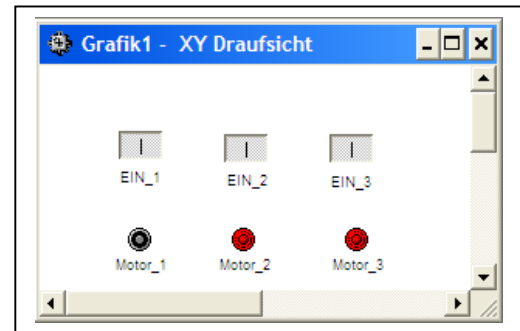
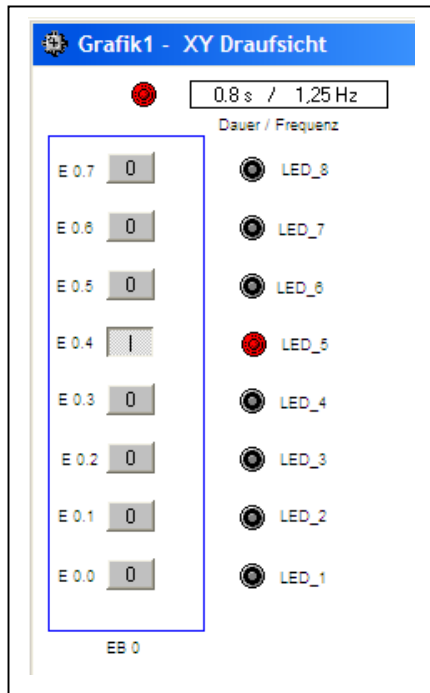


**Aufgabe:**

Von 3 möglichen Motoren dürfen wegen der Stromnetzbelastung nur maximal 2 zur gleichen Zeit laufen. Jeder Motor hat seinen eigenen EIN/AUS-Schalter. Jede Schaltkombinationen ist erlaubt, also alle Schalter können eingeschaltet werden. Der dritte Schaltbefehl darf aber nicht zum Starten des zugehörigen Motors führen.



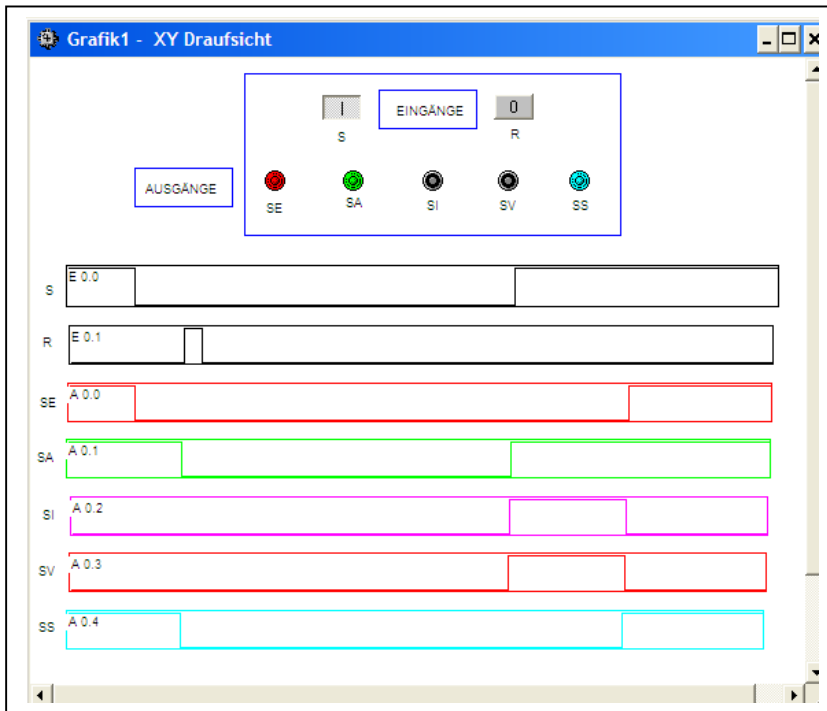


Step 7 und TrySim bieten die Möglichkeit, Taktsignale verschiedener Frequenz zu erzeugen.

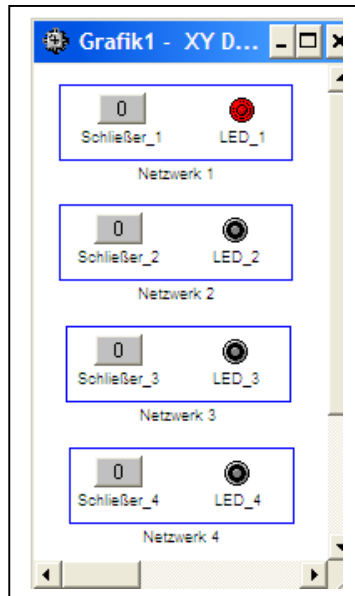
Dazu muss nur ein wahlfreies Merkeryte deklariert werden.

TrySim bietet die Möglichkeit, mittels Stufenschalter immer nur einen Schalter auf „1“ zu legen.

Durch die Abfrage verschiedener Eingangsbits sollen die unterschiedlichen Taktraten verdeutlicht werden.



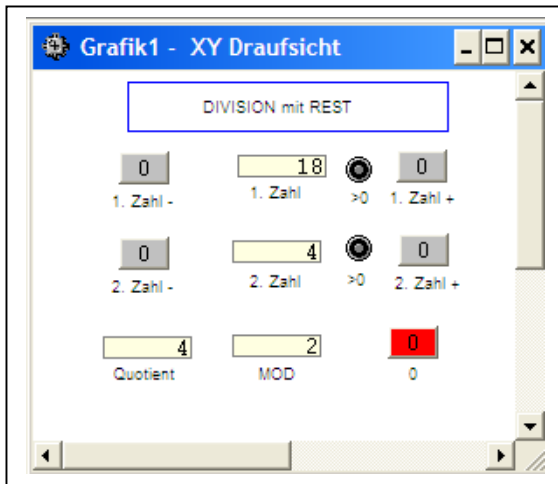
Alle TIMER-Typen können im Zeit-Ablauf-Diagramm verglichen werden.  
Hier geht es nicht um eine aufregende Programmierung, sondern mehr um die Darstellung der Möglichkeiten, die TrySim bietet.



### Binäruntersetzer

Bei jedem Tasterdruck soll im Wechsel die LED leuchten, bzw. erlöschen.

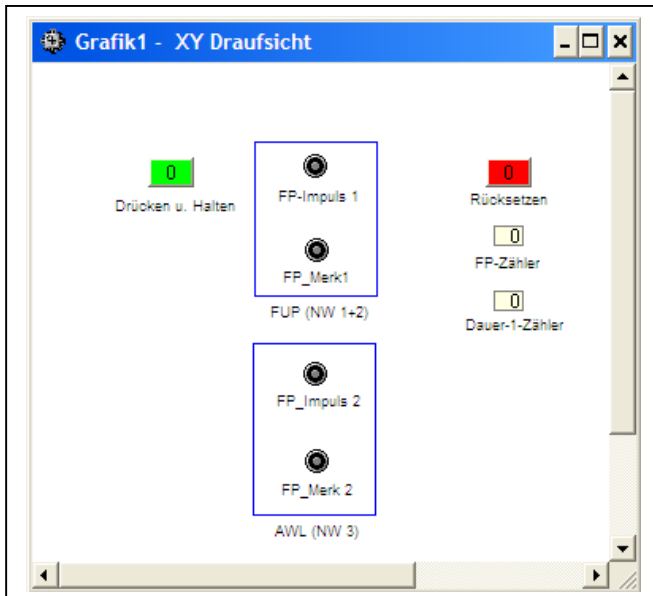
( 4 Varianten)



Die Operation „MOD“ soll durch dieses Projekt veranschaulicht werden.

Eine normale, ganzzahlige Division wird durch „/“ erreicht. Durch „MOD“ wird der ganzzahlige Rest dieser Division berechnet.

- Zur Abwechslung werden Dividend und Divisor ohne Zählerbaustein eingestellt.
- Die Ziffern können durch Einzel- oder Dauerdruck verändert werden.
- Beim Zahlenwert 0 blinken die TrySim-Blink-LEDs.
- Eine Division durch 0 wird verhindert.
- Alle Zahlen werden digital angezeigt.
- Mit Taster <0> werden alle Zahlen auf 0 gesetzt.



### Nachbildung der FP-Operation

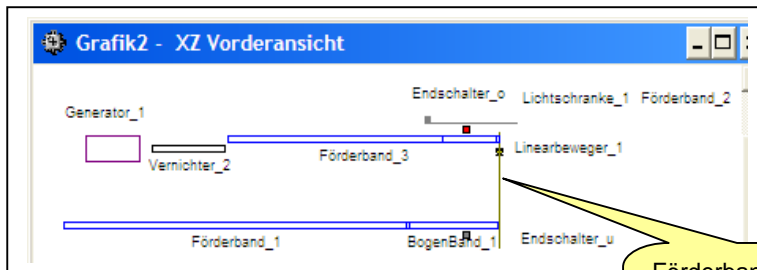
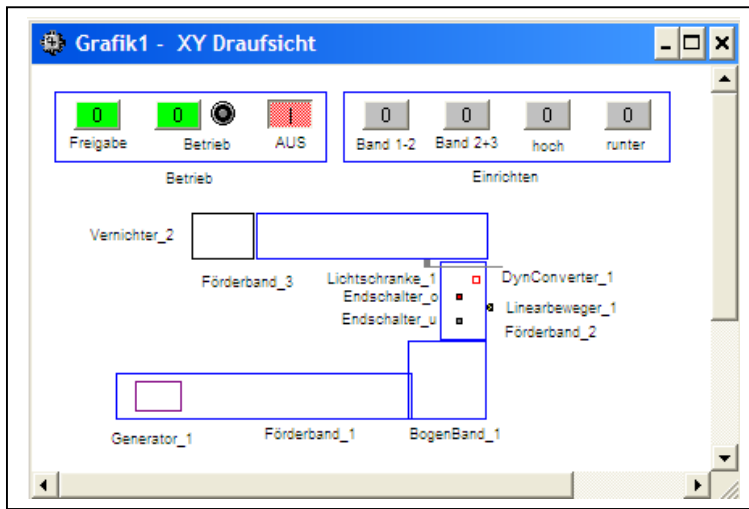
Das Innenleben der "FP"-Operation soll in FUP und AWL dargestellt werden.

