werden konnte. Ein solches Beispiel zeigt **Bild 69**. Diese Visualisierung erlaubt als „virtueller Bedienstand“ die Steuerung des Bandes am Trainingsrack samt Anmeldung durch Codeeingabe und Anzeige von Produktionszahlen.

Auf ein wichtiges Visualisierungselement ist hier noch zu verweisen: Gibt man bei der Konfiguration eines Elementes in der Kategorie *->Text* zusätzlich zum Text oder ausschließlich die Zeichen „%s“ ein und hinterlegt unter *-> Variablen -> Textausgabe* eine Variable, so wird deren Wert im Online-Modus dargestellt. Das "%s" wird dann durch den Wert ersetzt. Im Bild 69 werden Online auf diesem Wege Zählerwerte bzw. Codezahlen angezeigt.

Abschließend ist zu bemerken, daß in dieser Folge nur ein kleiner Teil der Elemente und Möglichkeiten der integrierten Visualisierung gezeigt werden konnte. Hier findet der interessierte Leser ein breites Betätigungsfeld und in der CoDeSys Hilfe zur Visualisierung Antwort auf Detailfragen. Viel Wissen kann spielerisch erworben werden.

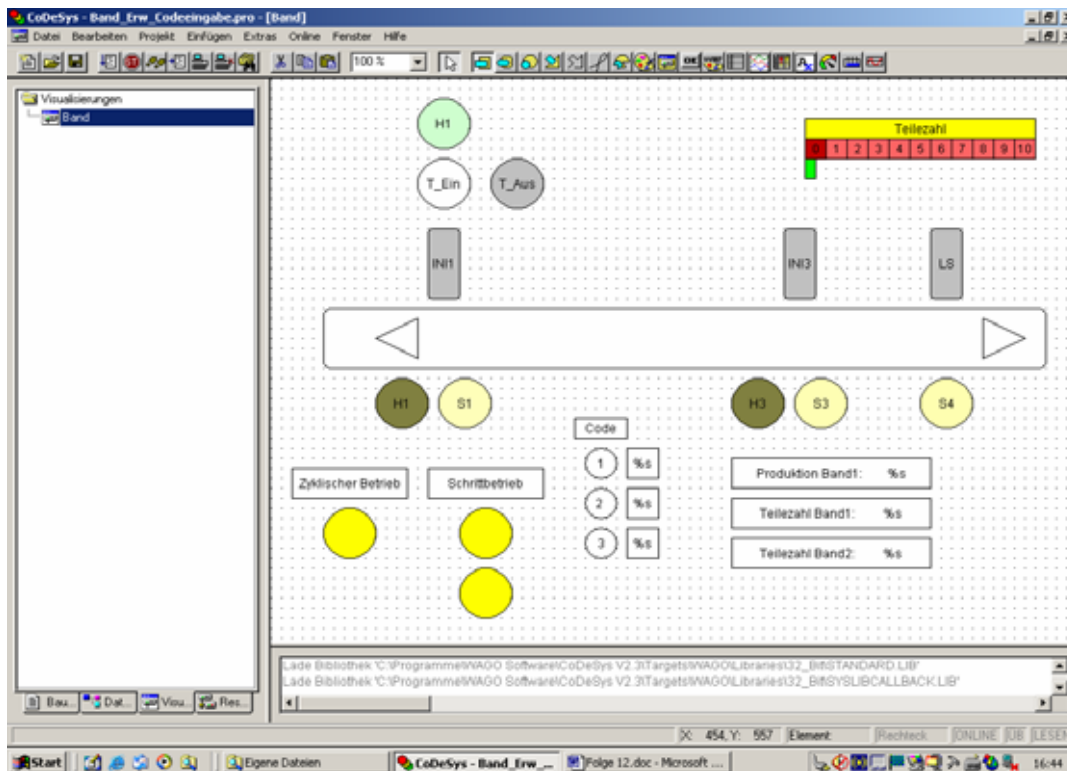


Bild 69: Beispiel eines „virtuellen Bedienstandes“ für das Bandmodell am Trainingsrack

Fazit:

In dieser Folge wurden Grundlagen der im System CoDeSys integrierten Visualisierung vorgestellt. Ohne zusätzliches Tool können damit Befehle in Steuerungen eingegeben und Zuständen oder Daten der Steuerung beobachtet werden. Bereits beim Schreiben eines Programms werden auf der gleichen PC-Oberfläche Visualisierungsmasken entwickelt, die Bestandteil des Projektes sind. Im Simulationsmodus können sie sofort und auch ohne angeschlossene Hardware genutzt werden. Dies ist vorteilhaft für Inbetriebnahme und Service. In späteren Folgen müssen allerdings andere Methoden der Visualisierung gegenübergestellt werden. In der nächsten Folge sollen jedoch zunächst die Kenntnisse über IEC Sprachen durch Vorstellung der Ablaufsprache erweitert werden.

Glossar:

DDE	engl. Dynamic Data Exchange (Dynamischer Datenaustausch). Software Schnittstelle der Fa. Microsoft für den Datenaustausch zwischen Anwendungen, z.B. zwischen Daten des Systems CoDeSys und Microsoft Office. DDE stellt eine Alternative zu OPC dar, verliert aber gegenüber diesem Standard an Bedeutung, denn DDE ist der Vorgänger zu OLE.
HMI	engl. Human Machine Interface (Mensch-Maschine-Schnittstelle) Allgemeiner, abstrakter Begriff für Geräte zum Bedienen und Beobachten von Maschinen und Anlagen. Im deutschen Sprachraum wird auch der Begriff B&B für Bedienen und Beobachten verwendet.
OPC OLE	engl. OLE for Process Control, wobei OLE für Object Linking and Embedding steht. Dies kann sinngemäß mit Verbinden und Einbetten von Objekten in der Automatisierungstechnik übersetzt werden. OPC ist ein weltweiter Standard für Datenschnittstellen in der Automatisierungstechnik mit großer Bedeutung für HMI. Dabei werden OPC Clients an OPC Server angeschlossen. Die OPC-Foundation setzt sich seit 1997 für die weltweite Normung ein (www.opcfoundation.org).
Panel PC	Kombination von Industrie PC und Touch Panel für Bedienen und Beobachten in einem Gerät
Touch Panel	engl. to touch: berühren. Allgemeine Bezeichnung für Bedien- und Beobachtungsgeräte, bei denen virtuelle Button (Eingabfelder auf der Oberfläche) durch Berühren betätigt werden.
ProTool WinCC WinCCflexible	Softwaretools für Visualisierungen im System Simatic S7 / Step7. Diese Tools integrieren sich in Step7 und ermöglichen das Anlegen von Visualisierungsobjekten im Step7-Projekt, die dann in die HMI-Geräte geladen werden. Derzeit löst WinCCflexible die älteren Tools ProTool und WinCC ab. WinCCflexible ist skalierbar und zukünftig einheitlich für alle HMI-Geräte der Siemens AG einsetzbar. Der Trend geht dabei von MPI- und Profibus-DP-Anschlüssen hin zur Profinet-Schnittstelle.
Visualisierung	Sammelbegriff für Beobachten von Automatisierungsanlagen und Steuereingriffe in diese. Die Visualisierung erfolgt über Visualisierungsoberflächen auf Basis moderner PC-Technologie.