

Teil 1 , Lösungsvorschlag Teil ohne Unterlagen

Name, Klasse :

Bei Ankreuz-Fragen können Sie, sollten Sie Probleme mit der Formulierung einer Frage haben, auch erklärenden Text anfügen. Es zählt ihr Verständnis !

1. Ein zentrales Thema der modernen Automatisierungstechnik ist die **Modularisierung**, die von der Anlagenstruktur bis hinunter zur SPS-Programmierung angestrebt wird.

Erläutern Sie hier die aus Ihrer Sicht wesentlichen Aspekte dieser Struktur :

**Modularisierung bringt kleinere, überschaubare Strukturen.
Besser zu planen, zu realisieren, zu warten und zu modifizieren.**

Wichtig sind definierte Modulschnittstellen

Es entsteht Kommunikationsbedarf zwischen den Modulen

2. Welche Aussagen sind richtig ?


- Handshakeverfahren benötigen Kommunikation in zwei Richtungen (Ein-und Ausgänge)
- Layer 1 von Ethernet ist identisch mit der Spezifikation RS485
- Profibus DP – Slaves kommunizieren, wenn sie in Token-Besitz sind
- IP-Adressen und Tokenadressen sind routingfähig
- 62.245.200.128 / 28 ist eine gültige IP-Adresse für eine SPS
- Statische Redundanz ist billiger als dynamische Redundanz
- Durch Nutzung des IP-Protokolls wird Ethernet in Zeitschlitzen deterministisch
- In der Master-SPS von Profibus DP werden die Slaves mittels IP-Adressen adressiert
- Ethernet-Kommunikation hat eine höhere Übertragungsrate als der ProfibusDP
- Profinet I/O verwendet in Layer 2 andere Adressenformate als Ethernet TCP/IP
- MAC-Adressen sind im Gegensatz zu Profibusadressen routingfähig
- Die aggregierte Bandbreite gibt die garantierte Kommunikationsleistung einer SPS an
- Mit Profinet IRT kann ein Windows-Rechner als deterministisches MES –System arbeiten
- In Layer 7 eines Feldbussystems befinden sich (falls vorhanden) Kommunikationsdienste
- In Layer 7 von Profibus DP befinden sich die Dienste L (=lade Byte) und T (=transfer Byte)
- In Layer 3 von Profibus DP sind die Funktionen zur Kommunikation mit anderen Mastern
- Layer 2 von Profibus DP ist ein Hybridsystem aus Token Passing und CSMA /CD
- Ethernet ist schneller als Profibus, weil die Framelänge (in Byte) kürzer ist

3. Welche Funktion hat der **Service Access Point (SAP)** in einem Kommunikationssystem ?

Er adressiert beim Übergang von Layer n-1 zu Layer n den in Layer n für das aktuelle Datenpaket richtigen Dienst

4. Hier ein Ausschnitt aus www.sps-forum.de :

Anonymous
Gast

 **Handshake-Automat**

Hi,

weiss jemand wie die Kommunikation zwischen einer zentralen SPS (übergeordnet) und paar untergeordneten SPS 'n aussieht ?
Habe nur gehört, dass ein Handshake-Ablauf erarbeitet werden muss ?
Es wird ein Profibus-DP benutzt.
Wo kann ich paar Beispiele finden oder Informationen finden ?

Danke

- erklären Sie ihm theoretisch, wie eine Handshake-Kommunikation im Kern funktioniert

- nennen Sie Vorteile des Handshake-Verfahrens

1) Handshake ist im Mittel schneller als ein zeitbasiertes Verfahren

2) Handshake zeigt, ob Partnergerät funktioniert

5. Was verstehen Sie unter dem Begriff **deterministische Kommunikation** ?

Der worst-case der Kommunikationszeit (genauer : Zeit bis zum Absetzen der Nachricht) ist berechenbar.

