

Yong Zhuo

*Entwurf eines rechnergestützten  
integrierten Systems für Konstruktion  
und Fertigungsplanung  
räumlicher spritzgegossener  
Schaltungsträger (3D-MID)*



Yong Zhuo

*Entwurf eines rechnergestützten  
integrierten Systems für Konstruktion  
und Fertigungsplanung  
räumlicher spritzgegossener  
Schaltungsträger (3D-MID)*

Herausgegeben von

Professor Dr.-Ing. Klaus Feldmann,

Lehrstuhl für

Fertigungsautomatisierung und Produktionssystematik

**FAPS**



Meisenbach Verlag Bamberg

Als Dissertation genehmigt von der Technischen Fakultät  
der Friedrich-Alexander-Universität Erlangen-Nürnberg

Tag der Einreichung: 16. Oktober 2006  
Tag der Promotion: 12. Januar 2007  
Dekan: Prof. Dr.-Ing. A. Leipertz  
Berichterstatter: Prof. Dr.-Ing. K. Feldmann  
Prof. Dr.-Ing. J. Gausemeier, Univ. Paderborn

### **Bibliografische Information Der Deutschen Bibliothek**

Die Deutsche Bibliothek verzeichnet diese Publikation in der  
Deutschen Nationalbibliografie; detaillierte bibliografische Daten  
sind im Internet über <http://dnb.ddb.de> abrufbar.

ISBN 978-3-87525-253-8  
ISSN 1431-6226

Dieses Werk ist urheberrechtlich geschützt.  
Alle Rechte, auch die der Übersetzung, des Nachdrucks  
und der Vervielfältigung des Buches oder Teilen daraus,  
vorbehalten.  
Kein Teil des Werkes darf ohne schriftliche Genehmigung des  
Verlages in irgendeiner Form (Fotokopie, Mikrofilm oder ein  
anderes Verfahren), auch nicht für Zwecke der Unterrichts-  
gestaltung - mit Ausnahme der in den §§ 53, 54 URG ausdrücklich  
genannten Sonderfälle -, reproduziert oder unter Verwendung  
elektronischer Systeme verarbeitet, vervielfältigt oder  
verbreitet werden.

© Meisenbach Verlag Bamberg 2007  
Herstellung: Gruner Druck GmbH, Erlangen-Eltersdorf  
Printed in Germany

Mit Unterstützung der  
Hanns-Seidel-Stiftung und der  
Bayerischen Forschungsstiftung



## Vorwort

Die vorliegende Dissertation entstand während meiner Tätigkeit als Stipendiat der Hanns-Seidel-Stiftung und der Bayerischen Forschungsstiftung am Lehrstuhl für Fertigungsautomatisierung und Produktionssystematik der Friedrich-Alexander-Universität Erlangen-Nürnberg.

Herrn Prof. Dr.-Ing. Klaus Feldmann, dem Leiter dieses Lehrstuhls am Institut für Maschinenbau und Fertigungstechnik, danke ich herzlich für die engagierte Förderung meiner Arbeit und das mir dabei entgegengebrachte Vertrauen. Herrn Prof. Dr.-Ing. Jürgen Gausemeier, dem Leiter des Heiz Nixdorf Instituts der Universität Paderborn, danke ich für die wohlwollende Übernahme des Korreferats.

Bei der Hanns-Seidel-Stiftung und der Bayerischen Forschungsstiftung und ihren Mitarbeitern bedanke ich mich für die besondere Unterstützung im Rahmen der Stipendien. Diese Förderungen haben meine Forschungsarbeit und mein Promotionsstudium in Erlangen ermöglicht.

Bei allen Kollegen am Lehrstuhl bedanke ich mich herzlich für die gute Zusammenarbeit. Besonders hervorheben möchte ich Herrn Dipl.-Wirtsch.-Ing. Christian Alvarez, Herrn Dipl.-Ing. Stefan Junker, Herrn Dipl.-Ing. Tobias Schmuck, Herrn Dipl.-Ing. Matthias Brossog, Herrn Dipl.-Inf. Markus Michl, Frau Dipl.-Inf. Dorothee Buschhüter und Herrn Dipl.-Inf. Matthias Weber für ihre fachlichen Anregungen, Korrekturen und Diskussionsbereitschaft. Ausdrücklich möchte ich mich auch bei Frau Svenja Frenzel bedanken, die mich durch das zeitintensive Korrekturlesen unterstützt hat.

Mein herzlichster Dank gilt meiner Frau Chunli, die all die Jahre Verständnis und immerwährende Geduld für die Entstehung der Arbeit aufgebracht hat. Sie hat mich bei meinen väterlichen Pflichten gegenüber unserer Tochter entlastet. Bei meinen Eltern und Schwiegereltern bedanke ich mich ganz herzlich für die umfassende Unterstützung bei der Erziehung meiner Tochter. Meine Tochter Jiayue bitte ich um Verständnis für die Zeit, die ihr Vater berufsbedingt getrennt von ihr in Deutschland verbracht hat.



# Entwurf eines rechnergestützten integrierten Systems für Konstruktion und Fertigungsplanung räumlicher spritzgegossener Schaltungsträger (3D-MID)

## Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b> .....	<b>1</b>
1.1	Ausgangssituation und Problemstellung .....	1
1.2	Zielsetzung und Aufbau der Arbeit .....	3
<b>2</b>	<b>Stand der Technik in der Konstruktion von MID-Produkten</b> .....	<b>7</b>
2.1	Konstruktionsverfahren für mechatronische Produkte .....	7
2.1.1	Generelles Vorgehen beim Konstruieren nach VDI 2221 .....	7
2.1.2	Mechatronisches Vorgehen nach VDI 2206 .....	8
2.2	Rechnerunterstützung in der mechatronischen Produktentwicklung .....	10
2.2.1	MCAD/ECAD-Kopplung über Austauschdateien .....	10
2.2.2	Kommerzielle Lösungen zur Konstruktion von MID-Produkten .....	12
2.3	Domänenübergreifender Produktdatenaustausch .....	14
2.3.1	STEP-Grundlage und STEP-Anwendungsprotokoll 210 .....	14
2.3.2	Produktdatenaustausch zwischen CAx-Programmen .....	17
2.3.3	Produktmodellbeschreibung im Bereich der Mechatronik .....	19
2.3.4	Integrationsstrategien .....	20
<b>3</b>	<b>Konzeption eines integrierten CAD-Systems für räumliche spritzgegossene Schaltungsträger</b> .....	<b>22</b>
3.1	Übersicht und Anforderungen des MID-Konstruktionsablaufs .....	22
3.1.1	Anforderungen an Konstruktionssysteme für räumliche spritzgegossene Schaltungsträger .....	22
3.1.2	Probleme des konventionellen Konstruktionsprozesses .....	23
3.1.3	Schwächen der kommerziellen Lösungen in der Konstruktion von MID- Produkten .....	24
3.2	Aufbau des MIDCAD-Systems .....	25

3.2.1	Konzept eines integrierten MIDCAD-Systems .....	25
3.2.2	3D-MCAD-System als Fundament des MIDCAD-Systems .....	26
3.2.3	Aufbau des MIDCAD-Systems .....	30
<b>4</b>	<b>Entwicklung automatischer 3D-Routing-Funktionen .....</b>	<b>33</b>
4.1	Übersicht der Verdrahtung in der Elektronik (EDA).....	33
4.1.1	Allgemeines Routingproblem .....	33
4.1.2	Globale Verdrahtung und detaillierte Verdrahtung.....	33
4.2	3D-Autorouting für MID-Produkte.....	34
4.2.1	Unterschiede zwischen Multilayer- und 3D-MID-Routing.....	34
4.2.2	Anforderungen an und Lösungsmöglichkeiten für das 3D-Autorouting....	35
4.3	Entwicklung neuer 3D-Routingalgorithmen für Zweipunktnetze auf räumlichen Schaltungsträgern .....	37
4.3.1	Rasterbasierter 3D-Routingalgorithmus.....	37
4.3.2	Linienbasierter 3D-Routingalgorithmus.....	43
4.3.3	Rasterfreier 3D-Routingalgorithmus .....	53
4.3.4	Topologischer 3D-Routingalgorithmus.....	62
4.3.5	Vergleich und Zusammenfassung der 3D-MID-Routingalgorithmen.....	65
4.4	3D-Rourting für Mehrpunktnetze .....	66
4.4.1	Minimaler Spannbaum (MST) und minimaler Steinerbaum (SMT) .....	66
4.4.2	Lösungsansätze für Mehrpunktnetze im 3D-MID-Routing .....	68
4.5	Realisierung der 3D-Routing-Funktionen in MIDCAD .....	71
4.5.1	Routing-relevante Zusatzfunktionen .....	71
4.5.2	Manuelle 3D-Routing-Funktion .....	73
4.5.3	Vorgehensweise bei der Durchführung des 3D-MID-Routings in MIDCAD . .....	74
<b>5</b>	<b>Entwicklung neuer MID-spezifischer Funktionen .....</b>	<b>76</b>
5.1	3D-Platzierung elektronischer Bauelemente .....	76
5.2	Definition der Netz- und Komponentenliste.....	78
5.3	Überprüfung der Konstruktions- und Fertigungsregeln (DRC/MRC).....	80
5.4	MID-spezifische Funktionen für 2K-Verfahren .....	82

---

5.4.1 Kurze Verfahrenbeschreibung des Zweikomponentenspritzgusses .....	82
5.4.2 Definition der Leiterbahnquerschnittsform für 2K-Verfahren.....	84
5.4.3 Automatische Erzeugung der Geometriemodelle des ersten und zweiten Schusses.....	85
<b>6 Entwicklung eines integrierten, MID-spezifischen Produktmodells .....</b>	<b>87</b>
6.1 Produktmodellierung.....	87
6.2 Konzeption eines integrierten MID-Produktmodells .....	89
6.3 Entwicklung einer mechanisch-elektronisch integrierten Bauelementebibliothek in MIDCAD.....	92
6.3.1 Konzeption einer 3D-Bauelementebibliothek.....	92
6.3.2 Datenbeschreibung von Bauelementen nach STEP AP210.....	94
6.3.3 Bauelementemodellgenerator .....	98
6.4 STEP AP210 als Grundlage für das mechanisch-elektronisch integrierte MID-Produktmodell.....	99
6.4.1 Beschreibung des integrierten Produkts und der Produktstruktur .....	100
6.4.2 Beschreibung von Information über die elektronische Konnektivität .....	102
6.4.3 Beschreibung des 3D-Schaltungsträgers und der 3D-Leiterbahnen .....	104
6.5 Feature als Basis des CAD-CAE integrierten Produktmodells.....	108
6.5.1 Feature-Technologie in CAx-Systemen.....	108
6.5.2 Erstellung einer Feature-Kategorie für räumliche spritzgegossene Schaltungsträger .....	111
6.5.3 Integration der CAE-Informationen in Features für die Spritzgussimulation .....	113
6.5.4 Implementierung der Feature-Technologie in MIDCAD.....	115
<b>7 Integration von CAE- und CAM-Werkzeugen für MID-Produkte .....</b>	<b>118</b>
7.1 Integration der Simulation des Spritzgussprozesses .....	118
7.1.1 Konzept der Integration der Spritzgussimulation .....	118
7.1.2 Direkte Verbindung zwischen MIDCAD und MOLDFLOW Plastics Insight... ..	119
7.1.3 MID-spezifische Materialbibliothek für die Spritzgussimulation .....	124
7.1.4 Integration der Spritzgussimulation im 3D-MID-Design .....	127
7.1.5 Automatische Gestaltoptimierung durch Spritzgussimulation .....	129

7.2 Entwicklung einer CAD/CAM-Kette für ein MID-Bestücksystem .....	130
7.2.1 Bestückssysteme für räumliche spritzgegossene Schaltungsträger .....	130
7.2.2 Konzept der Verbindung zwischen MIDCAD und dem MID-Bestücksystem .....	132
7.2.3 Definition der bestückungsrelevanten Informationen .....	133
7.2.4 Ableitung der Bestückposition der Bauelemente .....	135
7.2.5 Statische Kollisionsprüfung mit verschiedenen Bestückköpfen .....	136
<b>8 Realisierung eines integrierten CAD-Systems für räumliche spritzgegossene Schaltungsträger .....</b>	<b>138</b>
8.1 Prototyp eines integrierten MIDCAD-Systems .....	138
8.1.1 Architektur des Software-Prototyps .....	138
8.1.2 Verknüpfungen zwischen MIDCAD und externen Anwendungen .....	139
8.1.3 Integration von CAx-Systemen zu einer MID-Konstruktionsumgebung .....	141
8.2 Vorgehensweise des 3D-MID-Designs im MIDCAD-System .....	141
8.2.1 MID-Konstruktion in der frühen Konzeptphase und der domänenspezifischen Entwurfsphase .....	141
8.2.2 3D-MID-Design in der Phase der Systemintegration in MIDCAD .....	142
8.2.3 Integrierte Strukturoptimierung beim 3D-MID-Design in MIDCAD .....	145
8.3 Technischer und wirtschaftlicher Nutzen des integrierten MIDCAD-Systems .....	148
8.3.1 Bewertung der technischen Vorteile .....	148
8.3.2 Wirtschaftliche Bewertung .....	150
<b>9 Zusammenfassung und Ausblick.....</b>	<b>151</b>
<b>Summary .....</b>	<b>153</b>
<b>Abkürzungen.....</b>	<b>155</b>
<b>Literaturverzeichnis.....</b>	<b>157</b>
<b>Anhang .....</b>	<b>167</b>

# 1 Einleitung

## 1.1 Ausgangssituation und Problemstellung

Eine der grundlegenden und richtungsweisenden Innovationen im Bereich der Mechatronik ist die direkte stoffliche Integration von mechanischen und elektronischen Funktionen. Diese sog. MID-Technologie (*Molded Interconnect Devices*) kann sowohl auf spritzgegossenen räumlichen Komponenten als auch auf Folienschaltungen und mit keramischen Grundkörpern realisiert werden. Das hierdurch realisierbare technologische Potential sind höhere Integrationsdichten und die Verwendung umweltverträglicher Werkstoffe. Diese noch recht junge Technologie ermöglicht neue konstruktive Lösungen, Reduzierung von Fertigungs- und Montagekosten, Einsparung von Einzelkomponenten und Gewichten sowie eine höhere Funktionssicherheit. Dies ermöglicht es, neue Rationalisierungspotentiale zu erschließen [25], [26].

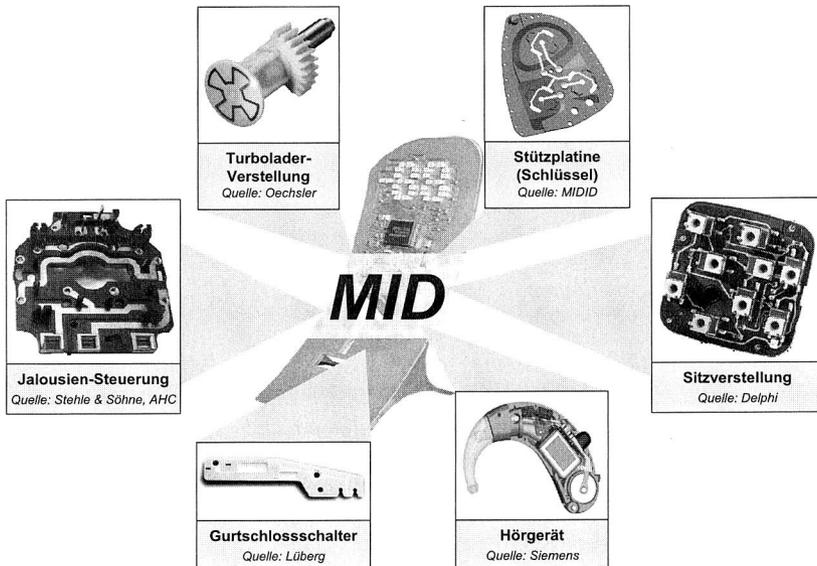


Bild 1: Aktuelle Anwendungsbeispiele der MID-Technologie [27], [28]

Einige Beispiele für aktuelle Anwendungen dieser Technologie sind in Bild 1 dargestellt. Die gezeigten Produkte weisen durch das Zusammenspiel von elektronischer und mechanischer Funktion teilweise einen erheblichen Komplexitätsgrad auf, welcher ein entsprechend effektives, rechnergestütztes Konstruktionssystem bei der Produktgestaltung erfordert. Die bisher verfügbaren Entwicklungswerkzeuge sind jedoch nicht in der Lage, das Potential dieser Funktionsintegration vollständig auszuschöpfen. Dies liegt daran, dass der Entwickler gezwungen ist, weiterhin mit getrennten Werkzeugen für die elektri-

schen und mechanischen Teilbereiche in der Entwicklung zu arbeiten. Die Verbundsysteme aus Elektronik und Mechanik lassen sich mit den bisher erhältlichen CAD-Systemen nicht effizient entwickeln, obwohl die derzeit in der Mechanik- bzw. Elektronikonstruktion eingesetzten rechnergestützten Werkzeuge für die Produktentwicklung in ihrem spezifischen Anwendungsbereich bereits einen hohen Reifegrad erreicht haben. So werden z. B. für den mechanischen Produktanteil MCAD-Systeme eingesetzt, die weitgehend beliebige geometrische Strukturen erzeugen und verarbeiten können. Der elektronische Anteil eines Produkts wird in ECAD-Systemen logisch entworfen und zweidimensional konstruiert. Eine echte dreidimensionale Konstruktion eines MID-Produkts konnte hingegen in verfügbaren ECAD-Systemen bislang nicht zufrieden stellend vorgenommen werden [25].

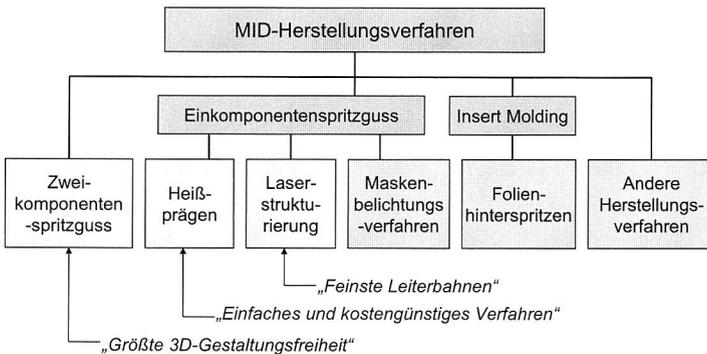


Bild 2: Herstellungsverfahren für MID-Produkte [86]

MID-Produkte können mittels unterschiedlicher Herstellungsverfahren hergestellt werden. Dabei unterscheidet man im Wesentlichen zwischen dem Zweikomponentenspritzguss (2K-Spritzguss), dem Einkomponentenspritzguss wie dem Heißprägen, der Laserdirektstrukturierung, dem Maskenbelichtungsverfahren und dem Folienhinterspritzen, z. B. In-Mould-Decoration (IMD) (siehe Bild 2). Da die existierenden Entwicklungswerkzeuge keine speziellen Module bzw. Funktionalitäten für MID-Produkte enthalten, ist es notwendig, spezielle Werkzeuge unter Berücksichtigung der oben vorgestellten Herstellungsverfahren - insbesondere für das 2K-Verfahren und die Laserstrukturierung - anzubieten, um eine fertigungsorientierte MID-Konstruktion zu unterstützen.

Ein optimales MID-Design wird von zahlreichen Faktoren aus der Produktion, dem Markt, dem Herstellungsverfahren usw. beeinflusst (siehe Bild 3). Um den Möglichkeiten und Vorteilen der MID-Technologie gerecht zu werden, die sich durch die Integration elektronischer und mechanischer Funktionen in einem Bauteil ergeben, ist es sinnvoll, die traditionelle Aufteilung der Aufgaben bei der Entwicklung der Produkte aufzuheben. Hierzu sollten Designer, Formteil- und Werkzeugkonstrukteure sowie Elektronikentwickler schon in der Konzeptphase eng zusammen arbeiten. Um dies zu unterstützen und ein MID-gerechtes Design zu ermöglichen, sollte in der Phase der System-

integration ein integriertes, praxisgerechtes Konstruktionswerkzeug zur Verfügung stehen.

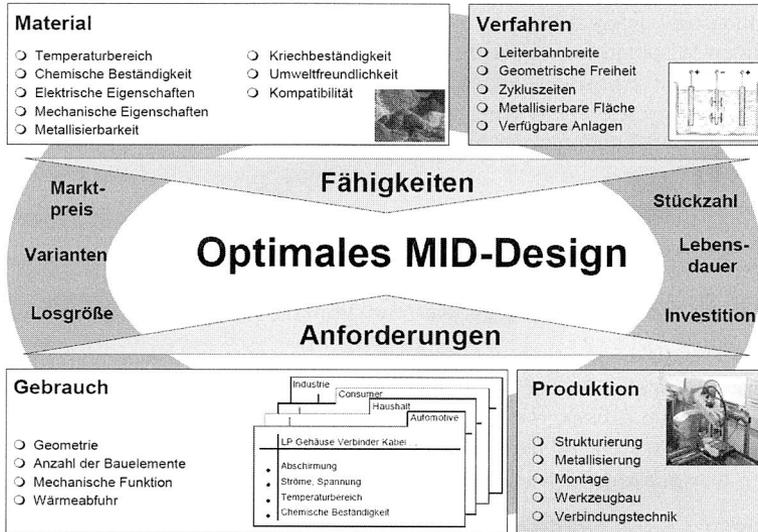


Bild 3: Einflussfaktoren auf das MID-Design [86]

Vor dem Hintergrund der wachsenden Verbreitung und einer damit einhergehenden zunehmenden wirtschaftlichen Bedeutung der 3D-MID-Technologie ist daher die Notwendigkeit der Entwicklung entsprechender Konstruktionswerkzeuge offensichtlich.

## 1.2 Zielsetzung und Aufbau der Arbeit

Die Entwicklung mechatronischer Systeme und die dadurch erforderliche engverzahte Zusammenarbeit unterschiedlicher Fachgebiete stellen neue Anforderungen an das Vorgehen und die eingesetzten Werkzeuge im Entwicklungsablauf [36]. Derzeit existieren für mechatronische Systeme neben grundsätzlichen Vorgehensmodellen (z. B. VDI 2206 [39]) keine konkreten Leitfäden oder Regelbasen, die dem Entwickler zu einer Aufgabe und bestehenden Randbedingungen, z. B. Empfehlungen für den Entwicklungsprozess inkl. durchzuführender Simulationen und Tests, geben, oder erfolgreiche Lösungen aus früheren Projekten vorschlagen.

Im Rahmen der vorliegenden Forschungsarbeit wird daher ein Beitrag zur Unterstützung des Produktentwicklungsprozesses für mechanisch-elektronisch integrierte Produkte, insbesondere für räumliche spritzgegossene Schaltungsträger, geleistet. Mit dem Ziel der Effizienzsteigerung in der Konstruktionsphase von MID-Produkten werden einerseits die MID-Produktmodellbildung und die MID-spezifischen Funktionalitäten, wie z. B. das dreidimensionale Autorouting, realisiert, andererseits aber auch die bisher

existierenden und weitgehend isoliert voneinander verwendeten Systeme für die MID-Produktentwicklung, wie z. B. ECAD, MCAD, CAE etc., zu einem neuartigen, integrierten MIDCAD-System verbunden. Auf Basis des neuartigen MIDCAD-Systems und der Verbindung bzw. Integration von CAx-Systemen und den anderen am Lehrstuhl Fertigungsautomatisierung und Produktionssystematik (FAPS) entwickelten Systemen wird eine integrierte Konstruktionsumgebung angeboten, um den MID-Entwicklungsprozess vom Konzept bis zur Herstellung zu unterstützen. Das zu entwickelnde MIDCAD-System wird dabei prototypisch realisiert und an systematisch ausgewählten Fallbeispielen erprobt.

Die wesentlichen Forschungs- bzw. Entwicklungsarbeiten sind in die folgenden Problembereiche gegliedert:

### **Entwicklung und Optimierung der Kernfunktionen in MIDCAD**

Die wesentliche Problemstellung der Konstruktion von MID-Produkten konzentriert sich auf die Definition der Bauelemente und Leiterbahnverläufe auf bzw. innerhalb eines dreidimensionalen Grundkörpers. Dabei gehören das 3D-Routing zwischen verschiedenen Anschlusspunkten und die 3D-Platzierung der Bauelemente zu den grundlegenden Kernfunktionen in einem MID-Konstruktionssystem.

Ein Schwerpunkt der vorliegenden Arbeit ist daher die Realisierung einer automatischen 3D-MID-Routing-Funktion. Da das 3D-MID-Routing ein komplexer und zeitaufwändiger Vorgang ist, ist die Realisierung des 3D-Autoroutings neben einer manuellen Routingfunktion ein wichtiger Vorteil des MIDCAD-Systems. Im Rahmen dieses Arbeitspunkts werden unterschiedliche Routingmethoden bzw. -algorithmen für das Routing in 3D-Grundkörpern, insbesondere für komplexere Teile und Oberflächen, untersucht und weiterentwickelt. Durch Unterstützung der interaktiven, automatischen Verdrahtung kann hierbei ein wesentlicher Produktivitätsfortschritt erreicht werden.

### **Entwicklung neuer MID-spezifischer Funktionsmodule**

Neben der Realisierung der 3D-Layoutfunktionen in MIDCAD sind weiterhin verschiedene MID-spezifische Funktionen für eine MID-gerechte Konstruktion notwendig. Somit kann das MIDCAD-System aufgrund seiner speziell auf die MID-Technologie ausgerichteten Funktionen einen wertvollen Beitrag zur Effizienzsteigerung bei der Konstruktion von MID-Produkten leisten. Weiterhin werden im Rahmen dieses Arbeitspunkts neue fertigungsorientierte Funktionen für angepasste Herstellungsverfahren der MID-Technologie entwickelt, wie z. B. die Entwicklung der neuen Funktionsmodule für das 2K-Verfahren

### **Integrierte MID-spezifische Produktmodellbildung**

Die Notwendigkeit von Produktmodellen für eine weitergehende Rechnerunterstützung bei der Konstruktion ist seit langem unumstritten. In MID-Produkten kommt eine Vielzahl

elektrischer, elektronischer und mechanischer Elemente zur Anwendung, die funktionell eng miteinander verknüpft sind und gemeinsam die erforderliche Gesamtfunktion erfüllen. Die Beschreibung dieser Daten im Produktmodell in organisatorischer, technischer und technologischer Hinsicht bildet einen weiteren Forschungsschwerpunkt in der vorliegenden Arbeit. Das Potential zur Effizienzsteigerung mittels des MIDCAD-Systems, das diese Arbeit ausschöpfen werden soll, liegt nicht nur in der Entwicklung der speziellen Funktionalitäten im System, sondern auch in der Entwicklung eines integrierten MID-Produktmodells. Dazu gehören auch die Entwicklung einer elektronischen 3D-Bauelementebibliothek und einer MID-spezifischen Featurebibliothek.

Darüber hinaus stellt der Aufbau eines solchen integrierten, domänenübergreifenden MID-Produktmodells eine wichtige Voraussetzung für die Verwirklichung einer integrierten Umgebung zur MID-Produktentwicklung dar. Kommunikationsbarrieren, die durch fachbereichsspezifische Begrifflichkeiten und Notationen historisch gewachsen sind, können durch eine einheitliche Produktdatenbeschreibung überwunden werden.

### Integration von Berechnungs-, Simulations- und Fertigungsapplikationen

Für die Sicherstellung der geforderten MID-Produktqualität in einer frühen Entwicklungsphase spielen Simulationen eine wesentliche Rolle, so z. B. die Spritzgussimulation, die thermische Simulation oder die 3D-Kinematiksimulation. Die Integration bzw. Verknüpfung dieser Simulationswerkzeuge ist eines der Ziele einer integrierten Konstruktionsumgebung für MID-Produkte. Im Rahmen dieses Arbeitspunkts wird die Spritzgussimulation in MIDCAD integriert und eine CAD/CAM-Schnittstelle zwischen MIDCAD und dem MID-Bestücksystem entwickelt. Für die Generierung der Berechnungs-, Simulations- und Fertigungsdaten aus der Konstruktion der MID-Produkte wird das oben genannte integrierte Produktdatenmodell verwendet.

#### Kapitel:

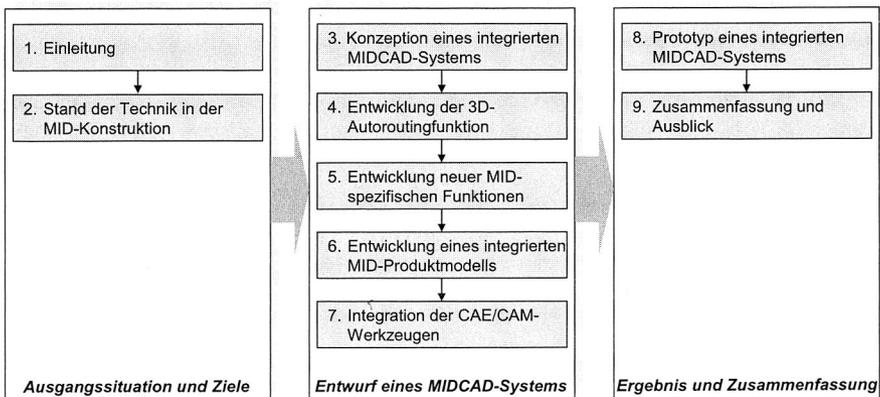


Bild 4: Aufbau der Arbeit bzw. Entwicklungsschwerpunkte

Die oben dargestellten Arbeitsschwerpunkte spiegeln sich in der Gliederung der Arbeit wieder. Bild 4 zeigt eine Übersicht über den Aufbau der Arbeit.

In Kapitel 2 werden bestehende Lösungsansätze für die Konstruktion von MID-Produkten vorgestellt. Dabei werden Konstruktionsverfahren für mechatronische Produkte, Technologien für den Produktdatenaustausch sowie die Integrationsstrategie behandelt. In diesem Kapitel werden auch die Grundlagen von STEP erläutert.

In Kapitel 3 werden die neuen Anforderungen an das MID-Konstruktionssystem, die Probleme der konventionellen Kopplung zwischen ECAD und MCAD sowie die Schwächen der kommerziellen Lösungen analysiert. Aus der Analyse werden das Konzept und der Aufbau des neuartigen integrierten MIDCAD-Systems abgeleitet.

Die Entwicklung der verschiedenen automatischen 3D-MID-Routingalgorithmen bzw. Methoden wird in Kapitel 4 beschrieben. In diesem Kapitel werden auch das manuelle Routing und die routingrelevanten Funktionen vorgestellt. Der Bedeutung für das MID-CAD-System entsprechend wird die Realisierung der 3D-MID-Routing-Funktion sehr ausführlich dargestellt. In Kapitel 5 werden einige typische MID-spezifische Funktionen beschrieben.

Als ein weiterer wichtiger Forschungsschwerpunkt wird in Kapitel 6 die Entwicklung eines integrierten, MID-spezifischen Produktmodells präsentiert. Dafür werden die Konzeption und die Anwendung von STEP AP210 sowie die Feature-Technologie zum Aufbau des integrierten Produktmodells behandelt. In diesem Kapitel werden auch die Entwicklung einer 3D-Bauelementebibliothek, die Kategorisierung der Features und die Definition der MID-spezifischen Features für räumliche spritzgegossene Schaltungsträger beschrieben.

Kapitel 7 behandelt zunächst die Integration der Spritzgusssimulation. Hierzu wird die Entwicklung der Programmierschnittstelle zwischen MIDCAD und MOLDFLOW, und der Materialbibliothek in MIDCAD vorgestellt. Ebenfalls präsentiert wird eine Methode zur automatischen Gestaltoptimierung mittels Spritzgusssimulation. Im Weiteren wird die Entwicklung einer CAD/CAM-Kette zwischen MIDCAD und dem MID-Bestücksystem vorgestellt. In diesem Zusammenhang wird die Entwicklung von zwei relevanten Funktionen in MIDCAD, die Ableitung der Bestückposition und die statische Kollisionsprüfung, beschrieben.

In Kapitel 8 werden das MIDCAD-System analysiert und evaluiert und ein Ausblick auf eine integrierte Konstruktionsumgebung gegeben. Die Arbeit schließt mit einer Zusammenfassung.

## 2 Stand der Technik in der Konstruktion von MID-Produkten

### 2.1 Konstruktionsverfahren für mechatronische Produkte

#### 2.1.1 Generelles Vorgehen beim Konstruieren nach VDI 2221

Für das Entwickeln und Konstruieren technischer Systeme und Produkte sind zahlreiche Vorgehensmodelle entwickelt worden, die in den VDI-Richtlinien zusammengefasst sind. Ein generelles anerkanntes Vorgehensmodell der Produktentwicklung ist der Ablaufplan nach VDI-Richtlinie 2221. Dort werden der Konstruktionsprozess und die Konstruktionsphasen ausführlich beschrieben. Der Konstruktionsprozess lässt sich in vier Hauptphasen zusammenfassen (siehe Bild 5):

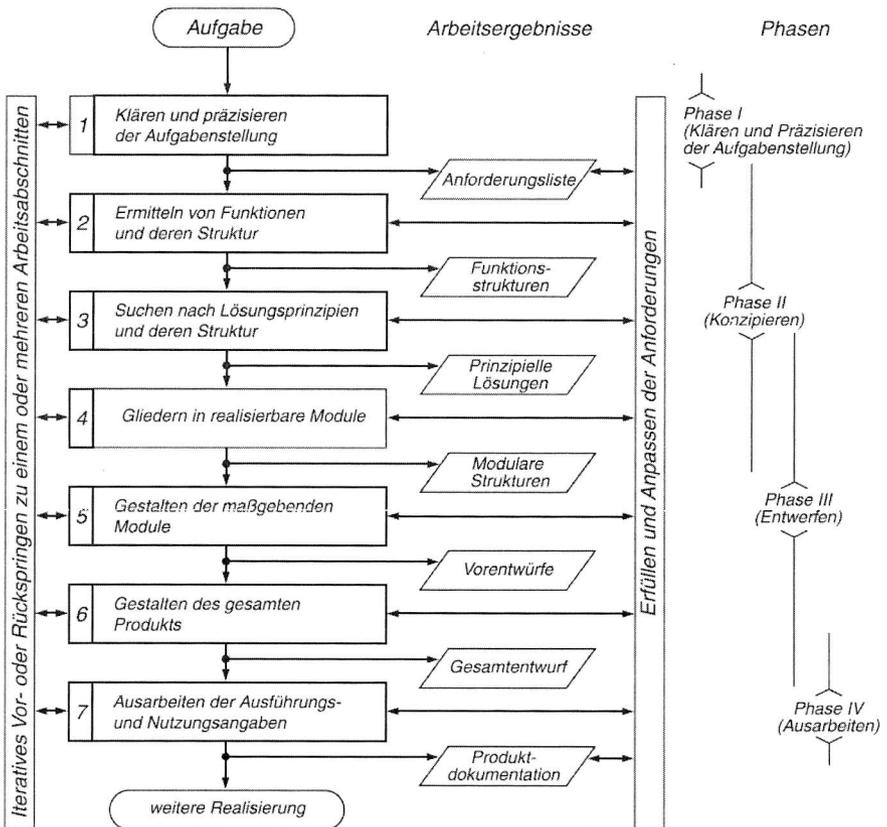


Bild 5: Generelles Vorgehen beim Konstruieren nach VDI 2221 [89]

Die erste Phase beinhaltet die Produktplanung sowie die Beschreibung von Zielerfordernungen. Es müssen Informationen über die an das Produkt gestellten Anforderungen und über die bestehenden Bedingungen gesammelt werden. Darauf baut die Aufgabenstellung auf. Danach erfolgt in der Phase des Konzipierens das Operieren im Problemraum, in dem Funktionsstrukturen aufgestellt und geeignete Wirkprinzipien gesucht werden. In dieser Phase stehen die Arbeitsabschnitte 2 bis 4 und hier wird die Lösung prinzipiell festgelegt. Die Phase des Entwerfens dient als Gestaltungsphase der Dimensionierung der ausgewählten, prinzipiellen Lösung und umfasst die Arbeitsabschnitte 4 bis 6. Die Gestalt und Struktur des Produkts sollen durch die exakte geometrische Beschreibung festgelegt werden. Das Gestalten beinhaltet auch die Auswahl geeigneter, bereits existierender Bauelemente und deren Anpassung an die jeweilige Einsatzbedingung oder die Konstruktion noch nicht vorhandener Bauelemente. Abgeschlossen wird der Konstruktionsprozess mit der Phase des Ausarbeitens, in der die herstellungstechnische Festlegung der Lösung erfolgt, die die erforderlichen Informationen für die nachfolgenden fertigungsvorbereitenden Tätigkeiten, für die Fertigung selbst sowie für den späteren Gebrauch und für die Wartung enthält.

Die Arbeitsabschnitte 1 bis 7 werden beim Ablauf des konstruktiven Entwicklungsprozesses nicht sequentiell abgearbeitet, sondern in Form von iterativen, manchmal auch rekursiven Ablauffolgen teilweise mehrfach durchlaufen.

### 2.1.2 Mechatronisches Vorgehen nach VDI 2206

Bei mechatronischen Systemen, wie beispielsweise MIDs, besteht eine wesentliche Besonderheit darin, dass Komponenten unterschiedlicher Disziplinen miteinander verknüpft werden müssen. Von großer Bedeutung für die Funktionalität des Gesamtsystems ist insbesondere die Wechselwirkung von mechanischen und elektronischen Komponenten. Während bei einem konventionellen System sowohl die Konstruktion als auch die Realisierung der mechanischen und elektronischen Komponenten unabhängig voneinander vorgenommen werden, zeichnet sich ein mechatronisches System dadurch aus, dass mit der Konzeptphase beginnend, ein räumlich und funktionell integriertes Gesamtsystem angestrebt wird (siehe Bild 6). Dem integralen Charakter entsprechend, sollte daher auch zur Konstruktion eines solchen mechatronischen Produkts ein integriertes Konstruktionssystem angewandt werden [37], [38], [39].

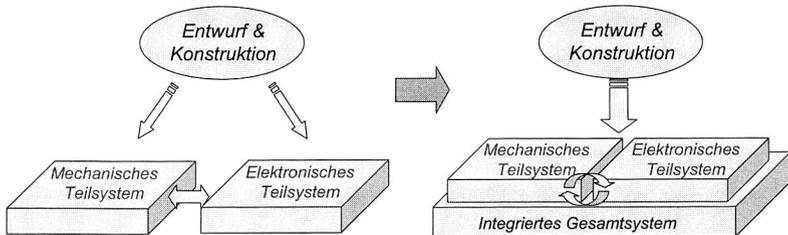


Bild 6: Vergleich von konventionellem und mechatronischem Vorgehen

Aus VDI 2206 stammt das aus der Softwareentwicklung übernommene V-Modell, das die logische Abfolge wesentlicher Teilschritte bei der Entwicklung mechatronischer Systeme beschreibt. Das V-Modell beschreibt das generische Vorgehen beim Entwurf mechatronischer Systeme, das fallweise auszuprägern ist (siehe Bild 7) [87].

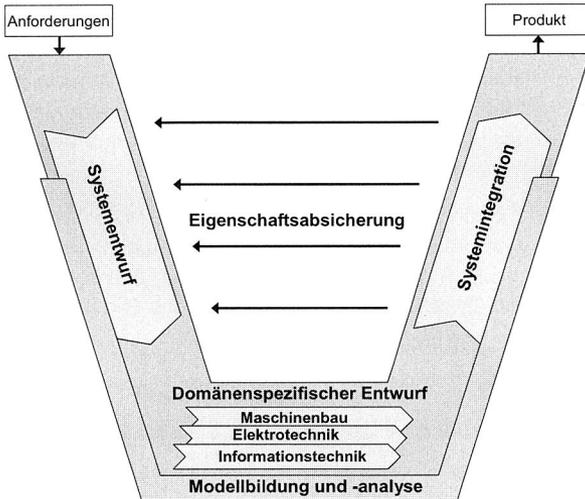


Bild 7: Das V-Modell als Makrozyklus in VDI 2206 [87]

Das V-Modell gliedert die Entwicklungsschritte in Anforderungen, Systementwurf, domänenspezifischer Entwurf, Systemintegration, Eigenschaftensicherung und Produkt. Die Anforderungen bilden den Ausgangspunkt, der einen konkreten Entwicklungsauftrag beschreibt. In der Phase „Systementwurf“ wird das domänenübergreifende Lösungskonzept festgelegt, das die wesentlichen physikalischen und logischen Wirkungsweisen des zukünftigen Produkts darstellt. Im domänenspezifischen Entwurf wird das gemeinsam entwickelte Lösungskonzept in den beteiligten Domänen weiter konkretisiert. Bei der Phase der Systemintegration werden die Ergebnisse aus den einzelnen Domänen zu einem Gesamtsystem zusammengeführt, um insbesondere deren Zusammenwirken untersuchen zu können. Danach erfolgt die Eigenschaftensicherung. Anhand des domänenübergreifenden Lösungskonzepts und der Anforderung wird überprüft, ob die tatsächlichen mit den gewünschten Systemeigenschaften übereinstimmen.

Ein besonderer Teil des V-Modells ist die Modellbildung und -analyse, die die oben beschriebenen Phasen unterstützt und begleitet, um die Systemeigenschaften mit Hilfe von Modellen und CAx-Werkzeugen abbilden und untersuchen zu können. Das Produkt ist das Ergebnis eines durchlaufenen mechatronischen Vorgehens. Dabei versteht man unter dem Produkt nicht ausschließlich das fertige, real existierende Erzeugnis, sondern die zunehmende Konkretisierung des zukünftigen Produkts. Die Entwicklung eines komplexen mechatronischen Produkts erfordert in der Regel mehrere Makrozyklen.

## 2.2 Rechnerunterstützung in der mechatronischen Produktentwicklung

### 2.2.1 MCAD/ECAD-Kopplung über Austauschdateien

Die Beherrschung der Komplexität und der extrem kurzen Innovationszyklen elektronischer Produkte zwingt zum Einsatz rechnerunterstützter Entwurfswerkzeuge (EDA, ECAD), die sich damit zum Standardhilfsmittel in der Elektronik entwickelt haben. Steigende Anforderungen an eine flexible Modellierung der Geometrie und deren hohe Varianz führen auch in der mechanischen Konstruktion zu einem breiten Einsatz rechnergestützter Konstruktionswerkzeuge (MCAD).

Dennoch werden mechatronische Produkte, deren Produktaufbau sowohl eine logisch-elektrische als auch eine dreidimensionale, geometrisch-mechanische Konstruktion erfordert, in der Praxis vielfach noch mit Hilfe gekoppelter, elektronischer und mechanischer CAD-Systeme (ECAD bzw. MCAD) entwickelt (siehe Bild 8).

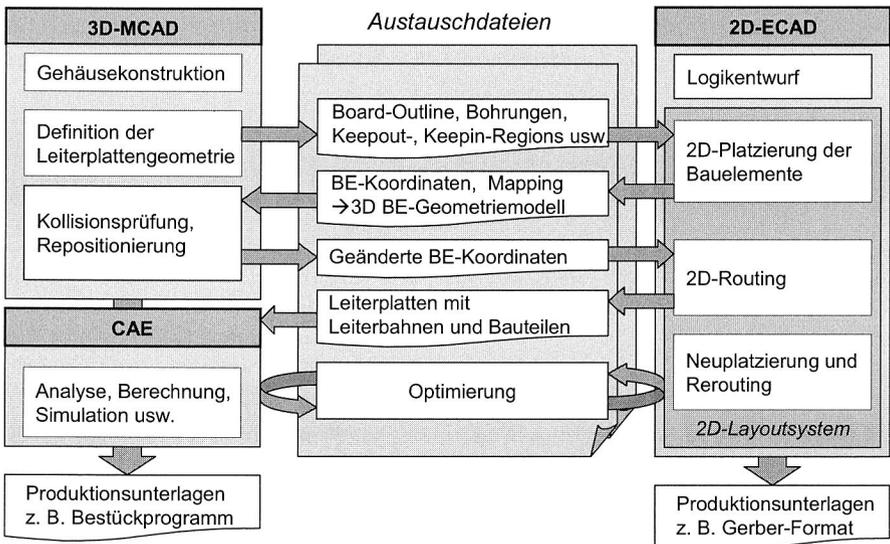


Bild 8: ECAD/MCAD-Kopplung über Austauschdateien [35]

Innerhalb der ECAD-Systeme kommen zweistufige, getrennte Module zur Anwendung. In logischen Entwurfssystemen werden elektrische Elementarfunktionen, die durch grafische Symbole repräsentiert werden, in einer Arbeitsfläche angeordnet. Die elektrische Verschaltung der Funktionsbausteine wird durch Netze realisiert. Der logische Entwurf wird dann in einem vom Logik-Entwurfswerkzeug getrennten, zweidimensionalen Layoutsystem weiterverarbeitet, das die Übertragung des elektrischen Schaltplans in eine zur Herstellung der Leiterplatten geeignete Form ermöglicht. Dabei müssen u. a. Leiter-

plattentechnologie, Bauteilgeometrien, Lage der Bauteile sowie Lage der äußeren Anschlüsse usw. berücksichtigt werden. Das Layoutsystem liest die Komponentenliste und die Netzliste ein. Dann werden die physikalischen Bauelemente auf der Leiterplatte positioniert und die jeweiligen Verbindungen geroutet. Innerhalb der MCAD-Systeme wird das Gehäusedesign bereitgestellt. Ebenfalls können in MCAD Bereiche definiert werden, in denen die Platzierung von elektronischen Bauteilen vermieden werden muss oder innerhalb derer die kritischen Komponenten positioniert werden sollen.

Die für die Elektronikonstruktion relevanten Informationen wie Leiterplattengeometrie, Bereichsdefinitionen etc. werden rechnerintern vom MCAD- an das ECAD-System übergeben. Umgekehrt wird die Lage der Bauteile vom Layoutsystem an das MCAD-System übermittelt. Das am häufigsten verwendete Dateiformat für die Kopplung zwischen ECAD- und MCAD-Systemen ist das IDF-Format (*Intermediate Data Format*). Beim Export oder Import der IDF-Datei zwischen ECAD und MCAD können die Boardgeometrie, die Geometrie der Komponenten (*Footprint* und Höhe), die Platzierung der Komponenten sowie Informationen über *Keepout/Keepin-Regions* übertragen werden. Eine IDF-Datei enthält jedoch keine Information über elektrische Verbindungen. Häufig treten folgende Probleme mit der IDF-Datei auf: Datenverlust durch die Ersetzung komplexer Geometrieobjekte mit Grundgeometrien und unvollständiger Datentransfer, wenn die Daten vom MCAD-System in das ECAD-System zurück importiert werden.

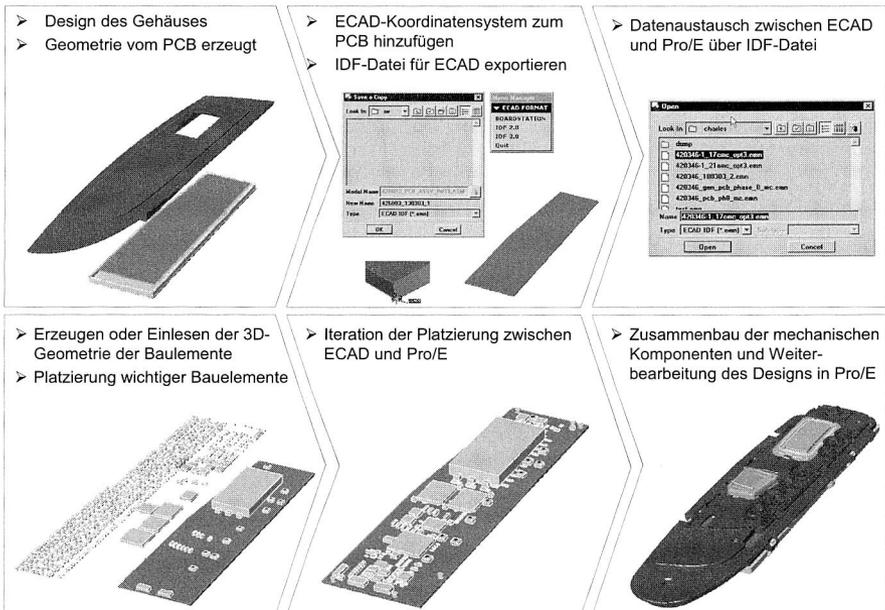


Bild 9: Anwendung von Pro/ECAD in der Konstruktion (Quelle: Fisher & Paykel)

Anhand des MCAD-Modells können bei komplexen Gehäusegeometrien sowohl Kollisionsfreiheit als auch einzuhaltende Mindestabstände überprüft werden. Sollte die Überprüfung ergeben, dass eine Änderung des Layouts notwendig ist, sind diese Änderungen vom Konstrukteur im ECAD-System vorzunehmen. Nach der Änderung des Layouts im ECAD-System müssen diese Änderungen wiederum in das MCAD-System übertragen werden. Die Schleife zwischen ECAD und MCAD muss gemäß der soeben beschriebenen Vorgehensweise solange durchlaufen werden, bis sämtliche relevanten Schwachstellen im Layout beseitigt sind.

CAD-Anbieter bieten zum Datenaustausch zwischen ECAD und MCAD Kopplungen auf unterschiedlichem Niveau an. So gibt es z. B. in Pro/ENGINEER ein zusätzliches Modul Pro/ECAD, das für die Kopplung mit ECAD zuständig ist (siehe Bild 9) [18]. Eine Konstruktionsunterstützung, die ein Rationalisierungspotential in den gekoppelten Systemen ausschöpfen würde, wird dabei jedoch nur in rudimentärer Form angeboten.

Die Besonderheit von MID-Produkten liegt in der untrennbaren Verbindung von elektronischer Funktion und Bauteilgeometrie. Die konventionelle Kopplung von MCAD- und ECAD-Systemen ist nicht in der Lage, diese Verbindung von elektronischer Funktion und Bauteilgeometrie zu beschreiben (siehe Kapitel 3.1). Deswegen können MID-Produkte weder mit Hilfe der ECAD-Systeme noch der MCAD-Systeme ganzheitlich entwickelt und konstruiert werden.

### **2.2.2 Kommerzielle Lösungen zur Konstruktion von MID-Produkten**

Ein bereits zuvor am Lehrstuhl FAPS entwickeltes Konstruktionssystem [56] für MID-Produkte ist vom Prinzip her ein 3D-Layoutsystem zur rechnergestützten Entwicklung räumlicher spritzgegossener Schaltungsträger. In diesem 3D-Layoutsystem sind die beiden Kernfunktionen 3D-Bauelementplatzierung und 3D-Routing verwirklicht. Dabei ist eine Platzierung der Bauelemente auf beliebigen Flächen möglich, gleichzeitig können 3D-Leiterbahnen auf der Oberfläche eines Schaltungsträgers interaktiv, auf manuelle Weise verlegt werden. Das System verfügt jedoch über keine 3D-Autorouting-Funktionen.