Um die Bedeutung dieser Operation zu erkennen, soll - ohne Zählerbaustein ! - gezählt werden.

Bei einer Dauer-1 würde die Operation <zaehle> in jedem Zyklus erneut ausgeführt. Dieses ungewollte Durchrattern kann nur durch eine Flankenabfrage verhindert werden.

Die Zählerstandsanzeigen werden in diesem Projekt ohne Zählerbaustein erstellt.

(Nur im Zählerbaustein ist zum Weiterzählen ein Flankenwechsel, also ein erneutes Aus- und Einschalten, erforderlich. In dem Baustein ist schon ein „FP“ integriert ).



Förderband 2  
Wird über den  
Liniarbeweger  
nach oben gefahren.

### Bogenband mit Hubtisch

Nach der Freigabe mit einem Schalter wird über Druck auf Taster <Betrieb> ein Rohling aus dem Lager (Generator) geholt. Gleichzeitig schalten Förderband 1, das Bogenband und das Förderband 2 ein, sofern F.-Band 2 unten steht (F.-Band 2 kann senkrecht verschoben werden). Beim Erreichen der Lichtschranke stoppen die Bänder und F.-Band 2 wird nach oben gefahren. Oben angekommen, schalten F.-Band 2 und F.-Band 3 ein und transportieren den Rohling bis zum Vernichter 2.

Nach Ablauf von 30 s schalten die Antriebe ab und F.-Band 2 fährt wieder nach unten. Sofern nicht über den Taster <AUS> abgeschaltet wurde, beginnt der Zyklus erneut, usw.

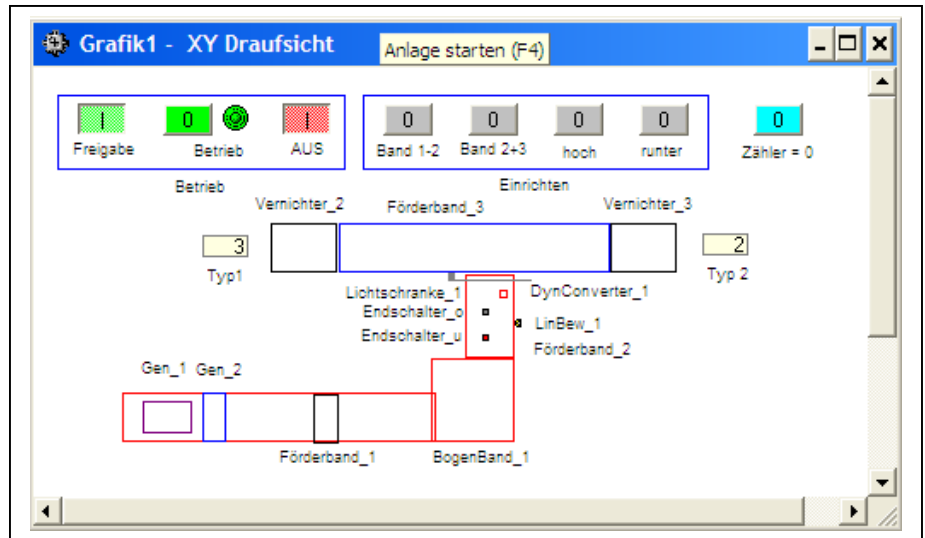
Wenn ausgeschaltet wurde aber die Freigabe noch eingeschaltet ist, können im Einrichtbetrieb die Bänder "von Hand" mit den entsprechenden Tastern betrieben werden. Dabei ist auf eine entsprechende Verriegelung zu achten, die beim Abräumen Fehlsteuerungen vermeidet.

Vor einem Neustart muss "nach Sicht" kontrolliert werden, dass alle Bänder frei sind. Diese Kontrolle muss das Programm nicht übernehmen.

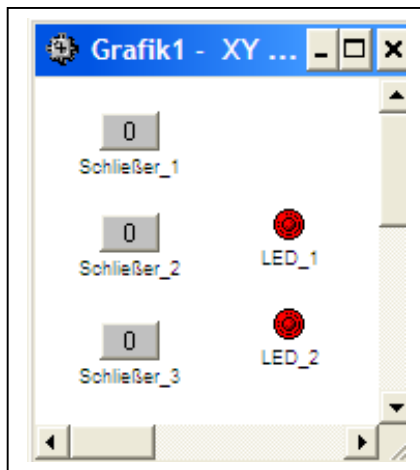
**Dieses Projekt ist eine Vertiefung des Projektes <BogenHub>**

### Verteilung je nach Art des Rohlings

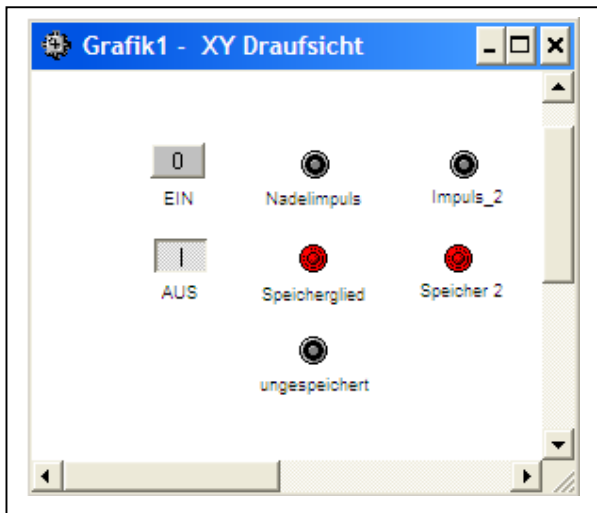
Im separaten OB 2 wird über einen Zufallsgenerator entschieden, welcher Rohling-Generator aktiv werden soll. Nach der Freigabe mit einem Schalter wird über Druck auf Taster <Betrieb> **ein** Rohling aus dem Lager (Generator) geholt. Gleichzeitig schalten Förderband 1, das Bogenband und das Förderband 2 ein, sofern F.-Band 2 unten steht (F.-Band 2 kann senkrecht verschoben werden).



Beim Erreichen der Lichtschranke stoppen die Bänder und F.-Band 2 wird nach oben gefahren. Oben angekommen, schalten F.-Band 2 und F.-Band 3 ein und transportieren den Rohling bis zum zugehörigen Vernichter. **Dabei wird Band 3 teileabhängig automatisch im Links- bzw. Rechtslauf eingeschaltet.**  
**Bedingung:** Es wird nirgends auf der Strecke der Rohlingstyp überprüft. Allerdings wird ein neuer Rohling erst frei gegeben, wenn der alte entsorgt ist.  
 Nach Ablauf von 30 s schalten die Antriebe ab und F.-Band 2 fährt wieder nach unten. Sofern nicht über den Taster <AUS> abgeschaltet wurde, beginnt der Zyklus erneut, usw.  
 Wenn ausgeschaltet wurde, aber die Freigabe noch eingeschaltet ist, können im Einrichtungsbetrieb die Bänder "von Hand" mit den entsprechenden Tastern betrieben werden. Dabei ist auf in jeder Situation auf eine entsprechende Verriegelung zu achten, die beim Abräumen Fehlsteuerungen vermeidet.  
 Vor einem Neustart muss "nach Sicht" kontrolliert werden, dass alle Bänder frei sind. Diese Kontrolle muss das Programm nicht übernehmen.



Die klassische Installationsschaltung, mit der von mindestens 3 Stellen unabhängig von einander eine oder mehrere Lampen im Wechsel ein- bzw. ausgeschaltet werden können, ist digital oder in AWL nachzubauen.

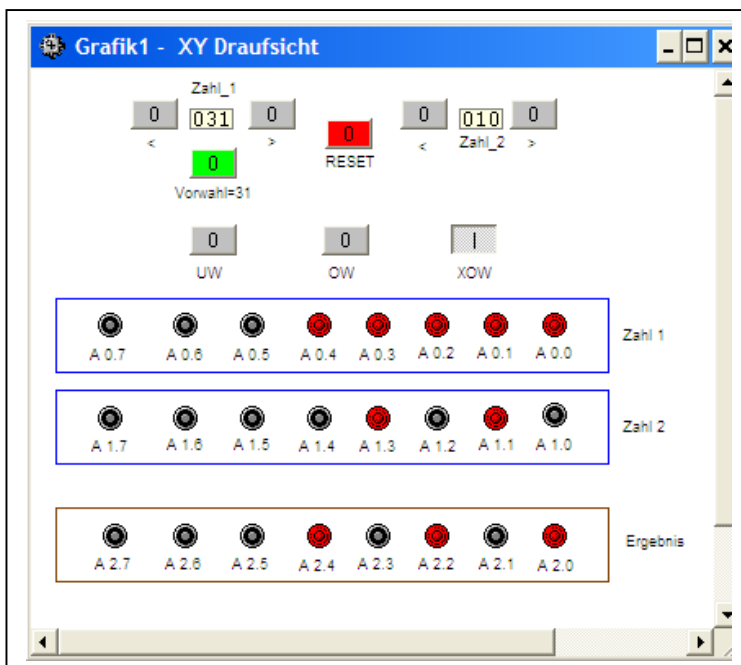


#### Prinzip der Einschaltverriegelung.

Durch diese "Selbstmordschaltung" wird auch bei dauernd oder mehrmals gedrückter "Ein"-Taste nur ein Impuls von der Länge der Zykluszeit gebildet. Dadurch wird verhindert, dass eine klassisch aufgebaute Schrittkette (ohne Graph 7) nachgetriggert werden kann. <Nadelimpuls> blinkt nur kurz auf, wenn <EIN> gedrückt wird.

Die <Speicherglied>-LED wird erst wieder durch die Quittierung auf <AUS> ausgeschaltet.

