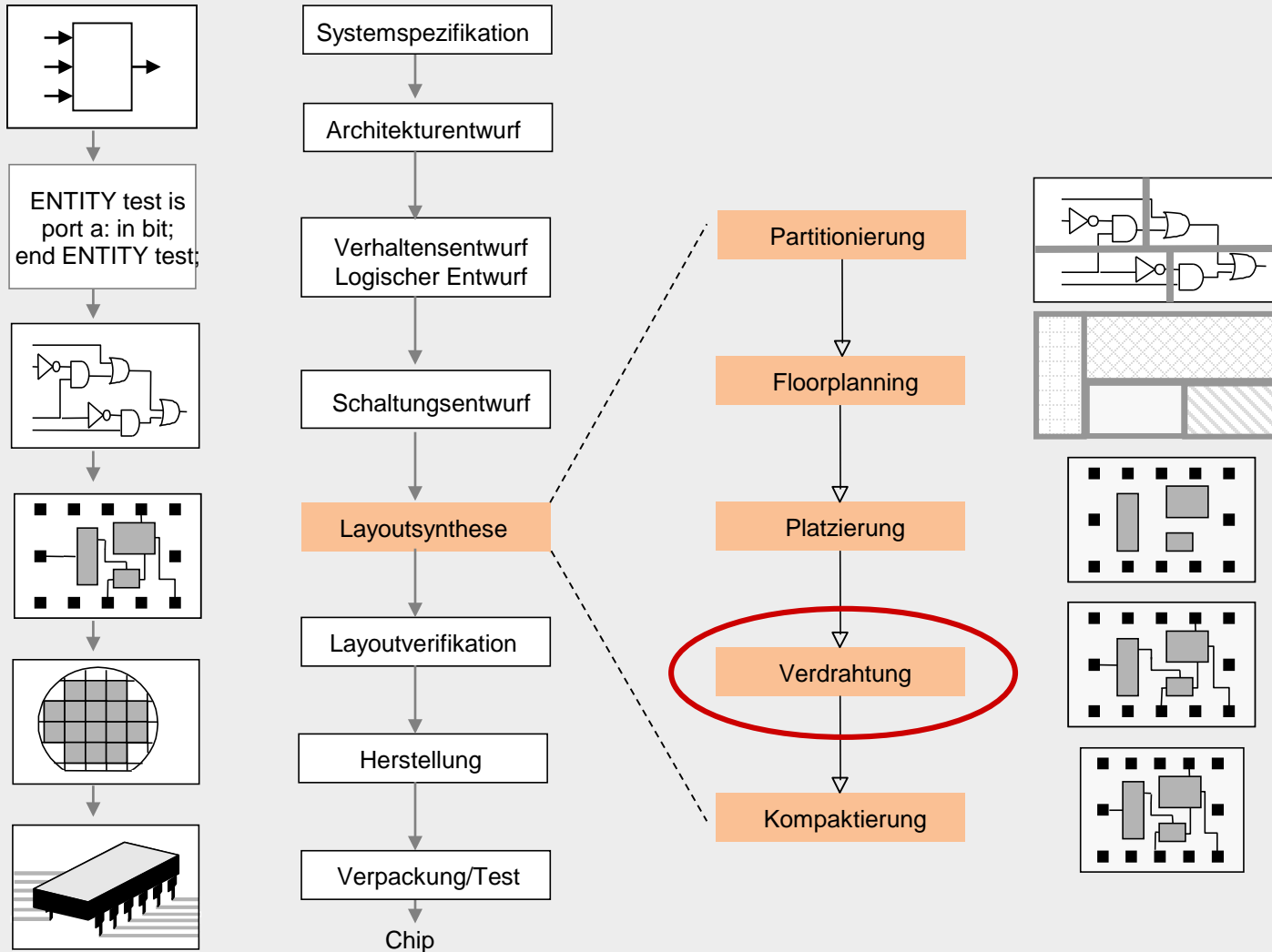
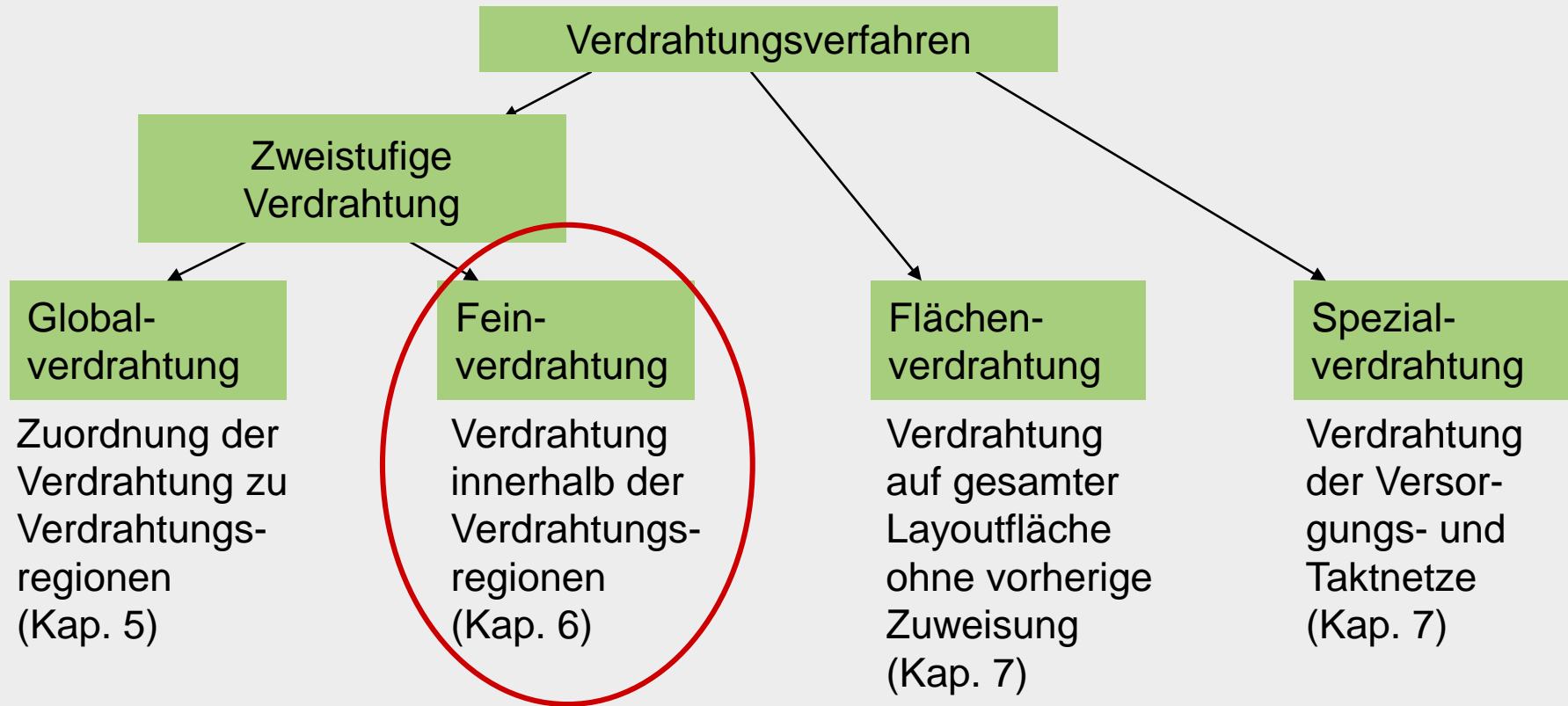


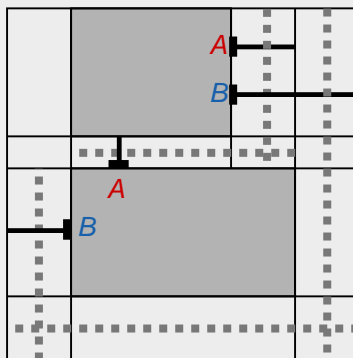
- 6.1 Einführung
- 6.2 Begriffsbestimmungen
- 6.3 Horizontaler und vertikaler Verträglichkeitsgraph
 - 6.3.1 Horizontale Verträglichkeitsdarstellung
 - 6.3.2 Vertikale Verträglichkeitsdarstellung
- 6.4 Optimierungsziele
- 6.5 Algorithmen für die Kanalverdrahtung
 - 6.5.1 Left-Edge-Algorithmus
 - 6.5.2 Dogleg-Left-Edge-Algorithmus
 - 6.5.3 Greedy-Kanalverdrahter (Greedy Channel Router)
- 6.6 Switchbox-Verdrahtung
- 6.7 OTC-Verdrahtung bei drei Lagen
- 6.8 Multilayer-OTC-Verdrahtung

6.1 Einführung

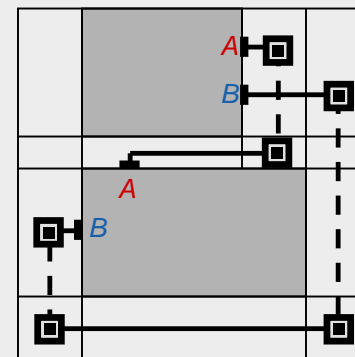




Die Aufgabe der Feinverdrahtung besteht darin, die bei der Globalverdrahtung einer Verdrahtungsregion zugeordneten Netzsegmente in dieser Region **detaillierte Verdrahtungswege und -ebenen** zuzuweisen.



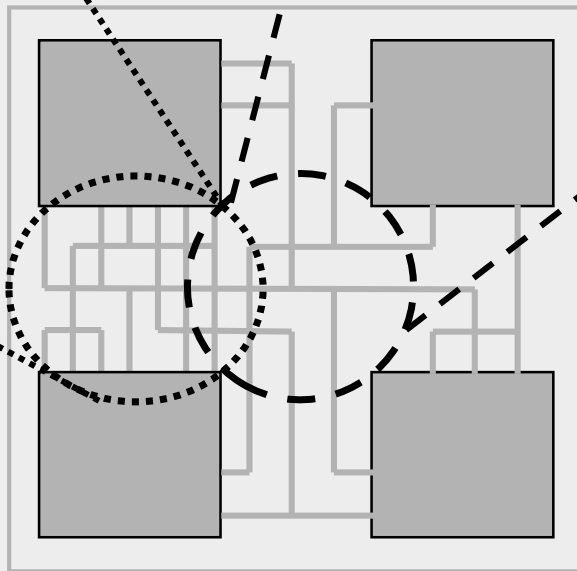
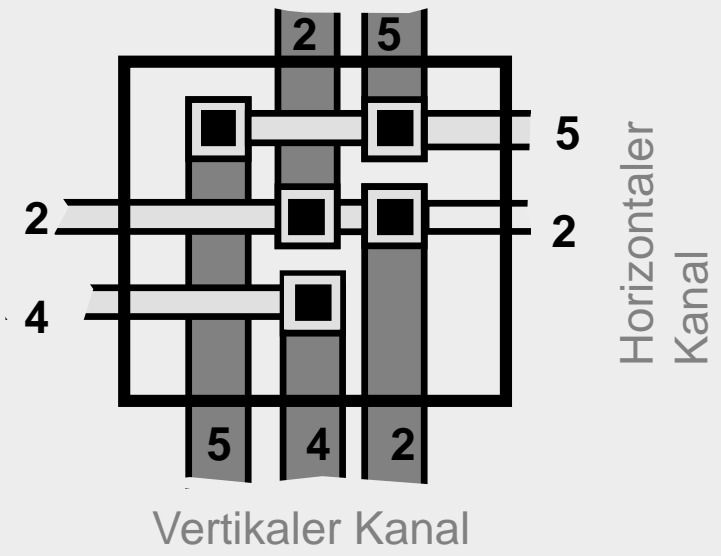
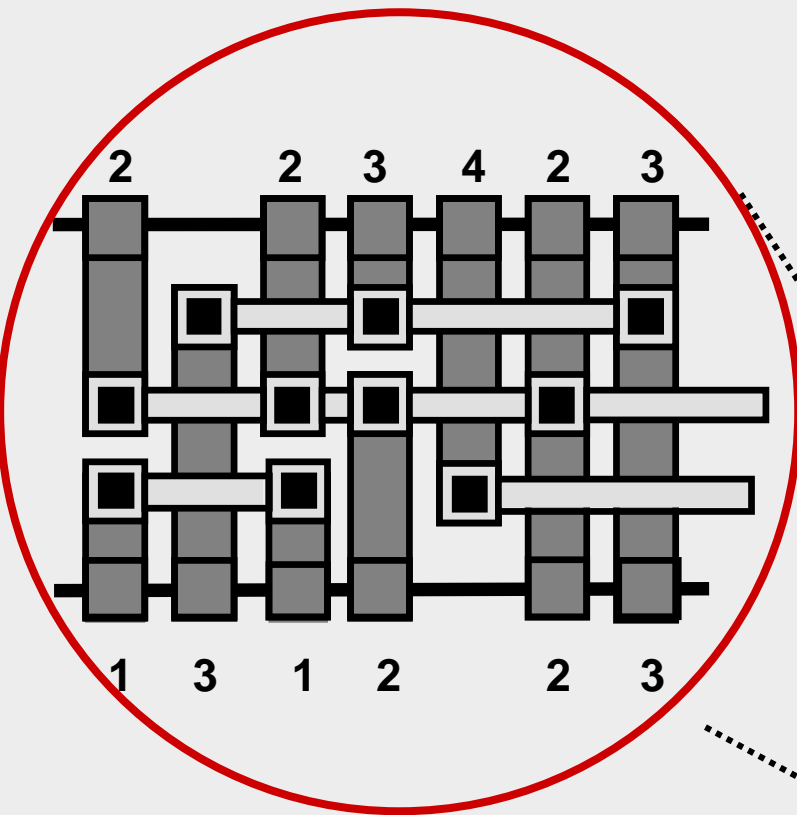
Globalverdrahtung



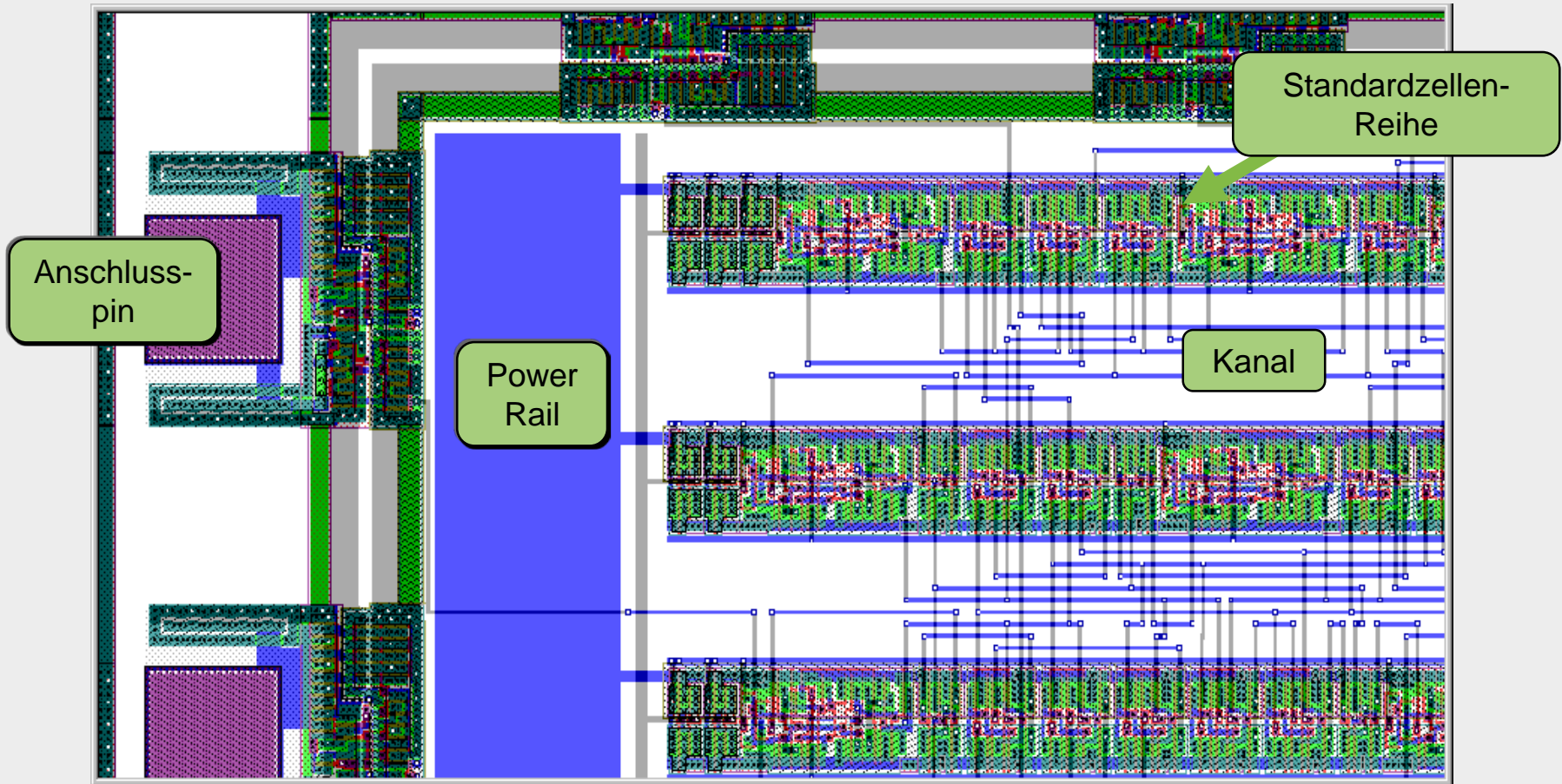
Feinverdrahtung

- Verdrahtungsregionen werden als **Kanäle** bezeichnet, sofern die Netzanschlüsse an zwei gegenüberliegenden Seiten angeordnet sind, und als **Switchboxen**, wenn sich Anschlüsse an allen vier Seiten befinden.
- Klassische Kanaldefinition verliert aufgrund Ebenenzunahme an Bedeutung.
- Mehrlagen (Multilayer)-Verdrahtung oftmals mit zellenunabhängigen Verdrahtungsbereichen (**Verdrahtungsregion, Tile, Box** usw.)

