

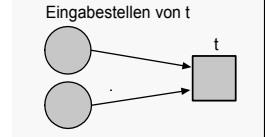
Inhalt

12 Petri-Netze

- 12.1 Grundlagen
- 12.2 Bedingungs/Ereignis-Netze
- 12.3 Stellen/Transitions-Netze
- 12.4 Prädikat/Transitions-Netze
- 12.5 Hierarchische Petri-Netze
- 12.6 Zeitbehaftete Petri-Netze
- 12.7 Strukturelemente und Strukturen von Petri-Netzen
- 12.8 Methodik
- 12.9 Analyse und Simulation von Petri-Netzen
- 12.10 Wertung.

12.1 Grundlagen

- Die Kanten dürfen jeweils nur von einer Sorte zur anderen führen
- Stellen, von denen Kanten zu einer Transition t laufen, heißen Eingabestellen von t .



12 Petri-Netze

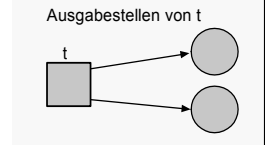
• Zur Historie

- ◆ Prof. Dr. Carl Adam Petri
*12.7.1926 in Leipzig
Professor an der Universität Hamburg
- ◆ Erfinder der Petri-Netze
- ◆ Entwicklung einer allgemeinen Theorie diskreter Systeme, die auf den Konzepten der Nebenläufigkeit, der Verteiltheit und der asynchronen Kommunikation aufbaut.



12.1 Grundlagen

- Stellen, zu denen – von einer Transition t aus – Kanten führen, heißen Ausgabestellen von t



- Objekte werden als Marken bezeichnet und als kleine schwarze Kreise in die Stellen des Petri-Netzes eingetragen



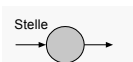
12.1 Grundlagen

• Petri-Netz

- ◆ Gerichteter Graph, der aus zwei verschiedenen Sorten von Knoten besteht: aus Stellen und Transitionen

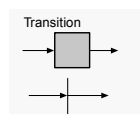
Stellen
(Plätze, Zustände)

entspricht einer Zwischenablage von Informationen



Transitionen
(Hürden, Zustandsübergänge)

beschreibt die Verarbeitung von Informationen

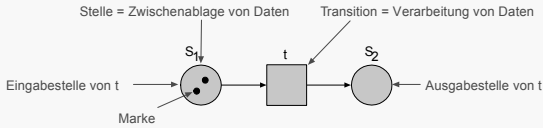


12.1 Grundlagen

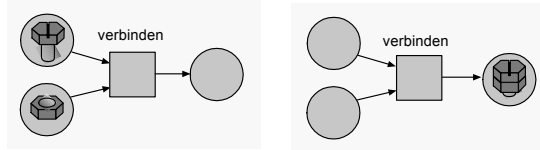
• Schaltregel:

- a Eine Transition t kann schalten oder »feuern«, wenn jede Eingabestelle von t mindestens eine Marke enthält
- b Schaltet eine Transition, dann wird aus jeder Eingabestelle eine Marke entfernt und zu jeder Ausgabestelle eine Marke hinzugefügt.

12.1 Grundlagen



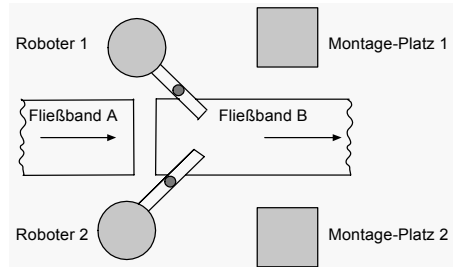
• Anschauliche Vorstellung der Schaltregel



12.2 Bedingungs/Ereignis-Netze

Beispiel

◆ Zwei Roboter bestücken Leiterplatten mit elektronischen Bauelementen, die auf einem Fließband A antransportiert werden



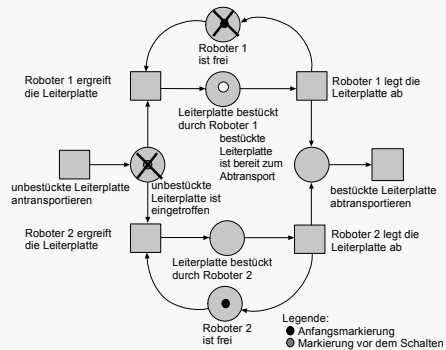
12.1 Grundlagen

In Abhängigkeit von der Art der Objekte unterscheidet man:

- ◆ Bedingungs/Ereignis-Netze (B/E-Netz)
- ◆ Stellen/Transitions-Netze (S/T-Netz)
- ◆ Höhere Petri-Netze.