<ungespeichert> leuchtet bei Druck auf Taster <EIN>

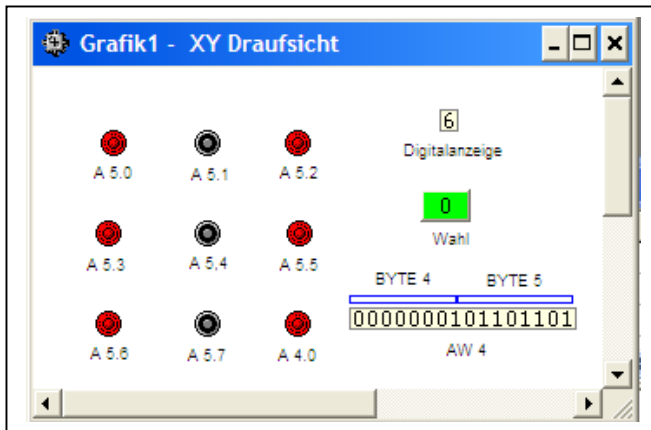


### Testen Sie die Operationen

**"UW", "OW" und "XOW"**

Informieren Sie sich unter  
**TrySim-Hilfe | Index | UW ...**

oder lernen Sie die Bedeutung der Operationen verstehen, indem Sie die LEDs beobachten. Sie können beide Zahlen mit den größer (>)- bzw. kleiner (<)-Tasten durch Dauerdruck verstellen. Außerdem kann Zahl 1 auf 31 vorgewählt werden. Beide Zahlen können mit <RESET> auf 0 zurück gesetzt werden.

**Würfeln durch Antippen der Wahl-Taste.**

Ein Zufallsgenerator wählt eine Zahl, die dem AW 4 zugeordnet wird.

The screenshot shows a graphical user interface for a number-guessing game. The window title is "Grafik1 - XY". The interface includes the following elements:

- Anzahl der Versuche:** A text label with a value of "1".
- Takt:** A green indicator light.
- START:** A button with a circular icon.
- neue Wahl:** A text label with a value of "123".
- zu klein:** A red indicator light.
- zu groß:** A red indicator light.
- Treffer:** A circular icon.
- SOLLWERT:** A text label with a value of "999".
- schummeln:** A cyan button with a value of "0".
- Hand-Vorwahl:** A slider control with a value of "123" and the label "< 1. HAND-VORWAHL >".

Annotations in yellow callouts provide additional information:

- Wahl der neuen Ratezahl:** Points to the "neue Wahl" field.
- Raten bis zum Treffer (ohne Takt):** Points to the "START" button.
- Aktuelle Zahl:** Points to the "neue Wahl" field.
- Verdeckte Ratezahl (Sollwert):** Points to the "SOLLWERT" field.
- Beweglicher Schieber (keine Funktion). Mit <schummeln> fährt er zur Seite und gibt die Lösung frei.** Points to the "schummeln" button.
- Einstellung der ersten Vorschlagszahl vor dem Start.** Points to the "HAND-VORWAHL" slider.
- Weiterschaltung zum nächsten Vorschlag:** Points to the "START" button.

**Aufgabe:**  
Eine verdeckte Zahl zwischen 0 und 255 soll in max. 9 Schritten gefunden werden.

**Bohrstation (Variante als Schrittkette mit Einrichtbetrieb)**

Nach dem Betätigen des Freigabe-Schalters kann mit <Start> die Anlage in Betrieb gesetzt werden, sofern der Bohrkopf den oberen Grenztaster S3 (Ö) berührt und die Lichtschranke <LS> nicht durch einen Rohling auf „1“ gesetzt ist. Diese Startvoraussetzung wird durch die Start-LED angezeigt.

Bei laufendem Prozess darf nicht mit erneutem oder längerem Druck auf <Start> nachgetriggert werden. Erst nachdem der gebohrte Rohling die Lichtschranke <LS\_2> passiert hat, leuchtet die Start-LED wieder. Das Magazin gibt je Zyklus nur einen Rohling frei.

Es kann über den <Betrieb>-Schalter zwischen Einzelzyklus und Dauerbetrieb jederzeit umgeschaltet werden. Mit dem <Stop>-Schalter kann der Prozess in jeder Position angehalten werden. Nach dem Ausschalten ist die Unterbrechung beendet.

Anders verhält es sich bei Betätigung des <NOT-AUS>-Tasters: Nach dem Loslassen darf die Anlage unter keinen Umständen wieder allein anlaufen.

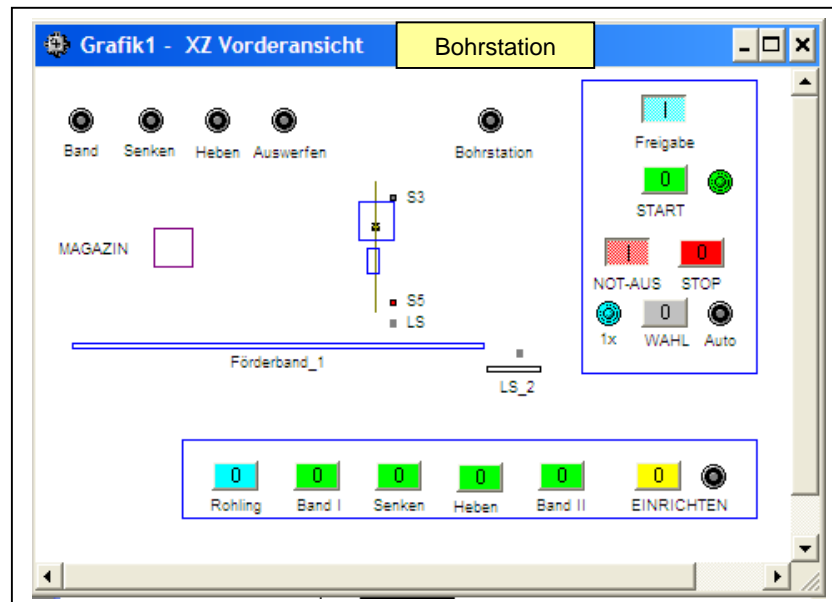
Bei Automatikbetrieb wird der neue Rohling freigegeben, wenn die Lichtschranke wieder frei ist.

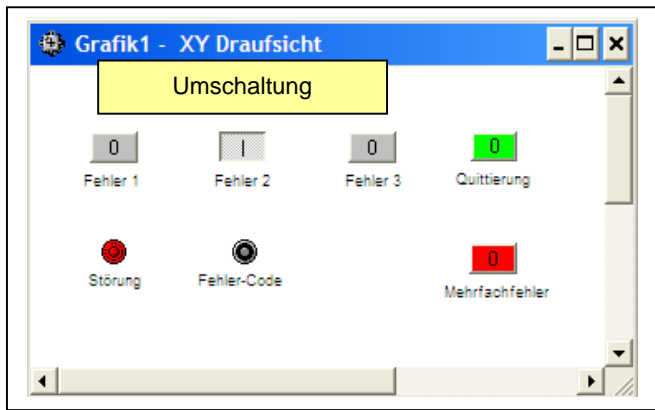
Da die Anlage als Schrittkette programmiert werden soll, kann sie nur wieder in Betrieb gehen, wenn der Rohling entfernt wurde und der Bohrkopf oben in Grundstellung steht (Grundschritt). Das kann durch den Einrichtbetrieb erreicht werden. Dazu ist die <Freigabe> in Nullstellung zu bringen und <Einrichten> einzuschalten. Dieser Zustand wird mit der LED angezeigt. Jetzt ist es möglich, jeden Schritt „von Hand“ mittels entsprechendem Tasterdruck auszuführen. Dabei sind die Endstellungen automatisch zu beachten.

Die Schritte sind über LEDs anzuzeigen. Der Betrieb des Bohrkopfes wird ebenfalls über eine LED angezeigt.

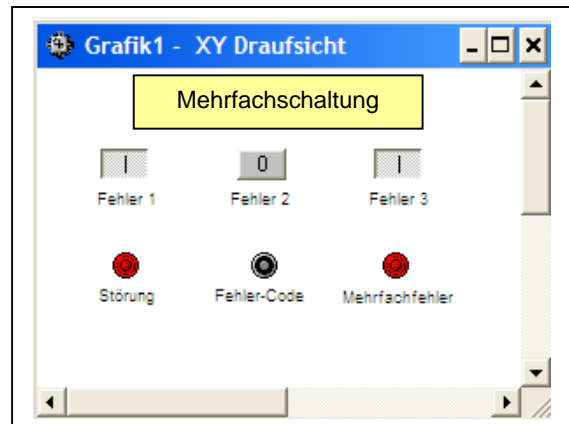
Das Einrichten und der Normalbetrieb sind in getrennten FCs zu programmieren. Die FCs sind nur für ihr Programm zu aktivieren!

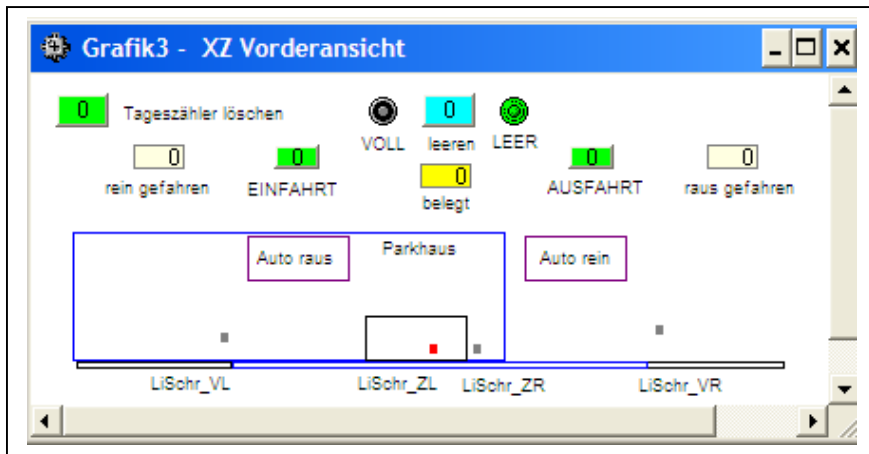
Achtung: Sie müssen verhindern, dass beim Umschalten des Betriebszustandes die falschen Ausgänge (innerhalb der FC) aktiv bleiben!





Verschiedene Störungen sollen durch unterschiedlich häufige Impulse an einer LED angezeigt werden. Zusätzlich leuchtet im beliebigen Störfall ein Dauerlicht. Durch einen zusätzlichen Fehler - simuliert durch <Mehrfachfehler> - ändert sich der Fehlercode in eine Dauerblink-Anzeige.





### Richtungsabhängiges Zählen an einem Tor.

Aufgabe: An einem Tor sollen richtungsabhängig rein- u. rausfahrende Autos gezählt werden. Gezählt wird mit 2 Lichtschranken, die waagrecht in der Zählzone so versetzt angebracht sind, dass ein Auto beide Lichtschranken zur gleichen Zeit abdecken kann.

Erst wenn ein Auto die Zählzone eindeutig in Fahrtrichtung verlassen hat, darf gezählt werden. Wenn ein Auto in die Zählzone einfährt, aber dann zurücksetzt, darf nicht gezählt werden, da es dann ja nicht die Belegung der Parkgarage verändert.