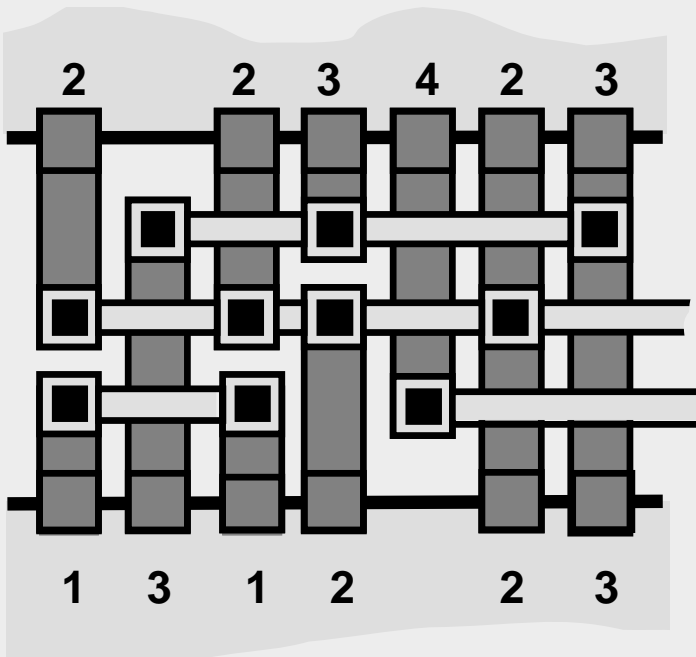
Kanal- und Switchbox-Verdrahtung



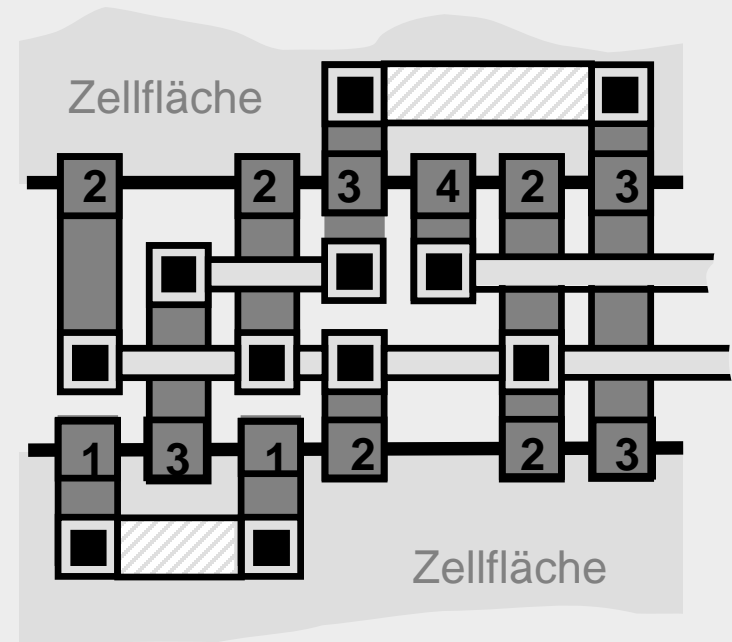
Kanalverdrahtung



2-Ebenen-Kanalverdrahtung

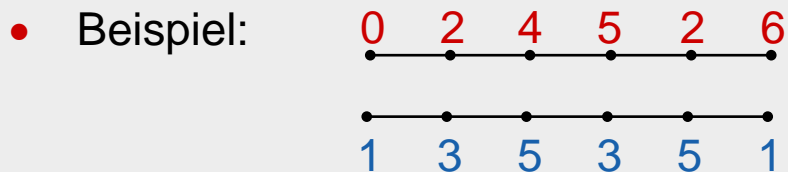


3-Ebenen-OTC-Verdrahtung
OTC: Over the cell



6.2 Begriffsbestimmungen

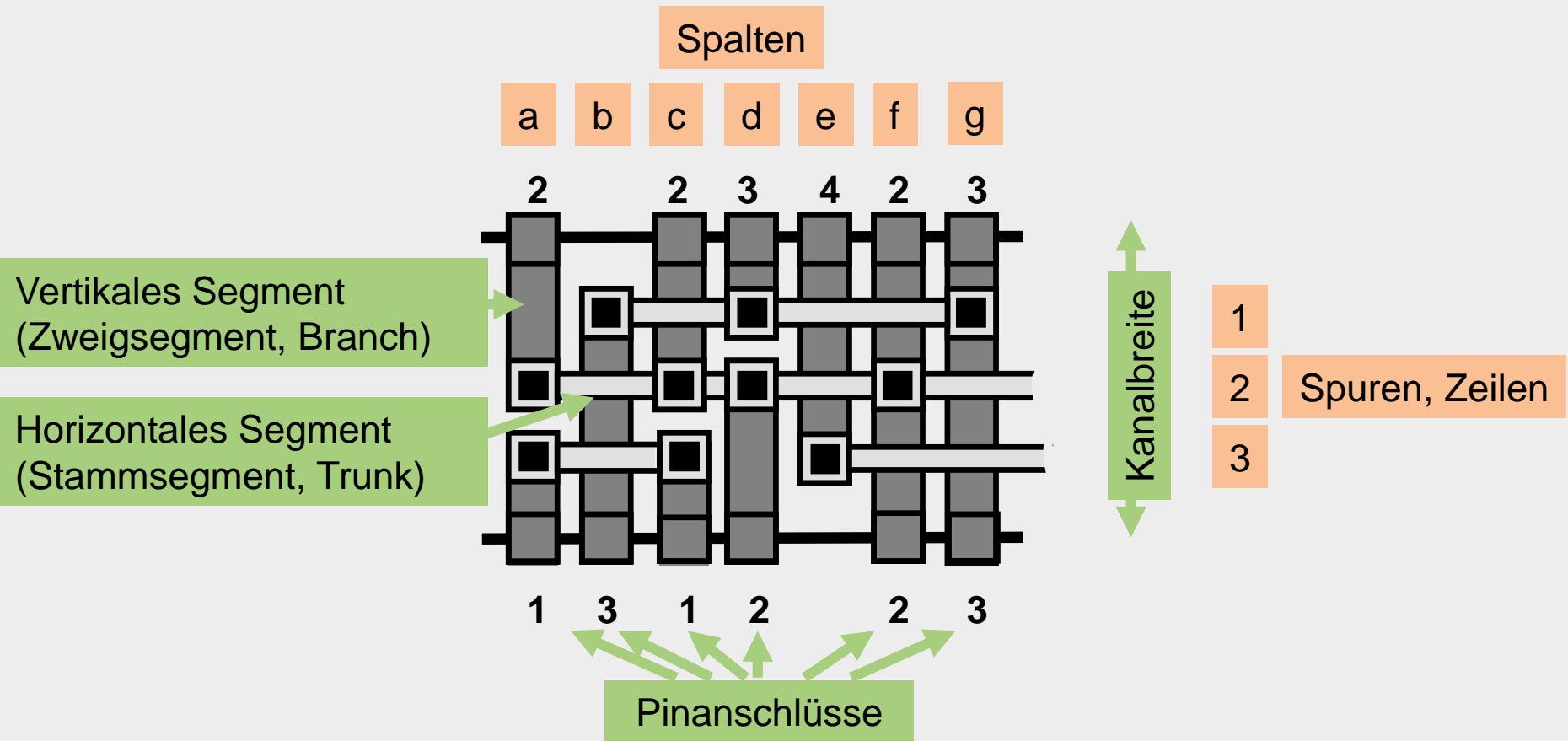
- Anschlussreihen (oben und unten) werden durch zwei Mengen gekennzeichnet, bei denen die Netznummer der jeweiligen Spaltenposition zugewiesen ist.
- Null markiert dabei ein nicht angeschlossenes Pin.
- Anschlüsse mit der gleichen Nummer sind Anschlüsse des gleichen Netzes und miteinander zu verbinden.
- Oft: Vektoren $TOP(k)$ und $BOT(k)$, welche die Gitterpositionen an der Ober- (TOP) und Unterseite (BOT) des Kanals in der Spalte k repräsentieren.



– $TOP = [0, 2, 4, 5, 2, 6]$ bzw. $TOP(1) = 0$, $TOP(2) = 2$, ...

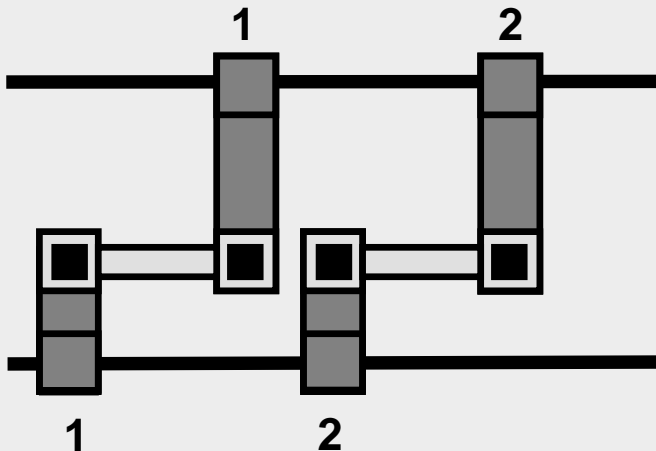
– $BOT = [1, 3, 5, 3, 5, 1]$ bzw. $BOT(1) = 1$, $BOT(2) = 3$, ...

6.2 Begriffsbestimmungen

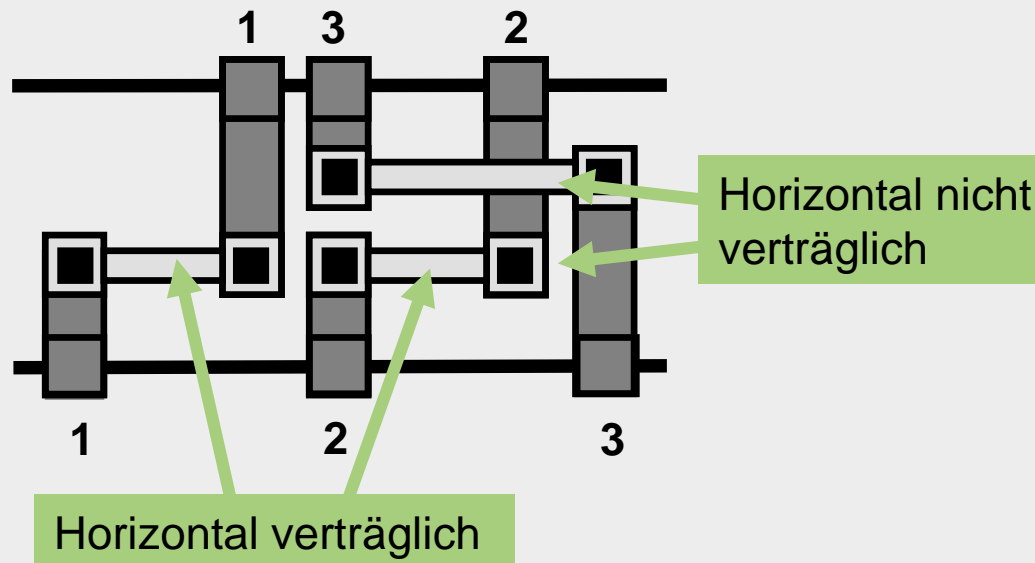


Horizontale Verträglichkeit

- Annahme: eine Ebene für die horizontale Verdrahtung.
- Sollten zwei horizontale Segmente verschiedener Netze keine Spalten-Überlappung haben, also nicht denselben Horizontalbereich beanspruchen, so sind sie auf gleicher Spur platzierbar; damit: „horizontal verträglich“.
- Ansonsten sind für beide Netze unterschiedliche Spuren zu reservieren.

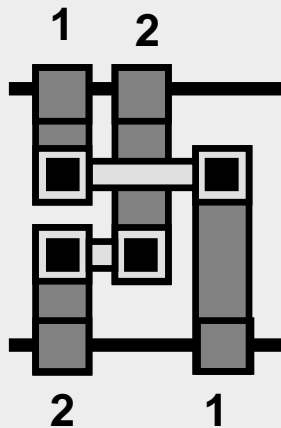


Horizontale Verträglichkeit

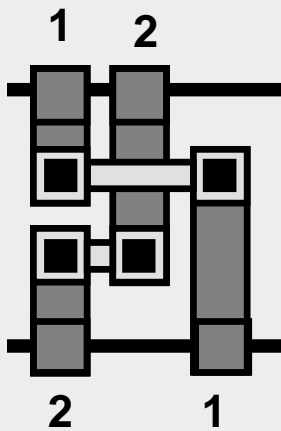


Vertikale Verträglichkeit

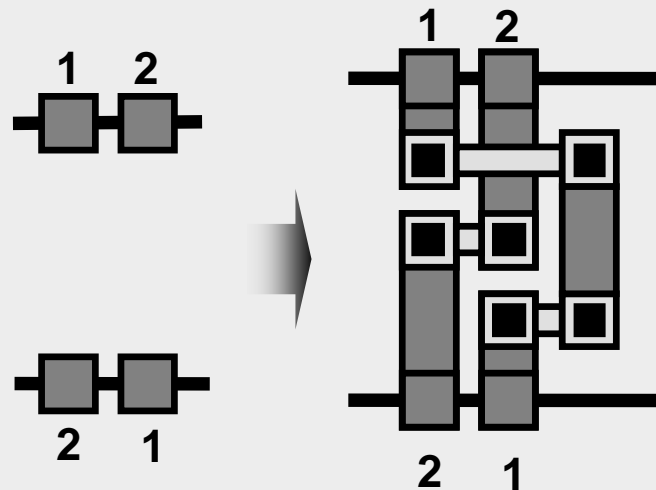
- Sollte nur eine vertikale Ebene zur Verfügung stehen, so dürfen sich zwei Netze nicht auf einer vertikalen Spalte überlappen.
- Der von „oben“ kommende Anschluss muss also rechtzeitig „aufhören“, um sich mit dem von „unten“ kommenden nicht zu überlagern.
- ⇒ Das horizontale Segment eines Netzes, welches am oberen Anschluss einer Spalte angeschlossen ist, muss über dem horizontalen Segment eines Netzes liegen, welches mit dem unteren Anschluss dieser Spalte verbunden ist.