Das für das Layoutsystem entwickelte Produktmodell besteht aus zwei Partialmodellen: Eines für die elektrische Schaltung und ein weiteres für die dreidimensionale Geometrie des Schaltungsträgers und des Gesamtprodukts. Das elektrische Partialmodell wird durch ein selbst entwickeltes Partialmodell für logische Konnektivität auf Basis von ISO 10303 Part 103 dargestellt. Das geometrische Partialmodell wird vom Modellierer ACIS<sup>1</sup> verwaltet.

---

<sup>1</sup> ACIS ist ein objektorientiertes dreidimensionales geometrisches Modellierungspaket von Spatial Technology Inc. (Spatial). Es wurde als Geometriekern für Anwendungen entworfen und nicht als eigenständiges CAD-Programm. Es wird gegenwärtig als Kernmodellierer in vielen großen CAD-Systemen (z. B. AutoCAD) eingesetzt.

Auf Basis des am Lehrstuhl FAPS entwickelten Systems wurde von der Firma ME-CADTRON GmbH das industriell eingesetzte 3D-ECAD-System „NEXTRA“ entwickelt [55], [70]. Ebenfalls auf den Entwicklungen des Lehrstuhls FAPS basiert das von der Firma ZUKEN GmbH vermarktete „Electro Mechanical Designer“ (EM-Designer [123], siehe Bild 10). Diese beiden Systeme kommen der Idee eines integrierten Konstruktionsystems für MID-Produkte im kommerziellen Sektor gegenwärtig am nächsten. Beide Systeme beinhalten zur Unterstützung des Konstruktionsprozesses beispielsweise eine Bauelementebibliothek sowie einen Katalog von Design- und Manufacturing-Rules (DRC/MRC).

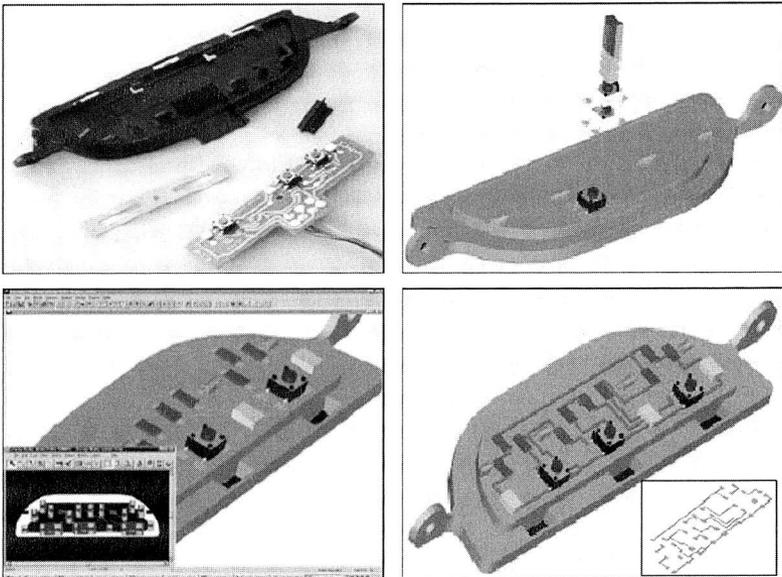


Bild 10: Routing und Platzierung im EM-Designer (Quelle: Zuken-Redac)

Die Konstruktion des räumlichen Schaltungsträgers wird in MCAD-Systemen wie z. B. Pro/ENGINEER vorgenommen und über STEP-Austauschdateien in das System eingebracht. Der Logikentwurf erfolgt in einem ECAD-System, der Schaltplan wird in Form einer Netzliste, z. B. im RINF-Format<sup>2</sup> in NEXTRA bzw. in EM-Designer eingelesen. Die benötigten Bauelemente sind in dreidimensionaler Form in der Bauelementebibliothek vorhanden. In NEXTRA bzw. im EM-Designer laufen schließlich alle Daten zusammen. Dort findet auch das eigentliche 3D-MID-Design statt, bei dem mit Hilfe des integrierten Place & Route-Editors die 3D-Bauteile auf dem Schaltungsträger platziert und die 3D-

<sup>2</sup> Das RINF-Format ist ein lesbares Dateiformat von Zuken-Redac zur Beschreibung der Netzliste des Schaltplans.

Leitungen verlegt werden. Darüber hinaus sind diese Systeme an einen 2D-Autorouter zur automatischen Entflechtung der Schaltung angebunden. Die Ausgabe der Daten an das MCAD-System erfolgt wiederum über die STEP-Schnittstelle. Dort werden die Stempel- und Werkzeuggeometrien erzeugt und die CAM-Daten an die Fräsmaschine zur Herstellung des Prägestempels bzw. Spritzgießwerkzeugs geschickt [99].

## 2.3 Domänenübergreifender Produktdatenaustausch

### 2.3.1 STEP-Grundlage und STEP-Anwendungsprotokoll 210

#### STEP-Grundlage

Ein Ansatz, sämtliche Produktdaten in einem einheitlichen Format abzubilden, soll mit STEP geschaffen werden. STEP steht für *Standard for Exchange of Product Model Data*, und wird seit 1984 von der *International Organisation of Standardization (ISO)* entwickelt. STEP ist ein Synonym für die Normenreihe ISO 10303. Das Ziel der Entwicklung von STEP ist es, eine „eindeutige Repräsentation von computer-interpretierbaren Produktinformationen für den gesamten Lebenszyklus eines Produkts“ zu schaffen [80].

Bild 92 im Anhang zeigt einen Überblick über den Inhalt der STEP-Norm mit den korrespondierenden Nummern der ISO 10303. Die ISO 10303 untergliedert sich entsprechend in eine Anzahl einzelner Standards (*Parts*). *Parts*, die in einem größeren gemeinsamen Zusammenhang stehen, sind in so genannten Serien zusammengefasst. Die STEP-Serien können in sechs Hauptgruppen eingeteilt werden: Beschreibungsmethoden (*Description Methods*), Basismodelle (*Integrated Generic Resources and Integrated Application Resources*), Anwendungsprotokolle (*Application Protocols*), Implementierungsmethoden (*Implementation Methods*) und Methoden für Konformitätstests (*Conformance Testing – Methodology and Framework*).

Die Modellierungssprache EXPRESS<sup>3</sup> bildet die Grundlage sämtlicher genormter Modellbeschreibungen innerhalb der ISO 10303. EXPRESS ist eine objektorientierte Datenbeschreibungssprache und keine funktionale Programmiersprache. Ihr Hauptaugenmerk liegt auf der Definition von Datenstrukturen. Neben EXPRESS ist auch eine grafische Darstellung, EXPRESS-G genannt, definiert. EXPRESS-G verwendet die grafisch verwertbare Teilmenge der EXPRESS-Definitionen und dient zur grafischen Darstellung der definierten Datenstrukturen.

Die Anwendungsprotokolle dienen als Grundlage für die Implementierung von STEP-Prozessoren. Viele Anwendungsprotokolle decken eine sehr breite Funktionalität ab. Für die Produktentwicklungsprozesse im Bereich der Mechatronik sind die folgenden Anwendungsprotokolle von besonders Bedeutung:

---

<sup>3</sup> Ein Beispiel für EXPRESS sowie für EXPRESS-G als graphische Untermenge und deren Zusammenhang mit ISO10303-21 befindet sich im Anhang der Arbeit.

- STEP Anwendungsprotokoll 203 (AP203 – *Configuration Controlled 3D Design of Mechanical Parts and Assemblies* [81]). AP203 gehörte zu den ersten standardisierten Anwendungsprotokollen und ist seit 1994 internationaler Standard. Das AP hat heutzutage die größte Bekanntheit für die mechanische Produktkonstruktion und wird von den meisten MCAD-Systemen unterstützt. Das AP dient dem Austausch und der Beschreibung von Produktdaten konfigurierbarer, zusammengesetzter Produkte mit assoziierter 3D-Geometrie in der Konstruktionsphase.
- STEP Anwendungsprotokoll 212 (AP212 – *Electrotechnical Design and Installation* [83]). AP212 ist seit Anfang 2001 internationaler Standard und beschreibt die Produktdaten von Entwicklungsprozessen in der Elektroindustrie insbesondere beim Cable und Harness Design. Gegenüber den im Bereich der Elektroindustrie eingesetzten Schnittstellenformaten wie IGES, EDIF oder DXF, die zumeist nur den Austausch grafischer Information in Form von 2D-Zeichnungen erlauben, berücksichtigt AP212 die elektrischen und mechanischen Eigenschaften eines elektrischen Produkts [93].
- STEP Anwendungsprotokoll 214 (AP214 – *Core Data for Automotive Mechanical Design Process* [84]). AP214 ist seit Anfang 2001 im Rahmen der ISO 10303 internationaler Standard. AP214 beschreibt die Produkt- und Betriebsmitteldaten von Entwicklungsprozessketten in der Automobil- und Flugzeugindustrie und hat sich mittlerweile als Datenmodell für den gesamten Maschinenbau durchgesetzt. Es wird von immer mehr MCAD-Systemen (z. B. Pro/ENGINEER) unterstützt. Es ist das zentrale AP für die gesamte Automobil- und Flugzeugindustrie.
- STEP Anwendungsprotokoll 210 (AP210 – *Electronic Assembly, Interconnect and Packaging Design* [82]). AP210 ist seit Anfang 2001 internationaler Standard. Es besitzt neben AP214 die größte Komplexität. Im Vergleich zu anderen Anwendungsprotokollen (z. B. AP212, AP214) ist AP210 für die Beschreibung der Produktdaten in Zusammenhang mit der MID-Technologie geeignet. Eine detaillierte Vorstellung von AP210 und seiner Anwendung für den Aufbau des MID-Produktmodells erfolgt im nächsten Kapitel und in Kapitel 6.

Die Anwendung von STEP als Datenbasis für sämtliche produktbezogenen Daten stellt auch einen Forschungsschwerpunkt im Bereich der digitalen Fabrik dar. Im Bereich der Geometrieverarbeitung wird STEP bereits als Datenaustauschformat eingesetzt. Die Fähigkeit, auch Produktstrukturen abbilden zu können, macht STEP zu einem geeigneten Werkzeug, um die Geometriemodelle einer heterogenen CAD-Umgebung zu einem digitalen Gesamtmodell zusammenzuführen.

Ein Problem bei der Verwendung von STEP als Austauschformat ist, dass STEP nicht in der Lage ist, sämtliche in den CAD-Systemen enthaltenen Informationen abzubilden. Insbesondere unterstützen die Datenmodelle in STEP derzeit nicht die Abbildung parametrischer Geometriemodelle. Eine weitere Erschwernis sind der erhebliche Umfang

der Dokumentation sowie die gegenseitige Verknüpfung der einzelnen Teile der Norm untereinander.

Abschließend soll an dieser Stelle darauf hingewiesen werden, dass trotz aller Probleme, die hinsichtlich der Entwicklung und Normierung von STEP aufgetreten sind, dieser Standard zukünftig mit Sicherheit eine zentrale Rolle im Hinblick auf den Austausch, die Speicherung und die Archivierung von Produktdaten einnehmen wird.

### **STEP-Anwendungsprotokoll 210**

Im September 2000 wurde STEP AP210 nach letzten Überarbeitungen und Optimierungen bei der ISO mit dem Ziel der Veröffentlichung eingereicht und 2001 angenommen. Es bildet mittlerweile den Standard für die Beschreibung elektronischer Designs von der Konzeption bis zur Produktion. Das AP wurde entwickelt, um die elektronische Montage, ihre zugehörigen Schaltungsträger/Leiterplatten, Bauelemente und die erforderlichen Daten für die Nutzung der Einzelteile im Design darzustellen. AP210 erfüllt weitgehend die Anforderungen an Design und Analyse sowie an die Produktionsphasen elektro-mechanischer Produkte [51], [104].

AP210 trägt sowohl den Anforderungen des Designtransfers als auch einer exakten Darstellung in folgenden Segmenten Rechnung [34]:

- Elektronische Baugruppen, Schaltungsträger,
- Gehäuse-Typen,
- Bauteile und sonstige Komponenten,
- Module,
- verdrahtete Leiterplattenbaugruppen sowie
- verdrahtete Schaltungen.

Dazu sind in der AP210-Datenstruktur die in Bild 11 gezeigten Informationen in ihren Strukturen in aller Deutlichkeit und Klarheit erfasst. AP210 unterstützt den Austausch und die Integration der Daten des elektromechanischen Designs, einschließlich spezieller Anforderungen und ihrer funktionalen Spezifikationen, *Pin Map* und Konfigurationserkennung mit Datensteuerung mit und ohne anhängende zwei- oder dreidimensionale Modellinformationen [101].

Bei AP210 Applikationsaktivitätsmodell (AAM) werden alle Aktivitäten dargestellt, die erforderlich sind, um ein elektronisches Produkt herzustellen. Dagegen geht es beim Applikationsreferenzmodell (ARM) um die detaillierte Strukturbeschreibung der dafür erforderlichen Daten. Im ARM werden über 700 Applikationsobjekte wie *Interconnect\_module*, *Packaged\_part*, *Bare\_die* etc. und deren Beziehungen untereinander definiert. AP210 unterstützt die Produktbeschreibung kompletter Designdaten einer elektronischen Entwicklung, vom *Die* über das aufnahmebereite Gehäuse bis hin zur kom-

pletten Baugruppe. AP210 selbst definiert keine Standard-Bauteile oder Leiterplatten-Technologien, sondern ein Datenmodell, um Bauteile und Leiterplatten-Technologien zu beschreiben.

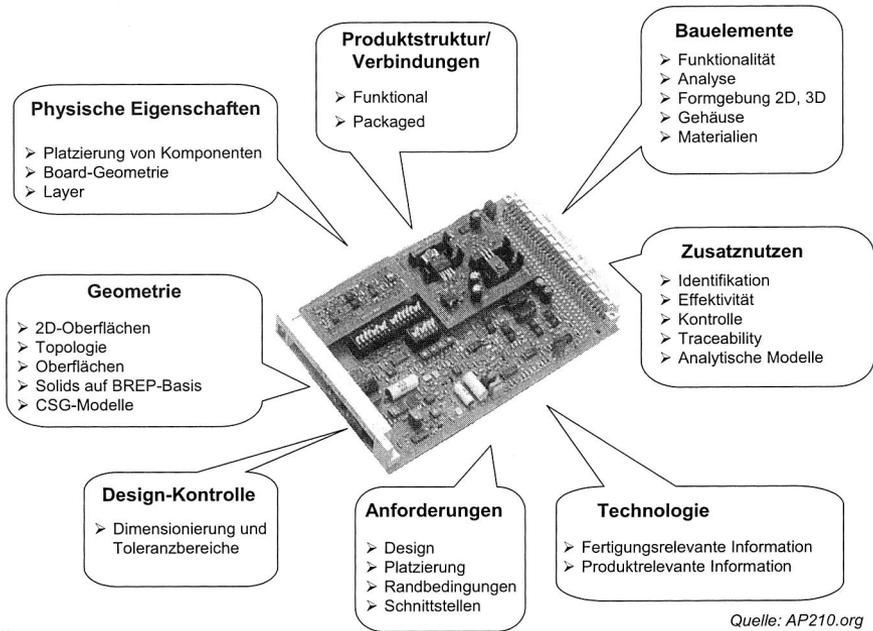


Bild 11: Übersicht über STEP-AP210

### 2.3.2 Produktdatenaustausch zwischen CAx-Programmen

Zur Realisierung des Produktdatenaustausches zwischen Programmen existieren grundsätzlich die folgenden drei Möglichkeiten [24], [41]:

- Systemstetspezifische Konvertierungsroutinen,
- Systemneutrale Schnittstellen (Bild 12 links),
- Verwendung eines integrierten Produktmodells (Bild 12 rechts).

Die erste Möglichkeit findet nur selten Anwendung, daher wird an dieser Stelle nicht näher darauf eingegangen. Für den Datenaustausch zwischen zwei Programmen über systemneutrale Schnittstellen werden je ein Prä- und ein Postprozessor benötigt. Ein Präprozessor wandelt das systemeigene, native Datenformat in ein systemneutrales Schnittstellenformat, z. B. STEP, um. Der Postprozessor des anderen Programms setzt das Schnittstellenformat in sein natives Datenformat um. Mittlerweile sind zahlreiche Entwicklungswerkzeuge, wie beispielsweise CAD-Systeme, bereits mit systemneutralen

Schnittstellen ausgestattet. Unter den vielen unterschiedlichen verfügbaren Datenschnittstellen wird im 2D-MCAD-Bereich der Schnittstelle DXF, im 3D-MCAD-Bereich den Schnittstellen STEP, IGES und STL die größte Bedeutung zuteil. Im ECAD-Bereich sind die Dateiformate Gerber, IDF, GenCAM, ODB++<sup>4</sup> und EDIF am weitesten verbreitet. Von diesen Formaten werden das IDF- sowie das DXF-Format von Unternehmen am häufigsten für den Datentransfer zwischen ECAD und MCAD eingesetzt.

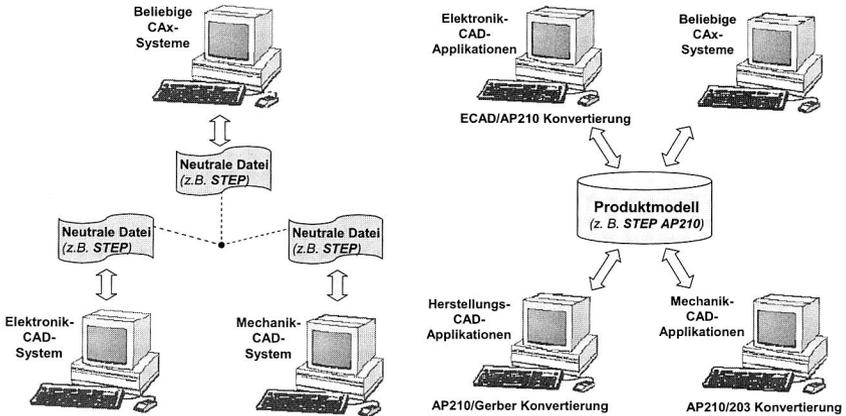


Bild 12: Datenaustausch über eine systemneutrale Schnittstelle (links) und über ein zentrales Produktmodell (rechts)

Mit der Zielsetzung, Daten über Produkte komplett oder auch nur partiell zwischen unterschiedlichen CAx-Systemen austauschen zu können, konzentrieren Unternehmen und internationale Normungsgremien ihre Bemühungen seit einiger Zeit auf die Integration und Weiterentwicklung des STEP-Standardisierungsprojektes. Mit Hilfe des STEP-Formates können nicht nur Geometriedaten, sondern auch eine Vielzahl weiterer Informationen übertragen werden.

AP210 ist in der Lage, alle von Gerber, GenCAM, ODB++, IDF und EDIF definierten Daten zu beschreiben (siehe Bild 13). Eine Untermenge von AP210 ist AP203, das von den meisten MCAD-Systemen unterstützt wird, aber keine elektrische Informationen beschreiben kann. Die vorhandenen Probleme beim Datenaustausch zwischen ECAD und MCAD mittels des IDF-Formats (wie in Kapitel 2.2.1 dargestellt) können durch den Einsatz von AP210 überwunden werden. Der Datenaustausch zwischen CAD und CAM mit verschiedenen Dateiformaten (wie z. B. Gerber, GenCAM, ODB++) kann auch durch den Einsatz von AP210 einheitlich realisiert werden.

<sup>4</sup> Das Gerber-Format, GenCAM-Format und ODB++ -Format sind drei standardisierte Dateistrukturen, die den Datenaustausch zwischen dem elektronischen CAD und CAM ermöglichen.

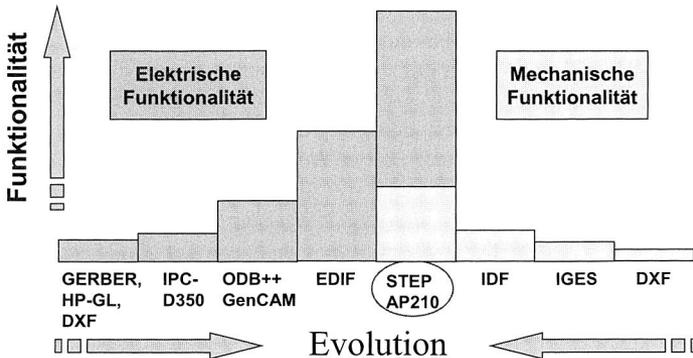


Bild 13: Zusammenführung mechanischer und elektrischer Information in STEP AP210 [104]

### 2.3.3 Produktmodellbeschreibung im Bereich der Mechatronik

Der Einsatz eines Produktmodells kann hierbei Abhilfe schaffen, indem es einen bidirektionalen Datenaustausch unterstützt. Beim Datenaustausch über ein Produktmodell wird dieses als zentrales Organ zur Verwaltung und Speicherung aller auszutauschenden Produktdaten definiert. Durch dieses zentrale Produktmodell können alle Austauschdaten der beteiligten Programme vereint und kontrolliert werden, so dass zu jedem Zeitpunkt nur ein gültiges Datenmodell vorhanden ist. Die Realisierung eines systemneutralen Produktdatenmodells geschieht meist auf der Basis von STEP, da dieser Standard sowohl für den Produktdatenaustausch mittels Dateien als auch für die Integration der Produktdaten in Datenbanken genutzt wird. Daten aus unterschiedlichsten Quellen werden einerseits unabhängig von ihren Anforderungen oder Anwendungen gespeichert, andererseits können Produkt- und Geometriedaten gemeinsam mit Administrativ-, Berechnungs- und Simulationsdaten logisch verknüpft werden [42], [124].

Für den Anwendungsbereich der Werkzeugmaschine wurde im Rahmen des Forschungsprojektes MECHASOFT des Instituts für Werkzeugmaschinen und Betriebswissenschaften der TU München ein integriertes, mechatronisches Produktdatenmodell entwickelt [85]. Das Produktdatenmodell verknüpft die einzelnen fachbereichsspezifischen Partialmodelle und Sichten miteinander und basiert auf einem Metamodell für Werkzeugmaschinen, um den durchgängigen Datenfluss von der Konzeptfindung bis hin zur Implementierung sicher zu stellen [96], [116], [117]. Ein weiteres Projekt zur Entwicklung eines integrierten Datenmodells für mechatronische Produkte wurde von der ProSTEP GmbH unter dem Namen MechaSTEP durchgeführt [75]. Ziel war der Datenaustausch zwischen unterschiedlichen mechatronischen Simulationsprogrammen. Dabei wurden bereits erste Schritte im Hinblick auf eine Normierung ergriffen.

Für die Beschreibung elektromechanischer Produktdaten ist insbesondere AP210 geeignet. Zur Erstellung eines elektromechanischen Produktmodells wird AP210 eine zentrale Rolle spielen, da es in der Lage ist, den Datenaustausch zwischen den beteiligten Anwendungen zu übernehmen. Es eröffnet somit die Chance, eine Vielzahl von CAX-Systemen zu verbinden (siehe Bild 12) [107]. Zahlreiche Projekte haben sich mit der Anwendung von AP210 befasst. So hat Wu [112] ein integriertes Datenmodell für ein Leistungselektronik-System auf Basis von AP210 beschrieben. Thurman [78], [106] hat ein Pilot-Projekt durchgeführt, das die Integration zwischen ECAD- und MCAD-Systemen, basierend auf AP210, gezeigt hat. Die Anwendung von AP210 bei Boeing wurde von Smith [101] dokumentiert. Der Einsatz des integrierten AP210-Datenmodells in der thermischen und mechanischen Analyse bzw. zur Simulation in der elektronischen Produktentwicklung wird beispielsweise in [6], [103] vorgestellt.

### 2.3.4 Integrationsstrategien

Im Rahmen der MID-Produktentwicklung werden derzeit in der Praxis eine Reihe unterschiedlicher, domänenspezifischer Werkzeuge für Logik-Entwurf, 3D-Konstruktion, Layout-Design, Berechnung und Simulation etc. eingesetzt. Mit der steigenden Produktkomplexität sowie den Anforderungen nach kürzeren Entwicklungszyklen und erhöhter Produktqualität nimmt jedoch der Bedarf nach einer tieferen CAX-Integration stetig zu. Zurzeit wird in zahlreichen Forschungsvorhaben versucht, den allgemeinen Entwicklungsprozess durchgängig rechnerunterstützt durchzuführen.

Die Verwaltung der Produktdaten in der Produktentwicklung wird in zunehmendem Maße mit Hilfe von Produktdatenmanagementsystemen (PDM-Systemen) realisiert. PDM-Systeme verwalten und archivieren die während der Produktentwicklung entstehenden Dokumente. Dazu gehören Metadaten, wie Bearbeiter, Konstruktionsversion sowie Projektinformation. Diese Systeme arbeiten dokumentorientiert, die inhaltliche Bearbeitung der Dokumente erfolgt mit den dafür vorgesehenen Programmen, wie CAD- und CAE-Systemen. Daher eignen sich PDM-Systeme gut für die Realisierung einer losen Kopplung [3], [98].

Die Integration von Gestaltung und Berechnung wurde in zahlreichen Projekten erforscht, z. B. in einer Forschungsarbeit über „Integration von Gestaltung und Berechnung mittels CORBA“ (*Common Object Request Broker Architecture*) [42]. Eine Architektur „virtueller Berechnungskompetenzzentren für die Integration von Gestaltung und Berechnung“ wird in [111] vorgestellt. An der Ruhr-Universität Bochum wurde ein Konzept für objektorientierte Integrationsprozessoren (INPRO) entwickelt und implementiert, wobei als Kommunikationssystem ebenfalls CORBA verwendet wurde [110].

Im Projekt ANICA (*ANalysis of access Interfaces of various CAX-Systems*) wurde ein System entwickelt, mit dem verschiedene CAD-Systeme direkt miteinander gekoppelt werden können [4], [5]. Der Bedarf für eine direkte Kopplung rührt aus der Tatsache, dass die bisher in der Praxis angewandte Kopplung über Dateien mit Datenverlust durch nicht umkehrbare Konvertierungen behaftet ist. Als Kommunikationsinfrastruktur

für ANICA wurde ebenfalls CORBA angewandt, womit eine direkte Kopplung von Applikationen möglich ist.

Das Leitprojekt „Innovative Technologien und Systeme für die Virtuelle Produktentstehung“ (iViP) ist ein von Industrie und Forschungseinrichtungen unterstütztes Projekt mit dem Ziel der Realisierung einer vollständig virtuellen Produktentstehung auf Basis durchgängiger, integrierter Prozesse [54]. Im Vordergrund des Projektes steht die Entwicklung innovativer Werkzeuge auf der Basis einer offenen Systemarchitektur. Im Mittelpunkt steht das integrierte Produktmodell „*Digital Master*“, das als verbindlicher Informationsträger alle relevanten Daten eines Produkts enthalten soll. Die dem Projekt zugrunde liegende iViP-Architektur bietet eine offene Integrationsplattform, die die Einbindung bestehender heterogener CAX-Systeme unterstützt.

Aus den oben vorgestellten Projekten und Systemen sowie der Vorstellung in [42], lassen sich für die CAX-Integration im Wesentlichen drei Strategien ableiten:

*a. Integriertes Produktmodell:* Bei dieser Strategie wird versucht, bestehende oder neu geschaffene Produktmodelle so zu erweitern, dass alle in der Produktentwicklung entstehenden Daten in dem Modell enthalten sind. Die konkrete Vorstellung bzw. Anwendung des integrierten Produktmodells erfolgt in Kapitel 6.

*b. Programmgesteuerte Zugriffsmethoden:* Diese Strategie beruht auf der Definition programmgesteuerter Zugriffsmethoden auf die bestehenden Modelle über eine Programmierschnittstelle (API). Das Spektrum der verwendeten Zugriffsmethoden reicht dabei von Kommunikationsstandards auf Systemebene, wie IPC (Interprozesskommunikation) oder TCP/IP, über CAD-systemspezifische statische oder dynamische Bibliotheken, bis hin zur objektorientierten Kapselung und Integration in eine verteilte Arbeitsumgebung.

*c. Modelltransformation:* Eine weitere Strategie ist die Modelltransformation, bei der die Modelle des einen Systems in die Modelle des anderen Systems überführt werden. Eine solche Modelltransformation wird als lose Kopplung bezeichnet. Für diese Kopplung ist ein Dateiaustausch erforderlich. Daher sind Modelltransformationen in der Regel mit Informationsverlust verbunden und somit nicht umkehrbar.

Im Hinblick auf eine Modellkonsistenz und eine prozessübergreifende Nutzung von Daten wäre die Anwendung eines integrierten Produktmodells die beste Lösung. Obwohl eine Modelltransformation wegen des auftretenden Datenverlusts keine optimale Strategie ist, wird sie in der Praxis zur Kopplung der CAX-Systeme am häufigsten eingesetzt. Für eine tiefere CAX-Integration sind daher programmgesteuerte Zugriffsmethoden anzustreben.

### 3 Konzeption eines integrierten CAD-Systems für räumliche spritzgegossene Schaltungsträger

#### 3.1 Übersicht und Anforderungen des MID-Konstruktionsablaufs

##### 3.1.1 Anforderungen an Konstruktionssysteme für räumliche spritzgegossene Schaltungsträger

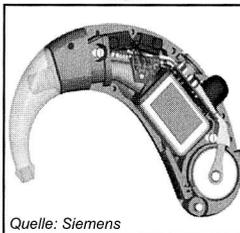
In mechatronischen Produkten stehen mechanische Geometrie und elektronische Funktionen in enger Wechselwirkung. Dies gilt insbesondere für MID-Produkte, in denen diese beiden Funktionen stofflich integriert sind. Dem multidisziplinären Charakter folgend, sollte daher auch zur Konstruktion von mechatronischen Produkten bzw. von MID-Produkten ein integriertes Konstruktionssystem zur Anwendung kommen. Dieses sollte nicht nur über mechanische sowie elektronische Funktionen verfügen, sondern den Anwender auch hinsichtlich design- und fertigungsrelevanter Fragestellungen unterstützen.

###### Anforderungen aus der geometrischen Gestaltungsfreiheit

- Abbildung der Produkte als 3D-Geometriemodelle im Konstruktionssystem
- Durchführung von 3D-Routing und 3D-Platzierung
- Abbildung und Erzeugung der Geometrie von 3D-Leiterbahnen

###### Anforderungen aus der Integration elektronischer und mechanischer Funktionen

- Integration von elektrischem und mechanischem Partialmodell
- Abbildung der Wechselwirkung der elektrischen Funktion und mechanischer Geometrie



MID-Konstruktionssystem

###### Forderungen nach fertigungsgerechten MID-Konstruktionsrichtlinien

- MID-spezifische Gestaltung bzw. Design- und Manufacturing Rule Checks
- Ableitung MID-spezifischer Fertigungsinformationen
- Thermische, kinematische Simulationen usw.

Forderungen durch herkömmliche Kopplung von Elektronik- und Mechanik-CAD nicht erfüllbar

Neues integriertes Konstruktionssystem mit eigenem MID-Produktmodell und dreidimensionaler Geometriepäsentation erforderlich

Bild 14: Anforderungen an ein Konstruktionssystem für räumliche spritzgegossene Schaltungsträger

Die Anforderungen an Konstruktionssysteme für räumliche spritzgegossene Schaltungsträger sind dementsprechend vielfältig. Die hierbei, speziell in Zusammenhang mit

der MID-Technologie zu berücksichtigenden Anforderungen, sind in Bild 14 dargestellt. Sie lassen sich nach Anforderungen aus der geometrischen Gestaltungsfreiheit, aus der Integration elektronischer und mechanischer Funktionen sowie nach design- und fertigungsbezogenen Anforderungen klassifizieren. Sie stellen die Grundlage für die Funktionsmodule des MIDCAD-Systems für räumliche spritzgegossene Schaltungsträger dar.

### 3.1.2 Probleme des konventionellen Konstruktionsprozesses

Bild 15 zeigt die Schwächen und Probleme in der MID-Konstruktion durch die in Kapitel 2.2.1 vorgestellten Kopplungen von ECAD- und MCAD-Systemen.

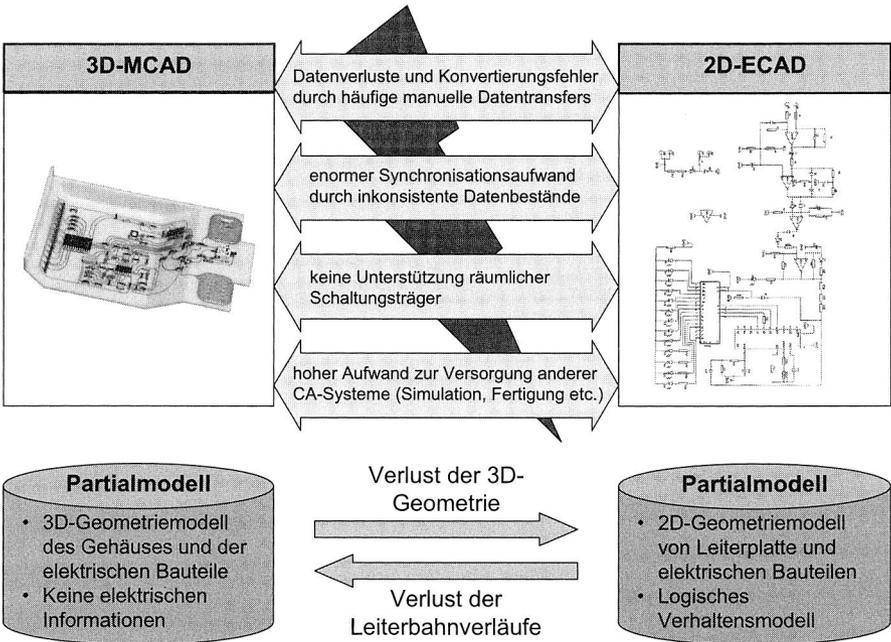


Bild 15: Schwächen und Probleme durch die Kopplung von ECAD und MCAD

Die geometrische Gestaltungsfreiheit von spritzgegossenen Schaltungsträgern verlangt eine dreidimensionale Repräsentation des Produkts im Konstruktionssystem. Die Platzierung elektronischer Bauelemente, das Verlegen der Leiterbahnen oder auch die Analyse mit Hilfe sog. Design- und Manufacturing Rule Checks (DRC/MRC) bei MID-Konstruktion können nur mit Hilfe einer dreidimensionalen Geometrierepräsentation sinnvoll durchgeführt werden. Dies kann von konventionellen 2D-ECAD-Systemen nicht geleistet werden. Wenn ein 3D-MCAD-System in Kombination mit einem 2D-ECAD-System verwendet wird, würde für den Elektronikentwurf eine Abwicklung der bestückten Flächen nötig sein. Dadurch wird die Gestaltungsfreiheit der Oberflächen erheblich eingeschränkt. Nicht abwickelbare Flächen, wie zum Beispiel Kugeloberflächen, die

während des Abwickelns entstehenden Überschneidungen oder beliebige räumliche Verläufe der Leiterbahnen ließen sich mit der herkömmlichen Methode nicht realisieren.

Die Daten der logisch-elektronischen Verbindungen oder der in ECAD gerouteten Leiterbahnen können vom ECAD- nicht in das 3D-MCAD-System übertragen und dargestellt werden. Die Existenz der Leiterbahnstrukturinformation im dreidimensionalen mechanischen Modell ist die Voraussetzung für die darauf folgende Generierung von Fertigungsinformationen für Laserstrukturierung oder andere Herstellungsverfahren. Das Datenmodell im 3D-MCAD-System ist jedoch nicht in der Lage, diese Information zu beschreiben. Die stoffliche Integration von mechanischer und elektronischer Funktion muss auch im Produktmodell abgebildet werden, um so z. B. eine korrekte 3D-Repräsentierung der Leiterbahnen zu gewährleisten. Die Beschreibung der stofflichen Integration der mechanischen und elektronischen Teilfunktionen von MID-Produkten sollte daher nicht in getrennten Partialmodellen abgebildet werden.

Mangels eines einheitlichen Produktmodells findet die Kommunikation zwischen ECAD- und MCAD-Systemen bisher in der Praxis in Form eines dateibasierten Datenaustausches mit Hilfe von systemspezifischen Direktkonvertern oder neutralen Dateiformaten statt. Der Datenaustausch wird dabei meist in mehreren Iterationsschleifen durchgeführt und ist oft mit Datenverlusten, Konvertierungsfehlern sowie Unterbrechungen der Entwicklungsaktivitäten verbunden. Er hemmt somit die Effektivität des Entwicklungsprozesses maßgeblich [5].

### **3.1.3 Schwächen der kommerziellen Lösungen in der Konstruktion von MID-Produkten**

Im kommerziellen Sektor kommen das von der Firma MECADTRON GmbH entwickelte 3D-ECAD-System „NEXTRA“ und das von der Firma ZUKEN entwickelte „EM-Designer“ der Idee eines integrierten Entwicklungssystems für mechatronische Produkte derzeit am nächsten. Obwohl mit „NEXTRA“ und „EM-Designer“ die Effizienz in der MID-Produktentwicklung gesteigert werden konnte, besteht weiterhin ungenutztes Entwicklungspotential:

- Die beiden bestehenden Systeme sind auf die Konstruktion allgemeiner mechatronischer Produkte ausgelegt und berücksichtigen daher bei der Konstruktion von 3D-MID-Produkten nur begrenzt deren spezielle Eigenschaften.
- Zur automatischen Entflechtung der elektrischen Schaltung verwendet der Autorouter in beiden Systemen im Wesentlichen einen 2D-Routingalgorithmus und besitzt somit kein echtes 3D-Autorouting für räumliche Schaltungsträger.
- Auch werden in beiden Systemen keine MID-spezifischen Features wie etwa Durchkontaktierungen, Batteriehalter und Steckverbindungen unterstützt.
- Weiterhin sind keine MID-spezifischen Funktionen, z. B. zur automatischen Generierung von Geometrieinformation für das 2K-Verfahren, enthalten.

- Darüber hinaus konzentrieren sich viele Arbeiten bei der MID-Konstruktion auf die Struktur- bzw. Gestaltoptimierung des 3D-Schaltungsträgers, die starke geometrische Modellierungsfunktionen benötigt und bisher nur bedingt in NEXTRA und EM-Designer durchgeführt werden kann.
- Da das von MCAD importierte Geometriemodell des Schaltungsträgers die historische Modellierungsinformation verliert, ist es auch schwierig, die Veränderung der geometrischen Struktur weiter durchzuführen.

Somit wird deutlich, dass die jetzigen Lösungsansätze nicht in der Lage sind, die speziell in Zusammenhang mit der MID-Technologie zu berücksichtigenden Anforderungen zu erfüllen.

## **3.2 Aufbau des MIDCAD-Systems**

Die oben dargestellten Probleme und Schwächen der jetzigen Konstruktionswerkzeuge bieten hohe Effizienzsteigerungspotentiale. Aus diesem Grund ist ein neues Konstruktionswerkzeug notwendig, das in der Konstruktion und der Herstellung den besonderen Anforderungen des elektronisch-mechanischen Aufbaus von MID-Komponenten Rechnung trägt.

### **3.2.1 Konzept eines integrierten MIDCAD-Systems**

Das zu entwickelnde System muss dabei die in der untrennbaren Verbindung von Bauteilgeometrie und elektronischer Funktion liegende Charakteristik von MID-Produkten berücksichtigen. Die bisher weitgehend isoliert verwendeten Funktionalitäten aus dem ECAD- und MCAD-Bereich sollen daher zu einem integrierten MID-Konstruktionssystem verknüpft werden [120] (siehe Bild 16). Der Aufbau eines entsprechenden Konstruktionswerkzeugs bedarf dazu eines interdisziplinären Ansatzes, der sowohl den mechanischen als auch den elektronischen Anforderungen des Produktentwicklungsprozesses in der MID-Technologie gerecht wird.

In dem zu entwickelnden System sollen jedoch nicht nur die verschiedenen Werkzeuge integriert werden, sondern gleichzeitig zusätzliche, MID-spezifische Funktionen entwickelt werden, die bisher nicht verfügbar waren. Dazu gehören beispielsweise das echte 3D-Routing, die automatische Generierung von Geometrieinformationen für das 2K-Verfahren sowie die automatische Überprüfung von Design- und Manufacturing Rules. Zur Unterstützung der Konstruktion von MID-Produkten soll auch ein zukunftsweisendes domänenübergreifendes integriertes Produktdatenmodell entwickelt werden, das über den gesamten MID-Produktentwicklungsprozess genutzt werden kann. Da viele Arbeiten bei der MID-Konstruktion sich mit der Gestaltung der richtigen geometrischen Struktur des Schaltungsträgers im Dreidimensionalen beschäftigen, um die elektronische Funktionalität stofflich optimal zu integrieren, sind im System starke geometrische Modellierungsfunktionen notwendig.

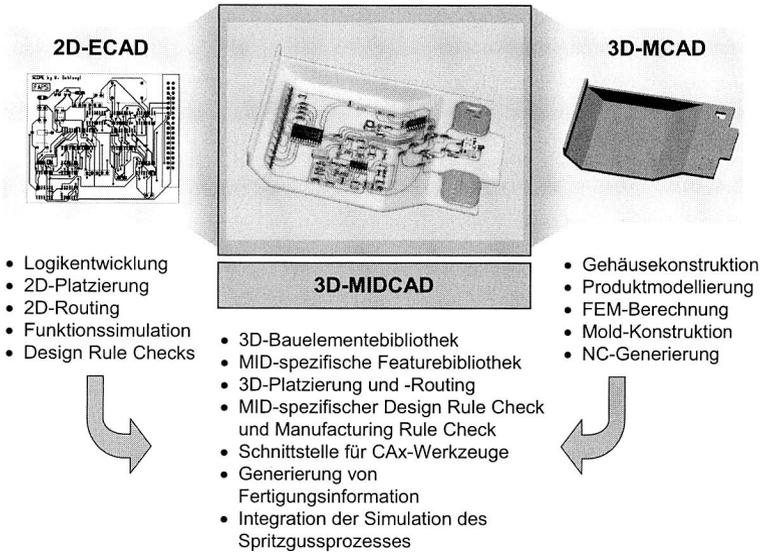


Bild 16: Integration von ECAD- und MCAD-Funktionen in MIDCAD

Deswegen wird MIDCAD unter der Maßgabe entwickelt, eine hohe geometrische Gestaltungsfreiheit bei gleichzeitiger Integration mechanischer und elektronischer Funktionen zu bieten. MIDCAD basiert auf dem kommerziellen 3D-MCAD-System Pro/ENGINEER und wurde mit Hilfe von Pro/TOOLKIT entwickelt [94], [95]. Dies ermöglicht die Nutzung der in Pro/ENGINEER enthaltenen geometrischen Modellierungsfunktionen und erleichtert die Generierung von Fertigungsdaten. Die geometrische MID-Konstruktion wird direkt in Pro/ENGINEER bzw. MIDCAD vorgenommen, weshalb keine zusätzlichen Datenaustauschvorgänge zur geometrischen Modellierung und Optimierung in späteren Konstruktionsphasen nötig sind.

### 3.2.2 3D-MCAD-System als Fundament des MIDCAD-Systems

Die Wahl eines 3D-MCAD-Systems als Fundament des MIDCAD-Systems liegt einerseits in der Notwendigkeit begründet, geometrische Fertigungsdaten für MID-Produkte generieren zu müssen, andererseits in dem erheblichen Aufwand, welcher für die Realisierung eines eigenen geometrischen Modellierers anfällt.

#### Pro/ENGINEER als Grundlage für die 3D-Geometriemodellierung

Das CAD-System Pro/ENGINEER ist ein durchgängig parametrischer featurebasierter 3D-Geometriemodellierer. Die Modellierung erfolgt durch Hinzufügen von intelligenten Konstruktionselementen (KE, Features) zu einem als Basiselement bezeichneten Ausgangskörper. Diese Features sind neutrale, mechanische Objekte, die das Wissen ihrer

Umgebung haben und sich vorausschauend mit der Änderung anpassen. Zum Beispiel existieren Features wie Bohrung, Fase und Rundung, die Ingenieuren geläufig und deswegen einfach zu benutzen sind. Außerdem können spezifische Features für Montage, Fertigung und andere Fachgebiete definiert und benutzt werden. Die Features werden mit Hilfe von Bedingungen und Parametern (einschließlich nichtgeometrischer Attribute) an bestehenden Objekten ausgerichtet.

Mehrere Bauteile können zu einer Baugruppe zusammengesetzt werden. Dabei kann die Positionierung der Komponenten sowohl anhand von Koordinaten für Lage und Ausrichtung erfolgen, als auch durch die Verknüpfung von geometrischen Objekten auf beiden Komponenten über einfache geometrische Relationen.

Die Parameter eines Bauteils können zueinander in Beziehung gesetzt oder aus einer gespeicherten Tabelle bezogen werden. In verschiedenen Modulen, zum Beispiel für die Ableitung von Fertigungszeichnungen oder für die Erzeugung von NC-Datensätzen, können die erzeugten Modelle weiter verarbeitet werden.

Alle Pro/ENGINEER-Lösungen sind vlassoziativ. Das bedeutet, dass eine Änderung, die an irgendeiner Stelle im Entwicklungsprozess durchgeführt wird, durch die gesamte Konstruktionskette weitergereicht werden kann. Pro/ENGINEER verwendet dabei für alle Darstellungen dieselbe Datenbasis, so dass bei Änderungen in einer Darstellung alle anderen Sichten automatisch nachgeführt werden [94].

### **Pro/TOOLKIT als Programmierschnittstelle**

Die offene Architektur von Pro/ENGINEER bietet die Chance, durch unternehmensspezifische Anpassungen und integrierte Zusatzapplikationen den Nutzen der CAD-Technologie deutlich zu erhöhen.

Für den programmgesteuerten Zugriff auf die Geometriemodelle in Pro/ENGINEER steht mit Pro/TOOLKIT eine leistungsfähige Programmierschnittstelle (API) zur Verfügung. Pro/TOOLKIT ermöglicht die Entwicklung von C/C++ Programmen, die auf die internen Datenmodelle und -strukturen zugreifen und diese verarbeiten können. Seit der Version „Wildfire“ steht in Pro/ENGINEER eine weitere Programmierschnittstelle mit J-Link zur Verfügung, die Javaprogramme benutzt. Im Vergleich mit Pro/TOOLKIT sind die Leistungsfähigkeit und Funktionalitäten von J-Link derzeit relativ schwach und begrenzt, weswegen Pro/TOOLKIT als Programmierschnittstelle zur Entwicklung des MIDCAD-Systems ausgewählt wurde.

Mit Pro/TOOLKIT sind alle Operationen möglich, die auch über die Benutzungsoberfläche ausführbar sind. Darüber hinaus lassen sich die Benutzungsoberflächen um weitere Menüpunkte und die internen Datenstrukturen um zusätzliche Merkmale erweitern. Es gibt zahlreiche Forschungsarbeiten über die Erweiterung der CAD-Systeme und Integrationsmethoden, die Pro/TOOLKIT als Programmierschnittstelle benutzen, wie zum Beispiel das System *mfk* als Assistenzsystem für die Beschreibung und Analyse von

Bauteilen und Baugruppen [66], [71], [109], oder das System für die Integration von Gestaltung und Berechnung mittels CORBA [42].

Es gibt immer mehr neu freigegebenen Funktionalitäten in Pro/TOOLKIT, die die Möglichkeit geben, Geometrielemente direkt via Schnittstellenfunktionen zu generieren. Dies sind in der aktuellen Version das Erzeugen von Achsen und komplexeren geometrische Konstrukten, wie beispielsweise *Sweep* und *Shell*. Hierzu muss ein elementspezifischer Baum (*Element-Tree*, siehe Bild 17) mit den jeweils benötigten Daten gefüllt werden.

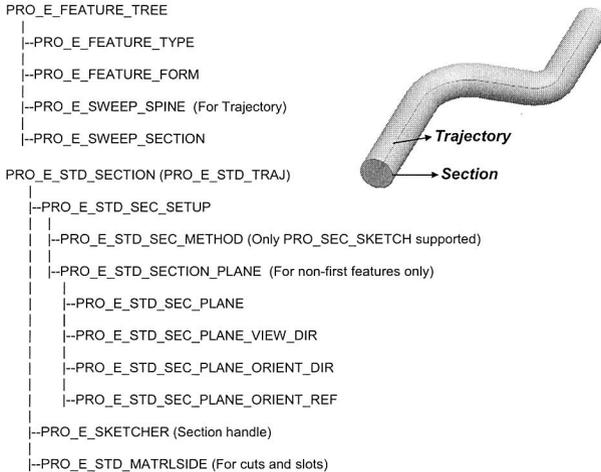


Bild 17: Elementbaum für die Generierung von Sweep-Geometrielementen [95]

Die Implementierung von Pro/TOOLKIT-Applikationen erfolgt entweder als dynamisch-gebundene Bibliothek oder als eigenständiges Programm, das über Interprozesskommunikation auf den Pro/ENGINEER-Prozess zugreift. Die letztere Variante bietet eine größere Flexibilität. Auf diese Weise kann Pro/ENGINEER als Teil einer übergeordneten Anwendung gestartet werden. Eine detaillierte Beschreibung von Pro/TOOLKIT und der Verbindung mit Pro/ENGINEER findet sich im „Pro/TOOLKIT User's Guide“ von PTC [95].

### C++ als Programmiersprache und Visual Studio als Entwicklungsumgebung

Da Pro/TOOLKIT als Programmierschnittstelle zur Entwicklung des MIDCAD-Systems basierend auf Pro/ENGINEER ausgewählt wurde, ist als Programmiersprache C/C++ vorgegeben. Die Programmiersprache C++, die auf Basis der konventionellen Programmiersprache C objektorientiert erweitert wurde, entspricht einer Hybridprogrammiersprache, in denen Objekte und konventionelle Daten nebeneinander existieren. Vorteilhaft bei dieser Sprache ist, dass C++ Programme nicht nur hinsichtlich Prozedur-

ren, Funktionen und Unterprogramme strukturiert sind, sondern auch zusätzlich in Objekte und Klassen unterteilt werden können, wobei Objekte mittels ihrer Operationen (Methoden) miteinander kommunizieren. Eigenschaften der Objekte und Operationen werden durch die Klassenhierarchie bestimmt. Ein weiterer Vorteil der Hybridsprachen ergibt sich daraus, dass elementare Daten und Objekte im Speicher verschieden dargestellt werden.

