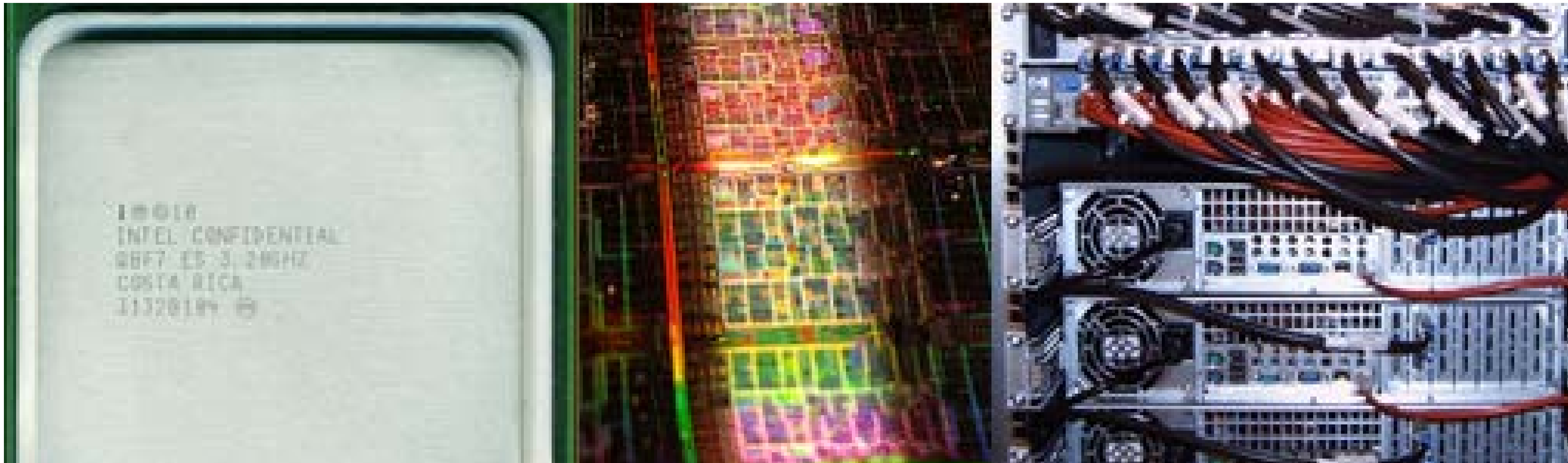
EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

MATHEMATISCH-
NATURWISSENSCHAFTLICHE FAKULTÄT



Einführung in die Technische Informatik

Laufzeiteffekte und Hasards

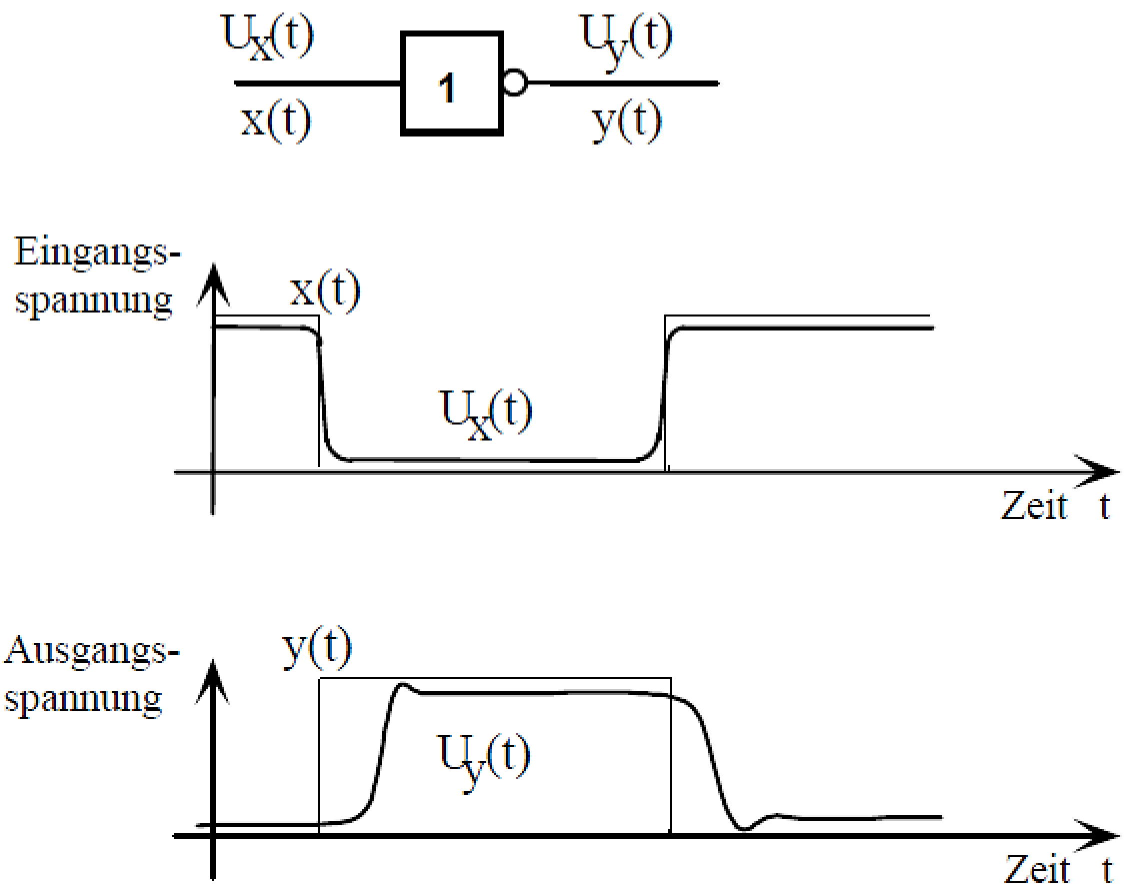
Laufzeiteffekte

Bislang wurden die Verknüpfungsglieder (Gatter) eines Schaltnetzes als ideale logische Verknüpfungen betrachtet.

In der Realität werden Gatter jedoch mittels Transistoren, Widerstände, Kapazitäten, etc. realisiert.

→ **Der zeitliche Signalverlauf eines realen Gatters weicht vom idealen Verlauf (gemäß einer Schaltalgebra definiert) ab**

Realer und idealer Signalverlauf (Inverter)



Laufzeitmodelle

Um die Einflüsse der technischen Realisierung eines Schaltnetzes besser zu beschreiben, gibt es eine Reihe verschiedener Modelle.

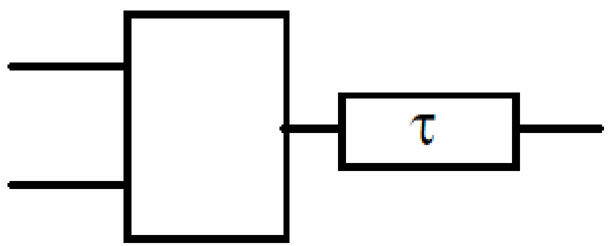
- Einfachstes Modell: **Totzeitmodell**

Es werden lediglich die durch Gatter und Leitungen entstehenden Verzögerungen berücksichtigt.

Das Totzeitmodell

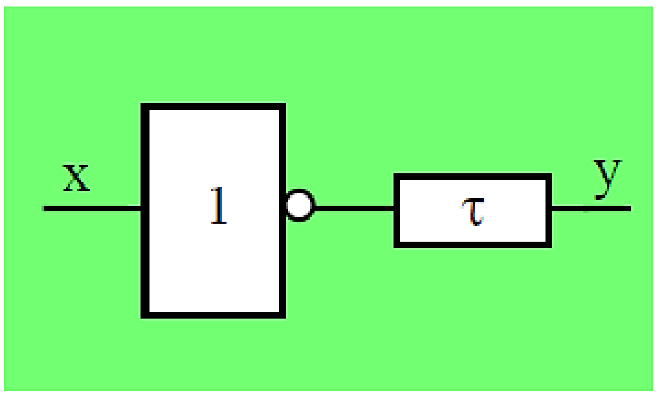
Beim **Totzeitmodell** wird ein reales Verknüpfungsglied (Gatter) modelliert durch:

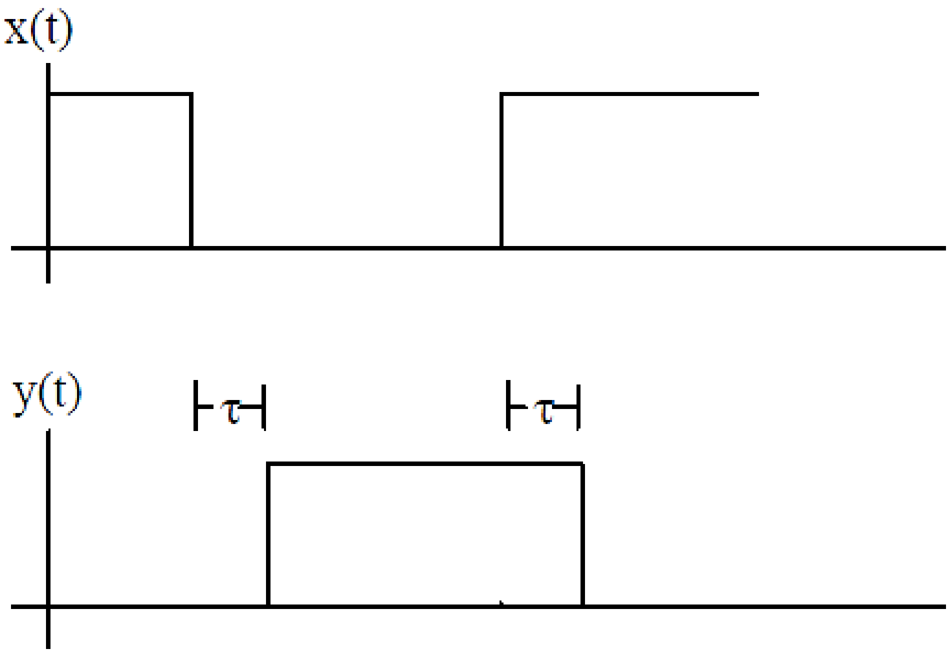
- Ein **ideales Verknüpfungsglied** ohne Verzögerungsanteil,
- ein **Totzeitglied** als reines Verzögerungsglied (steht für die Schaltzeit des Gatters und ggf. für Leitungsverzögerungen).



- Das zeitliche Verhalten einer binären Größe hinter einem Totzeitglied ist dasselbe wie dasjenige vor dem Totzeitglied, aber um die Zeit τ versetzt.

Beispiel: Totzeitmodell eines Inverters

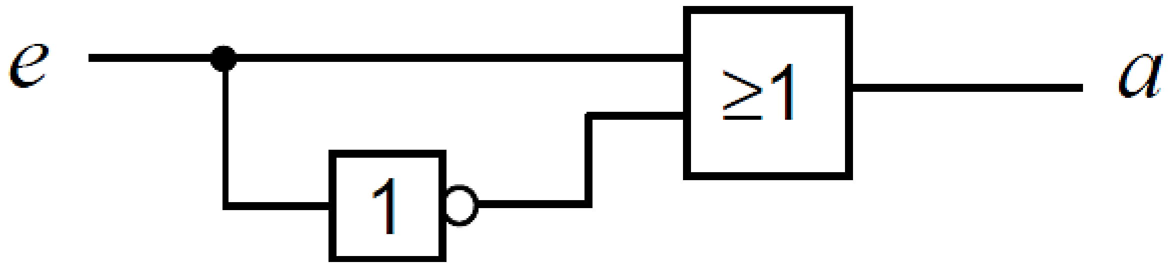




Mit Hilfe dieses einfachen Modells lassen sich Laufzeiteffekte bereits sehr gut modellieren.

Beispiel: Inverteranwendung

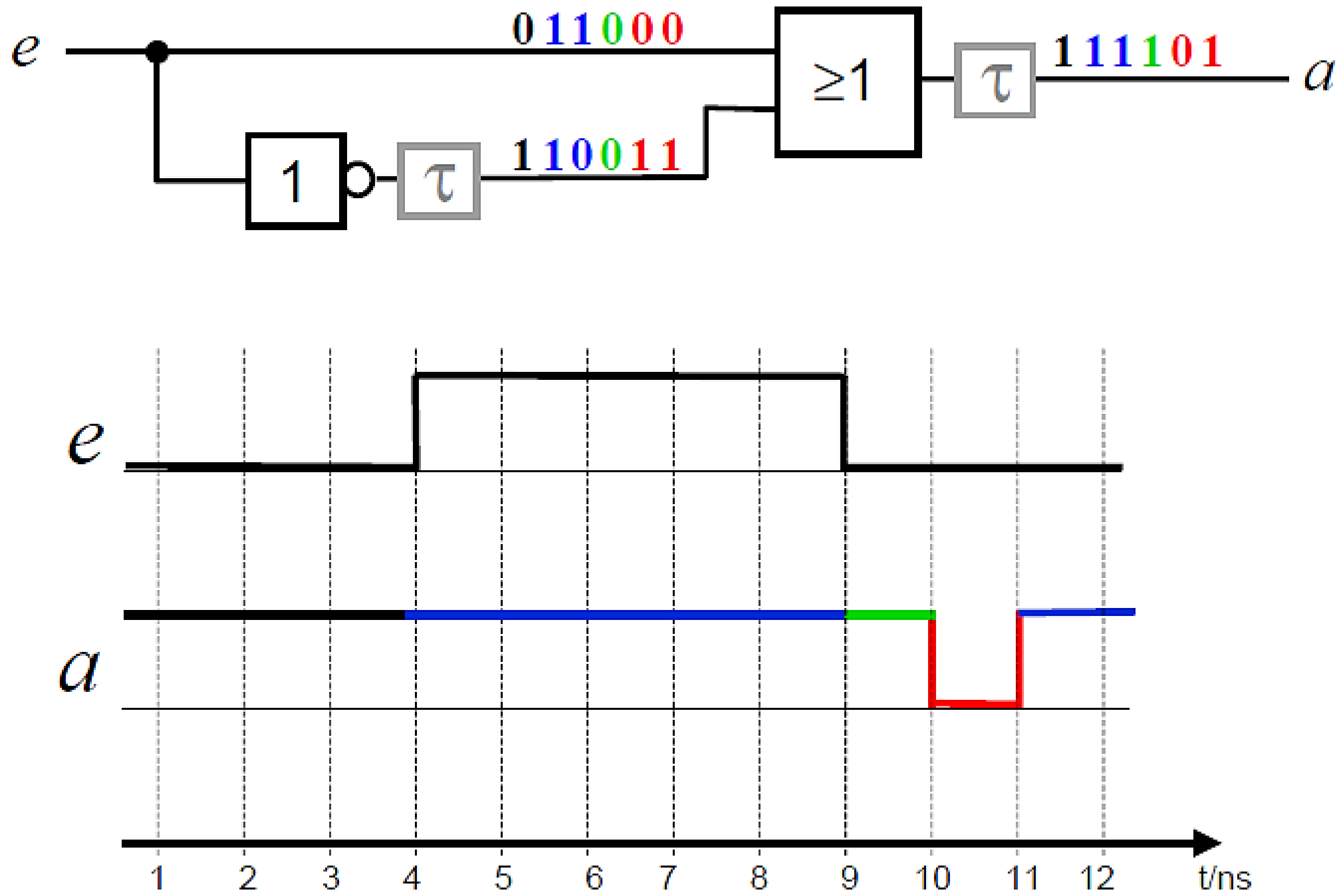
Gegeben:



$$a = e \vee \bar{e} = 1$$

Beide Gatter haben eine Verzögerungszeit von **1** *ns*

Zeit-Diagramm



Verhalten eines Schaltnetzes bei Änderung der Eingabebelegung

Ideales Schaltnetz:

- Das Ausgangssignal **ändert sich nicht**, wenn alte und neue Belegung denselben logischen Verknüpfungswert liefern.
- Das Ausgangssignal **ändert sich genau einmal**, wenn alte und neue Belegung verschiedene logische Verknüpfungswerte liefern.