Vertikale Verträglichkeit



Vertikal
verträglich



Vertikal nicht verträglich

6.1 Einführung

6.2 Begriffsbestimmungen

 6.3 **Horizontaler und vertikaler Verträglichkeitsgraph**

6.3.1 Horizontale Verträglichkeitsdarstellung

6.3.2 Vertikale Verträglichkeitsdarstellung

6.4 Optimierungsziele

6.5 Algorithmen für die Kanalverdrahtung

6.5.1 Left-Edge-Algorithmus

6.5.2 Dogleg-Left-Edge-Algorithmus

6.5.3 Greedy-Kanalverdrahter (Greedy Channel Router)

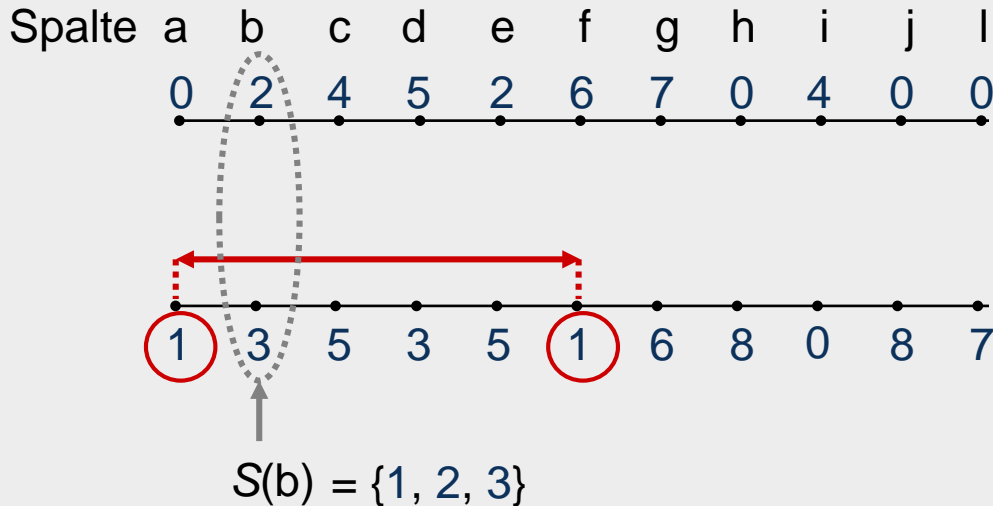
6.6 Switchbox-Verdrahtung

6.7 OTC-Verdrahtung bei drei Lagen

6.8 Multilayer-OTC-Verdrahtung

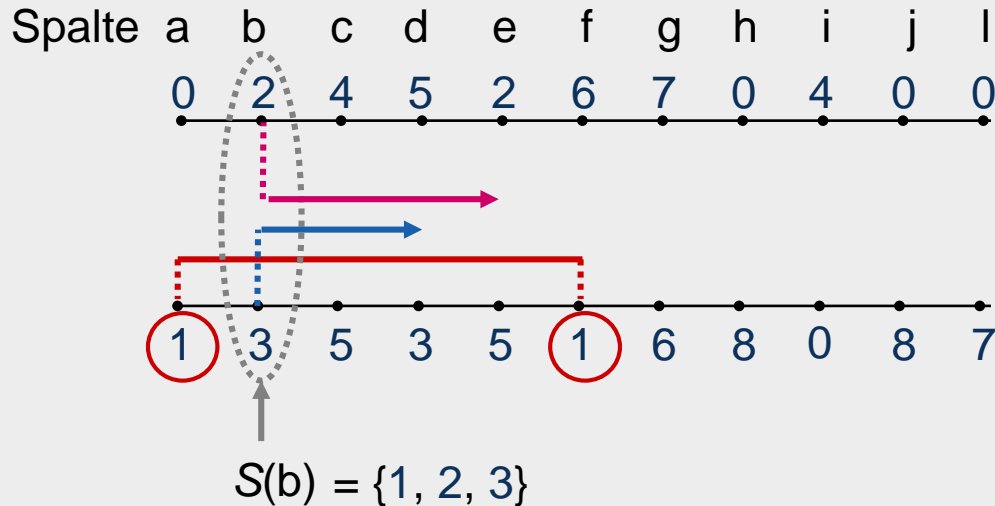
- Jedes Kanalverdrahtungsproblem kann mittels zweier Verträglichkeitsgraphen modelliert werden, welche die horizontale und vertikale Verträglichkeit abbilden.
- Damit sind schon vor Beginn der eigentlichen Kanalverdrahtung die minimal benötigte Spuranzahl und eventuelle Konfliktsituationen vorhersehbar.
- Außerdem steuern beide Graphen die Verdrahtungsreihenfolge und -anordnung bei einigen Algorithmen.

6.3.1 Horizontale Verträglichkeitsdarstellung



- Das horizontale Segment eines Netzes wird durch den äußeren linken und rechten Netzknoten festgelegt.
 - $S(k)$ sei die Menge der Netze, deren horizontale Segmente die Vertikalspalte k schneiden.
- ⇒ Jedes $S(k)$ enthält die Netze, die in Spalte k nach oben und unten angeschlossen werden und die Netze, deren Anschlüsse links und rechts von k liegen, die damit die Spalte k schneiden.

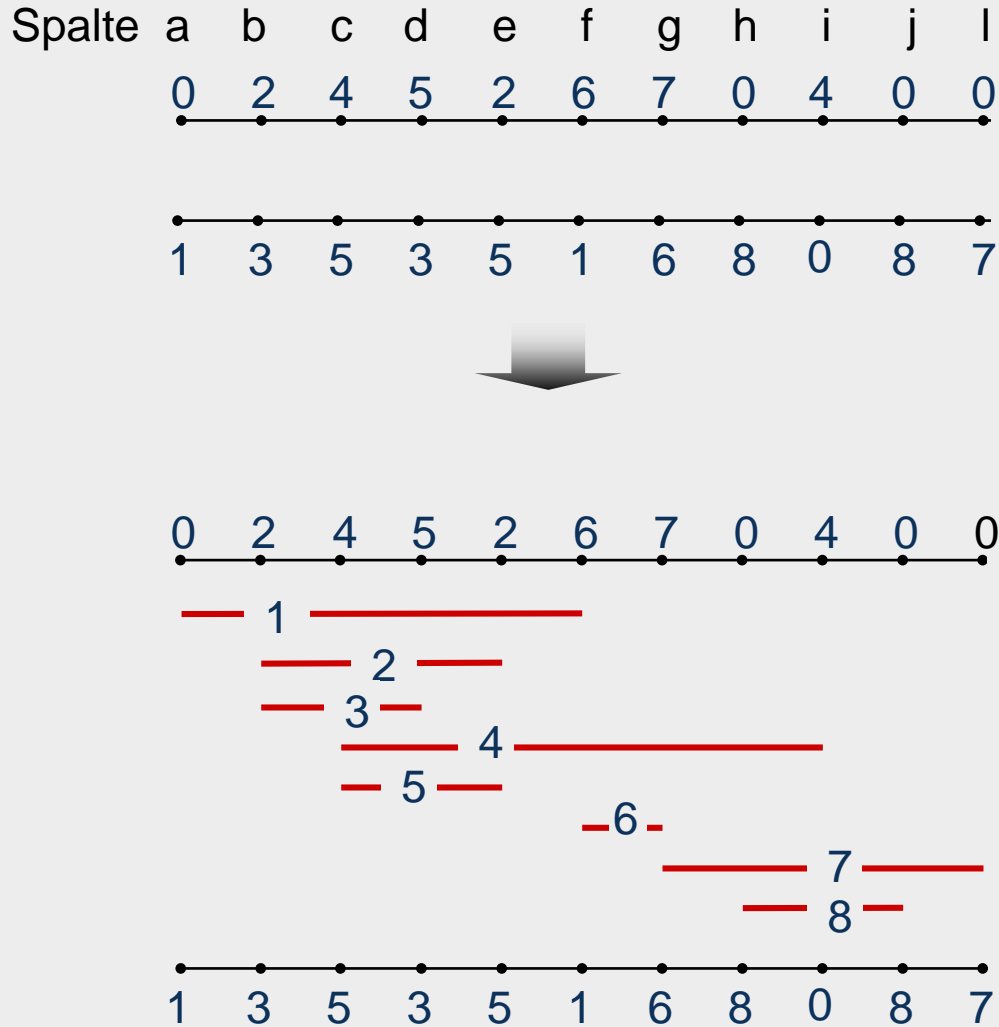
6.3.1 Horizontale Verträglichkeitsdarstellung



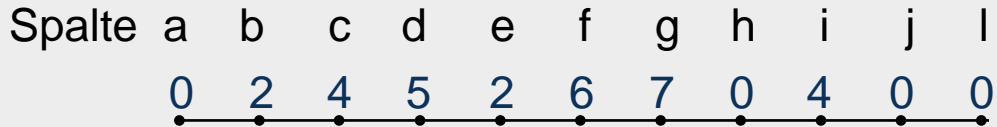
- Da sich die horizontalen Segmente verschiedener Netze nicht überlagern dürfen, ist es nicht erlaubt, zwei Netze aus $S(k)$ in der Spalte k auf der gleichen Horizontalspur zu platzieren.
 - Diese Bedingung muss in jeder Vertikalspalte eingehalten werden.
- ⇒ Minimale Spuranzahl des Kanals = maximale Mächtigkeit von $S(k)$ ($k = a, b, \dots$)

Anzahl der Elemente
der Menge

6.3.1 Horizontale Verträglichkeit: Zonendarstellung

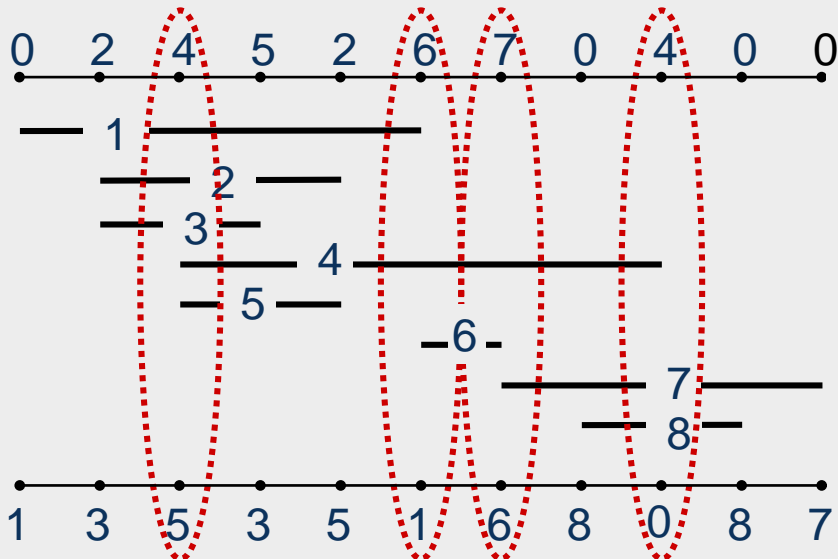


6.3.1 Horizontale Verträglichkeit: Zonendarstellung

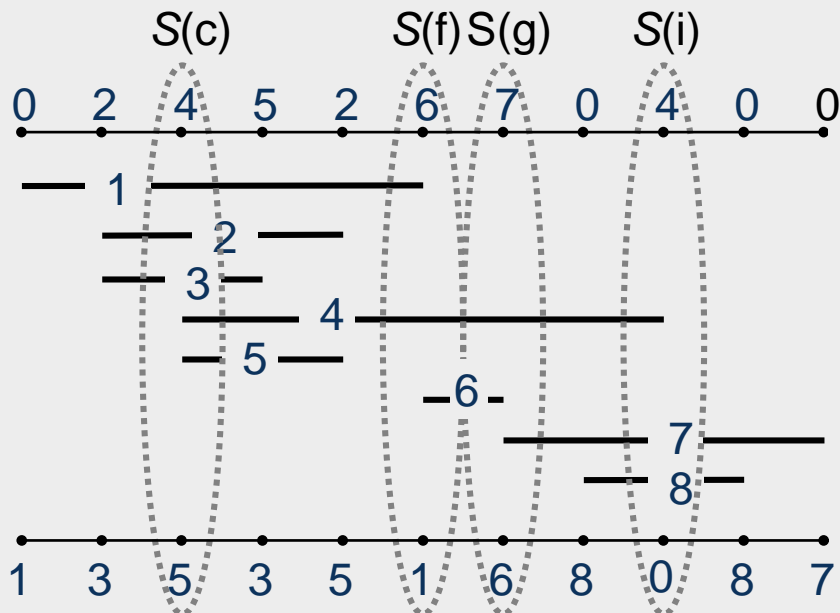


$S(k)$, die nicht Untermengen anderer Spalten sind

$S(a)$ $S(b)$ $S(c)$ $S(d)$ $S(e)$ $S(f)$ $S(g)$ $S(h)$ $S(i)$ $S(j)$ $S(l)$



6.3.1 Horizontale Verträglichkeit: Zonendarstellung



S(c)	S(f)	S(g)	S(i)
1		7	
2	6		8
3			
4			
5			

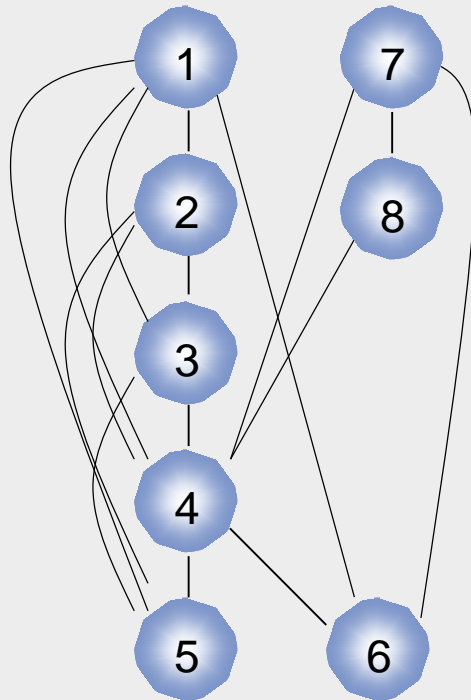
6.3.1 Horizontale Verträglichkeit: Zonendarstellung



Minimale Spuranzahl des Kanals = 5

	S(c)	S(f)	S(g)	S(i)
1			7	
2		6		8
3				
4				
5				

6.3.1 Horizontale Verträglichkeit: Graphendarstellung



	S(c)	S(f)	S(g)	S(i)
1			7	
2		6		8
3				
4				
5				

6.3.1 Horizontale Verträglichkeit: Graphendarstellung

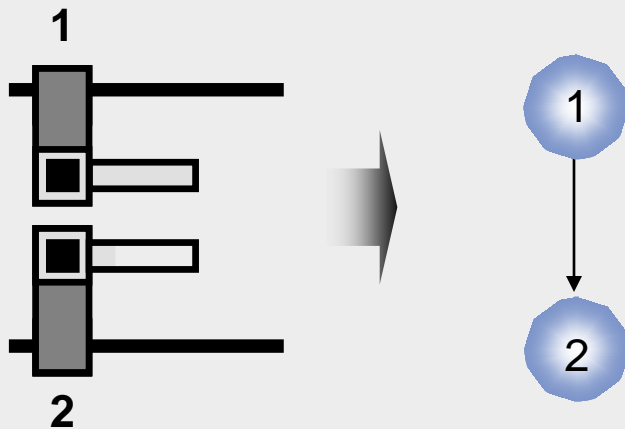


Zonendarstellung

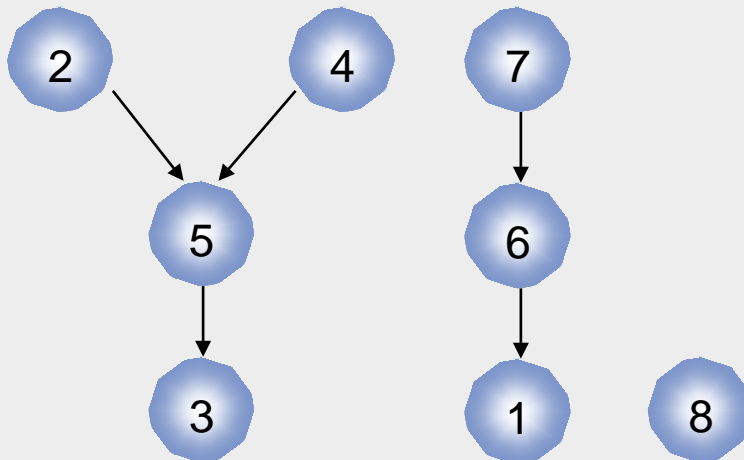
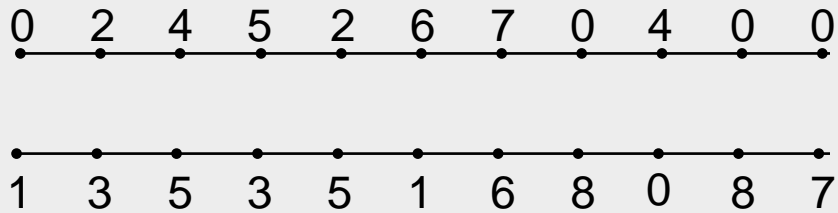
	S(c)	S(f)	S(g)	S(i)
1			7	
2		6		8
3				
4				
5				

6.3.2 Vertikale Verträglichkeitsdarstellung

- In einem vertikalen Verträglichkeitsgraphen repräsentiert ein Knoten $i \in V$ das Netz i .
- Eine gerichtete Kante zwischen den Knoten i und j gibt an, dass das Anschlusspin des Netzes i auf der oberen Kanalkante (*TOP*) sowie das Anschlusspin des Netzes j auf der unteren Kanalkante (*BOT*) auf gleicher Spaltenposition angeordnet sind.
- Damit muss das horizontale Segment des Netzes i über dem des Netzes j liegen, damit keine vertikale Überlappung in der betreffenden Anschlusspalte entsteht.