Das eigentliche Zählprogramm, das für den Einsatz am Tor getestet werden soll, wird im OB 1 geschrieben.

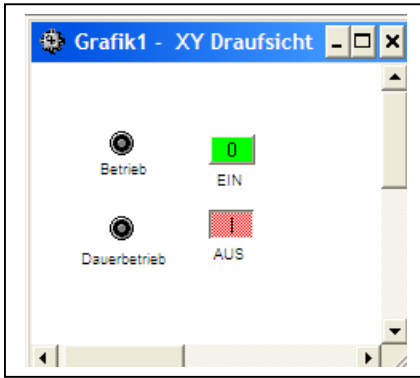
Um die Situationen am Tor zu simulieren, wird das Bewegungsprogramm im OB 2 erstellt. Das hat den Vorteil, dass nach erfolgreichem Test nur das gewünschte Zählprogramm des OB 1 in die SPS geladen werden muss. Der "Rest" spielt sich dann ja real vor dem Tor ab.

Die Simulation soll natürlich Fehlschaltungen vermeiden. Deshalb darf immer nur ein "Auto" in der Anlage sein.

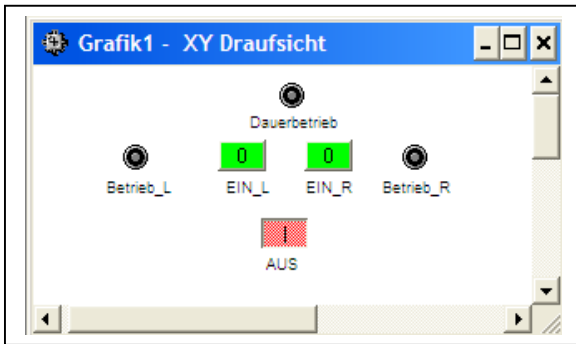
Die Bewegung der Autos erfolgt über Taster mit Dauerdruck, d.h. solange ein Richtungstaster gedrückt wird, kann das Auto rein / raus, bzw. vorwärts / rückwärts fahren.

Es sollen max. 10 Autos einparken können. Die Zustände "voll" und "leer" werden angezeigt.

Wenn kein Auto im Parkhaus steht, kann logischerweise auch keins rausfahren. Das ist bei der Simulation zu berücksichtigen.

**Aufgabe:**

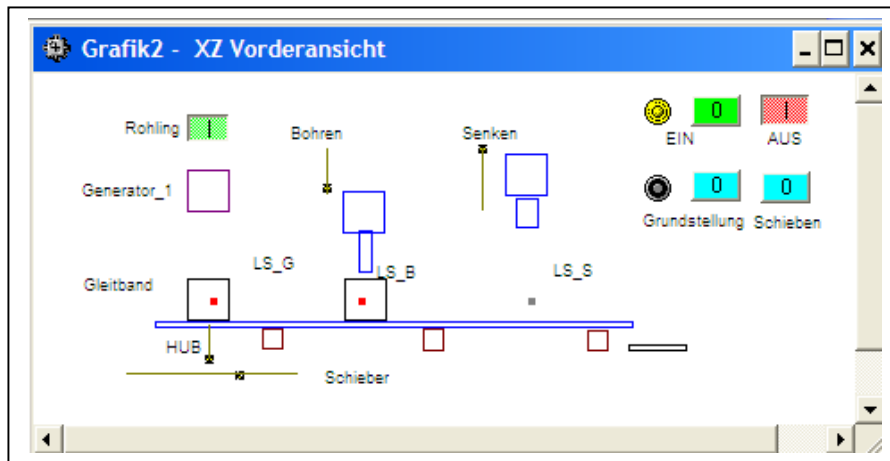
In der "Startphase" von 4 Sekunden gilt Tippbetrieb für die LED <Betrieb>. Erst nach dieser Zeit schaltet sich die Selbsthaltung ein, so dass dann nach dem Loslassen des EIN-Tasters der Dauerbetrieb über eine LED angezeigt wird und die LED <Betrieb> nur über <AUS> gelöscht werden kann.

**Aufgabe:**

2 Förderbänder können jeweils einzeln laufen, d.h., wenn ein Band läuft, kann das zweite Band nicht mehr eingeschaltet werden.

In der "Startphase" von 5 Sekunden gilt Tippbetrieb. Erst nach dieser Zeit schaltet sich die Selbsthaltung ein, so dass dann nach dem Loslassen des EIN-Tasters das entsprechende Band weiterläuft.

Der Dauerbetrieb ist über eine LED anzuzeigen. Ein gemeinsamer AUS-Taster stoppt die Anlage. (3 Netzwerke sollen für die Aufgabe ausreichen).



Ein Rohling soll nacheinander an 2 Stationen gebohrt und entgratet werden.

**Aufgabe 1:** Die Teile fallen aus dem Generator, wenn der <EIN>-Taster und der <Rohling>-Schalter eine „1“ liefern und noch kein Teil die linke Lichtschranke verdeckt. Es darf natürlich nur 1 Rohling je Arbeitszyklus frei gegeben werden.

**Aufgabe 2:** Ein Hubschieber transportiert alle Teile um eine Station weiter, wenn mindestens ein Teil eine Lichtschranke verdeckt. Dazu müssen die Antriebsköpfe der Arbeitsstationen in der oberen Endlage stehen. Dieser Hubschieber wird 1) von unten in die Transportbahn hineingehoben und dann 2) nach rechts bewegt, so dass alle Transportnasen die jeweiligen Teile bis zur nächsten Arbeitsstation mitnehmen. Anschließend wird der Hubschieber wieder abgesenkt und er fährt in die Grundposition zurück.

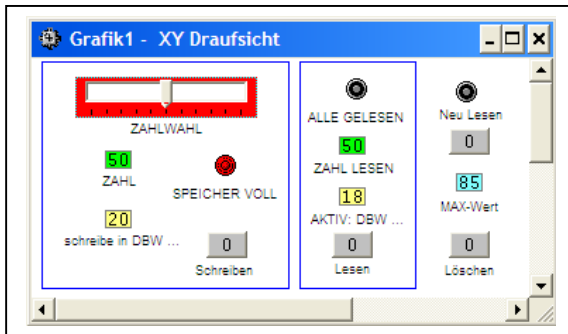
**Aufgabe 3:** Nur wenn die Lichtschranken der Arbeitsstationen feststellen, dass ein Rohling angekommen ist, senkt sich der jeweilige Antriebskopf. Nachdem der untere Endtaster berührt wurde, fährt der Antriebskopf sofort wieder in seine obere Endlage.

**Aufgabe 4:** Im Störfall wird die Anlage mit Druck auf den <AUS>-Taster (!) sofort stillgesetzt. Ein sinnvoller Neustart ist erst möglich, wenn die Rohlinge entsorgt sind. Dazu kann bei ausgeschalteter Anlage entweder <Schieben> benutzt werden, oder die Anlage kann / muss in die Grundstellung gefahren werden.

Auf keinen Fall darf die Maschine nach dem Loslassen von <AUS> wieder anlaufen.

**Hinweis:** Die Endtaster sind in den TrySim-Linearbewegern schon integriert.

Dieses Projekt eignet sich im Schulbetrieb gut für eine arbeitsteilige Projektierung. Die FCs können einfach unter <Projekt> | <Dateien hinzufügen> in ein gemeinsames Projekt eingefügt werden.



### **FIFO** First In – First Out

Beliebige INT-Daten sollen in der Reihenfolge des Eingangs im DB gespeichert und bei Bedarf auch in gleicher Reihenfolge wieder ausgelesen werden. In diesem Beispiel ist der Datenspeicher auf 5 Zeilen begrenzt; er kann aber durch geringfügige Anpassungen beliebig erweitert werden. Der Lesevorgang kann beliebig oft wiederholt werden. Der Maximalwert einer Datenserie wird beim Lesen angezeigt.

Wenn eine Datenserie aus weniger als 5 Wörtern besteht, kann nach dem Lesen aller bisherigen Daten (<ALLE GELESEN> leuchtet) der Speicher weiter gefüllt werden.



Über eine kleine Zählerbaustein-Steuerung soll die Zeile für den Eintrag im DB gewählt werden.

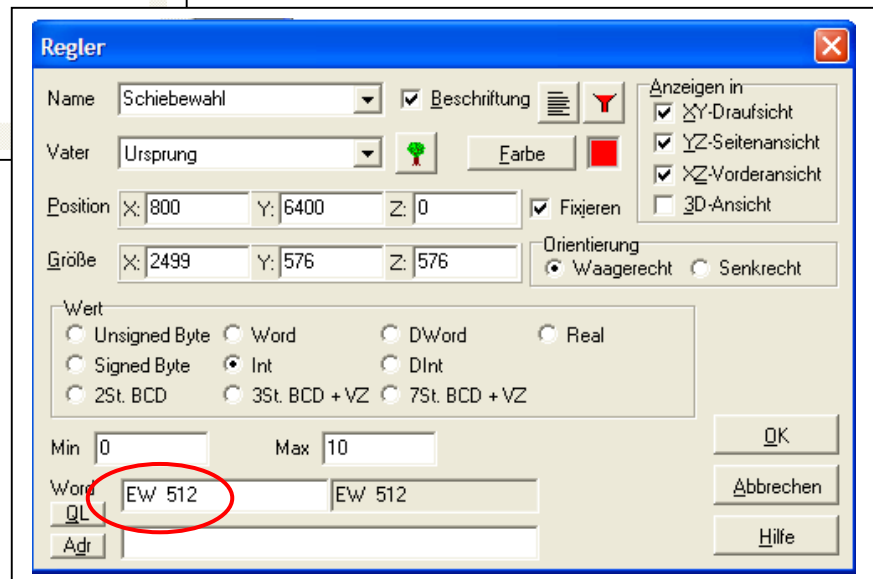
In diesem Projekt soll erklärt, bzw. nachvollzogen werden, wie die Operation LAR1 wirkt. Aufgabe ist es, eine INT-Information im DB zu speichern. Allerdings ist die „Speicherzelle“, die Nummer des DBWs nicht immer gleich. Das DBWx muss also flexibel programmiert werden. Dafür gibt es allerdings keine direkte Operation, sondern sie besteht aus mehreren Programmzeilen. Die Auswirkungen dieses Programms wird durch die Einträge im DB sichtbar gemacht.