12.2 Bedingungs/Ereignis-Netze: Beispiel

◆ B/E-Netz des Bestückungsroboters



Vor dem Schalten

Nach dem Schalten.

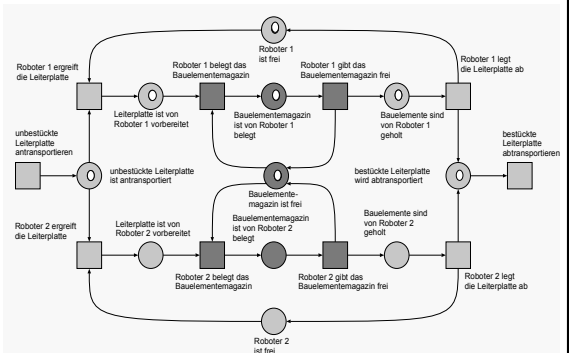
12.2 Bedingungs/Ereignis-Netze

B/E-Netz

- ◆ Wenn die Objekte bzw. Marken vom Datentyp *boolean* sind
- ◆ Die Transitionen werden als Ereignisse interpretiert
- ◆ Die Stellen werden als Bedingungen bezeichnet
- ◆ Jede Stelle kann entweder genau eine oder keine Marke enthalten
- ◆ Zusätzliche Schaltbedingung:
C Eine Transition t kann schalten, wenn jede Eingabestelle von t eine Marke enthält und wenn jede Ausgabestelle von t leer ist.

12.2 Bedingungs/Ereignis-Netze

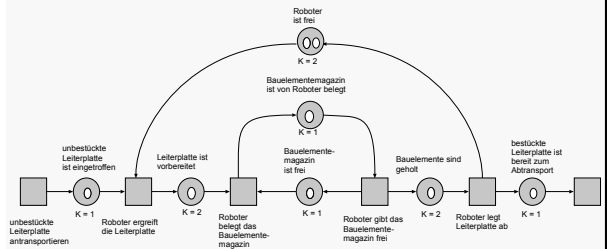
◆ B/E-Netz mit gemeinsamem Bauelemente-Magazin



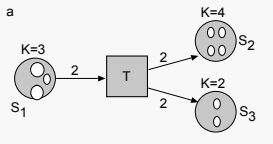
12.3 Stellen/Transitions-Netze

- S/T-Netze (P/T Net, Place/Transition Net)
 - ◆ Stellen können mehr als eine Marke enthalten (in B/E-Netzen nur eine Marke)
 - ◆ Transitionen müssen so viele Marken beim Schalten wegnehmen oder hinzufügen, wie die Gewichte an den Pfeilen angeben (in B/E-Netzen nur eine Marke)
 - ◆ Soll eine Stelle eine Kapazität größer 1 erhalten, dann wird dies durch »K = ...« an der Stelle notiert
 - ◆ Die Kapazität definiert die maximale Anzahl von Marken, die auf einer Stelle liegen dürfen.

12.3 Stellen/Transitions-Netze
S/T-Netz des Bestückungsroboters



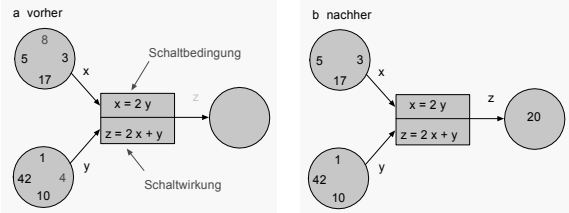
12.3 Stellen/Transitions-Netze
Schaltbedingungen bei S/T-Netzen