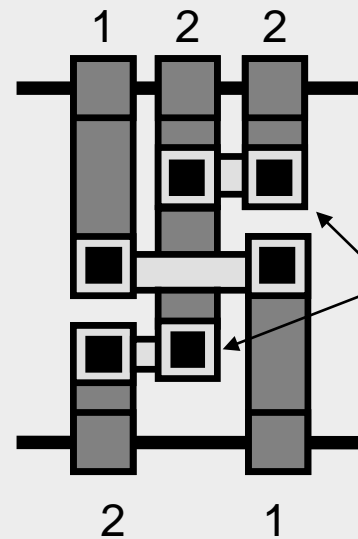
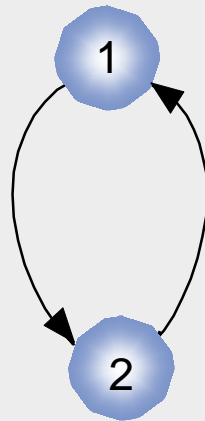
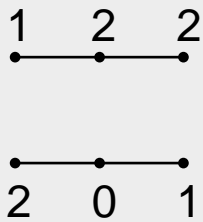


6.3.2 Vertikale Verträglichkeitsdarstellung



Vertikaler Verträglichkeitsgraph
(Vertical Constraint Graph, VCG)

6.3.2 Vertikale Verträglichkeitsdarstellung



Netzaufspaltung

Zyklischer Konflikt

Schlussfolgerungen aus horizontaler (HCG) und vertikaler Verträglichkeitsdarstellung (VCG)

- Der HCG gibt die minimal benötigte Spuranzahl an.
- Diese Spuranzahl ist aber nur realistisch, sollte der VCG keine Schleifen enthalten.
- Sollte der VCG keine Schleifen enthalten, dann kann jedes Netz mit einem horizontalen Segment verdrahtet werden (keine Netzaufsplittungen).
- Die Zuordnung der horizontalen Segmente zu den einzelnen Spuren ergibt sich aus dem VCG, da nur so Überlappungen bei den vertikalen Anschluss-Segmenten vermieden werden.
- Die Belegung einer Spur mit mehreren Netzen ergibt sich aus dem HCG, da dieser horizontal nicht-überlappende Netze angibt.

- 6.1 Einführung
- 6.2 Begriffsbestimmungen
- 6.3 Horizontaler und vertikaler Verträglichkeitsgraph
 - 6.3.1 Horizontale Verträglichkeitsdarstellung
 - 6.3.2 Vertikale Verträglichkeitsdarstellung
-  **6.4 Optimierungsziele**
- 6.5 Algorithmen für die Kanalverdrahtung
 - 6.5.1 Left-Edge-Algorithmus
 - 6.5.2 Dogleg-Left-Edge-Algorithmus
 - 6.5.3 Greedy-Kanalverdrahter (Greedy Channel Router)
- 6.6 Switchbox-Verdrahtung
- 6.7 OTC-Verdrahtung bei drei Lagen
- 6.8 Multilayer-OTC-Verdrahtung

- Das Ziel bei der Verdrahtung eines Kanals (einer Switchbox) besteht darin, für jedes Netz eine gültige Netzverbindung zu finden, welche sämtliche Anschlüsse dieses Netzes im jeweiligen Kanal (Switchbox) verbindet.
- Bei Standardzellen-Schaltungen besteht das Ziel auch darin, eine minimale Spuranzahl zu benutzen, um die Kanalbreite zu minimieren.
- Bei Gate-Array-Schaltungen besteht ein Ziel darin, die Verdrahtung innerhalb einer vorgegebenen Spuranzahl durchzuführen.

- 6.1 Einführung
- 6.2 Begriffsbestimmungen
- 6.3 Horizontaler und vertikaler Verträglichkeitsgraph
 - 6.3.1 Horizontale Verträglichkeitsdarstellung
 - 6.3.2 Vertikale Verträglichkeitsdarstellung
- 6.4 Optimierungsziele
- 6.5 Algorithmen für die Kanalverdrahtung**
 - 6.5.1 Left-Edge-Algorithmus
 - 6.5.2 Dogleg-Left-Edge-Algorithmus
 - 6.5.3 Greedy-Kanalverdrahter (Greedy Channel Router)
- 6.6 Switchbox-Verdrahtung
- 6.7 OTC-Verdrahtung bei drei Lagen
- 6.8 Multilayer-OTC-Verdrahtung

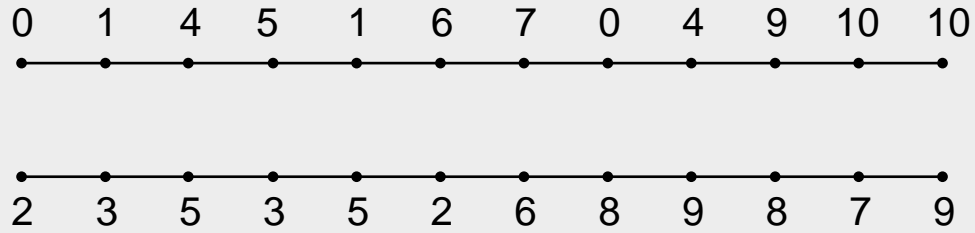
6.5.1 Left-Edge-Algorithmus

- Einfaches heuristisches Verfahren mit weiter Verbreitung als Basisalgorithmus
- Zuordnung von Netzen zu Spuren anhand
 - vertikalem Verträglichkeitsgraphen (VCG, zur Spurbestimmung) und
 - Zonendarstellung (horizontale Verträglichkeit, d.h. Mehrfachbelegung einer Spur).

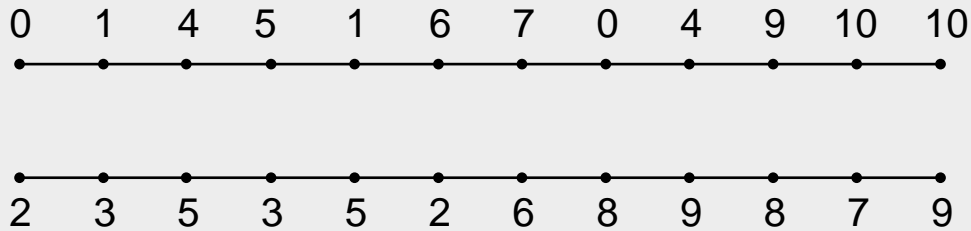
Ablauf

1. Aufbau des VCG und HCG (Zonendarstellung)
2. Aktuelle Spur $j = 1$ (obere Spur)
3. Für aktuelle Spur j
 - a) Für alle Netze ohne Vorgänger im VCG, Platzierung des am weitesten links liegenden Netzes in der Zonendarstellung auf Spur j und anschließend weitere nicht-überlappende (lt. Zonendarstellung) und vorgängerlose Netze (lt. VCG)
 - b) Löschen aller platzierten Netze im VCG und in der Zonendarstellung
4. Aktuelle Spur $j = j + 1$. Falls noch Netze im VCG vorhanden sind, weiter mit Schritt 3
5. ENDE.

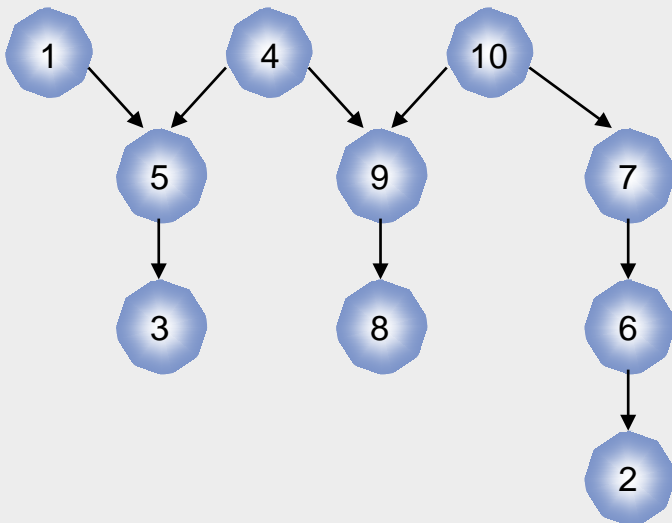
6.5.1 Left-Edge-Algorithmus: Beispiel



6.5.1 Left-Edge-Algorithmus: Beispiel

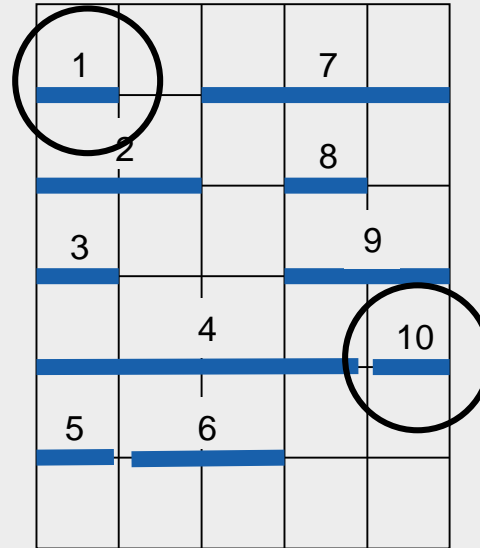
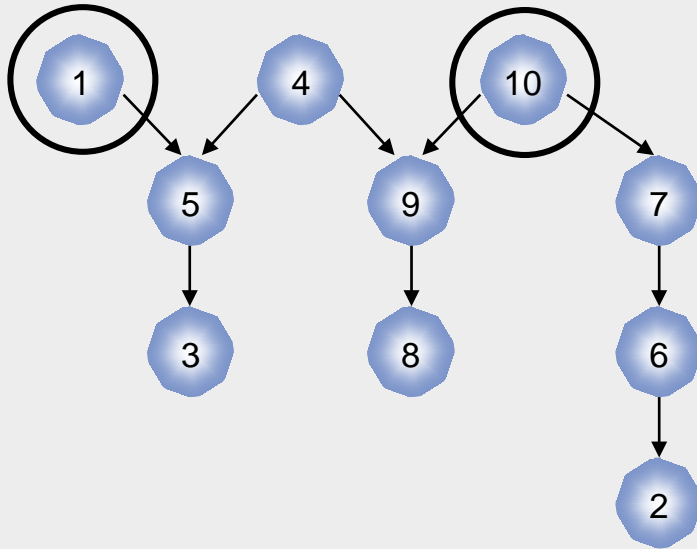


1. Aufbau des VCG und HCG (Zonendarstellung)



1			7	
2			8	
3			9	
	4			10
5	6			

6.5.1 Left-Edge-Algorithmus: Beispiel

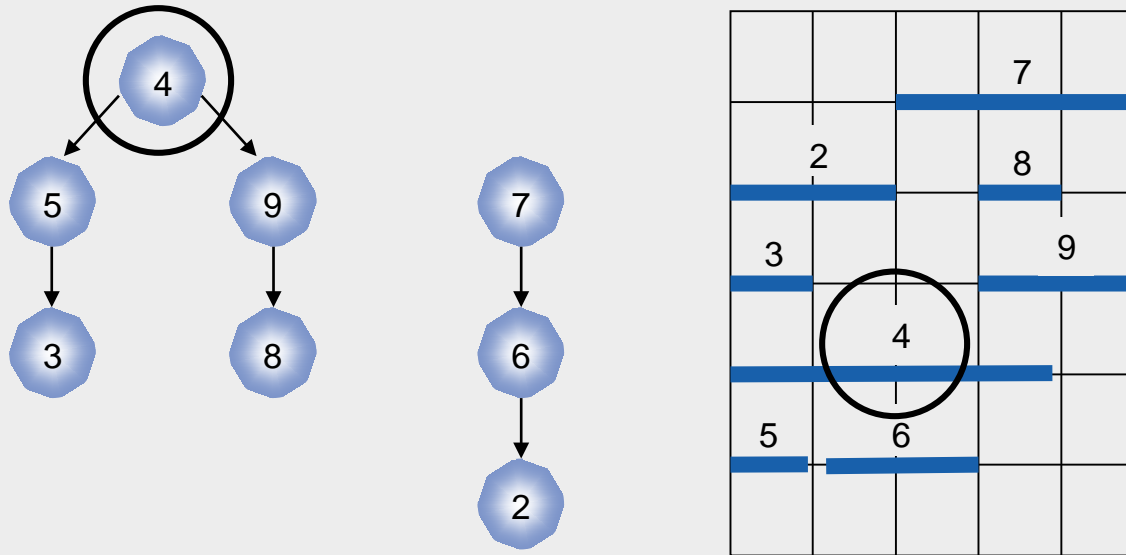


2. Aktuelle Spur $j = 1$ (obere Spur)
3. Für aktuelle Spur j
 - a) Für alle Netze ohne Vorgänger im VCG, Platzierung des am weitesten links liegenden Netzes in der Zonendarstellung auf Spur j und anschließend weitere nicht-überlappende und vorgängerlose Netze

Spur $j = 1$: Netz 1 Netz 10

3. b) Löschen aller platzierten Netze (1, 10) in VCG und Zonendarstellung

6.5.1 Left-Edge-Algorithmus: Beispiel



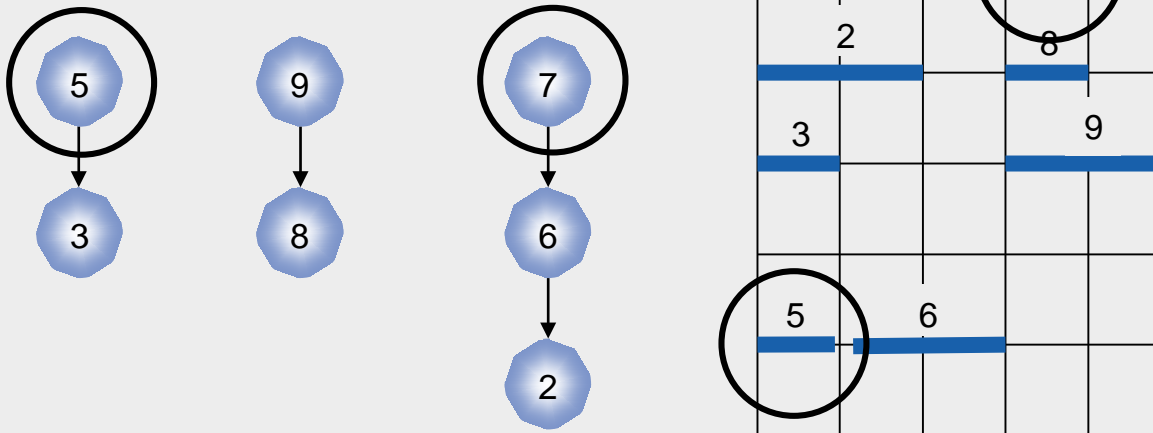
Aktuelle Spur $j = j + 1$, falls noch Netze in VCG, weiter mit **3**.

- 3.** Für aktuelle Spur j
 - a)** Für alle Netze ohne Vorgänger im VCG, Platzierung des am weitesten links liegenden Netzes in der Zonendarstellung auf Spur j und anschließend weitere nicht-überlappende und vorgängerlose Netze

Spur $j = 2$: Netz 4

- 3. b)** Löschen aller platzierten Netze (**4**) in VCG und Zonendarstellung

6.5.1 Left-Edge-Algorithmus: Beispiel



Aktuelle Spur $j = j + 1$, falls noch Netze in VCG, weiter mit **3**.

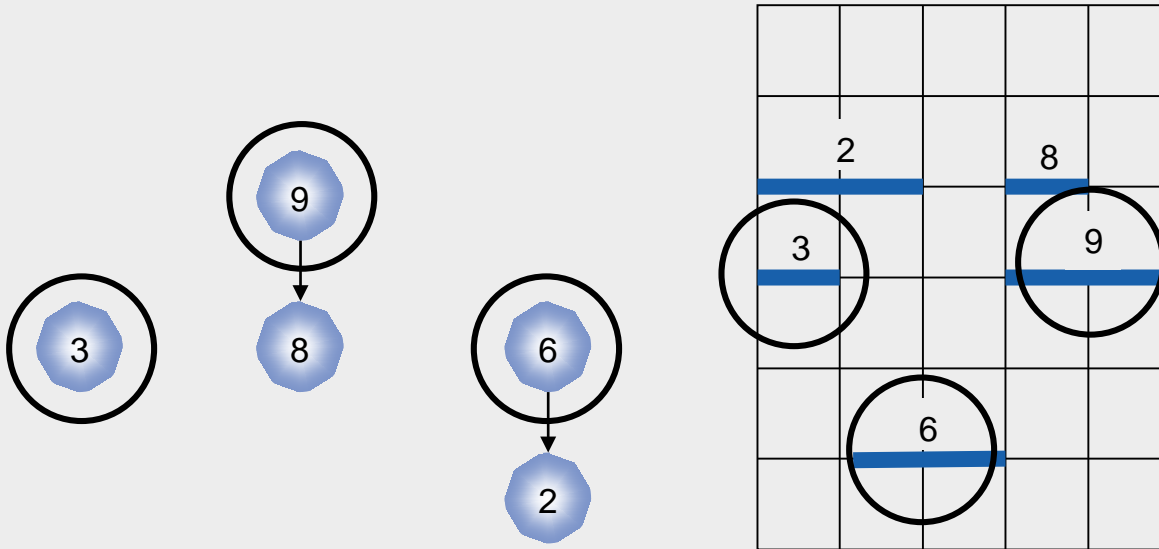
3. Für aktuelle Spur j

- a) Für alle Netze ohne Vorgänger im VCG, Platzierung des am weitesten links liegenden Netzes in der Zonendarstellung auf Spur j und anschließend weitere nicht-überlappende und vorgängerlose Netze

Spur $j = 3$: Netz 5 Netz 7

3. b) Löschen aller platzierten Netze (**5, 7**) in VCG und Zonendarstellung

6.5.1 Left-Edge-Algorithmus: Beispiel



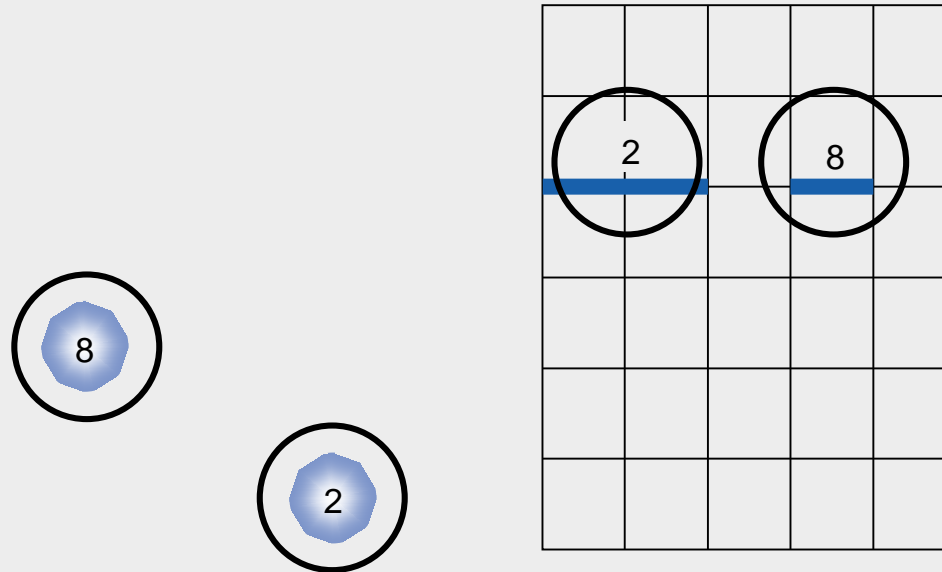
Aktuelle Spur $j = j + 1$, falls noch Netze in VCG, weiter mit **3**.