Verhalten eines Schaltnetzes bei Änderung der Eingabebelegung

Reales Schaltnetz:

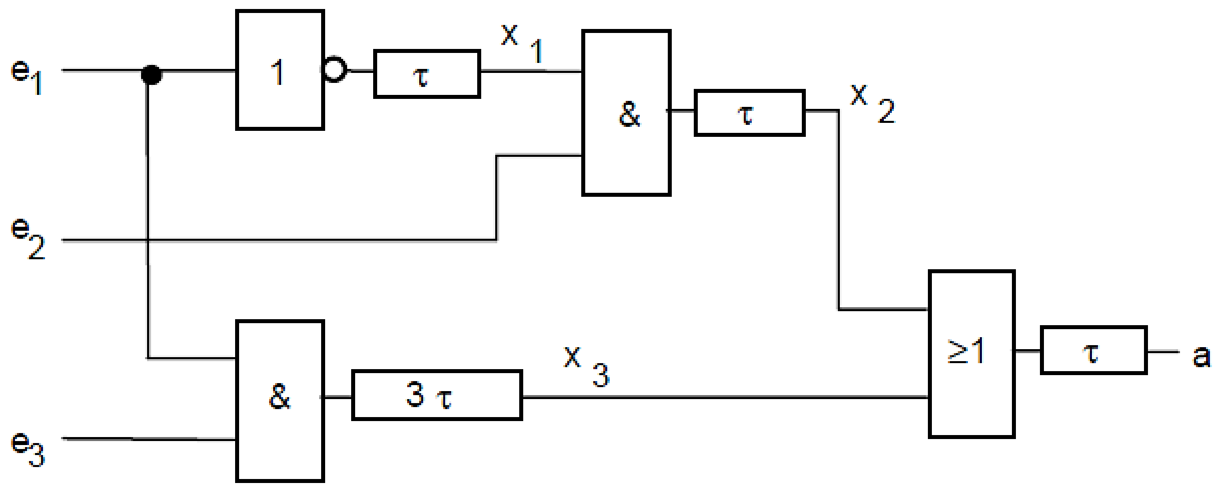
Die Änderung läuft auf verschieden langen Wegen mit verschiedenen Verzögerungen durch das Schaltnetz.

→ Mehrfache Änderungen des Ausgangssignals sind möglich, bis sich der stabile Endwert einstellt:

→ Hasardfehler

Beispiel

Funktion: $a = \bar{e}_1 e_2 \lor e_1 e_3$



		e_1	e_1		
			1		
	1		1	1	e_2
			e_3	e_3	

Eingabewechsel

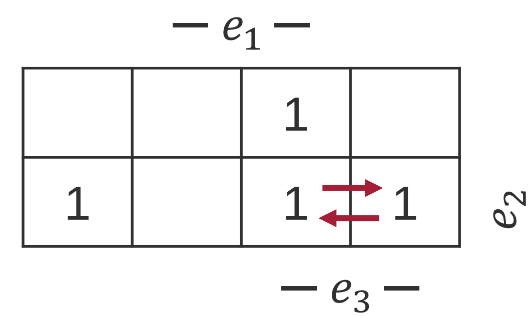
- Es sollen die folgenden Eingabewechsel betrachtet werden:

- Die Eingänge e_2 und e_3 seien konstant 1,
  der Eingang e_1 wechsle von 0 auf 1
  → $(\boldsymbol{e_3}, \boldsymbol{e_2}, \boldsymbol{e_1})$ = (1,1,0) → $(\boldsymbol{e_3}, \boldsymbol{e_2}, \boldsymbol{e_1})$ = (1,1,1)

- Die Eingänge e_2 und e_3 seien konstant 1,
  der Eingang e_1 wechsle von 1 auf 0
  → $(\boldsymbol{e_3}, \boldsymbol{e_2}, \boldsymbol{e_1})$ = (1,1,1) → $(\boldsymbol{e_3}, \boldsymbol{e_2}, \boldsymbol{e_1})$ = (1,1,0)

Funktion: $a = \bar{e}_1 e_2 \lor e_1 e_3$

- Funktionswerte bei Übergängen:
- $(\boldsymbol{e_3, e_2, e_1})$ = (1,1,0) → $\boldsymbol{a = 1}$
- $(\boldsymbol{e_3, e_2, e_1})$ = (1,1,1) → $\boldsymbol{a = 1}$

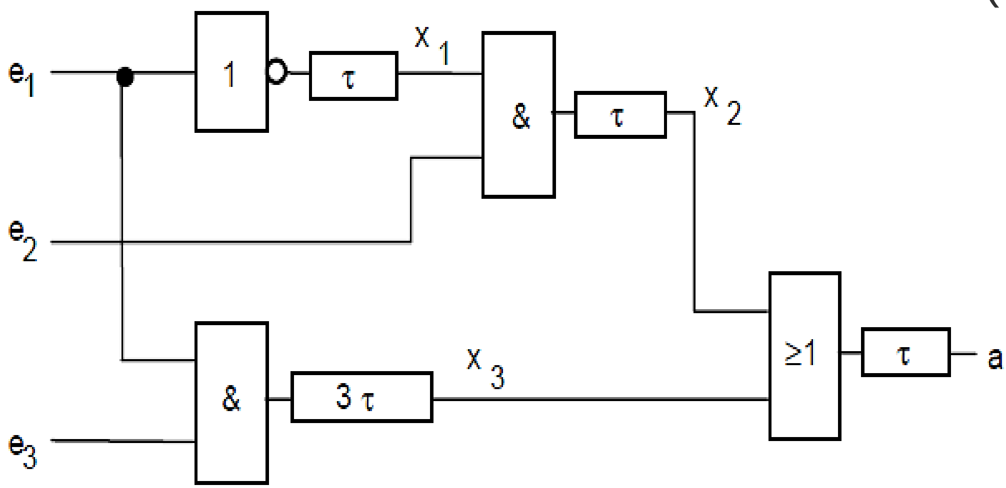


- korrektes Verhalten bei den Übergängen.

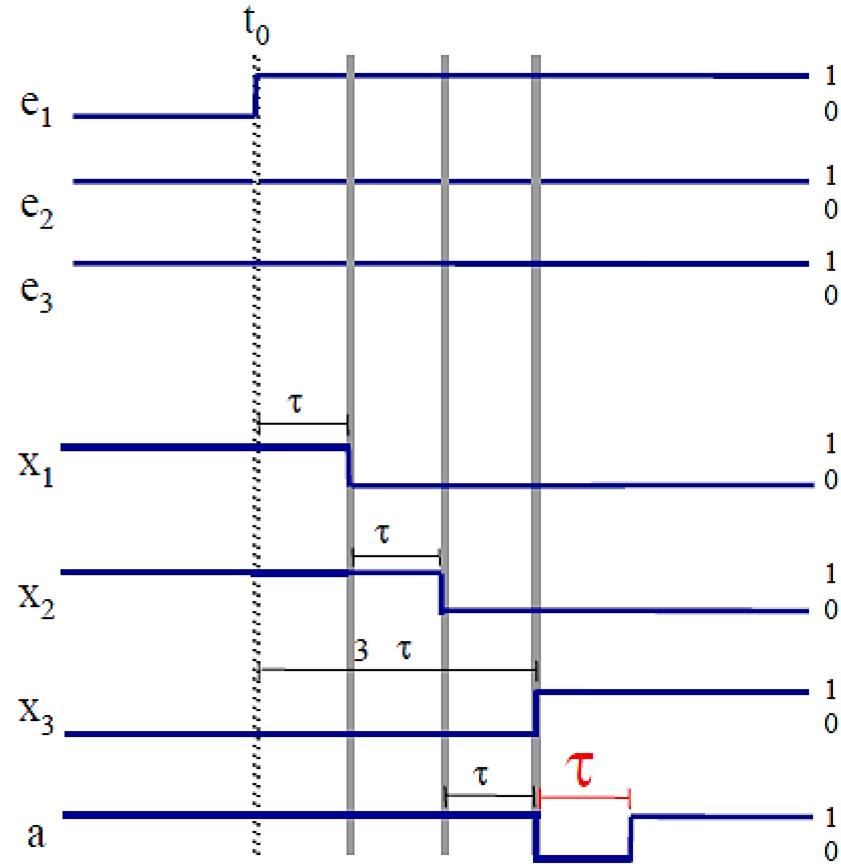
- **Logisches Verhalten:** Bei beiden Übergängen darf sich der Wert von a nicht ändern. a muss konstant 1 bleiben.

- **Reales Verhalten:** Genau dieses Verhalten kann jedoch nicht garantiert werden, sondern hängt von den Verzögerungszeiten ab!

Das Verhalten anhand des Totzeitmodells

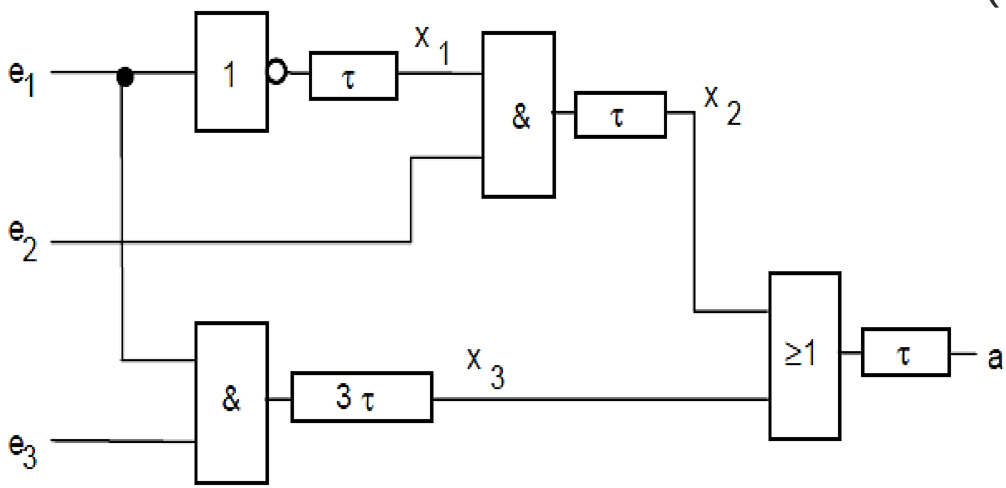


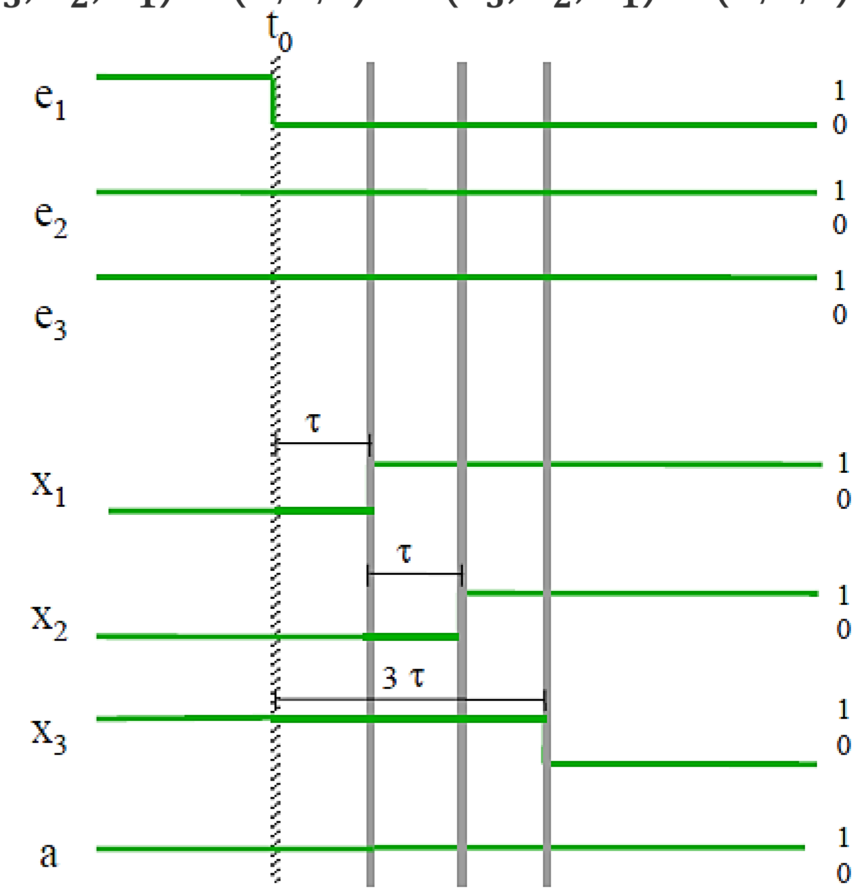
$(e_3, e_2, e_1) = (1,1,0) \rightarrow (e_3, e_2, e_1) = (1,1,1)$



Das Verhalten anhand des Totzeitmodells

$(e_3, e_2, e_1) = (1,1,1) \rightarrow (e_3, e_2, e_1) = (1,1,0)$





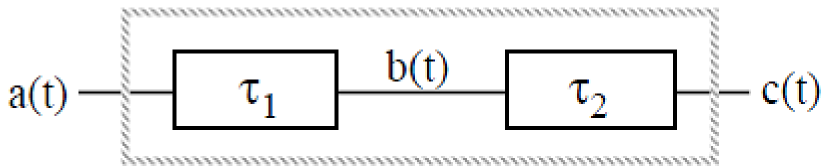
Ergebnis

- Beim Wechsel e_1 **von 0 auf 1** liefert das Ausgangssignal nicht ständig den korrekten Funktionswert
  → **Hasardfehler**

- Beim Wechsel e_1 **von 1 auf 0** ist das Ausgangssignal hingegen korrekt

Eigenschaften von Totzeiten

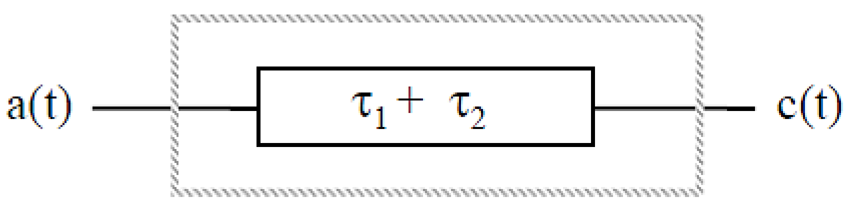
- **Addierbarkeit**: Zwei hintereinanderliegende Totzeiten können addiert und durch ihre Summe ersetzt werden.



$$b(t) = a(t - \tau_1)$$

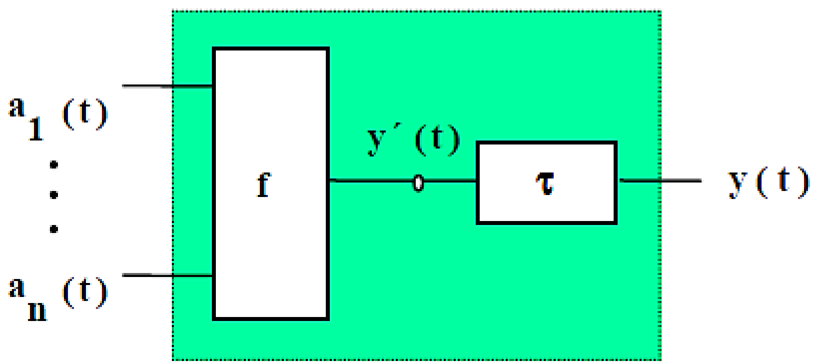
$$c(t) = b(t - \tau_2) = a(t - \tau_2 - \tau_1) = a\big(t - (\tau_2 + \tau_1)\big)$$