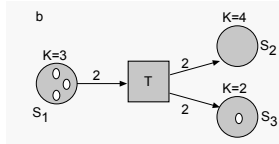
- ◆ T kann schalten
 - Anschl. sind in S₁ eine Marke, in S₂ 4 Marken und in S₃ 2 Marken

12.4 Prädikat/Transitions-Netze
Pr/T-Netze

- ◆ Verwenden individuelle, »gefärbte« Marken
- ◆ B/E- und S/T-Netze verwenden nur »schwarze« Marken, die alle gleich sind

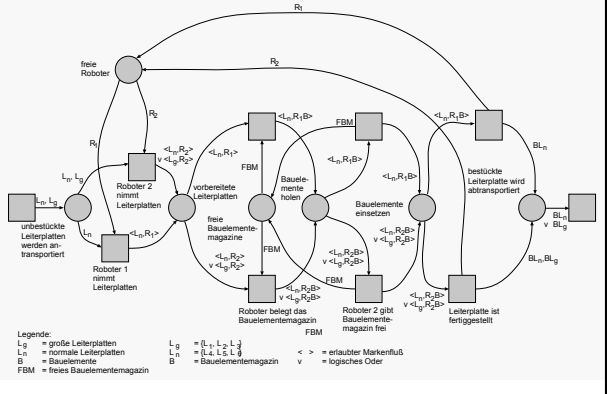


12.3 Stellen/Transitions-Netze
Schaltbedingungen bei S/T-Netzen



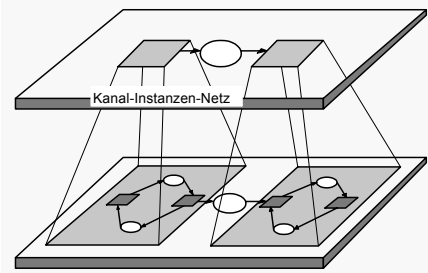
- ◆ T kann nicht schalten, da in S₃ dann 3 Marken liegen würden
 - Dies ist wegen K = 2 von S₃ nicht erlaubt.

12.5 Hierarchische Petri-Netze



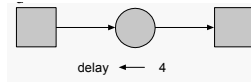
19 **12.5 Hierarchische Petri-Netze**
Strukturierte Darstellung eines komplexen Netzes

- ◆ Erweitern nicht das Netz-Modell

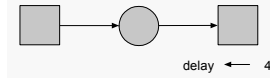


22 **12.6 Zeitbehaftete Petri-Netze**
Verschiedene Notationsmöglichkeiten

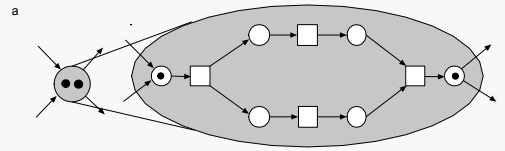
- ◆ Eine Marke wird erst nach einer Verzögerung von 4 Zeiteinheiten wirksam



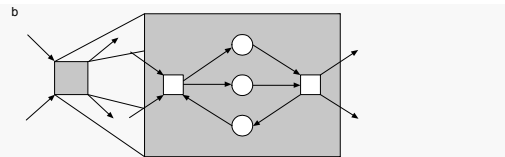
- ◆ Die Transition kann erst 4 Zeiteinheiten nach dem Eintreffen der Marke in der Eingangsstelle feuern



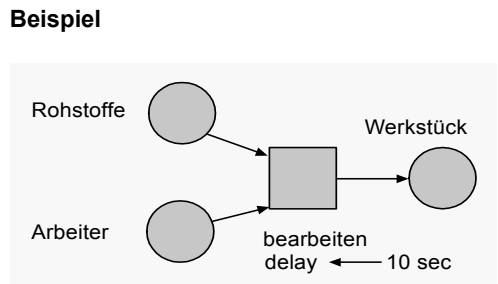
20 **12.5 Hierarchische Petri-Netze**
Markengetreue Verfeinerung einer Stelle



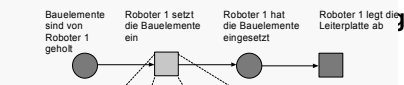
- ◆ Markengetreue Verfeinerung einer Transition



