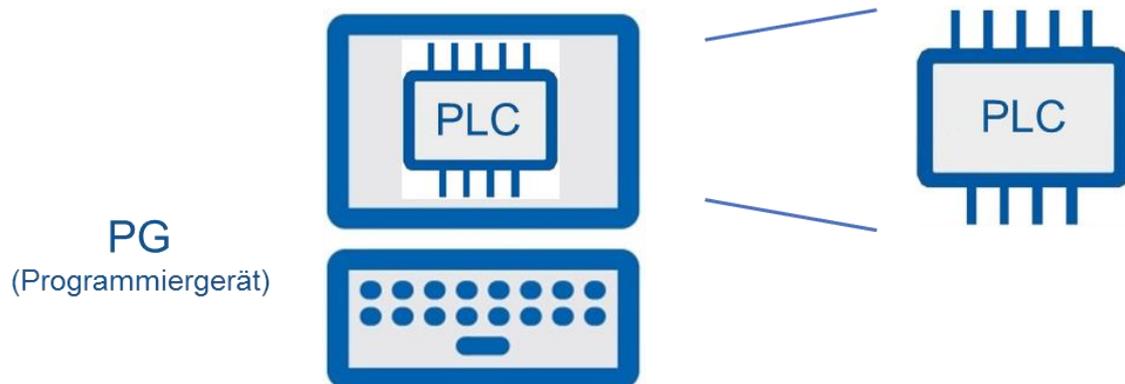




Akademie für Lehrerfortbildung

## *Digitale Transformation*

### *Grundlagen der SPS-Programmierung in einer industrietauglichen Projektierungsumgebung (TIA-Portal)*



*Interdisziplinäre Qualifizierung von Lehrkräften in den Berufsfeldern Elektrotechnik, Metalltechnik und Informationstechnologie*



**Inhalt**

Impressum .....	2
Didaktische Überlegungen .....	3
Exemplarische Lernsituationsbeschreibung.....	4
Technische Überlegungen .....	7
Lab 01 – Projekt anlegen und Hardware einfügen.....	8
Lab 02 – Adressierung und Variablentabelle.....	10
Lab 03 – Programmierung und Simulation.....	14
Lab 04 – Erweiterung des Programms.....	17
Lab 05 – PG mit PLC vernetzen und übertragen .....	19
Lab 06 – Programmcode organisieren .....	23
Lab 07 – Programm in Bibliothek ablegen.....	27
Lab 08 – Bibliotheken in anderen Projekten nutzen .....	29
Lab 09 – Programmieren und Inbetriebnahme eines mechatronischen Systems .....	31
Objekt- und Variablennennung .....	37
Ausstattung für Laborübungen .....	41
Wirtschaft 4.0 - Fortbildungsmodule .....	42

## IMPRESSUM

- Herausgeber: Akademie für Lehrerfortbildung und Personalführung  
Kardinal von Waldburg-Str. 6-7  
89407 Dillingen/Donau
- Redaktionsgruppe: Andreas Ferdinand, Staatliche Berufsschule 1 Kempten (Allgäu)  
Karim Mir Sadry, Staatliche Berufsschule 1 Passau  
Johannes Streifinger, Staatliche Berufsschule 1 Passau  
Stephan Waibel, Staatliche Berufsschule 1 Kempten (Allgäu)
- Redaktionsleitung: Michael Lotter, Akademie Dillingen
- URL: <http://alp.dillingen.de>
- Mail: [m.lotter@alp.dillingen.de](mailto:m.lotter@alp.dillingen.de)
- Stand: Oktober 2022

Dieses Dokument steht unter einer CC BY-SA 4.0-Lizenz. Urheber ist die genannte Redaktionsgruppe der Akademie für Lehrerfortbildung und Personalführung, Dillingen.



**Karl-Peter-Obermaier-Schule**  
**Staatliche Berufsschule 1 Passau**  
**Staatliche Fachschulen für Elektrotechnik und Maschinenbau**

## DIDAKTISCHE ÜBERLEGUNGEN

Das Fortbildungsmodul „Grundlagen der SPS-Programmierung in einer industrietauglichen Projektierungsumgebung (TIA-Portal)“ fördert Kompetenzen und Fertigkeiten, die für Lehrkräfte in den Berufsfeldern Metalltechnik, Elektrotechnik und Informationstechnologie gleichermaßen erforderlich sind, um die Anforderungen der Industrie 4.0-Entwicklungen im Unterricht beruflicher Schulen zu berücksichtigen.

Zum Tätigkeitspektrum eines Elektronikers für Betriebstechnik und Mechatronikers gehört das Programmieren von einfachen mechatronischen Systemen. Ein solches System wird durch das in diesem Modul verwendete Förderband dargestellt. Es bietet einen praxisorientierten Ansatz, um erste Lernschritte im Bereich der Automatisierungstechnik zu fördern. Diese sind erforderlich, um den Lernenden zu befähigen, sich komplexere Anlagen zu erschließen. Das mechatronische System kann in verschiedenen Ausbaustufen bis hin zu einer zukunftsorientierten, digitalisierten industriellen Produktion erweitert werden. Folgende Ausbaustufen sind z. B. möglich:

- Horizontale Vernetzung von Automatisierungskomponenten
- Vertikale Vernetzung des Produktionsprozesses
- Ortsunabhängiges Steuern und Überwachung von Prozessen
- Selbststeuernde Produktion

Durch die zunehmende Komplexität in gehobenen Ausbaustufen der Anlage ergeben sich Tätigkeitsfelder, welche auch andere Berufsbilder bedienen werden. Zum Beispiel berühren die Vernetzung und Konfiguration von industriellen Netzwerkkomponenten Tätigkeitsfelder eines Fachinformatikers.

Der Industriemechaniker zu dessen Aufgabenbereich u. a. die Sicherung der Betriebsfähigkeit automatisierter Systeme gehört, kann durch diese Fortbildung die benötigten Kompetenzen in diesem Bereich anbahnen.<sup>1</sup>

Berührungspunkte ergeben sich auch zu kaufmännischen Berufen. Durch die vertikale Vernetzung zu einem ERP-System, werden Absprachen zwischen Produktionsebene und Planungsebene in Zukunft immer wichtiger.

---

<sup>1</sup> vgl. Handreichung ISB: Wirtschaft 4.0 an beruflichen Schulen, 2018, S. 86

## EXEMPLARISCHE LERNSITUATIONSBESCHREIBUNG

### Am Beispiel des Berufsfeldes Mechatroniker/in

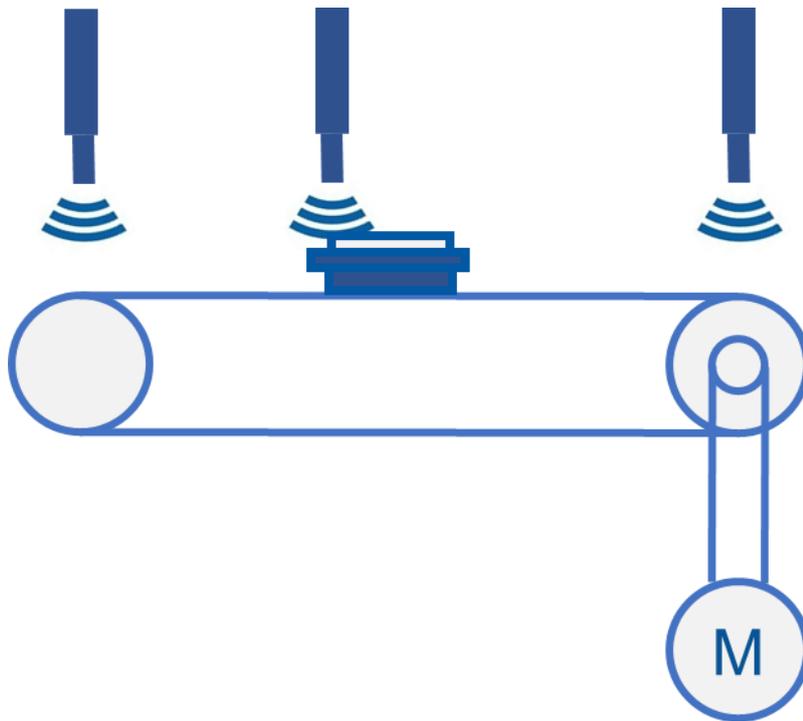
<b>Grundlegende Informationen</b>
<b>Beruf:</b> Mechatroniker/-in
<b>Jahrgangsstufe:</b> 11
<b>Lernfeld:</b> Realisieren mechatronischer Teilsysteme
<b>Thema:</b> Programmierung von einfachen Bewegungsabläufen und Steuerungsfunktionen
<p><b>Kernkompetenz des Lernfeldes:</b></p> <p>Sie kennen Möglichkeiten zur Realisierung von Linear- und Rotationsbewegungen mittels elektrischer, pneumatischer und hydraulischer Komponenten und wenden Kenntnisse über Steuerungen und Regelungen an, um Weg- und Bewegungsrichtung zu beeinflussen. Einfache Programmierverfahren werden beherrscht.</p>
<p><b>Ausgewählte Teilkompetenzen der Lernsituation</b></p> <ul style="list-style-type: none"> <li>➤ Programmieren von einfachen Bewegungsabläufen und Steuerungsfunktionen</li> <li>➤ Schaltungen entwerfen</li> <li>➤ Wirkungsweise von Sensoren und Wandlern verstehen</li> <li>➤ Funktion von Komponenten prüfen und Fehlern beseitigen</li> </ul>
<b>Geschätzter Zeitumfang:</b> 12 x 45 Minuten

## Am Beispiel des Berufsfeldes Elektroniker/in

<b>Grundlegende Informationen</b>
<b>Beruf:</b> Elektroniker für Betriebstechnik/Elektronikerin für Betriebstechnik
<b>Jahrgangsstufe:</b> 11
<b>Lernfeld:</b> Steuerungen für Anlagen programmieren und realisieren
<b>Thema:</b> Steuerungen unter Berücksichtigung der Anlagenfunktionen in Betrieb nehmen und programmieren
<p><b>Kernkompetenz des Lernfeldes:</b></p> <p>Die Schülerinnen und Schüler nehmen Steuerungen unter der Berücksichtigung der Anlagenfunktionen in Betrieb.</p> <p>Sie überprüfen selbständig die Funktion von Steuerungen.</p> <p>Sie beheben Fehler in dem von ihnen erstellten Steuerungsprogramm.</p> <p>Die Schülerinnen und Schüler verwenden im Arbeitsprozess Fachsprache und Fachtermini, auch in englischer Sprache.</p>
<p><b>Ausgewählte Teilkompetenzen der Lernsituation</b></p> <p><i>Die Schülerinnen und Schüler ...</i></p> <ul style="list-style-type: none"> <li>➤ erfassen und analysieren Steuerungsabläufe</li> <li>➤ konfigurieren und parametrieren die notwendigen Hard- und Softwarekomponenten</li> <li>➤ verschiedene Steuerungen in Betrieb nehmen</li> <li>➤ Steuerungsprogramme erstellen und testen</li> </ul>
<b>Geschätzter Zeitumfang:</b> 12 x 45 Minuten

## Lernsituation

In der Fertigungsumgebung eines Herstellers für Smartphones sollen diverse Steuerungen eingesetzt werden, um Fertigungsprozesse zu steuern. Im Zentrum dieser Lernsituation steht der Warentransport über Förderbänder, welcher an unterschiedlichen Stellen in der Anlage geplant ist.