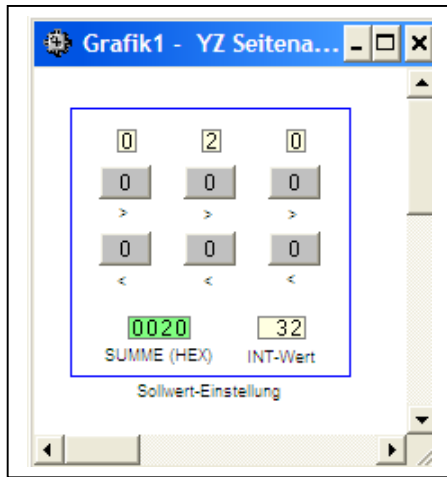
### Teil 2:

Dieses Projekt ist eine Fortführung von <LAR>. Es wurde als eigenständiges Projekt angelegt, damit die DBs beim Testen besser auf dem Monitor zu unterscheiden sind. In <LAR> wurden die Operationen „T DB**B** (x)“ auf einen DB mit BYTE als Typ bzw. „T DB**W** (x)“ auf einen DB mit WORD als Typ angewendet. Was passiert, wenn man Daten mit „T DB**W** (x)“ an einen DB schickt, der BYTE-strukturiert ist? Oder wenn Byte und Word gemischt sind?



Der Schieberegler erlaubt eine freie Einstellung der Zahlen von 0 bis 10.  
Nur bei der Zahl 5 soll <richtig> leuchten, sonst wird <zu klein> bzw. <zu groß> angezeigt.

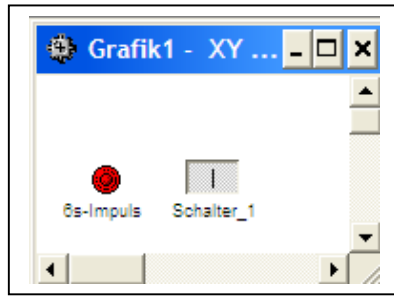




### Mehrstelliger Zifferneinsteller

Über <größer>- bzw. <kleiner>-Taster können 3 Ziffern zwischen 0 und 9 verändert werden. Da insgesamt 3 separate Ziffern gewählt werden sollen, ist 3 x das gleiche Programm durchzuführen. Dafür bietet sich eine FC an.

Anschließend sollen die 3 Ziffern als 3 Stellen einer Zahl verarbeitet und als eine Zahl angezeigt werden. Dabei ist zu klären, welcher Datentyp sich für diese Aufgabe eignet.



Es soll untersucht werden, wie aus einer schlichten Zahl ein S5TIME-Typ wird, damit ein Timerbaustein variabel angesteuert werden kann.

**Grafik1 - YZ Seitenansicht**

**Sollwert-Einstellung**

0 3 0

0 0 0

> > >

0 0 0

< < <

**Zeitfaktor**

+30

Zahl x Faktor = Zeit

0 1 0

Faktor: 0,1 Faktor: 1 Faktor: 10

LED\_1

TEST

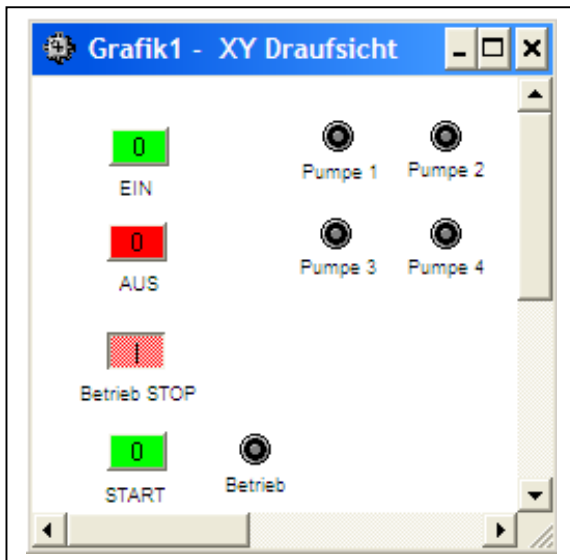
**Restzeit DU - INT**

9999 x 0,1 [s]

29 [s]

9999 x 10 [s]

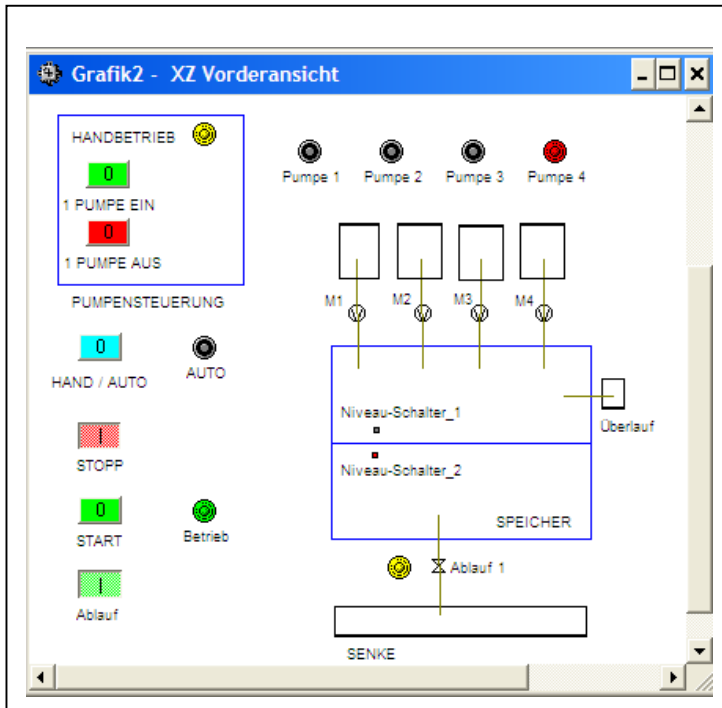
Durch eine 3-ziffrige Zahleneinstellung soll eine Zahl / Zeit vorgegeben werden, die für die Einschaltdauer einer LED benutzt wird. Der Zeitfaktor kann zwischen 0,1 -1- 10 gewählt werden.

**FI-FO** (First in – first out) – ohne Daten

Nach dieser Methode sollen 4 Pumpen ein- bzw. ausgeschaltet werden:

Im Normalbetrieb sollen 2 Pumpen den Öltransport in einer Pipeline aufrecht erhalten. Die umsichtigen Techniker haben die Last auf mehrere Pumpen verteilt, damit bei vollem Betrieb eine Reparatur bzw. ein Austausch möglich ist. Es können auch 3 Pumpen parallel betrieben werden, oder bei geringer Anforderung kann auch nur eine Pumpe laufen.

Entscheidend ist, dass immer die Pumpe, die schon am längsten lief, automatisch ausgeschaltet wird, wenn auf <AUS> gedrückt wird. Bei <EIN> startet die nächste freie Pumpe.



### FIFO – ohne Daten

Dieses Projekt ist eine Variante von <Pumpensteuerung>. Die Anlage wird mit <START> freigegeben und <STOPP> stillgelegt.

Im Handbetrieb können bis zu 3 Pumpen parallel arbeiten. Mit Tippen von Taster <1 PUMPE AUS> wird die Pumpe ausgeschaltet, die zuerst eingeschaltet wurde. Das Ablaufventil kann beliebig geschaltet werden.

Im Automatikbetrieb sorgt die Steuerung bei jeder Ausgangslage dafür, dass sich die Pumpen ständig abwechseln. Jeweils die nächste Pumpe wird eingeschaltet, wenn bei geöffnetem Ventil der untere Niveauschalter erreicht wird. Diese Pumpe wird ausgeschaltet, wenn der obere Niveauschalter erreicht wird.

Als „Erschwernis“ der Programmierung sollen die Pumpen im Automatikbetrieb erst verzögert ein- bzw. ausschalten.

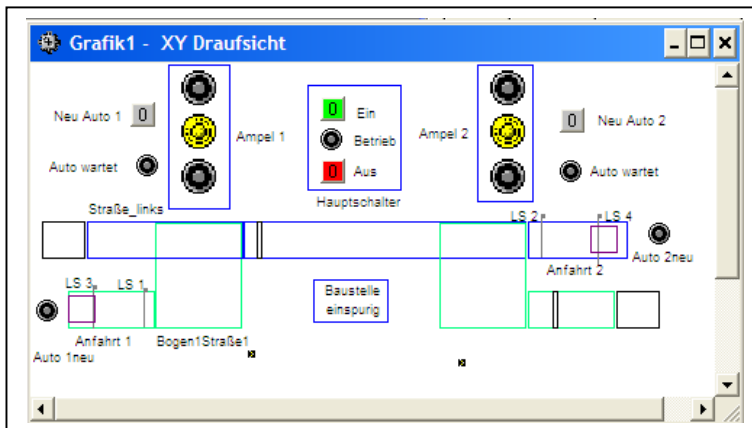
In diesem Beispiel soll die Verwendung des TrySim-**Bogenbandes** demonstriert werden. Es wird nicht nur das SPS-Programm erläutert, sondern auch die Konstruktion des Simulationsaufbaus.

### **Aufgabe: Baustellenampel**

Wenn nur von einer Seite Autos ankommen, schaltet die Ampel nicht nach festen Intervallen um, sondern erst, wenn von der Gegenseite auch ein Auto durchfahren möchte. Die Autos kommen nicht in fester Taktrate, sondern - wie im richtigen Leben - in unterschiedlicher Häufigkeit.

Für die Simulation ist es nicht hilfreich, wenn bei befahrener Straße plötzlich der Stecker gezogen wird (soll heißen: wenn ausgeschaltet wird). Dann würde beim Neustart leicht ein Blechschaden entstehen, weil noch die übrig gebliebenen Autos verlassen rumstehen. Deshalb sollen die Bänder (Straßen) nach dem Ausschalten ganz leer laufen.

Sollten dennoch bei Ihrer Programmierung die Autos kollidieren und ein großes Knäuel bilden, können Sie schnell - bei laufender Anlage - den Abschleppwagen beauftragen: Mit **Ansicht | Dynamiks | Normale löschen** können alle „Autos“ von der Fahrbahn entfernt werden.



Nach Tippen auf <EIN> blinken die gelben Lampen.

Die blaue Straße besteht aus Teilstücken, die je nach Betriebszustand separat oder gemeinsam eingeschaltet werden können.