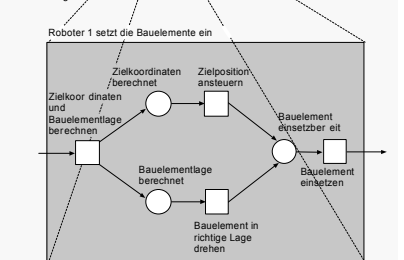
23 **12.6 Zeitbehaftete Petri-Netze**
Beispiel



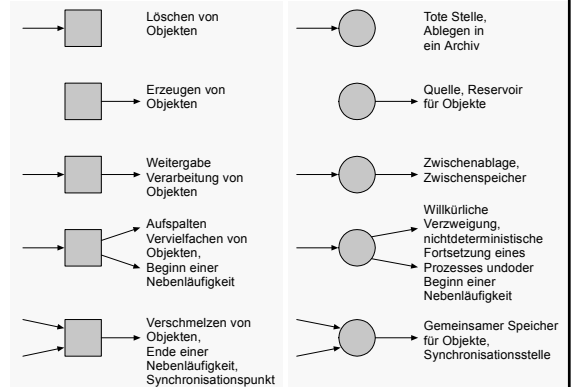
21 **12.5 Hierarchische Petri-Netze**



b nach der Verfeinerung

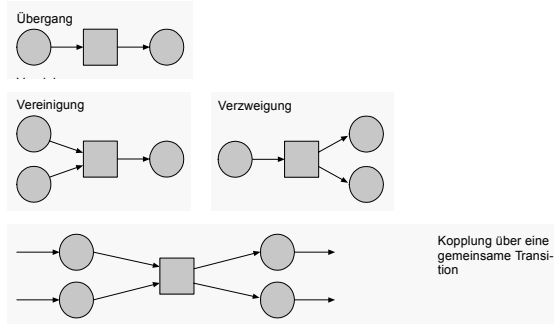


24 **12.7 Strukturelemente & Strukturen**

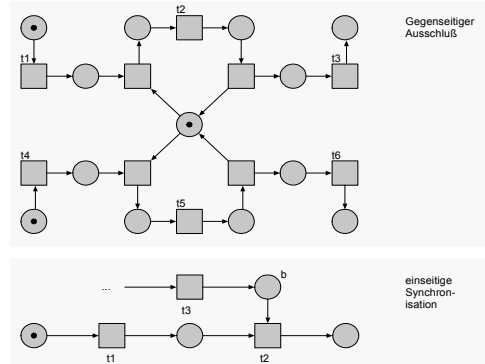


12.7 Strukturelemente & Strukturen

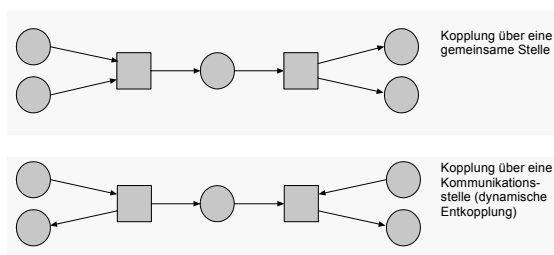
Strukturen



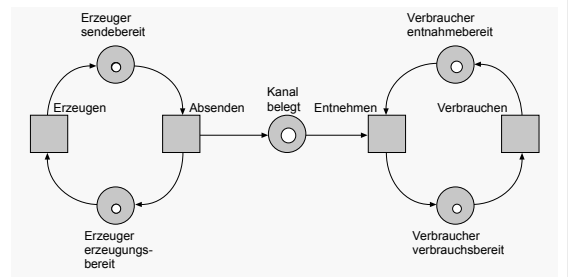
12.7 Strukturelemente & Strukturen



12.7 Strukturelemente & Strukturen



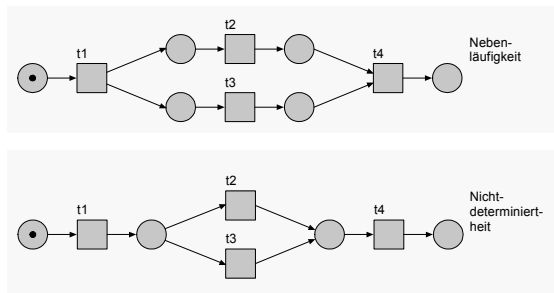
12.7 Strukturelemente & Strukturen Produzenten & Konsumenten bzw. Erzeuger & Verbraucher im Petri-Netz