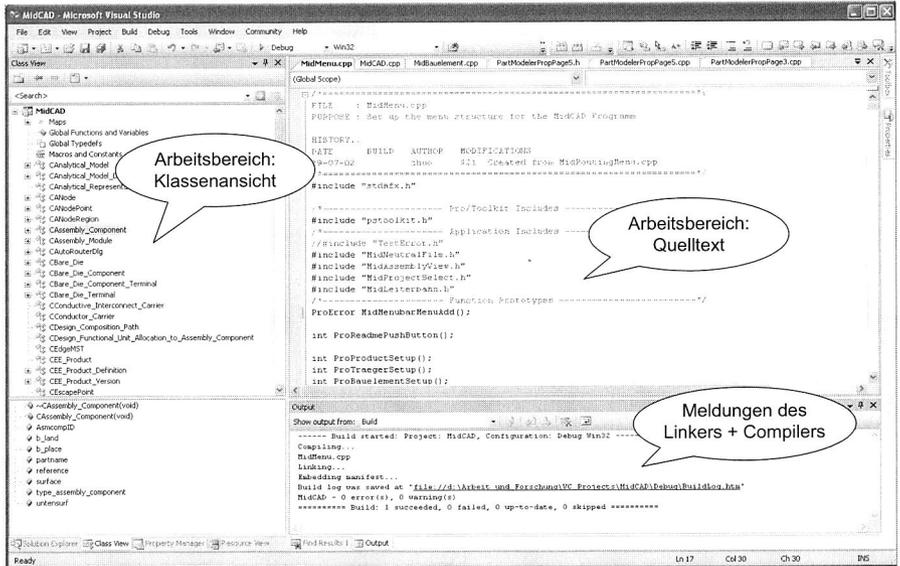


Bild 18: Visual C++ als Entwicklungsumgebung

Das *Visual Studio* von Microsoft ist eine windowsgestützte integrierte Entwicklungsumgebung, die von *Visual C++ (VC++)* und anderen Programmiersprachen (z. B. *Visual Basic*) genutzt wird, und in der der Quelltext, die Compiler- und Linkerangaben sowie der Arbeitsbereich mit sämtlichen Dateien strukturiert dargestellt werden (siehe Bild 18). Damit erfährt die Programmiersprache C++ eine einfachere Handhabung. Ein Projekt als zentraler Begriff ist eine Sammlung von miteinander verknüpften Quelldateien, die zu einem ausführbaren Windows-Programm oder einer DLL (*Dynamic Link Library*) kompiliert und gelinkt werden können. In Bild 18 ist das Projekt „MIDCAD“ mit seinen Teilbereichen beispielhaft dargestellt. Hauptvorteil bei der Verwendung von VC++ im *Visual Studio* ist das *Microsoft Foundation Class (MFC)* Anwendungsgerüst, das über eine Standardstruktur verfügt. Im Vergleich zur Klassenbibliothek ist ein Anwendungsgerüst eine Obermenge einer Klassenbibliothek. Während eine Klassenbibliothek eine bestimmte Menge von Klassen enthält, die darauf ausgelegt sind, in beliebige Anwendungen eingebaut zu werden, liefert ein Anwendungsgerüst die Struktur des eigentlichen Programms.

Die in [95] und [48] dargestellten Methoden der Nutzung der *Visual C++* als Entwicklungsumgebung für die Programmschnittstelle Pro/TOOLKIT können die Potenziale von *Visual C++* kaum ausschöpfen, weil die MFC-Bibliothek von *Visual C++* auf diese Weise nicht verbunden bzw. benutzt wird. Unter Verwendung dieser Methode ist der Entwurf eines Dialogfensters komplex und zeitaufwändig. Deswegen wurde eine andere, wie in [63] dargestellte Methode<sup>5</sup>, zur Anwendung von Pro/TOOLKIT mit der MFC-Bibliothek zur Entwicklung des MIDCAD-Systems benutzt. Mit dieser Methode können nicht nur der Entwurf des Dialogfensters, sondern auch die Verbindung und Verwaltung einer Datenbank leicht realisiert werden.

### 3.2.3 Aufbau des MIDCAD-Systems

Einen Überblick über den Aufbau sowie die wesentlichen Funktionen von MIDCAD enthält Bild 19.

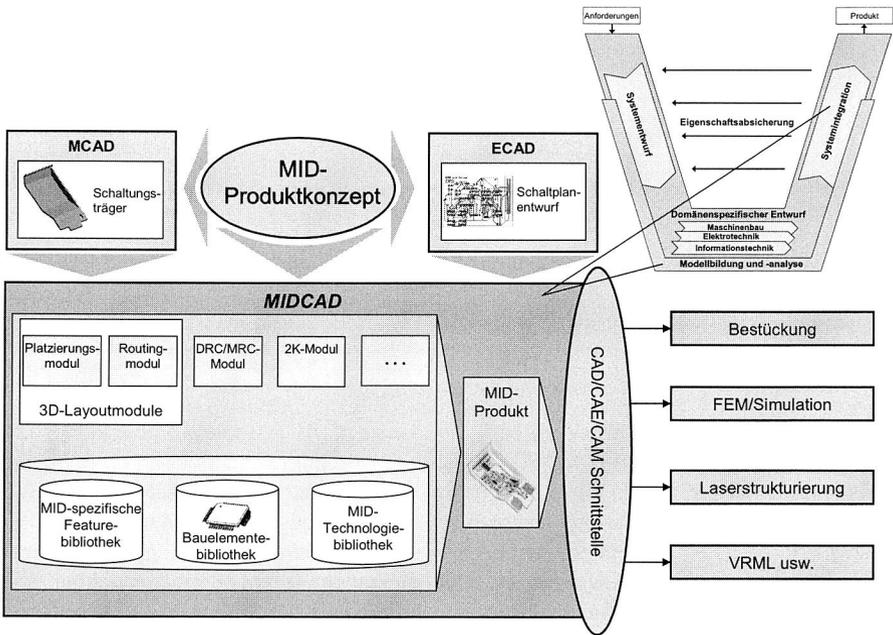


Bild 19: Aufbau des MIDCAD-Systems

Die Funktionalitäten von CAD-Systemen machen deutlich, dass diese nicht alle Arbeitsabschnitte in der Produktentwicklung unterstützen. Die meisten CAX-Systeme werden

<sup>5</sup> Eine kurze Beschreibung der Verbindung zwischen MFC und Pro/TOOLKIT befindet sich im Anhang der Arbeit.

ab dem Zeitpunkt der Produktgestaltung konsequent eingesetzt. Das MIDCAD-System unterstützt derzeit hauptsächlich die in Bild 5 dargestellte Phase des Entwerfens und Ausarbeitens. Gemäß dem V-Modell (siehe Bild 7), welches das generische Vorgehen beim Entwurf mechatronischer Systeme beschreibt, steht das MIDCAD-System derzeit hauptsächlich für die Konstruktion, Modellbildung und -analyse in der Phase „Systemintegration“ zur Verfügung, da es sich beim 3D-MID-Design heute hauptsächlich um ein Integrationsdesign der elektronischen und mechanischen Funktionen handelt.

Die Entscheidung zur Anwendung der MID-Technologie für ein Produkt ist nicht die Aufgabe des MIDCAD-Systems. Der weitere domänenspezifische Entwurf erfolgt meist getrennt in MCAD- und ECAD-Systemen. Die Geometrie des räumlichen Schaltungsträgers wird in Pro/ENGINEER oder anderen MCAD-Systemen entworfen und modelliert. Der elektronische Entwurf (z. B. Schaltplanentwurf) erfolgt im ECAD-System.

Um die Information über die logischen Verbindungen des Schaltplans aus dem ECAD-System einzulesen, enthält MIDCAD eine geeignete Schnittstelle. Über die Importmöglichkeiten hinaus verfügt auch MIDCAD über einige grundlegende Funktionen zur Definition der logischen Verbindungen zwischen Bauteilen. Da MIDCAD auf Pro/ENGINEER basiert, ist kein zusätzlicher Datenaustausch zwischen beiden Systemen notwendig. Für das Einlesen des Geometriemodells des Schaltungsträgers aus anderen MCAD-Systemen ist Pro/ENGINEER zuständig.

Das echte 3D-MID-Design findet in MIDCAD statt. Die beiden 3D-Layoutfunktionen (3D-Routing und 3D-Platzierung) stellen eine Erweiterung der 2D-Layoutfunktionen aus dem ECAD-Bereich auf den dreidimensionalen Raum dar, um die wesentliche Problemstellung des 3D-MID-Designs - Definition der Bauelemente und Leiterbahnverläufe auf bzw. innerhalb eines dreidimensionalen Grundkörpers - zu lösen.

Die 3D-Platzierungsfunktion wird auf Basis der in Pro/ENGINEER existierenden Montagefunktionen für den eigenen Bedarf weiterentwickelt. In MIDCAD wird die 3D-Platzierungsfunktion einerseits im Hinblick auf die Visualisierung und die Operation optimiert und andererseits mit DRG/MRC Funktionen verbunden, um mögliche Kollisionen und die Bestückbarkeit online zu überprüfen.

Ein wichtiger Teil der Entwicklungsarbeit für MIDCAD liegt in der Entwicklung der 3D-Routingalgorithmen. Mit Hilfe der neu entwickelten 3D-Routingalgorithmen werden die Leiterbahnen auf den Oberflächen des dreidimensionalen Schaltungsträgers automatisch verlegt. Durch Unterstützung des interaktiven, automatischen 3D-Routings wird hiermit ein wesentlicher Produktivitätsfortschritt erreicht.

In MIDCAD ist eine dreidimensionale elektronische Bauelementebibliothek für die Konstruktion von 3D-MID-Produkten obligatorisch. Ein wesentlicher Teil der Entwicklungsarbeit für MIDCAD liegt in der Erstellung dieser Bibliothek, in der neben einer dreidimensionalen geometrischen Beschreibung auch elektronische und andere Informationen, z. B. über die elektrische Funktionalität, enthalten sein können.

Bei der Konstruktion von MID-Produkten wird auch die Feature-Technologie in MIDCAD angewandt. Eine Featurekategorie dient der Beschreibung der räumlichen spritzgegossenen Schaltungsträger. Neben den allgemeinen geometrisch-mechanischen Features werden weiterhin MID-spezifische Features, die sich durch zusätzlich enthaltene elektronische Funktionen auszeichnen, in MIDCAD definiert. Dazu gehören beispielsweise Schnapp- und Steckverbindungen, Batteriehalter, Durchkontaktierungen usw. Durch die Bereitstellung der 3D-Bauelementebibliothek und der MID-Featurebibliothek kann die effiziente Konstruktion eines MID-Produkts erheblich gesteigert werden.

Neben der Bereitstellung der 3D-Layoutfunktionen sowie der 3D-Bauelement- und MID-Featurebibliothek enthält MIDCAD weiterhin MID-spezifische Funktionen, die die spätere Fertigung des MID-Produkts im Zweikomponentenspritzgussverfahren unterstützen. Das 2K-Verfahren eröffnet die größte geometrische Freiheit aller MID-Herstellungsverfahren und stellt deshalb neben der Heißprägetechnik das am häufigsten angewandte Verfahren zur Herstellung von MID-Produkten dar. In MIDCAD ist ein 2K-Modul enthalten, welches nach dem Routen die Geometriemodelle der 3D-Leiterbahnen sowie der ersten und zweiten Komponente, wie sie für das 2K-Verfahren benötigt werden, automatisch erzeugt.

Für die Technologie der räumlichen spritzgegossenen Schaltungsträger existieren spezifische Design- bzw. Manufacturing Rules. Für diese MID-Regeln sind angepasste Überprüfungsalgorithmen zu implementieren. Daher ist in MIDCAD ein Kontrollmodul enthalten, das Design- bzw. Manufacturing Rule Check (DRC/MRC) durchführen kann. Das Kontrollmodul bietet z. B. die Möglichkeit, Mindestleiterbahnabstände im Dreidimensionalen sicherzustellen. Dadurch kann ein Konstrukteur Probleme während des Entwicklungszeitraums frühzeitig entdecken und so das Design verbessern.

Zur Beschreibung der MID-Produkte wird ein integriertes Produktdatenmodell entwickelt. Von großer Bedeutung für ein integriertes MIDCAD-System sind die MID-Konstruktion und Datenbehandlung mittels eines einheitlichen, integrierten, domänenübergreifenden Produktmodells. Auf Basis des integrierten Produktmodells werden auch Schnittstellen zwischen MIDCAD und CAE- bzw. CAM-Anwendungen entwickelt, um eine konstruktionsintegrierte und gleichzeitig fertigungsorientierte Optimierung eines MID-Produkts durchführen zu können. Die Kopplungen zwischen MIDCAD und den in der MID-Entwicklung beteiligten Anwendungen können gemäß zweier Integrationsstrategien erfolgen, z. B. über die direkte Verknüpfung mit dem CAE-System MOLDFLOW durch die Entwicklung einer Programmierschnittstelle auf Basis der OLE/COM-Technologie (OLE - *Object Linking and Embedding*, COM - *Component Object Model*), oder über die lose Kopplung mit dem Bestücksystem und dem ECAD-System durch Austauschdateien.

## 4 Entwicklung automatischer 3D-Routing-Funktionen

### 4.1 Übersicht der Verdrahtung in der Elektronik (EDA)

#### 4.1.1 Allgemeines Routingproblem

Nach der Platzierung<sup>6</sup> der elektrischen Bauteile und weiteren Schaltungselementen erfolgt das Routing der in der Netzliste festgelegten Verbindungen. Das allgemeine Problem des automatisierten Routings lässt sich wie folgt darstellen [65], [69]:

*Gegeben seien eine oder mehrere Layoutflächen mit fest platzierten Bauteilen sowie festen Positionen der Anschlüsse (Pins). Eine Netzliste legt fest, welche Anschlüsse miteinander zu verbinden sind. Zu Berechnen ist nun der exakte Verlauf oder die Geometrie aller Verbindungsleitungen. Dabei gilt es eine Vielzahl von Randbedingungen (z. B. Mindestabstände und Mindestbreiten) einzuhalten und die Optimierung von Ziel-funktionen anzustreben, wie z. B. eine minimale Verbindungslänge oder eine minimale Anzahl der Durchkontaktierungen für Mehrebenen-Routing.*

Beim automatischen Routing sind verschiedene Randbedingungen in Form von geometrischen Regeln abgebildet. Im Wesentlichen sind dies die Entwurfsregeln (*design rules*), die technologieabhängig bei jedem Entwurf vorgegeben sind. Im Allgemeinen gilt, dass bei der Netzverbindung eine kurze Netzlänge anzustreben ist, weil damit die benötigte Verdrahtungsfläche minimiert und die Signalqualität verbessert werden.

Der Erfolg und die Qualität des Routings sind sehr stark abhängig von der Platzierung der Bauelemente. Es ist daher wünschenswert, die Platzierung und Verdrahtung simultan durchzuführen. Leider ist das gemeinsame Layoutproblem so komplex, dass es sich einer optimalen algorithmischen Lösung entzieht. Bereits in seiner allgemeinen Formulierung ist das Routingproblem nachgewiesenermaßen NP-vollständig (*Non-deterministic Polynomial time*) [52].

#### 4.1.2 Globale Verdrahtung und detaillierte Verdrahtung

Aufgrund der Komplexität des Routingproblems wird der automatische Routingprozess in der Regel in zwei Teilprobleme, das globale und das detaillierte Routing, unterteilt. Man unterscheidet die Globalverdrahtung, in der die Verdrahtungsbereiche für die Leitungen belegt werden, und die detaillierte Verdrahtung, in der die geometrische Form der Leitung festgelegt wird.

Beim globalen Routing werden ungefähre Verbindungswege auf einer Layoutoberfläche festgelegt. Zunächst wird die Gesamtfläche in Teilflächen (Regionen) aufgeteilt. An-

---

<sup>6</sup> Die Realisierung der 3D-Platzierung wird gemäß der Gliederung zusammen mit anderen MID-spezifischen Funktionen in Kapitel 5 vorgestellt.

schließlich wird jedem zu verdrahtenden Netz unter Berücksichtigung der jeweiligen Verdrahtungskapazitäten dieser Regionen eine dieser Teilflächen zugewiesen, durch die es später verlaufen wird. Es entsteht eine grobe Netztopologie, jedoch noch keine exakte Netzgeometrie.

Während der detaillierten Verdrahtung werden die Verbindungsleitungen, die zuvor bestimmten Flächen zugewiesen wurden, endgültig in einer bestimmten geometrischen Form auf den Flächen verlegt und mit den Anschlüssen der Bauelemente verbunden. Dazu sind die Anschlüsse der Verbindung aus der Platzierung sowie die Flächen, durch die die Verbindung verlaufen soll, aus der Globalverdrahtung bekannt. Bei dem detaillierten Routing werden verschiedene Routingmethoden eingesetzt. Je nachdem, ob das Routing ein Gitter voraussetzt oder nicht, gibt es zwei Grundtypen von Routingalgorithmen für traditionelle 2D-Routingprobleme im PCB-Entwurf [91].

### *Rasterverdrahtung (Grid-Routing)*

Die gesamte Layoutfläche wird zunächst auf ein Raster bzw. Gitter abgebildet, wobei ein Rasterelement (oder: Rasterpunkt) in der Regel genau die Breite einer Leitung plus den Minimalabstand zwischen zwei Leitungen hat. Dann suchen rasterbasierte Algorithmen einen Weg zwischen Start- und Zielpunkt eines Netzes unter Verwendung des gerasterten Verdrahtungsmodells. Der Lee-Algorithmus [60] ist der wohl bekannteste Vertreter der rasterbasierten Routingalgorithmen, wobei ein Markierungsverfahren zum Auffinden des Verbindungswegs verwendet wird. Da der Lee-Algorithmus eine symmetrische Breitensuche verwendet, ist die Anzahl der besuchten Rasterpunkte deutlich höher als die optimierten rasterbasierten Algorithmen, wie z. B. Soukups Fast-Maze-Router [102] oder Hadlocks Minimum-Detour-Algorithmus [43].

### *Linienvdrahtung (Line-Probe- oder Line-Search-Routing)*

Diese Verfahren zeichnen sich dadurch aus, dass sie ohne Raster arbeiten und so einen deutlich geringeren Speicherbedarf haben. Im Vergleich zu der Rasterverdrahtung, bei der eine Teillösung durch das Hinzufügen eines Rasterpunkts expandiert wird, werden die Teillösungen bei Liniensuchverfahren durch das Hinzufügen von Liniensegmenten erweitert. Jedes Liniensegment, dem in den meisten Fällen viele Rasterpunkte entsprechen, wird als Versuchslinie bezeichnet. Der Hightower-Algorithmus [46] ist einer der bekanntesten Vertreter dieser Routingmethode. Eine kurze Beschreibung des Hightower-Algorithmus befindet sich in Kapitel 4.3.2.

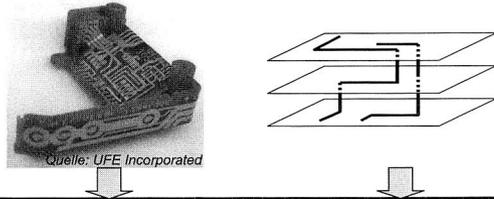
## **4.2 3D-Autorouting für MID-Produkte**

### **4.2.1 Unterschiede zwischen Multilayer- und 3D-MID-Routing**

Die Unterschiede zwischen einem herkömmlichen 2D-Routingproblem und dem 3D-Routing auf MIDs sind in Tabelle 1 dargestellt. Die dreidimensionale Verdrahtung der elektrischen Verbindungen für räumliche Schaltungsträger gestaltet sich durch zusätzli-

che Freiheitsgrade deutlich schwieriger als die Verdrahtung auf zweidimensionalen Multilayer-PCBs.

Tabelle 1: Vergleich zwischen 3D-MID- und Multilayer-Routing [122]



	3D-MID-Routing	Multilayer-Routing
<b>Geometrie</b>	Komplex; zusammengesetzt aus unterschiedlichen Oberflächen- und Linientypen	Einfach; in der Regel lediglich rechtwinklige Oberflächen
<b>Anordnung der Oberflächen</b>	Oberflächen können in beliebigem Winkel aufeinander treffen	Parallel zueinander
<b>Leiterbahnübergang zwischen den Oberflächen bzw. Ebenen</b>	An den gemeinsamen Kanten zweier Oberflächen oder über Durchkontaktierung	Prinzipiell an jedem Ort möglich

#### 4.2.2 Anforderungen an und Lösungsmöglichkeiten für das 3D-Autorouting

Während das automatische Routing auf herkömmlichen, zweidimensionalen Leiterplatten bereits seit den 70er Jahren einen hohen Entwicklungsstand erreicht hat, stellt das automatische Routing im dreidimensionalen Raum eine Problemstellung dar, für die es noch keine zufrieden stellende Lösung gibt. Traditionelle 2D-Multilayer-Routingverfahren können nur sehr begrenzt durch eine Abwicklung der bestückten Flächen direkt auf 3D-Schaltungsträger übertragen werden. Da durch die Abwicklung die meisten geometrischen Informationen über die Konnektivität zwischen den Flächen verloren gehen, sind der Erfolg und die Qualität des nachfolgenden 2D-Routings stark abhängig von den Abwicklungsergebnissen. Außerdem lassen sich nicht abwickelbare Flächen, wie zum Beispiel Kugeloberflächen, die während des Abwickelns entstehende Überschneidungen oder beliebige räumliche Verläufe der Leiterbahnen mit der 2D-Multilayer-Routingmethode nicht realisieren.

Seit Beginn der 90er Jahre ist die Anzahl der Verdrahtungsebenen, insbesondere bei Multichip-Modulen (MCMs), stetig gestiegen. Damit wurden einige neue sog. „dreidimensionale“ Routingalgorithmen entwickelt, die die Mehrlagenstruktur des Schaltungsträgers berücksichtigen. Die aktuellen Routingverfahren für derartige Mehrlagenstrukturen arbeiten im Wesentlichen jedoch nur 2.5-dimensional, d. h. der Verbindungsweg

wird in mehreren parallelen Ebenen gesucht, wobei Ebenenwechsel möglich sind. Lediglich die Anzahl der Verdrahtungsebenen ist größer als früher. Deswegen ist das automatische Routing in einem realen 3D-Raum für MID-Konstruktion immer eine neue Aufgabe, die folgende Probleme lösen muss:

- Effektive Beschreibung der elektrischen Verbindungswege im 3D-Raum und genaue Darstellung der 3D geometrischen Konnektivitäten der Flächen des Schaltungsträgers.
- Herausfinden eines Verbindungswegs mit minimaler Verbindungslänge für eine Leitung unter bestimmten Randbedingungen. Die Verbindungsleitung muss auch auf den Oberflächen, die komplexe geometrische Formen haben und nicht parallel zueinander angeordnet sind, verlegt werden können.
- Verringerung der hohen Komplexität und des großen Berechnungsaufwandes, der durch das Problem im 3D-Raum verursacht wird.

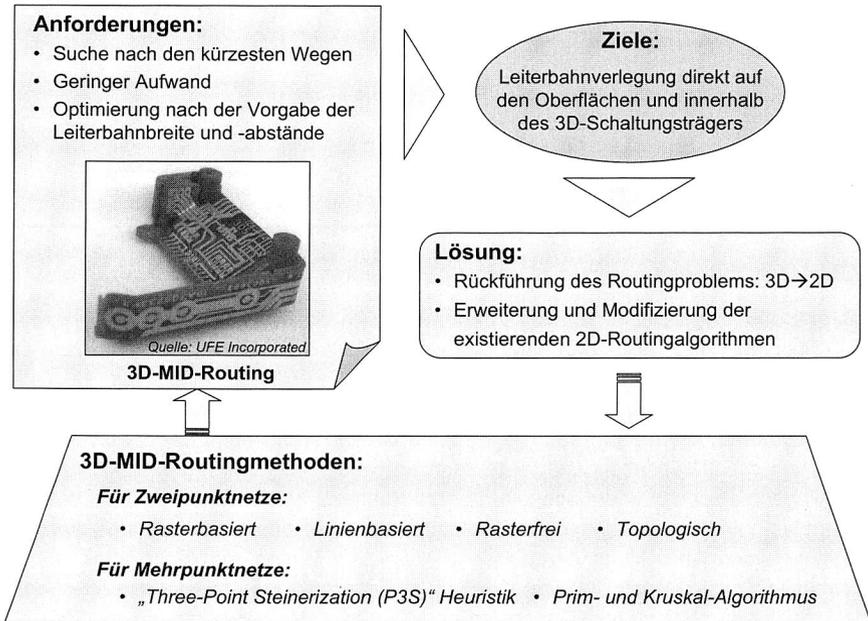


Bild 20: 3D-Autorouting – Anforderungen und Lösungsmöglichkeiten

Es sind daher neue Routingmethoden notwendig, die den spezifischen, vorwiegend durch die dreidimensionale Struktur des Schaltungsträgers bedingten Anforderungen entsprechen. Eine Möglichkeit, Routingprobleme im 3D-Raum zu lösen, besteht in der Zurückführung des dreidimensionalen Problems auf ein zweidimensionales und der anschließenden Anwendung bereits existierender oder modifizierter bzw. erweiterter 2D-

Routingverfahren. Die Konvertierung eines Routingproblems von 3D in 2D und die Erweiterung der bestehenden 2D-Verfahren stellen daher den Schlüssel zur Lösung des Routingproblems in Verbindung mit 3D-MID-Produkten dar (siehe Bild 20).

In den folgenden Abschnitten werden zunächst zwei grundlegende, für MIDCAD entwickelte 3D-Routingmethoden [29], [30], [32] für Zweipunktnetze dargestellt. Eine Methode ist rasterbasiert und erweitert den Hadlock-Algorithmus, die andere ist linienbasiert und beruht auf einer Erweiterung des Hightower-Algorithmus in Kombination mit dem A\*-Suchalgorithmus [17]. Anschließend wird eine weiter optimierte 3D-Routingmethode vorgestellt, die die Vorteile von beiden grundlegenden Methoden vereint und auf Basis eines rasterfreien Algorithmus modifiziert. Schließlich werden Routingmethoden für Mehrpunktnetze auf dem Schaltungsträger dargestellt.

Die entwickelten 3D-Routingmethoden zeichnen sich durch die folgenden zwei Eigenschaften aus:

- Neue 3D-Routingmethoden basieren auf der exakten Information über die geometrischen 3D-Konnektivitäten der Verdrahtungsflächen des Schaltungsträgers, anstatt die Flächen abzuwickeln.
- Diese 3D-Routingmethoden können die exakte Geometrie der Verbindungen auf dem Schaltungsträger finden und explizit beschreiben, während die prozessspezifischen Randbedingungen (z. B. minimaler Isolierabstand) im Dreidimensionalen eingehalten werden.

### **4.3 Entwicklung neuer 3D-Routingalgorithmen für Zweipunktnetze auf räumlichen Schaltungsträgern**

#### **4.3.1 Rasterbasierter 3D-Routingalgorithmus**

Das rasterbasierte 3D-Routing beruht auf einem dreidimensionalen Gitter, in welchem die Wellenausbreitung benachbarter Rasterpunkte nicht begrenzt auf einer Fläche durchgeführt wird, sondern auch auf anderen Oberflächen des Schaltungsträgers (manchmal auch in z-Richtung, d. h. durch Durchkontaktierung zwischen zwei zulässigen Oberflächen) weitergeführt wird. Der dabei gefundene Verdrahtungsweg ist der kürzeste Weg zwischen zwei Anschlusspunkten auf dem Schaltungsträger.

Das rasterbasierte Routingverfahren für 3D-MID-Produkte lässt sich in zwei Teilschritte unterteilen:

- a. Rasterung bzw. Vernetzung der Oberfläche des Schaltungsträgers und
- b. Routen auf der vernetzten Oberfläche analog zu 2D-Routingverfahren.

Die Rastergenerierung verläuft somit getrennt vom eigentlichen Routingprozess. Das Raster wird in Abhängigkeit von Leiterbahnbreite und Isolierabstand in Anlehnung an die Methode der finiten Elemente generiert. Anschließend wird eine Erweiterung von

Hadlocks Minimum-Detour-Algorithmus zur Lösung des eigentlichen Routingproblems auf dem Schaltungsträger verwendet.

### Erweiterung von Hadlocks Minimum-Detour-Algorithmus

Der Hadlock-Algorithmus gehört zu den optimierten rasterbasierten Routingalgorithmen. Zur Markierung der Nachbarpunkte wird die Detour-Nummer zur Indizierung verwendet (siehe Bild 21), anstatt die Wellenfortschrittswerte wie bei der Ausbreitung einer Welle anzuwenden. Als Nachbarpunkte werden ausschließlich horizontal und vertikal angrenzende Rasterelemente betrachtet, um eine orthogonale Verdrahtung zu gewährleisten. Der bereits besuchte Rasterpunkt mit der niedrigsten Detour-Nummer hat eine höhere Priorität und wird vorzugsweise ausgewählt und expandiert. In Bild 21 werden einige relevante Begriffe (z. B. Detour-Nummer) dargestellt und eine Beispielverdrahtung mit dem Hadlock-Algorithmus gezeigt. Eine kurze Beschreibung der relevanten Begriffe in Zusammenhang mit dem Manhattan-Abstand befindet sich im Anhang der Arbeit.

**$d(P)$ , Detour-Nummer:** Die Nummer der Rasterpunkte entlang des Pfades, die vom Ziel weg führen.

**$MD(S, T)$ :** Der Manhattan-Abstand zwischen Startpunkt ( $S$ ) und Endpunkt ( $T$ )

**$l(P)$ :** Die Länge eines Verdrahtungswegs ( $P$ )

$$l(P) = MD(S, T) + 2 \cdot d(P)$$

Der Wert von  $MD(S, T)$  ist konstant  $\rightarrow$  Auffinden des kürzesten Wegs durch Minimierung von  $d(P)$ .

Wenn der betrachtete Rasterpunkt mit  $i$  markiert ist, werden die zulässigen Nachbarelemente, die vom Ziel weg führen, mit  $i+1$ , ansonsten mit  $i$  markiert.

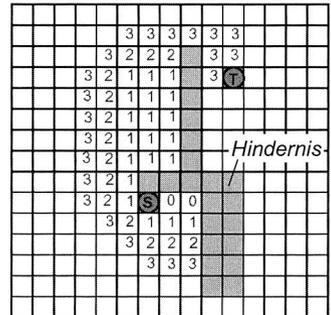


Bild 21: Hadlock-Algorithmus (Begriffe und Beispiel)

Die Modifikation bzw. Erweiterung des Hadlock-Algorithmus für das 3D-MID-Routing wird im Folgenden dargestellt.

Gegeben sei ein Rasterpunkt mit Koordinaten  $(x, y, z)$  auf dem erzeugten Raster. Es sei angenommen, dass die horizontale, die vertikale und die diagonale Verbindungslänge zwischen beiden Nachbarpunkten, wie z. B.  $(x, y, z)$  und  $(x+1, y, z)$ ,  $(x, y, z)$  und  $(x, y+1, z)$  oder  $(x, y, z)$  und  $(x+1, y+1, z)$ , gleich einer Einheit sind. Daraus folgt, dass die Länge eines Verdrahtungswegs, der  $k+1$  Rasterpunkte (inkl. Start- und Endpunkt) enthält,  $k$  Einheiten entspricht.

Gegeben sei ein Paar Rasterpunkte  $v = (x_v, y_v, z_v)$  und  $w = (x_w, y_w, z_w)$  und sei

$$d_{\min}(v, w) = \max[|x_v - x_w|, |y_v - y_w|, |z_v - z_w|] \quad (1)$$

Daraus folgt, dass der Wert von  $d_{\min}(v, w)$  gleich der unteren Schranke zur Länge jedes möglichen Weges zwischen den Rasterpunkten  $v$  und  $w$  ist.

Gegeben seien zwei nicht angrenzende Rasterpunkte  $v$  und  $q$ . So gilt eine der folgenden Gleichungen für jeden Nachbarpunkt  $w$  von Rasterpunkt  $v$ :

$$d_{\min}(w, q) = d_{\min}(v, q) - 1 \quad (2)$$

$$d_{\min}(w, q) = d_{\min}(v, q) \quad (3)$$

$$d_{\min}(w, q) = d_{\min}(v, q) + 1 \quad (4)$$

Gemäß der entsprechenden Gleichungen (2), (3) und (4) wird der Punkt  $w$  als der *Typ 1*, *Typ 2* und *Typ 3* Nachbarpunkt von  $v$  bezüglich  $q$  genannt.

Sei  $p(s, t) = [s = v_0, v_1, v_2, \dots, v_k = t]$  ein Verdrahtungsweg, der am Punkt  $s = v_0$  startet, über die nacheinander folgenden Zwischenpunkte  $v_1, v_2, \dots, v_{k-1}$  läuft und an Punkt  $t = v_k$  endet.

Demnach lassen sich die folgenden Mengen definieren:

$TYPE2(p(s, t)) = \{v_h \ (1 \leq h \leq k) \mid v_h \text{ sei ein Typ 2 Nachbarpunkt von } v_{h-1} \text{ bezüglich des Endpunkts } t\}$

$TYPE3(p(s, t)) = \{v_h \ (1 \leq h \leq k) \mid v_h \text{ sei ein Typ 3 Nachbarpunkt von } v_{h-1} \text{ bezüglich des Endpunkts } t\}$

Durch die Anwendung der oben definierten Menge kann die Detour-Nummer von  $p(s, t)$  wie im Folgenden (im modifizierten Algorithmus neu) definiert werden:

$$DETOUR(p(s, t)) = |TYPE2(p(s, t))| + 2 \times |TYPE3(p(s, t))| \quad (5)$$

wobei  $|\cdot|$  die Anzahl der Elemente in einer Menge bedeutet.

Schließlich kann die Länge des Verdrahtungswegs  $p(s, t)$  mit der folgenden Gleichung dargestellt werden:

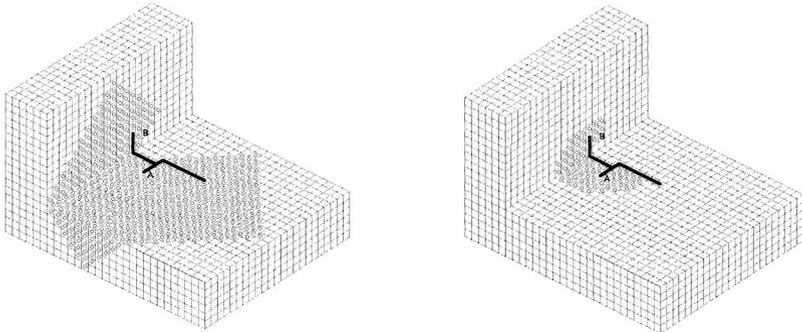
$$LENGTH(p(s, t)) = d_{\min}(s, t) + DETOUR(p(s, t)) \quad (6)$$

Da der Wert von  $d_{\min}(s, t)$  konstant ist, kann der kürzeste Weg zwischen den Rasterpunkten  $s$  und  $t$  durch die Minimierung der aktuellen  $DETOUR(p(s, t))$  gefunden werden. Zwar sind die Definition der Detour-Nummer und die Gleichung für die Länge des Verdrahtungswegs verglichen mit dem originalen Hadlock-Algorithmus verschieden, das Suchprinzip für den Verdrahtungsweg ist jedoch gleich, d. h. bei dem modifizierten 3D-MID-Routingprozess wird der zulässige Nachbarpunkt mit dem aktuellen Wert der Detour-Nummer markiert, anstatt die aktuelle Verbindungslänge zu benutzen. Auch die bereits besuchten Rasterpunkte mit der niedrigsten Detour-Nummer haben eine höhere Priorität und werden vorzugsweise ausgewählt und expandiert.

Der modifizierte Hadlock-Algorithmus hat die gleichen entscheidenden Vorteile wie 2D-Rasterverdrahtung, d. h. er kann immer einen kürzesten Verbindungsweg finden, wenn

der Weg existiert. Außerdem ist die Anzahl der besuchten Rasterelemente beim Hadlock-Algorithmus deutlich geringer als beim Lee-Algorithmus (siehe Bild 22).

Da die Rastergenerierung getrennt vom eigentlichen Routingprozess verläuft, sind die komplexen 3D-Geometriedaten vor dem Routing schon behandelt. Deswegen können die existierenden rasterbasierten 2D-Routingalgorithmen relativ leicht modifiziert bzw. erweitert werden, um sie für das 3D-MID-Routing anwenden zu können.



a. Lee-Algorithmus: 645 Rasterelemente wurden durchsucht  
 b. Modifizierter Hadlock-Algorithmus: Nur 56 Rasterelemente wurden durchsucht

Element A, B: Zwei Anschlüsse, die miteinander verbunden werden  
 Elemente, die mit O markiert sind: Durchsuchte Elemente  
 ————— : Existierte Verbindungswege oder Hindernisse

Bild 22: Vergleich zwischen dem modifizierten Hadlock-Algorithmus und dem Lee-Algorithmus für 3D-MID-Routing

### Vorgehensweise beim rasterbasierten 3D-MID-Routing

Zwar ist die in Bild 23 gezeigte Vorgehensweise des rasterbasierten 3D-MID-Routings dem rasterbasierten 2D-Routing sehr ähnlich, es gibt jedoch beim 3D-MID-Routing noch weitere spezielle Aspekte bzw. Behandlungen in jedem Schritt, die im Folgenden dargestellt werden.

-- *in Schritt 1:* Das geometrische Modell des 3D-Schaltungsträgers enthält unterschiedliche Oberflächen- und Linientypen. Die entwickelte Datenstruktur muss in der Lage sein, die verschiedenen geometrischen Daten zu speichern, insbesondere die geometrische 3D-Konnektivität der Oberflächen des Schaltungsträgers.

-- *in Schritt 2:* Da das 3D-Raster separat auf jeder Oberfläche erzeugt wird, ist eine spezielle Behandlung durchzuführen, um sicherzustellen, dass die Kanten der aufeinander treffenden Rasterelemente auf zwei unterschiedlichen Oberflächen an den gemeinsamen Kanten übereinstimmen. Gleichzeitig müssen die Design Rules, wie z. B. der Minimalabstand, berücksichtigt werden.

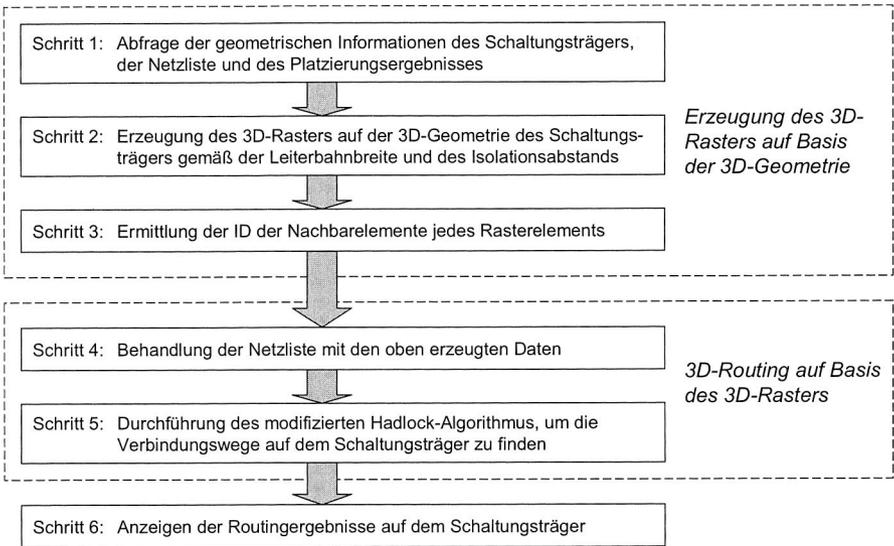


Bild 23: Vorgehensweise des rasterbasierten 3D-MID-Routings

-- in Schritt 3: Die Konnektivität und Beziehung zwischen den Rasterelementen im 3D-Routing sind komplexer als im 2D-Routing. Während im diagonalen 2D-Routing jedes Rasterelement maximal 8 Nachbarelemente hat, kann ein Rasterelement im diagonalen 3D-Routing über 8 Nachbarelemente haben (siehe Bild 24). Außerdem befinden sich einige Nachbarelemente auf anderen Flächen, wenn das Rasterelement direkt an der Grenze liegt. Deswegen ist die Information über die Konnektivitäten zwischen den Oberflächen des Schaltungsträgers eine wichtige Voraussetzung, um die richtigen Beziehungen zwischen den Rasterelementen zu bekommen.

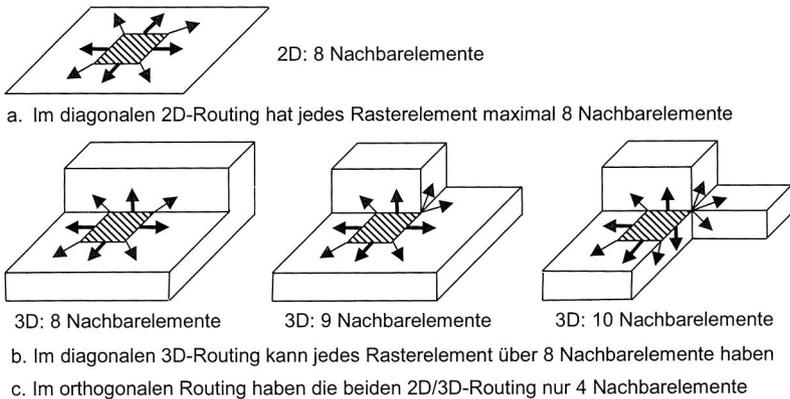


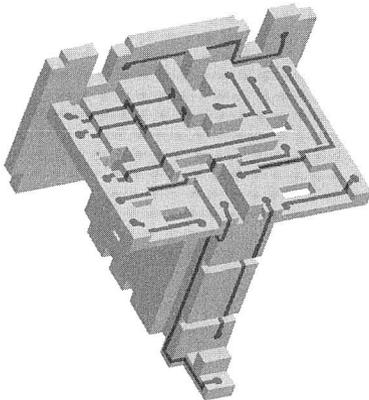
Bild 24: Die Beziehung der Rasterelemente im rasterbasierten Routingverfahren

-- in den Schritten 4 und 5: Der Verbindungsweg wird meist durch die Reihenfolge der zentralen Koordinaten der Rasterelemente repräsentiert. Da das Zentrum des Rasterelements immer innerhalb einer Fläche liegt, ist ein neuer Zwischenpunkt auf der gemeinsamen Kante zweier Flächen einzufügen, um sicherzustellen, dass der Verbindungsweg von einer Fläche über die gemeinsame Kante auf der anderen Fläche kontinuierlich weiter verlaufen kann, wenn der Weg auf mehr als einer Fläche verlegt ist.

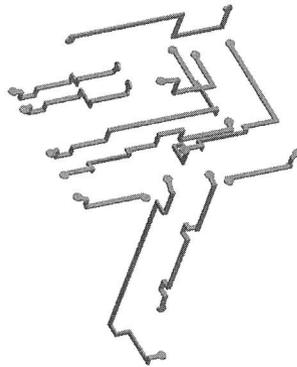
-- in Schritt 6: In MIDCAD wird der geroutete Verbindungsweg durch Linien grafisch dargestellt. Die konkrete Geometrie der Leiterbahnen wird danach als „Datum Curve“ und „Sweep“ Features durch die entwickelten Funktionen erzeugt (siehe Kapitel 4.5).

### Experimentelle Resultate

Wie bereits dargestellt, kann der modifizierte Hadlock-Algorithmus einen kürzesten Verbindungsweg durch die Suche in einem deutlich kleineren Rastergebiet finden als der Lee-Algorithmus. Ein Beispiel wurde vorher schon in Bild 22 dargestellt, auf dem das 3D-MID-Routing separat durch den Lee-Algorithmus und den modifizierten Hadlock-Algorithmus realisiert wurde. Bei dem Routing mit dem Lee-Algorithmus werden 645 Rasterelemente durchsucht, während bei dem Routing mit dem modifizierten Hadlock-Algorithmus nur 56 Rasterelemente durchsucht werden.



3D-Geometriemodell des Schaltungsträgers  
mit gerouteten Verbindungen



3D-Leiterbahnen ohne  
Schaltungsträger und Bauteile

*Bild 25: Anwendung des rasterbasierten 3D-Routingalgorithmus*

Bild 25 enthält ein anderes Routingbeispiel für den modifizierten Hadlock-Algorithmus. Geroutet sind 11 Verbindungen auf Basis des 3D-Rasters mit 3050 Rasterelementen auf dem Schaltungsträger. Das Geometriemodell des Schaltungsträgers hat insgesamt 128 Flächen und 442 Kanten. Die Berechnung dauerte 153 Sekunden auf einen Rechner mit einer Pentium IV 1.8GHz CPU und 512MB RAM.

## Vorteile und Nachteile

Der Hauptvorteil der rasterbasierten 3D-Routingmethode ist, dass der kürzeste Verbindungsweg gefunden wird, wenn dieser existiert. Als Nachteile sind ein hoher Zeit- und Speicherplatzbedarf zu nennen. Die Nachteile treten besonders dann hervor, wenn ein sehr fein aufgelöstes Raster oder ein komplexer Schaltungsträger vorliegen. Ein weiterer Nachteil liegt in der Beschränkung auf rechtwinklige flache Oberflächen, da das Raster nur aus rechtwinkligen Elementen gleicher Größe besteht. An den gemeinsamen Kanten zweier Oberflächen müssen auch die Kanten der aufeinander treffenden Rasterelemente übereinstimmen.

### 4.3.2 Linienbasierter 3D-Routingalgorithmus

Die Nachteile der rasterbasierten 3D-Routingmethode, insbesondere der große Speicherbedarf und die langen Rechenzeiten, führten zur Entwicklung von rasterfreien Algorithmen für das 3D-MID-Routing. Der linienbasierte 3D-Routingalgorithmus wird im Folgenden dargestellt.

Der linienbasierte 3D-Routingalgorithmus wird grundsätzlich in zwei Schritte unterteilt:

- Im ersten Schritt entspricht die Aufgabe derjenigen der globalen Verdrahtung. Die bekannte A\*-Suchheuristik wird zur Identifikation der für das 3D-Routing zu verwendenden Verdrahtungsoberflächen oder -regionen (nicht nur in einer Fläche) nach dem Prinzip des kürzesten Verbindungswegs verwendet, und damit eine Reduktion auf ein quasi 2D-Routingproblem erreicht. In dieser Phase werden nur ungefähre Verbindungswege auf einer Fläche oder einer Reihenfolge der Oberflächen des Schaltungsträgers festgelegt.
- Im Anschluss daran erfolgt das eigentliche Routing, indem auf den zuvor zugewiesenen Verbindungsflächen eine Erweiterung des Hightower-Algorithmus nach dem Prinzip der geradlinigen Expansion zu Anwendung kommt. Der Routingprozess erfolgt somit für jede Oberfläche getrennt. In dieser Phase wird die detaillierte Geometrie jedes Verbindungswegs festgelegt.

### Applikation des A\*-Algorithmus

Der A\*-Algorithmus dient zur Ermittlung des kostengünstigsten Pfades zwischen einem Start- und einem Zielknoten innerhalb eines Graphen oder Baums. Er wird bei vielen Suchproblemen im Bereich der Künstlichen Intelligenz [97] und bei 2D-Routingproblemen (z. B. [12], [114]) eingesetzt, die als die Suchprobleme in Graphen dargestellt werden können. Durch die Anwendung der auf dem A\*-Algorithmus basierenden Suchstrategie kann für das 3D-MID-Routing eine Reihenfolge der Flächen oder Regionen gefunden werden, auf die der kürzeste Verbindungsweg verlegt werden kann.

Für ein Zweipunktnetz, in dem der Startpunkt auf der Fläche  $s$  und der Zielpunkt auf der Fläche  $d$  liegen, ist es die Aufgabe des A\*-Algorithmus, eine Reihenfolge der Flächen

zwischen den Flächen  $s$  und  $d$  zu finden, auf denen der kostengünstigste (d. h. kürzeste) Pfad zwischen Start- und Zielpunkt liegt.

Sei die Fläche  $s$  der Startknoten eines Graphen. Auf der Suche nach der Fläche  $d$  (Zielknoten) über die Nachbarflächen entspricht jede besuchte Fläche einem Knoten des Graphen. Jede Kante zwischen zwei Knoten hat ein bestimmtes Gewicht, das die Wegkosten darstellt. Der Graph hat zuerst nur zwei Knoten  $s$  und  $d$  ohne Kante dazwischen und wird während des Suchprozesses dynamisch erzeugt, bis es einen Verbindungsweg zwischen den Knoten  $s$  und  $d$  gibt, d. h. bis eine Reihenfolge der Flächen für die Verbindung gefunden ist. Die Gesamtkosten des Verbindungswegs ist die Summe der Gewichte aller Kanten auf dem Weg. In der Regel ist es erwünscht, den kostengünstigsten Pfad im Graphen zu finden.

Der A\*-Algorithmus enthält eine Kosten-Funktion  $f^*(n)$ , die die günstigsten Kosten des Verbindungswegs über den Knoten  $n$  berechnet. Beim Expandieren der Knoten innerhalb eines Graphen wird diese Funktion eingesetzt. Die Funktion ist definiert mit

$$f^*(n) = g^*(n) + h^*(n) \quad (7)$$

wobei  $g^*(n)$  für die Summe der günstigsten Kosten vom Startknoten  $s$  bis Knoten  $n$  steht. Wenn der Verbindungsweg von  $s$  bis  $n$  schon gefunden ist, kann der Wert von  $g^*(n)$  exakt berechnet werden.  $h(n)$  ist eine heuristische Schätzfunktion und bezeichnet die Kosten vom Knoten  $n$  bis Zielknoten  $d$ . Den Wert der minimalen Kosten vom Knoten  $n$  bis Zielknoten  $d$  gibt  $h^*(n)$  an. Wichtig hierbei ist, dass die Schätzfunktion die untere Schranke bildet, also wirklich die theoretisch minimalen Kosten zur Erreichung des Zielknotens verwendet, um die Anwendung des A\*-Algorithmus zuzulassen. Da der Verbindungsweg vom aktuellen Knoten  $n$  zum Zielknoten  $d$  noch nicht gefunden wurde, ist die Definition der Schätzfunktion  $h^*(n)$  von den Eigenschaften der Problemdomäne abhängig. Für das 3D-MID-Routingproblem kann hier der euklidische Abstand zwischen den Knoten  $n$  und  $d$  im Dreidimensionalen als Schätzfunktion für  $h^*(n)$  dienen.

Jeder Knoten im Graphen entspricht nicht nur einer Fläche des 3D-Schaltungsträgers, sondern enthält auch einen so genannten Kostenpunkt, der auf der entsprechenden Fläche steht und zur Kalkulation der Wegkosten dieses Knotens verwendet wird. Nach dem Prinzip der günstigsten Kosten soll der aktuelle Kostenpunkt den kürzesten Abstand vom Kostenpunkt des Vorgängerknotens haben. Deswegen stehen die Kostenpunkte immer auf der gemeinsamen Kante zweier benachbarter Flächen – mit Ausnahme von Start- und Endknoten. Der Mittelpunkt der gemeinsamen Kante stellt eine einfache Auswahl für den Kostenpunkt dar. In Hinsicht auf die Definition des Kostenpunkts können die Kosten  $g^*(n)$  durch die Summe der Abstände vom Startpunkt (als Kostenpunkt des Startknotens  $s$ ) bis zu dem aktuellen Kostenpunkt des Knotens  $n$  berechnet werden. Die Kosten  $h^*(n)$  werden nach der euklidischen Distanz zwischen dem aktuellen Kostenpunkt und dem Zielpunkt berechnet.