3. Für aktuelle Spur j
 - a) Für alle Netze ohne Vorgänger im VCG, Platzierung des am weitesten links liegenden Netzes in der Zonendarstellung auf Spur j und anschließend weitere nicht-überlappende und vorgängerlose Netze

Spur $j = 4$: Netz 3 Netz 6 Netz 9

3. b) Löschen aller platzierten Netze (**3, 6, 9**) in VCG und Zonendarstellung

6.5.1 Left-Edge-Algorithmus: Beispiel



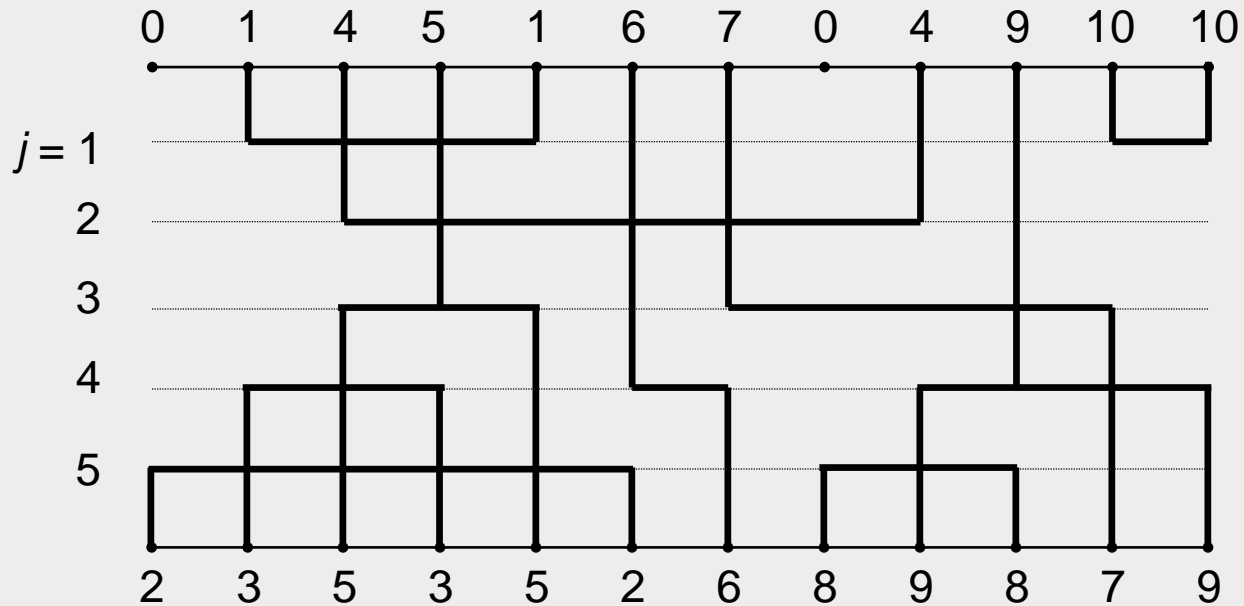
Aktuelle Spur $j = j + 1$, falls noch Netze in VCG, weiter mit **3**.

3. Für aktuelle Spur j
 - a) Für alle Netze ohne Vorgänger im VCG, Platzierung des am weitesten links liegenden Netzes in der Zonendarstellung auf Spur j und anschließend weitere nicht-überlappende und vorgängerlose Netze

Spur $j = 5$: Netz 2 Netz 8

3. b) Löschen aller platzierten Netze (2, 8) in VCG und Zonendarstellung

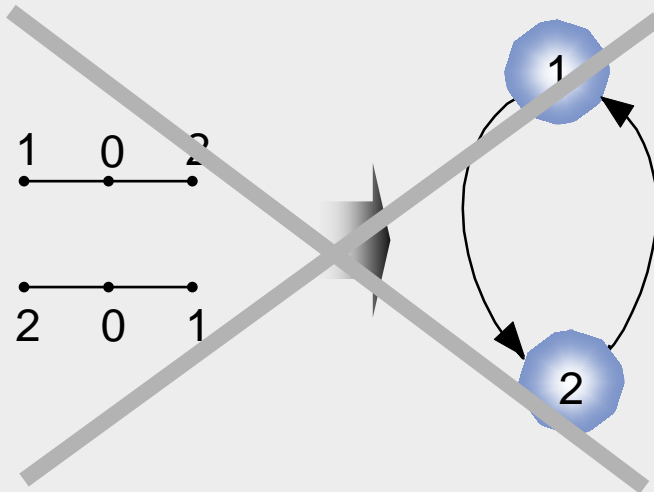
6.5.1 Left-Edge-Algorithmus: Beispiel



Verdrahtungsergebnis

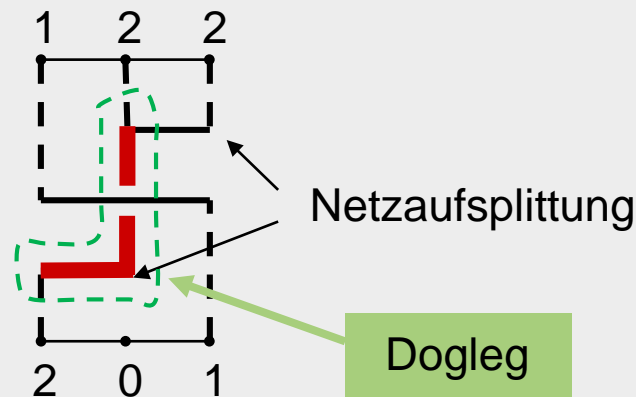
Zusammenfassung

- Einfache Heuristik
- Nachteil: erfordert schleifenfreien VCG, also keine vertikalen zyklischen Anschlusskonflikte

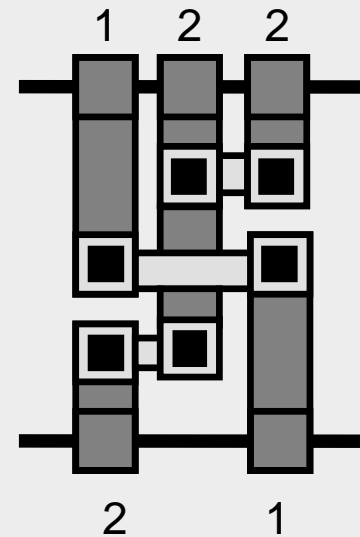
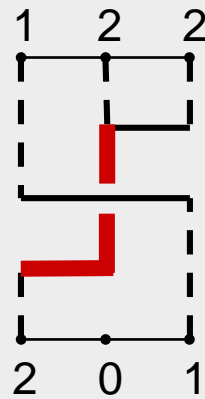
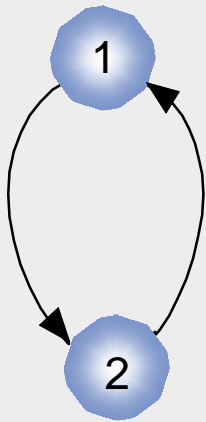


6.5.2 Dogleg-Left-Edge-Algorithmus

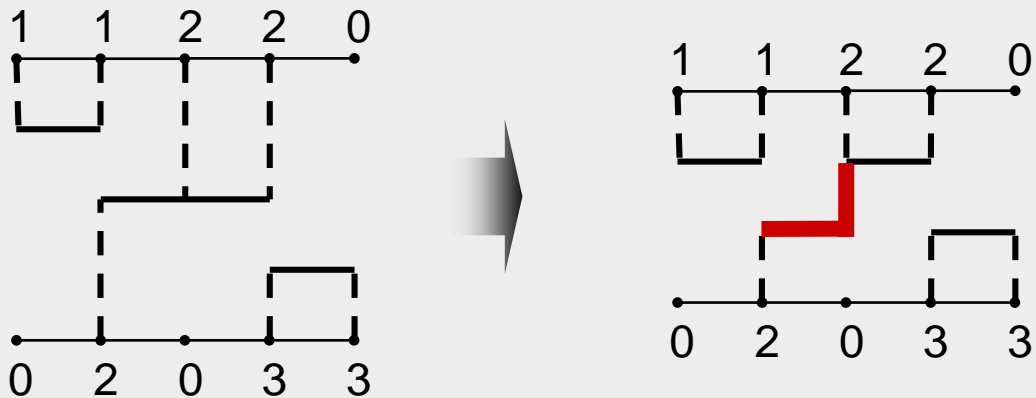
- Erweiterung des Left-Edge-Algorithmus mit Netzaufsplittungen
- Damit zwei wesentliche Vorteile:
 - auch bei zyklischen vertikalen Anschlusskonflikten anwendbar
 - Spureinsparung, d.h. Erhöhung der Kanaldichte



Lösung eines zyklischen Konflikts mittels eines Doglegs

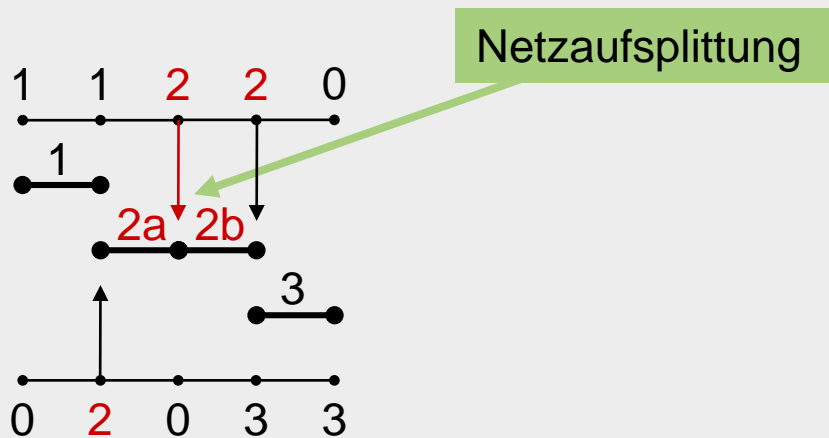


Spureinsparung mittels eines Doglegs

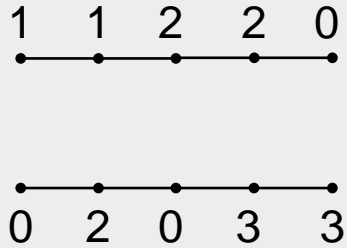


6.5.2 Dogleg-Left-Edge-Algorithmus

- Jedes p -Pin-Netz wird in (maximal) $p-1$ horizontale Segmente zerlegt
- Die Zerlegung erfolgt grundsätzlich an allen Spaltenpositionen, an denen dieses Netz einen Anschluss hat.



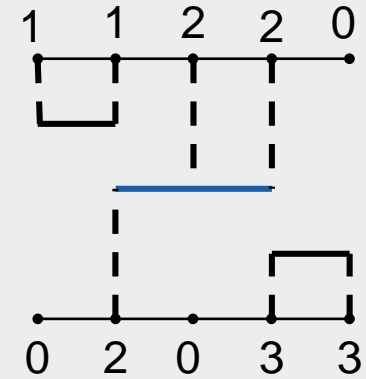
6.5.2 Dogleg-Left-Edge-Algorithmus



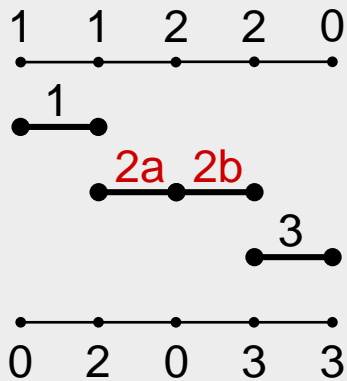
Kanalverdrahtungsproblem



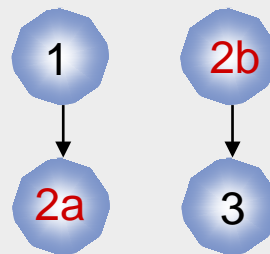
VCG ohne Netzaufspaltung



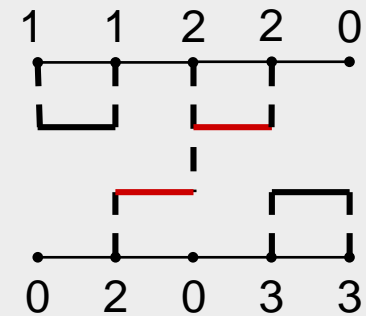
Verdrahtungsergebnis



Netzaufspaltung



VCG mit Netzaufspaltung

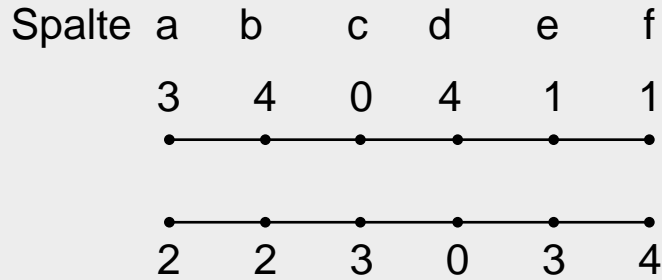


Verdrahtungsergebnis

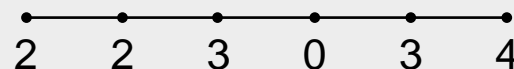
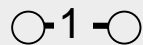
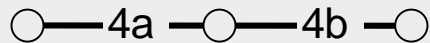
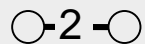
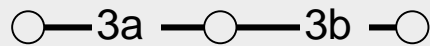
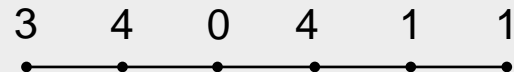
6.5.2 Dogleg-Left-Edge-Algorithmus: Beispiel

Spalte	a	b	c	d	e	f
	3	4	0	4	1	1
	•	•	•	•	•	•
	—————					
	•	•	•	•	•	•
	2	2	3	0	3	4

6.5.2 Dogleg-Left-Edge-Algorithmus: Beispiel



1. Netzaufspaltung und Bestimmung der Zonendarstellung bzw. der (maximalen) Mengen $S(k)$



$$S(a) = \{2, 3a\}$$

$$S(b) = \{2, 3a, 4a\}$$

$$S(c) = \{3a, 3b, 4a\}$$

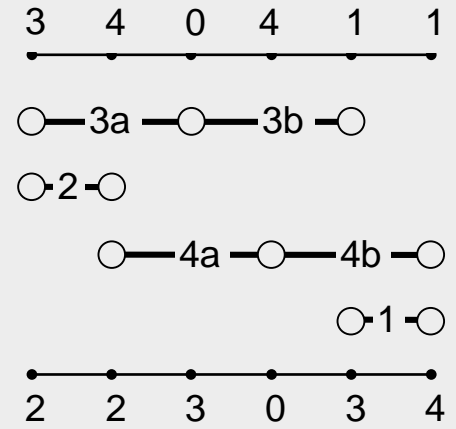
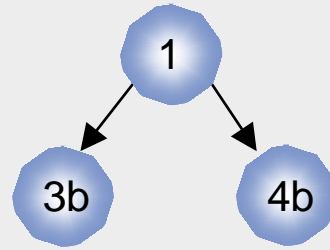
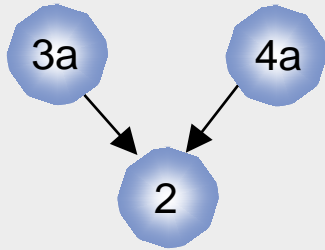
$$S(d) = \{3b, 4a, 4b\}$$

$$S(e) = \{1, 3b, 4b\}$$

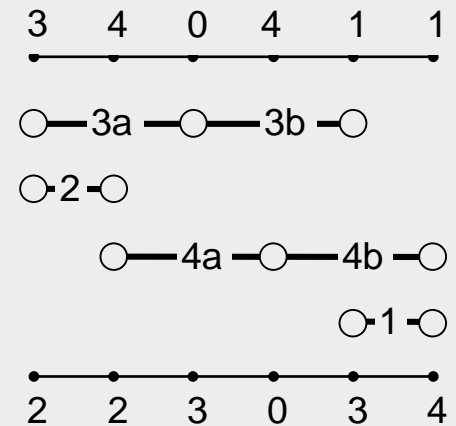
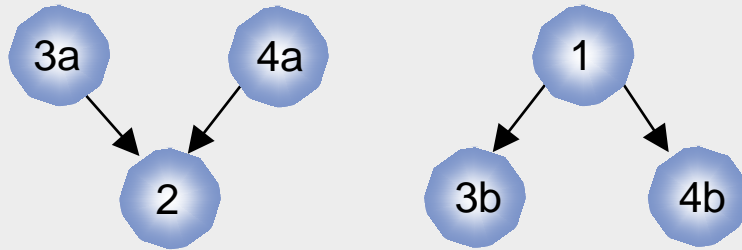
$$S(f) = \{1, 4b\}$$

Hinweis zur Mächtigkeit:
Teilnetze desselben
Netzes dürfen auf gleicher
Spur platziert werden.