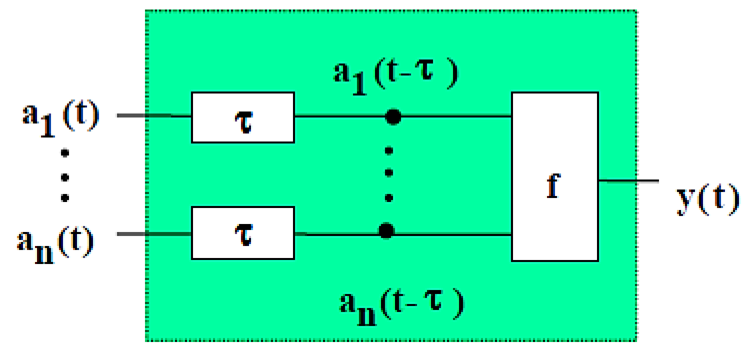
Also kann man stattdessen schreiben



Eigenschaften von Totzeiten

- **Durchschiebbarkeit:** Ein Totzeitglied, welches hinter einem Gatter mit beliebiger Verknüpfungsfunktion f liegt, kann durch das Gatter hindurch an alle Eingänge vorgeschoben werden.





$$y'(t) = f(a_1(t), \dots, a_n(t))$$

$$y(t) = y'(t-\tau) = f(a_1(t-\tau), \dots, a_n(t-\tau))$$

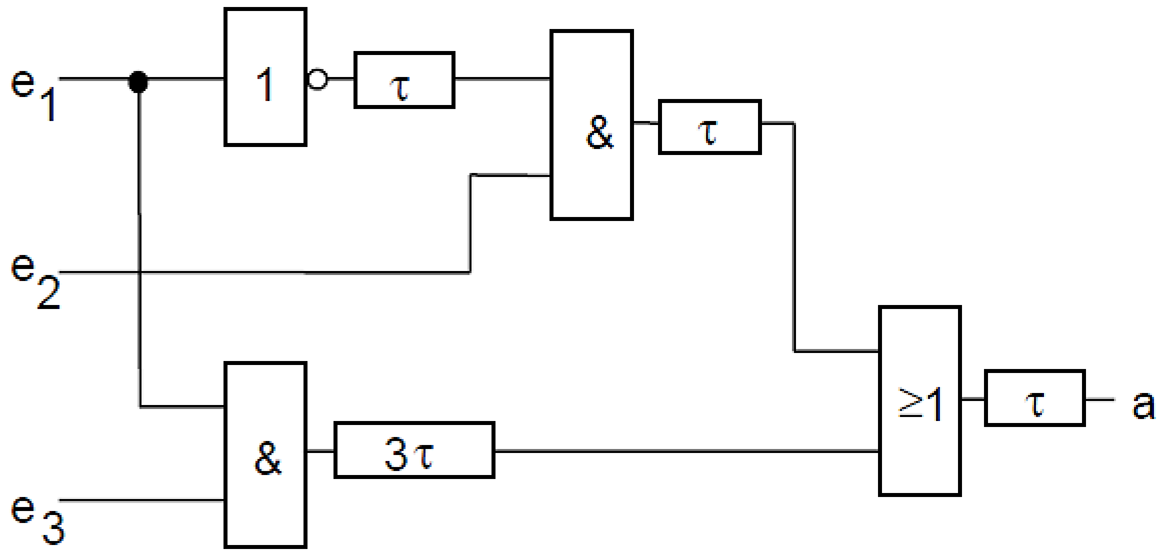
Trennung von Verzögerungs- und Verknüpfungsteil

Alle Verzögerungszeiten sukzessive zum Eingang des Schaltnetzes verschieben und aufaddieren.

Das Schaltnetz wird getrennt in

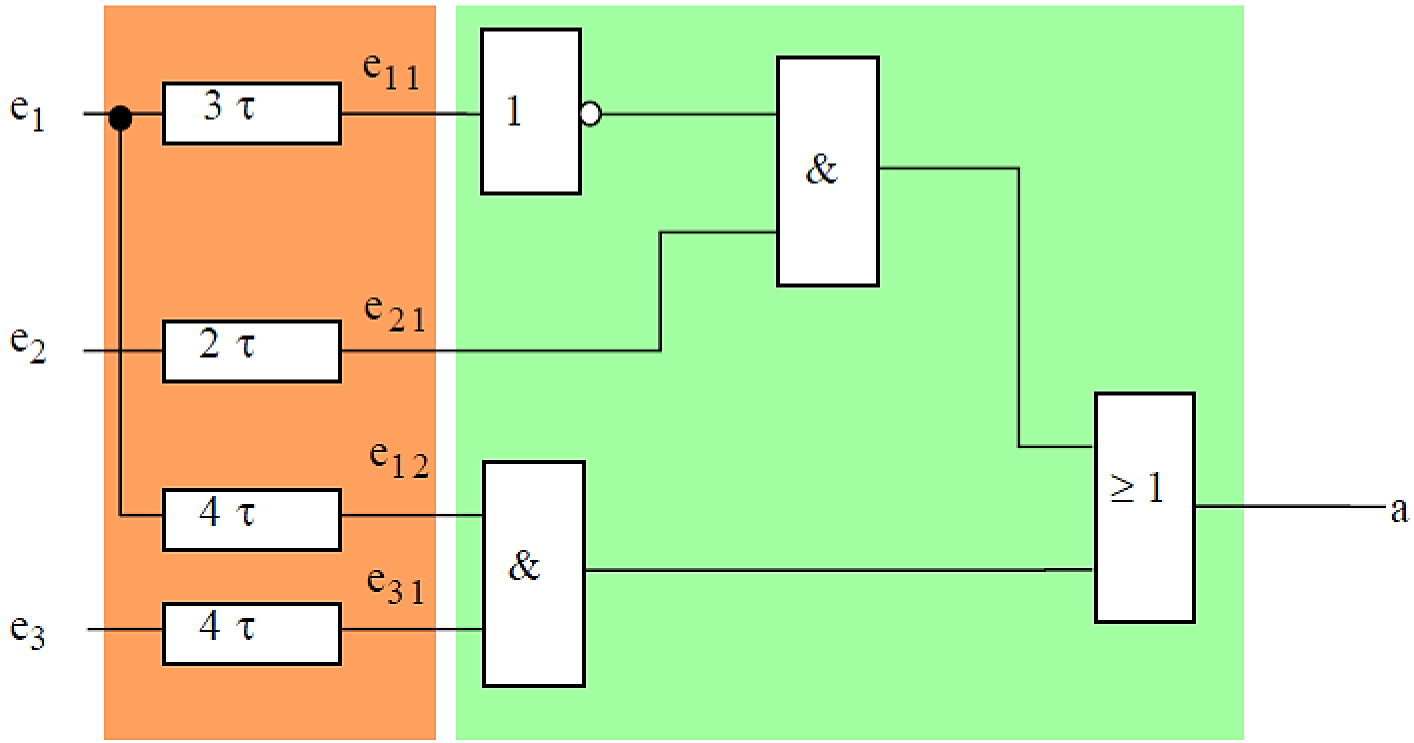
- einen **reinen Verzögerungsteil** und
- einen **reinen Verknüpfungsteil**.

Beispiel 1



Gemischte Verzögerungen und Verknüpfungen

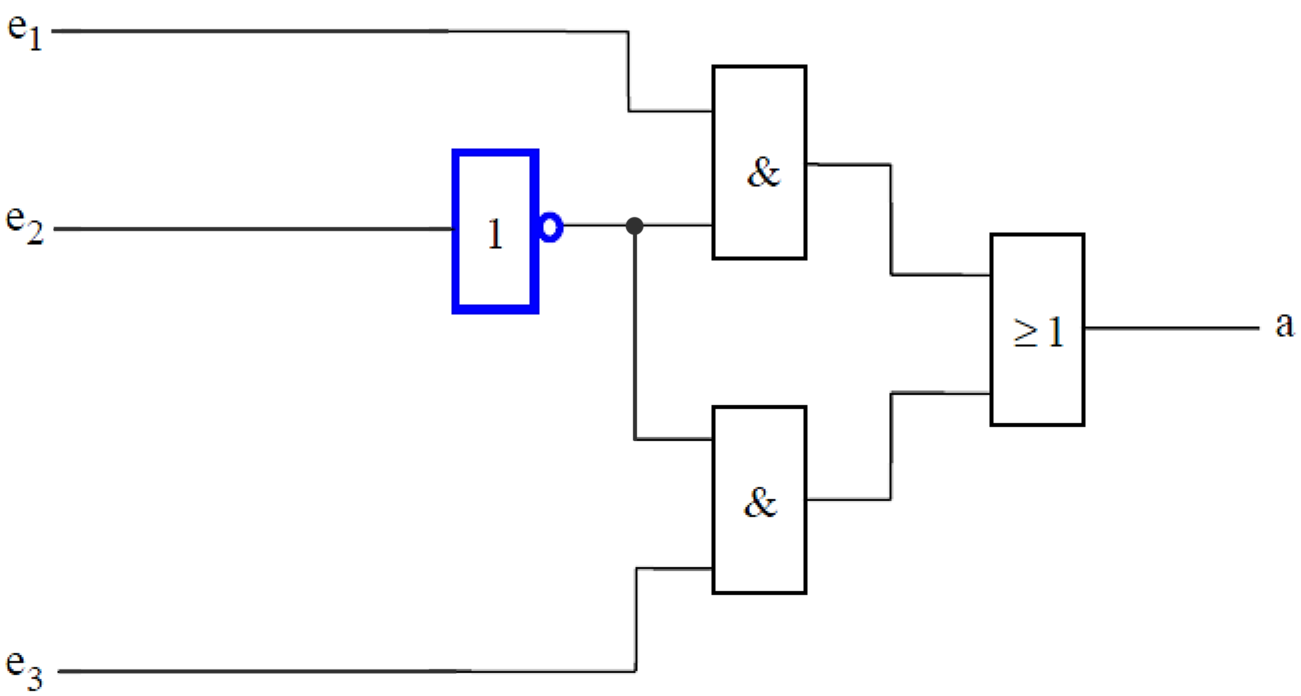
Trennung von Verzögerungs- und Verknüpfungsteil



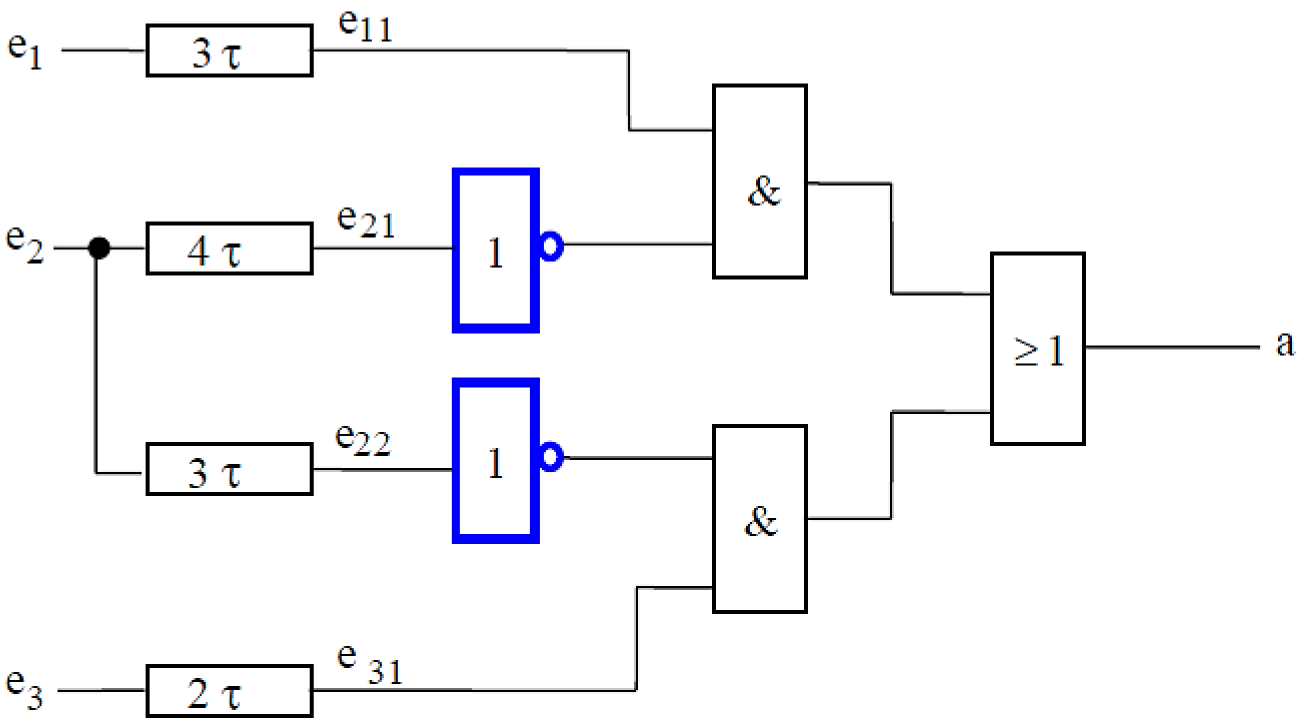
Verzögerungsteil Verknüpfungsteil

Beispiel 2

- Mehrfach genutzte Schaltungsteile werden durch mehrere einfach genutzte äquivalente Kopien ersetzt



Trennung von Verzögerungs- und Verknüpfungsteil



Verzögerungsanteil **Verknüpfungsanteil**

Pfadvektor und Strukturausdruck

Durch die Trennung entsteht immer ein zweigeteiltes Modell:

Verzögerungsteil:

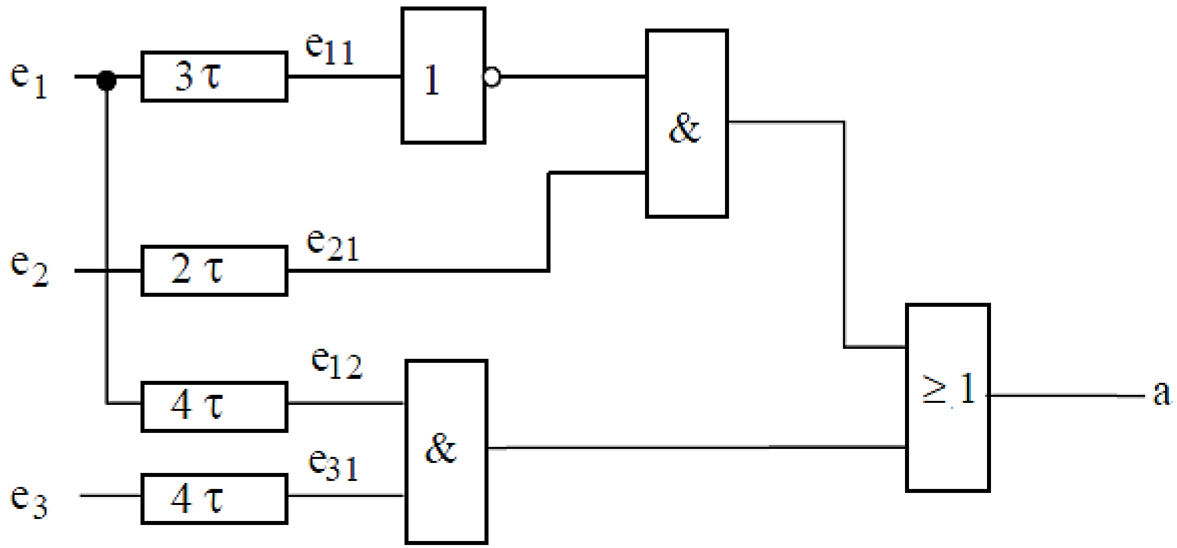
- Die Eingänge spalten sich in alle Signalpfade durch das Schaltnetz auf.
- Die Ausgangsgrößen des Verzögerungsteils heißen **Pfadvariable**
  - **Pfadvariable** werden **doppelt indiziert**.
  - Der **erste Index** kennzeichnet die **Eingangsvariable**, aus der die Pfadvariable stammt.
  - Der **zweite Index** unterscheidet zwischen **von gleichen Eingangsvariablen** stammenden Pfadvariablen.
- Alle Pfadvariablen zusammen bilden den **Pfadvektor**

Pfadvektor und Strukturausdruck

Verknüpfungsteil

- Bildet aus den Pfadvariablen durch ein **ideales Schaltnetz** verzögerungsfrei die Ausgangsgrößen
- **Jede Pfadvariable** kommt nur **einmal** in diesem **Schaltnetz vor**, es gibt **keine Verzweigungen** im Schaltnetz mehr. Das Schaltnetz hat **Baumstruktur**.
- Die algebraische Darstellungsform des Verknüpfungsteils heißt **Strukturausdruck** des Schaltnetzes.
  Der Strukturausdruck beschreibt neben der Funktion auch dessen Struktur.