Das eigentliche Vorgehen des A\*-Algorithmus ist relativ einfach: Bei jedem Fortschritt wird derjenige Knoten expandiert, dessen Kosten  $f^*$  am geringsten sind. Soll neben den

Kosten des günstigsten Pfades auch der Pfad selbst gespeichert werden, so ist neben den Kosten auch der Vorgänger eines jeden Knotens zu speichern. Die konkrete Darstellung des Verfahrens und die programmtechnische Realisierung können in der Literatur (z. B. [17], [31], [97]) nachgeschlagen werden.

Ein einfaches Beispiel in Bild 26 soll die Applikation des A\*-Algorithmus zur Ermittlung der Verbindungsflächen auf einem 3D-Schaltungsträger erläutern. Der Startpunkt und Endpunkt eines Zweipunktnetzes befinden sich jeweils auf den Flächen 1 und 5. Der rechte Graph zeigt die 3D-Konnektivität der Flächen, er wird bei der Suche dynamisch generiert. Jeder Knoten entspricht einer Fläche und eine Kante verbindet zwei Knoten, wenn die entsprechenden Flächen benachbart sind. Durch die Anwendung des A\*-Algorithmus wird eine Reihenfolge der Knoten (Flächen)  $[1 \rightarrow 3 \rightarrow 4 \rightarrow 5]$  im Graphen zwischen den Knoten 1 und 5 gefunden, auf der der kostengünstigste Pfad zwischen Start- und Endpunkt liegt. Im Vergleich zu anderen Suchalgorithmen (z. B. Breitensuche) ist der Hauptvorteil ein geringer Suchaufwand. Der Nachteil ist, dass die definierten Wegkosten der Kante die Länge des zu verlegenden Verbindungswegs nicht exakt repräsentieren können, weil der Mittelpunkt auf der gemeinsamen Kante als Kostenpunkt ohne Berücksichtigung anderer Einflüsse dient. Bei der Suche sind noch einige spezielle Aspekte zu berücksichtigen, z. B. verbotene Flächen, eine zu kurze Kante der Fläche usw. Damit kann ein gültiger Pfad in Graphen für das nachfolgende detaillierte Routing gefunden werden.

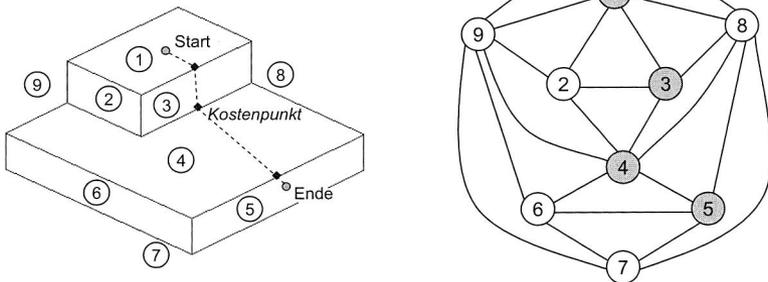


Bild 26: Applikationsbeispiel des A\*-Algorithmus beim 3D-MID-Routing

### Erweiterung des Hightower-Algorithmus

Der Hightower-Algorithmus ist eine linienbasierte 2D-Routingmethode, und verzichtet auf die Rasterung. In Bild 27 wird eine Routingmethode nach dem Hightower-Typ [10] dargestellt, in der die Versuchslinien eine bestimmte Breite haben können. Der Ablauf der linienbasierten Verdrahtung wird nachfolgend kurz dargestellt.

Zu Beginn werden sowohl vom Startpunkt  $S$  als auch vom Zielpunkt  $T$  ausgehende vertikale und horizontale Versuchslinien (Fluchtlinien, *escape lines*) erzeugt. Diese Linien

werden so weit ausgedehnt, bis sie auf ein Hindernis oder Layoutrand treffen. Wenn sich eine vom Startpunkt und eine vom Endpunkt ausgehende Fluchtlinie nicht schneiden, müssen neue, auf den bisher erzeugten Fluchtlinien senkrecht stehende Fluchtlinien durch die Fluchtpunkte (*escape points*) erzeugt werden. Die Fluchtpunkte sind dadurch gekennzeichnet, dass mindestens eine von ihnen ausgehende horizontale oder vertikale Fluchtlinie in einer Richtung nicht durch das aktuelle Hindernis verläuft. In Bild 27 sind die Punkte *E* derartige Fluchtpunkte. Bei mehreren Fluchtpunkten werden diejenigen ausgewählt, die dem aktuellen Ausgangspunkt am nächsten liegen. Dieses Vorgehen wird solange fortgesetzt, bis sich eine vom Startpunkt und eine vom Zielpunkt erzeugte Linie schneiden. Die anschließende Rückverfolgung auf den Fluchtlinien ergibt einen gültigen Verbindungsweg zwischen Start- und Zielpunkt.

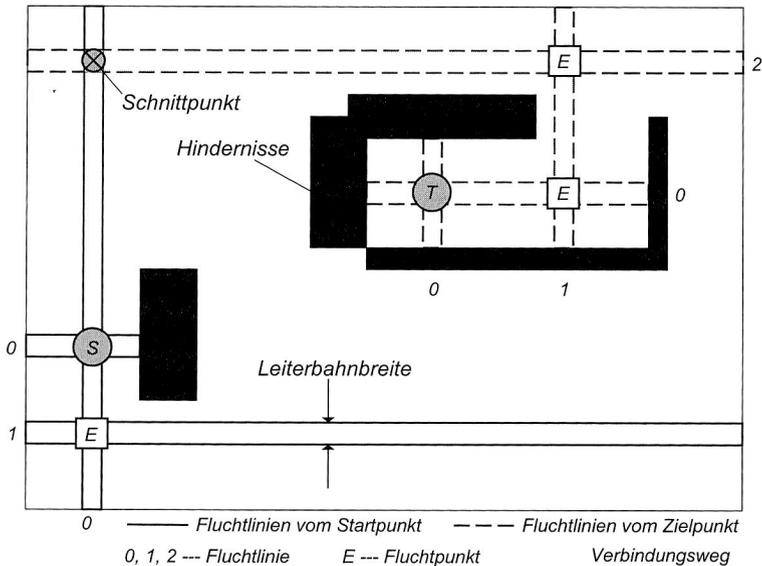


Bild 27: Routingverfahren nach dem Hightower-Typ

In Hinsicht auf die Eigenschaft des 3D-MID-Routings kann die linienbasierte Hightower-Typ Routingmethode den folgenden grundlegenden Anforderungen entsprechen:

- Geringe Rechenzeit und niedriger Speicherbedarf;
- individuelle Leiterbahnbreiten für unterschiedliche Leitungen;
- komplexe Verdrahtungsflächen, die weder flach noch rechteckig sind, oder verschiedene Formen, die komplexe innere und äußere Grenzen haben.

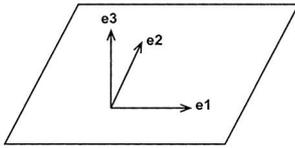
Da die originalen Hightower-Typ Routingalgorithmen nur das 2D-Routingproblem auf der kontinuierlichen Ebene behandeln können, sind sie zu erweitern, um das komplexe

3D-MID-Routingproblem lösen zu können. In die Erweiterung des Hightower-Algorithmus für das 3D-MID-Routing sind die folgenden Aspekte eingegangen:

*Modifizierte Datenstrukturen:*

Wenn eine geeignete Datenstruktur verwendet wird, kann der Algorithmus schnell und effektiv auf einer kontinuierlichen Ebene durchgeführt werden. Die Datenstruktur, die von Lauther in [59] präsentiert wurde, ist in der Lage, unterschiedliche Breiten der Leitungen im Algorithmus zu behandeln. Sie dient als Grundlage der Datenstruktur im modifizierten Hightower-Algorithmus.

Die wichtigste Technik für die Erweiterung des Algorithmus vom 2D-Routing ins 3D-Routing ist die Transformation der dreidimensionalen Koordinaten  $(x, y, z)$  der Punkte auf den Oberflächen des Schaltungsträgers in ein 2D-Format, das direkt im originalen 2D-Routingalgorithmus benutzt werden kann. Da die meisten geometrischen Flächen (z. B. Ebenen und Zylinder) mathematisch durch formelmäßig ausdrückbare Zusammenhänge beschreibbar (d. h. analytisch beschreibbar) sind und von zwei Variablen ( $u$  und  $v$ ) parametrisiert werden, können die Koordinaten  $(x, y, z)$  der Punkte auf der Fläche durch die Parameter  $u$ ,  $v$ , und  $ID$  der Fläche dargestellt werden. So werden z. B. die Koordinaten eines Punkts auf einer Ebene nach Formel (8) berechnet.



$$(x, y, z) = u * e1 + v * e2 + origin \quad (8)$$

*Bild 28: Geometrische Repräsentation einer Ebene*

In Formel (8) und Bild 28 repräsentiert  $e1$  eine Vektoreinheit in der  $u$ -Richtung,  $e2$  eine Vektoreinheit in der  $v$ -Richtung,  $e3$  die Normale der Ebene, und  $origin$  den Koordinatennullpunkt der Fläche. Die Werte von  $e1$ ,  $e2$ ,  $e3$  und  $origin$  sind von der  $ID$  der Fläche abhängig. Indem die Repräsentationen von  $x$  und  $y$  in dem Algorithmus und den Datenstrukturen durch die Parameter  $u$  und  $v$  ersetzt werden, kann der Algorithmus auf das Routing auf die Oberflächen des Schaltungsträgers angewandt werden. Statt ein  $HV$ -Routing in horizontaler und vertikaler Richtungen durchzuführen, ist das Routingverfahren eigentlich ein  $UV$ -Routing, d. h. das Routing wird in flächengehörigen  $u$ - und  $v$ -Richtungen auf unterschiedlichen Flächen mit unterschiedlicher  $ID$  durchgeführt.

Im 3D-MID-Routing verteilen sich alle Hindernisse (z. B. platzierte Bauelemente, geroutete Leitungen etc.) je nach Designbedürfnis auf den verschiedenen Oberflächen des Schaltungsträgers. Durch das Einfügen der  $ID$  in Datenstrukturen hat zur Beschreibung der Hindernisse die entsprechende Klasse, die alle Typen von Hindernissen repräsentiert, eine einheitliche Darstellung. Auf Basis der modifizierten Datenstruktur werden die entsprechenden 4-dimensionalen binären Suchbäume [7], [59] für jede Verdrahtungsfläche aufgebaut, um die Hindernisse auf derselben Fläche speichern zu können.

*Behandlung der inneren und äußeren Grenzen:*

Der originale Hightower-Algorithmus kann nur das 2D-Routing auf einer einfachen rechtwinkligen Ebene behandeln. Im „*escape*“ Verfahren des originalen Algorithmus erfolgt die Suche nach Fluchtpunkten auf Basis der Information über „*covers*“ (Hindernisse). Dabei werden die Auswirkungen der inneren und äußeren Grenzen der Ebene nicht berücksichtigt, weil die Grenzen für eine rechteckige Ebene zu einfach sind und im Algorithmus keine Hindernisse darstellen. Im Vergleich zu traditionellen PCBs sind die Flächen des 3D-Schaltungsträgers nicht immer rechteckig, haben verschiedene Formen und komplexe innere und äußere Grenzen, insbesondere bei 2K-Spritzguss-Teilen. Außerdem liegen keine Hindernisse in manchen Situationen auf einer Fläche, weshalb der Fluchtpunkt durch das „*escape*“ Verfahren nicht gefunden werden kann.

In der erweiterten Version für das 3D-MID-Routing werden alle Kanten der inneren Grenzen und die Kanten der äußeren Grenze, die mit den konkaven Punkten verbunden sind, zusätzlich als spezielle Hindernisse behandelt. Während das allgemeine Hindernis (z. B. platzierte Bauelemente und geroutete Leitungen) immer ein rechteckiges Gebiet im  $uv$ -Koordinatensystem besitzt, entspricht das spezielle Hindernis einer Linie und wird als ein spezielles Rechteck mit Nullbreite betrachtet. Es gilt, die folgenden drei Typen von speziellen Hindernissen zu unterscheiden (siehe auch Bild 29):

- Wenn die Kante (z. B.  $k_2$ ) eine Linie und parallel zur  $u$ -Richtung ist, ist die Breite des speziellen Hindernisses in  $v$ -Richtung *Null*, d. h. die  $v$ -Werte des Hindernisses in der entsprechenden Datenstruktur sind gleich.
- Wenn die Kante (z. B.  $k_1$ ) eine Linie und parallel zur  $v$ -Richtung ist, ist die Breite des speziellen Hindernisses in  $u$ -Richtung *Null*, d. h. die  $u$ -Werte des Hindernisses in der entsprechenden Datenstruktur sind gleich.
- Wenn die Kante (z. B.  $k_3$  oder  $k_4$ ) ein Bogen oder eine Linie ist, die weder zur  $v$ -Richtung noch zur  $u$ -Richtung parallel ist, dann werden zwei oder mehrere spezielle Hindernisse entlang der  $u$ - und  $v$ -Richtung erzeugt.

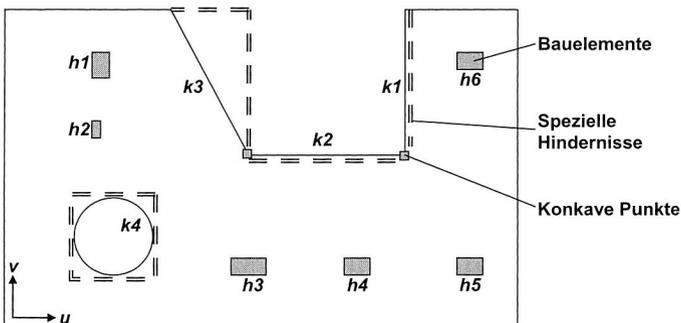


Bild 29: Definition der speziellen Hindernisse für 3D-MID-Routing

Die Auswirkung der speziellen Hindernisse entspricht nicht der von allgemeinen Hindernissen. Einerseits wird im modifizierten „*findcover*“ Verfahren die Länge der Versuchslinien nicht durch spezielle Hindernisse festgesetzt, sondern direkt durch das Durchschneiden mit der realen Grenze. Eine zusätzliche entwickelte Funktion wurde im „*findcover*“ Verfahren eingefügt, um die Versuchslinien durch die Grenze zu schneiden, wenn die getroffenen „*covers*“ die speziellen Hindernisse sind oder kein „*cover*“ getroffen wird. Andererseits müssen die speziellen Hindernisse im modifizierten „*escape*“ Verfahren berücksichtigt werden, weil sie bei der Suche nach Fluchtpunkten die gleiche Auswirkung wie die allgemeinen Hindernisse haben. Durch die Definition der speziellen Hindernisse und die zusätzliche Funktion hat der erweiterte Hightower-Algorithmus die Fähigkeit, das Routing auf der Fläche mit komplexen inneren und äußeren Grenzen durchführen zu können.

#### *Verbesserung der Tiefensuche (Depth-first search, DFS):*

Der Hightower-Algorithmus basiert auf einer einfachen Tiefensuchheuristik, um die Vorteile der Tiefensuchstrategie (geringe Rechenzeit und niedriger Speicherbedarf) zu erreichen, anstatt eine im rasterbasierten Algorithmus angewandte Breitesuchstrategie wie die Wellenausbreitung zu benutzen. Die Hauptschwierigkeit besteht nun darin, geeignete Fluchtpunkte zu finden. Im originalen Tiefensuchverfahren nach dem Hightower-Algorithmus wird nur ein Fluchtpunkt auf einer Versuchslinie heuristisch festgelegt. Wenn das Layout oder die Grenzen kompliziert sind, ist diese Vorgehensweise zwar günstig im Bezug auf Laufzeit- und Speicherplatzbedarf, kann jedoch dazu führen, dass eine ungeschickte Wahl von Fluchtpunkten ohne rekursive Funktion zu keinem Ergebnis führt, obwohl eine Verbindung existiert.

Im erweiterten Hightower-Algorithmus wurde das „*escape*“ Verfahren bei der Tiefsuche verbessert. Mehrere Fluchtpunkte werden im Verfahren gesucht. Zur Behandlung der gefundenen Fluchtpunkten wird eine zusätzliche Datenstruktur entwickelt, in der nicht nur die entsprechenden Werte der Parameter  $u$  und  $v$ , sondern auch die Fluchtrichtung und der Zeiger, der den *PARENT*-Fluchtpunkt zeigt, gespeichert werden. Auf Basis der gleichen Tiefensuchstrategie werden eine rekursive Funktion und ein Tiefgrenzwert im Verfahren eingefügt. Damit wird die Leistung des Hightower-Algorithmus verbessert, um die Chancen bei der Suche nach einer Verbindung auch auf Flächen mit komplexer Grenze und komplexem Layout zu vergrößern.

#### *Erweiterung auf Zylinderfläche:*

Da die Oberflächen des Schaltungsträgers nicht immer eben sind, muss der Algorithmus für das 3D-MID-Routing auf anderen Flächentypen erweitert werden. Da die Zylinderfläche neben der Ebene der häufigste Flächentyp in einem 3D-Geometriemodell ist, wird die Erweiterung des Hightower-Algorithmus auf Zylinderflächen im Folgenden dargestellt.

Durch die Anwendung der oben modifizierten Datenstrukturen kann der Algorithmus auf Zylinderflächen erweitert werden. Gemäß der geometrischen Repräsentation werden

die Koordinaten der Punkte auf einer Zylinderfläche durch Formel (9) berechnet (siehe auch Bild 30).

$$(x, y, z) = \text{radius} * [\cos(u) * e1 + \sin(u) * e2] + v * e3 + \text{origin} \quad (9)$$

In Formel (9) repräsentiert *radius* den Radius des Zylinders und hängt zusammen mit *e1*, *e2*, *e3* und *origin* von der Fläche (*ID*) ab. Deswegen können die Koordinaten (*x*, *y*, *z*) der Punkte auf der Zylinderfläche von Parameter *u*, *v*, und *ID* dargestellt werden. Indem die Repräsentationen von *x* und *y* im Algorithmus und den Datenstrukturen durch die Parameter *u* und *v* ersetzt werden, kann der Algorithmus für das Routing auf der Zylinderfläche angewandt werden. Mit dieser Methode kann der Algorithmus im Prinzip auf verschiedene analytisch beschreibbare Flächentypen erweitert werden. Das Routing auf der Zylinderfläche wird auch in der *uv*-Richtung durchgeführt, wobei die *v*-Richtung parallel zur Achse der Zylinderfläche ist und die *u*-Richtung gleich der peripherischen Richtung der Zylinderfläche ist. Während die Versuchslinie in *v*-Richtung einer Linie auf der Zylinderfläche entspricht, entspricht die Versuchslinie in *u*-Richtung einem Bogen auf der Zylinderfläche. Da der Wert von *u* in Formel (9) ein Winkelmaß ist, wurde ein zusätzlicher Parameter in den Algorithmus eingefügt, um die Länge der Versuchslinie in der *u*-Richtung zu berechnen. Für eine Zylinderfläche ist der Wert des Parameters gleich *radius*, für eine Ebene gleich 1. Damit ist der erweiterte Algorithmus in der Lage, das Routing auf beiden Flächentypen (Ebene und Zylinder) zu behandeln.

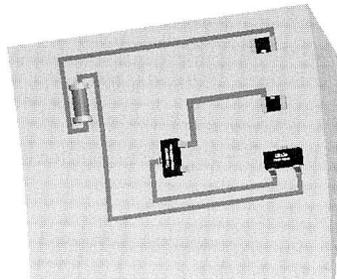
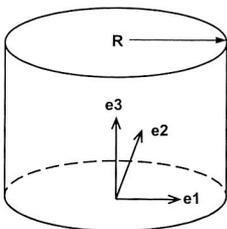


Bild 30: Geometrische Repräsentation eines Zylinders und Routingbeispiel auf einer Zylinderfläche

### Vorgehensweise beim linienbasierten 3D-MID-Routing

Auf Basis des oben dargestellten A\*-Algorithmus und der Erweiterung des Hightower-Algorithmus kann das linienbasierte 3D-MID-Routing in die folgenden Hauptschritte unterteilt werden.

**Schritt 1: Ausgangsdefinition und Einstellung;** Im Routing-Setup wird der Vorgabewert der Leiterbahnbreite für alle Leitungen definiert. Wenn nötig können die individuellen Breiten für einzelne Leitungen auch spezifiziert werden. Der minimale Isolierabstand, der eine Randbedingung darstellt und während des Routings einzuhalten ist, wird auch

hier eingestellt. Außerdem kann die Netzreihenfolge im Setup-Fenster manuell eingestellt werden, um ein besseres Routingergebnis zu erreichen.

*Schritt 2: Sammeln und Speichern von Informationen über Hindernisse;* Die Arbeiten in diesem Schritt dienen hauptsächlich der nachfolgenden Anwendung des modifizierten Hightower-Algorithmus. In diesem Schritt sind nicht nur die allgemeinen Hindernisse (z. B. die platzierten Bauelemente, die verlegten Leiterbahnen und die vordefinierten Keepout-Regions), sondern auch die speziellen Hindernisse (alle Kanten der inneren Grenzen und die mit dem konkaven Punkt verbundenen Kanten der äußeren Grenze) gesammelt. Gemäß der vordefinierten Datenstruktur für Hindernisse werden die 4-dimensionalen binären Bäume nach der Sammlung aufgebaut, um die Daten der Hindernisse zu speichern und in nachfolgenden Schritten schnell darauf zuzugreifen.

*Schritt 3: Sammeln und Speichern von Informationen über Verdrahtungsregionen auf dem Schaltungsträger;* Die Arbeiten in diesem Schritt dienen hauptsächlich der nachfolgenden Anwendung des A\*-Algorithmus. Verdrahtungsregionen sind alle Flächen des Schaltungsträgers mit Ausnahme der vordefinierten „Keepout Surfaces“ (siehe Kapitel 4.5). Zur Beschreibung der Verdrahtungsregionen wurde eine Datenstruktur entwickelt, die die entsprechende äußere Kontur und die gültigen Kanten (d. h. über diese Kante kann das Routing auf andere Nachbarfläche expandieren) enthält.

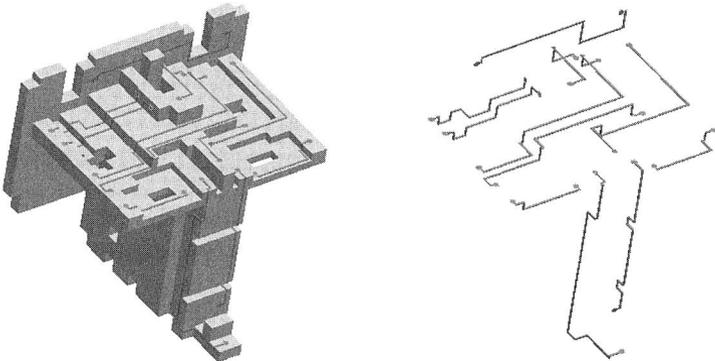
*Schritt 4: LOOP:* Entsprechend der Netzreihenfolge wird das Routing auf Basis der oben dargestellten Algorithmen für jedes Netz sequentiell durchgeführt.

- a. *Suche die Verbindungsflächen;* Auf Basis der Informationen über Verdrahtungsregionen wird eine Reihe miteinander benachbarter Flächen zwischen Start- und Endpunkt durch Anwendung des A\*-Algorithmus gefunden.
- b. *Detailliertes Routing;* Wenn Start- und Endpunkt auf derselben Fläche liegen, ist das Routing relativ einfach und der erweiterte Hightower-Algorithmus kann direkt auf dieser Fläche angewandt werden. Wenn Start- und Endpunkt nicht auf derselben Fläche liegen, beträgt die Anzahl der Verbindungsflächen mindestens zwei. Das detaillierte Routing muss vom Start- bis zum Endpunkt entlang den zuvor gefundenen Verbindungsflächen stattfinden. In diesem Fall besteht die Routingaufgabe auf jeder Verbindungsfläche nicht in der Suche nach einem Weg zwischen zwei Punkten, sondern zwischen einem Punkt und einer gemeinsamen Kante zweier benachbarter Verbindungsflächen. Durch die modifizierte Version des Hightower-Algorithmus kann das Routingverfahren auf den Verbindungsflächen zwischen Start- und Endpunkt bidirektional durchgeführt werden, um so ein besseres Routingergebnis zu erreichen.
- c. *Füge die neuen Hindernisse im Routingverfahren hinzu;* Die neuen verlegten Verbindungswege müssen als Hindernisse für die nachfolgend zu verdrahtenden Verbindungen behandelt werden und deshalb vor dem nächsten Routing in die entsprechenden Binärbäume eingefügt werden.

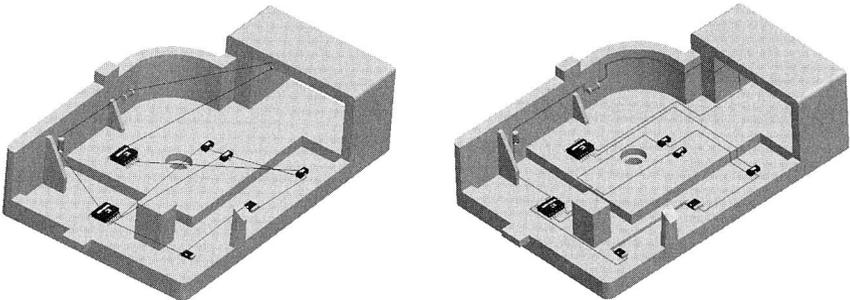
*Schritt 5: Rip-up und Reroute;* Wegen der Komplexität des 3D-MID-Routingproblems und des sequentiellen Routingverfahrens ist es möglich, dass in einer Schleife nicht für alle Netze ein erfolgreicher Verbindungsweg gefunden wird, weil die bereits verlegten Netze die noch zu verdrahtenden Verbindungen blockieren. Zur Lösung eines solchen Problems werden die Funktionen "Rip-up" und "Reroute" eingesetzt (siehe Kapitel 4.5.1).

### Experimentelle Resultate

Ein Routingbeispiel ist in Bild 31(a) dargestellt. Das 3D-Geometriemodell des Schaltungsträgers entspricht dem Modell in Bild 25, das insgesamt 128 Flächen und 442 Kanten hat. Für die gleiche Routingaufgabe auf einem Rechner mit Pentium IV 1.8GHz CPU und 512MB RAM ist die Laufzeit des rasterbasierten 3D-Routingalgorithmus ungefähr 153 Sekunden. Durch die Anwendung des linienbasierten 3D-Routingalgorithmus wird die Laufzeit auf nur 35 Sekunden reduziert. Ein anderes Routingbeispiel ist in Bild 31(b) dargestellt. Dadurch wird deutlich, dass der linienbasierte 3D-Routingalgorithmus in der Lage ist, ein Routingproblem auf einer Zylinderfläche und mit unterschiedlicher Leitungsbreite behandeln zu können.



a. Routing auf einem Schaltungsträger durch linienbasierten 3D-Routingalgorithmus



b. Routing auf einem Schaltungsträger mit komplexer Geometrie und komplexen Randbedingungen

**Bild 31:** Anwendungsbeispiele des linienbasierten 3D-Routingalgorithmus

## Vorteile und Nachteile

Die Vorteile der oben dargestellten linienbasierten 3D-Routingmethode liegen nicht nur in einem wesentlich geringeren, von der Oberflächenvernetzung unabhängigen Suchaufwand, sondern auch in der prinzipiellen Anwendbarkeit auf andere Flächentypen, wie beispielsweise Zylinder. Der Geschwindigkeitsvorteil gegenüber den rasterbasierten Verfahren ist beim 3D-MID-Routing mit wenigen, aber relativ großen elektronischen Bauelementen deutlich hoch. Durch den Verzicht auf eine diskrete Oberflächenvernetzung erfolgt das Routing außerdem kontinuierlich und es können simultan Verbindungen mit unterschiedlicher Leitungsbreite verlegt werden. Als nachteilig erweist sich, dass nicht immer der kürzeste Pfad gefunden wird. Weiterhin ist nicht sichergestellt, dass eine existierende Lösung auch tatsächlich gefunden wird.

### 4.3.3 Rasterfreier 3D-Routingalgorithmus

Das Streben nach hoher Effizienz von Rechenzeit und Speicherbedarf führte zur Entwicklung der rasterfreien Algorithmen für das 3D-MID-Routing. Der oben dargestellte linienbasierte 3D-Routingalgorithmus ist zwar rasterfrei, kann aber nicht sicherstellen, einen existierenden Verbindungsweg zu finden. Um die Nachteile des auf dem Hightower-Algorithmus basierenden Routings zu vermeiden, wurden viele neue rasterunabhängige Routingalgorithmen auf Basis der Technik der Berechnungsgeometrie entwickelt. Der gemeinsame Charakter dieser Algorithmen ist, dass sie einen Graphen benutzen, der zwar weniger Knoten als das originale Raster besitzt, aber immer einen Verbindungsweg zwischen Start- und Endpunkt enthält. Die Verdrahtung in rasterfreien Algorithmen wird als graphentheoretisches Suchproblem betrachtet und durch die geeigneten Suchalgorithmen (z. B. A\*-Algorithmus) gelöst.

In MID-Produkten verlaufen die Leiterbahnen nicht nur auf den Oberflächen des Schaltungsträgers, sondern auch innerhalb des Körpers. Über eine Durchkontaktierung kann die Leiterbahn beispielsweise von einer Fläche zu einer anderen, nicht benachbarten Fläche übergehen. Gegenüber dem 2D-Routing auf Mehrebenen-Leiterplatten mit Durchkontaktierung zeichnet sich das 3D-MID-Routing mit Durchkontaktierung durch die folgenden besonderen Eigenschaften aus (siehe auch Tabelle 1):

- Die beiden Flächen müssen nicht parallel und nicht gleich groß sein. Die Fläche kann außer einer Ebene auch eine Fläche eines anderen Typs sein, z. B. eine Zylinderfläche.
- Die Richtung der Leiterbahnen auf einer Fläche ist in der Regel nicht hinsichtlich der Richtung (z. B. entweder horizontal oder vertikal) beschränkt.
- Die Bauelemente können auf beiden Flächen platziert werden.

Die modifizierten Hightower- und Hadlock-Algorithmen können nur das 3D-MID-Routing auf den Flächen behandeln und berücksichtigen dabei keine Durchkontaktierungen. Deshalb waren weitere Algorithmen zu entwickeln, um das 3D-MID-Routing mit Durch-

kontaktierung zu realisieren. Auf Basis des rasterfreien 2D-Routingalgorithmus DUNE [13], [14], [15], [16] wurde ein 3D-Routingalgorithmus entwickelt, der nicht nur die Vorteile der Hadlock- und Hightower-Algorithmen besitzt, sondern auch das Routing innerhalb eines Körpers ermöglicht.

### Erweiterung des DUNE-Algorithmus

Der rasterfreie DUNE-Algorithmus benutzt einen so genannten Verbindungsgraph (*connection graph*), um das Routingproblem zu lösen. Deswegen ist die Erzeugung des geeigneten Verbindungsgraphen eine wichtige und zentrale Aufgabe zur Realisierung des rasterfreien Algorithmus. Im Folgenden wird zuerst die Erzeugungsmethode des Verbindungsgraphen im DUNE-Algorithmus vorgestellt. Weiterhin werden die relevanten Techniken, wie z. B. die Beschreibungsmethode des Verbindungsgraphen und deren Erweiterung auf das 3D-MID-Routing dargestellt.

#### *Erzeugung des Verbindungsgraphen*

Im DUNE-Algorithmus wurde ein solcher Verbindungsgraph benutzt, der als „*Non-Uniform Grid Graph*  $G_s$ “ bezeichnet wird. In diesem Verbindungsgraph  $G_s$  kann ein rasterfreier Verbindungsweg mit dem kürzesten Pfad für das Mehrebenen-Routingproblem mit verschiedenen Leiterbahnbreiten gefunden werden.

Der Verbindungsgraph wurde auf Basis der Ausbreitung der rechteckigen Hindernisse in Verdrahtungsregionen gemäß der Breite der Leiterbahnen, der Durchkontaktierung, und des Isolierabstands erzeugt. Die Hindernisse, die rechteckig, teilweise überlappt und auf verschiedenen Ebenen in den Verdrahtungsregionen verteilt sind, können als eine Menge  $R = \{r_1, r_2, \dots, r_N\}$  definiert werden. Gemäß den Entwurfsregeln wird um jedes Hindernis eine Sperrzone eingestellt, in der die zentralen Linien der Leitungen und die Zentren der Durchkontaktierungen nicht verlegt werden dürfen. Sei  $\tilde{R}$  eine Menge der rechteckigen Regionen, die aus den Hindernissen in  $R$  nach der Leiterbahnbreite und dem Isolierabstand expandiert sind; und sei  $\tilde{R}^v$  eine andere Menge der rechteckigen Regionen, die nach der Durchkontaktierungsbreite und dem Isolierabstand expandiert sind. So kann die Definition eines solchen Verbindungsgraphen  $G_s$  wie folgt dargestellt werden [15]:

Gegeben sei ein Mehrebenen-Routingproblem mit einer Menge der Hindernisse  $R$ , einem Startpunkt  $s$  und einem Endpunkt  $t$ . Der so genannte „*Non-Uniform Grid Graph*“  $G_s$  ist ein orthogonaler Gittergraph, dessen  $x$ -Gitterkoordinaten nicht nur den vertikalen Grenzen der  $\tilde{R}$  und  $\tilde{R}^v$ , sondern auch den  $x$ -Koordinaten des Start- und Endpunkts entsprechen. Analog können die  $y$ -Gitterkoordinaten des Verbindungsgraphen definiert werden (siehe Bild 32).

$G_s$  ist ein *starker* Verbindungsgraph, d. h. es ist garantiert, den kürzesten Weg zu finden, wenn dieser Verbindungsweg zwischen den Hindernissen  $R$  und nach Entwurfsregeln existiert. Im Vergleich zum einheitlichen Raster (siehe Bild 32(d)) hat der Verbin-

ungsgraph  $G_s$  deutlich weniger Knoten. Obwohl  $G_s$  mehrere Knoten als der andere Verbindungsgraph  $G_c$  hat (siehe Bild 32(c)), ermöglicht es die Gitter-Eigenschaft von  $G_s$ , eine implizite Beschreibung anzuwenden. Da die Verbindungsgraphen für jedes Netz unterschiedlich sind, soll der Verbindungsgraph beim Routingverfahren in Echtzeit erzeugt und daher möglichst implizit repräsentiert werden.

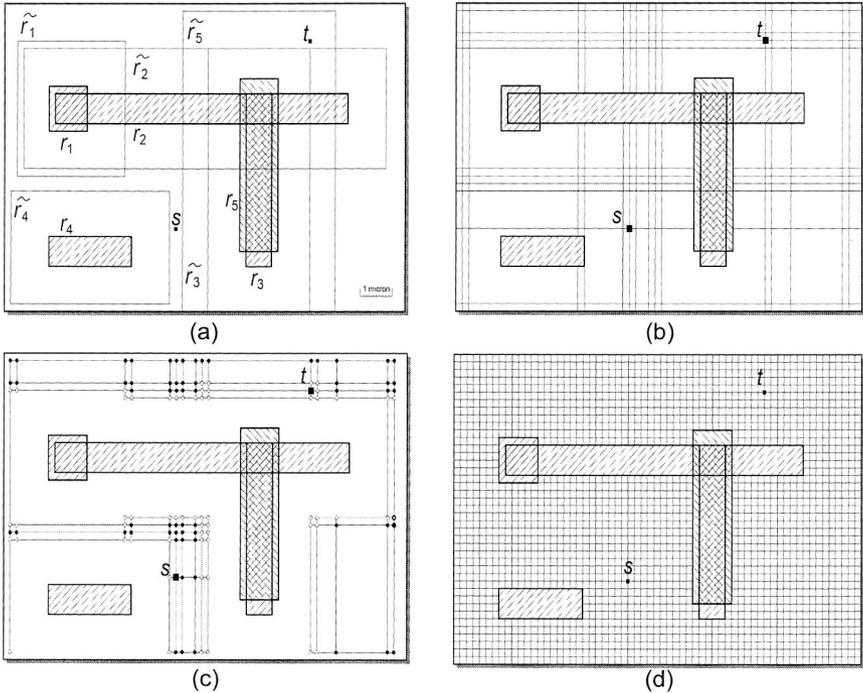


Bild 32: Erzeugung des Verbindungsgraphen: (a) Rechteckige Hindernisse  $R$  und expandierte  $\tilde{R}$ , (b) Erzeugung von  $G_s$  durch  $x$ - und  $y$ -Koordinaten von  $\tilde{R}$ ,  $\tilde{R}^v$ ,  $s$  und  $t$ , (c) Erzeugung von  $G_c$  durch Grenzen und ausgebreitete Linien von  $\tilde{R}$ ,  $\tilde{R}^v$ , (d) einheitliches Raster, das sehr dünne und dichte Herstellungsgitter benutzt. [15]

Die Anwendung bzw. Erweiterung eines solchen Verbindungsgraphen auf das 3D-MID-Routing kann durch die zur Erweiterung des Hightower-Algorithmus angewandte Methode realisiert werden. Die Hindernisse  $R$  und die entsprechenden Sperrzonen  $\tilde{R}$  und  $\tilde{R}^v$  auf dreidimensionaler Fläche werden durch zwei Parameter  $u$  und  $v$  statt der  $x$ - und  $y$ -Koordinaten dargestellt. Der Verbindungsgraph im 3D-MID-Routing ist ein in  $uv$ -Richtung orthogonaler, aber nicht-uniformer Gittergraph. Auf diese Weise kann der Verbindungsgraph auch auf einer Zylinderfläche erzeugt werden. Wegen des Winkelmaßes

von  $u$  muss eine zusätzliche Behandlung für die Erzeugung der in  $v$ -Richtung stehenden Gitterlinien mit dem Radius des Zylinders berücksichtigt werden.

#### *Implizite Repräsentation der Verbindungsgraphen $G_s$*

Für die Repräsentation der Hindernisse  $R$  können auch die im modifizierten Hightower-Algorithmus angewandten Datenstrukturen (siehe Kapitel 4.3.2) benutzt werden. Für die Hindernisse  $r_i$  auf jeder Verdrahtungsfläche wird ein entsprechender 4-dimensionaler binärer Baum aufgebaut. Wegen der gemeinsamen Benutzung der Information von Hindernissen kann der modifizierte DUNE-Algorithmus effektiv in das linienbasierte 3D-Routingverfahren integriert werden. Im Vergleich zur expliziten Repräsentation von  $R$  werden die Sperrzonen  $\tilde{R}$  und  $\tilde{R}^v$  nur bei der Erzeugung des Verbindungsgraphen  $G_s$  vorläufig kalkuliert.

Anstatt die Knoten und Kanten von Graphen explizit zu speichern, wird der Verbindungsgraph  $G_s$  im Algorithmus implizit repräsentiert. Zwei Datenfelder  $U_s$  und  $V_s$  werden jeweils für die Speicherung der Parameter  $u$  bzw.  $v$  der Gitterlinien im Verbindungsgraphen  $G_s$  verwendet. Wegen der Anwendung der impliziten Repräsentation müssen die Knoten und die Kanten bei der Graphsuche dynamisch (*on-the-fly*) berechnet und auf ihre Gültigkeit geprüft werden. Die Abfrage der Knoten in Graphen  $G_s$  wird in zwei Schritte unterteilt: Die Suche nach Nachbarknoten und anschließend die Überprüfung der Gültigkeit der Nachbarknoten.

Die Suche nach Nachbarknoten kann mit den erzeugten Datenfeldern  $U_s$  und  $V_s$  leicht realisiert werden. Wenn  $U_s[i]$  dem Parameter  $u$  des aktuellen Knotens entspricht, ist der Parameter  $u$  des Nachbarknotens entweder  $U_s[i+1]$  oder  $U_s[i-1]$ . Gleiches gilt für die Suche des Parameters  $v$ .

Damit ein Nachbarknoten für das Routing gültig ist, müssen zwei Voraussetzungen erfüllt sein: Erstens darf der Nachbarknoten für das Verlegen der Leiterbahn nicht in der Sperrzone  $\tilde{R}$  und für das Verlegen der Durchkontaktierung nicht in  $\tilde{R}^v$  liegen. Zweitens darf der Weg zwischen dem aktuellen Knoten und dem Nachbarknoten nicht über ein Hindernis führen.

Da nur die Hindernisse  $R$  explizit beschrieben sind, aber die Beschreibung der entsprechenden Sperrzonen  $\tilde{R}$  und  $\tilde{R}^v$  im Algorithmus eigentlich nicht vorhanden ist, breitet sich der Nachbarknoten nach Entwurfsregeln zu einer eigenen Sperrzone  $\tilde{r}_i$  oder  $\tilde{r}_i^v$  aus. Somit kann die Überprüfung der ersten Voraussetzung in eine Überprüfung auf Überschneidung der Sperrzone  $\tilde{r}_i$  oder  $\tilde{r}_i^v$  mit den Hindernissen  $R$  konvertiert werden. Auf Basis des für Hindernisse erzeugten 4-dimensional binären Baums kann die Abfrage der Hindernisse bei der Überprüfung mit  $\tilde{r}_i$ ,  $\tilde{r}_i^v$  oder dem Weg zwischen aktuellem Knoten und Nachbarknoten effektiv durchgeführt werden, um so einen zulässigen Nachbarknoten schnell zu finden.

Suche nach dem kürzesten Pfad im Verbindungsgraph  $G_s$ 

Auf Basis des Verbindungsgraphen  $G_s$  wird ein modifizierter linienbasierter Minimum-Detour-Algorithmus [118] angewandt, um den kürzesten Verbindungsweg zwischen Start- und Endpunkt in  $G_s$  zu finden.

Auf den Verbindungsgraphen  $G_s$  wird die Detour-Nummer wie folgt definiert. Der Kante vom Knoten  $a$  bis Knoten  $b$  in  $G_s$  wird eine Richtung zugewiesen.  $L$  ist eine Linie, die durch den Endpunkt  $t$  verläuft und rechtwinklig auf der Kante  $a \rightarrow b$  steht.  $dl(a \rightarrow b)$  repräsentiert die Detour-Nummer (hier Detour-Länge genannt) der gerichteten Kante  $a \rightarrow b$  in  $G_s$  in Bezug auf den Endknoten  $t$ . Die Detour-Länge berechnet sich wie folgt:

$$dl(a \rightarrow b) = \begin{cases} 0, & \text{wenn } a \text{ und } b \text{ sich auf der gleichen Seite von } L \text{ befinden, und gleichzeitig } a \text{ ferner von } L \text{ ist als } b \text{ (Bild 33a);} \\ l(a \rightarrow b), & \text{die Länge der Kante, wenn } a \text{ und } b \text{ sich auf der gleichen Seite von } L \text{ befinden, und gleichzeitig } a \text{ näher an } L \text{ liegt als } b \text{ (Bild 33b);} \\ l(c \rightarrow b), & \text{der Abstand von } c \text{ bis } b, \text{ wenn } L \text{ mit der Kante } a \rightarrow b \text{ schneidet, und } c \text{ der Schnittpunkt ist (Bild 33c).} \end{cases}$$

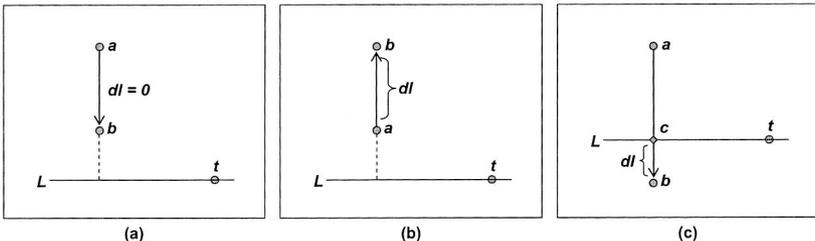


Bild 33: Definition der Detour-Länge in Verbindungsgraphen  $G_s$

Die Detour-Länge  $\delta(w)$  eines Knotens  $w$  ist in Bezug auf den Startknoten  $s$  und den Endknoten  $t$  gleich der Summe der Detour-Längen aller gerichteten Kanten auf einem Pfad von  $s$  bis  $w$  in  $G_s$ . Daraus folgt, dass die Länge eines gerichteten Verdrahtungswegs  $p(s \rightarrow t)$  von  $s$  bis  $t$  im Verbindungsgraph  $G_s$  mit der folgenden Formel dargestellt werden kann:

$$l(p(s \rightarrow t)) = M(s, t) + 2 * \delta(t) \quad (10)$$

wobei  $M(s, t)$  der Manhattan-Abstand zwischen  $s$  und  $t$  ist. Da der Wert von  $M(s, t)$  konstant ist, wird der kürzeste Weg zwischen  $s$  und  $t$  durch die Minimierung der  $\delta(t)$  aufgefunden. Bei der Suche auf den Graphen  $G_s$  werden die gültigen Nachbarknoten mit der Detour-Länge markiert, anstatt die aktuelle Verbindungslänge aus  $s$  zu benutzen. Außerdem haben die Knoten mit niedrigster Detour-Länge eine höhere Priorität und werden vorzugsweise ausgewählt und expandiert.

### **Erweiterung für das 3D-MID-Routing innerhalb des Körpers**

Das grundlegende 3D-MID-Routingproblem mit Durchkontaktierung kann wie im Folgenden definiert werden: Gegeben seien zwei gegenseitige Flächen auf dem Schaltungsträger, der Startpunkt  $s$  sei auf der einen Fläche, der Endpunkt  $t$  auf der anderen Fläche. Einige Hindernisse befinden sich auf beiden Flächen. Gemäß den Entwurfsregeln soll der kürzeste Verbindungsweg von  $s$  bis  $t$  über eine oder mehrere Durchkontaktierungen zwischen beiden Flächen gefunden werden.

Da die Durchkontaktierung in MIDCAD als MID-spezifisches Feature betrachtet wird, kann diese an einer bestimmten Stelle auf dem Schaltungsträger manuell erzeugt werden. Damit wird eine Verbindung durch zwei Teilverbindungen ersetzt, die jeweils auf einer eigenen Fläche verlaufen. Das Routing der Teilverbindung kann durch die oben dargestellten 3D-MID-Routingalgorithmen realisiert werden. Eine solche manuelle Lösung ist zwar flexibel, aber deutlich aufwendig. Da der Verbindungsgraph  $G_s$  im DUNE-Algorithmus die Informationen über Hindernisse auf mehreren Ebenen enthalten kann, bietet  $G_s$  eine Möglichkeit, die Anzahl und Lage der Durchkontaktierungen einer Verbindung automatisch zu finden. Im Folgenden wird die Erweiterung des DUNE-Algorithmus vorgestellt, um ein automatisches 3D-Routing mit Durchkontaktierung zu realisieren.

#### *Konzeption der Lösung für das oben dargestellte grundlegende Routingproblem*

Die Erzeugung des Verbindungsgraphen  $G_s$  für das Routing innerhalb einer Fläche braucht lediglich die Informationen über  $\tilde{R}$ ,  $s$  und  $t$  zu nutzen. Wenn jedoch das Routing mit Durchkontaktierung innerhalb des Körpers durchgeführt wird, müssen bei der Erzeugung der  $G_s$  die Sperrzone  $\tilde{R}^v$  zusammen mit  $\tilde{R}$ ,  $s$  und  $t$  berücksichtigt werden. Der erzeugte Verbindungsgraph  $G_s$  soll in der Lage sein, das Routingproblem mit Durchkontaktierung zu lösen.

Da das 3D-Routing mit Durchkontaktierung einige besondere Eigenschaften aufweist, müssen besondere Maßnahmen für die Erzeugung des geeigneten Verbindungsgraphen  $G_s$  und die Anwendung des modifizierten DUNE-Algorithmus getroffen werden. Damit kann das grundlegende 3D-Routingproblem mit Durchkontaktierung als Zweiebenen-Routingproblem behandelt werden.

Eine der zwei gegenseitigen Flächen, die mittels Durchkontaktierung verbunden werden kann, wird Hauptfläche genannt. Die andere heißt Nebenfläche. Die Hauptfläche sollte hierbei eine Ebene sein. Im Algorithmus wird die Hauptfläche als erste Verdrahtungsebene beobachtet und deshalb die Hindernisse auf der Hauptfläche als die auf der ersten Ebene der Hauptfläche behandelt. Da die Nebenfläche nicht parallel, nicht gleich groß in Bezug auf die Hauptfläche und eventuell auch keine Ebene sein muss, wird zur Behandlung der Nebenfläche eine spezielle virtuelle Verdrahtungsebene erzeugt, die deckungsgleich mit der ersten Ebene ist. Diese virtuelle Ebene wird im Algorithmus als die zweite Ebene der Hauptfläche betrachtet. Indem die Hindernisse auf der Nebenfläche auf die zweite Verdrahtungsebene projiziert werden, wird der Verbindungsgraph  $G_s$ ,

wie im ursprünglichen DUNE-Algorithmus, auf Basis der auf zwei Ebenen liegenden Hindernisse gemäß den Entwurfsregeln erzeugt (siehe Bild 34).

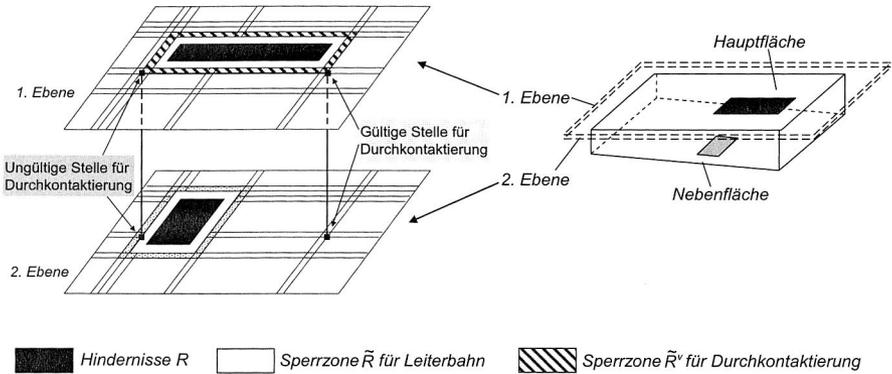


Bild 34: Erzeugung des Verbindungsgraphen  $G_s$  für zwei Verdrahtungsebenen [13]

#### Gültigkeit der Knoten in Mehrebenen-Verbindungsgraphen

Wenn das Routing nur innerhalb einer Fläche durchgeführt wird, ist für die erste Voraussetzung nur die Überprüfung, ob der Knoten in der Sperrzone  $\tilde{R}$  liegt, nötig. Beim Routing mit Durchkontaktierung sind nicht nur die gültigen Knoten für das Verlegen der Leiterbahn, sondern auch die möglichen Knoten für das Verlegen der Durchkontaktierung unter Berücksichtigung der Sperrzone  $\tilde{R}^v$  zu finden (siehe Bild 34). Für die zweite Voraussetzung sind bei der Überprüfung nur die auf derselben Ebene liegenden Hindernisse zu berücksichtigen. Eine zusätzliche Überprüfung dient dazu, herauszufinden, ob der Knoten auf der ersten Ebene im Bereich der Hauptfläche oder auf der zweiten Ebene im Bereich der Nebenfläche liegt.