6. Erklären Sie den Begriff **Redundanz** im Umfeld der Anlagentechnik :

Mehrfaches Vorhandensein von Modulen oder Komponenten mit gleicher Funktion, um Ausfallsicherheit zu erreichen

7. Die erste Phase des zeitlichen Verlaufs der Unverfügbarkeit von technischen Systemen (sog. "Badewannenkurve") kann durch gezielte Maßnahmen verbessert werden. Was passiert in dieser ersten Phase, und was kann dagegen getan werden ?

**Es treten Frühausfälle, meist Fertigungsfehler, auf.
Beispiel : Lötfehler, Steckverbinder**

**Frühausfälle müssen noch im Werk gefunden werden,
was durch gezieltes Altern (Burn-In) erreicht wird**

8. Erklären Sie das Funktionsprinzip, das Profinet IRT deterministisch macht :

In einem synchronen Zeitfenster wird die Ethernet-Komm. durch ein starres daisy-chain Verfahren ersetzt.

9. Für welche Funktionen benötigen Sie in der Anlagenkommunikation unbedingt ein System mit **Layer 3 – Protokoll** ?

Kommunikation mit nur über öffentliche Netze erreichbare Partner, meist Internet.

10. Erklären Sie, welche Funktionen ein **MES-System** in der Struktur einer aus ERP-, MES- und Prozessebene bestehenden Anlage hat :

Zerlegen von Produktaufträgen in Fertigungsschritte

Protokollieren von Fertigungsabläufen (Tag-logging)

Fertigungsdaten für ERP komprimieren (KPI)

Teil 2, mit Unterlagen

Wir betrachten einen Produktionsprozess : Produkte werden in einer Montageeinheit (Modul E) aus 3 Komponenten montiert. Zwei der Komponenten werden vor Ort selbst hergestellt. Hierzu sind zwei Fertigungslinien vorhanden, an denen jeweils zwei Bearbeitungsautomaten (A/B sowie C/D) arbeiten. Die dritte Komponente kommt just-in-sequence von einem Zulieferer.

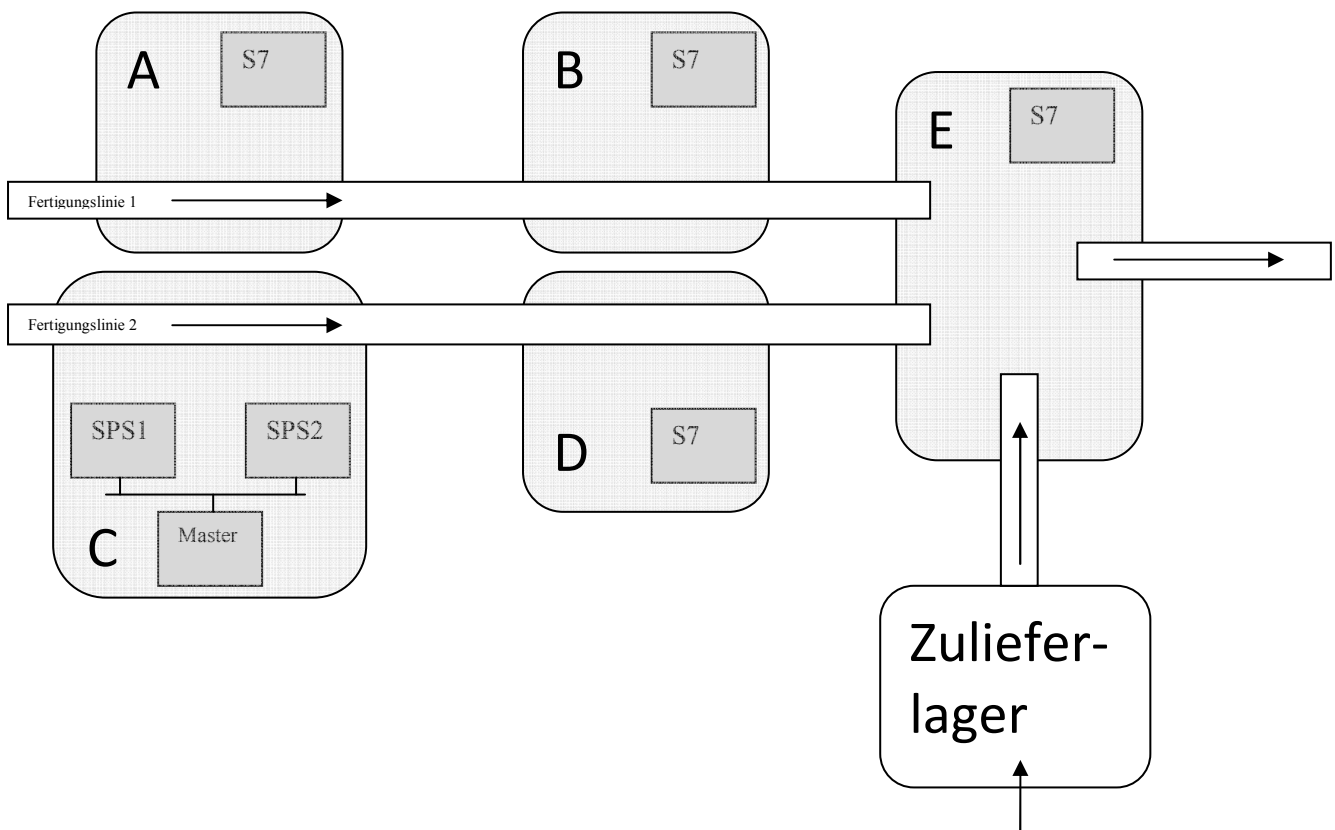
Die beiden parallel laufenden Fertigungslinien und das Montagemodul sind starr gekoppelt, der Prozess arbeitet in Losgröße 1. Die Chargengröße für die Fertigung ist auf 600 Stück festgelegt. Die Module A bis D können ohne Rüstzeit verschieden Varianten fertigen. E montiert nur (keine Varianten)