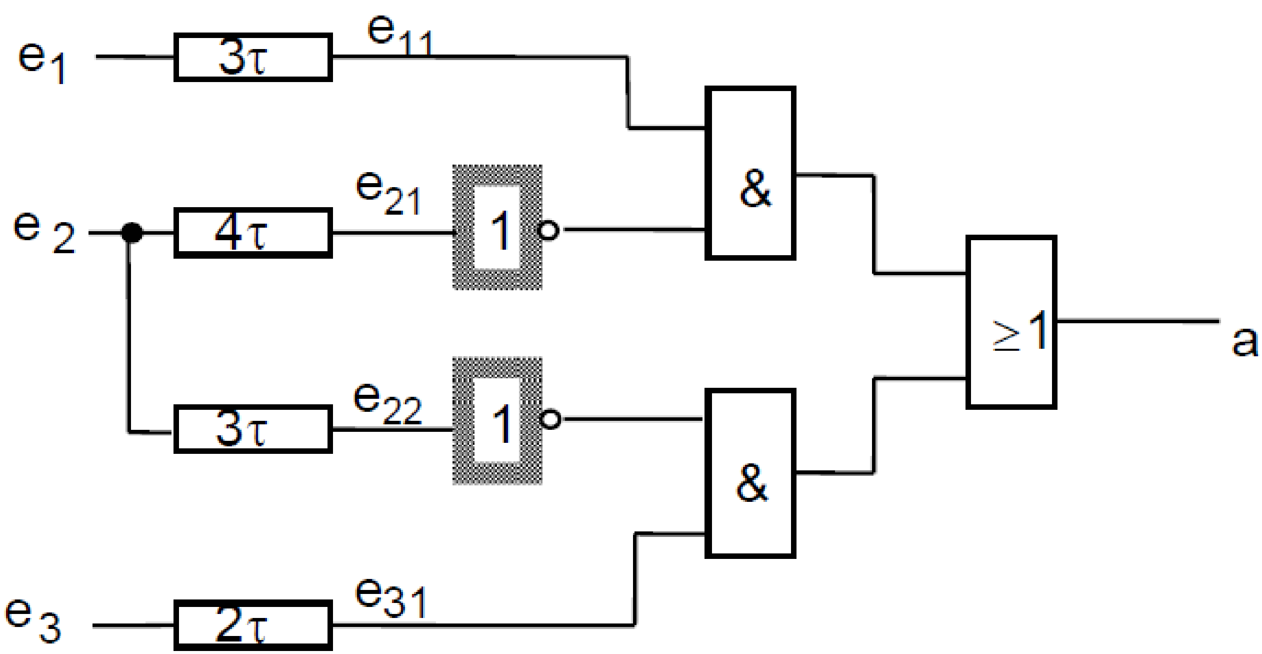
Im Beispiel 1:



Pfadvektor: $< e_{11}, e_{12}, e_{21}, e_{31} >$

Strukturausdruck: $\bar{e}_{11} e_{21} \vee e_{12}\, e_{31}$

Im Beispiel 2:



Pfadvektor: $< e_{11}, e_{21}, e_{22}, e_{31} >$

Strukturausdruck: $e_{11}\bar{e}_{21} \vee \bar{e}_{22}\, e_{31}$

Begriffe: Eingabewechsel, Übergang

Definition: Eingabewechsel

- Ein **Eingabewechsel** ist die Änderung einer oder mehrerer Eingangsvariablen zu einem bestimmten Zeitpunkt.
- Falls sich mehrere Eingangsvariablen ändern sollen, so müssen sie dies gleichzeitig tun.

Definition: Übergang

- Ein **Übergang** ist der Vorgang im Schaltnetz, der von einem Eingabewechsel ausgelöst wird.
  Er beginnt mit dem Eingabewechsel und endet mit dem Eintreten des neuen Ruhezustandes.

Eingabewechsel, Übergang

Jeder Übergang kann mit dem ihn auslösenden Eingabewechsel identifiziert werden.

Wechselt die Eingabe von der Eingabebelegung $\boldsymbol{B_i}$ zur Eingabebelegung $\boldsymbol{B_j}$, so wird sowohl der Eingabewechsel als auch der Übergang mit $\boldsymbol{B_i \rightarrow B_j}$ bezeichnet.

Beispiel:

- (e_3, e_2, e_1): (1,1,0) ⟶ (1,1,1) also $\boldsymbol{B_6 \rightarrow B_7}$

Einschränkungen

- Alle Eingabewechsel sollen nur so stattfinden, dass zwischen zwei aufeinanderfolgenden Eingabewechseln das Schaltnetz zur Ruhe kommt.

  Dies erlaubt die Betrachtung der einzelnen Übergänge unabhängig voneinander.

- Alle Verzögerungen (Schaltzeiten und Laufzeiten) seien endlich.

- Es soll immer nur ein Ausgang betrachtet werden.

Hasardfehler – Hasard

Definition: Hasardfehler

- Ein **Hasardfehler** ist eine mehrmalige Änderung der Ausgangsvariablen während eines Übergangs.

Definition: Hasard

- Ein **Hasard** ist die durch das Schaltnetz gegebene logisch-strukturelle Vorbedingung für einen Hasardfehler, ohne Berücksichtigung der konkreten Verzögerungswerte.

Hasardfehler – Hasard

Jeder Hasard ist eine Eigenschaft eines bestimmten Überganges im Schaltnetz.

Zur Betrachtung, ob ein bestimmter **Übergang hasardbehaftet** ist oder nicht, interessiert nur:

- Die **logische Funktion**, die durch das Schaltnetz realisiert wird.
- Die **Struktur des Schaltnetzes**, d. h. die Anzahl, die Verknüpfungsfunktionen und die genaue Anordnung der Gatter zur Realisierung der Funktion, nicht jedoch die tatsächlichen Verzögerungswerte der verwendeten Gatter.

Hasardbehaftete Übergänge

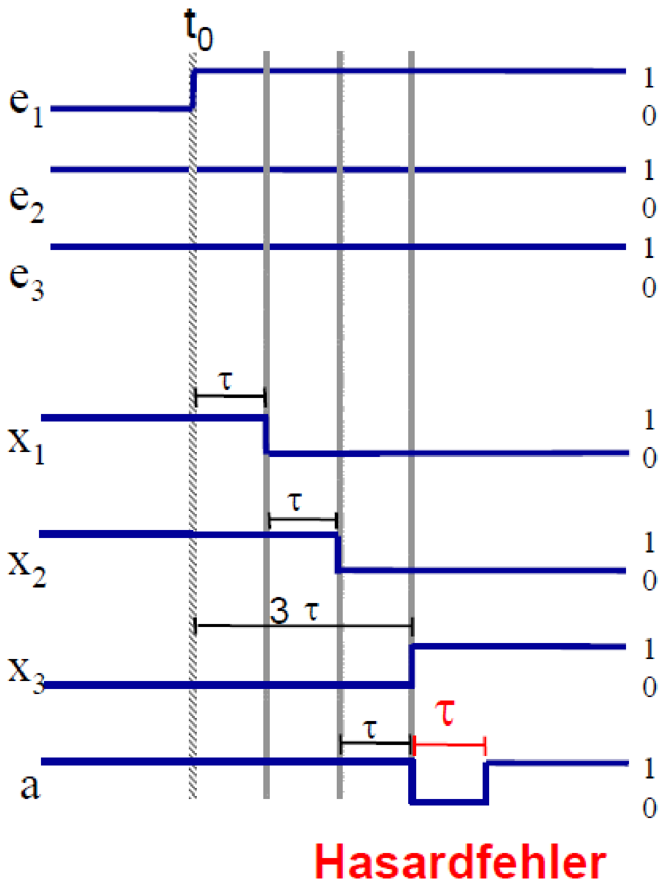
Tritt in einem konkreten Schaltnetz bei einem bestimmten Übergang ein Hasardfehler auf, so ist dieser Übergang hasardbehaftet, also:

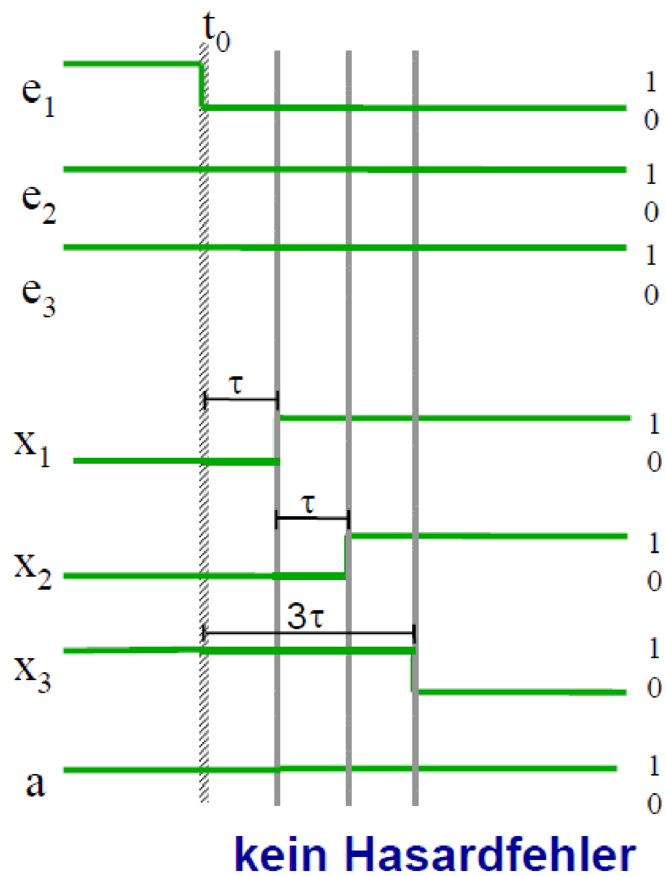
Hasardfehler → Hasard

Die Umkehrung gilt jedoch nicht: Ist ein Übergang hasardbehaftet, so folgt hieraus nicht notwendigerweise das Eintreten eines Hasardfehlers.

Hasard ∧ ungünstige Verzögerungswerte → Hasardfehler

Beispiel 1



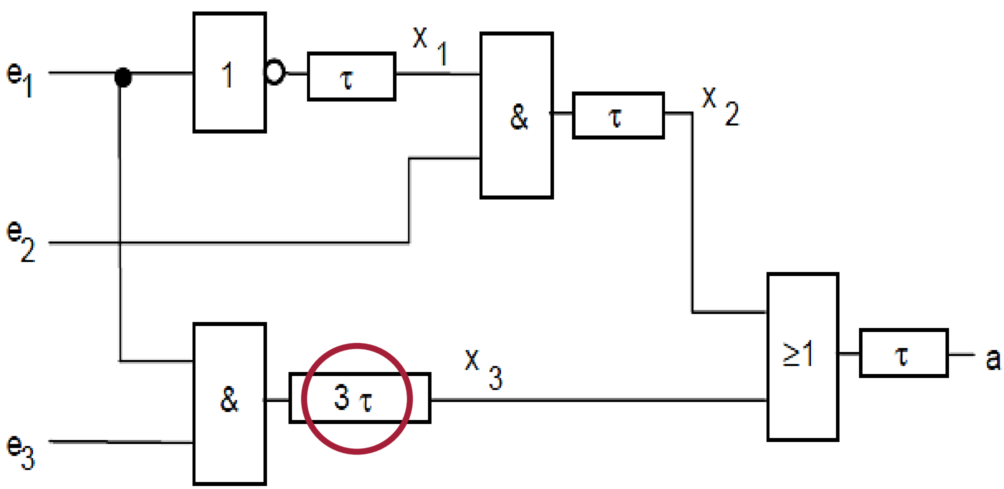


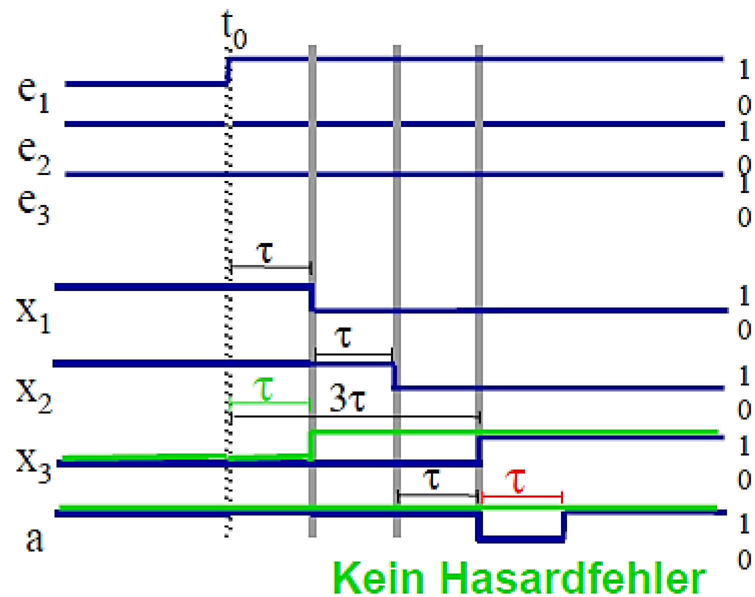
Der Übergang (e_3, e_2, e_1) : (1,1,0) → (1,1,1) ist hasardbehaftet, da es die Möglichkeit zu einem Hasardfehler gibt.

Beispiel 2

Ändert man nun die Totzeit mit dem Wert 3τ auf den Wert τ ab, so entsteht für den Übergang (1,1,0) → (1,1,1) kein Hasardfehler mehr.

Der Übergang ist jedoch nach wie vor hasardbehaftet, da für den Hasard konkrete Verzögerungswerte nicht interessieren.





Statischer Übergang

Statischer Übergang:

Ein Übergang, bei dem Anfangs- und Endwert des Ausgangssignals gleich sind (unabhängig davon, ob ein Hasardfehler eintritt oder nicht).