#### Erweiterung des Suchalgorithmus für den Mehrebenen-Verbindungsgraph

Der oben dargestellte, linienbasierte Minimum-Detour-Algorithmus wird für das Routing auf mehreren Ebenen erweitert, indem die Detour-Länge die Kosten für Ebenenwechsel in der Definition berücksichtigt wird. Für das Routing auf mehreren Ebenen wird der Knoten im Verbindungsgraphen  $G_s$  in mehrere Knoten unterteilt. So gibt es z. B. für das quasi Zweiebenen-MID-Routing zwei Knoten an derselben Stelle in  $G_s$ , die jeweils getrennt auf einer eigenen Ebene liegen. Das bedeutet auch, dass der Knoten nicht nur Nachbarknoten in den  $uv$ -Richtungen, sondern möglicherweise Nachbarknoten auf anderen Ebenen hat.

Wenn eine Durchkontaktierung an einen gültigen Knoten  $w$  verlegt wird, wird eine gerichtete Kante  $w_1 \rightarrow w_2$  oder  $w_2 \rightarrow w_1$  in  $G_s$  erzeugt. Die zusätzlichen Situationen zur Berechnung der Detour-Länge für Zweiebenen-Verbindungsgraphen sind wie folgt gegeben:

$dl(w1 \rightarrow w2) = \begin{cases} 0 & \text{wenn der Endknoten auf der zweiten Ebene ist, d. h.} \\ & \text{die Suche entlang der gerichteten Kante führt in Ziel-} \\ & \text{richtung (Bild 35a);} \\ N & \text{wenn der Endknoten auf der ersten Ebene ist, wobei } N \\ & \text{eine vordefinierte Konstante ist, die den Kosten für den} \\ & \text{Wechsel von einer Ebene zur Nachbarebene entspricht} \\ & \text{(Bild 35b).} \end{cases}$

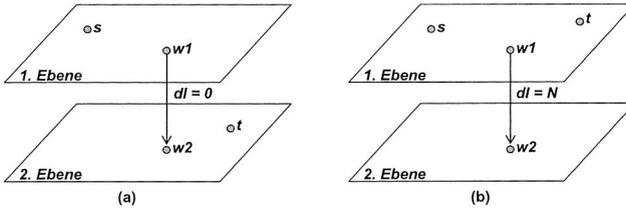


Bild 35: Definition der Detour-Länge beim Ebenenwechseln

Mit dieser Definition ist die Formel (10) für das Zweiebenen-Routing gültig, indem die Kosten  $N$  der Länge der Durchkontaktierung entspricht. Deshalb kann der linienbasierte Minimum-Detour-Algorithmus einwandfrei in Zweiebenen-Verbindungsgraphen eingesetzt werden. Die oben dargestellte Definition kann auch auf die Situation für mehrere Ebenen erweitert werden.

### Vorgehensweise beim rasterfreien 3D-MID-Routing

Der oben dargestellte, auf dem DUNE-Algorithmus basierende 3D-Routingalgorithmus wird im zweiten Schritt des linienbasierten 3D-Routingverfahrens integriert, d. h. der Algorithmus arbeitet mit dem modifizierten Hightower-Algorithmus auf den im ersten Schritt gefundenen Verbindungsflächen zusammen, um eine exakte Geometrie des Verbindungswegs festzulegen. Durch diese Integration wird ein rasterfreies 3D-MID-Routing auf bzw. innerhalb des Schaltungsträgers realisiert. Die Hauptschritte von des rasterfreien 3D-MID-Routings entsprechen den Schritten im linienbasierten Verfahren. Einige notwendige Ergänzungen werden im Folgenden vorgestellt.

Wenn nötig, muss vor der Anwendung des Routings mit Durchkontaktierung ein zusätzlicher Schritt eingefügt werden, in dem die Hauptflächen und die entsprechenden Nebenflächen auf dem Schaltungsträger bestimmt werden. Beim detaillierten Routing werden die modifizierten Hightower- und DUNE-Algorithmen sowie eine gemischte Strategie dann auf diesen Ebenen eingesetzt (siehe Bild 36).

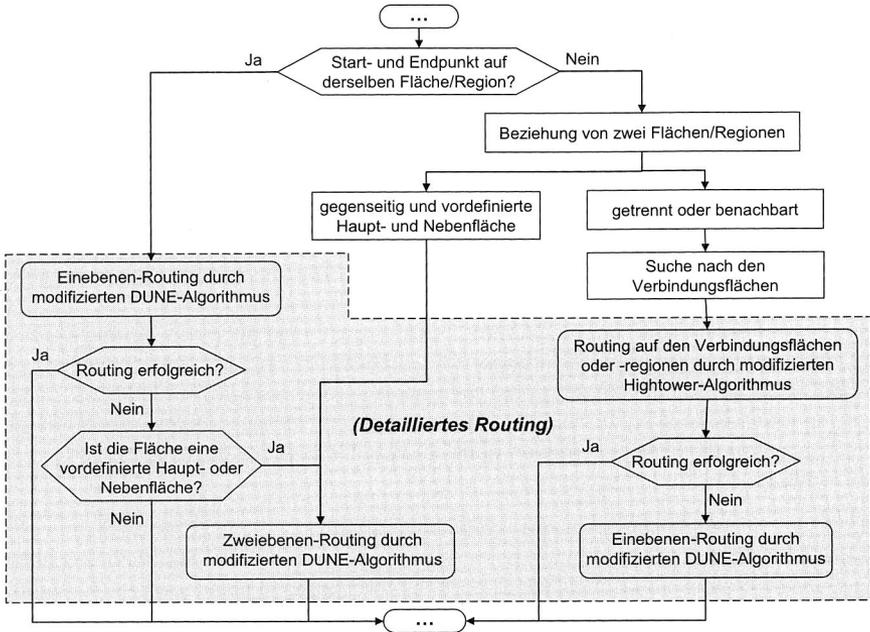


Bild 36: Rasterfreie 3D-MID-Routing durch Integration modifizierter Hightower- und DUNE-Algorithmen

**Experimentelle Resultate**

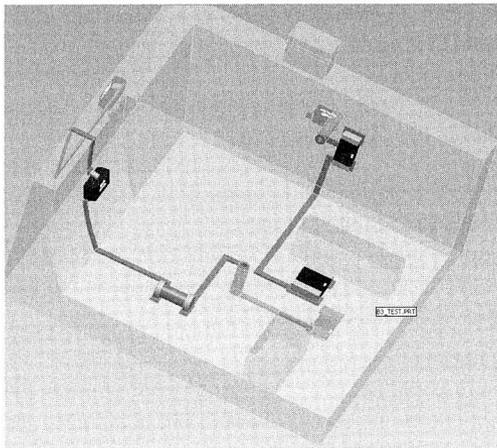


Bild 37: Routingbeispiel durch modifizierten DUNE-Algorithmus

In Bild 37 wird ein Routingbeispiel gezeigt, in dem zwei Verbindungen zum grundlegenden 3D-MID-Routingproblem mit Durchkontaktierung gehören. Durch den modifizierten DUNE-Algorithmus sind die beiden Verbindungen als Zweiebenen-Routing jeweils mit einer Durchkontaktierung verlegt. Die anderen Verbindungen werden zusammen im oben dargestellten rasterfreien 3D-MID-Routingverfahren realisiert.

### Vorteile und Nachteile

Im Vergleich zu den rasterbasierten Algorithmen bleibt beim modifizierten DUNE-Algorithmus der Hauptvorteil der rasterunabhängigen Algorithmen erhalten, d. h. durch den Verbindungsgraph  $G_s$  kann das Routing mit geringerem Suchaufwand realisiert werden. Außerdem kann der modifizierte DUNE-Algorithmus auch das 3D-MID-Routing bei Verbindungen mit unterschiedlicher Leiterbahnbreite und auf Zylinderflächen behandeln. Der Verbindungsgraph  $G_s$  ist ein *starker* Verbindungsgraph, d. h. es ist garantiert, den kürzesten Weg zu finden. Insgesamt hat das 3D-MID-Routing durch das gemischte rasterfreie Routingverfahren deutlich mehr Vorteile als das linienbasierte Routingverfahren.

Die Anwendungsmöglichkeiten des Verbindungsgraphen  $G_s$  im 3D-MID-Routing sind jedoch begrenzt. Da die uneinheitlichen Gitter in  $G_s$  immer in  $uv$ -Richtung eingestellt sind, sollten die meisten Bauelemente auch in  $uv$ -Richtung platziert werden. Außerdem ist die Erzeugung des Verbindungsgraphen  $G_s$  auf anderen Flächentypen als Ebene und Zylinder kaum realisierbar.

#### 4.3.4 Topologischer 3D-Routingalgorithmus

Die oben vorgestellten 3D-Autorouting-Algorithmen eignen sich zwar für die meisten Routingprobleme im Zusammenhang mit MID-Produkten, indem sie die Leiterbahnen und Durchkontaktierungen automatisch auf und innerhalb des dreidimensionalen Grundkörpers verlegen. Dennoch sind diese Routingalgorithmen bei der Bestimmung möglicher Routing-Pfade auf Pfade in flächengehörigen  $uv$ -Richtungen beschränkt. So wird beispielsweise die diagonale Richtung durch die oben vorgestellten Algorithmen noch nicht unterstützt. In einigen Situationen - insbesondere bei Anwendung des 2K-Verfahrens - müssen die Leiterbahnen in unregelmäßigen Richtungen und auf irregulär geformten Oberflächen des räumlichen Schaltungsträgers verlegt werden. Da sich dieses Herstellungsverfahren zunehmend durchsetzt, wird es immer schwieriger, das Routing mittels oben vorgestellter Routingalgorithmen durchzuführen. Es besteht also ein Bedarf nach einer neuen Routingmethode, bei der die Beschränkungen der oben vorgestellten Routingalgorithmen beseitigt werden.

Der topologische Ansatz beim Autorouting ist eine Möglichkeit, die keinen geometrischen Einschränkungen unterliegt. Diese Methode wurde für das 2D-Routing entwickelt und in der Literatur, wie z. B. in [19], [20], [44], [47], ausführlich diskutiert. Sie besteht hauptsächlich aus vier Aufgaben: Constrained Delaunay-Triangulation, Rubber-Band

Sketch, Überprüfung der Routbarkeit sowie geometrische Transformation. Im Folgenden werden die oben genannten vier Aufgaben kurz beschrieben.

### Kurze Beschreibung des topologischen Routings

#### *Constrained Delaunay-Triangulation (CDT):*

Eine Triangulation stellt eine Dreiecksvermaschung von Punkten dar. Sie ist eine besondere Art der Beschreibung einer Oberfläche. Die Delaunay-Triangulation ist ebenfalls eine Dreiecksvermaschung, bei der zusätzlich die gebildeten Dreiecke das Delaunay-Kriterium erfüllen. Dieses besagt, dass der Umkreis jedes Dreiecks keinen weiteren Punkt der Ausgangsmenge enthalten darf. Werden vorgegebene Kanten in einer Triangulation abgebildet, so ist eine echte Delaunay-Triangulation im allgemeinen Fall nicht mehr sicherzustellen. In der Umgebung der Zwangskante kann es lokal zu Abweichungen von dem Delaunay-Kriterium kommen. Man spricht von einer Constrained Delaunay-Triangulation (CDT). Ein Beispiel ist in Bild 38 zu sehen. Die Anwendung der CDT ist sehr weitläufig, insbesondere für die kürzeste Wegsuche [49]. Für das topologische Routing werden die Hindernisse, Rubber-Band-Segmente usw. als Zwangskanten behandelt.

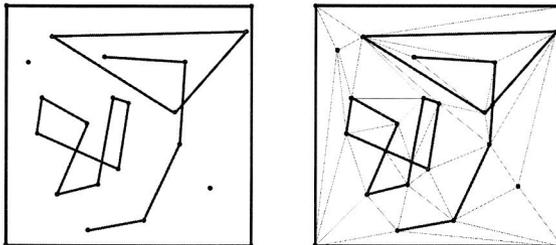


Bild 38: Erzeugung der CDT mit Zwangskanten [50]

#### *Rubber-Band Sketch (RBS):*

RBS repräsentiert eine Verbindung durch ein flexibles Gummiband, die durch das Dehnen und Krümmen die kürzeste Netztopologie darstellt (siehe Bild 39a). Die Suche nach einem „*Rubber-Band Sketch*“ kann auf Basis der oben dargestellten CDT durch die existierenden Suchalgorithmen [49] für den kürzesten Pfad realisiert werden. Bei der Suche wird die CDT nach den gefundenen RBS dynamisch ändert.

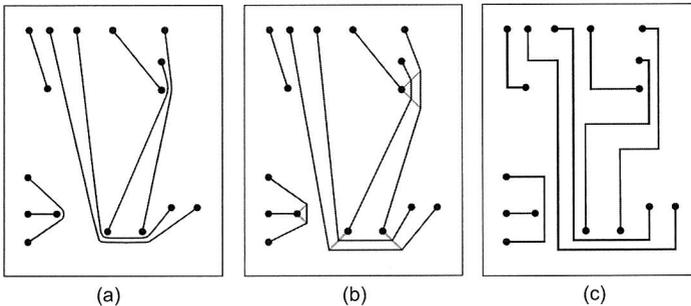
#### *Überprüfung der Routbarkeit:*

Zwar ist die Suche nach der topologischen Verbindung durch CDT schneller und mit geringerem Suchaufwand verbunden als die direkte Suche nach der konkreten geometrischen Verbindung, vor der geometrischen Transformation ist jedoch die Routbarkeit der Netztopologie nach Entwurfsregeln zu überprüfen. Die Überprüfung der Routbarkeit

kann wie im SURF-Autorouter [20] mit der Suche nach RBS getrennt durchgeführt oder durch die Methode von Hama und Etoh [44] in die Suche nach RBS integriert werden.

#### *Geometrische Transformation:*

Die geometrische Transformation entspricht dem detaillierten Routing, das nach der gefundenen RBS die konkrete Geometrie (rektilinear oder oktilinear) der Verbindung mit folgenden Eigenschaften festlegt: topologische Entsprechung mit RBS, Einhaltung der geometrischen Randbedingungen (z. B. Leiterbahnbreite und Isolierabstand) und kürzester Verbindungsweg. Bild 39 zeigt ein Verfahren der geometrischen Transformation vom RBS bis zur konkreten Netzgeometrie.



*Bild 39: Routingbeispiel mit dem topologischen Routing. (a) Erzeugung von RBS auf Basis von CDT, (b) die ausgebreitete RBS nach der Einführung der Speicher, (c) Rektilinearere Verbindungsweg [20]*

### **Erweiterung der topologischen Methode auf das 3D-MID-Routing**

Inwiefern das topologische Autorouting entsprechend der Charakteristik von 3D-MID-Produkten auf den dreidimensionalen Raum erweitert werden kann, wird im Rahmen dieser Forschungsarbeit überprüft.

Da die CDT eine wichtige Methode zur Vermaschung der Oberfläche bei FEM ist, wurden bereits viele Algorithmen zur Erzeugung der CDT erforscht und entwickelt [11], [50]. Zur Erzeugung der CDT auf den Oberflächen von 3D-Schaltungsträgern können die existierenden Algorithmen direkt eingesetzt werden. Für das 3D-MID-Routing sollen bei der Erzeugung der CDT die relevanten Hindernisse inkl. vordefinierter Leiterbahnen als Zwangskanten und andere Randbedingungen (z. B. *Keepout Surfaces* und *Regions*) berücksichtigt werden. Die dynamische Erzeugung einer richtigen CDT auf den Oberflächen des 3D-Schaltungsträgers ist eine grundlegende, wichtige Aufgabe für die Erweiterung der topologischen Methode auf das 3D-MID-Routing.

Die Positionen des Start- und Endpunkts können zwar nicht auf derselben Fläche liegen, aber bei der Suche nach entsprechenden RBS auf Basis der erzeugten CDT gibt es keinen Unterschied zu zwei Punkten auf unterschiedlichen Flächen. Der in [49] darge-

stellte und auf der Suche nach dem kürzesten Weg angewandte Suchalgorithmus kann ohne Änderung direkt in die Suche nach RBS eingesetzt werden. Der Unterschied zwischen dem RBS-Problem und dem Suchproblem für den kürzesten Weg ist, dass es beim RBS-Problem mehrere Verbindungen gibt. Das bedeutet, dass die entsprechenden RBS voneinander beeinflusst werden. Das erfolgreiche Finden aller nicht miteinander gekreuzten RBS für das topologische Routing ist deutlich schwieriger. Das Suchverfahren soll in der Lage sein, die Netzreihenfolge automatisch umzustellen. Gemäß dem Ergebnis der RBS ist die geometrische Transformation leichter als das direkte detaillierte Routing. Die Transformation rational und passend auf den Flächen des Schaltungsträgers durchzuführen, ist jedoch eine aufwendige Aufgabe.

Der Hauptvorteil liegt darin, dass die gefundenen RBS die Geometrie (rektilinear, oktilinear usw.) der Verbindung nach Bedarf erzeugen kann. Mit dem topologischen Routing ist es auch garantiert, den kürzesten Weg zu finden, wenn der Verbindungsweg zwischen den Hindernissen und nach Entwurfsregeln existiert. Der Nachteil ist, dass das 3D-MID-Routing mit Durchkontaktierung innerhalb des Schaltungsträgers durch die topologische Methode schwer realisierbar ist.

#### 4.3.5 Vergleich und Zusammenfassung der 3D-MID-Routingalgorithmen

Im Folgenden werden in Tabelle 2 die vorgestellten 3D-MID-Routingalgorithmen für Zweipunktnetze in den vier Aspekten Prinzip, Vorteile, Nachteile und Anwendungsgebiete miteinander verglichen und zusammengefasst.

Tabelle 2: Vergleich der 3D-MID-Routingalgorithmen für Zweipunktnetze

	Rasterbasierte Routingmethode	Linienbasierte Routingmethode	Rasterfreie Routingmethode	Topologische Routingmethode
<b>Prinzip</b>	<ul style="list-style-type: none"> <li>• Rasterung der Oberflächen</li> <li>• Erweiterung des Hadlock-Algorithmus</li> </ul>	<ul style="list-style-type: none"> <li>• Versuchslinien statt Rasterung</li> <li>• Suche nach den Verbindungsflächen durch den A*-Algorithmus</li> <li>• Erweiterung des Hightower-Algorithmus</li> </ul>	<ul style="list-style-type: none"> <li>• Verbindungsgraph auf den Oberflächen</li> <li>• Erweiterung des DUNE-Algorithmus</li> </ul>	<ul style="list-style-type: none"> <li>• Constrained Delaunay-Triangulation (CDT) der Oberflächen</li> <li>• Geometrische Transformation der Netztopologie</li> </ul>
<b>Vorteile</b>	<ul style="list-style-type: none"> <li>• Der kürzeste Pfad wird gefunden, wenn dieser existiert (A)</li> </ul>	<ul style="list-style-type: none"> <li>• geringer Suchaufwand (B)</li> <li>• Routing auf andere Flächen, z. B. Zylinder (C)</li> <li>• Routen für Leiterbahnen mit unterschiedlicher Breite (D)</li> </ul>	<ul style="list-style-type: none"> <li>• (A), (B), (C), (D)</li> <li>• Routen innerhalb des Körpers mit Durchkontaktierung (E)</li> </ul>	<ul style="list-style-type: none"> <li>• (A), (C), (D)</li> <li>• Routen in verschiedenen Richtung (F)</li> </ul>
<b>Nachteile</b>	<ul style="list-style-type: none"> <li>• Hoher Zeit- und Speicherplatzbedarf</li> <li>• Beschränkung auf rechteckige Oberflächen</li> </ul>	<ul style="list-style-type: none"> <li>• kürzester Pfad nicht immer ermittelbar</li> <li>• nicht sichergestellt, dass eine Lösung gefunden wird, obwohl diese existiert</li> </ul>	<ul style="list-style-type: none"> <li>• Bauelemente sollten in uv-Richtungen platziert werden</li> </ul>	<ul style="list-style-type: none"> <li>• Erzeugung der CDT für Schaltungsträger</li> <li>• Geometrische Transformation</li> </ul>
<b>Anwendungsgebiete</b>	<ul style="list-style-type: none"> <li>• sehr begrenzt, nur auf rechteckiger 3D-Schaltungsträger</li> </ul>	<ul style="list-style-type: none"> <li>• Anwendbarkeit auf verschiedene Flächen</li> </ul>	<ul style="list-style-type: none"> <li>• Routen auf Ebene, Zylinder und durch Durchkontaktierung</li> </ul>	<ul style="list-style-type: none"> <li>• Routen (rektilinear, oktilinear, usw.)</li> <li>• Unregelmäßiges Layout und komplexe Flächen</li> </ul>

## 4.4 3D-Routing für Mehrpunktnetze

Bei allen oben vorgestellten 3D-MID-Routingalgorithmen werden nur Verbindungen zwischen zwei Anschlusspunkten, dem Startpunkt  $s$  und dem Endpunkt  $t$ , verlegt. Die meisten Netze sind jedoch Mehrpunktnetze mit mehr als zwei Anschlusspunkten. Das Routing von Mehrpunktnetzen ist deutlich komplexer als das von Zweipunktnetzen. Um die oben vorgestellten Algorithmen weiter benutzen zu können, ist es erforderlich, das Mehrpunktnetz in Zweipunktnetzsegmente aufzuteilen. Zur Lösung dieses Problems sind minimale Steinerbäume geeignet, welche alle Anschlusspunkte eines Mehrpunktnetzes unter Einschluss von Zusatzpunkten, sog. Steinerpunkten, derart verbinden, dass die einzelnen Netzsegmente in Bezug auf die gesamte Netzlänge optimiert sind. Deswegen entspricht die Suche nach dem Weg mit minimaler Verbindungslänge für Mehrpunktnetze einer Steinerbaum-Generierung, die jedoch ein NP-hartes Problem darstellt und in der Regel durch die Erzeugung eines minimalen Spannbauams suboptimal realisiert wird.

### 4.4.1 Minimaler Spannbaum (MST) und minimaler Steinerbaum (SMT)

#### Minimaler Spannbaum (MST):

Der Begriff MST kommt aus der Graphentheorie. Sei  $G = (V, E)$  mit  $V =$  Knotenmenge und  $E =$  Menge ungerichteter Kanten ein zusammenhängender ungerichteter Graph. Ein ebenfalls zusammenhängender Teilgraph  $T$  von  $G$ , der dieselbe Knotenmenge  $V$  besitzt und ein Baum ist, heißt *Spannbaum* von  $G$ . Im kantengewichteten Graphen  $G$  wird die Summe seiner Kantengewichte als Gewicht  $W(G)$  des Graphen definiert. Gleiches gilt für das Gewicht  $W(T)$ . Einen Spannbaum  $T$  von  $G$  mit minimalem Gewicht bezeichnet man als *minimalen Spannbaum (MST)* von  $G$ , d. h. es gibt keinen anderen Spanbaum in demselben Graphen mit geringerem Gewicht. In Bild 40 wird ein Graph mit minimalem Spannbaum gezeigt.

In der Praxis können viele Probleme als Spannbaumproblem behandelt werden, wie z. B. die kürzeste Verdrahtung von Mehrpunktnetzen. Für das rektile Routing des Mehrpunktnetzes wird häufig der minimale rektile Spannbaum (siehe Bild 40c) benutzt, bei dem alle Knoten bzw. Anschlusspunkte nach minimaler Gesamtverbindungslänge rektilinear miteinander verbunden sind, ohne dass Steinerpunkte vorliegen.

Die wichtigen Algorithmen zum Finden eines MST sind der Kruskal-Algorithmus [58] und der Prim-Algorithmus [92]. Beide MST-Algorithmen werden auch für das 3D-MID-Routing von Mehrpunktnetzen eingesetzt. Der Prim-Algorithmus wird in das Autoroutingverfahren integriert, während der Kruskal-Algorithmus zur dynamischen Darstellung der logischen Verbindungen für Mehrpunkte vor dem Routing und im Ablauf des manuellen Routings in MIDCAD eingesetzt wird.

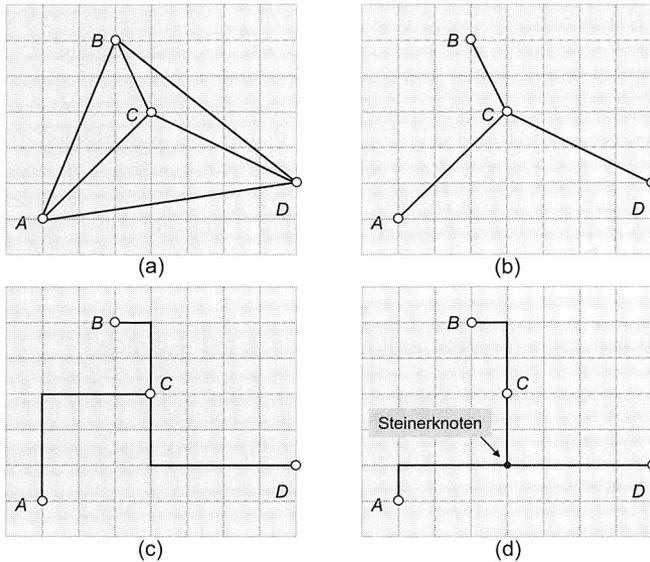


Bild 40: *Beispiel für MST und SMT. (a) Vollständiger Graph G, (b) Minimaler Spannbaum, (c) Minimaler rektilinear Spannbaum, (d) Minimaler rektilinear Steinerbaum*

### Minimaler Steinerbaum (SMT):

Das Steinerbaumproblem ist eine Verallgemeinerung des Spannbaum-Problems. Aus der Sicht der Graphentheorie ist der Steinerbaum wie folgt definiert: Sei  $G = (V, E)$  ein ungerichteter Graph ohne Mehrfachkanten, wobei jede Kante aus  $E$  ein Gewicht hat. Sei  $T \subseteq V$  eine Knotenmenge, die weniger oder gleich viele Knoten besitzt. Gesucht ist der Baum, der alle Knoten aus  $T$  verbindet und dabei falls nötig weitere Knoten aus  $V \setminus T$  (nicht original in  $T$ ) einfügt und benutzt, wobei das gesamte Gewicht minimal ist. Mit  $T = V$  ist der erzeugte Steinerbaum der Spannbaum von  $G$ .

In der Praxis wurde die Generierung des rektilinearen Steinerbaums (RST) mit minimaler Verbindungslänge intensiv erforscht. Sie wird heute häufig angewandt. Das rektilineare 2D-Routingproblem der Mehrpunktnetze entspricht der Generierung eines rektilinearen Steinerbaums. Knoten des RST, die nicht zu Anschlusspunkten gehören, werden als Steinerpunkte bezeichnet. Ein RST mit minimaler Kantenlänge wird als minimaler rektilinear Steinerbaum (MRST) bezeichnet (siehe Bild 40d).

Da das rektilineare oder euklidische Steinerbaumproblem immer NP-vollständig bleibt, versucht man in der Praxis durch die Approximationsalgorithmen (z. B. [1], [45], [61], [113], [119]) das Problem suboptimal zu lösen. Die meist effizienteren Algorithmen basieren auf einem minimalen Spannbaum, da die Länge des MST höchstens  $3/2$  der Län-

ge des MRST entspricht. Die Suche nach den richtigen Steinerpunkten dauert meistens länger als die Suche nach einem minimalen Spannbaum, insbesondere für das Routing mit vorgegebenen Hindernissen. Für das 3D-MID-Routing ist der in [1] vorgestellte Approximationsalgorithmus geeignet, weil er auf Basis der oben vorgestellten Verbindungsgraphen  $G_s$  oder  $G_c$  durchgeführt werden kann und in der Lage ist, die Steinerpunkte zwischen den vorgegebenen Hindernissen zu finden. Indem der entsprechende Steinerbaum durch die Integration des modifizierten DUNE-Algorithmus und der modifizierten „Three-Point Steinerization (P3S)“ Heuristik [1] generiert wird, kann das Routing für die auf einer Fläche laufenden Mehrpunktnetze auf den Schaltungsträger realisiert werden.

#### 4.4.2 Lösungsansätze für Mehrpunktnetze im 3D-MID-Routing

##### Die Strategie des Routingsverfahrens für Mehrpunktnetze

Die Strategie des Routingsverfahrens für Mehrpunktnetze ist in Bild 41 dargestellt.

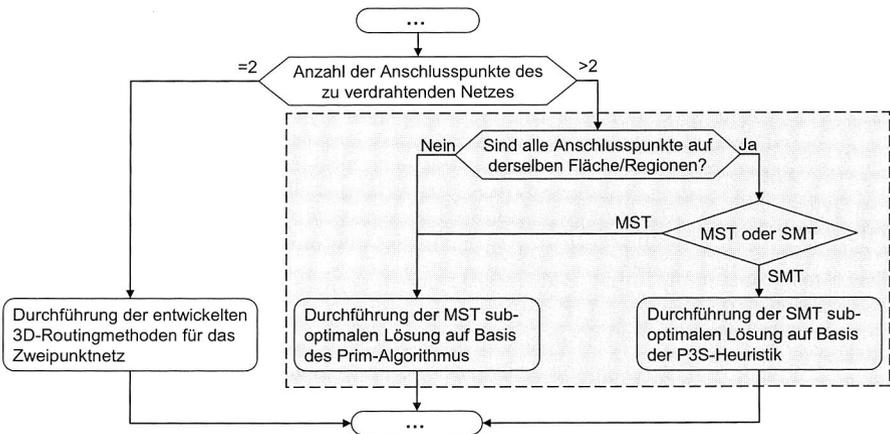


Bild 41: Auswahl der verschiedenen Routingmethoden im 3D-MID-Routing

Wenn die Anschlusspunkte eines Mehrpunktnetzes nicht auf derselben Fläche liegen, wird eine suboptimale Lösung durch die Verwendung des Prim-Algorithmus zur Generierung des entsprechenden minimalen Spannbaums verwendet. Da die Suche nach dem kürzesten Verbindungsweg zwischen zwei auf verschiedenen Flächen liegenden Punkten mit dazwischen liegenden Hindernissen schwer ist, ist die Generierung eines SMT für die Punkte, die auf verschiedenen Flächen liegen, viel komplexer als die für die Punkte, die sich auf einer Fläche befinden. Dafür gibt es bisher noch keine geeignete Lösungsmethode. Jedoch kann die Suche nach dem MST für die Punkte, die auf verschiedenen Flächen liegen, auf Basis der vorher für Zweipunktnetze entwickelten 3D-Routingalgorithmen zusammen mit dem Prim-Algorithmus realisiert werden.

Wenn alle Anschlusspunkte auf derselben Fläche sind, kann das Routing des Mehrpunktnetzes durch die beiden Methoden (Generierung des entsprechenden MST oder SMT) realisiert werden. Die Suche nach dem suboptimalen SMT, die auf Basis der P3S-Heuristik läuft, dauert zwar länger als die Suche nach dem MST, aber durch die zusätzlichen Steinerpunkte im SMT ist das Routingergebnis deutlich besser und rationaler. Im Folgenden werden die beiden Lösungsansätze kurz dargestellt.

### **Suboptimale MST-Lösung auf Basis des Prim-Algorithmus**

Der Prim-Algorithmus liefert, ebenso wie der Kruskal-Algorithmus, einen MST eines bewerteten ungerichteten Graphen. Im Vergleich mit dem Kruskal-Algorithmus läuft der Prim-Algorithmus schneller und effektiver ab. Der Prim-Algorithmus verfolgt einen ähnlichen Ansatz wie die Suche des kürzesten Weges in einem Graphen. Der Algorithmus beginnt mit einem Startgraph  $T$ , der lediglich einen beliebigen Knoten enthält. Zur Laufzeit wird in jedem Schritt eine Kante  $e$  mit minimalem Gewicht gesucht, die einen noch nicht in  $T$  enthaltenen Knoten  $v$  mit  $T$  verbindet. Die gefundene Kante  $e$  und der Knoten  $v$  werden zu  $T$  hinzugefügt. Die Suche wird solange fortgesetzt, bis alle Knoten in  $T$  vorhanden sind. Dann ist  $T$  ein minimaler Spannbaum. Die detaillierte Vorstellung und programmtechnische Umsetzung des Prim-Algorithmus können in der Literatur (z. B. [17], [92]) oder im Internet nachgeschlagen werden.

Auf Basis des gefundenen MSTs wird ein Mehrpunktnetz ( $N$  Anschlusspunkte) in eine Reihe von Zweipunktnetzen ( $N-1$ ) unterteilt. Damit ist das 3D-MID-Routing von Mehrpunktnetzen durch eine Reihe des Zweipunktnetz-Routings ersetzt.

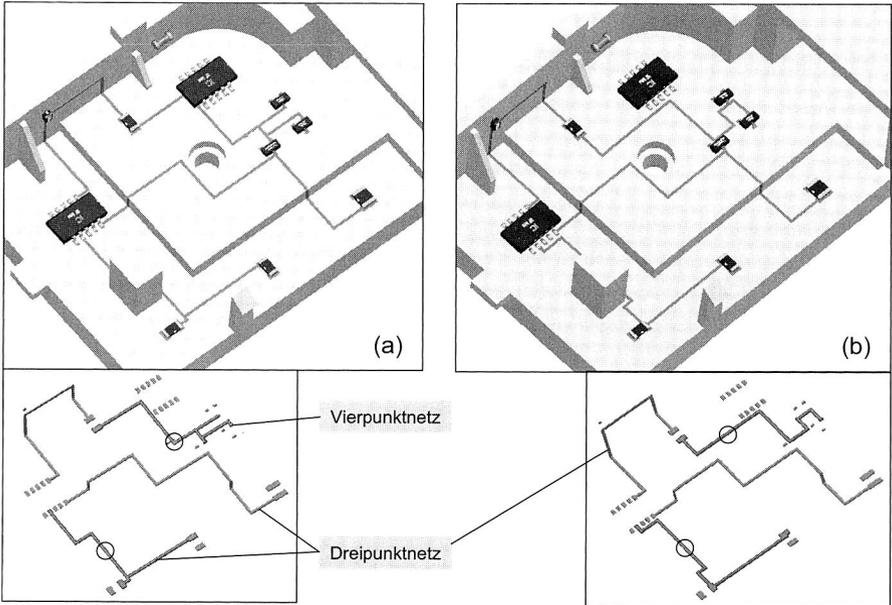
### **Suboptimale SMT-Lösung auf Basis der P3S-Heuristik**

Die P3S-Heuristik wurde zuerst von Lee, Bose und Hwang in [61] vorgestellt und später von Adler und Barke in [1] zur Realisierung eines CDRs (*Current Driven Routing*) mit Hindernissen modifiziert. Auf Basis des Verbindungsgraphen wird ein MRST durch sequentielles Einfügen eines Anschlusspunkts generiert. Jedes Mal wird ein Steinerpunkt für drei vorgegebene Punkte gesucht. Die Suche nach einem Steinerpunkt wird durch den Breitensuchalgorithmus im Verbindungsgraphen  $G_s$  realisiert. Der neu gefundene Steinerpunkt kann ein neu gefundener Punkt im Verbindungsgraphen  $G_s$  oder einer der drei vorgegebenen Punkte sein. Da die Suche nach dem Steinerpunkt nur in einem begrenzten Bereich (drei Punkte) stattfindet und die Einflüsse von anderen Punkten nicht berücksichtigt, ist der durch P3S-Heuristik gefundene RST ein suboptimaler MRST.

### **Experimentelle Resultate**

Ein Routingbeispiel ist in Bild 42 gezeigt. Auf dem 3D-Schaltungsträger gibt es vier Mehrpunktnetze. Die Anschlusspunkte von zwei mit Kreisen markierten Mehrpunktnetzen sind jeweils auf einer Fläche. Diese zwei markierten Netze wurden durch die SMT-Lösung auf Basis der P3S-Heuristik (siehe Bild 42a) und durch die MST-Lösung auf

Basis des Prim-Algorithmus(siehe Bild 42b) zwei Mal geroutet. Auf Bild 42 ist deutlich zu sehen, dass durch das Einfügen der Steinerpunkte das Routingergebnis von der P3S-Heuristik besser als das vom Prim-Algorithmus ist. Die anderen zwei Mehrpunktnetze wurden lediglich durch die MST-Lösungsmethode geroutet, weil die Anschlusspunkte nicht auf einer Fläche sind.



**Bild 42:** Routingbeispiel für Mehrpunktnetze. (a) Routing mit SMT-Lösung auf Basis der P3S-Heuristik, (b) Routing mit MST-Lösung auf Basis des Prim-Algorithmus

### Vorteile und Nachteile

Die Vor- und Nachteile der beiden Methoden wurden bereits zuvor dargestellt und werden im Folgenden kurz zusammengefasst. Das Routing durch die suboptimale MST-Lösung auf Basis des Prim-Algorithmus kann relativ effektiv und schnell durchgeführt werden und das Mehrpunktnetz mit den auf verschiedenen Flächen liegenden Anschlusspunkten behandeln. Im Gegenteil dazu läuft das Routing auf Basis der P3S-Heuristik relativ langsam, jedoch ist das Routingergebnis durch das Einfügen der Steinerpunkte in der Regel besser und rationaler. Weitere Nachteile der SMT-Lösung sind, dass die auf der Suche nach den Steinerpunkten basierende P3S-Heuristik nur auf einer Fläche durchgeführt wird, und wegen der Begrenzungen des Algorithmus die Ergebnisse in der Praxis bei manchen Situationen noch schlechter als die sich aus dem Prim-Algorithmus ergebenden sind.

## 4.5 Realisierung der 3D-Routing-Funktionen in MIDCAD

### 4.5.1 Routing-relevante Zusatzfunktionen

Die oben dargestellten Routingalgorithmen wurden bereits in MIDCAD als entsprechende Autoroutingfunktionen integriert. Um das 3D-MID-Routing in MIDCAD durchführen zu können, wurden darüber hinaus eine Reihe von routing-relevanten Funktionen (z. B. *Rip-up*) für die Vorbereitung und Nacharbeitung des Routings entwickelt.

#### Festlegung der Netzreihenfolge

Die Komplexität des Routingproblems führt dazu, dass die Netze bei den meisten Routingverfahren (inkl. den entwickelten 3D-Routingverfahren) sequentiell verlegt werden. Da sich die entwickelten 3D-Routingalgorithmen wie auch die meisten 2D-Routingalgorithmen immer nur auf die aktuell zu verlegende Verbindung konzentrieren, werden die nachfolgend zu verdrahtenden Verbindungen nur schwer oder gar nicht berücksichtigt. Somit hat die Verdrahtungsreihenfolge der Netze einen maßgeblichen Einfluss auf die Gesamtqualität des Routingergebnisses.

Für  $n$  Netze ist die Suche nach der optimalen Reihenfolge von der Laufzeitkomplexität  $O(n!)$  abhängig und bei der Bestimmung der Netzreihenfolge wird daher oft nach zuvor festgelegten Regeln verfahren [65]. Für das 3D-MID-Routing werden die folgenden zwei Regeln angewandt. Regel 1 hängt mit der Netzlänge  $L_{netz}$  zusammen und Regel 2 mit der Netztopologie hinsichtlich der Flächen, auf denen sich der Start- und Endpunkt befinden.

*Regel 1:* Routing vom Netz  $N_j$  vor  $N_k$ , falls  $L_j < L_k$ , d. h. kürzere Netze werden zuerst verdrahtet. Zur Kalkulation der Netzlänge vor dem Routing kann der Manhattan- oder der euklidische Abstand im Dreidimensionalen benutzt werden.

*Regel 2:* Routing vom Netz  $N_i$  vor  $N_j$  und  $N_j$  vor  $N_k$ , falls die Anschlusspunkte von  $N_i$  innerhalb einer Fläche, die Anschlusspunkte von  $N_j$  auf zwei benachbarte Flächen und die Anschlusspunkte von  $N_k$  auf die getrennten Flächen liegen.

Das Verfahren zur Bestimmung der Netzreihenfolge werde in die entwickelten 3D-MID-Routingalgorithmen integriert. Gemäß den oben dargestellten Regeln wird die Netzreihenfolge vor dem Routing bestimmt und im Routing-Setup Dialogfenster angezeigt. Aufgrund der Komplexität des 3D-Routings ist die nach Regeln bestimmte Netzreihenfolge in manchen Situationen nicht optimal, weshalb die Netzreihenfolge in dem Dialogfenster vor dem Routing manuell geändert werden kann.

#### Erzeugung der Trajektorie und des geometrischen Volumens der 3D-Leiterbahnen

Nach dem Routing sind zwar die Leiterbahnabschnitte zwischen den festgelegten Punkten durch Linien grafisch dargestellt, aber die realen Leiterbahnen zwischen den Punkten insbesondere auf der nicht-flachen Fläche sind nicht immer geradlinig. Zur genauen

Darstellung und weiteren Verbesserung des Routingergebnisses wird die detaillierte Trajektorie der Leiterbahnen als Feature *Datum Curve* auf dem Geometriemodell des Schaltungsträgers erzeugt.

Die als Feature *Datum Curve* dargestellten Verbindungen sind nur die Trajektorie der 3D-Leiterbahnen und besitzen jedoch kein reales dreidimensionales geometrisches Volumen. Zur Erzeugung des entsprechenden geometrischen Volumens der 3D-Leiterbahnen müssen zunächst die entsprechenden Leiterbahnquerschnitte definiert werden. Die Querschnittsformen können für die gesamte Schaltung, einzelne Leiterbahnen oder Leiterbahnabschnitte eingestellt werden (siehe Bild 43). Für das 2K-Verfahren wird ein spezielles Dialogfenster benutzt (siehe Bild 50). Die 3D-Leiterbahnzüge werden anschließend durch *Sweeping* des Querschnitts als Volumenfeature auf dem Geometriemodell des Schaltungsträgers erzeugt.

### **Keepout Surface/Region**

Auf dem Schaltungsträger kann es bestimmte geschlossene Gebiete oder besondere Oberflächen geben, wo keine Leiterbahnen verlaufen dürfen. Deswegen sollen vor dem Routing diese geschlossenen Gebiete als *Keepout Regions* oder die ganzen Oberflächen als *Keepout Surfaces* definiert werden. In MIDCAD wurden entsprechende Datenstrukturen und Funktionen zur Definition und Operationen (z. B. Löschen, Zeigen, usw.) der *Keepout Surface/Region* entwickelt. Damit kann das nachfolgende manuelle oder automatische Routing unter Berücksichtigung dieser versperrten Gebiete richtig durchgeführt werden.

### **Rip-up und Reroute**

Wegen der Komplexität des 3D-MID-Routingproblems und des sequentiellen 3D-MID-Routingverfahrens ist es möglich, dass die bereits verlegten Netze die noch zu verdrahtenden Verbindungen blockieren. Zur Lösung eines solchen Problems sind die Funktionen *Rip-up* und *Reroute* notwendig, mit denen bereits verlegte Netze wieder aufgetrennt (*Rip-up*) und nach Verlegen des bisher blockierten Netzes neu verdrahtet werden (*Reroute*) können. Zusammen mit den entwickelten 3D-Autoroutingalgorithmen können so auch komplexe 3D-MID-Routingprobleme gelöst werden. Alternativ kann auch im Routingverfahren die Strategie des *Push and shove* angewandt werden, welche das „Beiseiteschieben“ blockierender Verbindungen beinhaltet [65]. Die manuelle *Rip-up* und *Reroute* Funktionen sind bereits in MIDCAD realisiert. Die Entwicklung geeigneter Rip-up-and-reroute-Algorithmen bzw. Strategien für das 3D-MID-Routing ist eine der zukünftigen Entwicklungsaufgaben.

### **Redefine**

Nach dem Routing können die Leiterbahnen abschnittsweise interaktiv editiert werden. Dies erlaubt vor allem die Änderung der Abschnittsendpunkte, wie z. B. durch Einfügen,

Löschen und Umpositionieren eines Abschnittsendpunkts. Darüber hinaus wird die Möglichkeit angeboten, die Leiterbahnzüge parallel zu verschieben.

Nach der Erzeugung der Trajektorie der Leiterbahn sind jederzeit die im Anschluss beschriebenen Änderungen möglich. Die Leiterbahn kann durch *Rip-up* gelöscht werden und die entsprechende Verbindung wird wieder als Luftlinie dargestellt. Das *Routeprofil* (außerhalb des Start- und Endpunkts) der Leiterbahn innerhalb einer Fläche kann auch geändert werden, d. h. nicht nur die Änderung der Abschnittsendpunkte (Einfügen, Löschen, etc.), sondern auch eine Änderung der Linientypen oder das Einfügen von Eckrundungen sowie weitere Operationen sind möglich.

Weitere Änderungen sind nach der Erzeugung des 3D-Volumens der Leiterbahnen möglich. Die Querschnittsformen können neu definiert werden. Es ist auch möglich, dass die 3D-Leiterbahnengeometrie wegen einer Umpositionierung der Bauelemente automatisch angepasst wird, wobei die Informationen über den Leiterbahnquerschnitt erhalten bleiben.

Zu jedem Zeitpunkt kann eine Überprüfung auf Kurzschlüsse von verlegten Verbindungen (siehe Kapitel 5.3) durchgeführt werden. Dabei werden die editierten Leiterbahnen als richtig angenommen.

#### 4.5.2 Manuelle 3D-Routing-Funktion

Wegen der Begrenzungen der entwickelten Autoroutingfunktionen und der Komplexität des 3D-MID-Routings ist die manuelle 3D-Routingfunktion in MIDCAD obligatorisch. Dadurch können die komplexen 3D-Routingaufgaben zusammen mit anderen routingrelevanten Funktionen (z. B. Autorouting, *Rip-up* usw.) in MIDCAD erledigt werden.

Mit der manuellen Routingfunktion ist der Konstrukteur in der Lage, die Leiterbahnen in verschiedene Richtungen auf den 3D-Schaltungsträger manuell zu verlegen. Die Funktion soll das Verlegen der Leiterbahnen nicht nur auf flachen Flächen oder Zylinder, sondern auch auf den anderen Flächentypen unterstützen. Außerdem soll die manuelle Routingfunktion auf die Verdrahtung der Mehrpunktnetze angewendet werden.

Als wichtiges zusätzliches Hilfsmittel werde die Funktion für das manuelle 3D-Routing nach den oben genannten Anforderungen in MIDCAD entwickelt. Während des manuellen 3D-Routings für Mehrpunktnetze wird der Kruskal-Algorithmus auf der Suche nach MST eingesetzt, um korrekte Verbindungsbeziehungen bei der Mausbewegung zu zeigen. Außerdem wird beim Routing die Richtung der zu verlegenden Leiterbahn nach der Position der Maus automatisch beurteilt. Durch die Erkennung von Oberflächen-grenzen kann das Routing automatisch über eine gemeinsame Kante auf eine andere Fläche wechseln und fortgesetzt werden.

Die Grundidee des Kruskal-Algorithmus liegt darin, dass durch Hinzunahme von sicheren Kanten eine Menge von Bäumen solange ausgebaut wird, bis ein einziger minimaler Spannbaum entstanden ist. Zur Initialisierung werden alle Knoten als Teilbäume be-

trachtet. Dann werden zwei Teilbäume miteinander verbunden, indem einer Kante mit dem kleinsten Gewicht hinzugefügt wird. So werden die entsprechenden Knotenmengen dieser zwei Bäume zu einer Knotenmenge – einem neuen Baum – verschmolzen. Dieser Vorgang wird solange wiederholt, bis alle Knoten miteinander verbunden sind, und ein einziger Baum (MST) entstanden ist. Die detaillierte Vorstellung und programmiertechnische Umsetzung des Kruskal-Algorithmus kann in der Literatur (z. B. [17], [58]) oder auch im Internet nachgeschlagen werden.

Die Vorgehensweise beim manuellen Routing entspricht der Generierung eines Spannbauums durch den Kruskal-Algorithmus. Das zu verdrahtende Netz kann ein bisher nicht geroutetes Netz oder ein noch nicht fertig verdrahtetes manuell geroutetes Netz sein. Ein zu verdrahtendes Netz wird als eine Menge von Teilbäumen betrachtet, die am Anfang nur einen Anschlusspunkt haben und später mehrere Verbindungspunkte enthalten. Bei einem Vorgang wird ein Teilbaum ausgewählt und per Mausklick werden die Verbindungspunkte definiert und im ausgewählten Teilbaum hingefügt. Wenn die Position des Mausklicks auf einem anderen Teilbaum steht, werden die zwei Teilbäume verschmolzen. Sollte das Netz nur einen Baum enthalten, bedeutet dies, dass das Netz fertig geroutet ist.

#### **4.5.3 Vorgehensweise bei der Durchführung des 3D-MID-Routings in MIDCAD**

In MIDCAD werden die verschiedenen 3D-MID-Routingprobleme durch die oben dargestellten routingrelevanten Funktionen zusammen mit den entwickelten Autoroutingfunktionen gelöst. Bild 43 zeigt anhand eines Beispiels die Vorgehensweise beim 3D-MID-Routing auf einen räumlichen Schaltungsträger. Die Durchführung des 3D-MID-Routings lässt sich in die folgenden 4 Schritte unterteilen.

*Schritt 1: Die Vorbereitung des Routings;* Dazu gehören hauptsächlich die Definition und das Einlesen der Netzliste. Die Netzliste, die die elektrischen Verbindungen zwischen den Anschlüssen der Bauelemente beschreibt, bildet den Ausgangspunkt für das Routingverfahren. Falls nötig erfolgt die Definition der *Keepout-Surfaces/Regions* und der Beziehungen der gegenseitigen Flächen ebenfalls vor dem Autorouting.

*Schritt 2: Durchführung des Autoroutings bzw. des manuellen Routings;* Wenn die Autoroutingfunktion in MIDCAD erfolgt, wird zuerst ein Autorouting-Setup Dialogfenster gezeigt, in dem die nötigen Informationen wie z. B. die Leiterbahnbreite, Isolierabstand usw. eingestellt werden können. Für die verschiedenen 3D-Routingprobleme wird die entsprechende Routingmethode eingesetzt. Die konkreten Vorgehensweisen bei diesen 3D-MID-Routingmethoden wurden bereits in früheren Kapiteln beschrieben.

*Schritt 3: Visualisierung und Optimierung des 3D-MID-Routingergebnisses;* In einer Schleife wird möglicherweise nicht für alle Netze ein erfolgreicher Verbindungsweg gefunden. Der gefundene Verbindungsweg ist möglicherweise nicht optimal. In diesen Fällen werden die routingrelevanten Funktionen, wie z. B. *Rip-up* und *Reroute*, sowie die Platzierungsfunktion eingesetzt, um das 3D-MID-Routing zu vervollständigen bzw. zu

optimieren. Nach der Erzeugung der Trajektorie der 3D-Leiterbahnen kann die Funktion *Redefine* zur weiteren Optimierung verwendet.