Die grünen „Straßen“ bilden die An- und Abfahrt der Gegenspur. In der Einspurigen Baustelle wird das blaue Mittelstück in die Gegenrichtung geschaltet.



Wenn der Schalter eingeschaltet wird, sollen wiederkehrend zwei kurze LED-Blitze entstehen.  
Zur Lösung der Aufgabe wird ein einschaltverzögerter Timer verwendet, dessen Ablaufzeit mit festen Zahlenvorgaben verglichen werden. Bei Wertgleichheit blitzt die LED kurz auf.



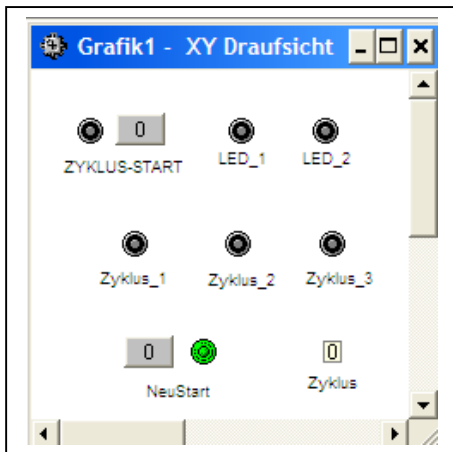
### **Ringregister - einmal anders**

1) Wenn der Schalter <Betrieb> auf "1" steht, wird mit einer Taktzeit von 0,5 s bis zur Zahl 10 hochgezählt. Anschließend beginnt die Zahlenfolge automatisch wieder bei -5.

Mit dem Taster <-5> kann jederzeit der Zählerstand auf -5 zurückgesetzt werden.

Das Taktsignal wird als LED und der Zählerstand wird digital angezeigt.

2) Ab -2 bis einschließlich +2 soll die LED <Bereich> leuchten.



### Einschalt-Verriegelung

**Aufgabe:** Eine LED zeigt an, dass neu gestartet werden kann. Bei Tasterdruck auf <NeuStart> zeigt die LED <Zyklus-Start> an, dass bei Tasterdruck auf <Zyklus-Start> ein zweiphasiger Zyklus beginnt. (Die LED erlischt bei Tasterdruck):

- a) LED\_1 leuchtet 2 s lang; dann erlischt sie wieder und
- b) LED\_2 leuchtet 3 s lang.

Danach ist eine erneuter Zyklusstart durch Tasterdruck möglich; die entsprechende LED leuchtet wieder.

Insgesamt sind 3 Zyklen zu durchlaufen. Welcher Zyklus gerade aktiv ist, wird durch die LEDs <Zyklus\_x> angezeigt. Außerdem ist die Zykluszahl digital anzuzeigen.

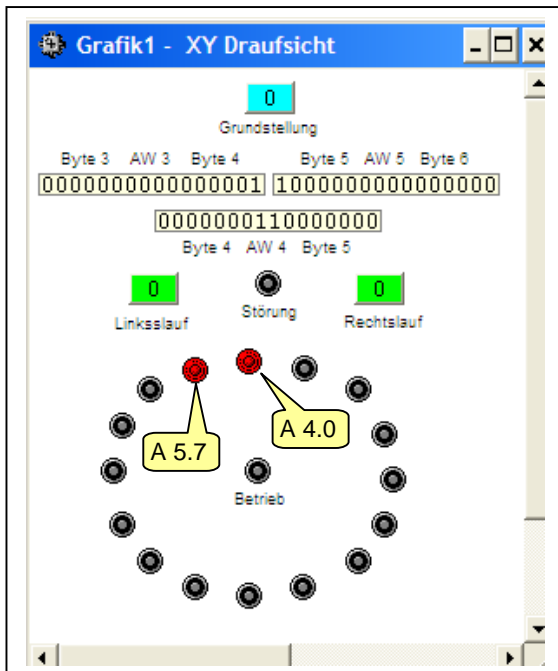
Nach dem 3. Zyklus leuchtet die NeuStart-LED. Durch <NeuStart> wird die Anzeige auf 0 gesetzt und der Zyklus-Start kann neu beginnen. Natürlich darf während des Ablaufes nicht nachgetriggert werden!

**Aufgabe:**

Durch wiederholtes Drücken eines Tasters sollen nacheinander 3 LEDs eingeschaltet werden. Die LEDs leuchten einzeln, d.h., wenn die zweite einschaltet, erlischt die erste, usw.

Im vierten Takt erlischt LED\_3, d.h. das Programm ist wieder in der Ausgangsstellung und startet mit einem erneuten Tasterdruck.

Das Programm reagiert auf den Tasterdruck (nicht auf das Loslassen des Tasters).

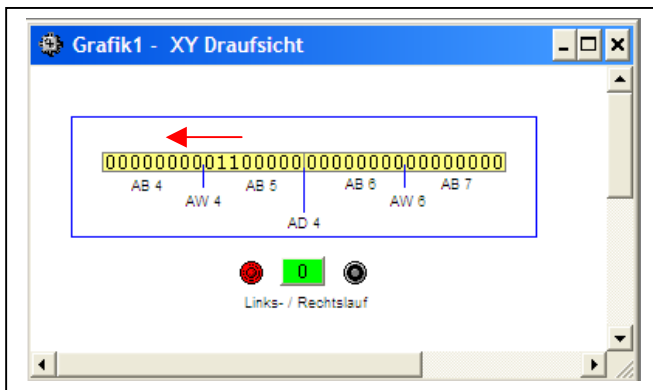


Im AW 4 liegt der Zahlenwert 384 (als Dualzahl angezeigt). (Die Zahlendarstellung beim <Baustein beobachten> kann bei TrySim unter Ansicht | Option gewählt werden). Die Eigenschaften der Digitalanzeige in der Grafik1 sind im Grafik-Editor dieser Elemente zu wählen. Zu erkennen ist die Lage von Byte 4 und Byte 5 im Word 4. Ebenso das Zusammenwirken von Word 3, 4 und 5.

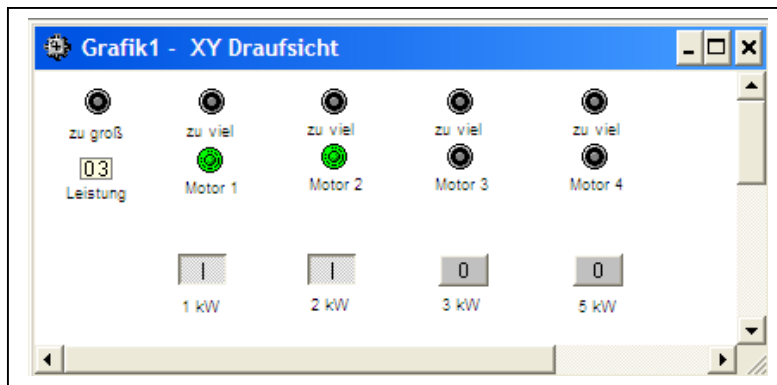
Der Zahlenwert 384 wird automatisch beim Start des Programm gewählt, bzw. bei Druck auf <Grundstellung>.

Bei Druck auf die Schalter <Linkslauf> bzw. <Rechtslauf> leuchtet die Betriebs-LED und die leuchtenden LEDs und die zugehörigen Bits der Digitalanzeige wandern weiter.

Werden beide Schalter eingeschaltet, blinkt die Störungs-LED und die aktiven Bits bleiben stehen.

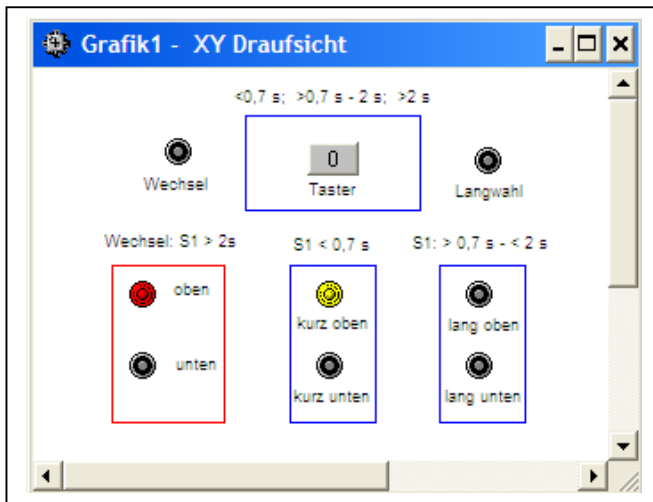


Beim Anlage-Start (nicht mit dem grünen Richtungsschalter, sondern mit dem TrySim-Button) starten im AD4 2 Bits als Wert 384, bewegen sich mit Taktgeschwindigkeit nach links, verlassen die Digitalanzeige und treten von rechts kommend wieder in die Anzeige ein. Die Bewegungsrichtung kann jederzeit geändert werden.

**Aufgabe:**

4 Motoren mit unterschiedlicher Leistungsaufnahme können in beliebiger Kombination eingeschaltet werden. Allerdings darf die Gesamtleistung 5 kW nicht überschreiten. Die momentan dem Netz entnommene bzw. gewünschte Leistung wird digital angezeigt.

Wird die zulässige Leistung überschritten, leuchtet eine zugehörige Warnlampe. Sonst ändert sich nichts. Das Programm soll rechnen und nicht über 2<sup>n</sup> Verknüpfungen fest vorprogrammiert werden.



Mit einem Taster sollen 4 Lampen geschaltet werden. Das kann nur über ein Impulsmuster oder durch eine Zeitsteuerung ermöglicht werden.

Hier wird die Zeitsteuerung vorgestellt.

Es werden 2 Gruppen à 2 Lampen unterschieden. Wenn der Taster länger als 2 s gedrückt wird, wird die jeweils nicht angewählte Gruppe für weitere Befehle vorgewählt.

Wird der Taster nur kurz ( $< 0,7$  s) gedrückt, wird die LED „kurz“ der entsprechenden Gruppe umgeschaltet.

Wird der Taster länger als 0,7 s aber kürzer als 2 s gedrückt, schaltet die LED „lang“ der vorgewählten Gruppe um.

**Aufgabe:**

Es ist ein einfacher 3-Punkt-Regler zu programmieren. Man könnte es auch einen 2-Punkt-Regler mit fester Hysterese nennen.