Durch parallele, starre Kopplung bestimmt das langsamste Modul den Anlagentakt.

Alle Fertigungsmodule bis auf C werden von jeweils einer Simatic S7 gesteuert.

Die Steuerung von Modul C besteht intern aus 3 SPS, die über einen ProfibusDP - Strang untereinander vernetzt sind. Die beiden Slaves steuern redundant aufgebaute Anlagenteile in C.

Der Materialtransport auf den Fertigungslinien funktioniert unabhängig, und muß nicht betrachtet werden.

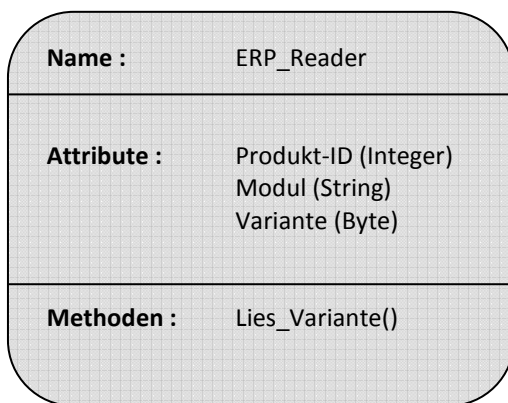


Ein ERP-System von SAP speichert redundanzfrei in seiner Datenbank alle Auftragsinformationen und alle Stücklisten. Die Produktaufträge können von seiner Webschnittstelle (als XML-Daten) abgefragt werden.