Aufgabe ist es die bereits installierte Hardware in einem Automatisierungssystem so zu konfigurieren und programmieren, dass der gewünschte Warentransport von Ausgangs- zu Endpunkt reibungslos funktioniert.

## TECHNISCHE ÜBERLEGUNGEN

Das in dieser Fortbildung gewählte Lernszenario umfasst ein Transportband, welches bereits an vielen Schulen vorhanden ist. Dieses sollte mit Sensoren ausgestattet sein, um die Position des Werkstücks am Bandanfang bzw. -ende zu erfassen.<sup>2</sup>

Für die hier abgebildeten Übungen werden die Ausstattungsanforderungen detailliert unter der entsprechenden Rubrik aufgeführt. Als Projektierungsumgebung wird die Software „TIA-Portal“ der Firma Siemens gewählt, weil hier die Industrierelevanz gegeben ist und entsprechend häufig die Ausstattung in beruflichen Bildungseinrichtungen vorzufinden ist.

Um Inhalte der Wirtschaft 4.0 abzubilden, werden in den Handreichungen des Staatsinstituts für Schulqualität und Bildungsforschung München (ISB) vier Ausbaustufen dieses Lernszenarios empfohlen. Diese Stufen zeigen auf, wie es möglich ist, dieses System an unterschiedliche Anforderungsniveaus anzupassen. Sie finden die Handreichungen unter folgendem Link:



[https://www.isb.bayern.de/download/20645/wirtschaft\\_4.0\\_an\\_beruflichen\\_schulen.pdf](https://www.isb.bayern.de/download/20645/wirtschaft_4.0_an_beruflichen_schulen.pdf)

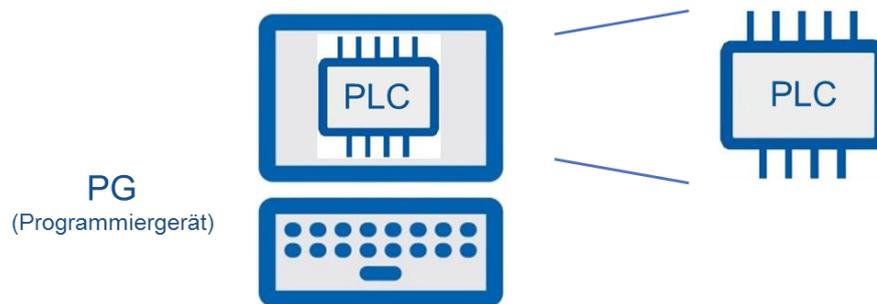
---

<sup>2</sup> vgl. ISB: Handreichung Wirtschaft 4.0 an beruflichen Schulen, S. 104

## LAB 01 – PROJEKT ANLEGEN UND HARDWARE EINFÜGEN

### Szenario

In einer Automatisierungssoftware wird ein Projekt angelegt. Die vorhandene Hardware wird in das Projekt eingefügt.



### Vorbereitung

- SPS Hardware
- PC mit Automatisierungssoftware

### Aufgaben

1. Legen Sie ein neues Projekt mit dem Namen „Tippbetrieb Bandmotor“ an.
2. Fügen Sie die Ihnen vorliegende Hardware in Ihr Projekt ein. Die Vorgehensweise wird im folgenden Video erklärt:



“Geräte in TIA-Portal V13 hinzufügen“

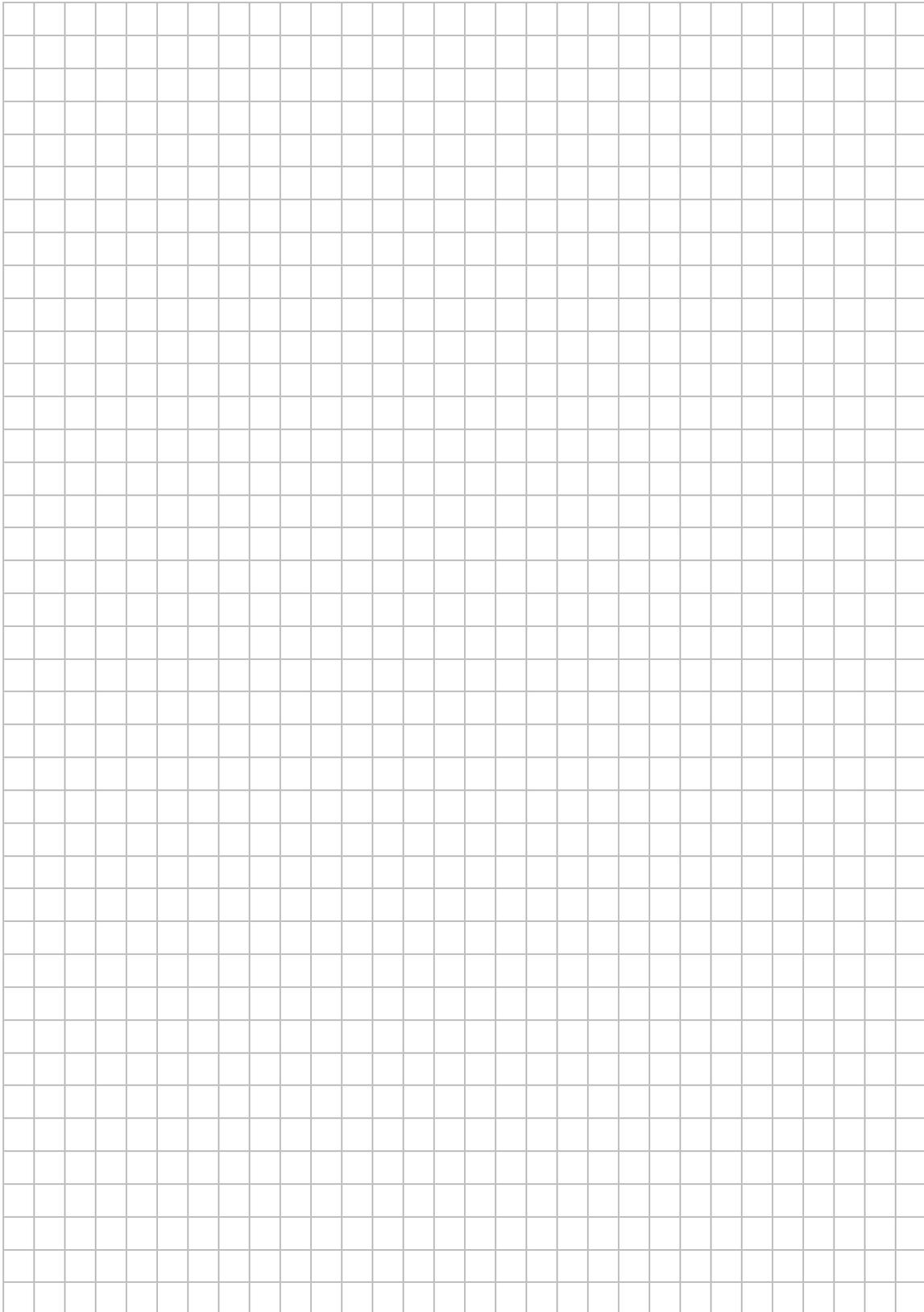
<https://www.youtube.com/watch?v=7S-YYDU6Pr0>



### Hinweise

Auf korrekte Bestellnummer und Anordnung der Module achten.

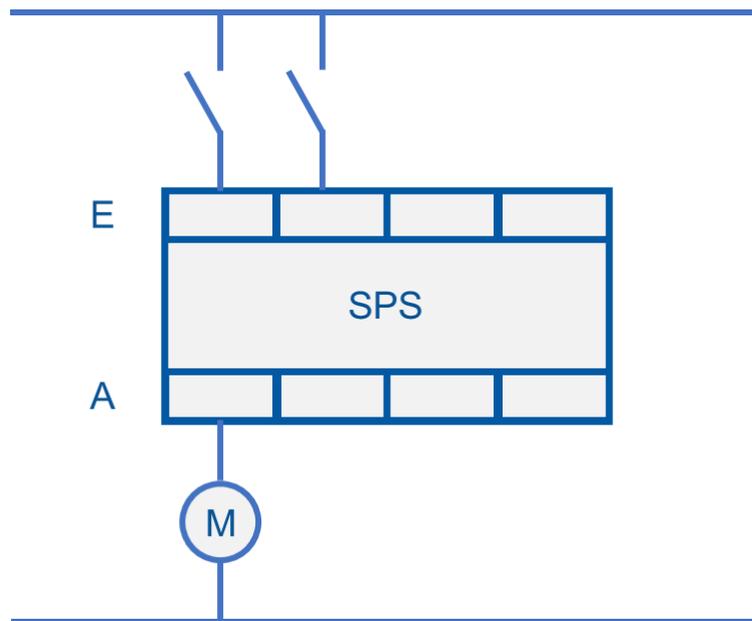
**Notizen**



## LAB 02 – ADRESSIERUNG UND VARIABLENTABELLE

### Szenario

In einer Automatisierungssoftware werden Adressbereiche für Ein- und Ausgabebaugruppen festgelegt. Darauf abgestimmt soll eine Variablen-tabelle angelegt werden.

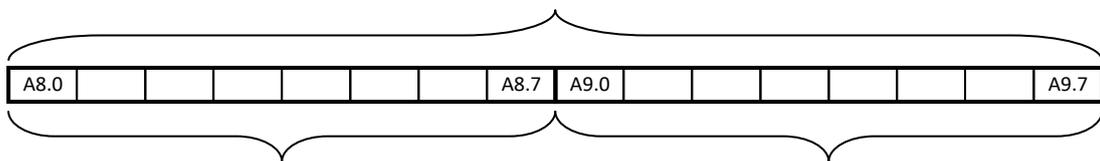
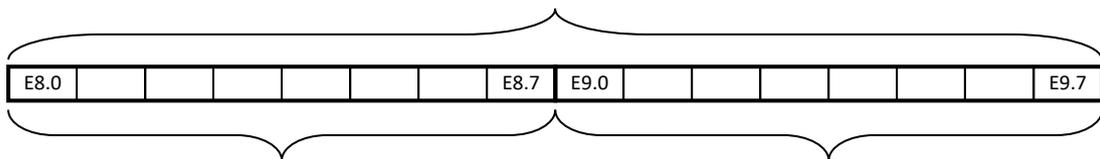


### Vorbereitung

- SPS Hardware
- PC mit Automatisierungssoftware
- Information zur Adressierung von Baugruppen

## Aufgaben

- Bei modularen CPUs gibt es die Möglichkeit der steckplatzorientierten Adressierung oder der freien Adressierung.  
Tragen Sie nachfolgend die fehlenden Bezeichnungen in die Felder ein.



- Bestimmen Sie die mit Hilfe der Automatisierungssoftware die Adressbereiche der Ein- und Ausgänge und tragen Sie diese in die folgende Tabelle ein.