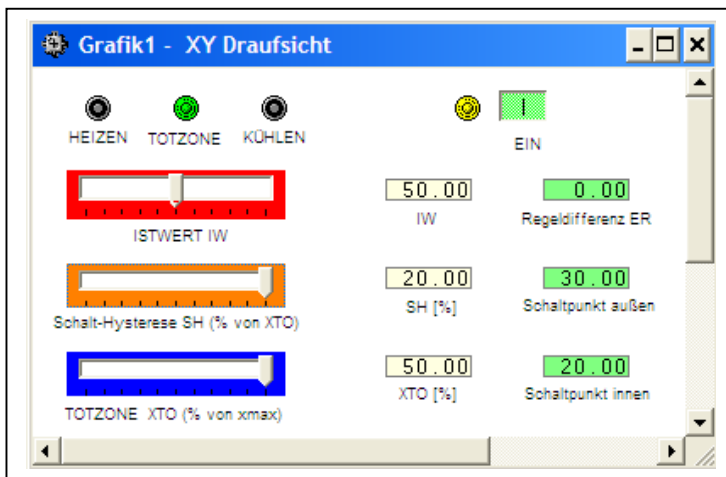
Unterhalb des unteren Schaltpunktes mit dem Wert 3 soll die LED „zu klein“ leuchten.

Ab dem Wert 8 und darüber soll die LED „zu groß“ leuchten.

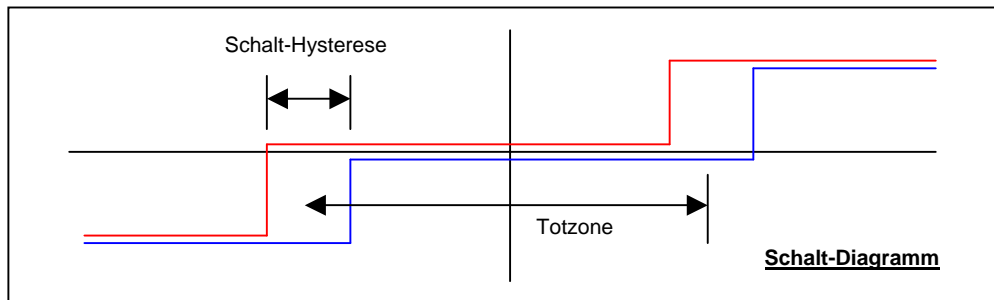
Bei einem Wert zwischen den Schaltpunkten leuchtet die LED „richtig“.

Der augenblickliche Istwert ist digital anzuzeigen.

**Hinweis:** Der Istwert kann am Schieberegler <Schiebewahl> verändert werden.



Bei diesem 3-Punkt-Regler kann die Totzone zwischen 0 und 50 % des Regelbereichs eingestellt werden. Zusätzlich kann die Schalt-Hysterese, also die Übergangszone zwischen 2 Schaltzonen, zwischen 0 und 20 % der Schaltzonen, eingestellt werden. (Die Einstellung der Schalt-Hysterese verändert sich mit dem Wert der Totzone).



### 3-Punkt-Regler mit Schalthysterese

Der 3-Punkt-Regler der Praxis hat für den oberen und unteren Schaltpunkt eine Schalthysterese, d.h., je nachdem, ob der Istwert größer oder kleiner wird, wird bei einer anderen Temperatur geschaltet.

Dieses Verhalten soll nachgebaut werden.

Um den Regler für verschiedene Regelstrecken zu verwenden, sollen die Parameter über Schieberegler einstellbar sein.

Der Istwert ist als Rechengröße und in Grad Celsius anzuzeigen.

Damit die internen Rechengrößen besser verfolgt und verstanden werden, sind die Regeldifferenz und die Schaltpunkte auch anzuzeigen.

Es ist symbolische Adressierung zu verwenden.

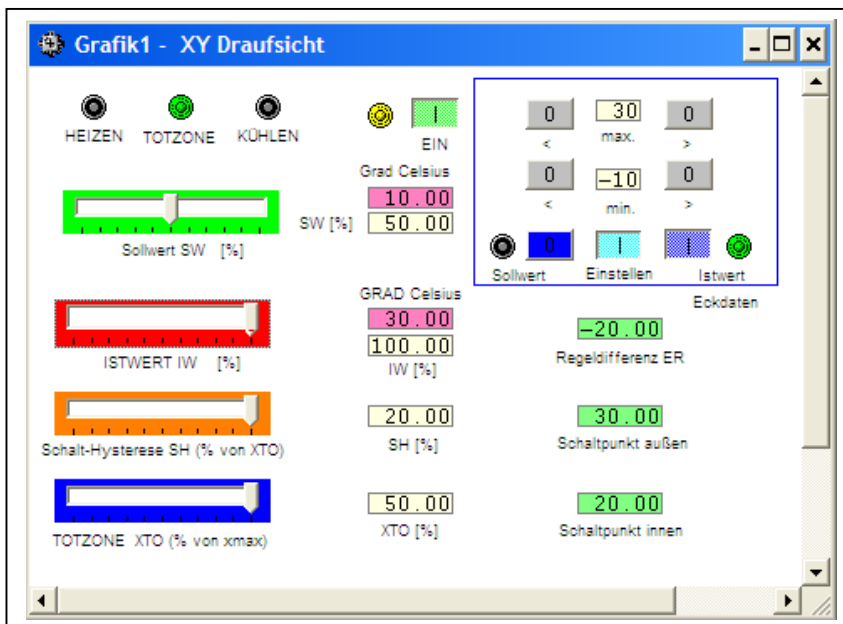
Im ausgeschalteten Zustand weist ein Blinklicht auf den Einschalter hin. Bei Betrieb leuchtet Dauerlicht bei gleicher LED. Die Schaltzustände werden ebenfalls durch LEDs angezeigt.

Die minimalen und maximalen Temperaturen sind über Vorwahltaster festzulegen. Dabei dienen durch eine Umschaltung die Taster sowohl für Soll- und Istwerte. Die Werte werden in Instanz-DBs gespeichert.

Die Werte können nur eingestellt, bzw. geändert werden, wenn Schalter <Einstellen> geschlossen ist.

Die Schieberegler für die Soll- und Istwerte sind für 0 - 100 % eingerichtet. Durch die Festlegung der Min.- und Maxwerte werden der Aussteuerung auch die Temperaturen zugeordnet.

Parallel zur %ualen Anzeige der Aussteuerung sind die Soll- und Isttemperaturen anzuzeigen. Die Werte sind also zu normieren.



Die Totzone XTO ist auf max. 50 % des Regelbereichs programmiert, die Schalthysterese auf max. 20 % XTO.

Der Soll- und Istwertbereich kann nacheinander mit beliebigen Ober- und Untergrenzen eingestellt werden.

Soll- u. Istwert können als %-Wert und in °C abgelesen werden.

Die inneren und äußeren Schaltpunkte werden in Abhängigkeit von der gewählten Schalt-Hysterese in % des Regelbereichs angezeigt.

**Aufgabe:**

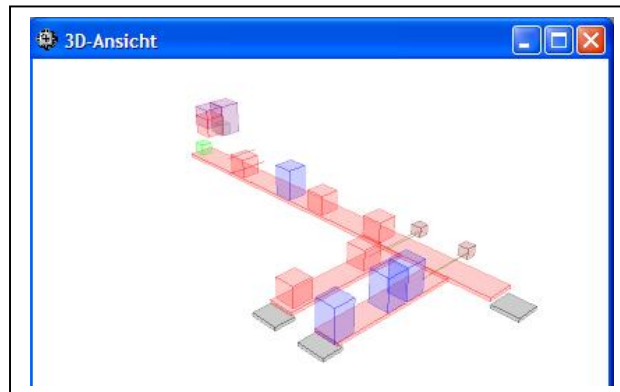
Auf einem Zulieferband kommen in unterschiedlicher Reihenfolge 2 verschiedene Teile an. An einer Mess-Station erkennen 2 Lichtschranken, ob es sich jeweils um ein großes oder kleines Teil handelt. Dementsprechend wird 1 Bit als "0" oder "1" in einem Wort abgelegt, welches das Teil beim Weitertransport begleitet.

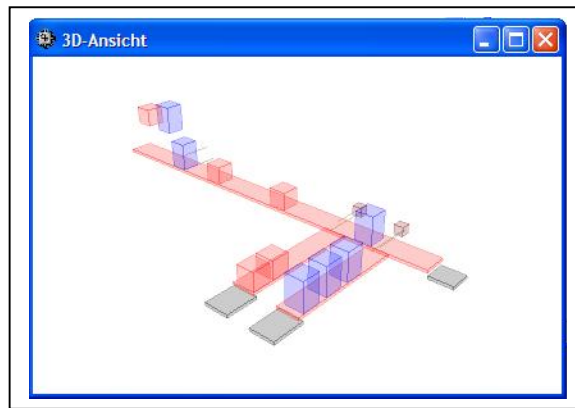
Am ersten Sortierband erkennt eine weitere Lichtschranke die Ankunft eines Teiles. Dort muss überprüft werden, ob es ein kleines Teil ist. Hier wird nicht gemessen, sondern durch Datenvergleich wird die Auswahl zum Auswurf getroffen.

Die Hauptaufgabe besteht darin, jedem Teil eine individuelle Marke für „kleines Teil“ oder „großes Teil“ mit auf den Weg zu geben. Dazu wird ein Schieberegister benutzt, in dem für jedes neue Teil ein zusätzliches Mess-Bit („0“ oder „1“) links von den bisherigen Bits (Teilen) eingefügt wird.

Dazu muss erfasst werden, wie viele Teile bereits durch die Messstation transportiert wurden, bzw. wie viele Teile noch auf dem Band zwischen der Messstation und der Auswurfweiche liegen.

Mit jedem neuen Teil wird ein Zähler um 1 heraufgesetzt und mit jedem ausgeworfenen Teil um 1 verringert.