12.7 Strukturelemente & Strukturen

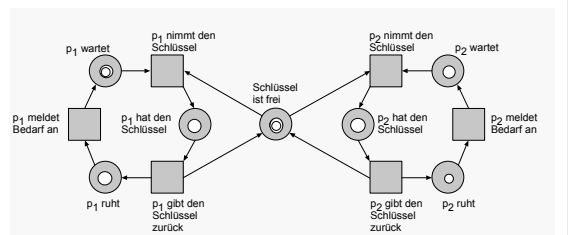
Typische Anwendungsmuster

◆ Aufgebaut aus Strukturen



12.7 Strukturelemente & Strukturen

• Leser & Schreiber im Petri-Netz

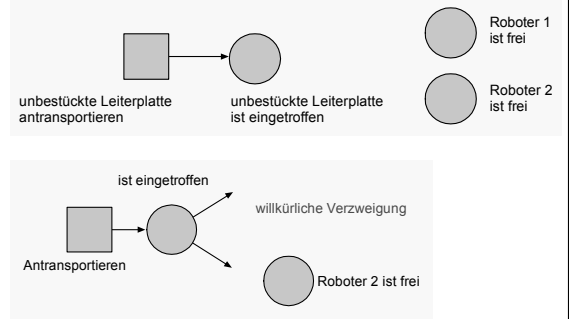


12.8 Methodik

1. Aktive und passive Komponenten identifizieren
2. Beziehungen ermitteln
3. Verfeinerung und Ergänzung
4. Festlegung der Objekte
5. Überlegungen zu Schaltregeln und Schaltwirkungen
6. Netztyp festlegen
7. Anfangsmarkierung festlegen
8. Analyse, Simulation.

12.8 Methodik: Beispiel

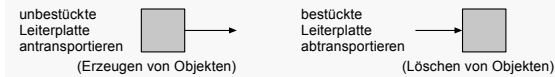
2. Schritt: Beziehungen ermitteln



12.8 Methodik: Beispiel

1. Schritt: aktive & passive Komp. identifizieren

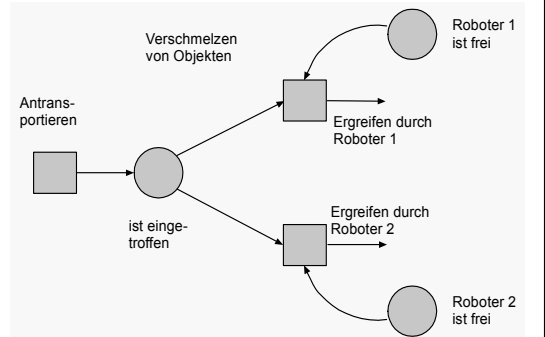
- ◆ Analysiert man die Problembeschreibung der Leiterplattenbestückung, dann werden die Leiterplatten vom Fließband A *antransportiert* und vom Fließband B *abtransportiert*



- ◆ Für diese Problemstellung ist uninteressant, was vor dem Antransport und nach dem Abtransport mit den Leiterplatten geschieht
- Schnittstellenmodellierung ist problemgerecht.

12.8 Methodik: Beispiel

◆ Verschmelzen von Roboterarm und Leiterplatte



12.8 Methodik: Beispiel

- ◆ Anhand der Problembeschreibung lassen sich passive Komponenten identifizieren