Das MES-System steuert die SPS über TCP/IP (in Modul C nur den Master)

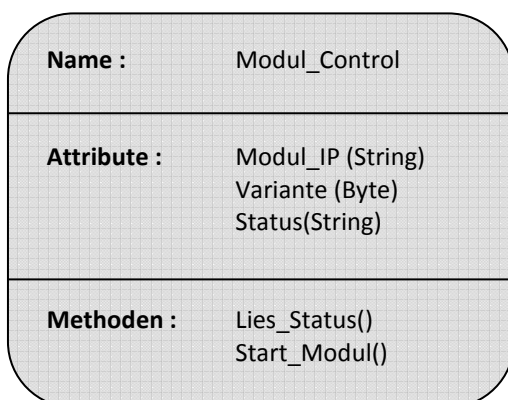
Die Programmierung des MES geschieht objektorientiert in VisualBasic. Hierzu steht eine Klassenbibliothek MESLIB zur Verfügung :

Klasse : ERP_Reader



Lies_Variante() greift auf das ERP-System zu . Die **Produkt-ID** (1-599 in jeder Charge) und das **Modul** (A-D) müssen vorher in die Attribute geschrieben werden, die am Modul auszuführende Fertigungsvariante kann dann im Attribut **Variante** gelesen werden.

Klasse : Modul_Control



Die Module müssen mit ihren IP-Adressen (xxx . xxx . xxx . xxx) parametrierung werden.

Im Attribut **Variante** wird dem Modul mitgeteilt, welche Fertigungsvariante durchzuführen ist.

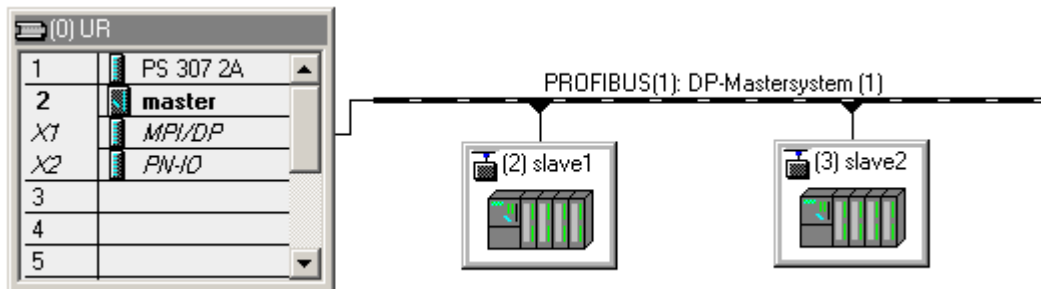
Die Methode **Start_Modul()** gibt der SPS den Startbefehl.

Die Methode **Lies_Status()** liest aus der SPS den aktuellen Betriebszustand und gibt ihn als String im Attribut **Status** aus.

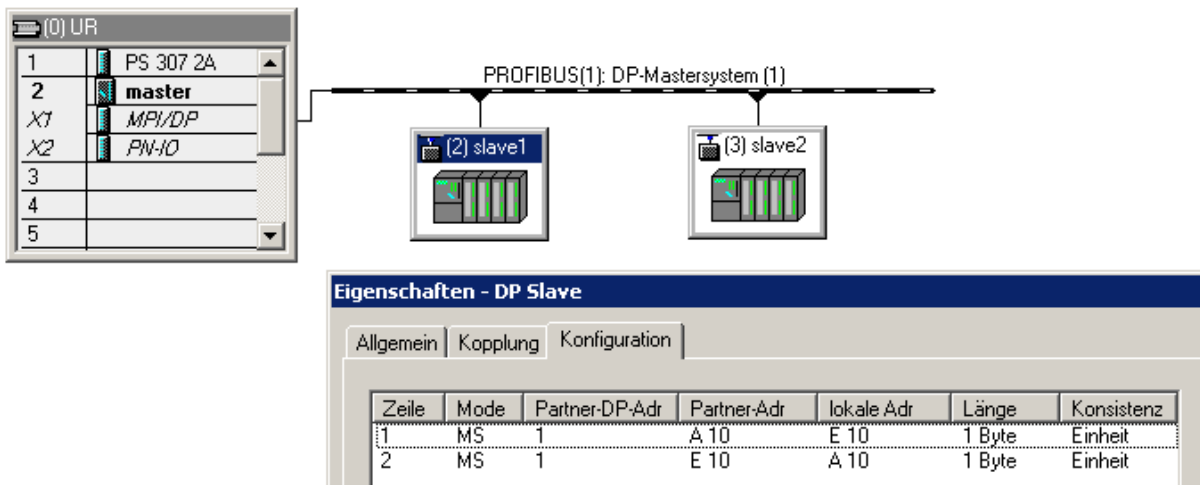
Möglich sind hier : READY und BUSY (wie bei den Modulen der digitalen Fabrik)