2. Ermittlung des vertikalen Verträglichkeitsgraphen (VCG)



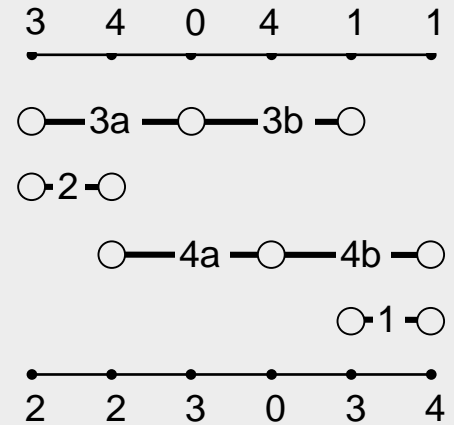
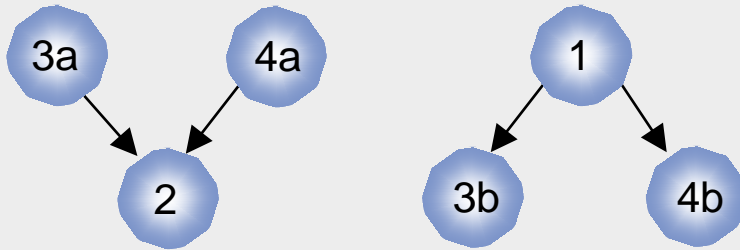
2. Ermittlung des vertikalen Verträglichkeitsgraphen (VCG)



3. Spurzuweisung:

- Spur $j = 1$: Netze 3a, 4a, 1 kommen in Betracht
 - Netz 3a ist links in Zonendarstellung, daher wird Netz 3a zuerst platziert
 - Von den verbleibenden Netzen 4a und 1 hat nur 1 keine Überlappung mit Netz 3a, daher ist 1 ebenfalls auf Spur 1 zu platzieren
 - Erneuerung des VCG.
- Spur $j = 2$: Netze 4a, 3b, 4b kommen in Betracht
 - Netz 4a ist links in Zonendarstellung, daher wird Netz 4a zuerst platziert
 - Von den verbleibenden Netzen 3b und 4b ist nur 4b auf gleicher Spur platzierbar (Teilnetz von Netz 4), daher ist 4b ebenfalls auf Spur 2 zu platzieren
 - Erneuerung des VCG
- Spur $j = 3$: Beide verbleibenden Netze 2, 3b kommen in Betracht
 - Netz 2 ist links in Zonendarstellung, daher sind Netz 2 und anschließend, da nicht überlappend, Netz 3b auf Spur 3 zu platzieren.

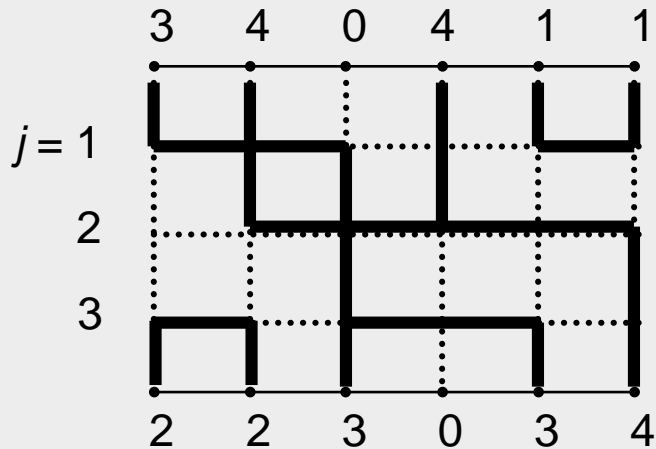
2. Ermittlung des vertikalen Verträglichkeitsgraphen (VCG)



3. Spurzuweisung



4. Ergebnis:



6.5.3 Greedy-Kanalverdrahter (Greedy Channel Router)

- 1982 von *Rivest* und *Fiduccia* vorgestellt.
- Greedy-Kanalverdrahter geht spaltenweise (Column by column) vor.
- Da er im Wesentlichen nur die lokalen Informationen einer Spalte benötigt, ist sein Speicherplatzbedarf unabhängig von der Kanallänge.
- Die spaltenweise Abarbeitung eines horizontalen Kanals beginnt links.
- In jeder Spalte versucht der Algorithmus durch Einsatz intelligenter Heuristiken, die Anzahl der in der folgenden Spalte zur Verfügung stehenden freien Spuren zu maximieren, d.h. Entscheidungen werden in jeder Spalte lokal gefällt.
- Ein Netz kann gleichzeitig mehrere Spuren einnehmen.
- Doglegs können an allen vertikalen Spaltenpositionen eingeführt werden.

Greedy-Algorithmus

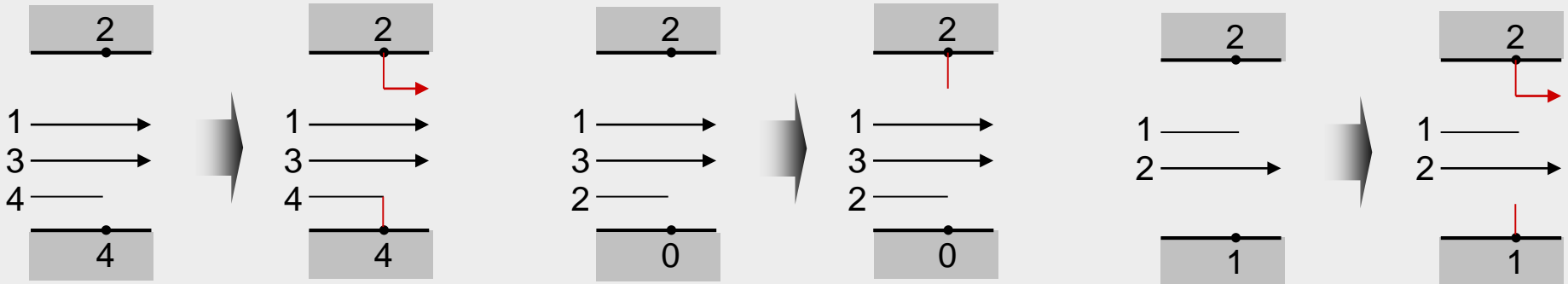
Festlegung einer (anfänglichen) Kanalbreite

Von links beginnend, in jeder Spalte

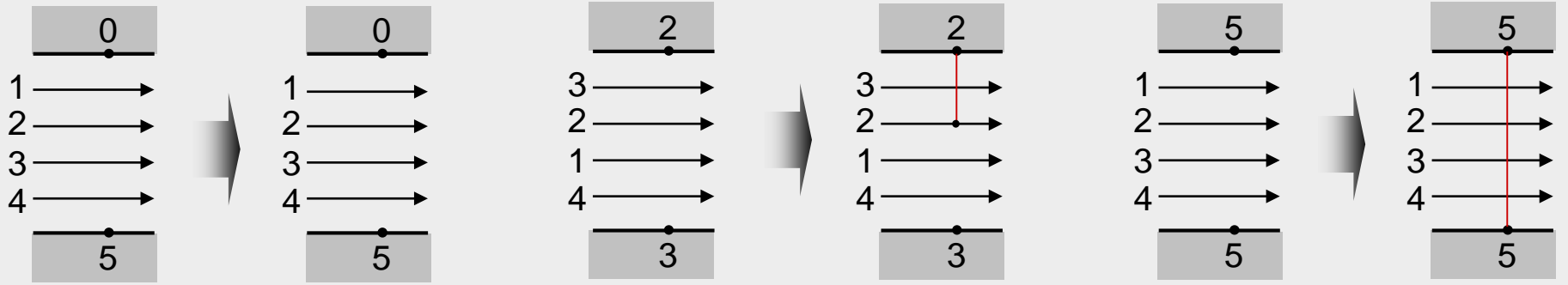
1. Erzeugen optimierter Anschlüsse nach oben und unten
2. Generieren der maximalen Anzahl freier Spuren durch Zusammenführung gespaltener Netze
3. Abstandsverminderung gespaltener Netze
4. Einfügen von Vertikalelementen (Jogs) zur Anschlussausrichtung der Netze
5. Kanalaufweitung zum Anschluss bisher unverbindbarer Pinanschlüsse
6. Übergang zur nächsten Spalte, weiter mit Schritt 1.

6.5.3 Greedy-Kanalverdrahter (Greedy Channel Router)

Schritt 1: Erzeugen optimierter Anschlüsse nach oben und unten



Verschiedene Möglichkeiten der Netzeinführung

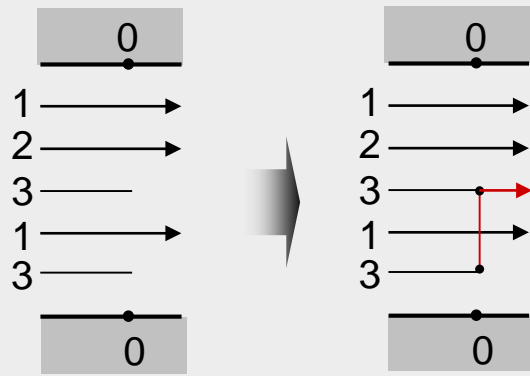


Verschiebung der Netzeinführung

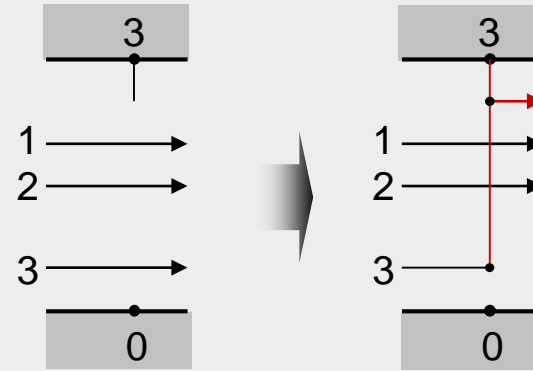
Minimale Verbindungslänge

Direktverbindung

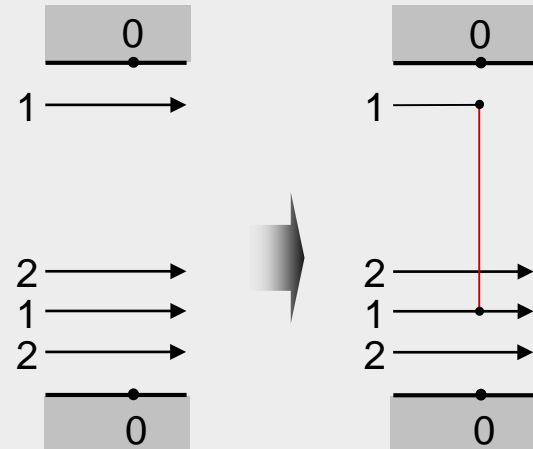
Schritt 2: Zusammenführung gespaltener Netze



Vertikalzusammenführungen

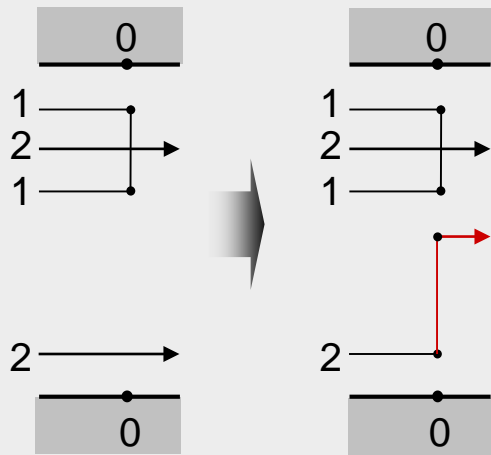


Seitennahe Netze bevorzugt zusammenführen



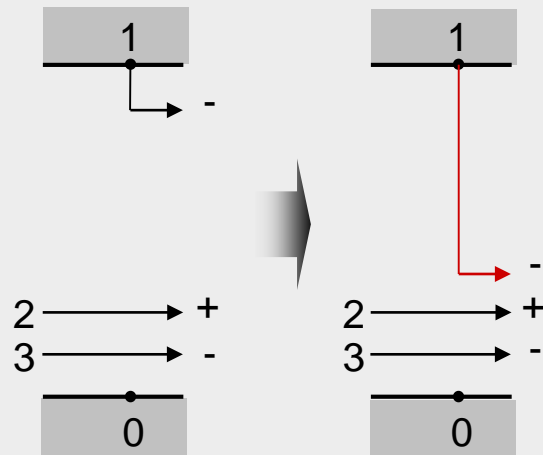
Längere Vertikalstrecke bevorzugen

Schritt 3: Abstandsverminderung gespaltener Netze



6.5.3 Greedy-Kanalverdrahter (Greedy Channel Router)

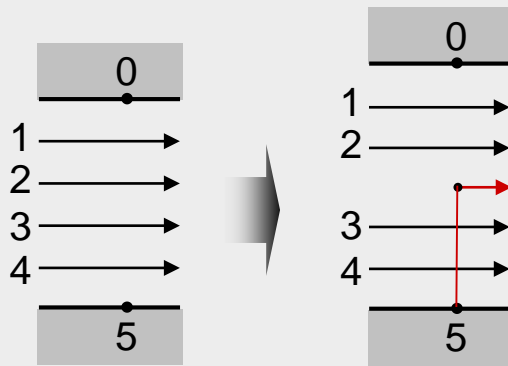
Schritt 4: Einfügen von Vertikalelementen zur Anschlussausrichtung der Netze



- + Nächster Anschluss nach oben
- Nächster Anschluss nach unten

6.5.3 Greedy-Kanalverdrahter (Greedy Channel Router)

Schritt 5: Kanalaufweitung zum Anschluss bisher unverbindbarer Pins

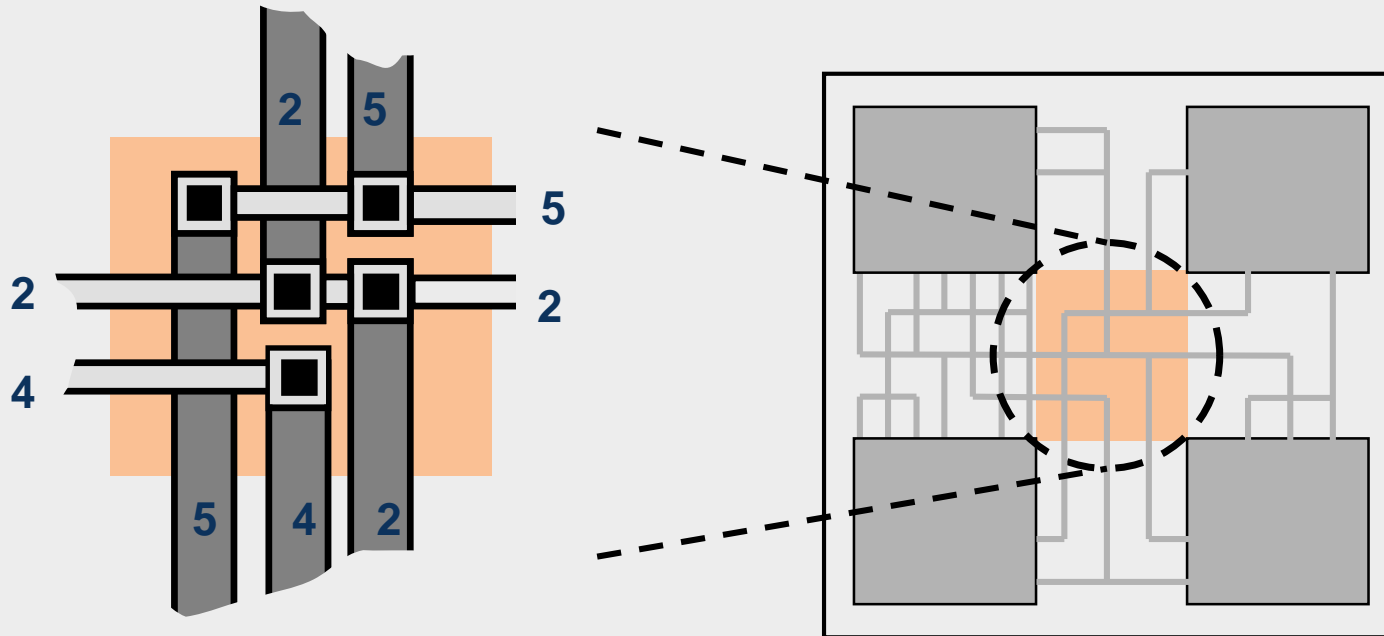


Eigenschaften

- Speicherbedarf unabhängig von Kanallänge
- Keine Verträglichkeitsgraphen, Konflikte haben keine Auswirkungen
- Jeder Kanal kann verdrahtet werden, evtl. Hinzunahme zusätzlicher Spuren
- Doglegs in allen Vertikalspuren, nicht nur in Spalte des Netzanschlusses

