

# Zwei Kreuzungen: C und D, D und E

## Verschiedene Arten, die Sicherheit zu formalisieren

The screenshot shows the LogicTraffic software interface. The main window displays a traffic intersection with four lanes. The top-left and bottom-right lanes are green, while the top-right and bottom-left lanes are grey. A vertical dashed green line labeled 'D' represents a road crossing the intersection. A horizontal dashed blue line labeled 'C' represents a road crossing the intersection. A horizontal dashed red line labeled 'E' represents a road crossing the intersection. Black arrows indicate traffic flow: right in the top-left, left in the top-right, right in the bottom-left, and up in the bottom-right. A yellow cone of light highlights the intersection area.

C	D	E	sicher
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

Buttons: Tipp holen, Testen, Löschen, Scrollen, DNF

Logic formula:  $(\neg C \wedge \neg E) \vee (\neg D)$

Formeleditor: A B C D E,  $\wedge$ ,  $\vee$ ,  $\neg$ ,  $\rightarrow$ , ( ), ) , <

Buttons: Parsen, Löschen, Zeige ParseBaum

# Wann ist eine Kreuzung von Spur 1 und Spur 2 sicher?

**Spur 1 grün => Spur 2 rot**

wenn Spur 1 rot, kann Spur 2 grün oder rot sein

**nicht (Spur 1 grün und Spur 2 grün)**

nicht beide Spuren gleichzeitig grün

# Eine Implikation für jede Kreuzung, UND-verknüpft

The screenshot shows the LogicTraffic software interface. At the top, the window title is "LogicTraffic". Below it is a menu bar with "Datei" and "Über". The main area is titled "Situation 4" and displays a traffic intersection diagram. A vertical road (D) crosses two horizontal roads (C and E). The intersection points are circled in red. A callout box with a yellow background and a blue border contains the text: "Für jede Kreuzung: 'Spur 1 grün impliziert Spur 2 rot'". Below the diagram, a logic formula is displayed:  $(D) \rightarrow (\neg C \wedge \neg E)$ . At the bottom, there is a "Formeleditor" (Formula Editor) with a keyboard layout and a text input field containing the formula  $(D \rightarrow \neg C) \wedge (D \rightarrow \neg E)$ . The formula is split into two parts,  $(D \rightarrow \neg C)$  and  $(D \rightarrow \neg E)$ , which are circled in red. Red arrows point from these circles to the callout box. The interface also includes buttons for "Tipp Holen", "Testen", "Löschen", "Scrollen", "Implikation", "Parsen", "Löschen", and "Zeige ParseBaum".

Für jede Kreuzung:  
"Spur 1 grün  
impliziert  
Spur 2 rot"

$(D) \rightarrow (\neg C \wedge \neg E)$

$(D \rightarrow \neg C) \wedge (D \rightarrow \neg E)$

# Eine Implikation für jede Kreuzung, UND-verknüpft

The screenshot shows the LogicTraffic software interface. At the top, the menu bar includes 'Datei' and 'Über'. Below it, the title 'Situation 4' is displayed. The main area features a traffic intersection diagram with four lanes: a top lane (green), a bottom lane (green), a left lane (blue), and a right lane (red). A central vertical lane is labeled 'D'. Red arrows indicate traffic flow from the top and bottom lanes towards the center, while blue and red arrows show flow from the left and right lanes towards the center. Two red circles highlight the intersection points. A yellow text box on the right contains the text: 'Für jede Kreuzung: "Spur 2 grün impliziert Spur 1 rot"'. Below this, a dropdown menu is set to 'Implikation', and a text box displays the formula  $(D) \rightarrow (\neg C \wedge \neg E)$ . At the bottom, the 'Formeleditor' section shows a keyboard interface with buttons for variables A-E, logical operators  $\wedge$ ,  $\vee$ ,  $\neg$ ,  $\rightarrow$ , and parentheses. The formula  $(C \rightarrow \neg D) \wedge (E \rightarrow \neg D)$  is entered, with the two sub-formulas highlighted in red boxes. Red arrows connect these boxes to the intersection points in the diagram. At the bottom right, there are buttons for 'Parsen', 'Löschen', and 'Zeige ParseBaum'.