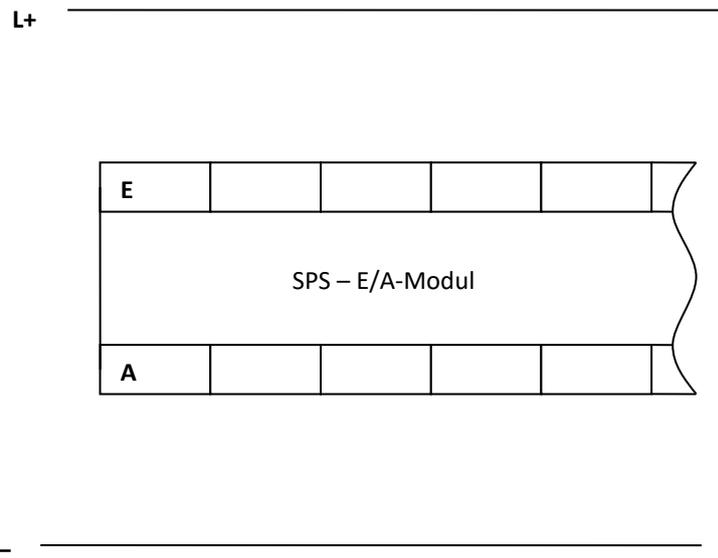
Modulname	Steckplatz	verwendete Eingangsbytes	verwendete Ausgangsbytes
	1		
	2		
	3		
	4		
	5		

- Verwenden Sie für die digitalen Ein- und Ausgänge den Adressbereich ab Byte 10.

4. Legen Sie nun eine Variablen-tabelle nach der folgenden Vorlage an.

Name	Datentyp	Adresse	Kommentar
S4_Motor_Ein	Bool	%E10.0	Taster Bandmotor einschalten (NO)
S1_Handbetrieb_Ein	Bool	%E10.1	Knebelschalter Handbetrieb (NO)
M1_Motor_Rechtslauf	Bool	%A10.0	Bandmotor Rechtslauf

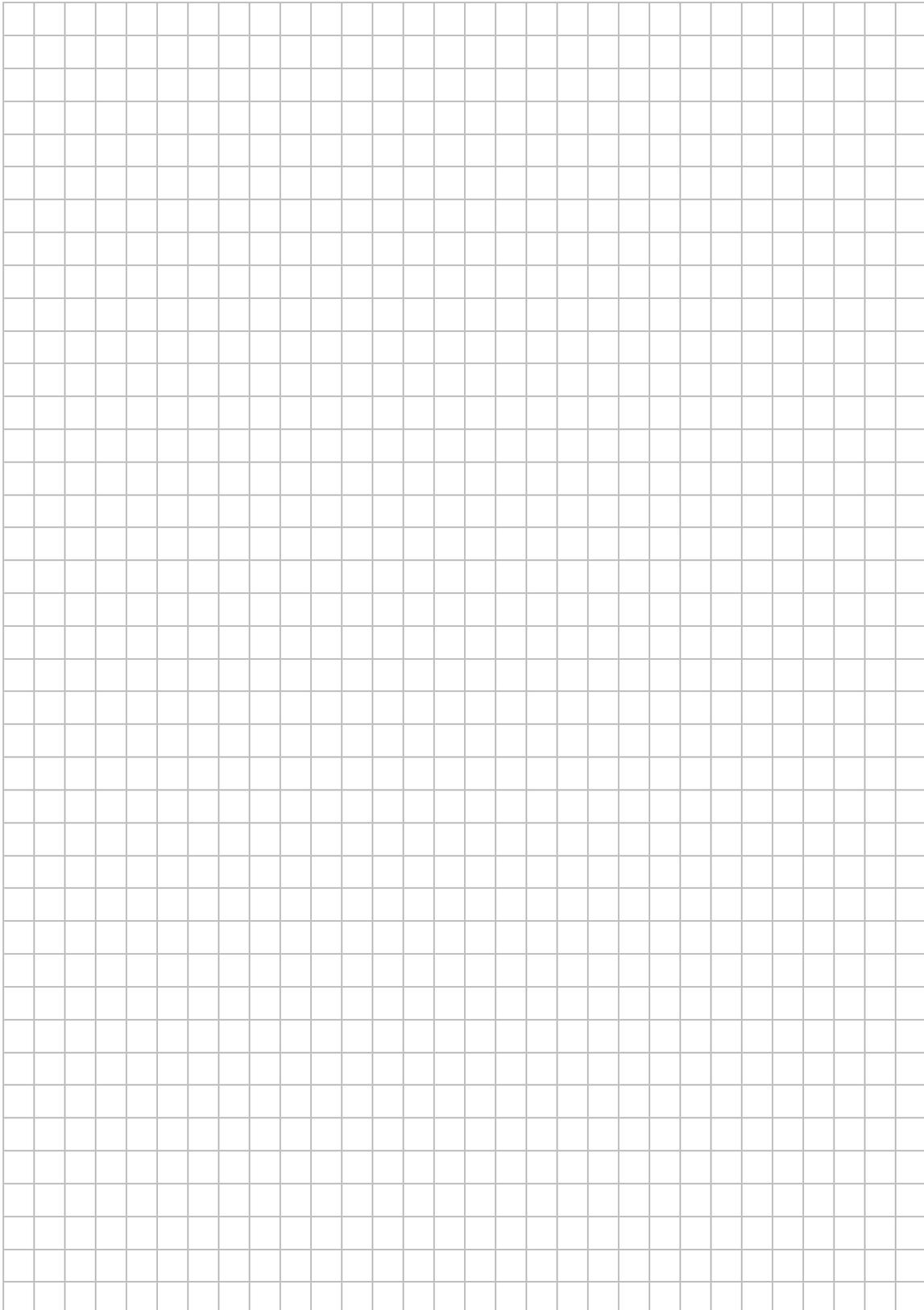
5. Ergänzen Sie den unten abgebildeten Klemmenanschlussplan.



## Hinweise

Verwenden Sie immer aussagekräftige Kommentare.

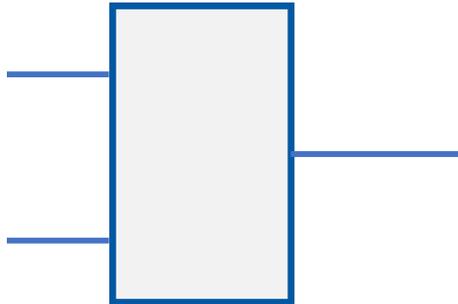
**Notizen**



## LAB 03 – PROGRAMMIERUNG UND SIMULATION

### Szenario

Der Bandmotor soll dann laufen, wenn der Knebelschalter sich auf der Schalterstellung „Hand“ befindet und der „Taster EIN“ gedrückt wird.



### Vorbereitung

- PC mit Automatisierungssoftware

### Aufgaben

1. Bearbeiten Sie das Arbeitsblatt „Programmiersprachen“ auf der nächsten Seite.
2. Erstellen Sie Ihr Programm im Baustein „Main“ in der Darstellungsart FBS.
3. Testen Sie die richtige Funktionsweise Ihres Programms mit Hilfe der Soft-SPS „PLCSIM“

### Zusatzinformationen



“Anwenderprogramm mit PLCSIM testen“

<https://www.youtube.com/watch?v=-5FYMLBG8PE>



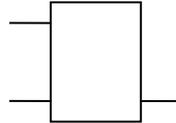
## Arbeitsblatt: Programmiersprachen

Das Transportband soll dann laufen, wenn der Knebelschalter (S4) sich auf der Schalterstellung „Hand“ befindet und die Taste Start (S1) gedrückt wird. -> Tipbetrieb

**Stromlaufplan:**

---

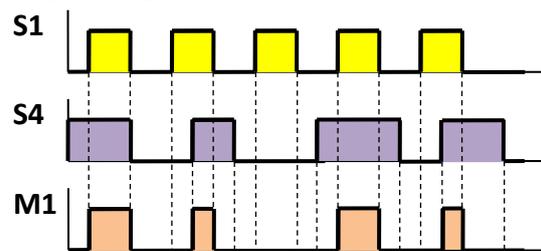
**Funktionsplan:**



**Funktionsgleichung:**

**Anweisungsliste:**


**Impulsdiagramm:**



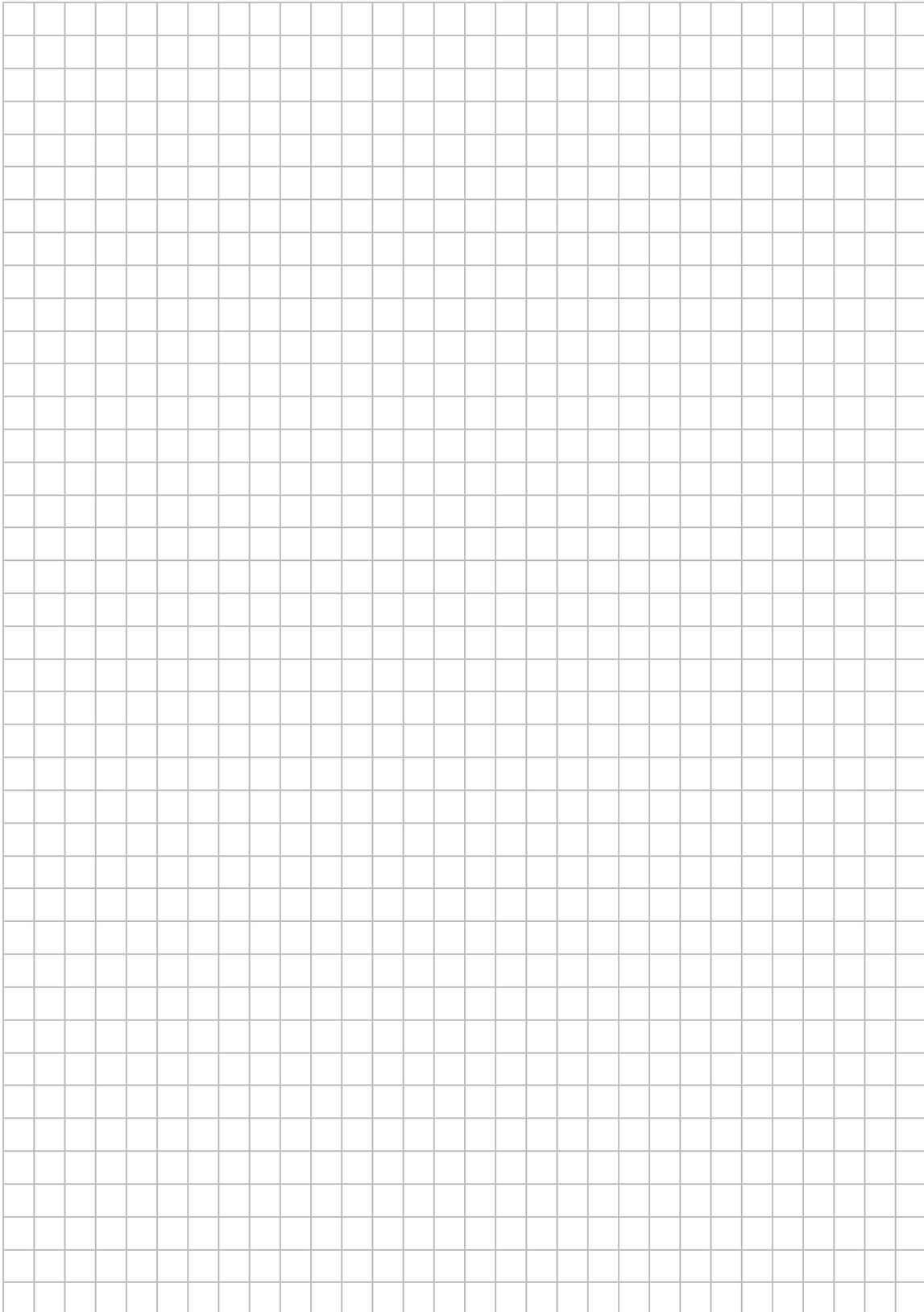
**Kontaktplan:**



**Wahrheitstabelle:**


**UND-Verknüpfung**  
 ⇨ „**Serielle Anordnung**“ der Symbole

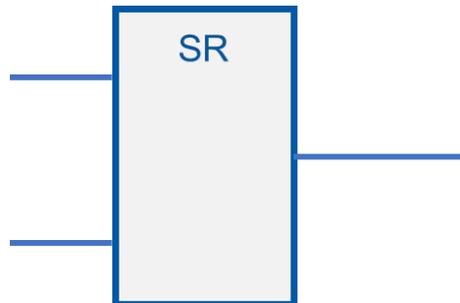
**Notizen**



## LAB 04 – ERWEITERUNG DES PROGRAMMS

### Szenario

Durch Betätigung eines Tasters soll der Bandmotor dauerhaft laufen.