**Schritt 4: Vorgabe der Leiterbahnquerschnitte und Erzeugung der 3D-Leiterbahnen;** Gemäß der Vorgabe der Querschnittsform werden die 3D-Leiterbahnzüge auf Basis des Routingergebnisses und der Geometrie des Schaltungsträgers erzeugt. Die Leiterbahnen und der originale Schaltungsträger sind zwar stofflich integriert, aber in MIDCAD wird die Geometrie der 3D-Leiterbahnen als ein getrenntes spezielles Bauteil erzeugt, um die nachfolgenden Geometrieoperationen, wie z. B. der Erzeugung der Geometrie der Schüsse für Molddesign, leicht durchführen zu können. Die Querschnittsform und der Verbindungsweg der erzeugten 3D-Leiterbahnen können auch im Nachhinein durch die routingrelevanten Funktionen und Platzierungsfunktion leicht optimiert werden.

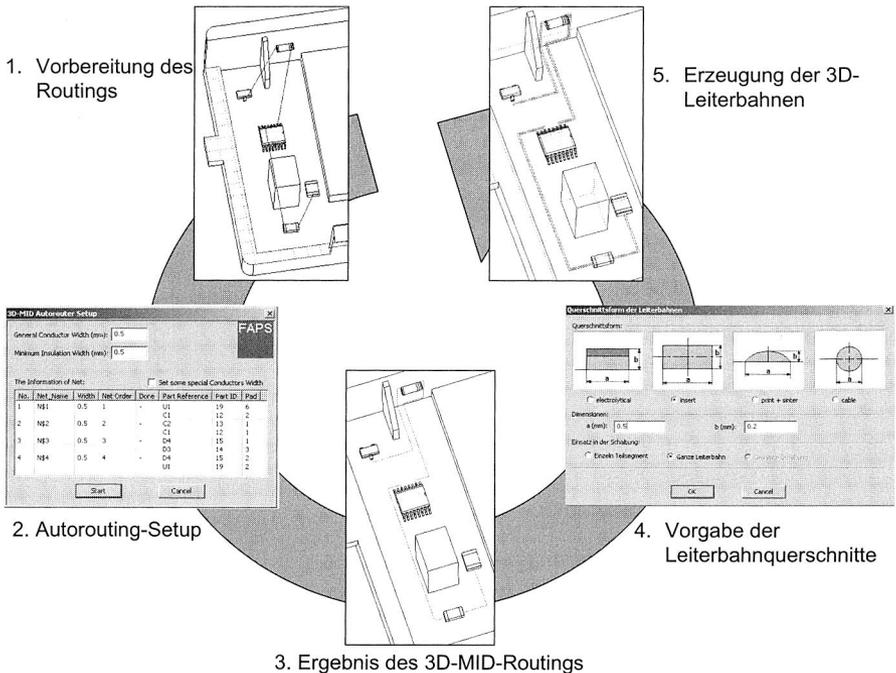


Bild 43: Vorgehen bei der Durchführung der 3D-Autoroutingfunktion in MIDCAD

## 5 Entwicklung neuer MID-spezifischer Funktionen

Neben der Realisierung und Integration der 3D-Routing-relevanten Funktionen in MID-CAD müssen weiterhin verschiedene grundlegende und MID-spezifische Funktionen zur Unterstützung der effizienten MID-Konstruktion entwickelt werden. Dadurch kann das MIDCAD-System aufgrund seiner speziell auf die MID-Technologie ausgerichteten Funktionen einen wertvollen Beitrag zur Effizienzsteigerung bei der Konstruktion von MID-Produkten leisten. In diesem Kapitel werden die 3D-Platzierung der Bauelemente und einige MID-spezifische Funktionen dargestellt.

### 5.1 3D-Platzierung elektronischer Bauelemente

Die Platzierung und das Routing bestimmen das Layout einer Schaltung. Die beiden Kernfunktionen, 3D-Platzierung und 3D-Routing, gehören zu den 3D-Layoutfunktionen in MIDCAD, die als Erweiterung der 2D-Layoutfunktionen in ECAD entwickelt wurden. Räumliche Schaltungsträger, insbesondere bei Verwendung des 2K-Verfahrens, haben eine sehr komplexe geometrische Struktur. Aus diesem Grund ist die 3D-Platzierung, also die Anordnung aller elektronischen Bauelemente auf den Flächen des räumlichen Schaltungsträgers, eine zentrale Aufgabe bei der MID-Konstruktion. Das globale Ziel der 3D-Platzierung ist die Lösbarkeit des 3D-Routingproblems und die Bestückbarkeit der Bauelemente.

Basierend auf den in Pro/TOOLKIT angebotenen Montagefunktionen kann die 3D-Platzierungsfunktion für den Bedarf von MIDCAD weiterentwickelt werden. Eine wichtige Voraussetzung ist, dass das zu platzierende Bauelement eine vordefinierte Bestückfläche enthält. Die Platzierung der Bauelemente unterliegt der Restriktion, dass jedes Bauelement mit seiner Bestückfläche auf einer Fläche des Schaltungsträgers positioniert werden muss. Als problematisch kann sich die Platzierung auf nicht-flachen Oberflächen herausstellen. Hier ist besonders bei Zylindern auf ausreichend große Radien zu achten.

Die Platzierung bzw. Positionierung des Bauelements erfolgt in der Regel interaktiv (siehe Bild 44): Das zu platzierende Bauelement wird durch die Selektion ausgewählt, danach werden in einer beliebigen Ansicht am Bildschirm per Mausklick eine Startposition bzw. eine bestimmte Fläche des Schaltungsträgers spezifiziert, die als erste vorübergehende Positionierfläche für das Bauelement behandelt und mit roter Farbe markiert wird. Das ausgewählte Bauelement wird an der Startposition unter korrekter Ausrichtung relativ zur ausgewählten Positionierfläche automatisch positioniert und weiter durch Verschieben der Maus auf der Fläche bewegt. Insbesondere auf einer nicht-flachen Fläche wird die Orientierung des Bauelements dynamisch so angepasst, dass die Flächennormalen der Bestückfläche des Bauelements und der ausgewählten Positionierfläche jederzeit parallel sind. Änderungen der Position des Bauelements und der farblich markierten Positionierfläche können jederzeit durchgeführt werden, d. h. mit Hilfe der Maus kann das Bauelement entweder auf der Oberfläche bewegt und gedreht

oder auf einer anderen Oberfläche platziert werden. Während des interaktiven Platzierungsvorgangs werden die dreidimensionalen Volumendarstellungen des Layouts und des zu platzierenden Bauelements dynamisch aktualisiert.

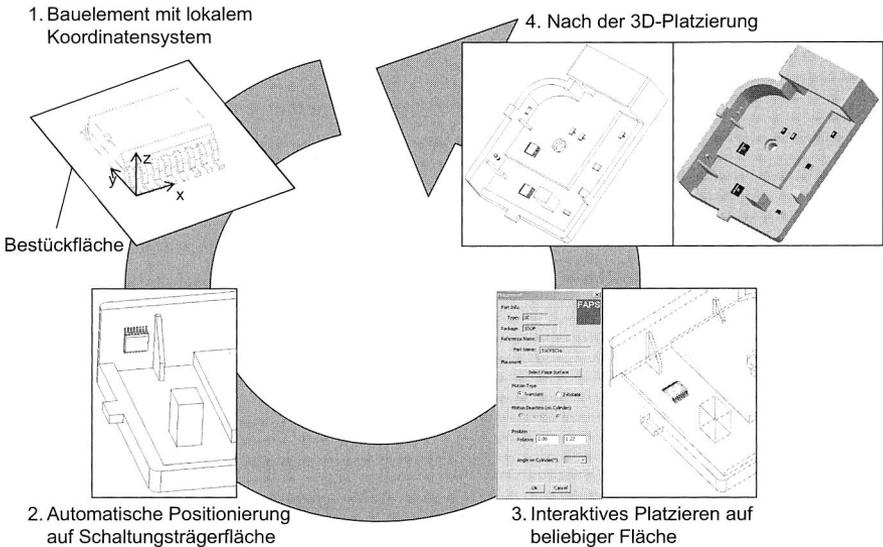


Bild 44: Vorgehen und Ergebnis der 3D-Platzierung

Zurzeit müssen alle Bauelemente interaktiv manuell platziert werden. In Zukunft soll zuerst die Platzierung der wichtigen Bauelemente und MID-spezifischen Features (z. B. Steckverbindung), die auf dem Schaltungsträger nur an bestimmten Stellen untergebracht werden können, interaktiv manuell durchgeführt werden. Anschließend soll die Platzierung unkritischer Bauelemente durch den zu entwickelnden Platzierhelfer hinsichtlich vorgegebener Platzierungsrichtlinien automatisch durchgeführt werden.

Die exakte Position des Bauelements auf der flachen Fläche kann durch die Eingabe der Verschiebungswerte in UV-Richtungen vom lokalen UV-Koordinatensystem der Fläche oder durch die Eingabe der Werte in einem vordefinierten Referenzkoordinatensystem bestimmt werden. Die exakte Richtung des Bauelements wird durch die Eingabe der Bezugswinkel im eigenen lokalen Koordinatensystem des Bauelements oder durch die Eingabe der Winkel in einem vordefinierten Referenzkoordinatensystem bestimmt. Die exakte Position des Bauelements auf nicht-flachen Oberflächen kann durch ein zusätzliches Koordinatensystem auf der Oberfläche des Schaltungsträgers und das lokale Koordinatensystem des Bauelements bestimmt werden. Zur Ausrichtung und dynamischen Erzeugung des zusätzlichen Koordinatensystems sind Informationen über Hauptmaße des Bauelements, die Richtung des Bauelements, die Eigenschaften der Oberfläche (z. B. Radius) usw. notwendig.

Platzierungs- und Verdrahtungsprobleme sind kombinatorische Optimierungsprobleme. Eine gute Platzierung ist Voraussetzung für ein optimales Routing, insbesondere im Dreidimensionalen. In der Regel werden die Informationen über die logischen Verbindungen der elektrischen Verschaltung vor der 3D-Platzierung durch das Einlesen der aus ECAD exportierten Netzliste ermittelt. Diese Verbindungen werden vor dem Routing durch Luftlinien (oder sog. Gummibänder), die die Anschlüsse der Bauelemente grafisch durch Linien verbinden, angezeigt. Dann wird die Platzierung der Bauelemente so durchgeführt, dass die verdrahtbaren Verbindungen unter Berücksichtigung gewisser Randbedingungen, wie z. B. einer minimalen Leiterbahnlänge zwischen den zu verbindenden Bauteilanschlüssen, auf dem räumlichen Schaltungsträger gefunden werden können.

Außerdem kann die 3D-Platzierungsfunktion mit DRC/MRC Funktionen verbunden werden, die die Kollision mit bzw. die Entfernung zu anderen platzierten Bauelementen oder den Wänden des Schaltungsträgers überprüfen können. Die 3D-Platzierung kann auch zu einem späteren Zeitpunkt zusammen mit der statischen Kollisionsprüfung angewandt werden, um kollisionsfreie Bestückpositionsdaten abzuleiten (siehe Kapitel 7.2.2 und Bild 84).

Die 3D-Platzierung und geometrische Struktur des Schaltungsträgers beeinflussen sich gegenseitig. Deswegen können Probleme durch die in Pro/ENGINEER existierenden Modellierungsfunktionen online gelöst werden, z. B. durch Planarisierung von Platzierungsflächen für hochpolige Bauelemente oder durch Vergrößerung des Schaltungsträgers an bestimmten Stellen zur Bestückung großer Komponenten. Damit können die Strukturoptimierung des Schaltungsträgers und die 3D-Platzierung der Bauelemente gleichzeitig in MIDCAD realisiert werden.

## 5.2 Definition der Netz- und Komponentenliste

Gemäß dem Konzept des MIDCAD-Systems (siehe Kapitel 3.2.3) erfolgt der elektrische Schaltplanentwurf im ECAD-System. Die Netzliste, die die Information über die elektrische Konnektivität zwischen den einzelnen Bauelementen enthält, und die Komponentenliste, die die Bauelemente in einem Schaltplan spezifiziert, werden durch die entwickelte Verknüpfung über Austauschdateien ins MIDCAD-System importiert. Dabei werden die entsprechenden 3D-Bauelemente aus der 3D-Bauelementebibliothek automatisch ausgewählt und die elektrischen Verbindungen zwischen den Anschlüssen der Bauelemente durch Luftlinien dargestellt.

Zwar ist der Schaltplanentwurf keine Kernfunktion von MIDCAD, aber über die Importmöglichkeit hinaus bietet auch MIDCAD selbst einige grundlegende Funktionen zur Definition und Editierung von Netz- und Komponentenlisten. Durch diese zusätzlichen Hilfsmittel können die speziellen Anforderungen beim 3D-MID-Design erfüllt werden.

Die Besonderheit der räumlichen Schaltungsträger ist die stoffliche Integration der mechanischen Geometrie und der elektronischen Funktionen. Einige geometrische Struk-

turen sind mit einer bestimmten elektronischen Funktionalität verbunden. Diese so genannten MID-spezifischen Features spielen eine wichtige Rolle für die elektrische Verbindung zwischen den Bauelementen. Wenn ein MID-spezifisches Feature auf den Schaltungsträger im MID-Design eingefügt wird, gibt es je nach verwendetem Feature zwei Möglichkeiten: Entweder die direkt elektrische Verbindung zwischen den Bauelementen wird durch zwei Teilverbindungen ersetzt (z. B. bei einem Via), oder eine neue elektrische Verbindung mit dem eingefügten MID-spezifischen Feature wird zusätzlich definiert (z. B. bei einer Steckverbindung oder einer Antenne).

Darüber hinaus haben viele MID-Komponenten, insbesondere bei der Applikation des 2K-Verfahrens, nur relativ einfache elektrische Schaltungen. In einer solchen Situation ist MIDCAD in der Lage, ohne ECAD die Netz- und Komponentenliste zu definieren.

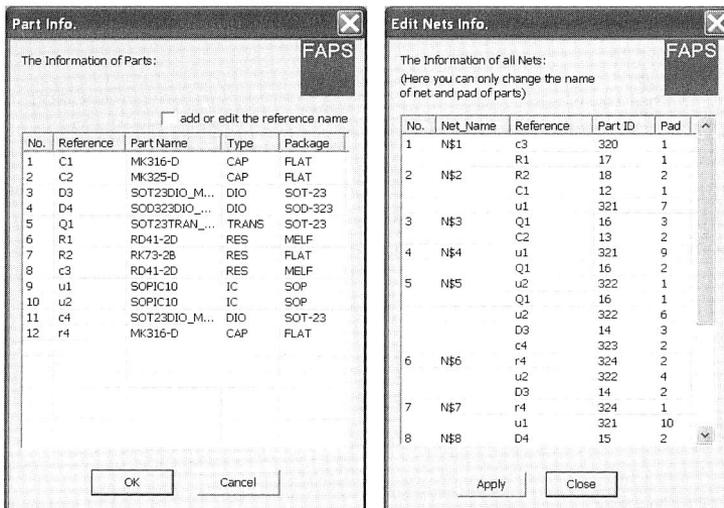


Bild 45: Definition und Anzeige der Netz- und Komponentenliste in MIDCAD

Aus der 3D-Bauelementebibliothek können die nötigen Bauelemente interaktiv ausgewählt und durch die oben beschriebene 3D-Platzierungsfunktion auf dem Schaltungsträger platziert werden. Es gibt auch die Möglichkeit, ein eingefügtes Bauelement wieder zu löschen. Die Daten der Komponentenliste werden im Hintergrund automatisch aktualisiert, und wenn nötig angezeigt (siehe Bild 45). Durch die Selektion der Bauelemente oder Features und einen weiteren Mausklick an den betroffenen Anschlüssen wird eine elektrische Verbindung in MIDCAD auf einfache Weise interaktiv erzeugt und die Information über diese Verbindung automatisch in die Netzliste eingefügt. Die Löschung einer Verbindung kann ebenfalls per Mausklick an den entsprechenden Anschlüssen erfolgen. Weitere Definitionen oder Änderungen der Verbindungen bzw. der Netzliste können in einem Dialogfenster (siehe Bild 45), in dem die Netzliste angezeigt wird, durchgeführt werden.

### 5.3 Überprüfung der Konstruktions- und Fertigungsregeln (DRC/MRC)

Design- und Manufacturing Rule Checks (DRC/MRC) sind ein wichtiger Schritt in der Konstruktionsphase, bevor ein Produkt an die Fertigung weitergegeben wird. Dadurch kann ein Konstrukteur die Probleme während des Entwurfszeitraums frühzeitig entdecken und so das Design verbessern. Deswegen soll die Überprüfung auf Einhaltung der MID-relevanten Design- und Manufacturing Rules möglichst schon in MIDCAD erfolgen.

Bestandteil der zu implementierenden DRC/MRCs sind insbesondere die minimal möglichen Abstände zwischen den verschiedenen elektronischen Komponenten und MID-spezifischen Features wie SMD-Bauelemente, Leiterbahnen, Pads, Vias etc., die auch im Dreidimensionalen abgeprüft werden müssen, um mögliche Kurzschlüsse zu vermeiden. Die Überprüfung wird im jetzigen Entwicklungsstand mit Hilfe selektiver Anforderung, d. h. halbautomatisch, in Bezug auf ausgewählte Design- bzw. Manufacturing Rules, durchgeführt.

Räumliche Schaltungsträger stellen besondere Anforderungen an die automatisierte Bestückung elektronischer Bauteile. Bevor die Bestückpositionen der Bauelemente in MID-Bestücksystemen abgeleitet werden, soll eine statische Kollisionsprüfung in MID-CAD durchgeführt werden, um Kollisionen während des Bestückens zu vermeiden (siehe Kapitel 7.2.5). Dazu muss die Geometrie der potentiellen Kollisionspartner, z. B. des Bestückkopfes, des räumlichen Schaltungsträgers und der Bauelemente im Detail analysiert werden. Um den gesamten Bestückungsablauf auf Kollisionsfreiheit zu prüfen und die kollisionsfreie Anweisungsliste abzuleiten, ist eine 3D-Kinematiksimulation durchzuführen (siehe Bild 84).

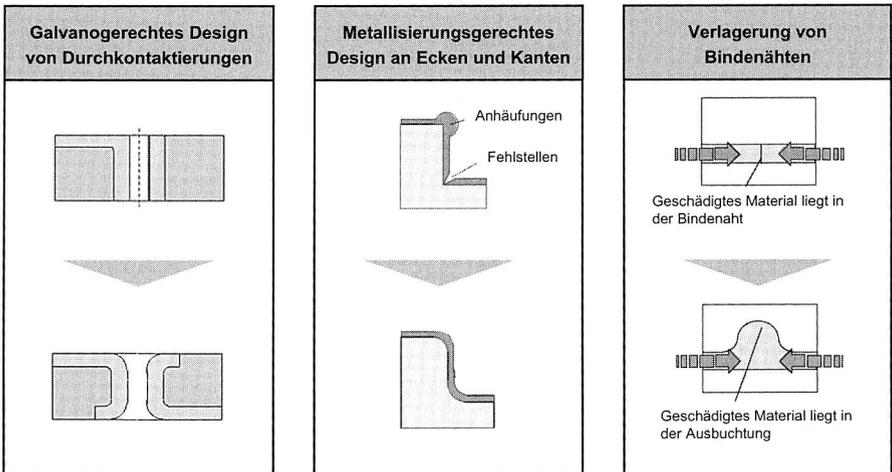


Bild 46: Designregeln für eine fertigungsgerechte Auslegung von MID-Produkten [68]

Die Überprüfung kann immer nur im Hinblick auf ein bestimmtes Herstellungsverfahren erfolgen. Da die vielfältigen Herstellungstechnologien für räumliche Schaltungsträger, z. B. 2K-Verfahren, Heißprägen, Laserdirektstrukturierung etc., auch eine völlig unterschiedliche geometrische Gestaltung der Schaltungsträger selbst und deren Leiterbilder erfordern und erlauben, ist konzeptionell vorgesehen, die konkrete geometrische Ausprägung des Schaltungslayouts so weit wie möglich über die Design Rules zu steuern. Ein Beispiel einer spritzguss- und metallisierungsgerechten Konstruktion einer MID-Komponente ist die Gestaltung der Übergänge in Form von Radien, sowohl bei Durchkontaktierungen als auch an Ecken und Kanten (siehe Bild 46).

Um Design- und Manufacturing Rule Checks automatisch durchführen zu können, befindet sich derzeit ein wissensbasiertes Kontrollmodul in der Entwicklung. Eine wichtige Voraussetzung dafür ist eine Technologiedatenbank, in der die MID-relevanten Konstruktions- und Fertigungsregeln gespeichert werden können. Die bereits vom traditionellen PCB-Entwurf bekannten Konstruktionsregeln müssen für räumliche Schaltungsträger erweitert werden. Die Design- und Manufacturing-Rules sollen einerseits aus den am Lehrstuhl gewonnenen Forschungsergebnissen im Bereich der MID-Produkte abgeleitet, und andererseits in Zusammenarbeit mit MID-Konstrukteuren sowie basierend auf dem 3D-MID-Handbuch [86] entwickelt werden.

Auf Basis der MID-Technologiedatenbank wird die Überprüfung mittels der aus dem Produktmodell entnommenen Daten und Beziehungen der Features automatisiert, so dass beispielsweise bei Nichteinhaltung einer Design- bzw. Manufacturing Rule automatisch eine entsprechende Meldung ausgegeben wird. Bild 47 zeigt die Vorgehensweise bzw. das Konzept des wissensbasierten Kontrollmoduls. Die Definitionen des Produktmodells und die allgemeinen und MID-spezifischen Features werden in Kapitel 6 dargestellt.

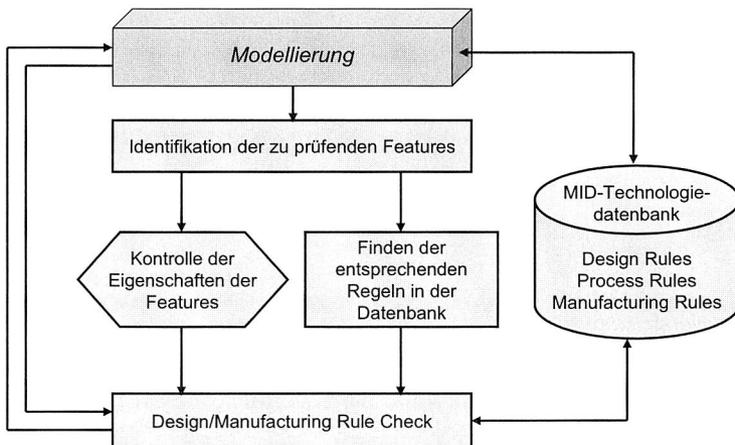


Bild 47: Konzept des Kontrollmoduls auf Basis der Technologiedatenbank

## 5.4 MID-spezifische Funktionen für 2K-Verfahren

Das Zweikomponentenspritzgussverfahren zählt zu den wichtigsten und am häufigsten angewandten Herstellungsverfahren für MID-Produkte. Die fertigungsgerechten Konstruktions- bzw. Gestaltungsrichtlinien sind zwar bereits in der MID-relevanten Literatur dokumentiert, dennoch basiert die Konstruktion bei Anwendung des 2K-Verfahrens bisher vorwiegend auf der persönlichen Erfahrung weniger Experten und ist sehr schwer im CAD-System durchzuführen. In MIDCAD sind einige spezielle Funktionen enthalten, um eine fertigungsgerechte Konstruktion unter Verwendung des 2K-Verfahrens zu unterstützen.

### 5.4.1 Kurze Verfahrenbeschreibung des Zweikomponentenspritzgusses

Der Zweikomponentenspritzguss ist ein Verfahren, bei dem eine metallisierbare und eine nicht metallisierbare Werkstoffkomponente in zwei Arbeitsschritten aufeinander gespritzt werden.

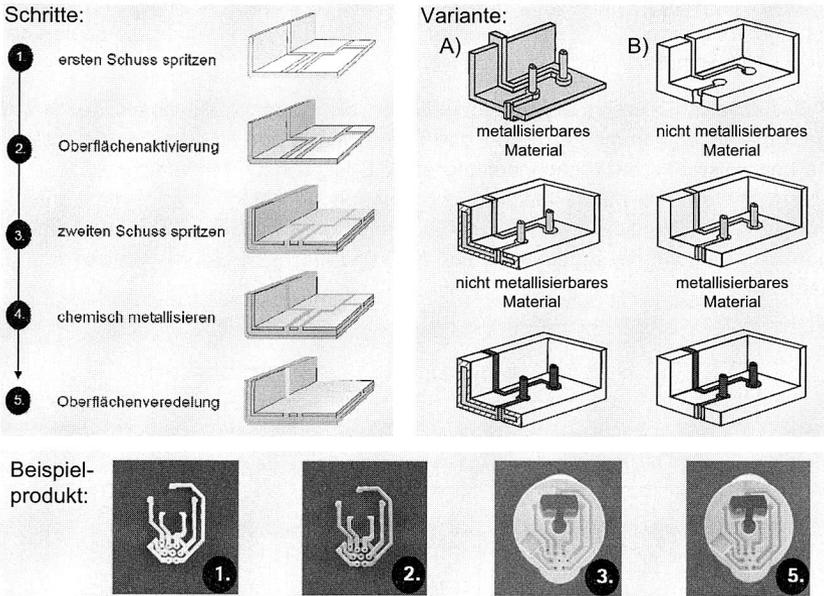


Bild 48: Herstellungsschritte und Varianten des 2K-Verfahrens [86]

Für den ersten Schuss kann zunächst ein metallisierbarer, nicht elektrisch leitender Kunststoff verwendet werden. Dabei wird die Leiterbahngeometrie des MID erhaben abgebildet. Im zweiten Schuss werden die Bereiche zwischen den Leiterbahnen mit einem nicht metallisierbaren Kunststoff aufgefüllt (Bild 48, Variante A). Das Beispielprodukt in Bild 48 gehört zu diesem Typ.

Alternativ dazu kann im ersten Schuss die Leiterbahnstruktur als Vertiefung aus der nicht metallisierbaren Komponente gespritzt und im zweiten Schuss mit der metallisierbaren Komponente aufgefüllt werden (Bild 48, Variante B). Zu dieser Variante gehört beispielhaft das Produkt Steckerplatine für ein ABS-System (siehe Bild 49).

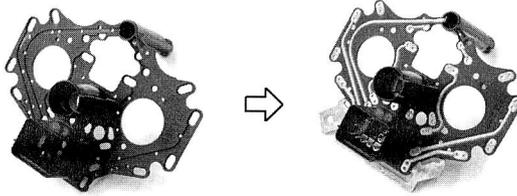


Bild 49: Ein Beispielprodukt des Zweikomponentenspritzgusses [68]

Bild 48 zeigt detailliert die Herstellungsschritte beim 2K-Verfahren. Nach dem zweiten Schuss hat das MID-Teil seine endgültige geometrische Form. Im nachfolgenden Schritt werden die entsprechenden Metalle auf den metallisierbaren Kunststoff aufgebracht. Die Oberfläche des metallisierbaren Kunststoffes wird aktiviert und die gewünschte Kupferschichtdicke chemisch oder galvanisch aufgebracht. Im anschließenden Schritt wird die vorgesehene Oberflächenveredelung, z. B. Ni-Au, appliziert. Wegen der Verwendung von nasschemischen Bädern eignet sich das 2K-Verfahren nur bedingt für dekorative Oberflächen [86].

Das 2K-Verfahren bietet die größte Gestaltungsfreiheit aller MID-Herstellungsverfahren. Es kann die Dreidimensionalität optimal nutzen. Für andere Verfahren problematische Leiterbahngeometrien sowie Durchkontaktierungen können im 2K-Verfahren realisiert werden. Während des 2K-Verfahrens erfolgt gleichzeitig die Strukturierung der Leiterbahngeometrie. Da die Strukturierung der Leiterbahnen direkt beim Spritzgießen stattfindet, ist die Prozesskette kurz und effizient. Wegen der wenigen Metallisierungsschritte ist die Metallisierung preiswert und bietet eine hohe Prozesssicherheit. Mit dem 2K-Verfahren sind zudem hohe Ströme realisierbar.

Allerdings sind immer zwei Spritzformen für das 2K-Verfahren notwendig und meistens müssen bei der Layoutänderung der Leiterbahngeometrie beide Kavitäten der Spritzgusswerkzeuge umgearbeitet oder neu erstellt werden. Die Folge sind hohe Werkzeugkosten und ein hoher Änderungsaufwand. Da die Strukturierung durch Spritzgießen von Kunststoff realisiert wird, sind sehr feine Leiterbahnstrukturen schwierig darstellbar. Die kleinste mit diesem Verfahren erreichbare Leiterbahnbreite liegt bei 0,25 mm. Sie ist abhängig von den Fließigenschaften des Kunststoffes und den Fließlängen im Werkzeug. Folgende Randbedingungen kennzeichnen die Anwendung des 2K-Verfahrens:

- Komplexe dreidimensionale Geometrie des Schaltungsträgers, insbesondere Leiterbahnen,
- geringe Anzahl separater Leiterzüge und elektronischer Bauteile,

- Leiterbreiten bzw. Abstände möglichst größer als 0,25 mm,
- hohe Stückzahlen aufgrund der hohen Entwicklungs- und Werkzeugkosten.

### 5.4.2 Definition der Leiterbahnquerschnittsform für 2K-Verfahren

Bei der Konstruktion von 2K-Spritzguss-Teilen muss man zuerst entscheiden, welche Variante (A oder B) beim 2K-Verfahren angewandt wird, d. h. ob der metallisierbare Kunststoff im ersten oder im zweiten Schuss gespritzt wird. Diese Entscheidung wirkt sich auf alle darauf folgenden Konstruktionsschritte aus. Weiterhin muss die Form der Leiterbahnen bestimmt werden, die erhaben, flach oder versunken sein können. Aufgrund der größeren Oberfläche können erhabene Leiterbahnen auf gleicher Breite ein Vielfaches an Stromstärke übertragen. Wie in Bild 50 dargestellt, ist  $b$  bei Typ 6 gleich  $a/2$ , daher kann die Leiterbahn zweimal so viel Strom übertragen wie bei Typ 4. Dreimal so viel Strom kann übertragen werden, wenn  $b = a$  ist. Durch horizontale, gestufte oder angewinkelte Shut-offs wird eine gute Abdichtung für den zweiten Schuss erzielt. Obwohl vertikale Shut-offs nicht optimal sind, werden die Typen mit dem vertikalen Shut-off (z. B. Typ 3, 6, 8) bei manchen Situationen wegen der engen Raumverhältnisse dennoch ausgewählt.

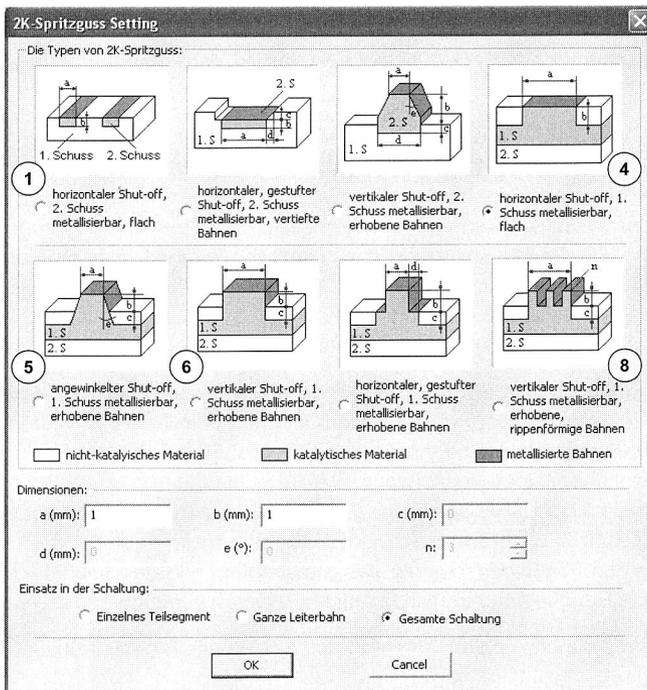


Bild 50: Definition der Leiterbahnquerschnittsform für das 2K-Verfahren

Bild 50 verdeutlicht, dass in MIDCAD ein Dialogfenster speziell für das 2K-Verfahren enthalten ist. Dadurch kann man den Typ des 2K-Verfahrens (Variante A: 4-8, Variante B: 1-3) auswählen und gleichzeitig verschiedene Leiterbahnquerschnittsformen definieren. Im Vergleich dazu ist das in Bild 43 gezeigte Dialogfenster für die Definition der Leiterbahnquerschnittsformen für andere MID-Herstellungsverfahren zuständig. Danach wird auf Basis der Routingergebnisse die Geometrie der 3D-Leiterbahnen automatisch erzeugt, was eine wichtige Voraussetzung für die nachfolgende Erzeugung der Geometriemodelle des ersten und zweiten Schusses ist.

### 5.4.3 Automatische Erzeugung der Geometriemodelle des ersten und zweiten Schusses

Wegen der fehlenden Funktionalitäten im CAD-System wurden die beiden Schüsse vorher meist als zwei getrennte Teile konstruiert. Auf diese Weise wurde nicht nur der Zusammenhang zwischen beiden Schüssen, sondern auch ihre Abhängigkeit von den Geometrien des gesamten Schaltungsträgers und der 3D-Leiterbahnen nicht deutlich dargestellt. Diese führt zu vielerlei Problemen und einer niedrigen Effizienz in der Konstruktion, wenn z. B. die Geometrie oder das Routing der Leiterbahnen geändert werden. In diesem Fall müssen auch die entsprechenden Geometrien der beiden Schüsse getrennt voneinander manuell korrigiert werden.

Für MIDCAD wurde eine spezielle Funktion entwickelt, die nach dem Routen die Geometriemodelle der 3D-Leiterbahnen sowie des ersten und zweiten Schusses, wie sie für das 2K-Verfahren benötigt werden, automatisch erzeugt. Dies ist insbesondere bei der Realisierung komplexer 3D-Leitungsstrukturen von Vorteil.

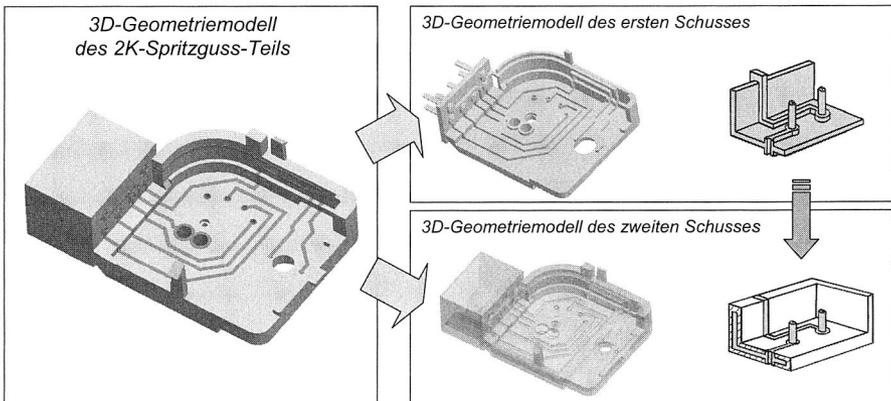


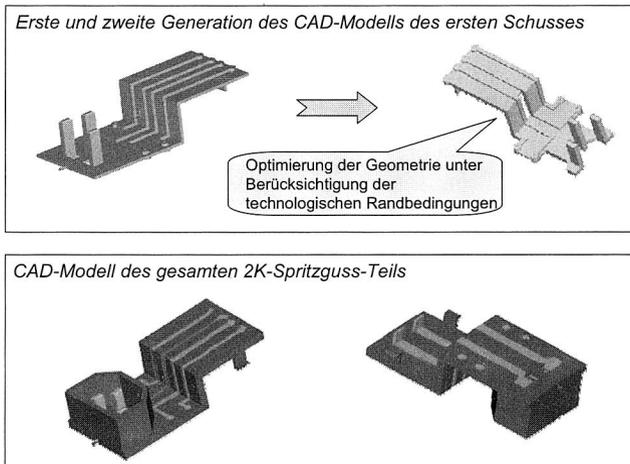
Bild 51: Automatische Generierung der Geometriemodelle des ersten und zweiten Schusses

Bild 51 zeigt ein im 2K-Verfahren hergestelltes MID-Produkt. Zur Erzeugung des Geometriemodells der 3D-Leiterbahnen wurden der Typ des 2K-Verfahrens (Typ 4, Variante

A) und die Querschnittsformen der Leiterbahnen nach dem Routen definiert (siehe Bild 50). Aus den Geometrien des dreidimensionalen Schaltungsträgers und der Leiterbahnen können die Geometriemodelle der ersten und zweiten Komponente durch eine Reihe von geometrischen und booleschen Operationen und einen Optimierungsalgorithmus automatisch erzeugt werden.

Bei der weiteren Optimierung und Verarbeitung der geometrischen Strukturen sollen die besonderen Anforderungen und technologischen Randbedingungen des Spritzgussverfahrens berücksichtigt werden [40], [68]. Die Geometrie des ersten und zweiten Schusses beeinflusst auch die Haftfestigkeit zwischen metallisierbarem und nicht metallisierbarem Kunststoff. Normalerweise ist die Haftung bei Variante A besser (siehe Bild 48), da der zweite Schuss den ersten Schuss besser auffüllen kann. Je mehr der zweite Schuss den ersten umschließt, desto stärker ist die mechanische Verhakung. Um genug Haftung zu erzielen, sind oft zusätzliche Hinterschneidungen oder Naben nötig.

Bild 52 zeigt, dass zusätzliche Durchbrüche und Verkleinerungen der Oberfläche in zweiter Generation des ersten Schusses durch lokale Verminderung des Spritzdruckes den Schmelzfluss des zweiten Schusses verbessern und eine mechanische Verzahnung der beiden Schüsse erzeugen. Diese Verarbeitungen werden zwar in der Regel manuell durchgeführt, einige topologische Optimierungen sollen bei der Erzeugung der Geometriemodelle unter Berücksichtigung der Gestaltungsrichtlinien jedoch automatisch durchgeführt werden.



Quelle: Singapore Institute of Manufacturing Technology

Bild 52: Zwei Generationen des CAD-Modells des ersten Schusses [40]

## 6 Entwicklung eines integrierten, MID-spezifischen Produktmodells

Die zentrale Problematik innerhalb der MID-Produktentwicklung liegt im Fehlen eines integrierten Produktdatenmodells, das sämtliche relevanten ECAD/MCAD-Daten sowie zusätzlich CAE/CAM-Daten der Prozesskette aufnehmen, und den an der Entwicklung beteiligten Applikationen in einem einheitlichen Format zur Verfügung stellen kann. Mangels eines geeigneten Datenmodells geschieht der Transfer der Daten von ECAD zu MCAD und weiter zu einem Berechnungs- oder Simulationswerkzeug über spezifische Austauschdateien. Darüber hinaus sind im Rahmen der Produktentwicklung verschiedene CAX-Werkzeuge im Einsatz, die selbst bei ähnlichem Inhalt unterschiedliche Formate für den Im- und Export der Daten benötigen. Dadurch wird der Redundanz- und fehlerfreie Datenaustausch erschwert, da eine Umwandlung in andere Dateiformate oder das manuelle Auslesen der benötigten Daten unumgänglich ist. Der dafür erforderliche Zeitaufwand sowie die durch Übertragungsfehler, Redundanzen und Inkonsistenzen bedingten möglichen Fehlerquellen sind für einen effizienten Entwicklungsprozess nicht tragbar.

### 6.1 Produktmodellierung

Ein Produktmodell stellt ein Abbild eines Produkts, Systems oder Vorgangs dar. Es ist das Resultat des Produktentwicklungsprozesses, in dem alle relevanten Eigenschaften eines Produkts herausgearbeitet und im Produktmodell dokumentiert werden. Das Produktmodell entsteht durch Instanziierung des Produktdatenmodells.

Dem Ansatz des Produktmodells (auch integriertes Produktmodell genannt), liegen folgende spezialisierte Anforderungen zu Grunde [41]:

- Abbildung von Produktinformationen aus allen Produktlebensphasen, wie z. B. Produktplanung, Konstruktion, Produktherstellung usw.
- Vereinigung von verschiedenen physikalischen und technischen Produkteigenschaften, wie z. B. mechanische, elektrische oder elektronische Eigenschaften.
- Berücksichtigung der Sichtweise eines Anwendungsgebietes, um den Zusammenhang von Produktdaten und ihrem Kontext herstellen zu können.

Im Folgenden werden bestehende Produktmodelltypen hinsichtlich ihres Aufbaus und ihrer Funktionalität dargestellt [23]:

**Strukturorientierte Produktmodelle:** Die Gliederung der Produkte nach physikalischen oder funktionalen Teilsystemen liefert strukturorientierte Produktmodelle, die in Form von Aufbauübersichten, Baugruppenstrukturen und verschiedenen Stücklistenformen, wie der Struktur- oder Baukastenstückliste, beschrieben werden. Die Abbildung des Produktmodells kann hierarchisch in Form von Strukturebenen, Gruppen oder Gliederungen dargestellt werden.

derungsebenen erfolgen. Mit diesem Konzept ist der Produktmodelltyp nicht in der Lage, eine redundanzfreie und einheitliche Datenhaltung zu ermöglichen, wie sie für eine interdisziplinäre MID-Produktentwicklung erforderlich ist.

**Geometrieorientierte Produktmodelle:** Die geometrieorientierten Produktbeschreibungen werden meist beim geometrischen Modellieren benutzt, um eine digitale Darstellung geometrischer Produkte oder Objekte zu erzeugen und zu verwenden. In Abhängigkeit von der Repräsentationsform der geometrischen Information unterscheidet man bei Geometriemodellen zwischen Draht-, Flächen- und Volumenmodellen. Diese Modelle bilden die Basis von CAD-Systemen. Die Datenstrukturen von Geometriemodellen sind jedoch so speziell, dass weitere administrative, technologische, elektrische und andere nichtgeometrische Daten, die von Applikationsprogrammen im Entwicklungsablauf benötigt oder erzeugt werden, schlecht integriert werden können. Daher ist die Beschreibung von mechatronischen Produkten, insbesondere von MID-Produkten, auf der Basis von geometrieorientierten Produktmodellen nicht möglich.

**Featureorientierte Produktmodelle:** Eine Erweiterung der geometrieorientierten Produktmodelle stellen featureorientierte Produktmodelle dar. Ein Feature ist in diesem Zusammenhang die Aggregation von Geometrieelementen *und/oder* Semantik [88]. Um ein mechanisch-elektronisch integriertes MID-Produkt vollständig zu beschreiben, müssen zusätzlich zur geometrischen und strukturellen Beschreibung auch technische Zusammenhänge und Eigenschaften abgebildet werden. Das featureorientierte Produktmodell wird durch ein anwendungsunabhängiges Geometriemodell und ein anwendungsspezifisches, technologisches Modell repräsentiert [53]. Zum Beispiel werden spezifische Informationen für Spritzgussimulation sowie Fertigungs- und Montageinformationen an das Geometriemodell angehängt. Die Feature-Technologie ist prinzipiell geeignet, um die komplexen mechatronischen MID-Produkte im gesamten Produktentwicklungsprozess zu beschreiben.

**Wissensbasierte Produktmodelle:** Ein Modell bildet in diesem Sinne eine Wissensrepräsentation. Das Produktwissen umfasst die produktspezifischen Erfahrungswerte, Regeln, Technologie usw. Wissensbasierte Produktmodelle ermöglichen in der Entwicklung von MID-Produkten die Integration von Technologiewissen und Erfahrungswissen in den Entwicklungsprozess. Durch die Integration der Feature-Technologie, insbesondere durch die Repräsentation von Wissen in Form von Features, kann prinzipiell sämtliches Expertenwissen gespeichert und bearbeitet werden. Zugleich wird damit Simultaneous/Concurrent Engineering unterstützt [88].

**Integrierte Produktmodelle:** Integrierte Produktmodelle fassen die Teilbereiche struktur-, geometrie-, feature-, und wissensorientierter Produktmodelle zusammen. Während der Produktkonstruktion werden integrierte Produktmodelle eingesetzt, um den Detaillierungsprozess schrittweise von der Konstruktionsidee bis zum fertigen Produkt zu unterstützen. Ziel der Produktentwicklung auf Basis eines integrierten Produktmodells ist die Abbildung aller relevanten Daten eines Produkts auf ein einziges, einheitliches Datenmodell. Ein durchgängiges und integriertes Produktmodell ermöglicht es Unternehmen,

rechnergestützte Werkzeuge einzusetzen und über die verschiedenen Funktionsbereiche des Unternehmens die Produktdaten gemeinsam zu nutzen. Integrierte Produktmodelle bilden aufgrund ihrer Offenheit und Erweiterbarkeit die beste Plattform für eine interdisziplinäre MID-Produktentwicklung.

**STEP-Produktmodelle:** Die Entwicklung und Standardisierung des integrierten Produktmodells erfolgt derzeit innerhalb der ISO unter der Bezeichnung „*ISO 10303 Product Data Representation and Exchange*“ und ist unter dem Namen STEP bekannt geworden. STEP-Produktmodelle erlauben die einheitliche Beschreibung von allen im Lebenszyklus eines Produkts anfallenden Informationen, die für den Austausch von Produktdaten zwischen verschiedenen Anwendungsprogrammen verwendet oder für die gemeinsame Datenbasis genutzt werden können. Zielsetzung ist die redundanzfreie Abbildung aller Informationen, die innerhalb des gesamten Produktlebenszyklus anfallen bzw. generiert werden.

Im Zusammenhang mit integrierten Produktmodellen ist der Begriff „Sichten“ (*views*) von besonderer Bedeutung. Während das integrierte Produktmodell alle relevanten Daten in den verschiedenen Lebensphasen beschreiben soll, ist das Partialmodell ein Modell, das einen einzelnen Aspekt eines Produkts innerhalb des Produktlebenszyklus (z. B. die Anforderungen, die Funktion, die Gestaltung, die Berechnung usw.) abbildet. Deswegen wird die Unterstützung der einzelnen Phasen des Entwicklungsprozesses über Partialmodelle realisiert, die eine Bearbeitung der Produktdaten unter anwendungsspezifischen Gesichtspunkten in Form so genannter „Sichten“ ermöglichen. Eine Sicht repräsentiert eine spezifische Sichtweise auf die Produktbeschreibung, die mit bestimmten Eigenschaftsklassen und bestimmten Phasen des Produktlebenszyklus in Zusammenhang steht. Jede Sicht, z. B. Mechanik oder Elektronik, CAD oder CAE, beschreibt eine bestimmte Lebensphase anhand sichtspezifischer, featureorientierter Partialmodelle und die jeweils benötigten Sichten werden zum integrierten Produktmodell zusammengeführt.

## 6.2 Konzeption eines integrierten MID-Produktmodells

Eine wesentliche Charakteristik der heutigen mechatronischen Produktentwicklung ist die Interdisziplinarität. In MID-Produkten kommt eine Vielzahl elektrischer, elektronischer und mechanischer Elemente zur Anwendung, die funktionell eng miteinander verknüpft sind und gemeinsam die geforderte Gesamtfunktion erfüllen. Das Produktmodell beinhaltet in diesem Fall die Informationen aus den Domänen Elektronik, wie z. B. integrierte Schaltkreise, Elektrotechnik wie z. B. Stromlaufpläne und Mechanik, wie z. B. Schaltungsträger. Innerhalb der einzelnen Domänen kommen oftmals spezialisierte CAx-Systeme in verschiedenen Entwicklungsphasen zum Einsatz. Damit wird neben der horizontalen Integration domänenspezifischer Informationen auch die vertikale Integration der Informationen aus den verschiedenen Phasen eines Produkts in ein gemeinsames, integriertes Produktmodell gefordert (siehe Bild 53). Zur Beschreibung eines solchen integrierten Produktmodells werden die relevanten Techniken wie z. B.

Multi-Sichten (*multiple views*) [9], [62], Feature-Technologie, STEP-Norm und Partial- bzw. Multi-Modelle [115] eingesetzt.

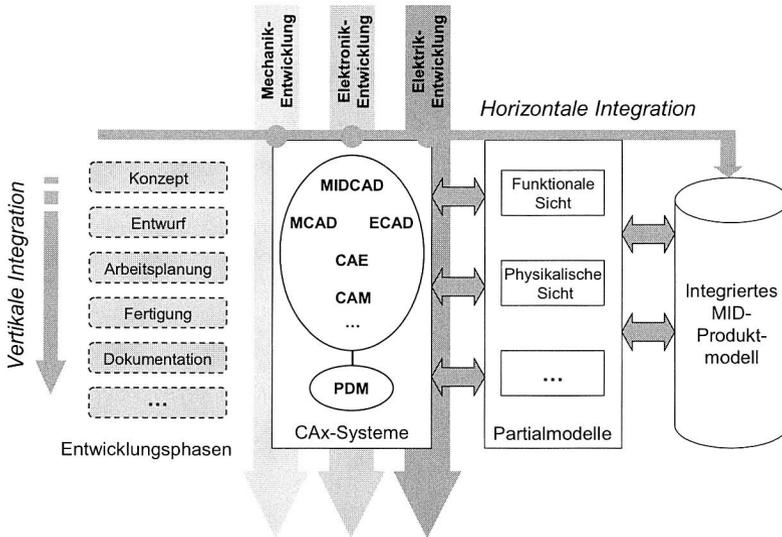


Bild 53: Konzept eines integrierten Produktmodells

Die Konzeption und Anwendung eines integrierten domänenübergreifenden MID-Produktmodells bildet einen zentralen Schwerpunkt in der vorliegenden Forschungsarbeit. Die Idee der Integration von Produktdaten auf Basis eines integrierten Produktmodells basiert auf der Definition einer Anzahl von Partialmodellen. Entsprechend den Bedürfnissen bzw. Anforderungen der verschiedenen Anwendungsdomänen (mechanische Entwicklung, elektronische Entwicklung usw.) werden die jeweils benötigten Partialmodelle zu einem integrierten Schema zusammengeführt und somit spezialisierte Anwendungsmodelle später aus dem integrierten Modell generiert.

Aus diesem Grund basiert das integrierte MID-Produktmodell auf mehreren multidisziplinären Partialmodellen. Das integrierte MID-Produktmodell ist dabei als Aggregat aus Partialmodellen aufgebaut und bildet die realen Beziehungen zwischen den Partialmodellen unterschiedlicher Disziplinen ab. Das integrierte MID-Produktmodell wird grundsätzlich auf zwei Partialmodellen aufgebaut: Eines für die elektrische Schaltung und ein weiteres für die dreidimensionale Geometrie des Schaltungsträgers. Dadurch kann die Besonderheit von MID-Produkten - die stoffliche Integration mechanischer und elektronischer Funktionen - in einem integrierten Produktmodell beschrieben werden (siehe Bild 54). Neben den oben genannten 3D-MCAD- und ECAD-Partialmodellen sind die Informationen der anderen relevanten Fachbereiche (z. B. CAE, CAM), unter Berücksichtigung der Anforderungen der entsprechenden Anwendungen, in das Produktmodell einzufügen.