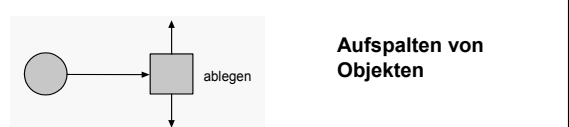


- ◆ Es ergeben sich weitere aktive Komponenten

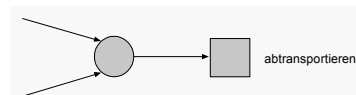


12.8 Methodik: Beispiel

Leiterplatte bestückt durch Roboter 1 **Leiterplatte bestückt?**

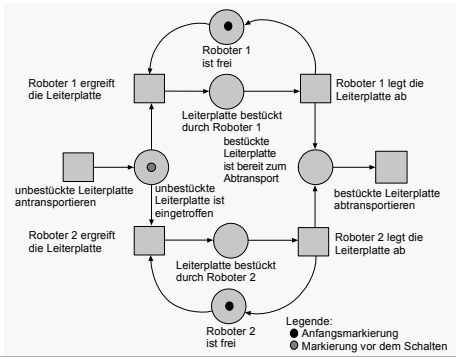


Die getrennt bearbeiteten Leiterplatten werden anschließend wieder auf dem Transportband B vereinigt



12.8 Methodik: Beispiel

◆ Ergebnis



12.9 Analyse und Simulation

- Typische Fragestellungen zur Analyse:
 - ◆ Terminiert das Netz?
 - Können, ausgehend von einer Anfangsmarkierung, stets nur endlich viele Transitionen schalten?
 - ◆ Ist jede Transition lebendig?
 - Können, ausgehend von einer Anfangsmarkierung, die Transitionen stets so schalten, daß eine vorgegebene Transition t im weiteren Verlauf nochmals schalten kann?
 - ◆ Treten vermeidbare Verklemmungen auf?
 - Gibt es Situationen, in denen keine Transition schalten kann, die aber bei anderer Schaltreihenfolge hätten vermieden werden können?

12.8 Methodik: Beispiel

3. Verfeinerung und Ergänzung

◆ Nach dem Erstellen eines Kanal-Instanzen-Netzes kann man iterativ...

- Instanzen und Kanäle verfeinern
- das erstellte Netz verfeinern

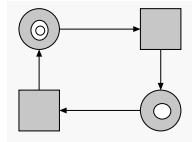
4. Festlegung der Objekte

◆ Welche konkreten Objekte können die Kanäle bzw. Stellen beinhalten?

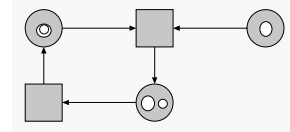
- Reichen anonyme Objekte aus oder werden individuelle Objekte benötigt?
- Daraus ergeben sich mögliche Netztypen.

12.9 Analyse und Simulation

• Das Netz ist lebendig, da die Transitionen immer abwechselnd schalten



• Das Netz ist todesgefährdet, da nach dem 2. Schalten keine weitere Transition mehr schalten kann.



12.8 Methodik: Beispiel

5. Überlegungen zu Schaltregeln und Schaltwirkungen

◆ Welche Schaltregeln und Schaltwirkungen sind zur Modellierung nötig?

6. Netztyp festlegen

◆ Aus den Objekt- und Schaltüberlegungen ergibt sich der benötigte Netztyp

7. Anfangsmarkierung festlegen

◆ Anfangsmarkierung überlegen

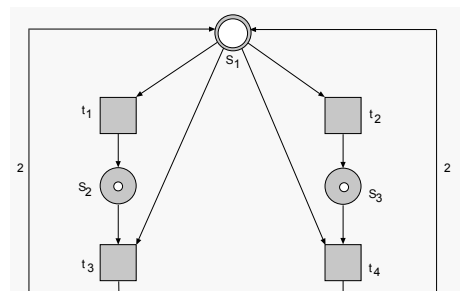
8. Analyse, Simulation

◆ Netz analysieren und simulieren.

12.9 Analyse und Simulation

Verklemmung (deadlock)

◆ Wenn eine Stellenmenge – einmal ohne Marken – nie mehr markiert werden kann



- 43
- ### 12.10 Wertung
- Petri-Netze eignen sich besonders gut zur Modellierung von Systemen mit kooperierenden Prozessen
 - Das Anwendungsspektrum umfaßt daher insbesondere
 - ◆ diskrete
 - ◆ ereignisorientierte
 - ◆ verteilte Systeme
 - Sie werden auch für die Vorgangsmodellierung von Bürovorgängen eingesetzt (*work flow*).