- **Statischen 0-Übergang**: Anfangs- und Endwert des Ausgangssignal sind beide 0
- **Statischen 1-Übergang:** Anfangs- und Endwert des Ausgangssignal sind beide 1

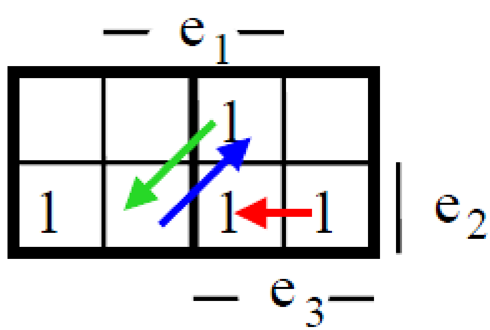
Dynamischer Übergang

Dynamischer Übergang:

Ein Übergang, bei dem Anfangs- und Endwert des Ausgangssignals verschieden sind

- **Dynamischen 01-Übergang:** Anfangswert des Ausgangssignals 0, der Endwert 1
- **Dynamischen 10-Übergang**: Anfangswert des Ausgangssignals 1, der Endwert 0

Beispiel



Statischer 1-Übergang:

Übergang (e_3, e_2, e_1): (1,1,0) → (1,1,1)

Dynamischer 01-Übergang:

Übergang (e_3, e_2, e_1): (0,1,1) → (1,0,1)

Dynamischer 10-Übergang

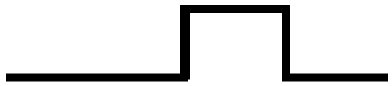
Übergang in umgekehrter Richtung: (1,0,1) → (0,1,1)

Statischer 0-Hasard

Analog zu den Übergängen werden die **Hasards** als **statisch** bzw. **dynamisch** bezeichnet, je nachdem, bei welcher Art von Übergang sie auftreten.

Ein Hasard in einem statischen 0-Übergang heißt **statischer 0-Hasard**.

Beispiele für statische 0-Hasardfehler:



Statischer 1-Hasard

Ein Hasard in einem statischen 1-Übergang heißt **statischer 1-Hasard**.

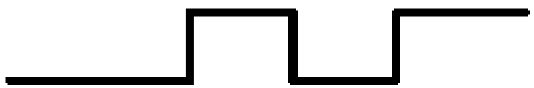
Beispiele für statische 1-Hasardfehler:

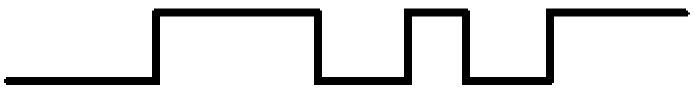
Der Übergang (1,1,0) → (1,1,1) im Beispiel enthält also einen statischen 1-Hasard.

Dynamischer 01-Hasard

Ein Hasard in einem dynamischen 01-Übergang heißt **dynamischer 01-Hasard**.

Beispiele für dynamische 01-Hasardfehler:

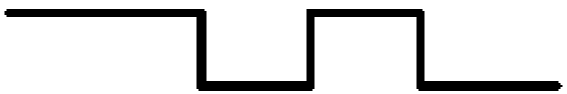


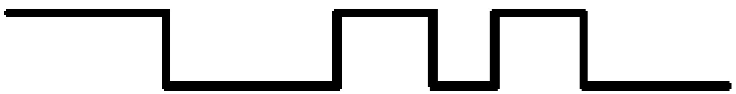


Dynamischer 10-Hasard

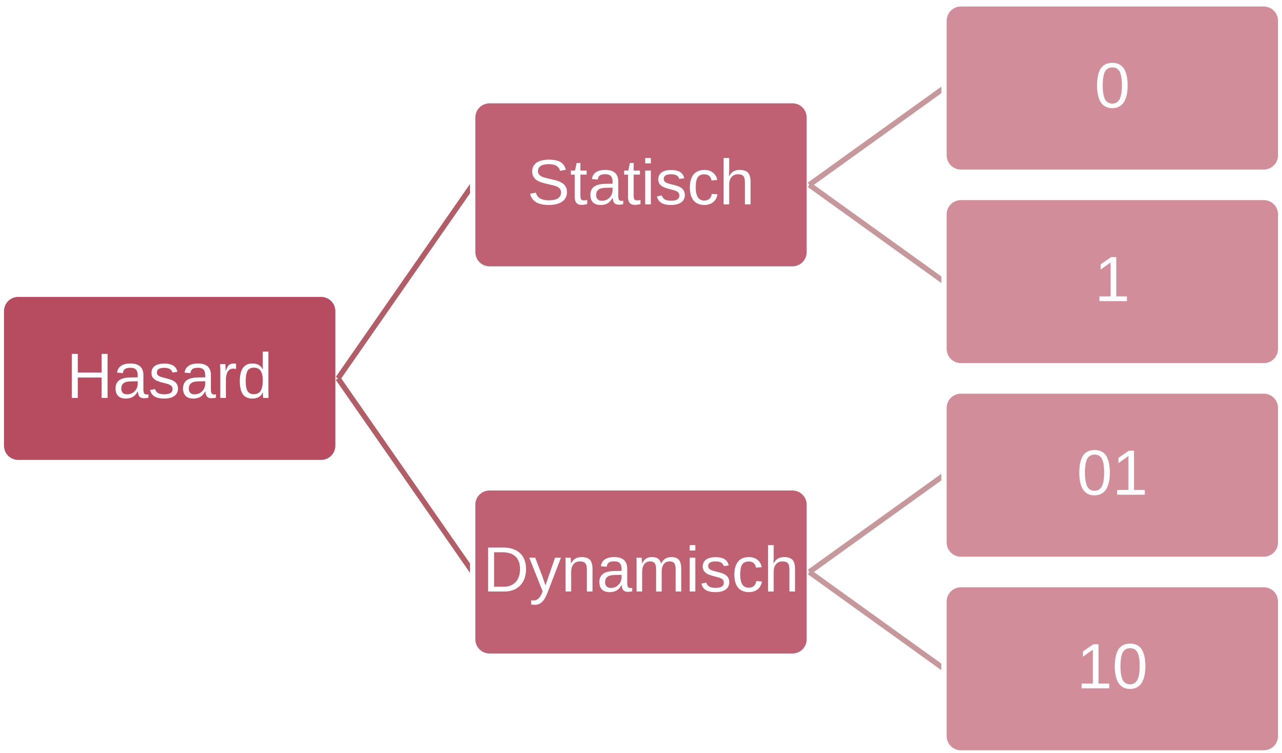
Ein Hasard in einem dynamischen 10-Übergang heißt **dynamischer 10-Hasard**.

Beispiele für dynamische 10-Hasardfehler:





Klassifikation von Hasards



Funktionshasard – Strukturhasard

Weitere Unterscheidung von Hasards nach ihrer Ursache:
Funktionshasards und **Strukturhasards.**

- Bei einem **Funktionshasard** liegt die Ursache in der zu realisierenden Funktion selbst.

- Bei einem **Strukturhasard** dagegen liegt die Ursache in der Struktur des realisierten Schaltnetzes.

Beseitigung von Funktionshasard

- Ein Funktionshasard **tritt in jedem möglichen Schaltnetz** zur Realisierung dieser Schaltfunktion auf und kann **nicht allgemeingültig behoben werden**.

- Für ein konkretes Schaltnetz mit Funktionshasard kann zwar der Funktionshasardfehler durch günstige Wahl der Verzögerungswerte behoben werden, nicht jedoch der Hasard selbst.

Strukturhasard

- Da die Ursache in der Struktur des realisierten Schaltnetzes liegt, kann ein **Strukturhasard immer durch Änderung der Schaltnetzstruktur** bei gleicher Schaltnetzfunktion **behoben werden**.

- Es ist grundsätzlich möglich, ein anderes Schaltnetz zu entwerfen, welches dieselbe Funktion realisiert und den Strukturhasard beseitigt.

Klassifikation von Laufzeiteffekten

	Funktion des Schaltnetzes	Struktur des Schaltnetzes	Laufzeiten der Gatter (und Leitungen)
Funktions-hasards			
Struktur-hasards			
Funktions-hasardfehler			
Struktur-hasardfehler			

Die dunkelroten Felder markieren die notwendigen Informationen
Die hellroten Felder markieren die daraus folgenden Informationen

Analyse von Hasards

Methode: Alle möglichen Wege eines Übergangs werden im KV-Diagramm untersucht

Erkennen eines Funktionshasards

Vorgehensweise:

- Man stellt das KV-Diagramm der Funktion auf und **markiert** das **Feld** der **Anfangsbelegung** und das der **Endbelegung** des **betrachteten Eingabewechsels**.
- Nun betrachtet man **alle Wege**, die **der Übergang nehmen kann**. **Ausgangspunkt:** **Welche Eingangsvariablen** sind am Eingabewechsel **beteiligt**.

Erkennen von Funktionshasards

- Nacheinander muss dann jeweils **genau einmal** an den **Achsen** der **beteiligten Eingangsvariablen gespiegelt** werden, um einen Weg von der Anfangsbelegung bis zur Endbelegung zu erhalten.

- Mit ***m*** Eingangsvariablen, die am Wechsel beteiligt sind, gibt es ***m*!** mögliche Wege (→ Jede mögliche Reihenfolge der ***m*** Spiegelungen muss berücksichtigt werden.)

- Der Übergang ist genau dann **funktionshasardbehaftet**, wenn es **mindestens einen Weg** gibt, für den die Folge der zugehörigen Funktionswerte (0 oder 1 im KV-Diagramm) **nicht monoton** ist, d.h. in der mindestens zweimal der Funktionswert wechselt.

Erkennen von Funktionshasards

Beispiele **monotoner** Folgen von Funktionswerten:

$0 \rightarrow 0 \rightarrow 0 \rightarrow 0, \quad 0 \rightarrow 0 \rightarrow 1 \rightarrow 1,$

$1 \rightarrow 1 \rightarrow 1 \rightarrow 0, \quad 1 \rightarrow 1 \rightarrow 1 \rightarrow 1.$

Beispiele **nicht monotoner** Folgen von Funktionswerten:

$0 \rightarrow 0 \rightarrow 1 \rightarrow 0, \quad 0 \rightarrow 1 \rightarrow 0 \rightarrow 1,$

$1 \rightarrow 0 \rightarrow 1 \rightarrow 0, \quad 1 \rightarrow 0 \rightarrow 0 \rightarrow 1.$

Erkennen von Funktionshasards

- Es ist leicht einzusehen, dass **jeder ermittelte Weg** eine **mögliche Übergangsfolge** im Schaltnetz darstellt.

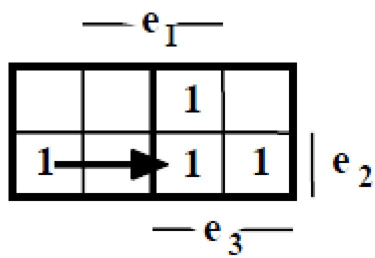
- **Welcher Weg eingeschlagen** wird, ist von den **Verzögerungszeiten abhängig**

- **Existiert mindestens** ein **nichtmonotoner Weg**, so kann dieser bei ungünstigen Verzögerungswerten auch eingeschlagen werden

→ **Bereitschaft zum Hasardfehler**

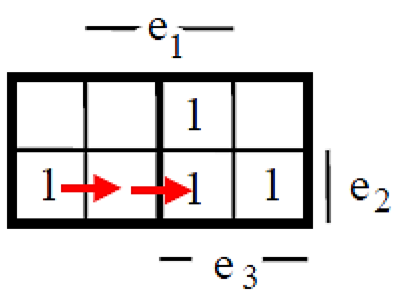
Beispiel

Übergang (e_3, e_2, e_1): (0,1,0) → (1,1,1), im KV-Diagramm:



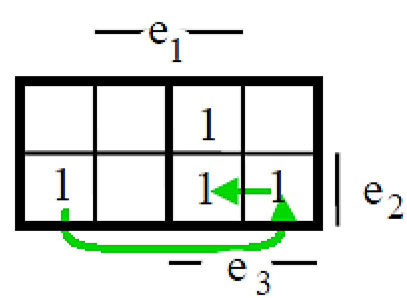
Hier sind die Variablen e_1 und e_3 am Übergang beteiligt.

Es gibt 2 Wege:



$(0,1,0) \rightarrow (0,1,1) \rightarrow (1,1,1)$

$\mathbf{1} \rightarrow \mathbf{0} \rightarrow \mathbf{1}$



$(0,1,0) \rightarrow (1,1,0) \rightarrow (1,1,1)$

$\mathbf{1} \rightarrow \mathbf{1} \rightarrow \mathbf{1}$

Erklärung zum Beispiel

- Der erste dieser Wege liefert als Folge der Funktionswerte die nicht monotone Folge **1** → **0** → **1**.
  → **Der Übergang ist mit einem Funktionshasard behaftet.**

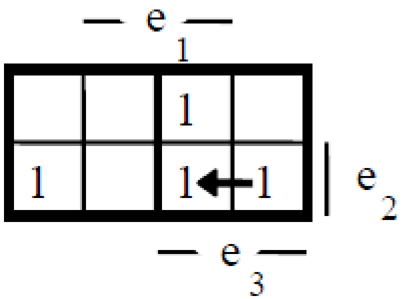
- Bereits eine nicht monotone Folge ist dafür hinreichend.

- Dass der zweite Weg eine monotone Folge liefert, spielt keine Rolle mehr.

Es handelt sich um einen **statischen 1-Funktionshasard.**

Beispiel 2

Übergang (e_3, e_2, e_1): (1,1,0) → (1,1,1) im KV-Diagramm:



Es ändert sich nur die Variable e_1. Es gibt daher nur einen Weg, der aus Anfangs- und Endwert des Übergangs besteht.

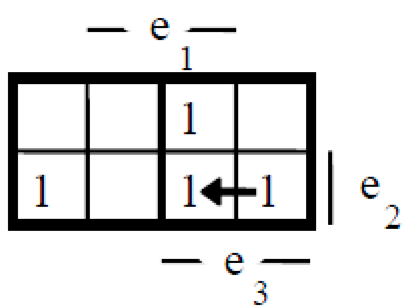
Die Folge der dazugehörigen Funktionswerte muss monoton sein.
Die Folge heißt hier 1→1.

→ **Dieser Übergang enthält also keinen Funktionshasard.**

Ein Übergang, bei welchem nur eine Eingangsvariable wechselt, kann keinen Funktionshasard enthalten.

Beispiel 2

Übergang (e_3, e_2, e_1): (1,1,0) → (1,1,1)



Der Übergang enthält keinen Funktionshasard.

Da wir jedoch von früher wissen, dass der Übergang hasardbehaftet ist, muss es sich hierbei um einen **Strukturhasard** handeln.

Erkennen eines Strukturhasards

- Ähnlich wie bei Funktionshasards, sind **Strukturhasards** mit Hilfe des **KV-Diagramms erkennbar**.

- Da ein **Strukturhasard** eine Eigenschaft eines Übergangs in einem Schaltnetz bestimmter Struktur ist, **muss** hier **diese Struktur** in die **Überlegungen einbezogen werden**.

- Die Struktur des Schaltnetzes liefert Information darüber, über wie viele und welche Pfade sich Änderungen an den Eingangsvariablen auf den Ausgang auswirken.
  $\Rightarrow$ **Ausgangspunkt ist der Strukturausdruck des Schaltnetzes**

Erkennen eines Strukturhasards

- Stelle für den **Strukturausdruck** des Schaltnetzes das **KV-Diagramm** auf.
  Anstelle der Eingangsvariablen treten hier die Pfadvariablen.

- **Markiere** wieder die **beiden Felder der Anfangs- und der Endbelegung.**
  In diesen **müssen die Werte aller Pfadvariablen einer Eingangsvariablen gleich** sein.