Konfiguration des Profibus DP – Stranges in Modul C :

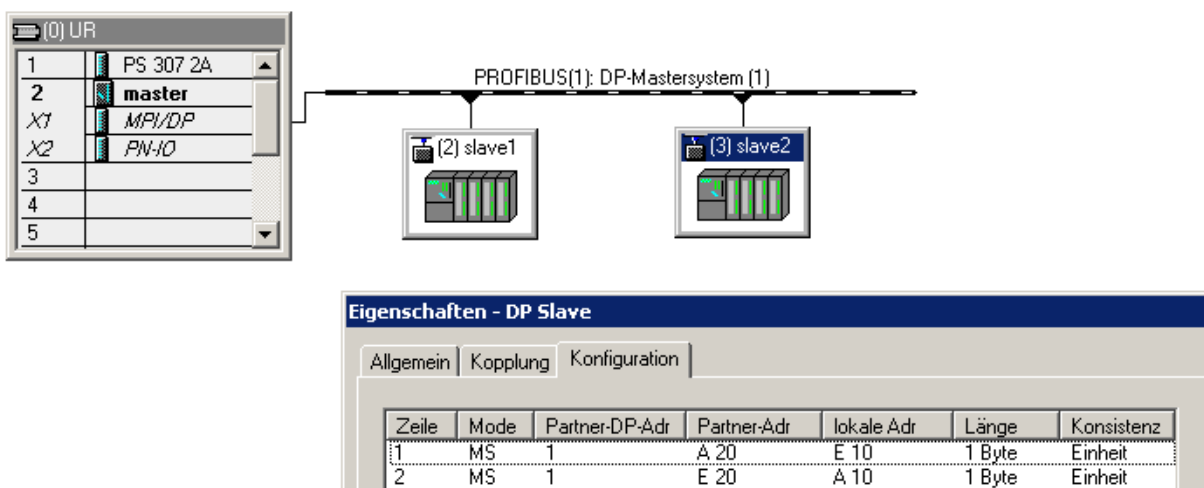
Gesamtbild :



Konfiguration von Slave1 am Master :



Konfiguration von Slave2 am Master :



1. Aufgabe (Anlagenkonzept)

Erklären Sie kurz, was der Begriff „just-in-sequence“ hier bedeutet :

Durch die Losgröße 1 in der Fertigung muß der Zulieferer die Bauteile In der richtigen Reihenfolge (=Sequenz) liefern.

2. Aufgabe (Profibus DP)

Bei Ausfall der Mechanik an SPS1 in C übernimmt SPS2 mit ihrer Mechanik die Funktion. Hierzu ist folgende Kommunikation definiert :

SPS1 meldet mit **FAULT** an SPS2, daß ihre Mechanik defekt ist.

SPS2 bietet daraufhin die Redundanzfunktion an : **CHANGE** wird an SPS1 geschickt.