### Aufgaben

1. Verändern Sie das bestehende Programm wie folgt. Durch Betätigung des Tasters S4 (S4\_Motor\_Ein) wird der Motor dauerhaft eingeschaltet. Eine Betätigung des Taster S5 (S5\_Motor\_Aus) schaltet den Motor aus. Damit der Bandmotor läuft, muss sich der Knebelschalter S1 (S1\_Handbetrieb\_Ein) in Schalterstellung "Hand" befinden.

S5 bei dieser Aufgabe als Schließerkontakt (NO) ausführen.



“Verwendung von Speicherbausteinen“

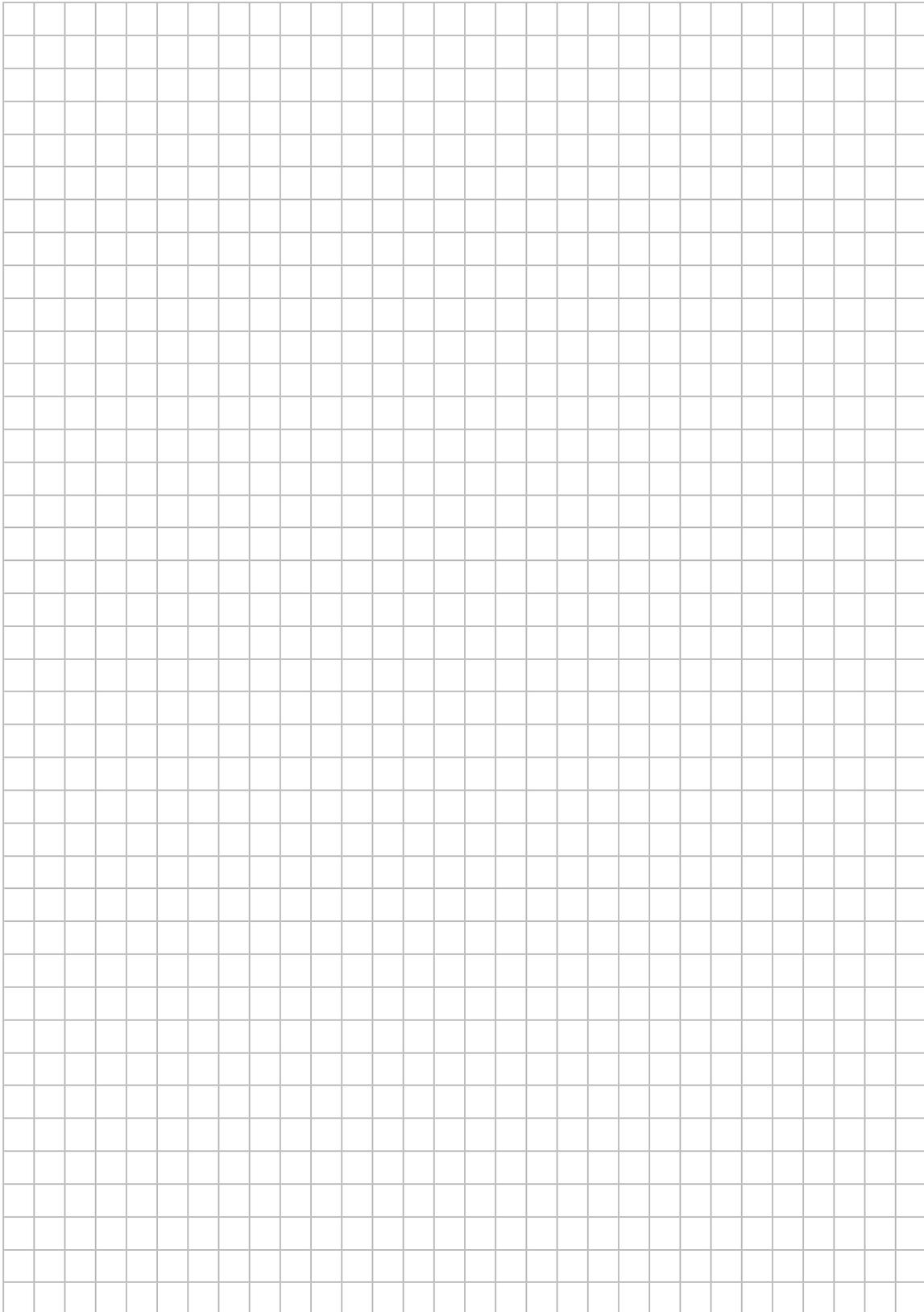
<https://www.youtube.com/watch?v=OpIKpVbY0kk>



2. Testen Sie das Programm mit Hilfe von PLCSIM.

### Hinweise

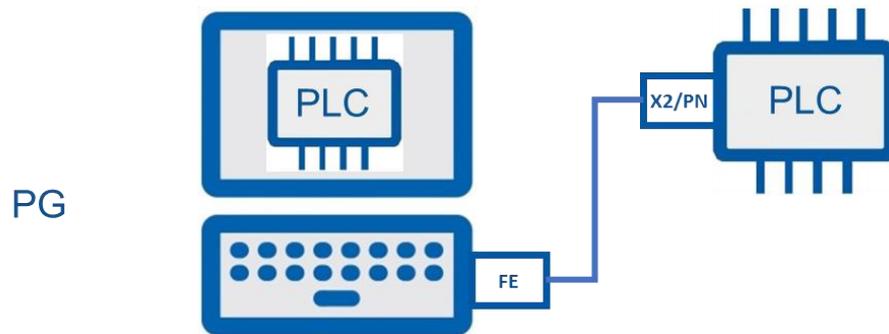
**Notizen**



## LAB 05 – PG MIT PLC VERNETZEN UND ÜBERTRAGEN

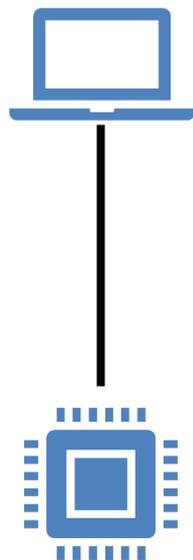
### Szenario

Das Programmiergerät wird über Profinet mit der PLC vernetzt. Anschließend wird das Anwenderprogramm übertragen und getestet.



### Aufgaben

1. Ermitteln Sie die verwendete IPv4-Adresse und Subnetzmaske Ihres Rechners und tragen Sie diese in die unten abgebildete Grafik ein.



#### PC/PG

IP-Adresse: \_\_\_\_\_

Subnetzmaske: \_\_\_\_\_

#### SPS/PLC

IP-Adresse: \_\_\_\_\_

Subnetzmaske: \_\_\_\_\_

2. Welcher Adressbereich steht für die SPS zur Verfügung? Nutzen Sie hierzu die Unterstützung eines „IP-Rechners“ im Internet.

Host<sub>min</sub>: \_\_\_\_\_

Host<sub>max</sub>: \_\_\_\_\_

3. Bestimmen Sie eine gültige IP-Adresse für die PLC und tragen Sie diese in die Grafik aus Aufgabe 1 ein. Führen Sie die Konfiguration für die PLC anschließend im TIA-Portal durch.



“Datenübertragung vom TIA-Portal auf SPS“

<https://www.youtube.com/watch?v=ml6gWBKQ2p4>



4. Verdrahten Sie Ihre PLC mit den Sensoren/Aktoren. Übertragen Sie Ihr Programm auf die CPU und testen Sie es anschließend.
5. Simulieren Sie einen Drahtbruch in der Tasterleitung von S5, indem Sie den Stecker vom entsprechenden Eingang der SPS entfernen. Welches Problem können Sie nun feststellen?
6. Wie kann das Problem aus Aufgabe 5 behoben werden?
7. Ändern Sie Ihr Programm gemäß Aufgabe 6 und testen Sie anschließend, ob nun die Drahtbruchsicherheit gewährleistet ist.