- **Betrachte alle möglichen Wege von der Anfangs- zur Endbelegung.**

Erkennen eines Strukturhasards

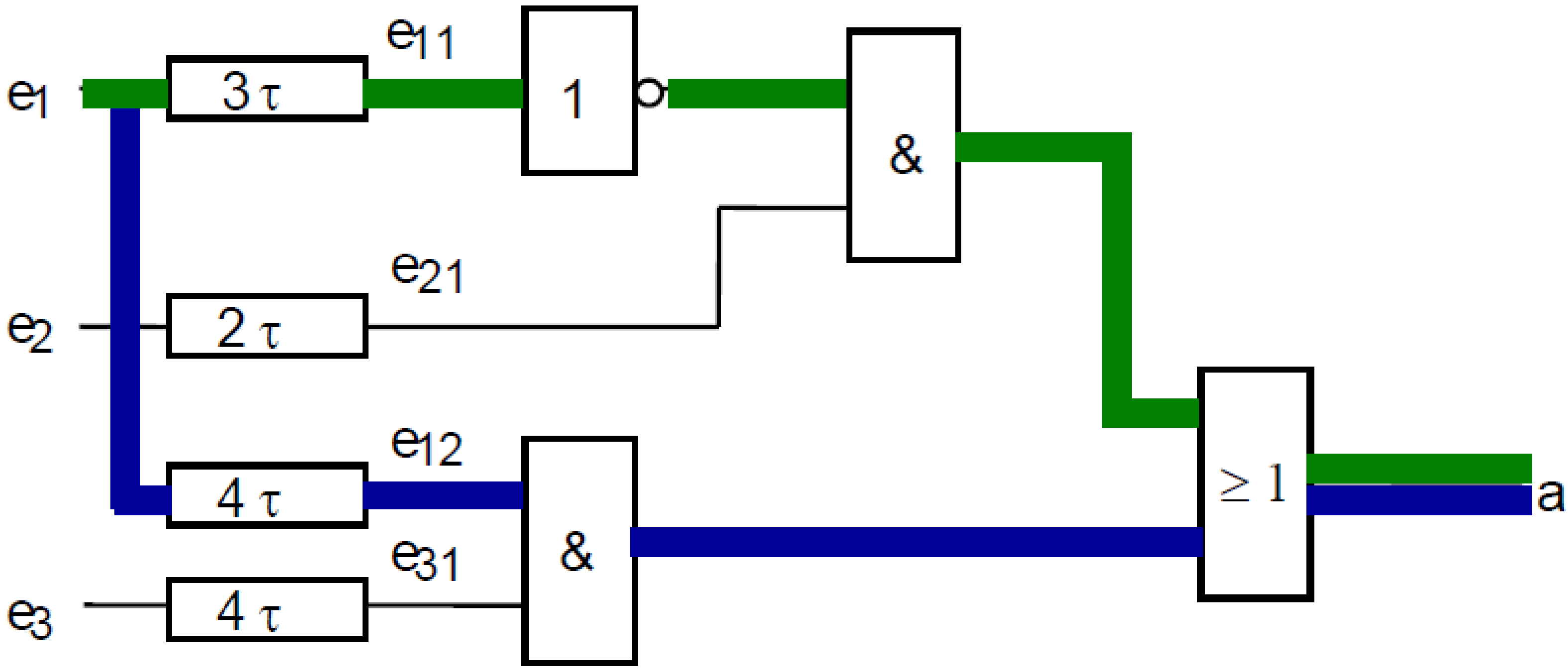
Der Übergang ist genau dann **strukturhasardbehaftet**, wenn es **mindestens einen Weg** gibt, für den die Folge der zugehörigen Werte des Strukturausdrucks **nicht monoton** ist.

D.h. der **Wert** des **Strukturausdrucks ändert** sich bei diesem **Weg mindestens zweimal**.

Beispiel

Schaltfunktion: $a = \bar{e}_1 e_2 \lor e_1 e_3$

Totzeitmodell:



Strukturausdruck: $a = \bar{e}_{11} e_{21} \lor e_{12} e_{31}$

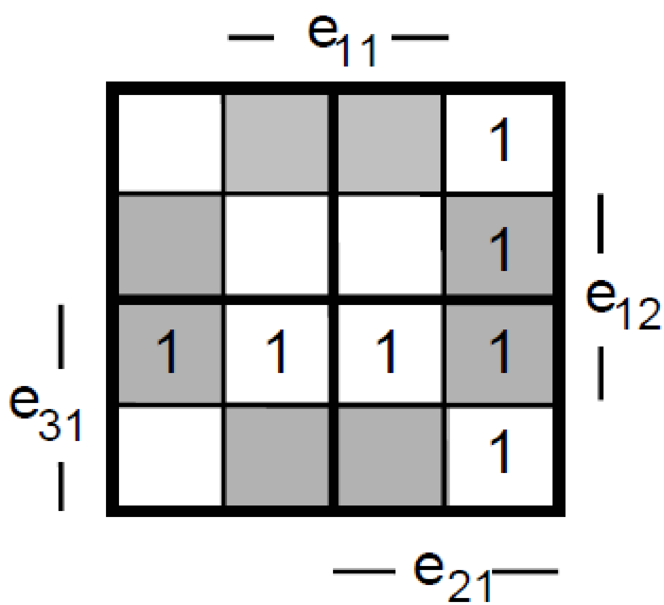
Beispiel

Man sieht:

- Änderungen von e_2 bzw. e_3 breiten sich **nur über einen Pfad** aus.
- Änderungen von e_1 breiten sich hingegen **über zwei Pfade** (blau und grün hervorgehoben) aus.

- Es sind e_{11}, e_{21}, e_{12} und e_{31} die neuen Pfadvariablen, wobei jede genau einem Pfad entspricht.

Das strukturspezifische KV-Diagramm



Diejenigen Felder, die nur während eines Überganges eingenommen werden können, wenn $e_{11} \neq e_{12}$, sind grau unterlegt.

Weiß verbleiben immer genau so viele Felder, wie das KV-Diagramm enthält, welches die Funktion darstellt.

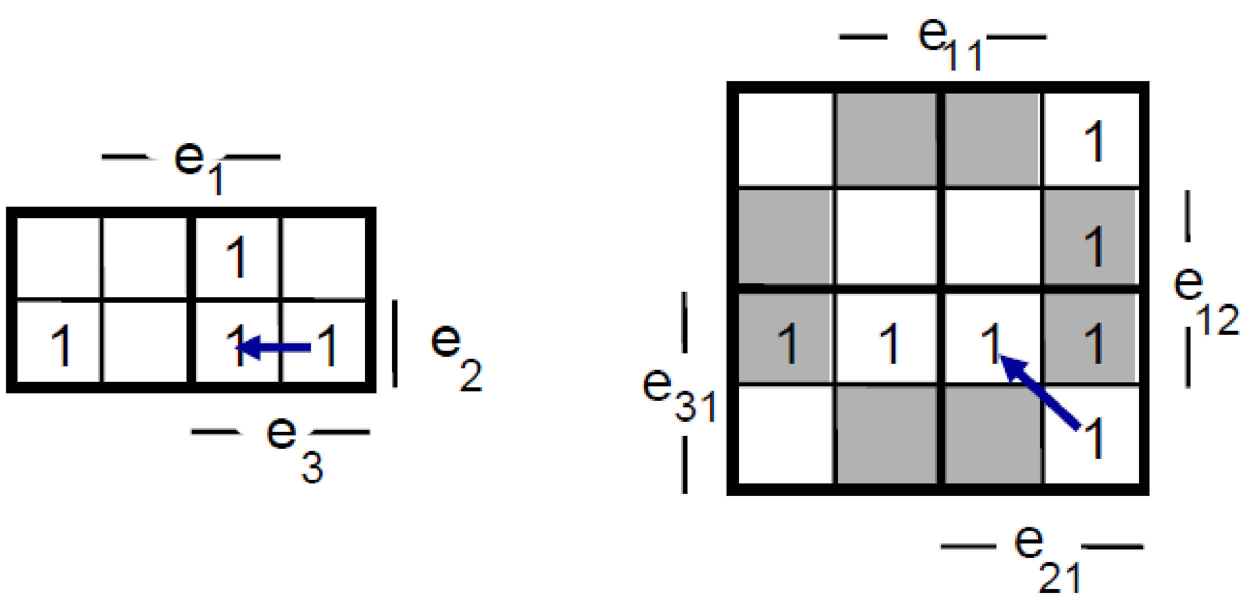
Das strukturspezifische KV-Diagramm

Betrachte den Übergang (e_3, e_2, e_1) : (1,1,0) → (1,1,1)

Mit Hilfe der Pfadvariablen ausgedrückt lautet dieser Übergang:

$$(e_{31}, e_{21}, e_{12}, e_{11}) : (1,1,0,0) \rightarrow (1,1,1,1).$$

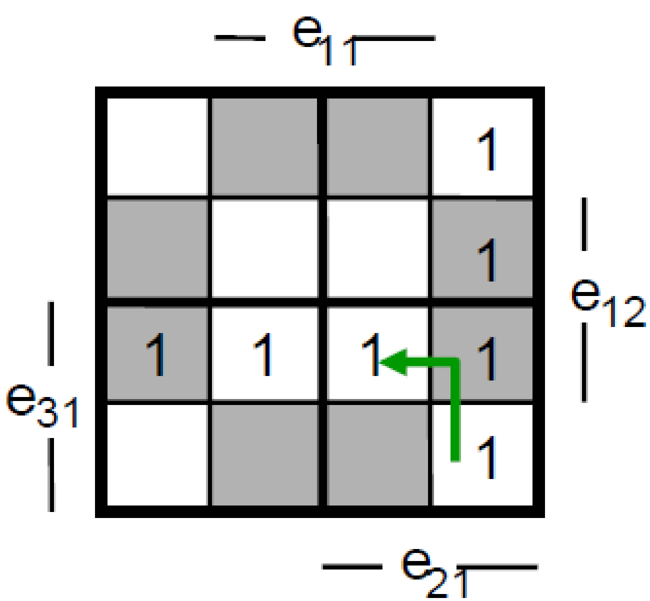
Im KV-Diagramm wird der Übergang eingezeichnet:



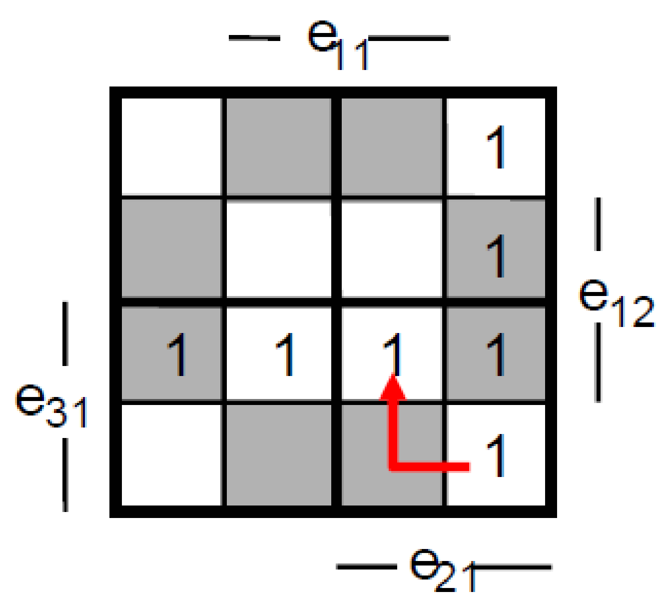
Das strukturspezifische KV-Diagramm

Hier kann man die schon bei den Funktionshasards erprobte Methode zur Hasardbestimmung anwenden:

Es sind zwei mögliche Wege zu erkennen, die eingeschlagen werden können:



Kein Hasardfehler!

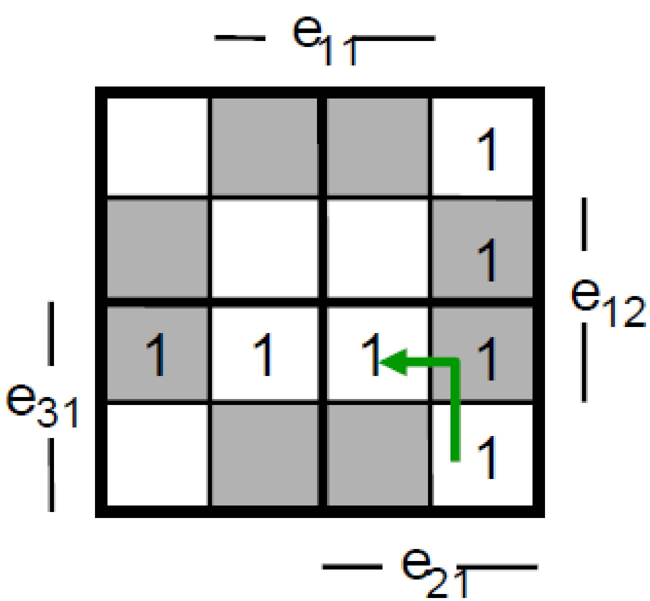


Hasardfehler!

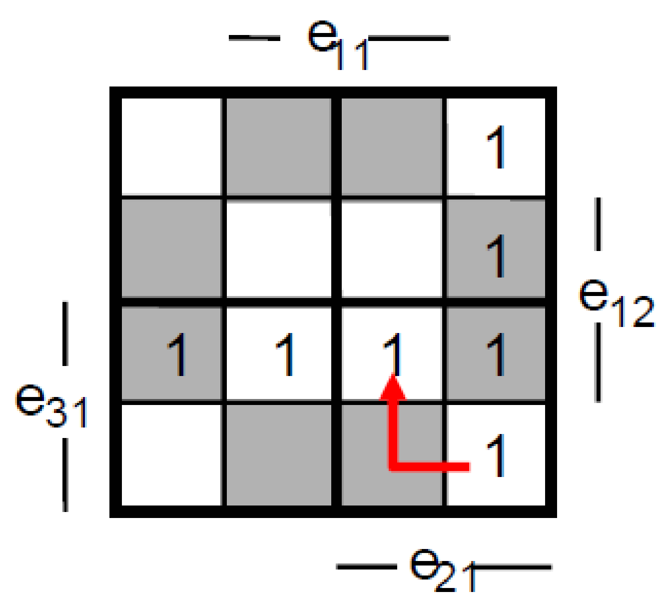
Das strukturspezifische KV-Diagramm

Wird der zweite Weg eingeschlagen, so erhält man einen statischen 1-Hasardfehler.

→ **Dieser Übergang ist hasardbehaftet.**
(wie schon früher festgestellt)



Kein Hasardfehler!

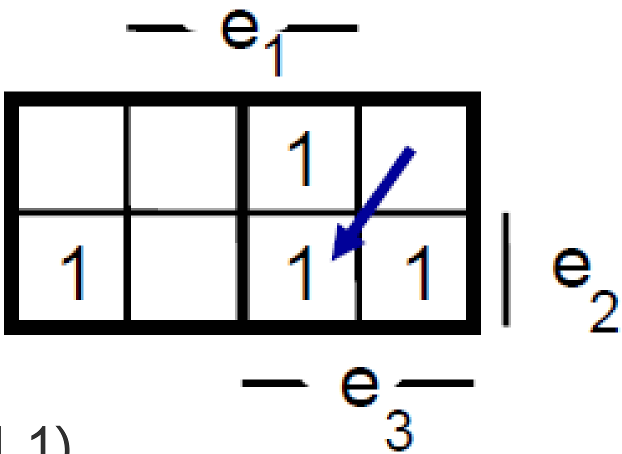


Hasardfehler!