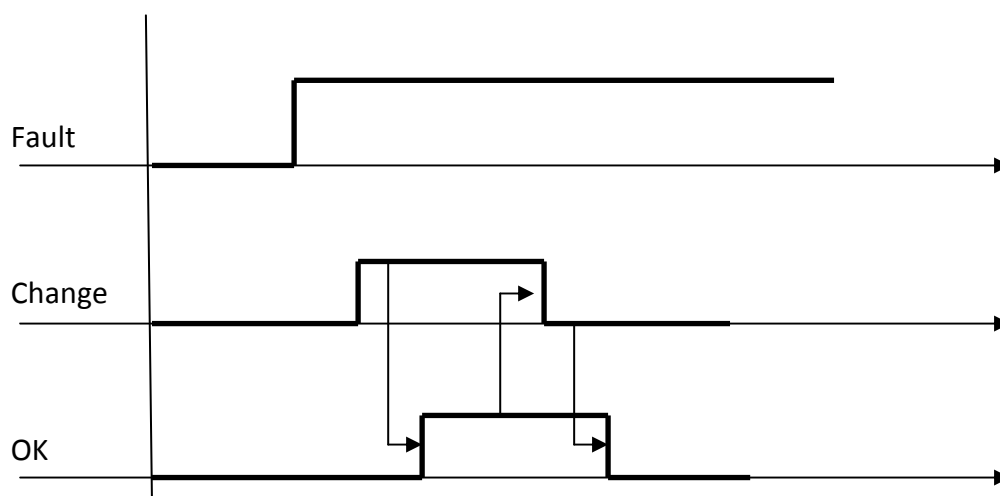
SPS1 quittiert mit **OK** .

FAULT an SPS1 bleibt 1, das Zurücksetzen auf die ursprüngliche Funktion nach Reparatur der Mechanik erfolgt durch manuellen Neustart der SPS.

CHANGE und **OK** sind Handshake-Signale. Verdeutlichen Sie diese Funktion im Timing durch Bedingungspeile.

Aufgabenteil 2.1

Zeichnen Sie hier ein Timingdiagramm des oben beschriebenen Protokolls :



Aufgabenteil 2.2

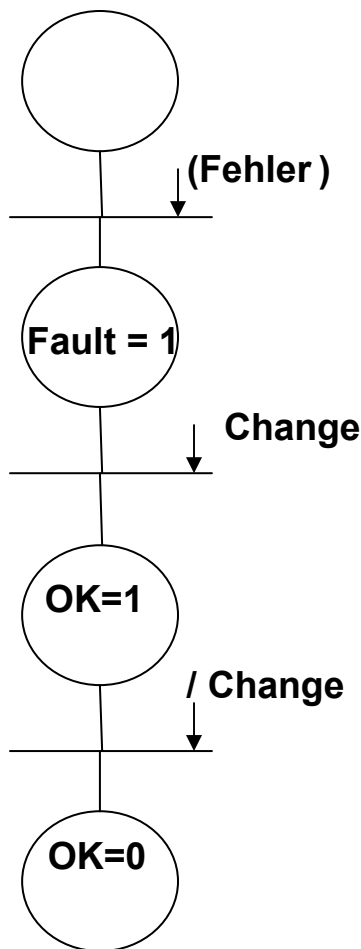
Die Adressbelegung in den Slave-SPS ist folgende :

SPS1 : FAULT = A 10.0
 OK = A 10.1
 CHANGE = E 10.0

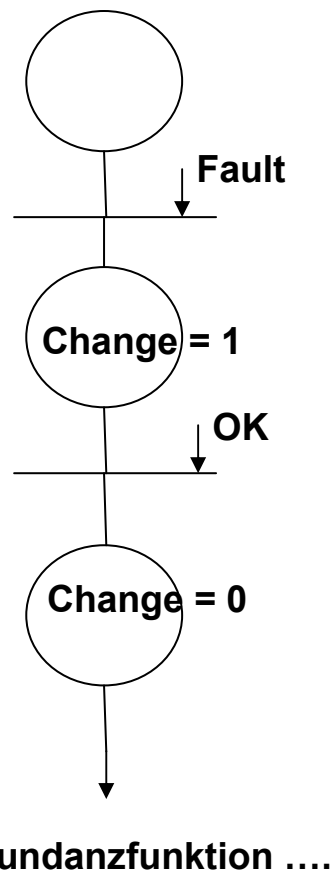
SPS2 : CHANGE = A 10.0
 FAULT = E 10.0
 OK = E 10.1

Geben Sie in allen an der Kommunikation beteiligten SPS je ein Programmstück zur Realisierung der beschriebenen Kommunikation an. In SPS1 soll ein symbolisches Signal „Fehler“ den Ablauf auslösen. Zur Programmierung in den Slaves nutzen Sie bitte Graph7 –ähnliche Syntax, im Master schreiben Sie bitte AWL (oder FUP).