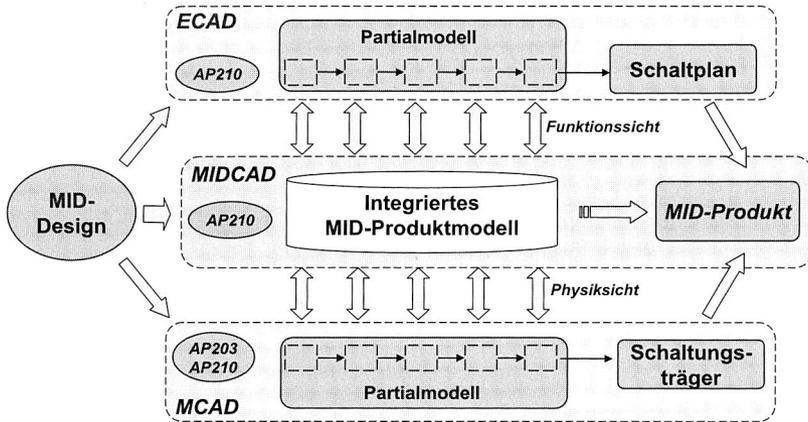


Bild 54: Verbindung elektrischer und mechanischer Partialmodelle zu einem integrierten MID-Produktmodell

Zur Beschreibung der ECAD- und MCAD-Partialmodelle sind zwei Sichten von Bedeutung: Die funktionale Sicht und die physikalische Sicht. Mit Hilfe eines Funktionsmodells können elektronik- und strukturbezogene Aspekte von MID-Produkten abstrakt beschrieben werden. Entsprechend einem Funktionsmodell kann ein Physikmodell eingebunden werden, das für die Beschreibung der mechanik- und geometriebezogenen Daten zuständig ist. Dieses Schema, das sich analog auch in STEP AP210 wieder findet, dient als Grundlage für den Aufbau eines integrierten MID-Produktmodells und die horizontale Integration mechanischer und elektronischer Informationen.

Zur Datenbeschreibung der mechatronischen Produkte wurden die verfügbaren STEP-Anwendungsprotokolle, insbesondere AP210, AP203, AP212 sowie AP214 in Hinblick auf ihre Eignung zur Repräsentation von MID-Produktdaten überprüft. Nach der Vorstellung in Kapitel 2.3 ist dabei AP210 für die MID-Produktdatenbeschreibung von Bedeutung. Daher ist vorgesehen, das integrierte MID-Produktmodell in MIDCAD auf Basis von AP210 zu beschreiben. Dabei existieren zwei Möglichkeiten, die es zu untersuchen gilt. Zunächst muss geprüft werden, ob eine Projektion der bestehenden Partialmodelle im Anwendungsprotokoll (hier AP210) auf den Anwendungsfall, d. h. für das MID-Produkt, möglich ist. Können bestimmte MID-Produktmerkmale nicht direkt durch diese Partialmodelle von AP210 repräsentiert werden, ist eine Erweiterung dieser Partialmodelle in Erwägung zu ziehen.

Darüber hinaus besteht die Möglichkeit des Einsatzes der Feature-Technologie zur Beschreibung berechnungs-, fertigungs- und montagerelevanter Informationen im integrierten MID-Produktmodell. Mit Hilfe der Feature-Technologie können bereits bei der Konstruktion im MIDCAD-System berechnungsrelevante Informationen (z. B. Materialeigenschaften, Randbedingungen usw.) im integrierten Produktmodell mitgeführt werden, um sie direkt für eine nachfolgende Spritzgussimulation zu nutzen.

## 6.3 Entwicklung einer mechanisch-elektronisch integrierten Bauelementebibliothek in MIDCAD

### 6.3.1 Konzeption einer 3D-Bauelementebibliothek

Die neuen Anforderungen an das CAD-System für den domänenübergreifenden, mechatronischen Entwurf führen zu einer Erweiterung des Informationsinhalts und der Informationsbreite der elektronischen Bauelementebibliothek [35], [79]. Die Existenz einer dreidimensionalen elektronischen Bauelementebibliothek ist für das integrierte MID-CAD-System obligatorisch, um eine MID-Konstruktion in 3D durchführen zu können. Das Konzept einer elektronischen 3D-Bauelementebibliothek ist in Bild 55 dargestellt.

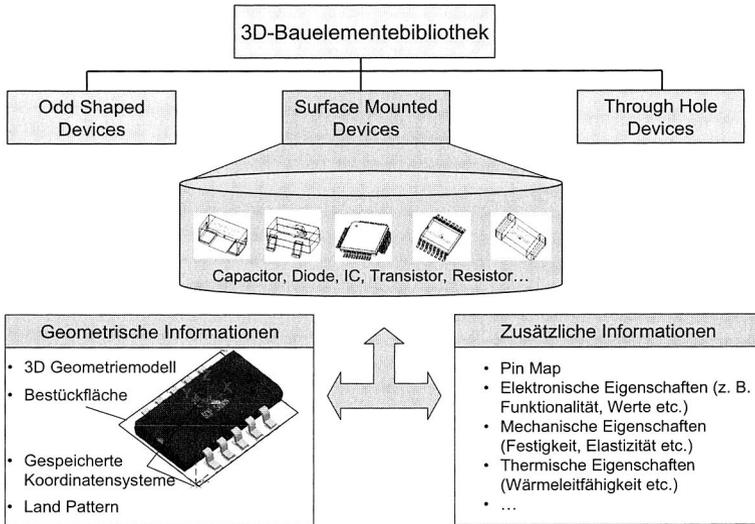


Bild 55: Konzept einer 3D-Bauelementebibliothek

Zur Berücksichtigung der Systemintegration aus dem elektronischen und mechanischen Bereich und deren Wechselwirkungen sowie des Bedarfs für die spätere Berechnung, Simulation und Produktion, müssen rechnerintern umfangreiche Informationen von Bauelementen in der Bibliothek gespeichert und verarbeitet werden können:

- Geometrische Eigenschaften: 3D-Geometriemodell, Bestückfläche, verschiedene Koordinatensysteme,
- elektronische Eigenschaften (elektronische Funktionalität, elektrische Werte, Pin-Zuordnung usw.),
- mechanische Eigenschaften (Festigkeit, Elastizität etc.),
- thermische Eigenschaften (Kapazität, Wärmeleitfähigkeit, Strahlung etc.),

- elektronisch-mechanische Eigenschaften (*Pin Map, Land Patterns* etc).

Neben dem erforderlichen 3D-Geometriemodell ist eine zusätzliche Bestückfläche für die 3D-Platzierung des Bauelements notwendig. Darüber hinaus kann ein zusätzlich in der Bauelementebibliothek hinterlegtes Koordinatensystem als definierter Greif- oder Ansaugpunkt für die Ableitung der Bestückposition verwendet werden. Neben der rein geometrischen Beschreibung der Bauelemente sollen in der Bibliothek die nichtgeometrischen Informationen gespeichert werden. Hierzu gehören beispielsweise elektrische Informationen, die in das Autorouting einfließen können, oder Werkstoffkennwerte wie die Wärmeleitfähigkeit, um das thermische Verhalten zu simulieren.

Bezüglich der oben dargestellten Informationen sind die Bereitstellung der geometrischen Eigenschaften und die Spezifizierung der elektronischen Funktionalität von Bauelementen in der Bibliothek für MIDCAD obligatorisch.

Der Aufbau einer mechanisch-elektronisch integrierten Bauelementebibliothek kann sich an AP210 orientieren [105]. Bild 56 zeigt die Struktur einer solchen Bauelementebibliothek. Das ganze Modul in MIDCAD besteht hauptsächlich aus vier Teilen: 3D-Bauelementebibliothek, 3D-Gehäusebibliothek, Funktionsbibliothek und Bauelementemodellgenerator.

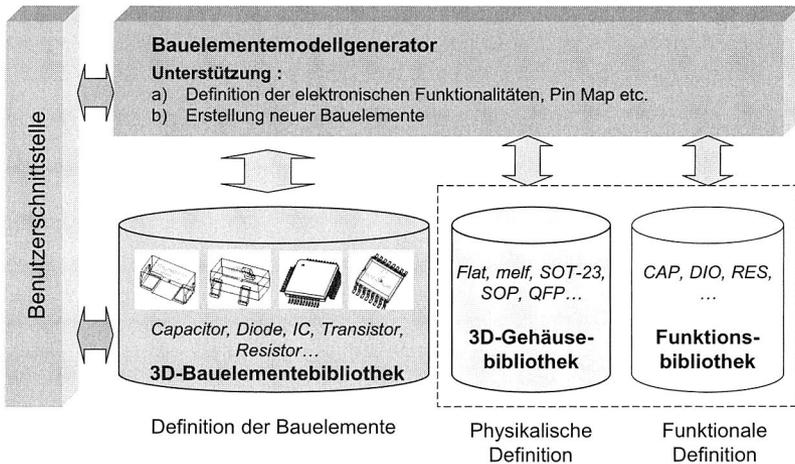


Bild 56: Struktur einer 3D-Bauelementebibliothek in MIDCAD

Der wichtigste Teil ist die 3D-Bauelementebibliothek, in der die einzelnen konkreten Bauelemente gespeichert werden. Der Konstrukteur kann direkt auf diese Bibliothek zugreifen, um die einzelnen Bauelemente auszuwählen. Wegen der prototypischen Entwicklung und der Konzentration auf die MID-Produktentwicklung werden lediglich SMD-Bauelemente (*Surface Mounted Devices*) in der Bibliothek gespeichert. THD (*Through Hole Devices*) und *Odd Shaped Devices* sind hierbei nicht berücksichtigt.

In der vordefinierten 3D-Gehäusebibliothek werden verschiedene SMD-Gehäusetyper gespeichert. Jeder SMD-Gehäusertyp wird als ein einziges Geometriemodell mit Pro/ENGINEER parametrisch modelliert und besitzt eine Familientabelle. Neben dem 3D-Geometriemodell sind zusätzliche geometrische Informationen einzufügen, wie z. B. die Bestückfläche und lokale Koordinatensysteme. Die konkreten Gehäuseinstanzen werden nicht einzeln in der Gehäusebibliothek abgelegt, sondern bei der Generation der Bauelemente spezifiziert. Eine Option bei der zukünftigen Entwicklung dieser Bibliothek ist die Bereitstellung der entsprechend vereinfachten Simulationsmodelle.

Die vordefinierte Funktionsbibliothek enthält verschiedene vordefinierte elektronische Funktionalitäten, die dem in AP210 definierten Applikationsobjekt *Functional\_unit\_usage\_view* entsprechen (siehe Bild 57). Es definiert ein bestimmtes elektronisches Funktionselement und stellt gleichzeitig die Definition aller Anschlüsselemente dar. Die Definition der entsprechenden schematischen Symbole für 2D-ECAD sowie die funktionalen Simulationsmodelle sind noch nicht in der derzeitigen Bibliothek berücksichtigt.

Der Bauelementemodellgenerator spielt eine entscheidende Rolle bei der Verwaltung der drei oben genannten Bibliotheken. Während der Konstrukteur die 3D-Bauelemente der Bauelementebibliothek direkt benutzen kann, ist der Zugriff auf die anderen zwei Bibliotheken (Gehäuse- bzw. Funktionsbibliothek) nur durch den Generator möglich. Der Generator ermöglicht dem Konstrukteur durch die Verwendung eines Assistenten sowohl die Erstellung neuer Bauelemente, als auch die Definition der elektronischen Funktionalität. Die detaillierte Beschreibung des Generators erfolgt in Kapitel 6.3.3.

### 6.3.2 Datenbeschreibung von Bauelementen nach STEP AP210

Die Datenbeschreibung der Bauelemente in der Bibliothek orientiert sich an AP210. Zur Beschreibung der Bauelemente sind drei Begriffe von besonderer Bedeutung: Sicht, Funktionsmodell und Physikmodell. Die Schemen und die Beziehungen zwischen Funktionsmodell und Physikmodell, die sich in AP210 befinden, dienen als Grundlage für die Datenbeschreibung von Bauelementen in der Bibliothek. In den folgenden Kapiteln werden die angewandten Applikationsobjekte von AP210-ARM<sup>7</sup> bzw. die erweiterten, neu definierten Applikationsobjekte zur Beschreibung der Bauelemente dargestellt. Zur Programmierung wird jedes Applikationsobjekt von einer entsprechenden C++ Klasse repräsentiert.

#### Beschreibung der elektronischen Funktionalität – Funktionale Definition

Wie oben dargestellt, ist das in AP210 definierte Applikationsobjekt *Functional\_unit\_usage\_view* zur Beschreibung der elektronischen Funktionalität von großer Bedeutung. Das Applikationsobjekt *Functional\_unit\_usage\_view\_terminal\_definition* beschreibt je-

---

<sup>7</sup> Alle in der Dissertation vorgestellten AP210-Applikationsobjekte sind in ARM (nicht in AIM - Application Interpreted Model) definiert.



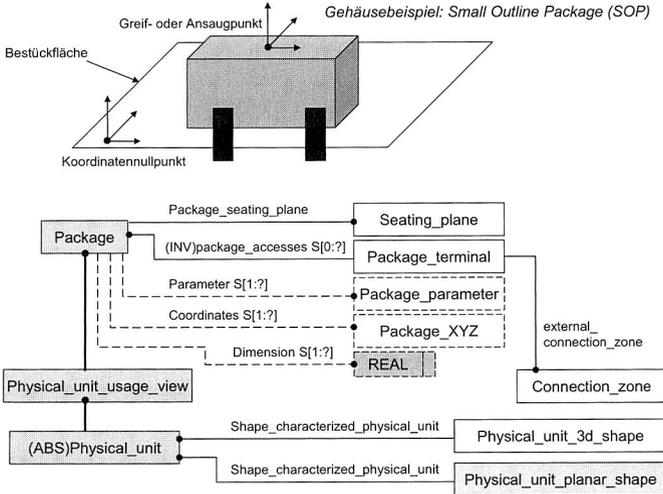


Bild 58: Wichtige Objekte und ihre Beziehungen in der Definition der Gehäuse in AP210 und Erweiterung durch zusätzliche Objekte und Beziehungen

**Beschreibung der gesamten Bauelementmodelle**

Unter „Gehäuse“ versteht man einen Behälter mit einer rein mechanischen Struktur ohne elektronische Funktion. Im Gegensatz dazu versteht man unter „Bauelement“ in der Bibliothek das echte elektronische 3D-Teil, das ein Gehäuse benutzt und eine elektronische Funktion besitzt. Das bedeutet, dass das Bauelement aus physikalischer Sicht ein geometrisches Gehäuse und gleichzeitig aus funktionaler Sicht ein elektronisches Funktionselement ist.

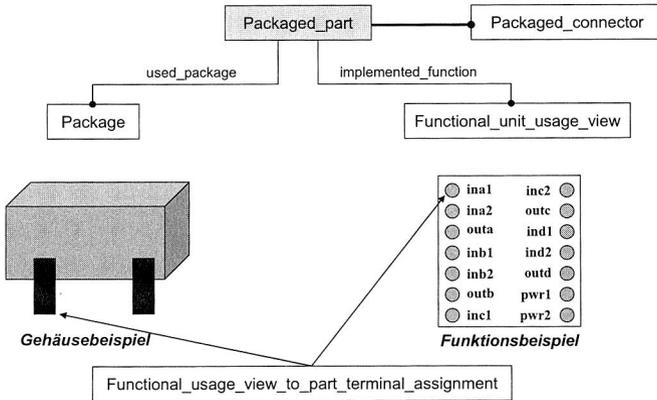


Bild 59: Definition der Bauelemente in der Sichtenebene

Während das Gehäuse und die elektronische Funktionalität durch die Applikationsobjekte *Package* und *Functional\_unit\_usage\_view* in AP210 beschrieben werden, können die Bauelemente durch das Applikationsobjekt *Packaged\_part* dargestellt werden. Bild 59 zeigt die Beziehung zwischen den drei Applikationsobjekten *Package*, *Functional\_unit\_usage\_view* und *Packaged\_part*. Das Objekt *Packaged\_part* enthält zwei Eigenschaften: *used\_package* und *implemented\_function*. Die Eigenschaft *used\_package* ist eine Instanz des Objekts *Package*, das das verwendete Gehäuse bzw. die geometrischen Informationen beschreibt. Die andere Eigenschaft *implemented\_function* ist eine Instanz des Objekts *Functional\_unit\_usage\_view*, das die elektronische Funktion des Bauelements definiert.

Durch die beiden oben genannten Partialmodelle *Functional\_unit\_usage\_view* und *Package* ist jedoch noch nicht genug Information zur Beschreibung der Bauelemente gegeben. Die Verbindung bzw. Beziehung zwischen beiden Partialmodellen ist noch nicht hergestellt. Dafür ist die Information „*Pin Map*“ verantwortlich, die die Verhältnisse zwischen physikalischen Anschlussteilen und logischen Anschlusselementen definiert. Ohne diese Information kann das Bauelement in der integrierten elektromechanischen Konstruktion im MIDCAD-System nicht eingesetzt werden.

Für die Beschreibung der Information „*Pin Map*“ ist in AP210 das Applikationsobjekt *Functional\_usage\_view\_to\_part\_terminal\_assignment* zuständig, das das Objekt *Packaged\_part\_terminal* mit dem Objekt *Functional\_unit\_usage\_view\_terminal\_definition* verknüpft. Das Schema und die Beziehungen zwischen relevanten Objekten sind in Bild 60 dargestellt.

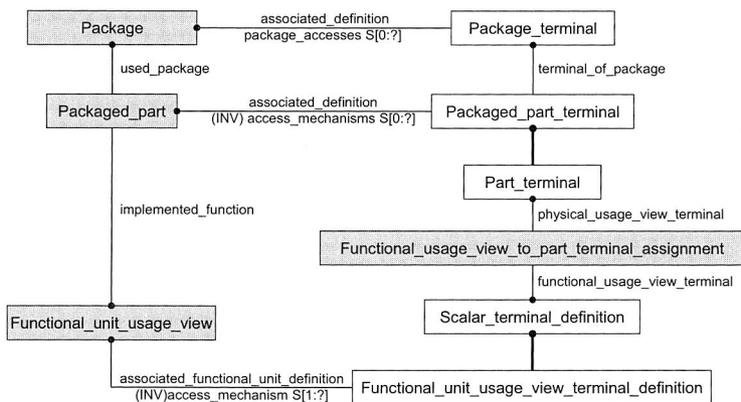


Bild 60: Wichtige Objekte und ihre Beziehungen in der Definition der Bauelemente und der *Pin Map*

Das Datenmodell und die Beschreibungsmethode von Bauelementen in der Bibliothek dienen als Grundlage für die Beschreibung der mechanisch-elektronisch integrierten Produktmodelle in Kapitel 6.4.

### 6.3.3 Bauelementemodellgenerator

Das Geometriemodell bzw. die zusätzlichen geometrischen Eigenschaften können direkt mit Pro/ENGINEER parametrisch modelliert werden. Pro/ENGINEER ist jedoch nicht in der Lage, die benötigten elektronischen Funktionen und die „Pin Map“ für die Bauelemente weiter zu definieren. In Hinblick auf die oben genannte Datenbeschreibung müssen diese Beschränkungen überwunden werden, um das Potential einer dreidimensionalen Bauelementebibliothek für die MID-Produktentwicklung voll nutzen zu können. Daher wurde ein Bauelementemodellgenerator in MIDCAD entwickelt. Der entwickelte Modellgenerator besitzt die folgenden Hauptfunktionen:

- Neue elektronische Funktionen definieren und in die Funktionsbibliothek einfügen,
- neue Gehäuse definieren und in die 3D-Gehäusebibliothek einfügen,
- neue Bauelemente definieren und in die 3D-Bauelementebibliothek einfügen,
- die „Pin Map“-Information in den Bauelementen definieren,
- die definierten Bauelemente in der Bibliothek editieren,
- die Mapping-Datei zwischen MIDCAD und 2D-ECAD editieren,
- die zusätzlichen Informationen von Bauelementen definieren etc.

1. Allgemeine Definition des neuen Bauelements

2. Physikalische Definition des neuen Bauelements

4. „Pin Map“-Definition des neuen Bauelements

Nr. in Platz	Signal Name	Nr. in Package
1	no name	1
2	no name	2

3. Funktionale Definition des neuen Bauelements

Bild 61 Hauptschritte bei der Erstellung neuer Bauelemente im Modellgenerator

Die wichtigste Aufgabe des Modellgenerators ist die Erstellung neuer Bauelemente, um die stetige Erweiterbarkeit und Anpassungsfähigkeit der 3D-Bauelementebibliothek in MIDCAD sicherzustellen. Über den Assistenten kann nicht nur zur funktionalen Definition des neuen Bauelements die in der Funktionsbibliothek vordefinierte elektronische Funktionalität ausgewählt, sondern auch die „*Pin Map*“-Information des neuen Bauelements definiert werden. Auch die parametrisch-geometrischen Dimensionen und Gehäuseeigenschaften in Familientabellen, wie z. B. die Anzahl der Gehäusepins, lassen sich durch den Assistenten festlegen. Dadurch können elektronische Bauelemente, die auf dem gleichen Gehäusemodell basieren, schnell und einfach erstellt und der Bibliothek hinzugefügt werden. Bild 61 zeigt die Hauptschritte bei der Erstellung eines neuen Bauelements durch den Modellgenerator.

Darüber hinaus ermöglicht der Modellgenerator dem Konstrukteur durch die Verwendung eines Assistenten die Erstellung neuer elektronischer Funktionen in der Funktionsbibliothek. Durch das Dialogfenster im Assistenten können die Anzahl und die Namen jedes Anschlusselements des neuen elektronischen Funktionselements definiert werden. Das in Pro/ENGINEER erzeugte parametrische Geometriemodell mit Familientabelle kann durch den Modellgenerator als neuer Gehäusotyp definiert und in die Gehäusebibliothek eingefügt werden. Der Modellgenerator kontrolliert auch die nötigen geometrischen Informationen, wie z. B. die Bestückfläche.

Als eine gute Erweiterungsmöglichkeit können die zusätzlichen Dialoge nach Bedarf im Assistenten weiter eingefügt werden. Damit können zusätzliche Informationen im Generator definiert und editiert werden, wie z. B. mechanische und thermische Eigenschaften und Mappinginformationen, die die Verknüpfung zwischen dem internen Namen des 3D-Bauelements mit dem entsprechenden Namen in verschiedenen ECAD-Systemen herstellen.

Das 3D-Geometriemodell bzw. die anderen geometrischen Eigenschaften können automatisch in der Pro/ENGINEER-Datei gespeichert und wieder verwendet werden. Neben den geometrischen Informationen sind andere nichtgeometrische Informationen, wie z. B. *Pin Map* und elektronische Daten, zu speichern. Durch die Verwendung der Technik *data classes and data slots* von Pro/TOOLKIT wurde eine interne Schnittstelle entwickelt. Damit können die nichtgeometrischen Informationen des Bauelements als äußere Daten in einer entsprechenden Pro/ENGINEER-Datei zusätzlich gespeichert und zu einem späteren Zeitpunkt wieder in MIDCAD eingelesen werden.

## **6.4 STEP AP210 als Grundlage für das mechanisch-elektronisch integrierte MID-Produktmodell**

Zur Beschreibung der horizontalen Integration domänenspezifischer Daten in das MID-Produktmodell dient das Schema von AP210 als Grundlage (siehe Bild 54). Der aktuelle Stand von AP210 ermöglicht es, ein mechanisch-elektronisch integriertes Datenmodell für den Entwurf von MID-Produkten zu realisieren.

Mit Hilfe der MID-Technologie wird innerhalb eines Produkts die stoffliche Integration von mechanischen und elektronischen Funktionen realisiert. Die Beschreibungen des elektronischen Partialmodells, des mechanischen Partialmodells und der Produktstruktur auf Basis von AP210, sind mit relativ wenigen Schwierigkeiten verbunden, da es entsprechende Applikationsobjekte in AP210 gibt. Die Problematik liegt vielmehr darin, dass eine Erweiterung eines Schaltungsträgermodells (inklusive Leiterbahnen) auf den dreidimensionalen Raum notwendig ist, um die speziellen Strukturen wegen der stofflichen Integration von Elektronik und Mechanik zu beschreiben.

### 6.4.1 Beschreibung des integrierten Produkts und der Produktstruktur

Das Produktmodell in AP210 ist mehr als ein einfaches Modell, weil in AP210 Produkte bzw. Bauteile und Baugruppen im zeitlichen Verlauf in jeweils neuen Versionen abgelegt werden können. Für jede Version können auch viele Sichten bzw. Partialmodelle erzeugt werden, die nur relevante Daten beinhalten. Zur Beschreibung der mechanisch-elektronisch integrierten Produktdaten muss die Definition der Sichten für die Bereiche der Elektronik und Mechanik vorgenommen werden. Die Datenmodelle aus funktionaler Sicht sind in AP210 grundsätzlich für die Beschreibung der elektronischen Funktionalität zuständig, während die Datenmodelle aus physikalischer Sicht zur Beschreibung der mechanisch-geometrischen Informationen eingesetzt werden [76].

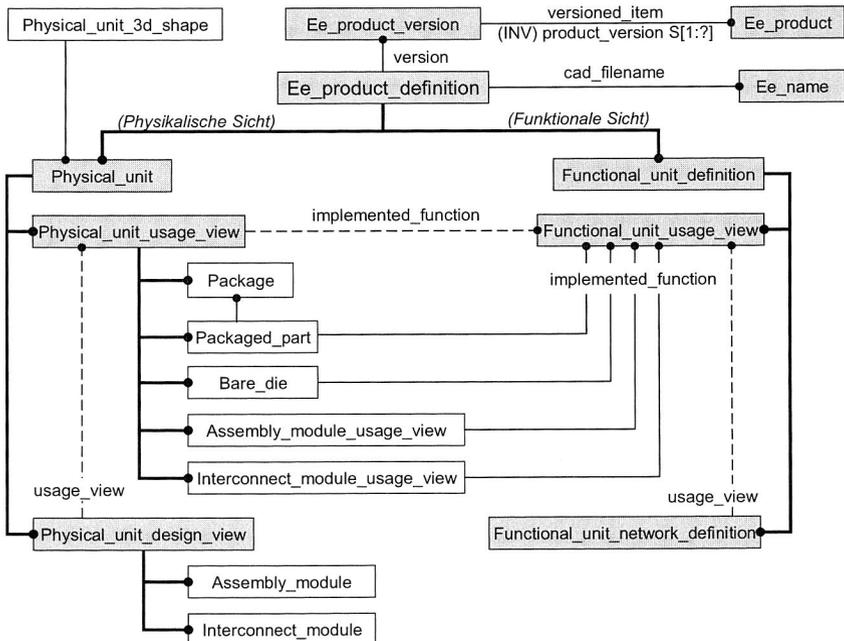


Bild 62: Definition der mechanisch-elektronisch integrierten Produkte

Die funktionale und die physikalische Sicht haben jeweils eine eigene Entwurfssicht (*design view*) und eine Anwendungssicht (*usage view*). Die relevanten Objekte und die Beziehungen zwischen der funktionalen und physikalischen Sicht bzw. zwischen der Entwurfssicht und der Anwendungssicht, sind in Bild 62 dargestellt.

Aus funktionaler Sicht werden die elektronischen Funktionalitäten hauptsächlich durch das Objekt *Functional\_unit\_usage\_view* beschrieben, das aus Anwendungssicht definiert ist. Die funktionale Entwurfssicht in AP210 ist eigentlich das Objekt *Functional\_unit\_network\_definition*, das die entsprechende elektronische Funktionalität aus Entwurfssicht intern strukturell definiert.

Aus physikalischer Sicht werden die geometrie-relevanten Informationen hauptsächlich durch das Objekt *Physical\_unit\_usage\_view* beschrieben, das aus Anwendungssicht definiert ist. Darunter können die Objekte *Packaged\_part* und *Package* die elektronischen Bauteile physikalisch beschreiben, die detaillierten Anwendungen und Beziehungen mit *Functional\_unit\_usage\_view* wurden bereits in Kapitel 6.3.2 dargestellt. Die anderen Objekte sind zuständig für die physikalische Beschreibung der ganzen Produkte bzw. anderer Bauteile und Baugruppen, wie z. B. *Bare\_die* für einige Typen der MID-spezifischen Features, *Interconnect\_module\_usage\_view* für den Schaltungsträger und *Assembly\_module\_usage\_view* für die ganze MID-Produktstruktur. Zur Beschreibung der physikalischen Strukturen des ganzen MID-Produkts und des entsprechenden dreidimensionalen Schaltungsträgers aus Entwurfssicht, werden die Objekte *Assembly\_module* und *Interconnect\_module* unter dem Objekt *Physical\_unit\_design\_view* benutzt (siehe Bild 62).

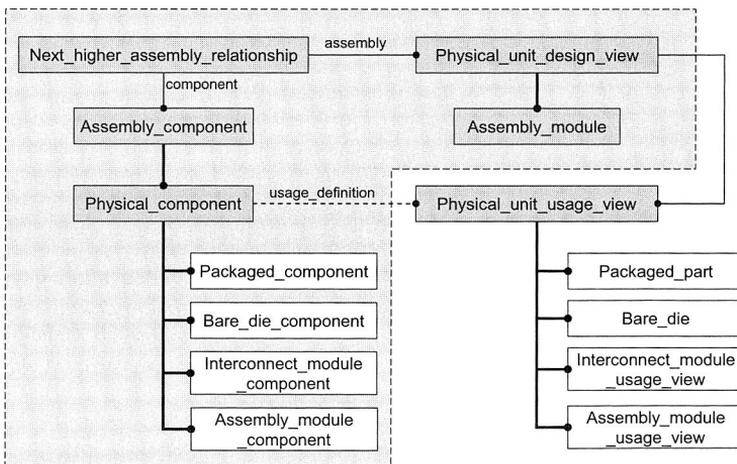


Bild 63: Wichtige Objekte in der Definition der Montagebeziehungen

Wie bereits oben erwähnt, wird das Objekt *Assembly\_module* angewandt, um die Hierarchie und Struktur der Produkte zu beschreiben. Die Bauteile und Baugruppen des

Produkts aus Montagesicht (*assembly view*) sind die Komponenten, die durch das Objekt *Assembly\_component* dargestellt werden. Dafür ist zum Beispiel das Objekt *Packaged\_component* für die Beschreibung der montage-relevanten Informationen der elektronischen Bauelemente zuständig. Die Beziehung der Montagehierarchie wird durch das Objekt *Next\_highter\_assembly\_relationship* beschrieben. Die anderen relevanten Objekte und ihre Beziehungen sind in Bild 63 dargestellt.

Neben der Definition der Montagebeziehung müssen die Verbindungen bzw. Beziehungen mit den entsprechenden physikalischen und funktionalen Definitionen dargestellt werden. In Bild 64 sind die relevanten Objekte für die Beschreibung der elektronischen Bauelemente aufgezeigt. Ein ähnliches Schema ist für andere Typen von Bauteilen und Baugruppen anzuwenden. Damit können die vollständigen Informationen aus Elektronik und Mechanik in entsprechende Datenmodelle integriert werden.

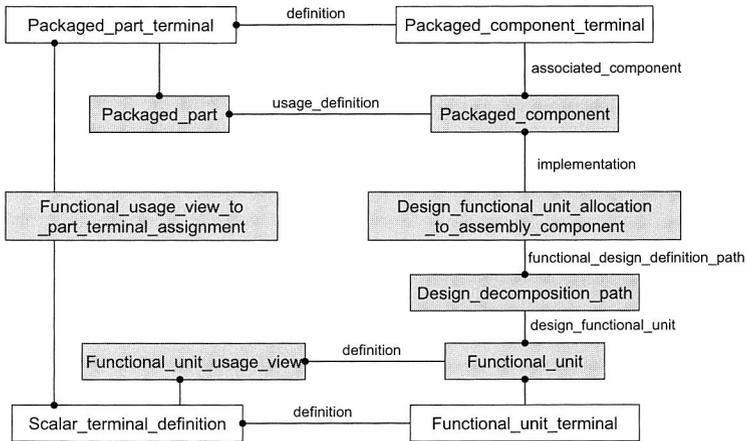


Bild 64: Definition der Beziehungen zwischen funktionaler und physikalischer Sicht

Die meisten montage-relevanten Daten (z. B. die Position der Bauteile) sind zwar durch die in Pro/TOOLKIT angebotenen Funktionen erhältlich, aber einige nötige Montagedaten müssen noch in entsprechenden Objekten als deren Eigenschaften gespeichert werden. Zum Beispiel sind für *Packaged\_component* und *Bare\_die component* die Informationen, ob das Bauteil bereits platziert ist, und die entsprechende Platzierungsfläche nötig. Eine wichtige Eigenschaft, die alle Objekte haben müssen, ist die innere Identität der verweisenden Komponenten in Pro/ENGINEER.

### 6.4.2 Beschreibung von Information über die elektronische Konnektivität

Die Information des Schaltplans kann in die Netz- und Komponentenlisten geteilt werden. Die Netzliste enthält die Information über die elektronische Konnektivität zwischen den einzelnen Bauelementen, während die Komponentenliste die elektronischen Bau-

elemente in einem Schaltplan spezifiziert. Die Beschreibung des Schaltplans wird hauptsächlich aus funktionaler Sicht durchgeführt.

Aus funktionaler Sicht können die elektronischen Komponenten in der Komponentenliste durch das Applikationsobjekt *Functional\_unit* beschrieben werden. *Functional\_unit* ist eine Instanz von *Functional\_unit\_usage\_view*. Für die Beschreibung der elektronischen Komponenten aus physikalischer Sicht sind die Objekte *Packaged\_component* und *Packaged\_part* zuständig. Die Beziehung zwischen *Functional\_unit\_usage\_view* und *Functional\_unit* ist in Bild 57 und Bild 64 aufgezeigt.

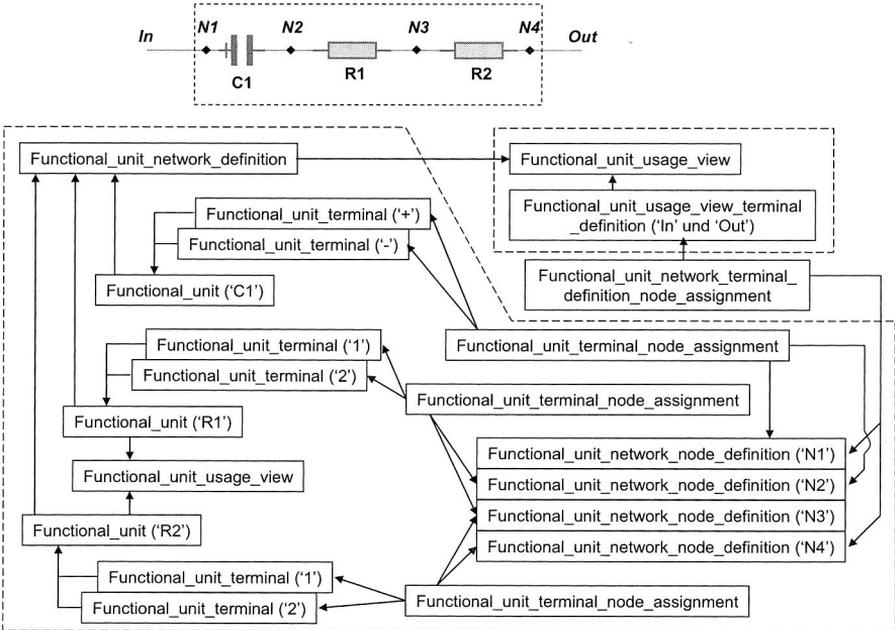


Bild 65: Beispiel zur Beschreibung der logischen Verbindung

Aus funktionaler Sicht werden die elektronischen Verbindungen durch das Applikationsobjekt *Functional\_unit\_network\_definition* repräsentiert. Die Beschreibung der physikalischen Konnektivität wird später in Kapitel 6.4.3 zusammen mit der Beschreibung der 3D-Leiterbahnen und des Schaltungsträgers dargestellt. Jedes Netz in der Netzliste entspricht einem Knoten im gesamten elektronischen Netzwerk und wird durch das Applikationsobjekt *Functional\_unit\_network\_node\_definition* repräsentiert. Deshalb wird der Schaltplan in AP210 hauptsächlich durch die drei oben genannten Applikationsobjekte *Functional\_unit\_network\_definition*, *Functional\_unit\_network\_node\_definition* und *Functional\_unit* beschrieben. Die konkret miteinander verbundenen Anschlusselemente in einem Netz werden durch das Applikationsobjekt *Functional\_unit\_terminal\_node\_*



Da der räumliche Schaltungsträger die mechanische Struktur und elektronische Funktionalität stofflich integriert, besteht dieser hauptsächlich aus zwei Teilen: Während ein Teil eine rein mechanisch-geometrische Struktur aufweist, ist der andere speziell und zusätzlich mit elektronischer Funktionalität verbunden (siehe Bild 67). Im Folgenden wird die Beschreibung der speziellen Strukturen diskutiert.

Die speziell auf elektronische Funktionalität ausgerichteten geometrischen Strukturen können in zwei Typen kategorisiert werden:

- *Typ I:* Dieser Typ verhält sich wie ein normales, elektronisches Bauteil, das im Schaltungsträger eingebettet ist. Hier sind Steckverbindungen, Batteriehalter und Durchkontaktierungen als Beispiel zu nennen. Deswegen werden die Objekte *Bare\_die* und *Bare\_die\_component* zur Beschreibung dieses Typs benutzt. Mit dem in Bild 64 dargestellten Schema können die Beziehung zwischen beiden Objekten und die Verbindung mit den Objekten *Functional\_unit* und *Functional\_unit\_usage\_view* beschrieben werden. Ein neues Objekt *Component\_feature* wird zur Erweiterung der Beschreibung dieses Typs definiert und mit dem Objekt *Bare\_die\_component* verbunden. Das neue Objekt *Component\_feature* entspricht dem Datenmodell zur Beschreibung der MID-spezifischen Features (siehe Kapitel 6.5).
- *Typ II:* Dieser Typ besitzt eine Funktion, die die auf dem Schaltungsträger platzierten elektronischen Bauelemente und die speziellen Strukturen vom *Typ I* elektronisch verbindet, wie z. B. die Geometrie der 3D-Leiterbahnen. Die Objekte *Stratum* und *Stratum\_feature* sind zur Beschreibung dieses Typs geeignet. Das Objekt *Stratum\_feature* wurde auch zur Beschreibung der MID-spezifischen Features erweitert.

Obwohl der Schaltungsträger nur ein einzelnes Bauteil (durch die stoffliche Integration von verschiedenen geometrischen Strukturen) ist, werden in MIDCAD bzw. in der Datenstruktur diese speziellen Bestandteile als getrennte Bauteile und der Schaltungsträger als Baugruppe behandelt. Die Hierarchie der Montage und die Beziehungen der Objekte sind in Bild 68 aufgezeigt.

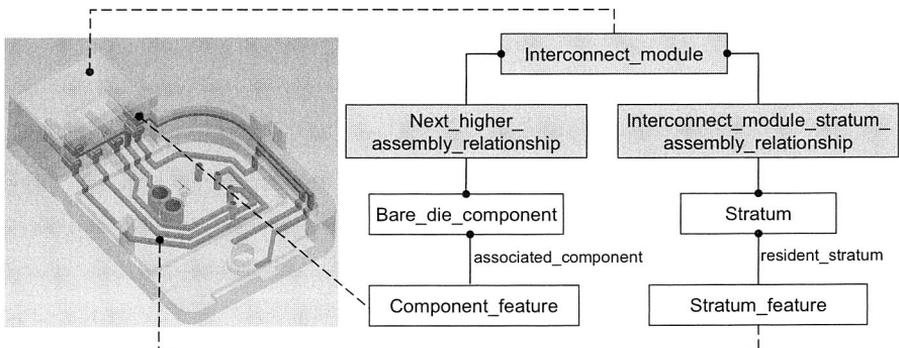


Bild 68: Definition der internen Montagebeziehung des Schaltungsträgers

Zu den speziellen Strukturen (*Typ II*) gehört die Geometrie der 3D-Leiterbahnen. Die Konstruktion und Modellierung der 3D-Leiterbahnen, basierend auf dem Schaltungsträger, sind eine wichtige Aufgabe beim 3D-MID-Design. Die 3D-Leiterbahnen sind die geometrische Darstellung der elektronischen Verbindungen. Deshalb wird die funktionale Definition der speziellen Strukturen aus Entwurfssicht durch die Objekte *Functional\_unit\_network\_node\_definition* und *Functional\_unit\_network\_definition* beschrieben, die schon in Kapitel 6.4.2 vorgestellt wurden. Die physikalische Definition der 3D-Leiterbahnen wird durch das Objekt *Physical\_network* beschrieben, das aus Entwurfssicht die physikalische Konnektivität in MID-Produkten definieren kann. Die Beziehung zwischen *Functional\_unit\_network\_node\_definition* und *Physical\_network* wird auch beschrieben. Damit kann die funktionale und physikalische Beschreibung der 3D-Leiterbahnen komplett dargestellt werden.

Die Besonderheit von MID-Produkten ist, dass die Leiterbahnen direkt auf den Oberflächen oder im Inneren des Schaltungsträgers verlaufen. Deswegen hat die physikalische Konnektivität jeder Leiterbahn die folgenden drei Verbindungsbeziehungen, die den drei Objekten *Intra\_stratum\_join\_relationship*, *Inter\_stratum\_join\_relationship* und *Stratum\_embedded\_component\_join\_relationship* entsprechen können (siehe Bild 69 und Bild 70):

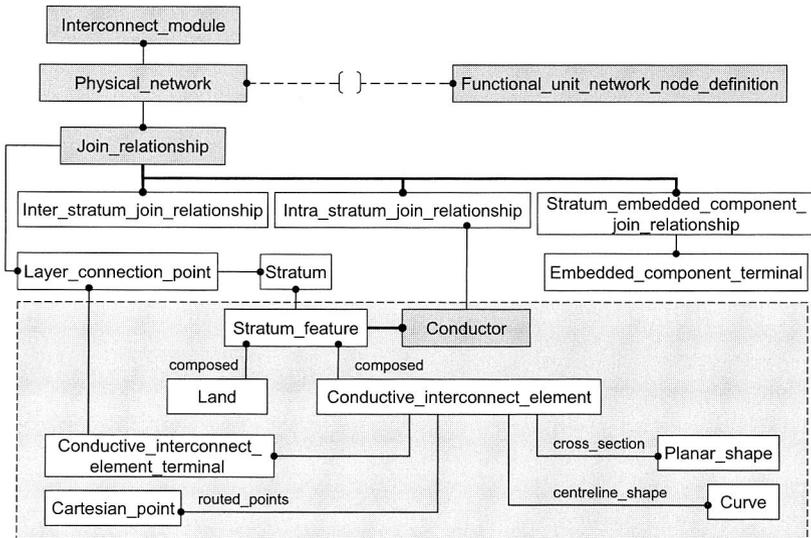


Bild 69: Wichtige Objekte und ihre Beziehungen in der Definition der 3D-Leiterbahnen

- Die erste Verbindungsbeziehung ist die Verbindung zwischen zwei Leiterbahnabschnitten, die zu einer Fläche gehören. Diese Beziehung wird durch das Objekt *Intra\_stratum\_join\_relationship* dargestellt. In AP210 repräsentiert der Begriff „Stratum“ normalerweise eine Ebene eines Multilayer-PCB. Aufgrund der Besonderheit

von MID-Produkten wird der Begriff „Stratum“ hier als Oberfläche des Schaltungsträgers betrachtet. Die Verbindungspunkte, deren Positionen aus dem Routingergebnis oder von den Lagen der Bauelementeanschlüsse ermittelt werden, werden durch das Objekt *Layer\_connection\_point* repräsentiert. In diesem Fall müssen alle Verbindungspunkte auch auf der gleichen Fläche liegen.

- Die zweite Verbindungsbeziehung ist die Verbindung zwischen zwei Leiterbahnabschnitten, die auf zwei benachbarten Flächen sind und durch einen Punkt an der gemeinsamen Kante zweier Flächen verbunden werden. Diese Beziehung wird durch das Objekt *Inter\_stratum\_join\_relationship* dargestellt. Der Verbindungspunkt an der gemeinsamen Kante wird durch das Objekt *Layer\_connection\_point* beschrieben.
- Die dritte Verbindungsbeziehung ist die Verbindung zwischen einem Leiterbahnabschnitt und einem MID-spezifischen Feature (*Typ I*, z. B. Steckverbindung, Durchkontaktierung) oder einem eingebetteten elektronischen Bauteil. Diese Beziehung wird durch das Objekt *Stratum\_embedded\_component\_join\_relationship* dargestellt. Der Verbindungspunkt und der entsprechende Anschluss werden durch die Objekte *Layer\_connection\_point* und *Embedded\_component\_terminal* beschrieben. Die Verbindung zwischen zwei Leiterbahnabschnitten über einer Durchkontaktierung kann durch diese Beziehung dargestellt werden, indem zwei Leiterbahnabschnitte separat mit zwei Anschlusspunkten der Durchkontaktierung auf zwei Flächen verbunden werden.

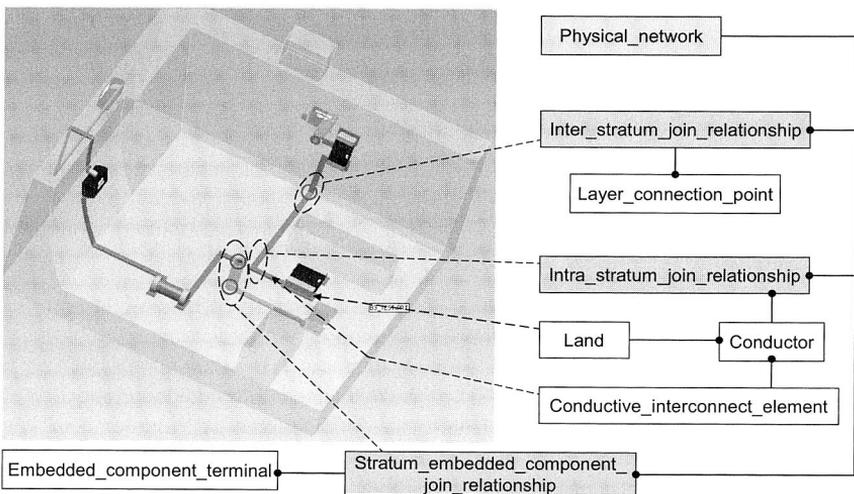


Bild 70: Beispiel zur Beschreibung der 3D-Leiterbahn

Durch die oben genannten Objekte können die Bestandteile der physikalischen Konnektivität der Leiterbahnen dargestellt werden. Die detaillierte Geometrie der Leiterbahn

muss auch im Datenmodell beschrieben werden. Die geometrische Beschreibung der Leiterbahnabschnitte auf einer Oberfläche (*Intra\_stratum\_join\_relationship*) ist die Grundlage für die Beschreibung der ganzen Leiterbahn.

Der Leiterbahnabschnitt auf einer Oberfläche entspricht dem Objekt *Conductor*, das die elektronische Verbindung zwischen den auf derselben Fläche stehenden Punkten implementiert. Das Objekt besteht wiederum aus den zwei Objekten *Land* und *Conductive\_interconnect\_element*. Das Objekt *Land* unterstützt die Beschreibung der Konnektivität zwischen dem Anschluss des elektronischen Bauteils und dem Verbindungspunkt der Leiterbahn, während das Objekt *Conductive\_interconnect\_element* den Leiterbahnabschnitt zwischen den Verbindungspunkten (*Layer\_connection\_point*) beschreibt. Die geometrische Form von *Land* ist von der Lage und dem Typ der platzierten Bauteile abhängig. Die Geometrie der 3D-Leiterbahnen ist dagegen hauptsächlich von dem Routingergebnis und der Querschnittsform der Leiterbahnen abhängig. Die zusätzlichen Eigenschaften des Objekts *Conductive\_interconnect\_element* beschreiben jeweils die Querschnittsform, die zentrale Linie (Trajektorie) und vom Routing definierte Punkte.

Durch die obige Vorstellung konnten die Beschreibung des elektronisch-mechanisch stofflich integrierten Schaltungsträgers sowie die Beschreibung der 3D-Leiterbahnen in MID-Produkten realisiert werden. Innerhalb des Produktmodells konnten die elektronischen und mechanischen Informationen durch die Verwendung und Erweiterung der Applikationsobjekte von AP210 erfolgreich integriert werden. Um weitere CAE/CAM Informationen in das Produktmodell zu integrieren, ist die Feature-Technologie zu benutzen, die im folgenden Kapitel vorgestellt wird.

## **6.5 Feature als Basis des CAD-CAE integrierten Produktmodells**

### **6.5.1 Feature-Technologie in CAx-Systemen**

Zur Integration oder zum Informationsaustausch sind geometrische und nichtgeometrische Informationen nötig (konventionell: Konstruktion geometrischer Grundkörper ohne Semantik). Die Einführung der Objekte als Träger der semantischen Informationen führt zur Verwendung von Features.

Die Feature-Technologie ist eine Methode zur Verknüpfung von Geometrie und nichtgeometrischen Informationen. Die Definition „Feature = Aggregation von Geometrie-elementen *und/oder* Semantik“ macht deutlich, dass durch die datentechnische Verbindung der Geometriestruktur mit den nichtgeometrischen Informationen, wie den berechnungs- und fertigungsrelevanten Daten, alle Produktdaten im Produktmodell integriert werden. Mit Semantik werden sämtliche Daten bezeichnet, die mit einer bestimmten Geometrie verknüpft werden. Die verknüpfte Semantik (nichtgeometrische Eigenschaften) ist vom Ziel abhängig, das mit dem Feature verfolgt werden soll, und kann so z. B. konstruktions-, berechnungs-, fertigungs-, montage- und qualitätsorientiert sein.

Features bieten die Möglichkeit, phasenübergreifend Informationen zu handhaben und weiterleiten zu können, d. h. Hintergrundinformationen zu Entwicklungsergebnissen zu erfassen und die im Produktentwicklungsprozess entstandenen Daten mit den in nachfolgenden Schritten benötigten Daten zu verknüpfen. Dies kann entweder dazu genutzt werden, Informationen für spätere Phasen (teil-)automatisch abzuleiten oder Aspekte der späteren Phasen in den Entwicklungsprozess einzubringen, wodurch sich in der Produktentwicklung Zeit, Kosten und Qualität positiv beeinflussen lassen. Dies ist ein entscheidender Schritt in Richtung Integration der an der Produktentwicklung beteiligten CAx-Systeme [88], [108].