# Eine negierte Konjunktion für jede Kreuzung, UND verknüpft

The screenshot shows the LogicTraffic software interface. The main window displays a traffic intersection with four lanes: a top lane (D) with a green light, a bottom lane (E) with a red light, a left lane (C) with a blue light, and a right lane (D) with a red light. A central vertical lane is labeled 'D'. A yellow text box on the right contains the text: "Für jede Kreuzung: 'nicht (Spur 1 grün und Spur 2 grün)'". Below this text box is the logic formula:  $(\neg D \vee \neg E) \wedge (\neg C \vee \neg D)$ . At the bottom, the 'Formeleditor' (Formula Editor) shows the formula  $\neg(D \wedge E) \wedge \neg(C \wedge D)$ , with red boxes highlighting the sub-expressions  $\neg(D \wedge E)$  and  $\neg(C \wedge D)$ . The interface includes a menu bar (Datei, Über), a toolbar (Tipp Holen, Testen, Löschen, Scrollen), and a status bar (KNF, Parsen, Löschen, Zeige ParseBaum).

# Zur Erinnerung: De Morgan'sche Gesetze

**nicht (a und b) = nicht a oder nicht b**

**nicht (a oder b) = nicht a und nicht b**

[http://de.wikipedia.org/wiki/De\\_Morgansche\\_Regel](http://de.wikipedia.org/wiki/De_Morgansche_Regel)

# Konjunktive Normalform: Eine Disjunktion für jede Kreuzung, UND verknüpft

The screenshot shows the LogicTraffic software interface. The main window displays a traffic intersection with four lanes: a top lane (green), a middle lane (grey), a bottom lane (green), and a central lane (grey). The lanes are labeled with letters: 'E' for the top lane, 'C' for the middle lane, and 'D' for the bottom lane. A yellow callout box points to the intersection with the text: "Für jede Kreuzung: 'Spur 1 rot oder Spur 2 rot'". Below this, the formula  $(\neg D \vee \neg E) \wedge (\neg C \vee \neg D)$  is shown. The bottom part of the interface is the 'Formeleditor' (Formula Editor), which contains the same formula  $(\neg D \vee \neg E) \wedge (\neg C \vee \neg D)$ . The formula is broken down into its components:  $(\neg D \vee \neg E)$  and  $(\neg C \vee \neg D)$  are highlighted with red boxes, and red arrows point from these boxes to the corresponding disjunctions in the callout box. The interface also includes a menu bar (Datei, Über), a toolbar (Tipp Holen, Testen, Löschen, Scrollen), and a status bar (KNF, Parsen, Löschen, Zeige ParseBaum).

LogicTraffic

Datei Über

Situation 4

Für jede Kreuzung:  
"Spur 1 rot  
oder Spur 2 rot"

$(\neg D \vee \neg E) \wedge (\neg C \vee \neg D)$

Formeleditor

$(\neg D \vee \neg E) \wedge (\neg C \vee \neg D)$

Tipp Holen  
Testen  
Löschen  
Scrollen

KNF

Parsen Löschen Zeige ParseBaum

# Kanonische Disjunktive Normalform: Eine Konjunktion für jede sichere Situation

LogicTraffic

Datei Über

Situation 4

Jeder Klammersausdruck beschreibt eine sichere Zeile in der Tabelle.

Die Kreuzungen sind sicher, wenn mindestens eine der sicheren Zeilen erfüllt ist. Daher werden die Zeilen ODER-verknüpft.

C	D	E	sicher
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

$(\neg C \wedge \neg D \wedge \neg E) \vee (\neg C \wedge \neg D \wedge E) \vee (\neg C \wedge D \wedge \neg E) \vee (C \wedge \neg D \wedge \neg E) \vee (C \wedge \neg D \wedge E)$

Formeleditor

$(\neg C \wedge \neg D \wedge \neg E) \vee (\neg C \wedge \neg D \wedge E) \vee (\neg C \wedge D \wedge \neg E) \vee (C \wedge \neg D \wedge \neg E) \vee (C \wedge \neg D \wedge E)$

Parsen Löschen Zeige ParseBaum

Tippen, Testen, Löschen, Scrollen, KDNF

# Kanonische Konjunktive Normalform: Eine Disjunktion für jede unsichere Situation

LogicTraffic

Datei Über

Situation 4

Jeder Klammerausdruck beschreibt die Negation einer unsicheren Zeile in der Tabelle.