- 6.1 Einführung
- 6.2 Begriffsbestimmungen
- 6.3 Horizontaler und vertikaler Verträglichkeitsgraph
 - 6.3.1 Horizontale Verträglichkeitsdarstellung
 - 6.3.2 Vertikale Verträglichkeitsdarstellung
- 6.4 Optimierungsziele
- 6.5 Algorithmen für die Kanalverdrahtung
 - 6.5.1 Left-Edge-Algorithmus
 - 6.5.2 Dogleg-Left-Edge-Algorithmus
 - 6.5.3 Greedy-Kanalverdrahter (Greedy Channel Router)
-  **6.6 Switchbox-Verdrahtung**
- 6.7 OTC-Verdrahtung bei drei Lagen
- 6.8 Multilayer-OTC-Verdrahtung

6.6 Switchbox-Verdrahtung



- Anschlüsse an allen vier Seiten, feste Abmessungen, keine Hindernisse
- Jeder Anschluss definiert durch $TOP(i)$, $BOT(i)$, $LEFT(j)$, $RIGHT(j)$ und Netzkennzeichnung *netz*:
 $LEFT(j) = \textit{netz}$, $RIGHT(j) = \textit{netz}$, $TOP(i) = \textit{netz}$ und $BOT(i) = \textit{netz}$

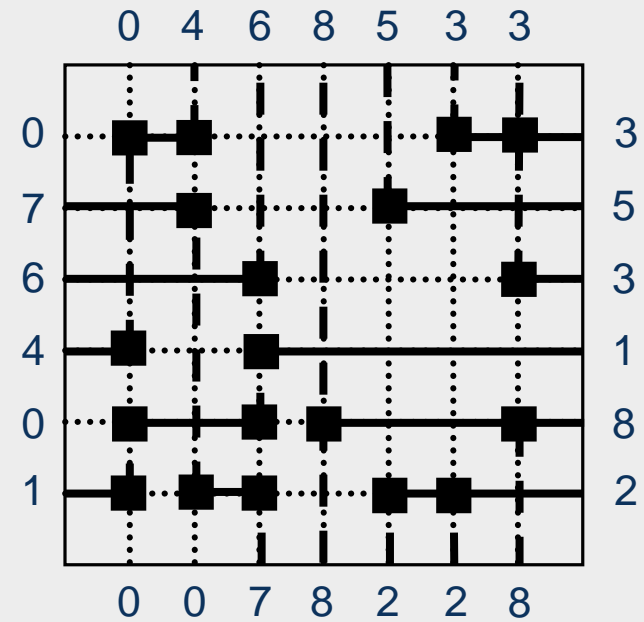
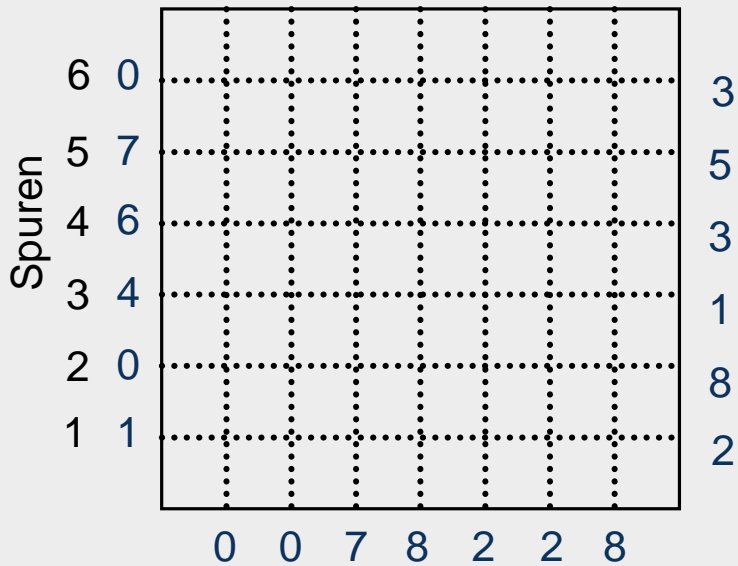
6.6 Switchbox-Verdrahtung

$$R = \{0, 1, 2, \dots, 8\} \times \{0, 1, 2, \dots, 7\}$$

- TOP = (1, 2, ..., 7) = [0, 4, 6, 8, 5, 3, 3]
- BOT = (1, 2, ..., 7) = [0, 0, 7, 8, 2, 2, 8]
- LEFT = (1, 2, ..., 6) = [1, 0, 4, 6, 7, 0]
- RIGHT = (1, 2, ..., 6) = [2, 8, 1, 3, 5, 3]

Spalten

a b c d e f g
0 4 6 8 5 3 3



- *Luk* stellte 1985 einen der am weitesten verbreiteten Verdrahtungsalgorithmen für Switchboxen vor.
- Dabei wurde der Greedy-Kanalverdrahtungsalgorithmus von *Rivest* und *Fiduccia* für das Switchbox-Verdrahtungsproblem weiterentwickelt. Wesentliche Änderungen sind
 - die Zuordnung von Pinanschlüssen auch links und rechts,
 - das direkte Einbringen der linken Pinanschlüsse als horizontale Spuren, und
 - Vertikal-Zusammenführungen (Jogs), die nicht mehr nur für die oberen und unteren Anschlüsse eingebracht werden, sondern auch für horizontale Anschlüsse mit dem Ziel, den rechts liegenden horizontalen Anschluss zu erreichen.

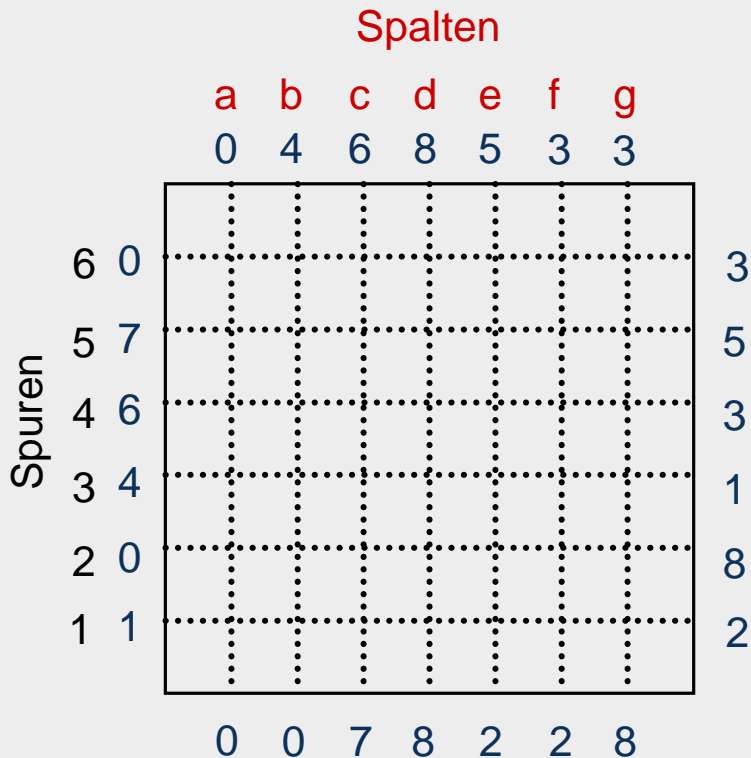
6.6.2 Algorithmen für die Switchbox-Verdrahtung: Beispiel

TOP = (1, 2, ..., 7) = [0, 4, 6, 8, 5, 3, 3]

BOT = (1, 2, ..., 7) = [0, 0, 7, 8, 2, 2, 8]

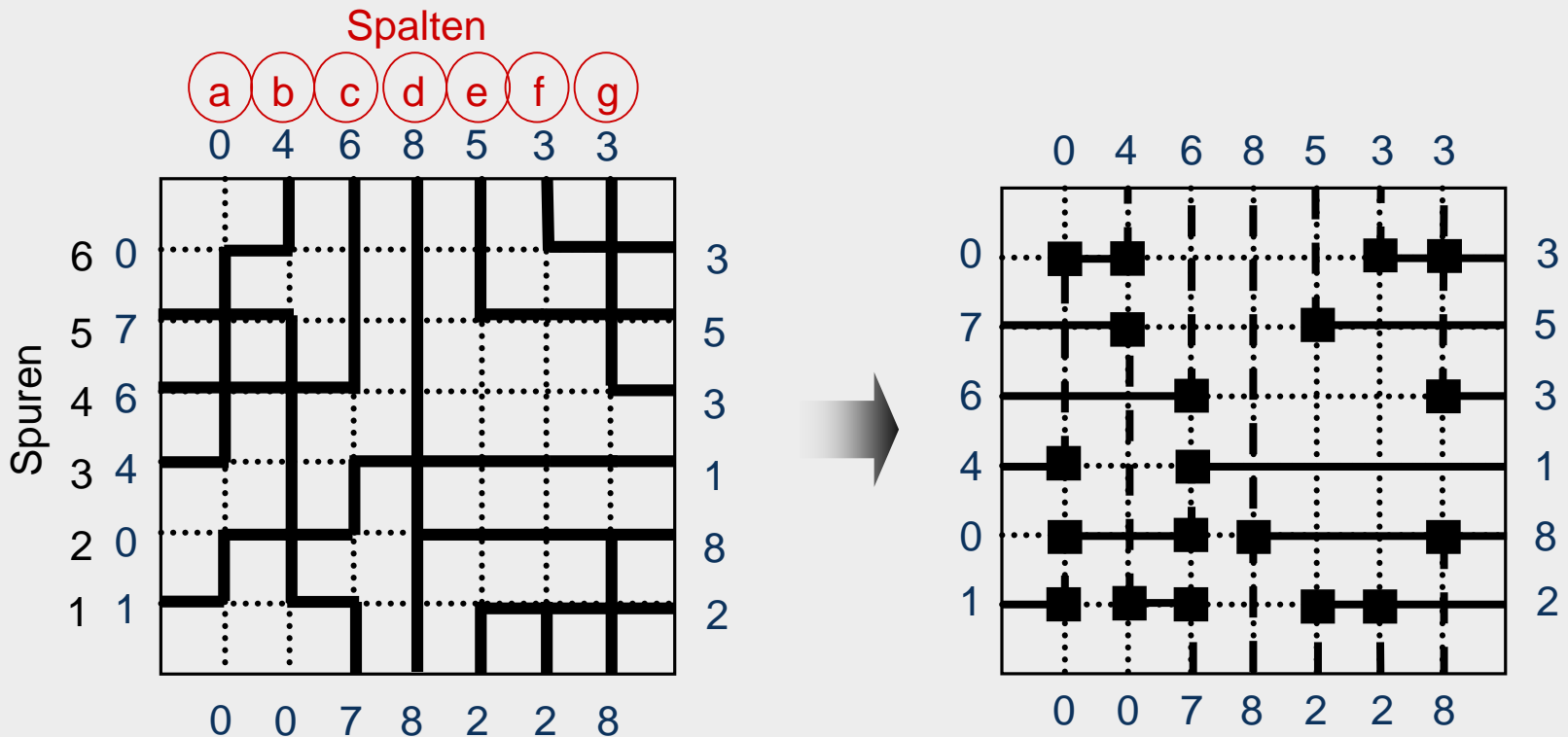
LEFT = (1, 2, ..., 6) = [1, 0, 4, 6, 7, 0]

RIGHT = (1, 2, ..., 6) = [2, 8, 1, 3, 5, 3]



6.6.2 Algorithmen für die Switchbox-Verdrahtung: Beispiel

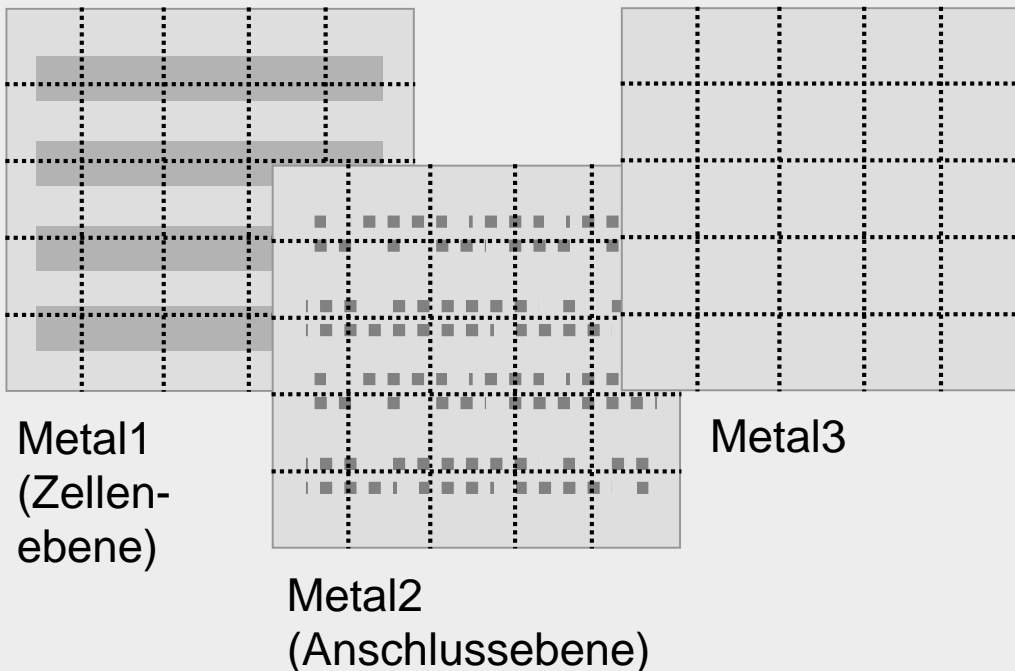
$TOP = (1, 2, \dots, 7) = [0, 4, 6, 8, 5, 3, 3]$
 $BOT = (1, 2, \dots, 7) = [0, 0, 7, 8, 2, 2, 8]$
 $LEFT = (1, 2, \dots, 6) = [1, 0, 4, 6, 7, 0]$
 $RIGHT = (1, 2, \dots, 6) = [2, 8, 1, 3, 5, 3]$



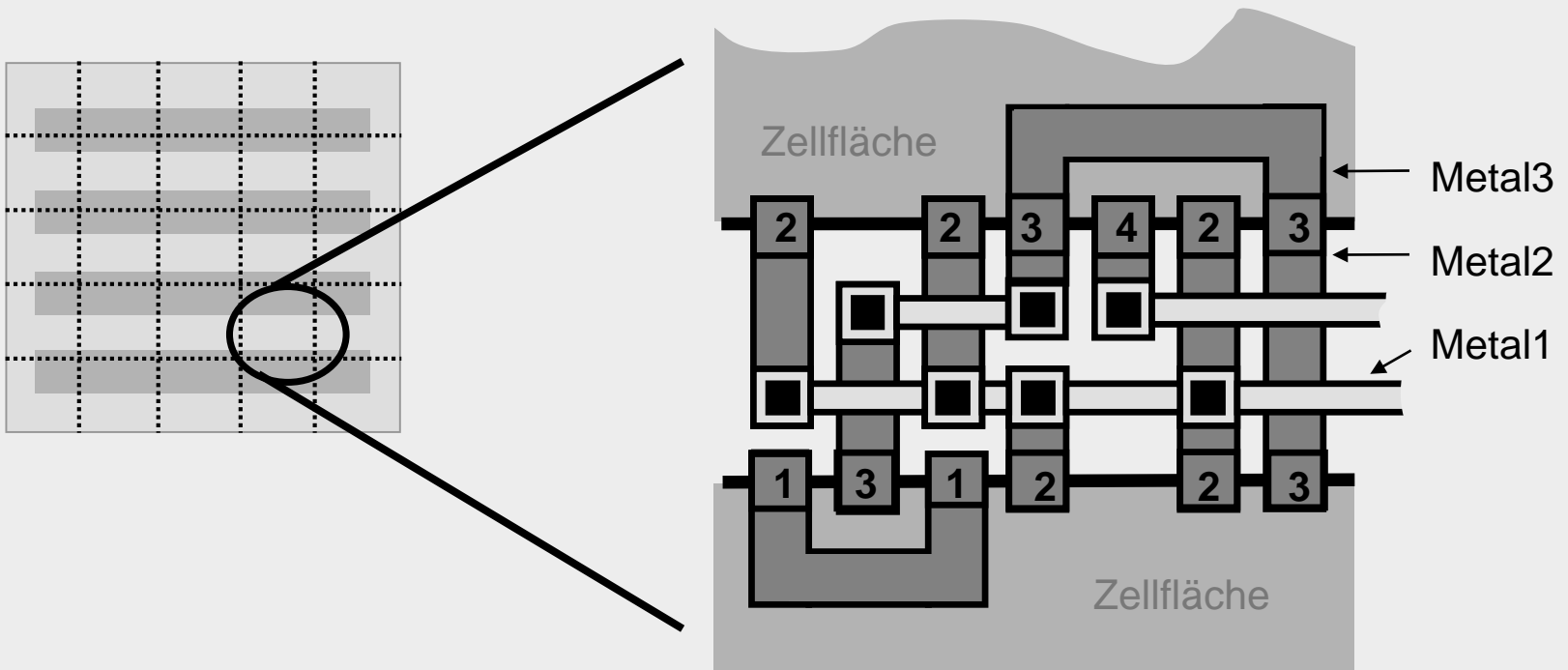
- 6.1 Einführung
- 6.2 Begriffsbestimmungen
- 6.3 Horizontaler und vertikaler Verträglichkeitsgraph
 - 6.3.1 Horizontale Verträglichkeitsdarstellung
 - 6.3.2 Vertikale Verträglichkeitsdarstellung
- 6.4 Optimierungsziele
- 6.5 Algorithmen für die Kanalverdrahtung
 - 6.5.1 Left-Edge-Algorithmus
 - 6.5.2 Dogleg-Left-Edge-Algorithmus
 - 6.5.3 Greedy-Kanalverdrahter (Greedy Channel Router)
- 6.6 Switchbox-Verdrahtung
-  **6.7 OTC-Verdrahtung bei drei Lagen**
- 6.8 Multilayer-OTC-Verdrahtung

6.7 OTC-Verdrahtung bei drei Lagen

- Auf den Ebenen der internen Zellenverdrahtung, d.h. meist Poly und Metal1, werden zwischen den Zellen noch Kanäle angelegt, diese aber nur teilweise zur externen Verdrahtung zwischen den Zellen benutzt.
- Geeignete Netze, die z.B. nur auf einer Kanalseite anzuschließen sind, werden in der darüber liegenden „kanallosen“ Lage (also Metal2, evtl. Metal3) verlegt, wobei dort die gesamte Chipfläche zur Verfügung steht.