Anderer Übergang im gleichen Schaltnetz

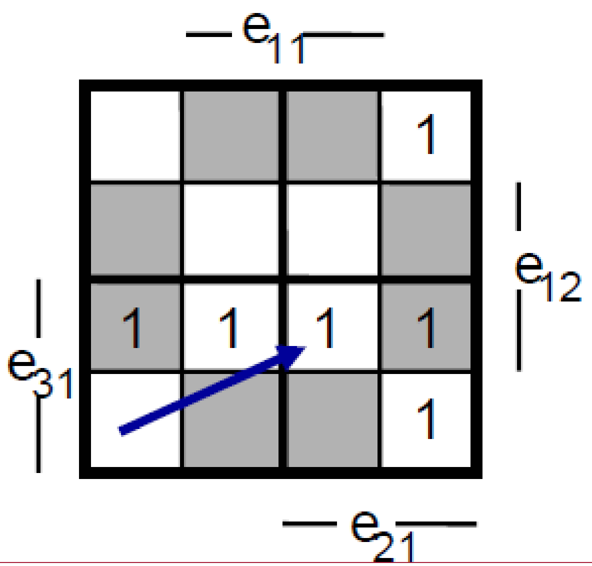
- Betrachtet wird der Übergang:

  (e_3, e_2, e_1): $(1,0,0) \rightarrow (1,1,1)$



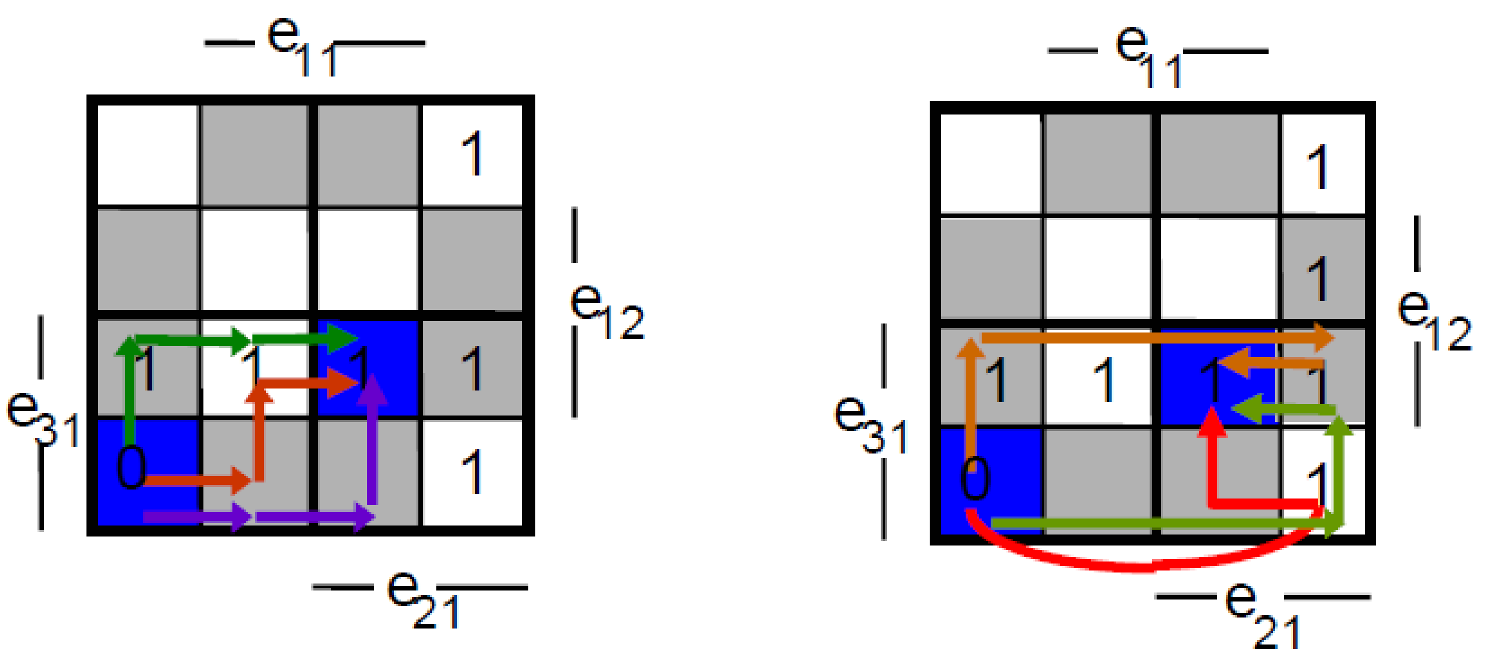
- Übergang in den Pfadvariablen:

  $(e_{31}, e_{21}, e_{12}, e_{11})$: $(1,0,0,0) \rightarrow (1,1,1,1)$



Anderer Übergang im gleichen Schaltnetz

- $(e_{31}, e_{21}, e_{12}, e_{11})$: (1,1,0,0) → (1,1,1,1)

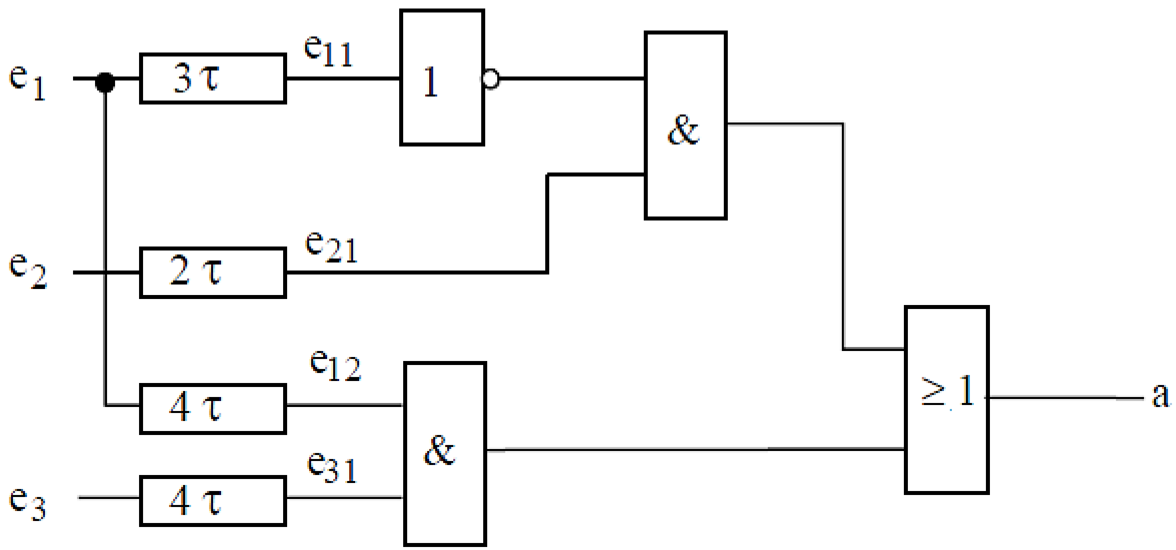


- Insgesamt 6 Wege, wovon **einer** eine nicht monotone Ausgangsfolge liefert

→ Der Übergang ist mit einem dynamischen 01-Strukturhasard behaftet.

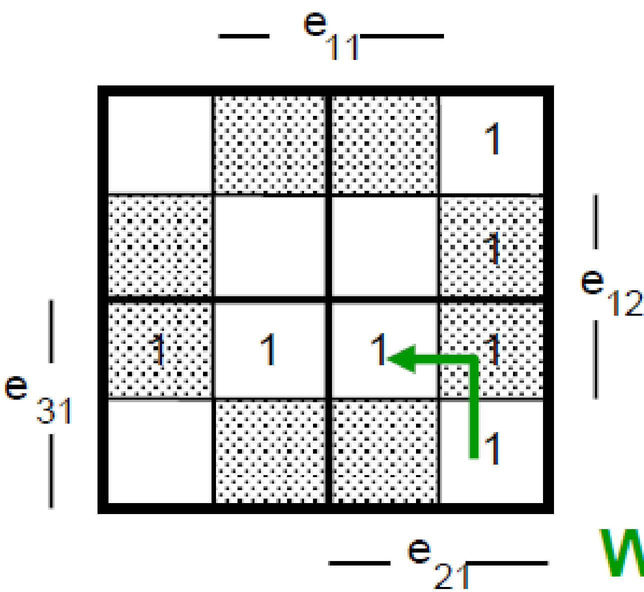
Zeitbedingungen für Hasardfehler

- Mit Hilfe des Totzeitmodells und des KV-Diagramms des Strukturausdrucks sind die Zeitbedingungen für das Auftreten des Hasardfehlers leicht zu ermitteln.
- **Beispiel:**

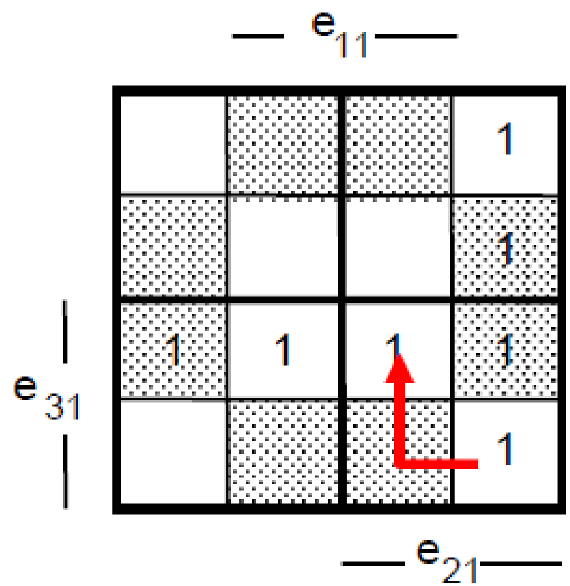


- **Übergang:** (e_3, e_2, e_1): (1,1,0) → (1,1,1)

  $(e_{31}, e_{21}, e_{12}, e_{11})$: (1,1,0,0) → (1,1,1,1)



Weg 1



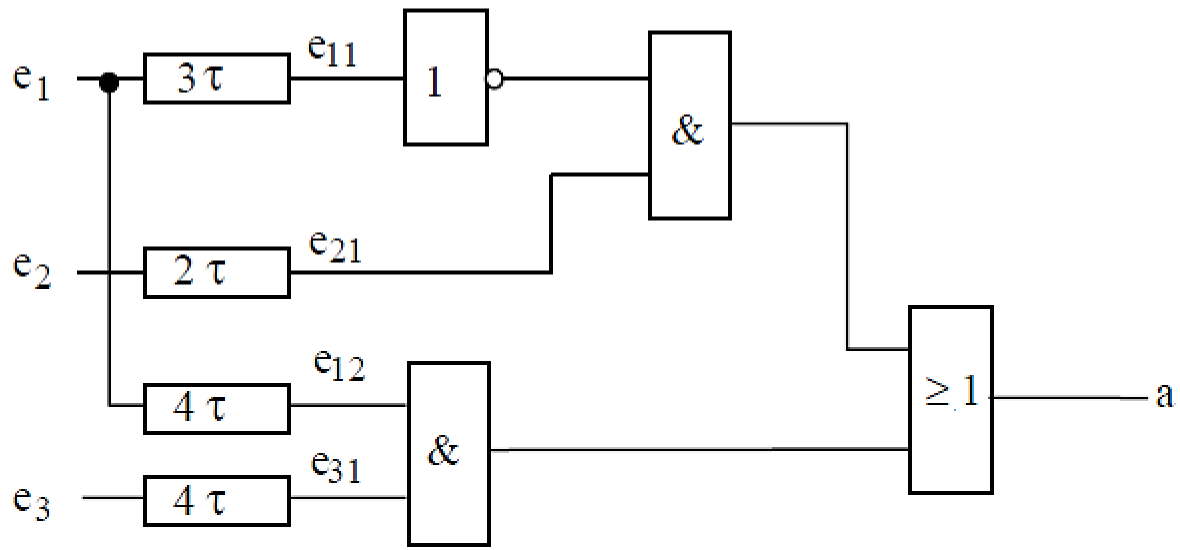
Weg 2

- Es existieren 2 Wege. Weg 2 erzeugt einen Hasardfehler.
- Dieser Weg wird eingeschlagen, wenn sich erst e_{11} und dann e_{12} ändert

→ Der Hasardfehler tritt auf, wenn die Totzeit in Pfad e_{11} kleiner als die Totzeit in Pfad e_{12} ist

Zeitbedingungen für Hasardfehler

- Der Hasardfehler tritt auf, wenn die Totzeit in Pfad e_{11} kleiner als die Totzeit in Pfad e_{12} ist.



- Aus den Totzeitmodell folgt: Totzeit e_{11}: 3τ
  Totzeit e_{12}: 4τ

→ **Der Hasardfehler tritt immer auf!**

Beheben von Hasards

- **Funktionshasards:**

  Funktionshasards können nicht behoben werden, da sie in der Schaltfunktion begründet sind.

  - Jede Behebungsmaßnahme würde zu einer Änderung der Schaltfunktion führen.
  - Einzige Möglichkeit:
    Vermeidung des Hasardfehlers durch die Wahl geeigneter Verzögerungszeiten.

- **Strukturhasards**

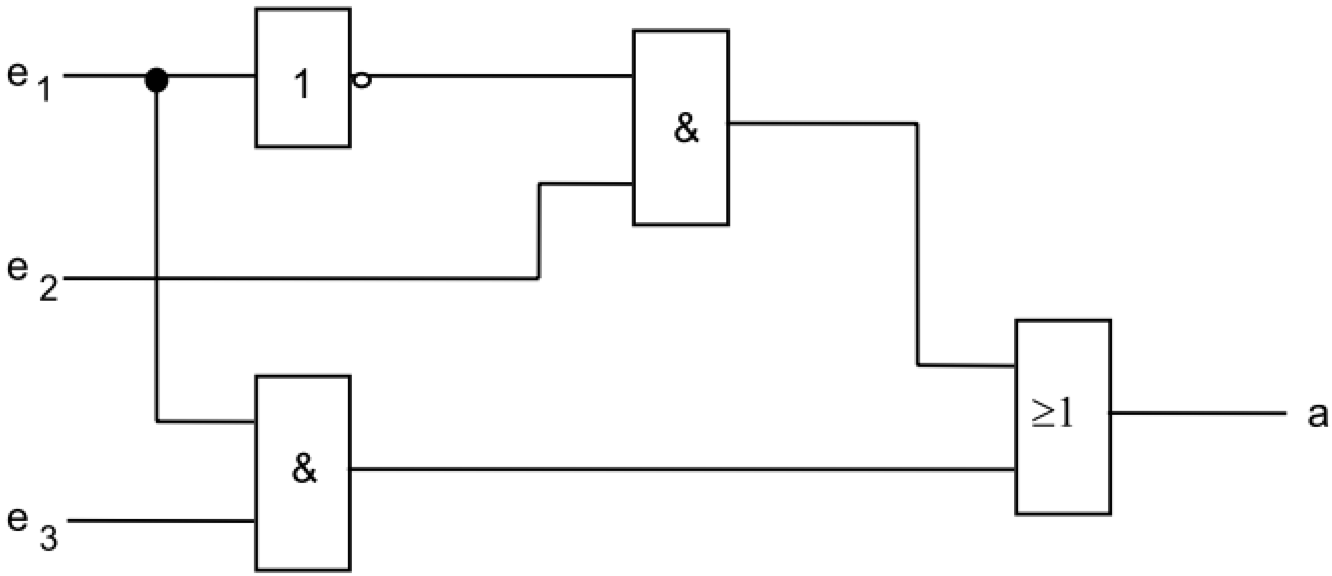
  Strukturhasards können durch geeignete Veränderung der Struktur bei gleicher Schaltfunktion behoben werden.

  - Behebung statischer Strukturhasards:
  - Prinzip:

    Die beim Übergang konstant bleibenden Eingangsvariablen müssen in die Lage versetzt werden, das Ausgangssignal konstant zu halten.