Die Kreuzungen sind sicher, wenn alle negierten unsicheren Zeilen erfüllt sind. Daher werden die Zeilen UND-verknüpft.

C	D	E	sicher
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

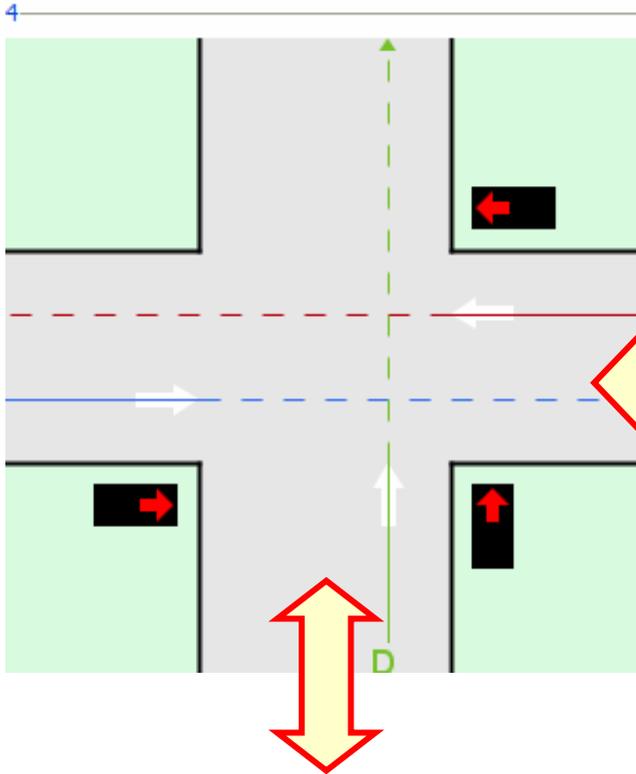
$(C \vee \neg D \vee \neg E) \wedge (\neg C \vee \neg D \vee E) \wedge (\neg C \vee \neg D \vee \neg E)$

$(C \vee \neg D \vee \neg E) \wedge (\neg C \vee \neg D \vee E) \wedge (\neg C \vee \neg D \vee \neg E)$

^^ Zur Erinnerung: nicht (nicht C und D und E) = C oder nicht D oder nicht E

# Zwei Kreuzungen: C und D, D und E

## Verschiedene Arten, die Sicherheit zu formalisieren



C	D	E	sicher
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

**Implikation:**

$$(D \rightarrow \neg C) \wedge (D \rightarrow \neg E)$$

**KNF:**

$$\neg(D \wedge E) \wedge \neg(C \wedge D)$$

**KDNF:**

$$(C \vee \neg D \vee \neg E) \wedge (\neg C \vee \neg D \vee E) \wedge (\neg C \vee \neg D \vee \neg E)$$

**KKNF:**

$$\begin{aligned} &(\neg C \wedge \neg D \wedge \neg E) \vee (\neg C \wedge \neg D \wedge E) \vee (\neg C \wedge D \wedge \neg E) \\ &\vee (C \wedge \neg D \wedge \neg E) \vee (C \wedge \neg D \wedge E) \end{aligned}$$

# **Zwei Kreuzungen: C und D, D und E**

## **Verschiedene Arten, die Sicherheit zu formalisieren**

**Aus den Kreuzungen können die Formeln für Implikation und KNF abgeleitet werden, und umgekehrt können aus den Formeln für Implikation und KNF die Kreuzungen erstellt werden.**

**Die Kreuzungen beschreiben den Inhalt der Wahrheitstabelle. Umgekehrt beschreibt die Tabelle die Kreuzungen, aber es ist schwieriger, aus der Tabelle heraus die Kreuzungen zu bestimmen.**

**Aus der Wahrheitstabelle heraus können die Formeln für KDNF und KKNF abgeleitet werden, und umgekehrt kann von KDNF und KKNF die Wahrheitstabelle erstellt werden.**