- Auf den Ebenen der internen Zellenverdrahtung, d.h. meist Poly und Metal1, werden zwischen den Zellen noch Kanäle angelegt, diese aber nur teilweise zur externen Verdrahtung zwischen den Zellen benutzt.
- Geeignete Netze, die z.B. nur auf einer Kanalseite anzuschließen sind, werden in der darüber liegenden „kanallosen“ Lage (also Metal2, evtl. Metal3) verlegt, wobei dort die gesamte Chipfläche zur Verfügung steht.



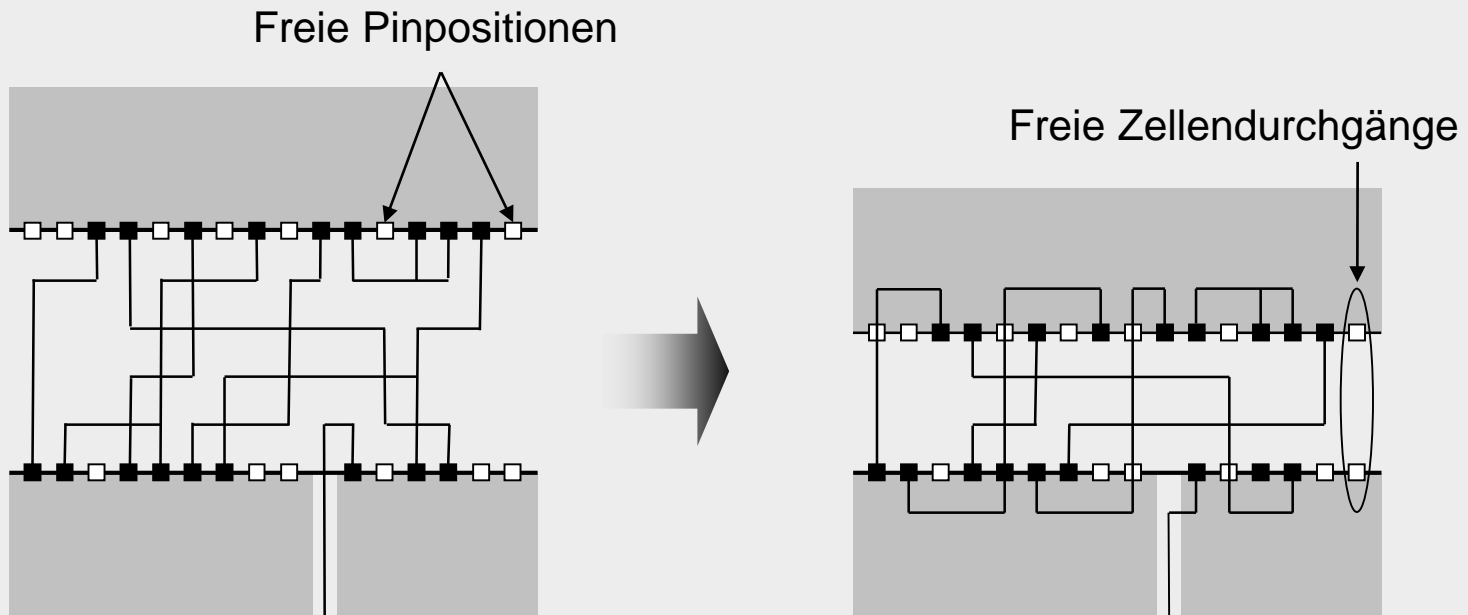
Ablauf (1)

1. Auswahl von Netzen bzw. Netzsegmenten, welche über den Zellen und damit außerhalb des Kanals verdrahtet werden können
2. Verdrahtung dieser Netze bzw. Netzsegmente in der OTC-Fläche
3. Verdrahtung der restlichen Netze bzw. Netzsegmente innerhalb des Kanals

Ablauf (2)

1. Verdrahtung aller Netze bzw. Netzsegmente innerhalb des Kanals
2. Auswahl von Netzen bzw. Netzsegmenten, welche über den Zellen und damit außerhalb des Kanals verdrahtet werden können
3. Verlegung dieser Netze bzw. Netzsegmente in der OTC-Fläche

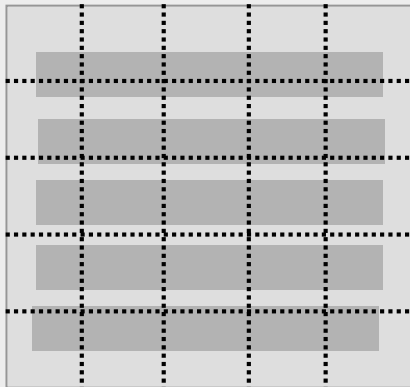
Ablauf (2) – Beispiel



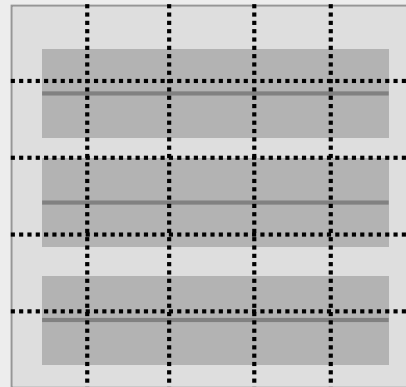
- 6.1 Einführung
- 6.2 Begriffsbestimmungen
- 6.3 Horizontaler und vertikaler Verträglichkeitsgraph
 - 6.3.1 Horizontale Verträglichkeitsdarstellung
 - 6.3.2 Vertikale Verträglichkeitsdarstellung
- 6.4 Optimierungsziele
- 6.5 Algorithmen für die Kanalverdrahtung
 - 6.5.1 Left-Edge-Algorithmus
 - 6.5.2 Dogleg-Left-Edge-Algorithmus
 - 6.5.3 Greedy-Kanalverdrahter (Greedy Channel Router)
- 6.6 Switchbox-Verdrahtung
- 6.7 OTC-Verdrahtung bei drei Lagen
- 6.8 Multilayer-OTC-Verdrahtung**

6.8 Multilayer-OTC-Verdrahtung

- Standardzellen werden ohne Kanalzwischenräume platziert bzw. immer zwei Zellenreihen werden zusammengefasst („Back to back“).



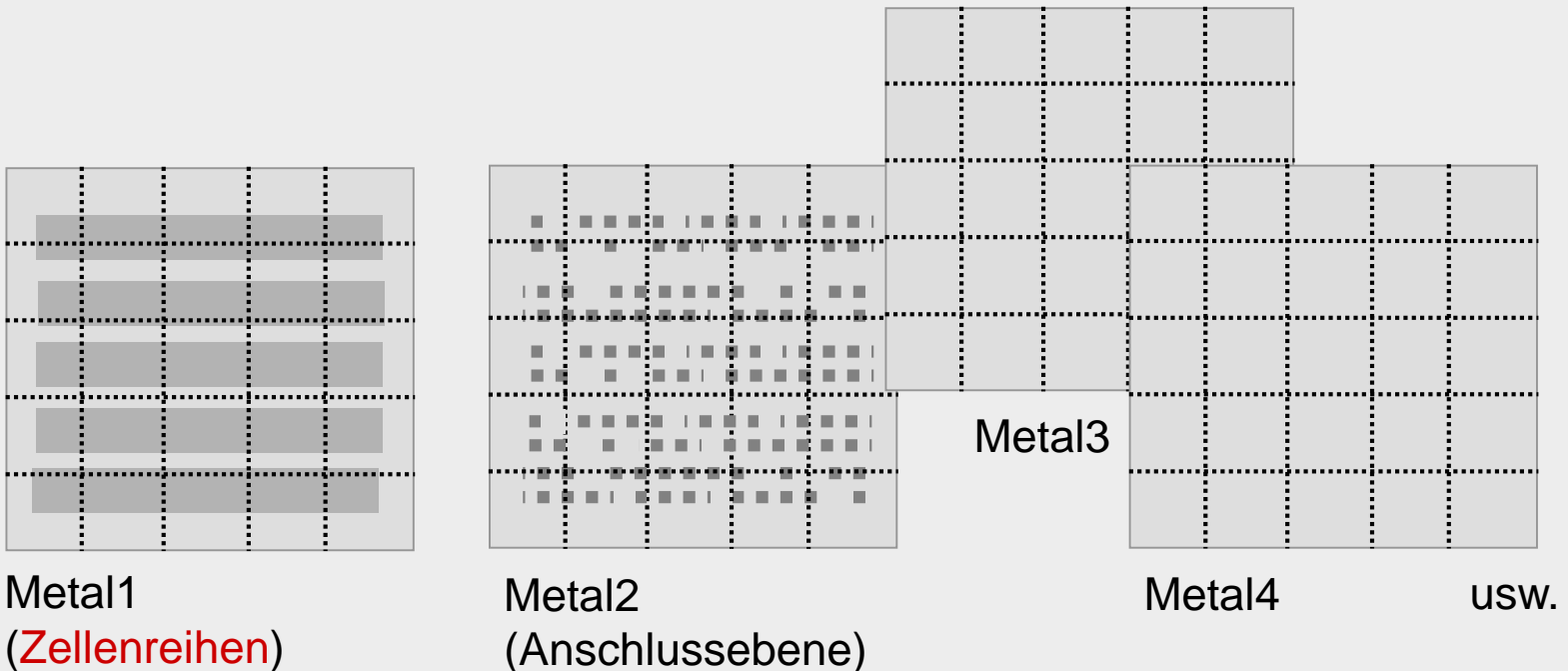
„Ohne“ Zwischenräume



Back to back

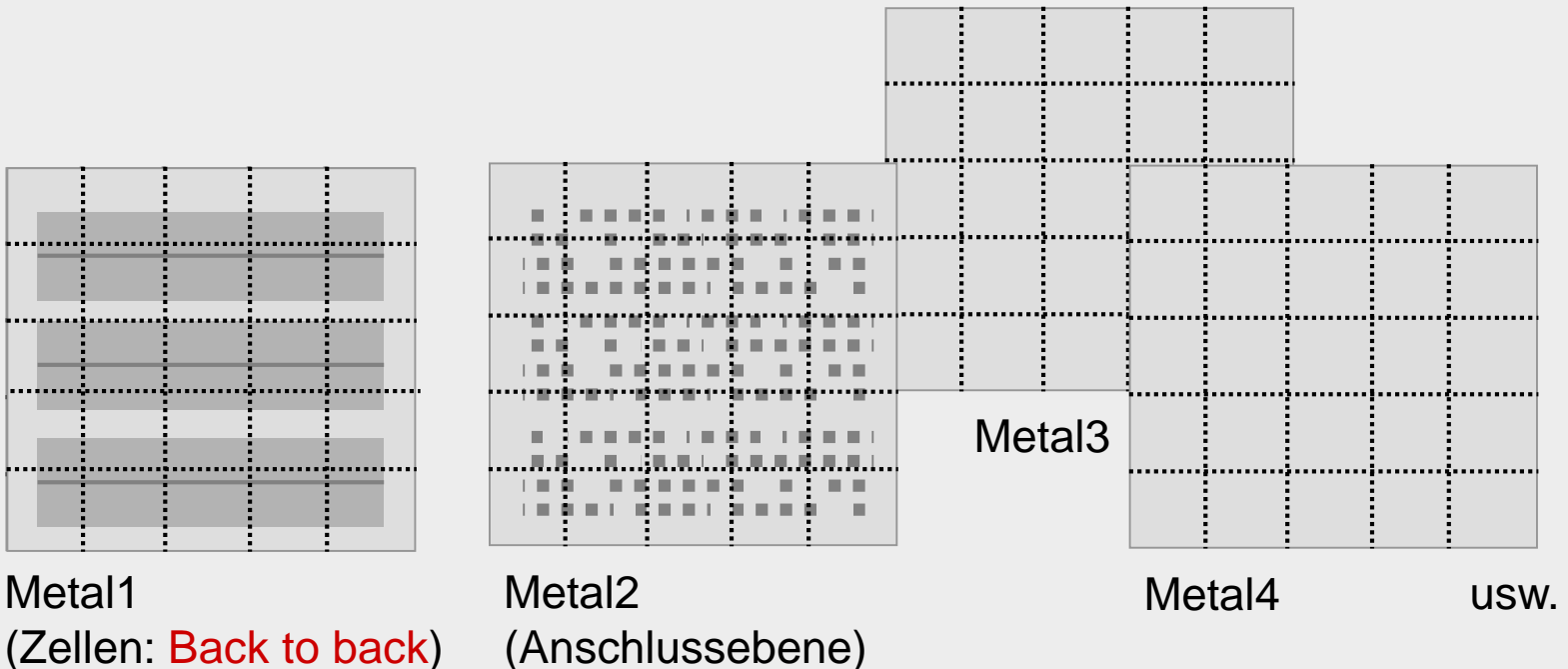
6.8 Multilayer-OTC-Verdrahtung

- Standardzellen werden ohne Kanalzwischenräume platziert bzw. immer zwei Zellenreihen werden zusammengefasst („Back to back“).
- Die darüber liegenden Ebenen Metal2, Metal3 usw. stehen komplett, d.h. ohne Hindernisse, zur externen Verdrahtung der Zellen zur Verfügung.
- Globalverdrahtung im Raster aus gleichmäßig verteilten Rechtecken (Tiles, Boxen), Feinverdrahtung anschließend unter Zusammenfassung der Tiles.



6.8 Multilayer-OTC-Verdrahtung

- Standardzellen werden ohne Kanalzwischenräume platziert bzw. immer zwei Zellenreihen werden zusammengefasst („Back to back“).
- Die darüber liegenden Ebenen Metal2, Metal3 usw. stehen komplett, d.h. ohne Hindernisse, zur externen Verdrahtung der Zellen zur Verfügung.
- Globalverdrahtung im Raster aus gleichmäßig verteilten Rechtecken (Tiles, Boxen), Feinverdrahtung anschließend unter Zusammenfassung der Tiles.



- 6.1 Einführung
- 6.2 Begriffsbestimmungen
- 6.3 Horizontaler und vertikaler Verträglichkeitsgraph
 - 6.3.1 Horizontale Verträglichkeitsdarstellung
 - 6.3.2 Vertikale Verträglichkeitsdarstellung
- 6.4 Optimierungsziele
- 6.5 Algorithmen für die Kanalverdrahtung
 - 6.5.1 Left-Edge-Algorithmus
 - 6.5.2 Dogleg-Left-Edge-Algorithmus
 - 6.5.3 Greedy-Kanalverdrahter (Greedy Channel Router)
- 6.6 Switchbox-Verdrahtung
- 6.7 OTC-Verdrahtung bei drei Lagen
- 6.8 Multilayer-OTC-Verdrahtung