## Hinweise

Grundlagen zum Aufbau einer IPv4-Adresse



“IPv4-Adressen“

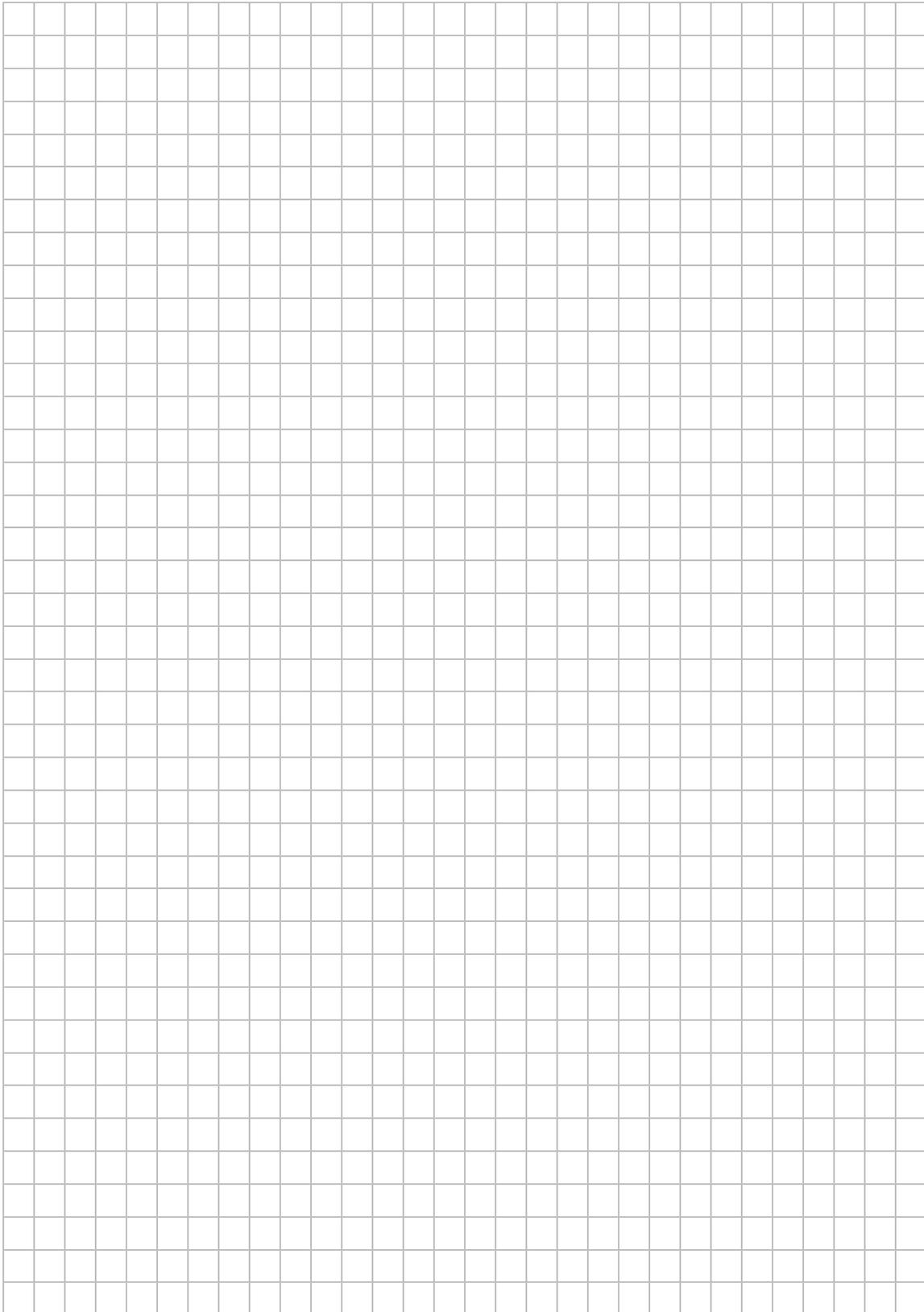
<https://www.youtube.com/watch?v=ZfOULSYpL8s>



Drahtbruchsicherheit:

Für Einschaltbefehle werden Schließerkontakte (NO) und für Ausschaltbefehle Öffnerkontakte (NC) verwendet!

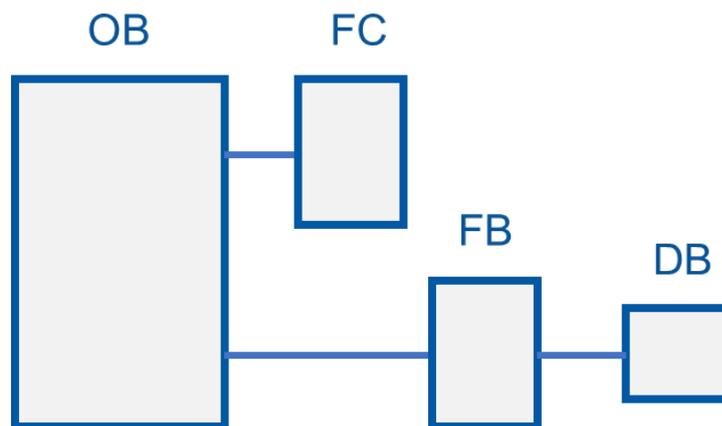
**Notizen**



## LAB 06 – PROGRAMMCODE ORGANISIEREN

### Szenario

Wir erhöhen die Übersichtlichkeit durch Strukturierung des Programms in Bausteinen.



### Vorbereitung

- PC mit Automatisierungssoftware

### Aufgaben

1. Beantworten Sie mit Hilfe des Infovideos „Arbeitsweise einer SPS“ die nachfolgenden Fragen.



„Arbeitsweise einer SPS“

<https://www.youtube.com/watch?v=QgBuG3SUvbE>



- 1.1 Nennen Sie zwei grundsätzliche Arbeitsweisen einer SPS.

1.2 Entscheiden Sie, von welchem Typ der Baustein „Main“ im TIA-Portal ist!

- Funktionsbaustein
- Datenbaustein
- Organisationsbaustein
- Funktion

1.3 Woran erkennen Sie, ob eine SPS-Applikation in „linearer“ oder „strukturierter“ Programmierung vorliegt?

1.4 Nennen sie zwei Vorteile der „strukturierten Programmierung“ gegenüber der „linearen Programmierung“.

1.5 Erklären Sie, warum der Baustein „Main“ in einer TIA-Applikation sehr wichtig ist und nicht gelöscht werden darf.

2. Legen Sie einen Funktionsbaustein (FB) mit dem Namen „Ansteuerung Motor Automatik“ und bearbeiten Sie anschließend folgende Schritte:

- Deklarieren Sie die Bausteinschnittstelle
- Programmcode aus OB1 ausschneiden und in FB einfügen
- Ein- und Ausgangsbelegung anpassen
- FB im Baustein OB1 aufrufen und anschließen
- Programm auf CPU übertragen und testen

## Zusatzinformationen



“Aufruf eines FB“

<https://www.youtube.com/watch?v=Sen401mLbbY>

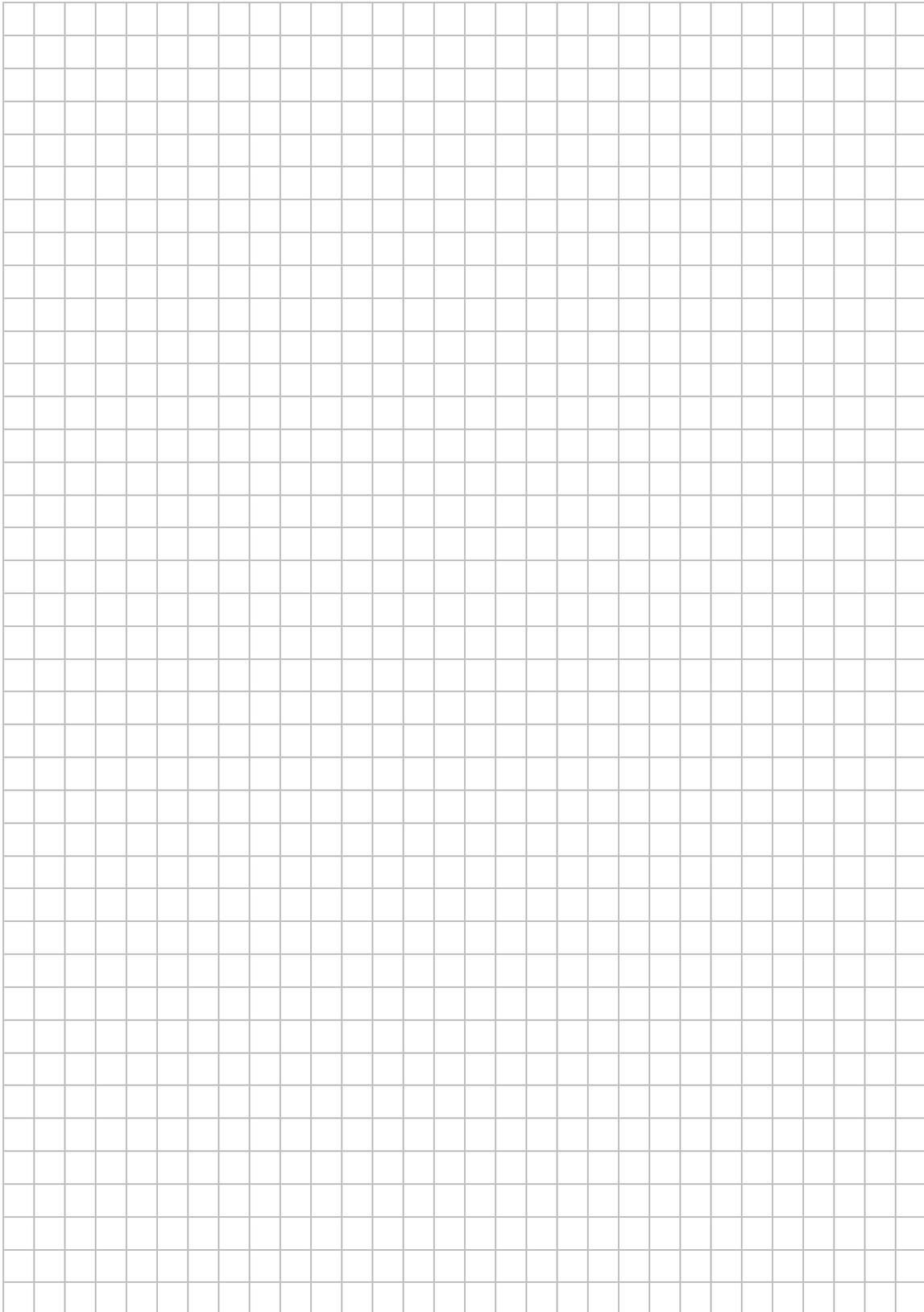


“Vergleich FC mit FB“

[https://www.youtube.com/watch?v=hFuE32\\_uHRI&t=26s](https://www.youtube.com/watch?v=hFuE32_uHRI&t=26s)



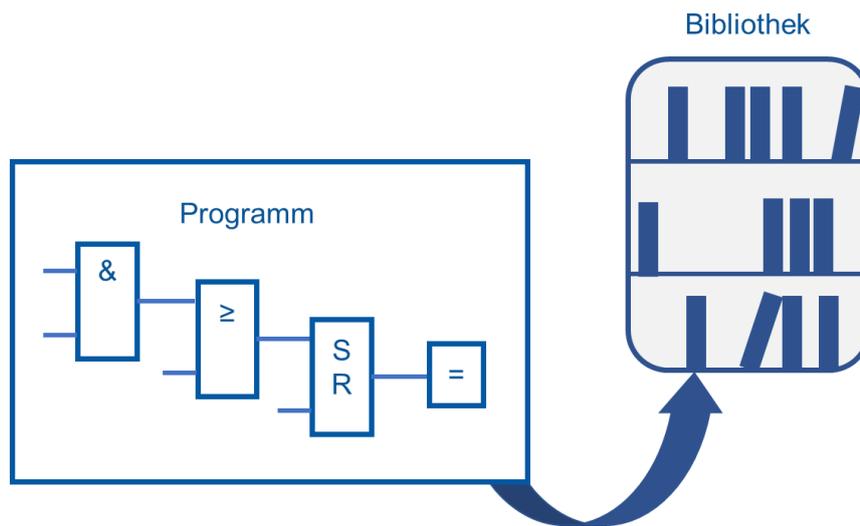
**Notizen**



## LAB 07 – PROGRAMM IN BIBLIOTHEK ABLEGEN

### Szenario

Das Programm wird in einer Bibliothek abgelegt, dadurch kann dieser Code in anderen Projekten wiederverwendet werden.



### Vorbereitung

- PC mit Automatisierungssoftware

### Aufgaben

1. Sehen Sie das Informationsvideo „Arbeiten mit Bibliotheken“ an. Legen Sie anschließend den FB „Ansteuerung Motor Automatik“ als Globale Bibliothek an.