- 46
- ### 12.10 Wertung
- B/E-Netze
 - ◆ Gut geeignet für die Beschreibung des Kontrollflusses bei kooperierenden Prozessen
 - Dabei wird der augenblickliche Zustand jedes Prozesses durch eine Marke repräsentiert
 - S/T-Netze
 - ◆ Erweitern die Modellierungsmöglichkeiten, da beliebig viele – aber weiterhin – anonyme Marken pro Stelle abgelegt werden können.

- 44
- ### 12.10 Wertung
- Ähnlichkeiten mit Zustandsautomaten:
 - ◆ Die Stellen lassen sich als Zustände interpretieren
 - ◆ Die Transitionen lassen sich als Zustandsübergänge interpretieren
 - Zustandsautomaten
 - ◆ Ein System befindet sich zu jedem Zeitpunkt in genau einem Zustand
 - Petri-Netz
 - ◆ Ein System kann sich zu einem Zeitpunkt in mehreren Zuständen, dargestellt durch die aktuelle Markenbelegung, befinden.

- 47
- ### 12.10 Wertung
- Pr/T-Netze
 - ◆ Sind übersichtlicher als S/T-Netze
 - ◆ Die Modellierungsmächtigkeit ist größer, zumal auch beliebige, kontinuierliche Funktionen benutzt werden können
 - Beispiel: Höhe > 4000 m als Schaltbedingung
 - ◆ Nachteil: schwierigere Handhabung dieser Netze
 - ◆ Schwierig, korrekte Schaltbedingungen und Schaltwirkungen zu entwerfen
 - Generell gilt:
 - ◆ Je größer die Modellierungsmächtigkeit, desto geringer die Analysemöglichkeiten.

- 45
- ### 12.10 Wertung
- Je nach Struktur des Petri-Netzes können in einem System unabhängig voneinander Zustandsübergänge erfolgen
 - Synchronisationen zwischen nebenläufigen Systemen können durch eine geeignete Netzstruktur erzwungen werden
 - Petri-Netze besitzen...
 - ◆ eine größere Mächtigkeit
 - ◆ erlauben die Modellierung von Problemklassen, für die Zustandsautomaten nicht geeignet sind.

- 48
- ### 12.10 Wertung
- Vorteile
 - + Bestehen aus wenigen und einfachen Elementen
 - + Sind grafisch gut darstellbar
 - + Marken erlauben eine gute Visualisierung des jeweiligen Systemzustands
 - + Besitzen ein solides theoretisches Fundament
 - + Petri-Netze können – im beschränkten Rahmen – analysiert und simuliert werden
 - + Es gibt mehrere Petri-Netz-Werkzeuge, die die Erstellung, Analyse, Simulation und Code-Generierung erlauben
 - + Einziges weit verbreitete Basiskonzept zur Modellierung kooperierender Prozesse.

12.10 Wertung

• Nachteile

- Für die Praxis sind höhere Petri-Netze nötig, für die es keine einheitliche Notation gibt
- Höhere Petri-Netze sind schwer zu erstellen und zu analysieren
- Petri-Netze sind mit anderen Basiskonzepten bisher nicht kombiniert worden, d.h. es ist ein vollständig für sich stehendes Konzept
- Petri-Netze besitzen eine statische Struktur
 - Will man z.B. Vorgänge beschreiben, bei denen dauernd neue Prozesse erzeugt werden (z.B. task-Konzept in Ada), dann erweisen sich Petri-Netze als ungeeignet
- Es gibt keine allgemeine Erstellungs-Methode.

Klassifikation von Petri-Netzen

Allgemeine Bezeichnung	grafische Repräsentation	B/E - Netze	S/T - Netz	Pr/T - Netz	Zeitbasiertes Netz	Hierarchisches Netz
Stelle		Bedingung markiert oder unmarkiert	Stelle Kapazität > 0	Prädikat Kapazität > 0	Stelle Zeitintervall	Kanal
Transition		Ereignis	Transition	Ereignis Schaltbedingung & Schaltwirkung	Transition Zeitintervall	Instanz
Pfeil (Flußrelation)		ungewichtet	gewichtet	gewichtet konstante und/ oder variable Beschriftung	je nach Netztyp	je nach Netztyp
Marke		Marke uniform	Marke uniform	Objekt individuell	je nach Netztyp	je nach Netztyp