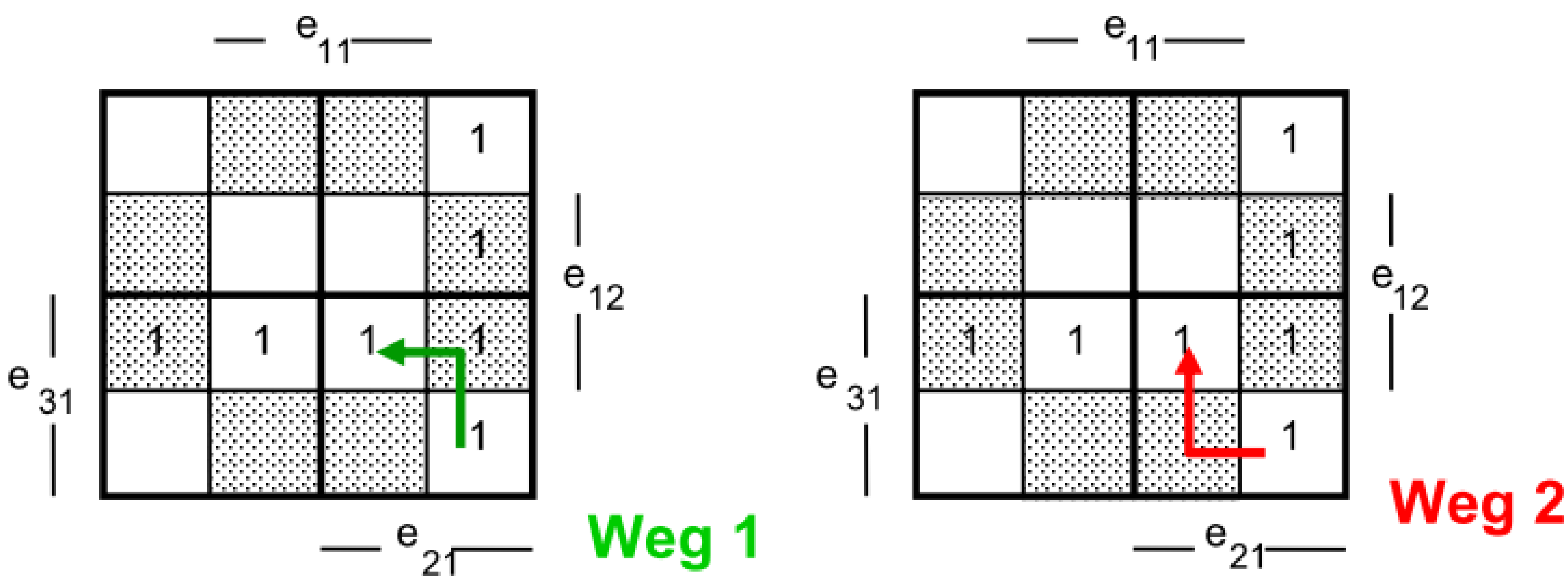
Behebung statischer 1-Strukturhasards

- **Gegeben:**
  - Ein funktionshasardfreier, aber strukturhasardbehafteter 1-Übergang in einem Schaltnetz
- **Beispiel** (bereits bekanntes Schaltnetz):



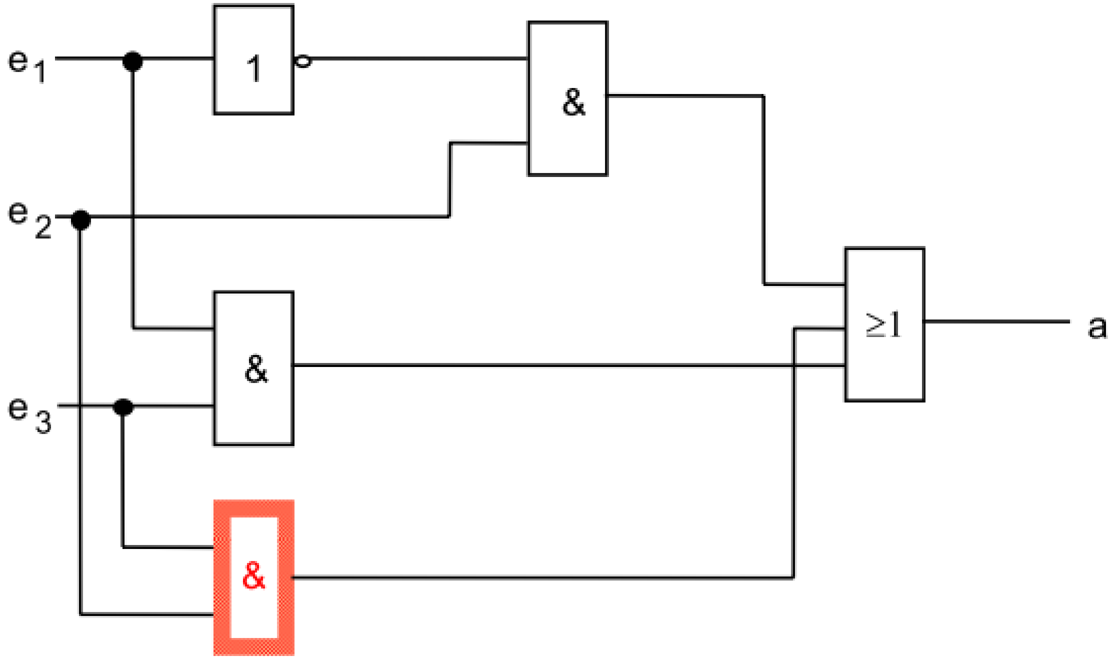
Behebung statischer 1-Strukturhasards

- Übergang: $(e_3, e_2, e_1) : (1,1,0) \rightarrow (1,1,1)$
  - Dieser Übergang enthält, wie wir bereits wissen, einen statischen 1-Strukturhasard.



Behebung statischer 1-Strukturhasards

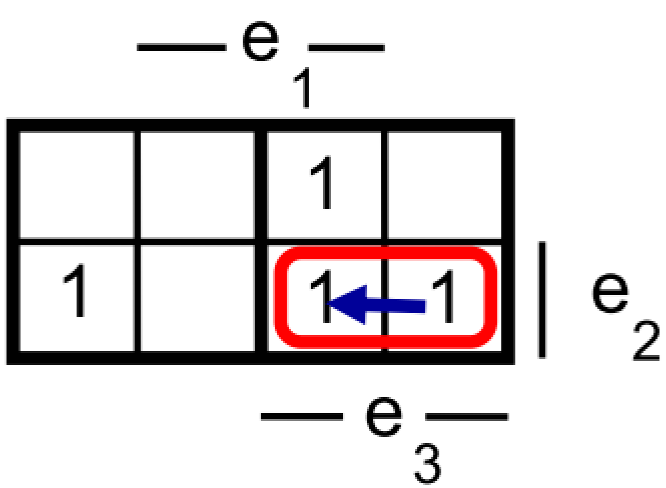
- Die Variablen e_2 und e_3 bleiben konstant 1
  → **Zusätzliches UND-Glied $e_2 e_3$**



- Das zusätzliche UND-Glied hält den Ausgang während des Übergangs auf konstant 1

Behebung statischer 1-Strukturhasards

- Wir betrachten die Situation im KV-Diagramm:
  - Das zusätzlich realisierte UND-Glied wird im KV-Diagramm durch einen Einsblock e_2e_3 repräsentiert, der sowohl die Anfangs- als auch die Endbelegung des Übergangs enthält:

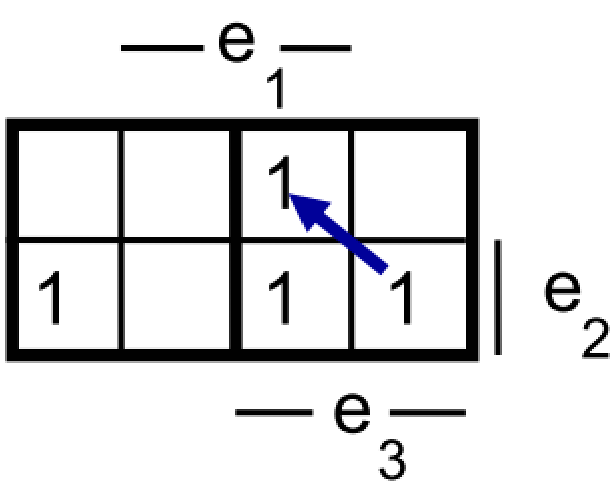


Behebung statischer 1-Strukturhasards

- Ein solcher Einsblock kann **immer** für einen 1-Strukturhasard konstruiert werden, da anderenfalls mindestens eine 0 im Übergangsbereich vorhanden ist

  → **Nicht behebbarer Funktionshasard**

- Beispiel:



Behebung statischer 0-Strukturhasards

- Kosten dieser Art von Hasardbehebung ist umso günstiger, je größer der zusätzlich realisierte Einsblock ist.

  - am günstigsten ist somit ein **Prim-Einsblock**, der sowohl die Anfangs- als auch die Endbelegung umschließt.

  - **Allgemeine Regel zur Beseitigung eines 1-Strukturhasards**

    Man realisiere mit einem UND-Glied einen Primimplikanten, der sowohl Anfangs- wie auch Endbelegung des Übergangs enthält und verknüpfe den UND-Ausgang disjunktiv mit dem Schaltnetz-Ausgang.

Behebung statischer 0-Strukturhasards

- Auf gleiche Weise kann mit Nullblöcken vorgegangen werden:
  - **Allgemeine Regel zur Beseitigung eines 0-Strukturhasards**

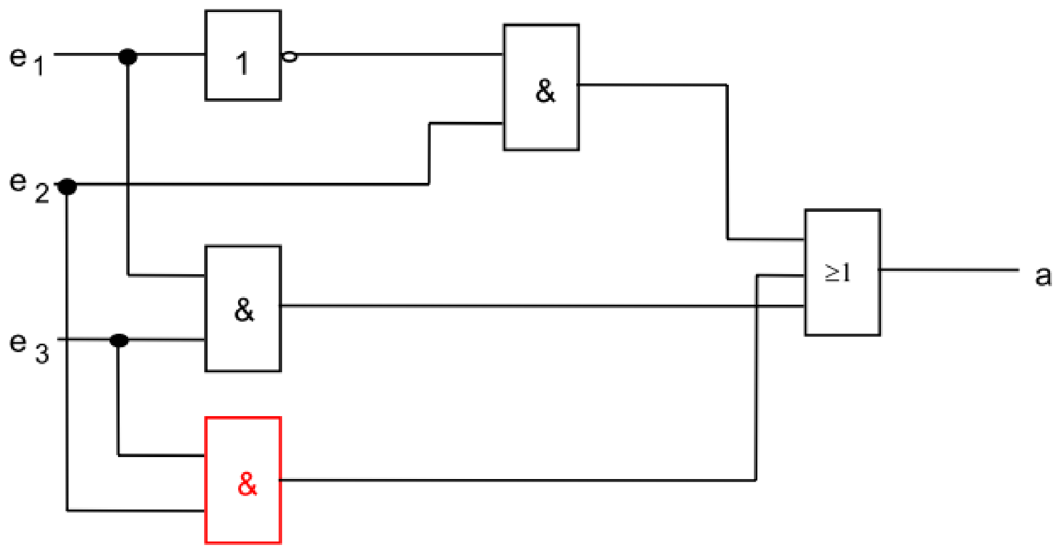
    Man realisiere mit einem ODER-Glied ein Primimplikat, der sowohl Anfangs- wie auch Endbelegung des Übergangs enthält und verknüpfe den ODER-Ausgang konjunktiv mit dem Schaltnetz-Ausgang.

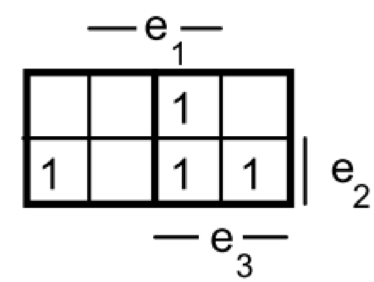
Satz von Eichelberger

- Verallgemeinerung der Regeln:
- Ein Schaltnetz, das durch die Disjunktion aller Primimplikanten oder die Konjunktion aller Primimplikate einer gegebenen Funktion realisiert wird, ist frei von
  - allen statischen Strukturhasards
  - allen dynamischen Strukturhasards, falls nur eine Eingabevariable wechselt.

Beispiel

- Schaltnetz von vorhin:





- Hier sind alle Primimplikanten realisiert.

  → Dieses Schaltnetz ist frei von allen statischen Strukturhasards und allen dynamischen Strukturhasards, bei denen nur eine Eingangsvariable wechselt.

Behebung dynamischer Strukturhasards

- Die Behebung dynamischer Strukturhasards ist erheblich schwieriger als die Behebung statischer Strukturhasards !

- Problem:

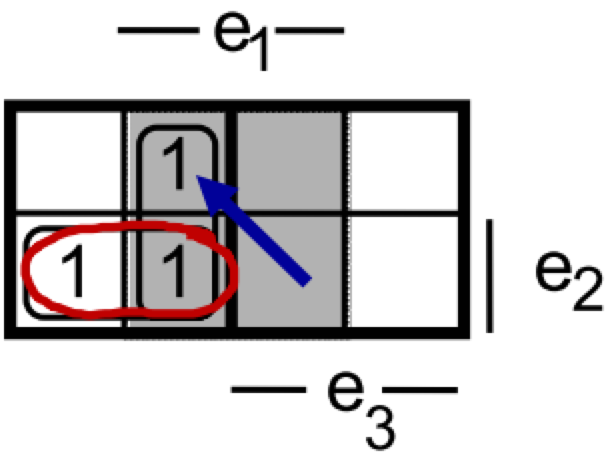
  Die Behebung eines dynamischen Strukturhasards für einen Übergang kann einen Strukturhasard für einen anderen Übergang erzeugen.

Behebung dynamischer Strukturhasards

- **Vorgehensweise**
  - Einvariablenwechsel → Satz von Eichelberger
  - **Mehrvariablenwechsel:**
  - Hier soll kurz ein Verfahren für zweistufige Schaltnetze gezeigt werden:
  - **Regel für zweistufige Schaltnetze in disjunktiver Form:**
  - Ein dynamischer Strukturhasard bei einem Übergang wird genau dann vermieden, wenn jeder Einsblock, der in den Übergangsbereich hineinragt, die Einsbelegung des Übergangs umschließt.

Behebung dynamischer Strukturhasards

$$y = e_1\overline{e_3} \lor e_2\overline{e_3}$$

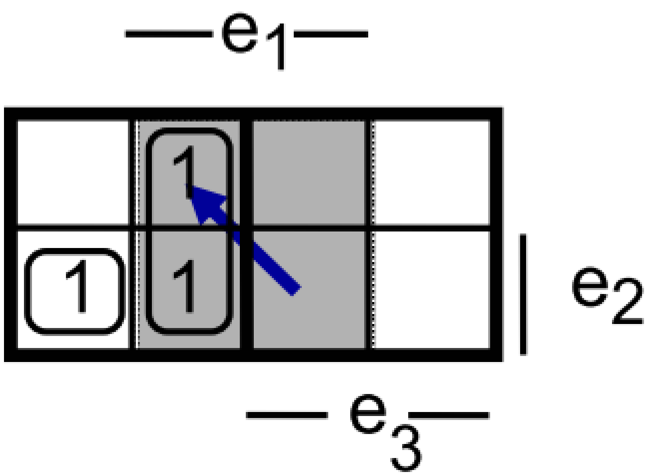


- Der Übergang $(e_3, e_2, e_1) : (1{,}1{,}1) \rightarrow (0{,}0{,}1)$ ist mit einem dynamischen 01-Strukturhasard behaftet.

Behebung dynamischer Strukturhasards

- Beseitigung des Hasards durch Aufspalten des Einsblocks $e_2 e_3$, der in den Übergangsbereich hineinragt, die Einsbelegung aber nicht umschließt.

$$y = e_1 \overline{e_3} \vee \overline{e_1} e_2 \overline{e_3}$$



- **Hasardfreier Übergang**
  - Durch die Beseitigung des 01-Strukturhasards handelt man sich einen 1-Strukturhasard im Übergang:

    $(e_3, e_2, e_1) : (0,1,1) \rightarrow (0,1,0)$ ein!

Wieso funktioniert dieses Verfahren?

- Einsblöcke, die in den Übergangsbereich hineinragen (im Beispiel:$e_2\overline{e_3}$), die Einsbelegung aber nicht umschließen, sind am Anfang und am Ende des Übergangs sicher 0.

- Während des Übergangs können diese jedoch kurzfristig zu 1 werden.

  → **Hasardfehler**

- (Dies lässt sich auch auf Nullblöcke übertragen.)