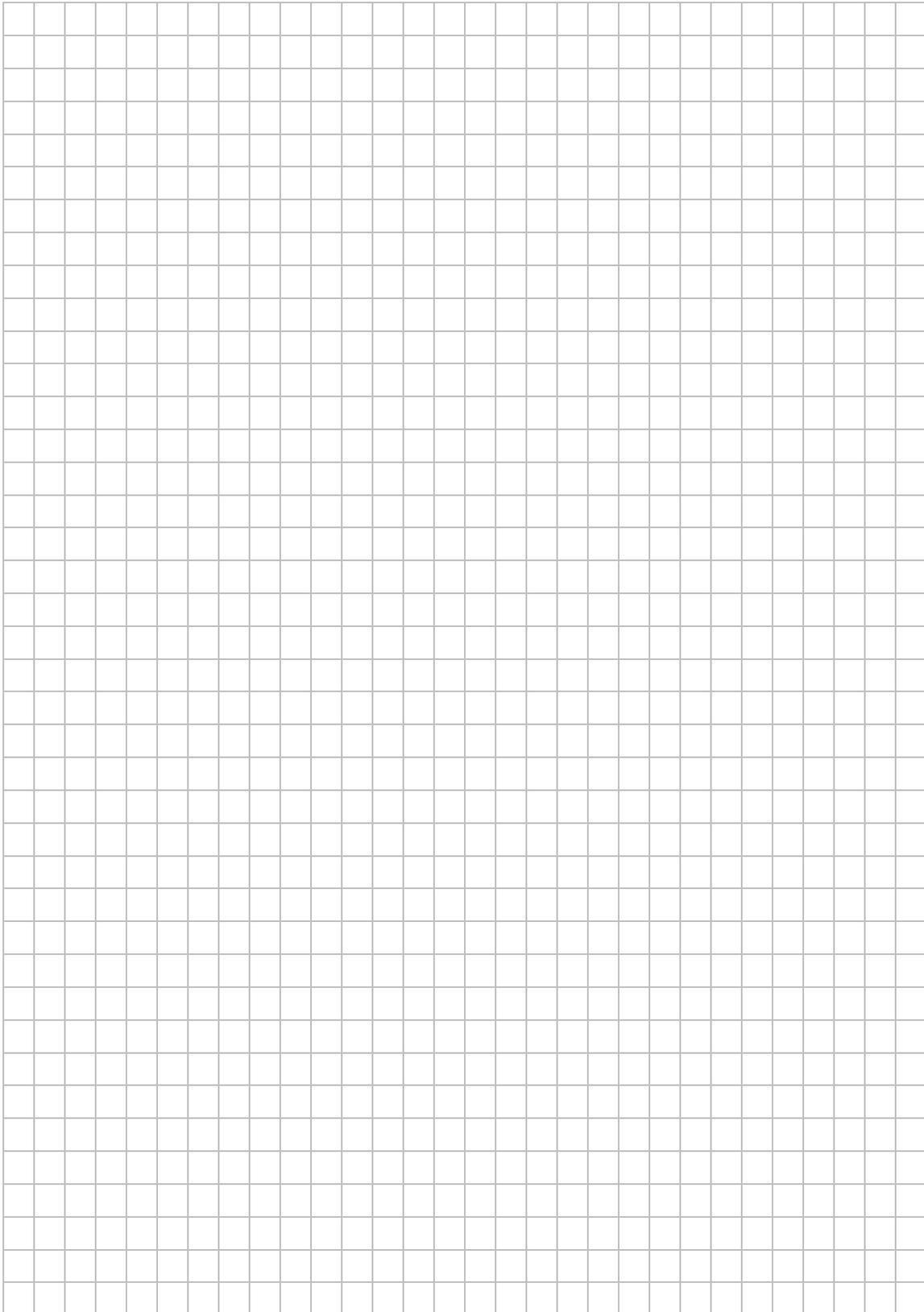


„Arbeiten mit Bibliotheken“

<https://www.youtube.com/watch?v=uFqPbqqMCBE>



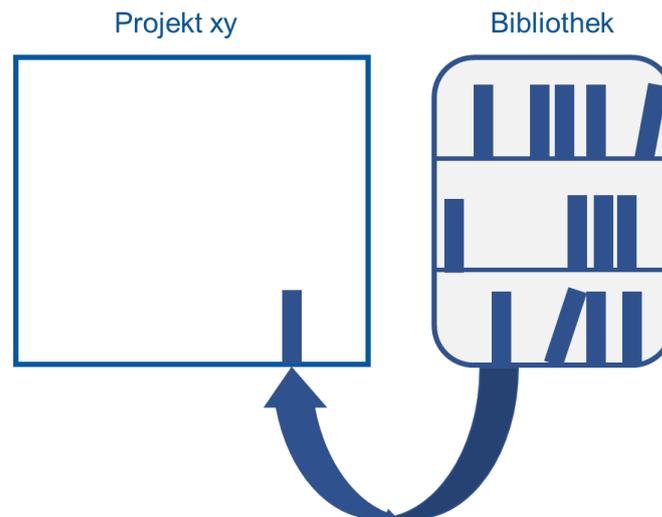
**Notizen**



## LAB 08 – BIBLIOTHEKEN IN ANDEREN PROJEKTEN NUTZEN

### Szenario

Die in LAB 07 erstellte Bibliothek wird in einem anderen Projekt mit anderer Hardwareausstattung wiederverwendet.



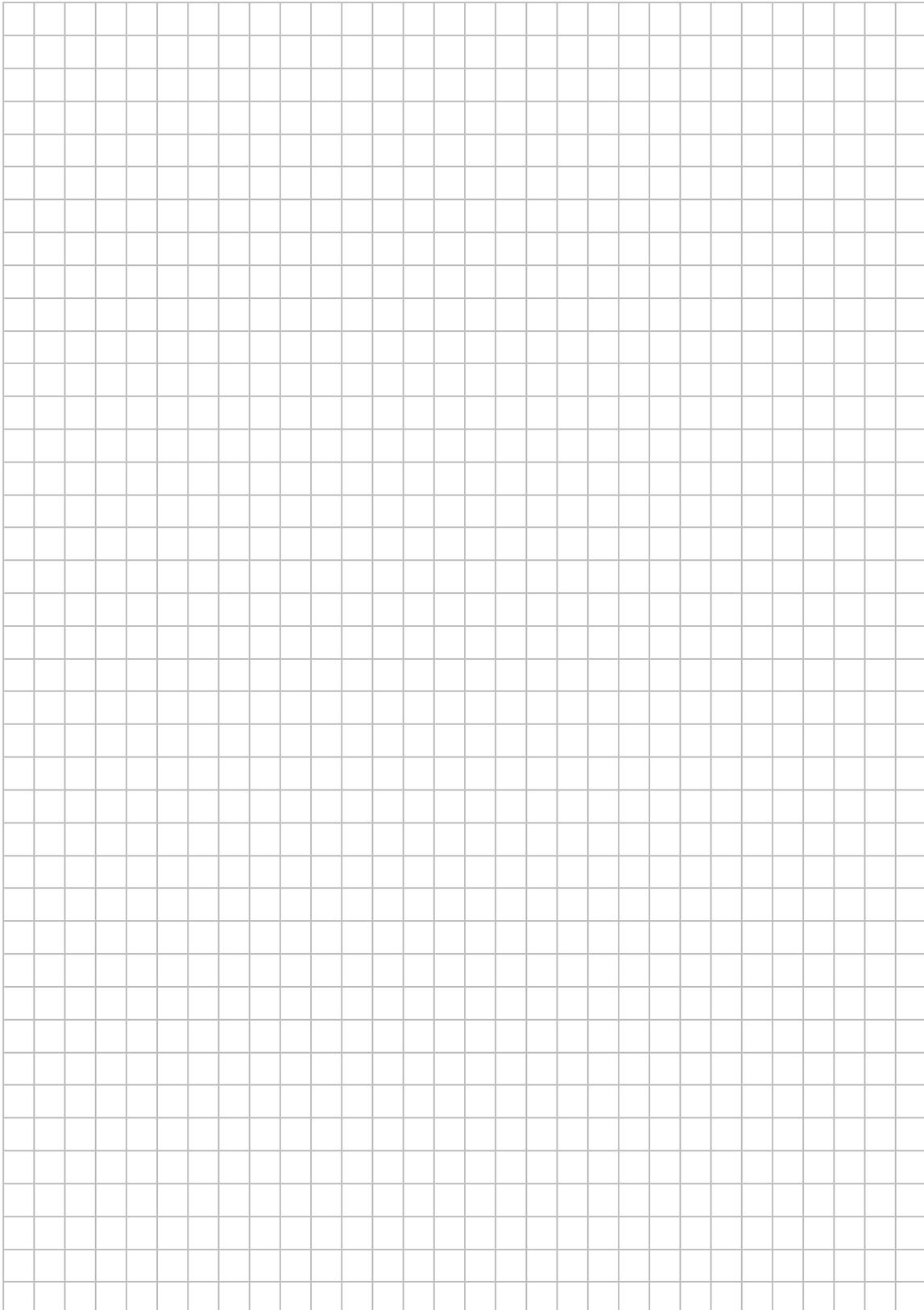
### Vorbereitung

- PC mit Automatisierungssoftware
- PLC mit Verbindung zum PC

### Aufgaben

1. Legen Sie ein neues Projekt mit Namen „Bandmotor S7-1500 an“.
2. Fügen Sie die Ihnen vorliegende Hardware in Ihr Projekt ein. Nutzen Sie hierzu die automatische Hardwareerkennung (-> Hilfe -> Suchen -> „nicht spezifizierte CPU anlegen“)
3. Laden Sie den vorher erstellten Funktionsbaustein „Ansteuerung Motor Automatik“ aus der Bibliothek in Ihr Projekt.
4. Führen Sie notwendige Anpassungen in Ihrem Projekt durch.
5. Testen Sie anschließend Ihr Programm mit PLCSIM.
6. Laden Sie das Programm in die Steuerung. Führen Sie einen Funktionstest durch.

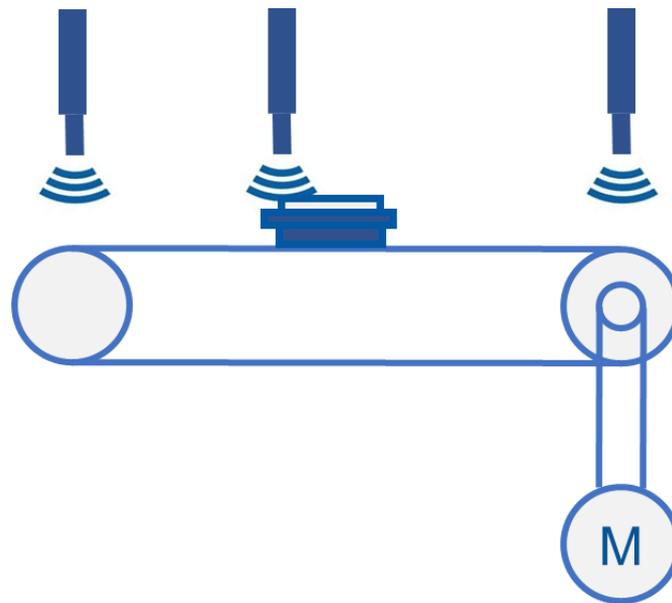
**Notizen**



## LAB 09 – PROGRAMMIEREN UND INBETRIEBNAHME EINES MECHATRONISCHEN SYSTEMS

### Szenario

Ziel ist es das Förderband als mechatronisches Teilsystem mit Hilfe des vorher erstellten Funktionsbausteins „Ansteuerung Motor Automatik“ zu bewegen. An jeder Anlage befindet sich ein Bedienpult mit den Tastern/Schaltern.