### **Sortieranlage (Anwendung des Datenbausteins)**

**Dieses Projekt ist eine Variante von SortierStrecke\_4. Die Steuerbits werden im DB abgelegt.**

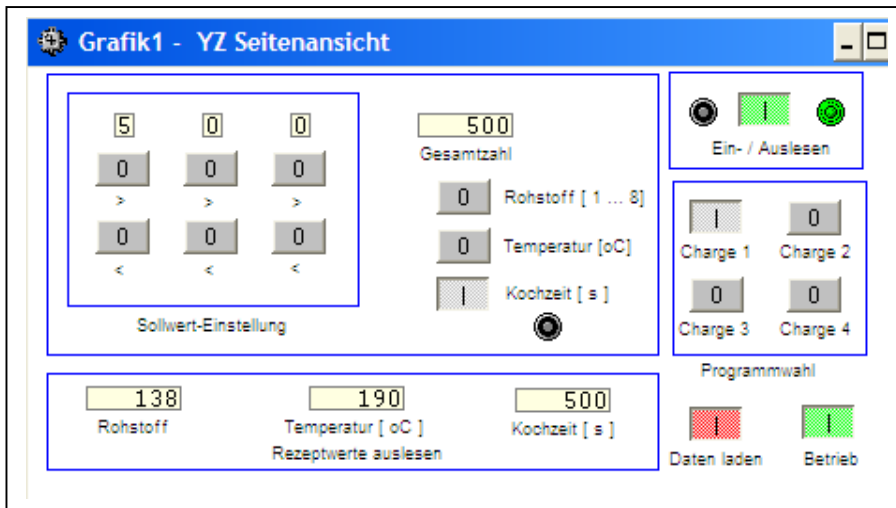
Auf einem Zulieferband kommen in unterschiedlicher Reihenfolge 2 verschiedene Teile an. An einer Mess-Station erkennen 2 Lichtschranken, ob es sich jeweils um ein großes oder kleines Teil handelt. Dementsprechend wird 1 Bit als "0" oder "1" in einem Wort abgelegt, welches das Teil beim Weitertransport begleitet.

Am ersten Sortierband erkennt eine weitere Lichtschranke die Ankunft eines Teiles. Dort muss überprüft werden, ob es ein kleines Teil ist. Hier wird nicht gemessen, sondern durch Datenvergleich wird die Auswahl zum Auswurf getroffen.

Die Betriebsart (Art der Zulieferung) kann bei laufendem Band umgeschaltet werden.

Bei Handbetrieb kann jedes Magazin einzeln angesteuert werden, indem der entsprechende Taster gedrückt wird. Bei längerem Drücken fallen nach eingestellter Verzögerungszeit neue Teile auf das Band.

Die Wirkungsweise der Auswertelogik kann somit ganz individuell verfolgt werden.



In einer Konservenfabrik sind auf Knopfdruck 4 verschiedene Produkte zu bearbeiten. Für jedes Produkt sind jeweils 3 Sollwerte zu liefern.

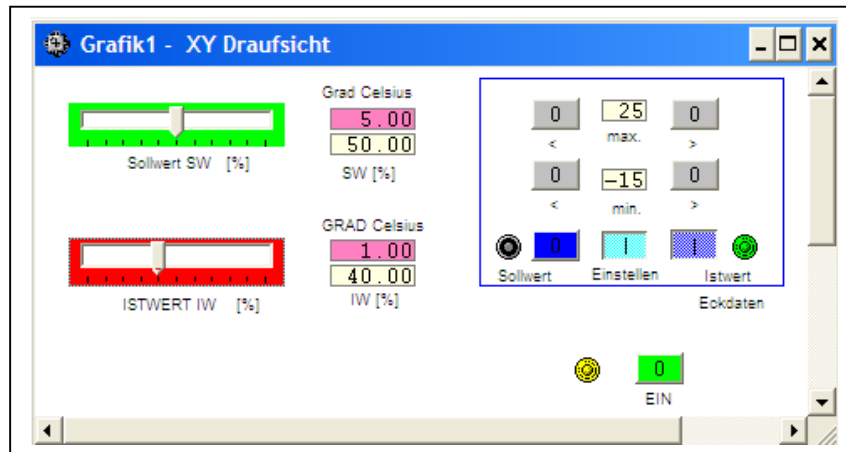
Die Parameter für den Rohstoff, die Bearbeitungstemperatur und die Kochzeit können für jedes Produkt mit einem 3-stelligen Zifferneinsteller (der für alle Werte gilt) vorgegeben werden. Dafür ist nur zwischen den Schaltern <Rohstoff>, <Temperatur> bzw. <Kochzeit> umzuschalten. Das Programm stellt die 3 getrennten Ziffern zu einer Zahl zusammen.

Zum Einlesen neuer Werte muss der Schalter <Daten laden> betätigt und das entsprechende Produkt (die "Charge") ausgewählt werden. Der <Ein- / Auslesen>-Schalter steht auf "0", d. h. auf "Einlesen". (Bei ausgeschaltetem Schalter <Daten laden> kann also an den Einstellungen gefahrlos gespielt werden).

Die neuen Chargendaten werden erst mit <Ein- / Auslesen> = „1“ in den Datenspeicher geladen. Nachdem <Daten laden> ausgeschaltet wurde, kann mit <Betrieb> das Programm aktiviert werden.

Zum wirksamen Umschalten zwischen den Chargen, d.h. zum Laden neuer Parameter aus dem DB sind <Daten laden> = „1“ und <Auslesen> = „1“ erforderlich.

Die aktuelle Rezepturparameter werden angezeigt.



### Normierung von Messwerten

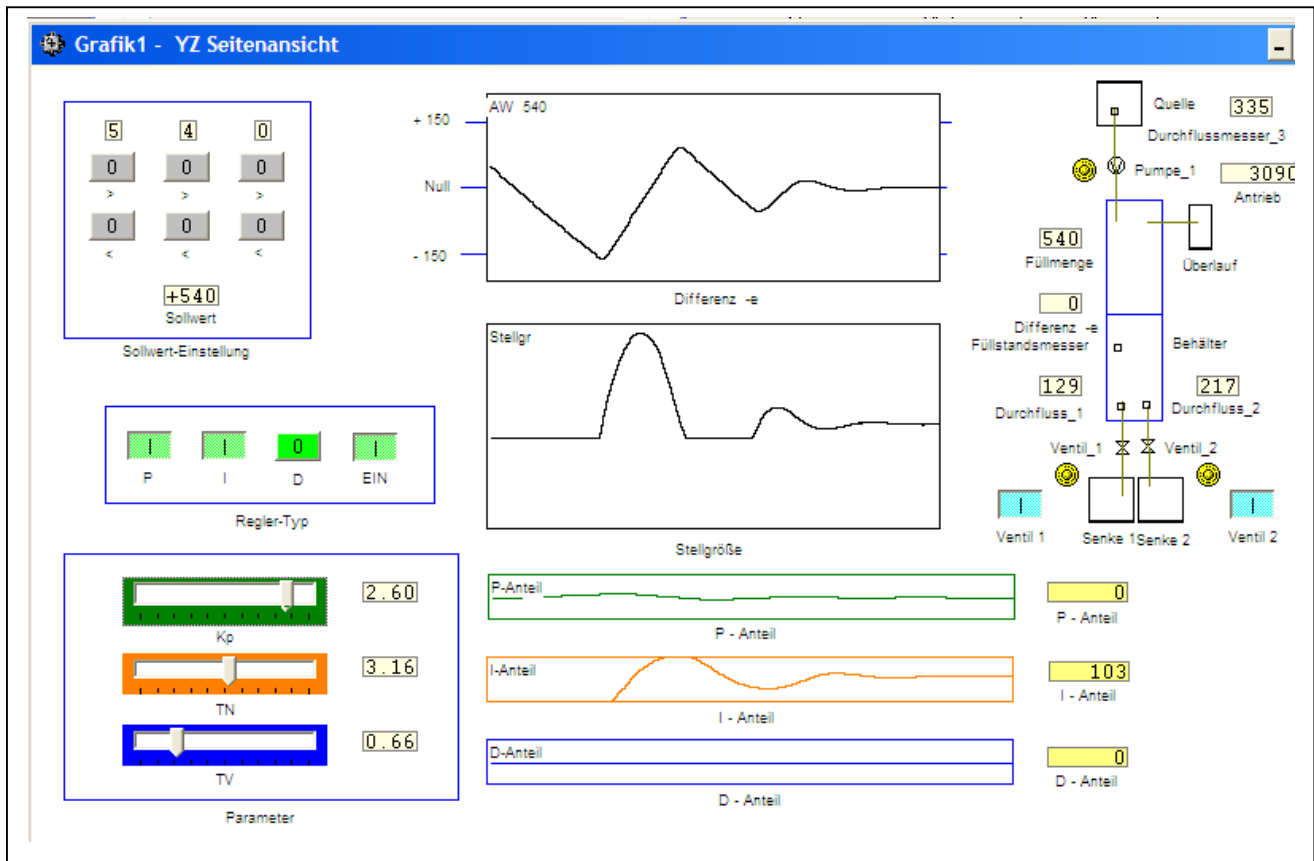
Um einen Regler für verschiedene Regelstrecken verwenden zu können, sollen die Parameter über Schieberegler im Bereich von 0 - 100 % einstellbar sein. Für den Bediener ist aber die tatsächliche physikalische Größe wichtig. Deshalb sind Soll- und Istwert als Rechengröße und in Grad Celsius anzuzeigen. Die Werte sind also zu normieren.

Die Werte können nur eingestellt, bzw. geändert werden, wenn Schalter <Einstellen> geschlossen ist.

Die Schieberegler für die Soll- und Istwerte sind für 0 - 100 % eingerichtet. Durch die Festlegung der Min.- und Maxwerte werden der Aussteuerung auch die Temperaturen zugeordnet.

Es ist symbolische Adressierung zu verwenden.

Im ausgeschalteten Zustand weist ein Blinklicht auf den Einschalter hin. Bei Betrieb leuchtet Dauerlicht bei gleicher LED. Die minimalen und maximalen Temperaturen sind über Vorwahltaster festzulegen. Dabei dienen durch eine Umschaltung die Taster sowohl für Soll- und Istwerte. Die Werte werden in Instanz-DBs gespeichert.



### Dieses Projekt simuliert einen kompletten Füllstands-Regelkreis.

Über einen dreistelligen Sollwert-Zahleneinsteller und separat wählbaren P, I und D-Anteilen kann das Regelverhalten an einer Füllstandsstrecke untersucht werden.  $K_p$ ,  $T_N$  und  $T_V$  können über Schieberegler verstellt werden. Die Füllstandshöhe (Istwert) wird vom (unscheinbaren) Füllstandsmesser erfasst. Das Zeitverhalten wird durch Oszillographen aufgezeichnet. 2 gesteuerte Ablaufventile simulieren unterschiedliche Störgrößen. Die Regelung funktioniert erst, nachdem der <EIN>-Schalter betätigt und ein Reglerparameter  $> 0$  eingestellt wurde. Eventuell muss auch ein Ablaufventil geöffnet werden, damit die Regelung eingreifen kann.