SPS 1 :



SPS 2 :



Master :
 L EB20
 T AB10
 L EB10
 T AB20

oder Bitebene : U E 10.0 (fault)
 = A 20.0
 U E 20.0 (change)
 = A 10.0
 U E 10.1 (ok)
 = A 20.1

Aufgabenteil 2.3

Die Kommunikation innerhalb Modul C soll von Profibus DP auf ein Ethernet-basiertes System umgerüstet werden. Hier sind 3 Varianten in der Diskussion :

Profinet I/O, Profinet IRT und offene Ethernet TCP/IP-Kommunikation

Frage 2.3.1 : Welche Änderungen in der Hardware benötigen die jeweiligen Varianten ?

Frage 2.3.2 : Welche Unterschiede im Echtzeitverhalten haben die Systeme ?

Frage 2.3.3 : Bei welchen Varianten wäre ein Zugriff auf die Kommunikationsdaten von einem entfernten Netz über das Internet möglich ?

2.3.1 I/O und TCP/IP normales Ethernet, IRT Spezialhardware

2.3.2 IRT deterministisch, Rest nicht

2.3.3 TCP/IP weil nur dort Layer 3

Aufgabe 3

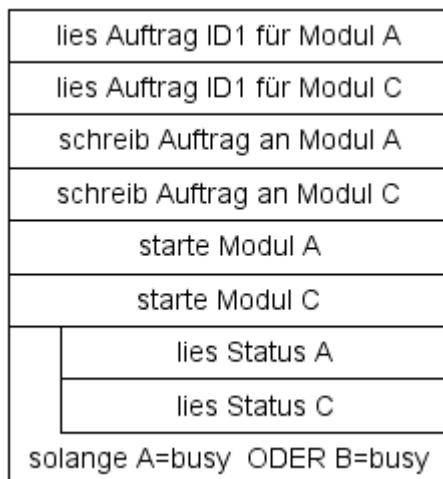
Ein kompletter Chargenlauf besteht aus Anlauf, Serienbetrieb und Auslauf.
Geben Sie einen tabellarischen work-plan für den Chargenanlauf an (nur Anlauf !).

	Takt 1	Takt 2	Takt 3		
Modul A	Teil 1a	Teil 2a	Teil 3a	Teil 4	
Modul B	-	Teil 1b	Teil 2b	Teil 3	
Modul C	Teil 1c	Teil 2c	Teil 3c	Teil 4	
Modul D	-	Teil 1d	Teil 2d	Teil 3	
Modul E	-	-	Teil 1	Teil 2	

(Zulieferer nicht gefragt...)

Aufgabe 4

Skizzieren Sie ein Struktogramm, das alle nötigen Aktionen zur Ausführung des ersten aktiven Taktzyklus (nur ein Takt !) ihres work-plan darstellt.



Wichtig ist, daß die Aktionen nicht seriell (nacheinander) ausgeführt werden !

Aufgabe 5

Codieren Sie das Struktogramm in VB. Variablendeklarationen (falls sie welche brauchen) und Referenzierung von Klassen geben Sie bitte mit an !

Imports MESLIB

Public class ersterschritt

Private A as New Modul_Control

Private B as New Modul_Control

Private ERP as New ERP_Reader

A.IP = 1.1.1.1

B.IP = 1.1.1.2

Private Sub irgendwas

ERP.ID = 1

ERP.Modul = "A"

ERP.lies_variante()

A1 = ERP.variante

ERP.ID = 1

ERP.Modul = C

ERP.lies_variante()

C1 = ERP.variante

A.variante = A1

C.variante = C1

A.start()

C.start()

Do

A.lies_status()

C.lies_status()

Loop while A.status = "busy" OR B.status = "busy"

End Sub

End Class