### Vorbereitung

- PC mit Automatisierungssoftware
- PLC mit Verbindung zum PC
- Band mit Antrieb
- Bedienfeld mit Tastern und Leuchtmeldern

### Aufgaben

1. Das Transportband läuft von links nach rechts, wenn der Taster „S4\_Motor\_EIN“ gedrückt wird und der Werkstückträger (WT) sich am Bandanfang links befindet. Das Band läuft bis der „Sensor\_Band\_Rechts“ anspricht. Nachdem der WT abgenommen wird und der „Sensor\_Band\_Rechts“ nicht mehr aktiv ist, kann ein neuer Vorgang begonnen werden. Solange das Band läuft, leuchtet der Leuchtmelder\_Band\_EIN.

- 1.1. Verwenden Sie das vorher erstellte Projekt „Bandmotor S7-1500“.
- 1.2. Nur am Standort Kempten:

Fügen Sie die ET200 aus der globalen Bibliothek ein und vernetzen Sie diese mit der PLC. Hinweis: Dies wird vom Lehrgangsteam gezeigt.

Nur am Standort Passau:

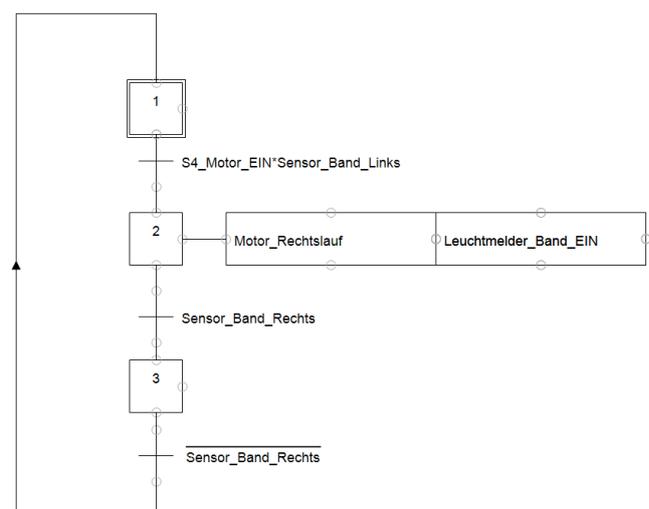
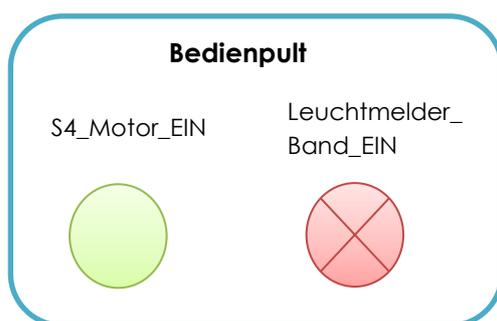
Verbindung von der SPS zum Transportband über die SUB-D Steckerverbindung herstellen.

- 1.3. Passen Sie die Variablen-tabelle und die Adressbelegungen an.
- 1.4. Erstellen Sie Ihr Programm in einem FB gemäß der vorgegebenen Aufgabenstellung.
- 1.5. Übertragen Sie das Programm auf die PLC und testen Sie es anschließend.

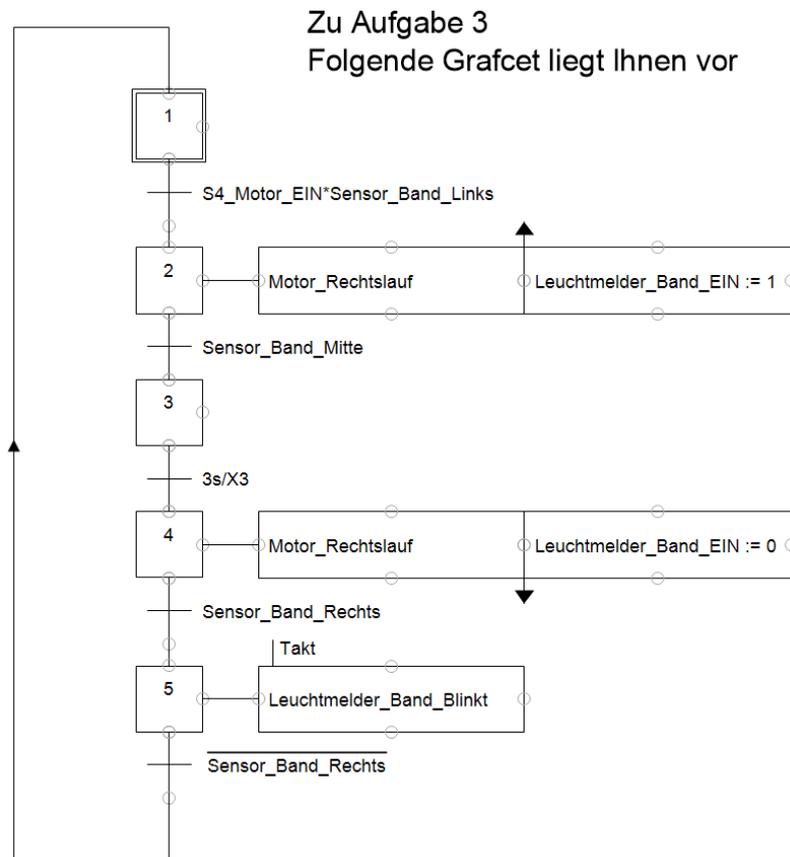
## Optionale Aufgaben

2. Nun soll die vorherige Aufgabenstellung als Schrittkette programmiert werden. Als Hilfestellung dient Ihnen die nebenstehende Grafcet.

Zu Aufgabe 2  
Folgende Grafcet liegt Ihnen vor

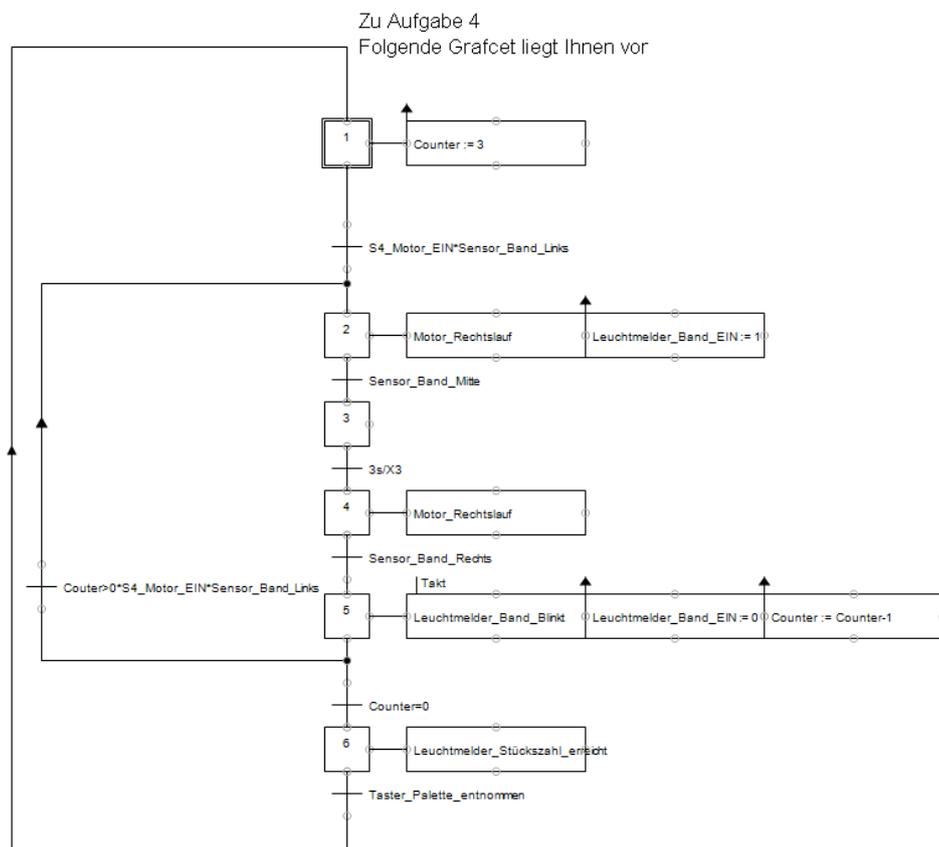
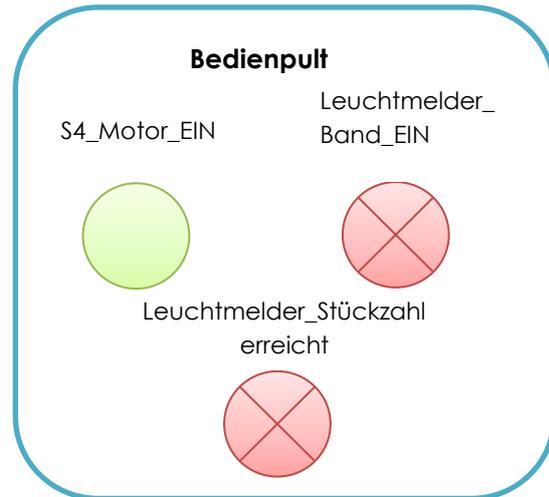


3. Start und Ende wie bei der Aufgabe 2. Das Transportband läuft bis zu dem „Sensor\_Band\_Mitte“ und hält dann 3 Sekunden an. Danach wird der WT automatisch nach rechts transportiert bis er den „Sensor\_Band\_Rechts“ erreicht hat. Weiterhin gilt: Solange das Band läuft, leuchtet der Leuchtmelder\_band\_EIN dieser blinkt beim Erreichen des „Sensor\_Band\_Rechts“.



**Hinweis:** Nutzen Sie die Hilfefunktion um sich über das Thema Taktmerker zu informieren.

4. Situation wie bei der Aufgabe 3. Das Procedere soll allerdings nur 3 Mal durchlaufen werden. Danach soll das Band gestoppt, der Warenträger gegen einen neuen ausgetauscht und die Produktion erneut gestartet werden. Solange das Band läuft, leuchtet der Leuchtmelder dauerhaft. Wenn das Procedere 3 Mal durchlaufen ist, soll der Leuchtmelder dies mit einem Blinken (hier mit einer Frequenz von 2 Hz) signalisieren. Somit kann der zuständige Mitarbeiter den Behälter ausleeren, damit weiter produziert werden kann.



## Hinweise



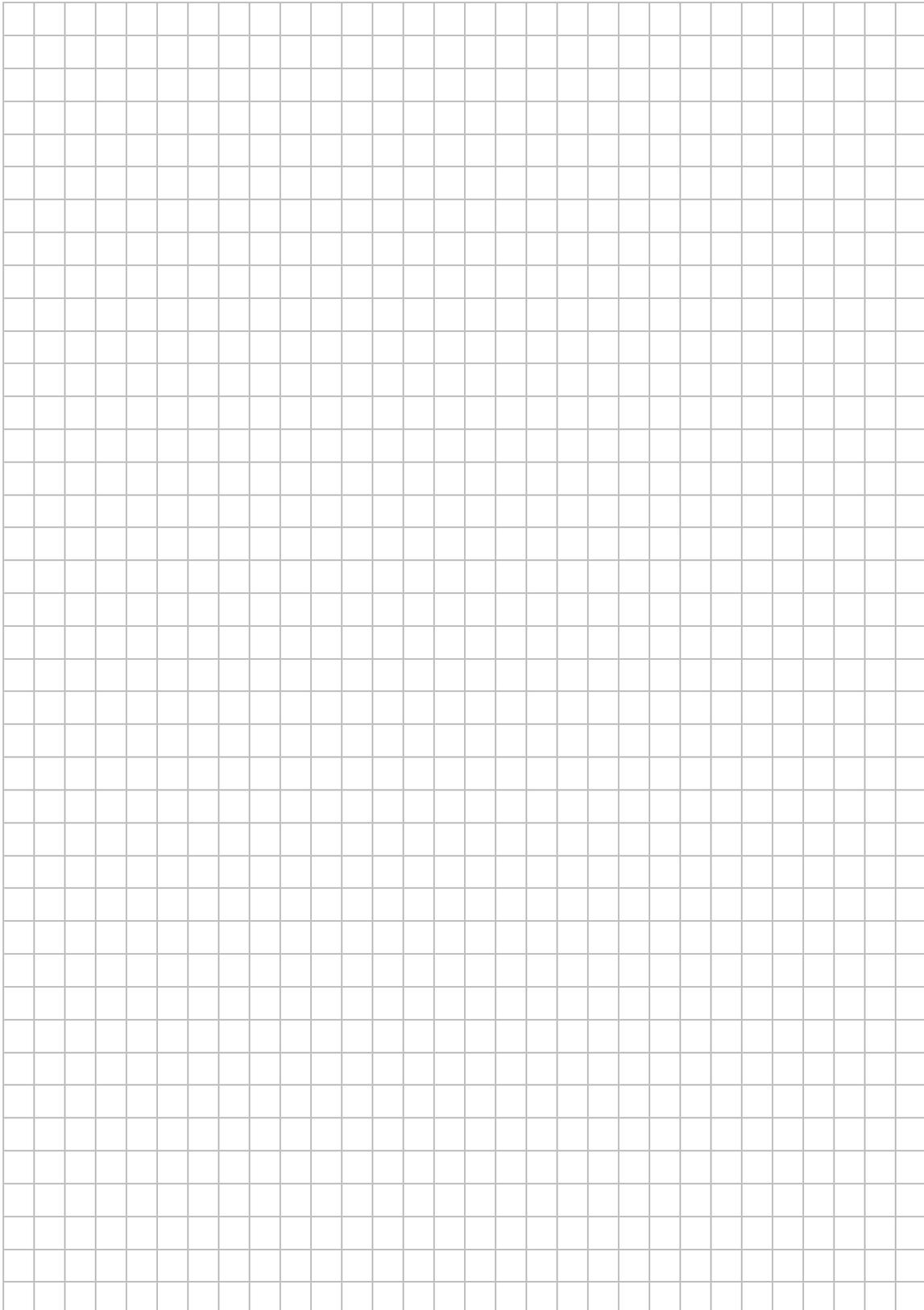
“IEC-Zähler“ <https://www.youtube.com/watch?v=T3GZmfssMQ>



**Weitere optionale Aufgaben:**

1. Mit dem rastenden Wahlschalter -SF3 kann zwischen Hand- und Automatikbetrieb umgeschaltet werden. Im Handbetrieb läuft das Band im Tippmodus. Im Automatikbetrieb wie in Aufgabe 3.
2. Bei Betätigung des Not-Halt Schlag-Tasters (-SF5) stoppt die Bewegung sofort. Erst nach zurücknehmen des Not-Aus und Quittierung mittels Reset-Taster (-SF4) läuft das Band wieder an.

**Notizen**



## OBJEKT- UND VARIABLENNENENNUNG

### Verwendung von Präfixen

Zur besseren Lesbarkeit von SPS-Programmen bestehen häufig Richtlinien (z.B. firmenintern, Liefervorschrift, ...), die die Namensgebung von Variablen und viele andere Dinge, die die SPS-Programmierung betreffen, vereinheitlichen.

### 1. Bausteinnamen

**Bausteinnamen** sind so kurz wie möglich und so aussagekräftig wie nötig zu halten. Die Bezeichner von Bausteinnamen beginnen mit einem Bausteinpräfix, gefolgt von einem Großbuchstaben und stehen im CamelCasing.

**Datenbausteine** (global-, type-, array-DB) sind das Präfix "db" voranzustellen.

Ebenso müssen die Namen von globalen Baustein-Instanzen mit dem Präfix "db" beginnen, gefolgt der zugehörigen FB-Bezeichnung und eventuell einer weiteren Kennung.

Präfix	Datentyp
<b>fc</b>	Funktion
<b>fb</b>	Funktionsbaustein
<b>db</b>	Datenbaustein

**%DB1**  
"dbDriveFunction"

**%FB1**  
"fbDriveFunction"

Einzelinstanz: "db,, => **dbDriveFunction** // Instanz-DB des FB „**fbDriveFunction**“

### 2. Bausteinprogrammierung

Entsprechend der vorrangigen Funktionalität/Aufgabe des Bausteins ist es empfehlenswert auch dessen Bausteinsprache zu wählen. Folgende Umsetzung wird empfohlen.

Funktionalität / Aufgabe des Bausteins	Bausteinsprache
<b>Logik, Grundverknüpfungen</b>	FUP
<b>Bitoperationen</b>	AWL
<b>Algorithmik</b>	SCL
<b>Datenverwaltung</b>	SCL

Alle Bausteine sind als parametrierbare **bibliotheksfähige Bausteine** gemäß **IEC 61131-3** zu entwickeln. Damit darf innerhalb eines Bausteins auf keine globalen

Informationen zugegriffen und nur Formalparameter verwendet werden. Sämtliche Bausteine sind mittels der **Multiinstanztechnik** zu entwickeln.

### 3. Bezeichnung von Variablen

Variablen erhalten prinzipiell einen aussagekräftigen Namen. Des Weiteren führt jede Variable ein Präfix in Kleinbuchstaben, der den Datentyp der Variable beschreibt. Über die Schreibweise der Variablen wird zudem der Variablentyp definiert.

Präfix	Datentyp
<b>x</b>	Bool
<b>i</b>	Integer
<b>ui</b>	Unsigned Integer
<b>r</b>	Real
<b>di</b>	Double Integer
<b>udi</b>	Unsigned Double Integer
<b>tp</b>	Timerinstanz Impuls, TP
<b>ton</b>	Timerinstanz Einschaltverzögerung, TON
<b>tof</b>	Timerinstanz Ausschaltverzögerung, TOF
<b>t</b>	TIME
<b>ctu</b>	Zählerinstanz Aufwärtszähler, CTU
<b>ctd</b>	Zählerinstanz Abwärtszähler, CTD
<b>ctud</b>	Zählerinstanz Auf-/Abwärtszähler, CTUD
<b>b</b>	Datencontainer Byte
<b>w</b>	Datencontainer Word
<b>dw</b>	Datencontainer Double Word
<b>inst</b>	Multiinstanz eines Funktionsbausteins
<b>ty</b>	Variable vom Typ eines PLC-Datentyps oder einer lokalen Struktur
<b>a</b>	Array
<b>s</b>	Zeichenkette, strings
<b>c</b>	Character
<b>fc</b>	Funktion
<b>fb</b>	Funktionsbaustein
<b>db</b>	Datenbaustein

Variablenschreibweise entsprechend dem Variablentyp:

Variablentyp	Schreibweise	Beispiele
<b>Formalparameter IN, OUT, IN_OUT</b>	Präfix klein, der Rest in PascalCasing, Sign: #	xStartMotor
<b>Formalparameter STAT und TEMP</b>	Präfixe klein, der Rest in PascalCasing, Sign: #	xstatMotorRechtslauf
<b>Globale PLC- Variablen</b>	Präfix klein, Betriebsmittelkennzeichen in Großbuchstaben, der Rest in UnderlineCase Sign: ""	xSF1_Start_Motor
<b>Konstanten</b>	Präfix klein, der Rest in Großbuchstaben	rPI

## 4. Bezeichnung von Arrays

Arrays sind im Plural zu bezeichnen. Der zu definierende Array-Index beginnt ausschließlich bei Index 0 und muss mit einer Konstante für die Obergrenze enden.

Beispiel:           aTempValues [0..10] of Real;

## 5. Vorschläge zur Namensgebung

Empfehlungen für Abkürzungen (nur eine pro Kennung)	
<b>Min / Max</b>	Minimum / Maximum
<b>Act</b>	Aktuell
<b>Next / Prev</b>	Nächster / vorheriger
<b>Avg</b>	Durchschnitt
<b>Diff / Sum</b>	Differenz / Summe
<b>Pos</b>	Position
<b>Ris / Fal</b>	Steigende / fallende Signalflanke
<b>Sim</b>	Simuliert
<b>Old</b>	alter Wert
<b>Dir</b>	Richtung
<b>Err</b>	Fehler
<b>Warn</b>	Warnung
<b>Cmd</b>	Befehl

## 6. Bezeichnung von PLC-Datentypen

Benutzerdefinierte PLC-Datentypen werden in Strukturen definiert. Die Bezeichnungen solcher Datentypen beginnen mit dem **Großbuchstaben T**, gefolgt von **PascalCasing**.

Beispiele: TMeldung, TNetConfig

## AUSSTATTUNG FÜR LABORÜBUNGEN

Zur Durchführung der Laborübungen wird neben den Computern und Notebooks der Schulen folgende Ausstattung von der Fachgruppe „Steuerung/Regelung“ empfohlen, damit ist u. a. die didaktische Eignung und Industrietauglichkeit gewährleistet. Bei Abweichungen von den Ausstattungsempfehlungen ist auf diese Kriterien zu achten, damit die beabsichtigten Intentionen der Laborübungen für Lehrerfortbildung und Unterricht erreicht werden.

Die Ausstattung ist ausgelegt für 12 Teilnehmer.

Nr.	Bezeichnung	Menge	Lab
1	Automatisierungsgerät S7-300 bzw. S7-1200	6	01 - 07
2	Bedienpult mit Tastern/Schaltern/Meldeleuchten	6	05 -09
3	Automatisierungsgerät S7 1500	6	08, 09
5	Förderbänder mit Gleichstromantrieb und Sensorik	6	08,09
6	Programmiersoftware Siemens TIA-Portal (ab V13) + PLCSIM	12	01-09
7	PC mit folgenden Eigenschaften <ul style="list-style-type: none"> <li>• Internetzugang</li> <li>• zusätzliche Netzwerkkarte</li> <li>• Bildschirm mit 16:9 Format</li> </ul>	12	01-09
8	Zubehör (Patchkabel, Profinetleitungen, ...)	6	05-09

# WIRTSCHAFT 4.0 - FORTBILDUNGSMODULE

Akademie für  
Lehrerfortbildung  
und Personalführung

## Digitale Transformation – Wirtschaft 4.0 – Fortbildungsmodule der ALP Dillingen