Features, die in einem bestimmten Anwendungsbereich im Produktlebenszyklus effizient eingesetzt werden sollen, müssen gezielt auf dessen Belange zugeschnitten sein. Da die Repräsentation und Verarbeitung der relevanten Informationen in ihrer Gesamtheit sehr vielschichtig und schwierig ist, sind derzeit kaum Systeme vorhanden, die alle Belange des Produktlebenszyklus abdecken können. Vielmehr existieren eine Reihe feature-basierter Systeme für spezielle Anwendungsgebiete. Einige Anwendungsbereiche der Feature-Technologie werden im Folgenden kurz beschrieben.

- Die Hauptanwendung der Feature-Technologie ist die Geometriemodellierung. Hierfür reicht die Verwendung rein geometrie-orientierter Features (Form-Features) aus, die zurzeit in vielen kommerziellen CAx-Systemen eingesetzt werden. Grundlage sind Bibliotheken parametrisierter Form-Features, die benutzerspezifisch anpassbar sein müssen (Löschen, Manipulieren und Ergänzen von Features und Feature-Hierarchien).
- Ein weiteres Anwendungsfeld der Feature-Technologie in der Produktentwicklung ist das Abbilden funktionaler Zusammenhänge und zugeordneter Lösungsprinzipien im CAx-System entsprechend dem Voranschreiten im Konstruktionsprozess. Dies erfordert Features, mit denen sowohl die in den frühen Phasen wesentlichen Produkteigenschaften (Anforderungen, (Teil-)Funktionen, Lösungsprinzipien) als auch die im weiteren Verlauf des Entwicklungsprozesses (Entwerfen, Ausarbeiten) festgelegten Details (Geometrie, Technologie) beschrieben werden können.
- Eine weitere Möglichkeit des Einsatzes der Feature-Technologie liegt im Bereich der Berechnung und Optimierung, häufig mit direktem Bezug zur Finite-Elemente-Berechnung. Heute findet man meist eine unzureichende oder zum Teil sogar fehlende Kopplung von CAD und Berechnungs- bzw. Optimierungssoftware: Viele berechnungsrelevante Informationen sind im CAD-Modell nicht enthalten (z. B. Materialeigenschaften, Randbedingungen), andere Informationen müssen, obwohl prinzipiell vorhanden, häufig in einer ganz anderen Systemwelt erneut eingegeben werden. Mit Hilfe der Feature-Technologie können bereits bei der Konstruktion im CAD-System berechnungsrelevante Informationen mitgeführt werden, um sie bei einer nachfolgenden Berechnung zu nutzen. Durch Informationen über die Funktionen, die in den einzelnen Features enthalten sind, können beispielsweise für eine Berechnung unerhebliche Geometrielemente erkannt und ausgeblendet werden, was den

Einsatz automatischer Vernetzer für eine FEM-Berechnung erheblich beschleunigt und Vernetzungsfehler vermeidet.

- Die weitere Anwendung der Feature-Technologie ist die fertigungs-, montage-, demontage-, recycling- und prüfgerechte Produktgestaltung. Während der Konstruktion können mit Hilfe der Feature-Technologie Zusatzinformationen aus anderen Bereichen des Produktlebenszyklus zur Verfügung gestellt werden (z. B. Fertigungsverfahren, Maschinen, Werkzeuge, Standards usw.). In einem ersten Schritt können diese Informationen Grundlage für die Analyse des feature-basiert erstellten Produktmodells im Hinblick auf Kriterien wie fertigungs-, montage- oder prüfgerechte Gestaltung sein („*Design Checker*“). In einem zweiten Schritt kann man darauf aufbauend Systeme entwickeln, die auf der Basis der Feature-Information selbst Vorschläge zur Modifizierung des Produktmodells machen können („*Design Advisor*“).

Die detaillierte Vorstellung der oben dargestellten und weiteren Anwendungsfelder (wie z. B. Arbeitsplanerstellung, NC-Planung und Programmierung, Montageplanung usw.) der Feature-Technologie befinden sich in der VDI-Richtlinie 2218 [88].

Nach einigen Jahren intensiver Forschung im Bereich der Feature-Technologie werden heute nicht nur in den Forschungssystemen, sondern auch in den kommerziellen CAx-Systemen Features eingesetzt. Durch die Anwendung der Feature-Technologie können folgende Vorteile vereint werden:

- Datenhaltung von nichtgeometrischer Information zur Geometrie.
- Umfassende Produkt- und Prozessdarstellung in einem integrierten Datenmodell (durchgängige Semantik der Daten und Information über den gesamten Konstruktions- und Entwicklungsprozess, keine redundante Information).
- Verbesserung der Kommunikation
  - innerhalb des Produktentwicklungsprozesses (Produktdaten können jederzeit in den verschiedenen Phasen des Prozesses genutzt werden),
  - zwischen Benutzer und CAx-System (Kommunikation auf hohem Niveau),
  - zwischen verschiedenen CAx-Systemen für Konstruktion, Berechnung, Arbeitsvorbereitung und Fertigung.
- Verkürzung der Produktentwicklungszeit.

Bei der Konstruktion von MID-Produkten wird die Feature-Technologie nach dem derzeitigen Stand der Technik, d. h. entsprechend ihrer Realisierung in den kommerziellen verfügbaren CAD-Systemen, angewandt. Die hauptsächlichen Anwendungen der Feature-Technologie in MIDCAD sind die weitere geometrische Konstruktion bzw. Modellierung des 3D-Schaltungsträgers durch die stoffliche Integration elektronischer Funktion und die Unterstützung der Berechnung und Optimierung bei der Spritzgussimulation.

Die Bereitstellung der benutzerspezifischen Featurebibliothek in MIDCAD soll dabei eine wesentlich effektivere Konstruktion des 3D-Schaltungsträgers ermöglichen.

### 6.5.2 Erstellung einer Feature-Kategorie für räumliche spritzgegossene Schaltungsträger

Wie in den vorigen Abschnitten erläutert, erfordert die Anwendung der Feature-Technologie den Aufbau einer MIDCAD-internen Feature-Kategorie für räumliche spritzgegossene Schaltungsträger, um die Beschreibung und Modellierung des Schaltungsträgers umfassend zu unterstützen.

Die meisten räumlichen Schaltungsträger sind dünnwandige, thermoplastische spritzgegossene Bauteile. Die Kategorisierung und die Definition der allgemeinen Features für solche Spritzgieß-Teile sind bereits in vielen wissenschaftlichen Arbeiten (z. B. [2], [21], [22]) diskutiert worden. Basierend auf diesen Arbeiten wurden die allgemeinen Features für die Beschreibung der rein mechanischen Konstruktionselemente des 3D-Schaltungsträgers in Bild 71 hierarchisch definiert.

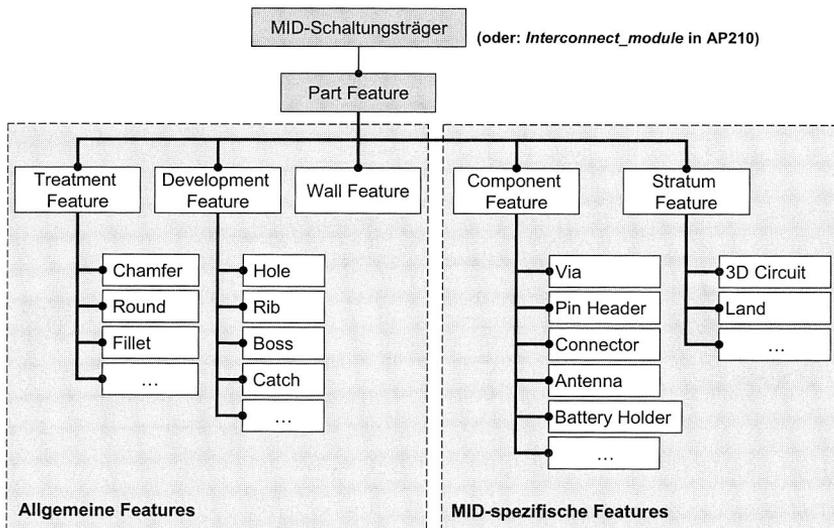


Bild 71: Feature-Kategorie für MID-Schaltungsträger

Die Typen des allgemeinen MID-Features sind im Folgenden vorgestellt:

- *Part Feature*: Das Feature repräsentiert nicht ein konkretes Konstruktionselement, sondern den ganzen Schaltungsträger. Es ist ein abstraktes Feature, das die allgemeinen Informationen des 3D-Schaltungsträgers beschreibt. Vor allem werden auf *Part Feature* die wichtigen analyserelevanten Informationen (wie z. B. Material, Randbedingungen etc.) für die nachfolgende Spritzgussimulation gespeichert.

- *Wall Feature*: Dieser Feature-Typ bildet die grundlegende, geometrische Struktur, „die dünnen Wände“, für die dünnwandigen thermoplastischen Schaltungsträger.
- *Development Feature*: Unter „Development“ versteht man, dass dieser Typ von Feature, basierend auf der grundlegenden Geometrie des Schaltungsträgers, weiter modelliert bzw. entwickelt ist, zum Beispiel *Hole*, *Rib*, *Boss*, *Catch* usw.
- *Treatment Feature*: Dieser Feature-Typ verfeinert die Geometriestruktur durch Fasenerzeugung, Verrundung usw.; dazu gehören *Chamfer*, *Round* und *Fillet*. Die Definition eines solchen Feature-Typs wird in modernen CAD/CAM-Systemen häufig verwendet.

Gegenüber den rein mechanisch-geometrischen Features (z. B. *Wall*, *Hole*, *Rib* usw.) für die dünnwandigen, thermoplastischen und spritzgegossenen Schaltungsträger zeichnen sich die MID-spezifischen Features durch die zusätzlich enthaltenen elektronischen Funktionen aus. Wie in Kapitel 6.4.3 erläutert, können die MID-spezifischen Features in die entsprechenden zwei Typen der speziellen geometrischen Strukturen (*Component feature* und *Stratum feature*) kategorisiert werden (siehe Bild 71). Einige Beispiele der MID-spezifischen Features enthält Bild 72.

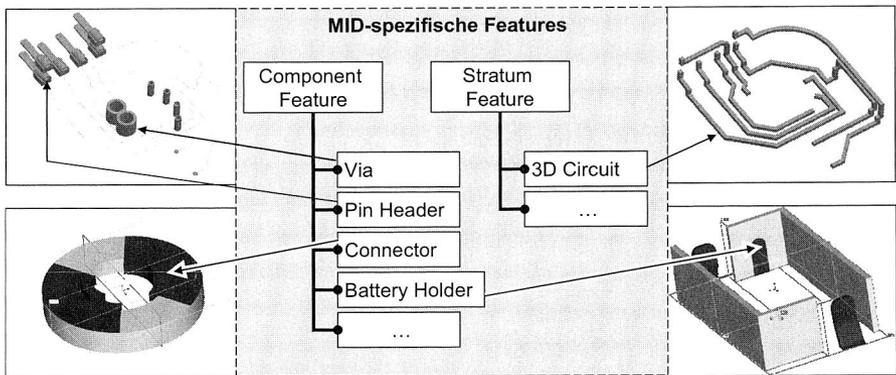


Bild 72: Einige Beispieldarstellungen der MID-spezifischen Features

Die Definition der zwei Typen der MID-spezifischen Features ist wie folgend dargestellt:

- *Component Feature*: Dieser Typ von Feature entspricht dem in Kapitel 6.4.3 dargestellten Applikationsobjekt *Component feature*, das den *Typ I* der speziellen geometrischen Strukturen darstellt. Dazu gehören Steckverbinder, Batteriehalter, Durchkontaktierung usw. Beispielsweise enthält das MID-spezifische Feature „Durchkontaktierung“ neben der geometrischen Struktur gleichzeitig die zusätzliche elektronische Funktion, d. h. an dieser Stelle kann eine stromleitende Verbindung zwischen zwei Oberflächen existieren. Auf diese Weise kann die Durchkontaktierung in das Routing miteinbezogen werden.

- *Stratum Feature*: Dieser Typ von Feature entspricht dem in Kapitel 6.4.3 dargestellten Applikationsobjekt *Stratum\_feature*, das den *Typ II* der speziellen geometrischen Strukturen darstellt. Dazu gehören hauptsächlich *3D-Circuit* und *Land*. Das MID-spezifische Feature *Land* entspricht dem Objekt *Land*, welches die Geometrie der elektronischen Konnektivität zwischen dem Anschluss des elektronischen Bauteils und dem Verbindungspunkt der Leiterbahn beschreibt. Das MID-spezifische Feature *3D-Circuit* entspricht dem Objekt *Conductive\_interconnect\_element*, welches die Geometrien der dreidimensionalen Leiterbahnabschnitte beschreibt.

### 6.5.3 Integration der CAE-Informationen in Features für die Spritzgussimulation

Die in Abschnitt 6.5.2 definierten MID-Features sind nicht nur einfache Form-Features, die aus strukturorientierten Gruppierungen geometrischer Elemente bestehen. Die MID-Features enthalten sowohl die Geometrie (geometrische Eigenschaften) als auch die nichtgeometrischen Eigenschaften (Semantik) (siehe Bild 73). Die nichtgeometrischen Eigenschaften können elektronikrelevant bzw. berechnungs-, fertigungs-, montage- und qualitätsorientiert sein. Mit Hilfe der MID-Features kann bereits beim 3D-MID-Design in MIDCAD die analyserelevante Information mitgeführt werden, um sie für eine nachfolgende Spritzgussimulation in MOLDFLOW nutzen zu können.

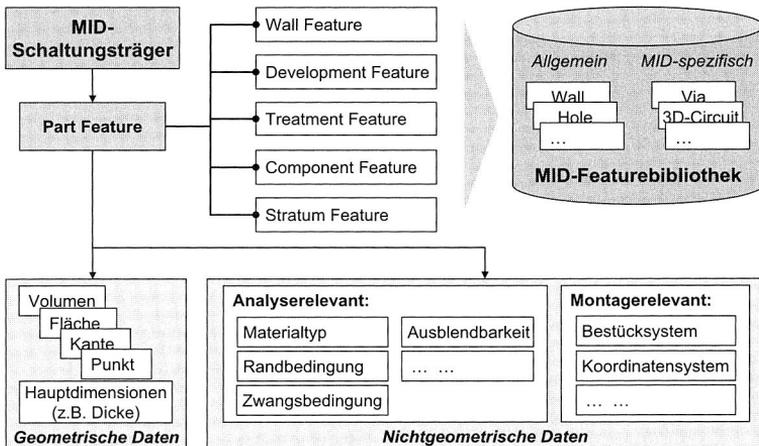


Bild 73: Verknüpfung der geometrischen und nichtgeometrischen Informationen in MID-Features

Wie in Tabelle 3 dargestellt, enthält das *Part Feature* nicht nur eine geometrische Eigenschaft „Identität des Schaltungsträger“, die auf das geometrische Modell des Schaltungsträgers verweist, sondern auch die meisten analyserelevanten Eigenschaften für die Spritzgussimulation: Materialdaten, Zwangsbedingungen, Typ der Analyse und Rand- bzw. Spritzgießparameter (wie z. B. Angusspunkte, Temperaturen der Schmelze und der Gießform, Einspritzdauer etc.).

Mit dem Feature *Wall* kann die analyserelevante Zwangsbedingung hinsichtlich der Lage der Angusspunkte (*Gate Location*) spezifiziert werden.

Eine spezielle analyserelevante Eigenschaft von Features ist die Ausblendbarkeit, die beinhaltet, ob die mit dem Feature verknüpften Geometrielemente für die Spritzguss-simulation unerheblich sind und ausgeblendet werden müssen. Dies beschleunigt den Einsatz automatischer Vernetzer für eine Spritzguss-simulation erheblich und vermeidet Vernetzungsfehler. Mit den Ausnahmen von *Part Feature* und *Wall* kann diese Eigen-schaft für die anderen MID-Features definiert werden.

Die Zuweisung zu einer bestimmten geometrischen Struktur für alle MID-Features wird durch das Einfügen der Eigenschaften (z. B. Identitätsgruppe der Geometrie) realisiert. Zur Unterstützung des parametrischen Modellierungsansatzes und der geometrischen Beschreibung werden in MID-Features die Hauptdimensionen bzw. wichtige Parameter als Eigenschaften gespeichert. Zwar ist Pro/ENGINEER für die geometrische Modellie-rung und die entsprechende boolesche Operationen zwischen den Features zuständig, aber die geometrischen Beziehungen mit anderen Features und die Abhängigkeit von anderen Features sollen auch in jedem Feature dargestellt werden. Diese Beziehungen und Abhängigkeiten werden durch die verschiedenen Zeiger (*Pointers*) als Eigenschaft dargestellt.

Tabelle 3: Eigenschaften in MID-Features

Feature-Typ	Hauptsächliche Eigenschaften in MID-Features
Part Feature	Feature-Identität, Identität des Schaltungsträgers, Hauptdimension: Dicke, Material, Analysetyp, Zwangs- bzw. Randbedingung, Spritzgießparameter etc.
Wall Feature	Feature-Identität, Identitätsgruppe der Geometrie, Beziehung zu anderen Features, Hauptdimensionen: Dicke, Zwangsbedingung etc.
Development Feature (Hole, Rib, Boss usw.)	Feature-Identität, Identitätsgruppe der Geometrie, Beziehung zu anderen Features, Hauptdimensionen: Durchmesser oder Dicke, Zwangsbedingung, Ausblendbarkeit etc.
Treatment Feature (Chamfer, Round usw.)	Feature-Identität, Identitätsgruppe der Geometrie, Beziehung zu anderen Features, Hauptdimensionen: Radius, Länge, Zwangsbedingung, Ausblendbarkeit etc.
Component Feature (Pin Header, Connector usw.)	Feature-Identität, Identität der Komponente, Verweisung zum Objekt in AP210, leitende Werkstoffe und Schichtaufbau, Beziehung zu anderen Features, Hauptdimensionen: Dicke, Durchmesser, Länge, Zwangsbedingung, Ausblendbarkeit etc.
Stratum Feature (3D-Circuit, Land usw.)	Feature-Identität, Identitätsgruppe der Geometrie, Verweisung zum Objekt leitende Werkstoffe und Schichtaufbau, Beziehung zu anderen Features, Hauptdimensionen: Breite, Länge, Höhe, Querschnitt, Zwangsbedingung, Ausblendbarkeit etc.

Neben den analyserelevanten Eigenschaften können mit Hilfe der MID-Features bei der Konstruktion in MIDCAD montageorientierte Informationen mitgeführt werden, um sie in der nachfolgenden Entwicklung der CAD/CAM-Kette zwischen MIDCAD und dem MID-Bestücksystem einzusetzen (siehe Kapitel 7.2.3).

### 6.5.4 Implementierung der Feature-Technologie in MIDCAD

Die Implementierung bzw. Anwendung der Feature-Technologie in CAx-Systemen hat allgemein drei Strategien: feature-basiertes Modellieren (*Design by Feature*), Feature-Erkennung (*Feature Recognition*) und Feature-Transformation (*Feature Mapping*) [88]. Vor allem wurde die Feature-Technologie mit der ersten Strategie, feature-basiertes Modellieren, in MIDCAD angewandt. Einerseits wird die Feature-Technologie zur vertikalen Integration der CAE/CAM-Daten im Produktmodell eingesetzt, andererseits kann zur stofflichen Integration elektronischer Funktionen die weitere geometrische Modellierung des Schaltungsträgers durch Einfügen der MID-spezifischen Features realisiert sowie die Strukturoptimierung auf Basis der Feature-Technologie durchgeführt werden.

Die vordefinierten Features, die alle drei Strategien benötigen, werden in der Regel in den so genannten Feature-Bibliotheken zusammengefasst. In MIDCAD wurde basierend auf der oben genannten Feature-Kategorie im Hintergrund auch eine entsprechende MID-spezifische Featurebibliothek entwickelt. Darüber hinaus werden die Funktionen bzw. Methoden zur Erzeugung vordefinierter und benutzerspezifischer Features (*User Defined Features*) benötigt, um solche Bibliotheken auf die speziellen MID-Produkte abstimmen zu können.

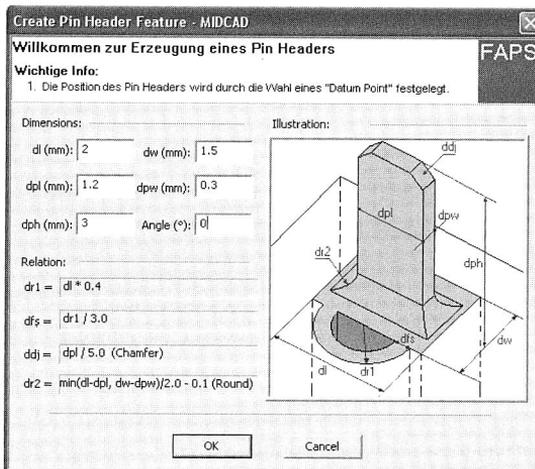


Bild 74: Beispiel zur Erzeugung des Features „Pin Header“

Nachdem die grundlegende Geometrie des Schaltungsträgers in MIDCAD bzw. Pro/ENGINEER direkt entworfen bzw. modelliert worden ist, können die Geometrien der

allgemeinen Features durch die Zuweisung der bestimmten geometrischen Konstruktionselemente definiert werden. Die Geometrie der speziellen *Component\_features* und einiger allgemeiner Features (z. B. *Catch*) kann durch die entwickelten Modellierungsfunktionen von MIDCAD automatisch auf dem Schaltungsträger erzeugt werden, indem die gewünschten Features aus der Featurebibliothek ausgewählt und im Anschluss an die entsprechende Dimensionierung und Lagebestimmung im Schaltungsträger platziert werden. Bild 74 zeigt ein Dialogfenster für die Erzeugung des MID-spezifischen Features *Pin Header*. Die Geometrie des MID-spezifischen Features *3D-Circuits* wird nach Vorgabe der Querschnitte und des Ergebnisses des 3D-Routings automatisch erzeugt. Da die Geometrie oder Kontur des speziellen Features *Land* von der Position und dem Typ der kontaktierten elektronischen Bauelemente abhängig ist, kann die Erzeugung und Positionierung der 3D-Geometrie des Features *Land* nach der Platzierung der Bauelemente durchgeführt werden. Nichtgeometrische Angaben sowie analyserelevante Eigenschaften und Zwangsbedingungen werden bereits während oder nach der Definition dieser Features durch eine grafische Benutzerschnittstelle spezifiziert.

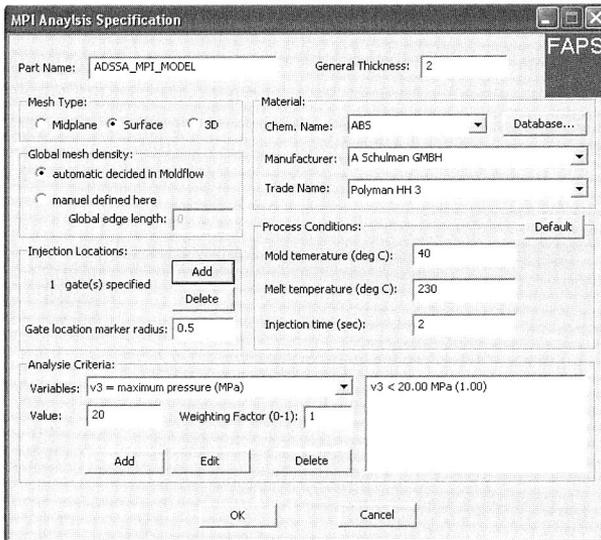


Bild 75: Spezifizierung der analyserelevanten Daten in MIDCAD

Nach den Definitionen aller Features können die Definitionen des abstrakten *Part Features* bzw. dessen Eigenschaften durchgeführt werden. Hier können die wichtigen, analyserelevanten Informationen (Randbedingung, Material, Spritzgießparameter usw.) und Kriterien für die nachfolgende Spritzgussimulation genau definiert werden (siehe Bild 75). Durch die Selektion eines oder mehrerer *datum points* auf dem Schaltungsträger kann die Lage der Angusspunkte interaktiv definiert und durch Kreise auf dem Schaltungsträger markiert werden.

Die allgemeinen und MID-spezifischen Features können durch einen selbst entwickelten Feature-Baum definiert und übersichtlich dargestellt werden (siehe Bild 76). Um auf die erzeugten MID-Features weiter zugreifen zu können, müssen die Features neben den Eigenschaften noch die inneren Behandlungsfunktionen besitzen. Die allgemeinen Operationen der Features sind Modifizierung, Spezifizierung, Löschen usw. Diese Behandlungsfunktionen werden durch den Feature-Baum aufgerufen. Zum Beispiel kann die Geometrie der Features aufgrund der Verwendung eines parametrischen Modellierungsansatzes im Nachhinein leicht modifiziert werden.

Nachdem alle nötigen Features bzw. *Part Feature* erzeugt oder definiert, und deren analyserelevanten Eigenschaften spezifiziert worden sind, können diese Features mit ihren Eigenschaften nach den bereits in Kapitel 6.4.3 dargestellten Schemen automatisch mit dem integrierten Produktmodell kombiniert werden. Das Produktmodell ist bisher sowohl elektronisch-mechanisch als auch CAD-CAE integriert.

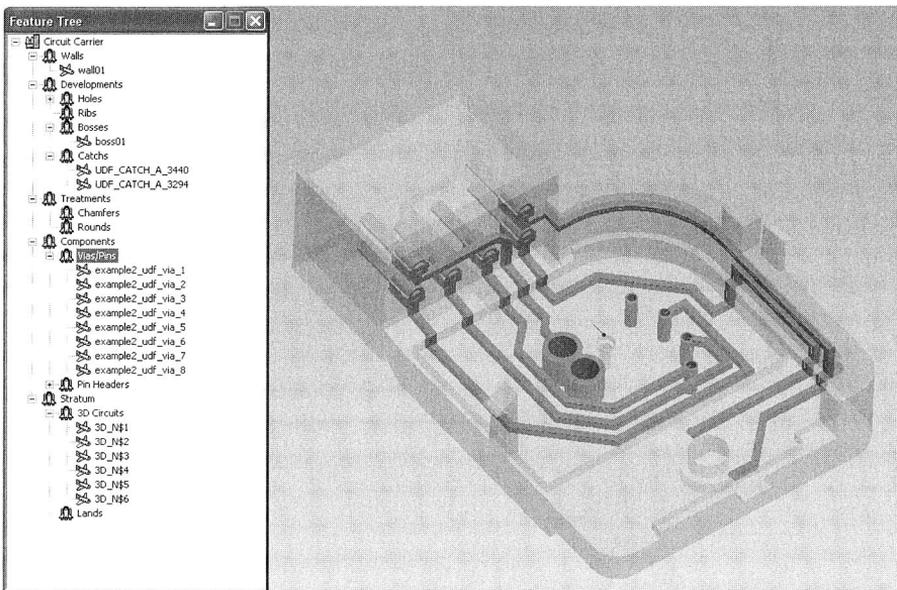


Bild 76: Feature-Baum in MIDCAD

# 7 Integration von CAE- und CAM-Werkzeugen für MID-Produkte

## 7.1 Integration der Simulation des Spritzgussprozesses

### 7.1.1 Konzept der Integration der Spritzgussimulation

Der Einkomponenten- oder Zweikomponentensspritzguss zählt zu den wichtigsten Verfahren bei der Herstellung von MID-Produkten. Deswegen spielt die Simulation des Spritzgussprozesses eine wesentliche Rolle, um die geforderte Qualität und Spritzgießbarkeit (*mouldability*) des Schaltungsträgers sicherzustellen. Mit Hilfe der Spritzgussimulation kann bereits in einer frühen Entwicklungsphase eine Verbesserung der MID-Produktqualität und die Rationalität der Produktstruktur erreicht werden. Durch die Spritzgussimulation können beispielsweise die Geometriemodelle des ersten und des zweiten Schusses für das 2K-Verfahren optimiert werden.

Für eine Integration der Spritzgussimulation gibt es zwei notwendige Voraussetzungen: Zum einen das integrierte Produktmodell, das die zur Durchführung der Simulation benötigten Produktdaten und -eigenschaften (z. B. Rand- bzw. Spritzgießparameter, Materialdaten) enthält und zum anderen die Verbindung zwischen dem MIDCAD-System und dem Simulationswerkzeug (MOLDFLOW). Die erste Voraussetzung wurde bereits in Kapitel 6.5 in Zusammenhang mit der Feature-Technologie erläutert. In diesem Kapitel werden die zweite Voraussetzung und die dabei relevanten Technologien vorgestellt.

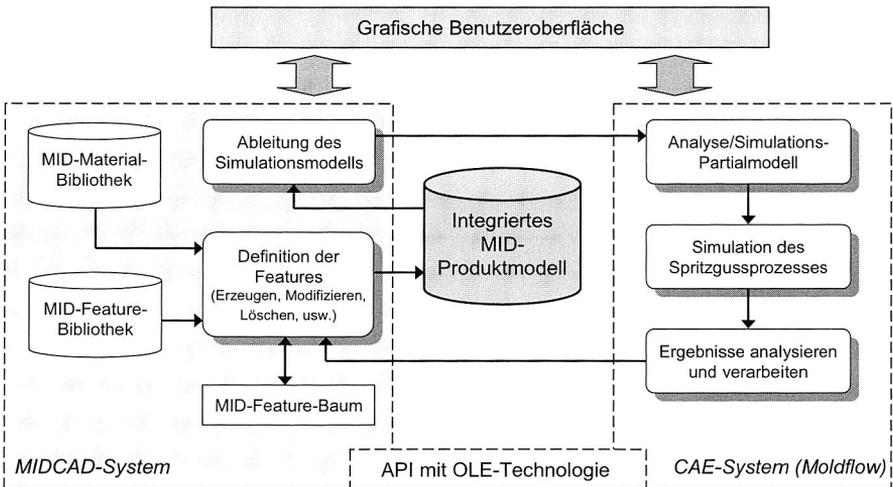


Bild 77: Konzept der Integration der Spritzgussimulation

Bild 77 zeigt die Struktur und die wichtigen Module zur Integration der Spritzgussimulation, wie z. B. das integrierte Produktmodell, die feature-relevanten Module, die MID-Materialbibliothek, die Ableitung des Simulationsmodells, die API mit OLE-Technologie für MOLDFLOW und die analyserelevanten Module in MOLDFLOW. Mit Hilfe der Feature-Technologie können die analyse-relevanten Informationen im Produktmodell mitgeführt werden. Die neu entwickelte Materialbibliothek ist speziell für MID-Produkte erstellt worden, bei der nur die Daten der in MID angewandten, thermoplastischen Materialien gespeichert werden. Das Analyse- bzw. Simulationsmodell mit entsprechenden Geometrie- und Nichtgeometrieinformationen kann aus dem integrierten Produktmodell unter Verwendung der entwickelten Schnittstelle abgeleitet und im CAE-System exportiert werden. Um eine enge Verknüpfung zwischen MIDCAD und MOLDFLOW zu gewährleisten, soll die Programmierschnittstelle basierend auf der OLE-Technologie und den MFC-Klassen entwickelt werden. Dadurch hat MIDCAD direkten Zugriff auf die Analysefunktionen, die Einstellungsfunktionen der Randbedingungen und weitere Funktionen in MOLDFLOW. Eine Konvertierung der Daten sowie die dadurch auftretenden Schwierigkeiten können dadurch vermieden werden. Nach der Spritzgussimulation werden die Ergebnisse verarbeitet, analysiert und hinsichtlich vordefinierter Kriterien (z. B. Höchstdruck) untersucht. Wenn nötig, muss die Geometrie des Schaltungsträgers geändert werden, um eine optimale und fertigungsgerechte Struktur bereits in einer frühen Entwicklungsphase zu erreichen und das Konzept „*Design for Mouldability*“ aus dem „*Concurrent Engineering (CE)*“ zu unterstützen. Im Rahmen der vorliegenden Forschungsarbeit werden nur die integrationsrelevanten Technologien erforscht. Im Gegensatz dazu sind die analyserelevanten Technologien wie z. B. Analyse und Verarbeitung der Simulationsergebnisse, Einstellung und Untersuchung der Kriterien nicht in der Forschungsarbeit enthalten.

### 7.1.2 Direkte Verbindung zwischen MIDCAD und MOLDFLOW Plastics Insight

#### **MOLDFLOW Plastics Adviser (MPA) und MOLDFLOW Plastics Insight (MPI)**

Das CAE-System MOLDFLOW gilt als Weltmarktführer im Bereich der Spritzgießsimulation. Tausende von Firmen auf der ganzen Welt haben Millionen von Analysen mit den MOLDFLOW Konstruktionslösungen durchgeführt, zu denen MPA und MPI gehören [74].

Die MPA-Programme sind leicht zu erlernende, 3D-Solids-basierte Kunststoff-Füllsimulationsprogramme, mit denen Konstrukteure von Kunststoffformteilen und Spritzgussformen die Produzierbarkeit und die Qualität ihrer Konstruktionen früh bestimmen können, um dadurch mögliche Folgeprobleme zu vermeiden, die zu Verzögerungen und Mehrkosten führen könnten. Zwar hat Pro/ENGINEER eine Zusatzapplikation Pro/PlasticAdvisor, die den MPA-Programmen von MOLDFLOW entspricht. Die Verbindung dazwischen steht jedoch noch auf niedrigem Niveau, weil sie kein gemeinsames Produktmodell haben. Die analyserelevanten Daten, wie z. B. Materialtyp, Rand- und Spritzgießparameter, müssen getrennt in Pro/PlasticAdvisor definiert werden.

Die MPI-Programme bieten die größte Palette an Spritzgussprozesssimulationstools in der Kunststoffindustrie. Sie sind die weltweit führenden Simulationsprodukte, um Formteil- und Werkzeugdesign zu überprüfen. Die Analysefunktionen von MPI simulieren die Füll-, Nachdruck- und Kühlphasen beim Thermoplast-Spritzguss mit verschiedenen Werkstoffen und auch glasfaserverstärkten Kunststoffen. MPI unterstützt besonders die 2K-Spritzgussimulation. Wichtig ist, dass MPI eine Programmschnittstelle (API) [73] bietet, mit der der Anwender und der Partner in der Lage sind, eine Vielzahl der MPI Funktionen zu erweitern, zu automatisieren und auf ihre eigenen Bedürfnisse zuzuschneiden. Das auf OLE/COM-Technologie basierende MPI legt die Methoden und Eigenschaften seiner Objekte, für den Zugriff von einer anderen Software aus, offen. Mit Hilfe der API kann der Anwender außerdem eine Schnittstelle zu externen Programmen einbauen.

Deswegen wird im Rahmen dieses Arbeitspunkts MPI als Simulationstool ausgewählt und eine zusätzliche Schnittstelle zwischen MIDCAD und MPI auf Basis der OLE/COM-Technologie entwickelt, um die Integration der Spritzgussimulation zu realisieren.

### **OLE-Automation**

Die OLE/COM-Technologie, die die Basis aller Microsoft-Programme ist, bietet die Möglichkeit, Daten programmübergreifend auszutauschen. Einer der Hauptvorteile von OLE/COM ist, dass verschiedene Software-Anwendungen integriert werden können. Dabei erfolgt keine Datenübersetzung. OLE/COM ermöglicht den Zugriff auf Daten, aber nicht deren Austausch.

Die Automatisierung (Automation oder OLE-Automation) bietet eine Möglichkeit, dass ein Programm seine Funktionen einem anderen zur Verfügung stellt. Anwendungen können den Zugriff auf ihre Funktionen durch die in ihren COM-Objekten dargestellten Schnittstellen ermöglichen. Jede Software, die in der Lage ist, Methoden eines COM-Objekts aufzurufen, kann anschließend diese Funktionen verwenden. Es lassen sich Programme schreiben, die alles, was die Benutzer manuell tun, automatisch erledigen. Diese Art allgemeiner Programmierbarkeit unter Verwendung von COM wird OLE-Automation oder Automatisierung genannt.

Die OLE-Automation besteht immer aus einem *Server* und einem *Client*. Mit Hilfe der Automatisierung sind in vielen Anwendungen (*Server*) ihre Funktionen für *Clients* verfügbar. Mit Hilfe von relativ einfachen, graphisch orientierten Programmierwerkzeugen (Visual Basic, Skriptsprachen etc.) können z. B. OFFICE-Anwendungen untereinander automatisiert verbunden werden, d. h. eine Anwendung greift auf die Funktionalität einer weiteren Applikation zu [77].

Alle COM-Objekte, die automatisiert werden können, besitzen eine Standardschnittstelle, die *IDispatch* genannt wird. Mit der Standardschnittstelle *IDispatch* werden die internen Eigenschaften und Methoden eines COM-Objekts offen gelegt und von einem *Client* benutzt. Auf diese öffentlichen Eigenschaften und Methoden eines COM-Objekts

kann auf zwei Arten zugegriffen werden: Zum einen mit der Hilfe einer so genannten Verteilerschnittstelle (*dispinterface*), und zum anderen mit dem so genannten *vtable*, d. h. einer virtuellen Tabelle von Funktionszeigern, die die Zeiger auf die Methoden der Schnittstelle enthält. Mit der Methode *Invoke* lässt sich eine beliebige Methode aus einer Gruppe von Methoden aufrufen und ihr alle erforderlichen Parameter übergeben. Vor dem Aufruf *IDispatch::Invoke* muss jedoch die bestimmte Verteiler-ID (*DispID*) der aufzurufenden Methode durch die Methode *GetIDsOfNames* bestimmt und genau angegeben werden [100]. Ein Beispiel ist in Bild 78 gezeigt.

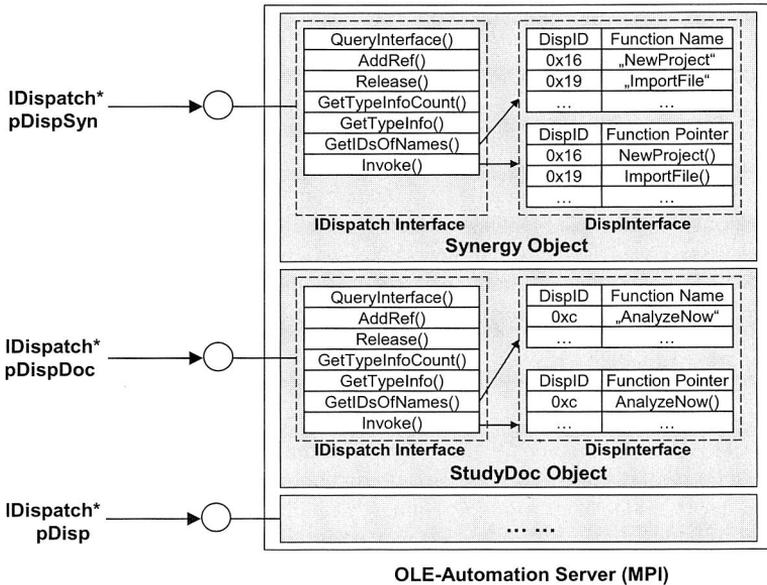


Bild 78: Schnittstelle und Repräsentation der COM-Objekte in MPI

### Zusätzliche MPI-Programmschnittstelle basierend auf MFC

MPI ist jetzt ein OLE-Automation *Server*, aber zur Anwendung der Funktionen von MPI in MIDCAD (MIDCAD als *Client*) gibt es noch folgende Schwierigkeiten:

Zum einen kann die in MPI angebotene Programmschnittstelle, wie die API in den meisten OLE-*Servers* (z. B. OFFICE-Anwendungen), direkt von Programmierwerkzeugen (z. B. VB, VBA, Skriptsprachen etc.) benutzt werden. Wenn aber VC++ als Programmierwerkzeug für die Entwicklung eines *Clients* verwendet wird, muss die Syntax der Funktionen in der Programmschnittstelle übersetzt werden, um sie der Programmiersprache C++ anzupassen. Da MIDCAD basierend auf Pro/ENGINEER und mit VC++ und Pro/TOOLKIT entwickelt wird, ist die Aufgabe der Übersetzung notwendig, um MIDCAD als OLE-Automation *Client* weiter entwickeln zu können.

Zum anderen ist die Information über die Funktionalität der COM-Objekte in der Regel in einer so genannten Typenbibliothek (oder: Objektbibliothek) gespeichert, die eine spezielle Datei oder ein Teil von einer speziellen Datei ist. So enthält beispielhaft die WORD-Anwendung die Datei *Msword9.olb* (für die Version WORD 2000), die für *Clients* die Informationen über die öffentlichen Methoden und Eigenschaften der COM-Objekte in WORD spezifiziert. Unter Verwendung der Typenbibliothek und MFC kann eine zusätzliche Schnittstelle für die Programmiersprache C++ automatisch erzeugt werden. Da MPI diese Typenbibliothek nicht angeboten hat, ist die zusätzliche Arbeit notwendig, um die Informationen (insbesondere: *DispID*) der COM-Objekte in MPI zu bekommen.

MFC bietet eine Klasse *COleDispatchDriver* an, die für die Behandlung der Standard-schnittstellen *IDispatch* der COM-Objekte zuständig ist und den Vorgang der Übersetzung vereinfacht. Wenn die Typenbibliothek existiert, werden die entsprechenden *wrapper classes* von jedem COM-Objekt durch den Visual C++ Classwizard automatisch generiert. Die automatisch erzeugten *wrapper classes*, die von der Klasse *COleDispatchDriver* abstammen, enthalten die Methoden, die den öffentlichen Eigenschaften und Methoden eines COM-Objekts entsprechen. Einige wichtige Funktionen werden von der Klasse *COleDispatchDriver* unterstützt, um mit dem OLE-Automation *Server* zu kommunizieren. Zum Beispiel entspricht die Methode *COleDispatchDriver::InvokeHelper* der Methode *IDispatch::Invoke*; die Methode *COleDispatchDriver::CreateDispatch* kann eine neue Instanz im *Server* erzeugen und das *COleDispatchDriver* Objekt mit der entsprechenden Schnittstelle *IDispatch* verbinden.

Da MPI die entsprechende Typenbibliothek nicht angeboten hat, können die *wrapper classes* der COM-Objekte von MPI nur manuell erzeugt werden. Vor der Erzeugung der *wrapper classes* müssen die genauen Informationen über die Methoden und Eigenschaften der COM-Objekte, wie z. B. Namen, *DispID*, Datentyp der Parameter und Ergebnisse usw. bekannt sein. Im Folgenden wird ein Beispiel der Erzeugung der *wrapper class* eines COM-Objekts *Synergy* von MPI vorgestellt.

Die Informationen über COM-Objekte in MPI sind in der Hilfsdokumentation „MPI Online Document: MPI Application Programming Interface“ [73] dargestellt, und können auch durch die Anwendung der Funktion *macro recorder* in MPI erhalten werden.

- Es gibt 30 öffentliche COM-Objekte in MPI wie z. B. *Synergy*, *StudyDoc*, *Boundary-Conditions* usw.
- Die Klasse *Synergy* hat 8 öffentliche Methoden (z. B. *NewProject*, *OpenProject*, usw.) und 21 öffentliche Eigenschaften (z. B. *Project*, *ImportOptions*, *StudyDoc*, usw.). Die Klasse *Synergy* spielt in API eine zentrale Rolle, weil die meisten Objekte von MPI als ihre Eigenschaften behandelt werden.
- Weiterhin ist die detaillierte Beschreibung der Methoden auch in der Hilfsdokumentation dargestellt, wie z. B. *Boolean NewProject (string aName, String aDir)*.

Um die entsprechende *DispID* der Methode oder Eigenschaft zu bekommen, wurde ein kurzes C++ Programm als *Client* geschrieben. Die wichtigen angewandten Funktionen sind *QuerInterface()* und *GetIDsOfNames()*. Wie in Bild 78 gezeigt, entspricht der Wert 0x16 von *DispID* der Methode *NewProject*.

Die *wrapper class* der Klasse *Synergy* kann basierend auf der MFC-Klasse *COleDispatchDriver* und der oben dargestellten Information erzeugt werden.

```
In Moldflow.h:
...
//Synergy wrapper class-----
class CMpiSynergy : public COleDispatchDriver
{
public:
    CMpiSynergy() {} // Calls COleDispatchDriver default constructor
    CMpiSynergy(LPDISPATCH pDispatch) : COleDispatchDriver(pDispatch) {}
    CMpiSynergy(const CMpiSynergy& dispatchSrc) : COleDispatchDriver(dispatchSrc) {}

// Attributes
public:
    LPDISPATCH GetImportOptions();
    LPDISPATCH GetMeshGenerator();
    LPDISPATCH GetBoundaryConditions();
    LPDISPATCH GetMaterialSelector();
    LPDISPATCH GetStudyDoc();
    ...
// Operations
public:
    BOOL OpenProject(LPCTSTR lpszProject);
    BOOL NewProject(LPCTSTR lpszName, LPCTSTR lpszDir);
    BOOL ImportFile(LPCTSTR lpszFile, LPDISPATCH pOpts, BOOL bShowLog);
    ...
};

In Moldflow.cpp:
...
//Synergy wrapper class-----
BOOL CMpiSynergy::NewProject(LPCTSTR lpszName, LPCTSTR lpszDir)
{
    BOOL result;
    static BYTE parms[] = VTS_BSTR VTS_BSTR;
    InvokeHelper(0x16, DISPATCH_METHOD, VT_BOOL, (void*)&result, parms, lpszName, lpszDir);
    return result;
}
...
```

Bild 79: Erzeugung der wrapper class für die Klasse Synergy von MPI

Tabelle 4: Übersetzungsformat für die Programmierschnittstelle

Name in API		Name in Wrapper Class
Klasse (oder COM-Objekt)		<b>CMpiClass</b> (z.B. <i>CMpiSynergy</i> )
Eigenschaft	Erhalten einer Eigenschaft (DISPATCH_PROPERTYGET)	<b>GetFunction</b> (z.B. <i>GetStudyDoc()</i> )
	Einstellen einer Eigenschaft (DISPATCH_PROPERTYSET)	<b>SetFunction</b> (z.B. <i>SetMeshType()</i> )
Methode (DISPATCH_METHOD)		<i>Function</i> (z.B. <i>NewProject()</i> )

Mit der oben dargestellten Methode, die *wrapper classes* für die COM-Objekte von MPI erzeugt, wird eine zusätzliche Programmierschnittstelle zur Verfügung gestellt, die eine Verbindung zwischen MIDCAD und MPI realisiert.

### **Verbindung zwischen MIDCAD und MPI**

Nachdem alle *wrapper classes* für die Programmierschnittstelle von MPI nach der oben dargestellten Methode definiert und erzeugt sind, kann MIDCAD als ein OLE-Automation *Client* mit dem *Server* MPI verbunden werden. Neben der Definition der *wrapper classes* sind noch die folgenden zwei Schritte notwendig:

- Die MFC-Funktion *AfxOleInit()* muss ins MIDCAD-Programm eingefügt werden, um die Fähigkeit der OLE-Automation in MIDCAD zu ermöglichen.
- Ein Objekt der Klasse *CMpiSynergy* muss definiert und die Funktion *CreateDispatch („synergy.Synergy“)* aufgerufen werden, um die aktuelle Verbindung zwischen MIDCAD und MPI zu erzeugen.

Nachdem die Verbindung erfolgreich erzeugt ist, können die öffentlichen COM-Objekte und deren öffentlichen Methoden von MPI direkt in MIDCAD benutzt und aufgerufen werden, wodurch die Integration der Spritzgussimulation in MIDCAD realisiert wird. Zum Beispiel kann durch den Aufruf der Methode *AnalyseNow()* des COM-Objekts *StudyDoc* der Simulationsprozess in MPI automatisch stattfinden.

#### **7.1.3 MID-spezifische Materialbibliothek für die Spritzgussimulation**

MPI bietet mit mehr als 7.800 Rohstoffen die weltweit größte Materialdatenbank, bei der die verschiedenen thermoplastischen Materialtypen sowie deren detaillierte Eigenschaften für die Spritzgussimulation gespeichert werden. Der MID-Konstrukteur kann zwar in MPI leicht das Material spezifizieren, möchte aber in den frühen Designphasen das Material festlegen. Deswegen wurde eine eigene spezialisierte Materialbibliothek, die nur die Materialien zur Herstellung von MID-Produkten speichert, in MIDCAD entwickelt. Die Anzahl der Materialien in der Bibliothek ist wegen der besonderen Ausrichtung auf MID-Produkte deutlich geringer als in der Bibliothek von MOLDFLOW. Durch die Bereitstellung der Materialbibliothek kann der Konstrukteur leicht das gewünschte Material direkt in MIDCAD auswählen, ohne zusätzlich die Materialbibliothek von MOLDFLOW durchsuchen zu müssen.

In der MID-spezifischen Materialbibliothek ist nicht nur die Anzahl der Materialtypen geringer als in MOLDFLOW, sondern auch die Datenmenge von jedem Material fällt geringer aus als in MOLDFLOW, weil die analyserelevanten Eigenschaften der Materialien nicht so ausführlich wie in MOLDFLOW beschrieben werden müssen. In der MID-spezifischen Materialbibliothek werden hauptsächlich die folgenden Informationen jedes Materials beschrieben:

- *Trade Name*: Der Markenname des Materials. Dieser Name gilt einzig für das Material wie z. B. *Ultramid T KR 4357G6*,
- *Family abbreviation (Materialtyp)*: Der Familientyp des Materials wie z. B. *PA6/6T*,
- *Manufacturer*: Der Hersteller des Materials wie z. B. *BASF*,
- *Index in Mpi*: Die Identität des Materials in MPI. Der Wert gilt einzig für das Material wie z. B. *10523* für das Material *Ultramid T KR 4357G6*,
- *FileName in Mpi*: Der Name der Datei, in der die Informationen des entsprechenden Materials gespeichert werden wie z. B. *BASF.21000.udb*,
- *FileType in Mpi*: Der Typ der Datei, in der die Informationen des entsprechenden Materials gespeichert werden. Es gibt zwei Werte für diese Eigenschaft: „*System*“ bedeutet, dass die Datei vom MPI-System angeboten wird und vordefiniert ist; „*User*“ bedeutet, dass das Material vom Benutzer selbst definiert und in der MPI-Materialbibliothek eingefügt ist.

Die Werte der oben dargestellten Eigenschaften und die in MID-Produkten angewandten Materialien, wurden aus der MPI-Materialdatenbank extrahiert, um die Anwendung der anderen in der MPI-Materialbibliothek definierten Materialeigenschaften bei der Spritzgussimulation zu sichern. In MIDCAD wird das Material zusammen mit den anderen analyserelevanten Daten als eine Eigenschaft von *Part Feature* spezifiziert. Bild 75 zeigt die Spezifikation des Materials für einen Schaltungsträger, der im Einkomponentenspritzgussverfahren hergestellt wird. Zum Suchen und Auswählen eines Materials sind der Materialtyp zusammen mit dem Markennamen und dem Namen des Herstellers als Index verfügbar. Die Identität des Materials und die entsprechende Datei werden nur intern für die selbst entwickelte Programmierschnittstelle angewandt und bleiben dem Konstrukteur verborgen.

Viele analyserelevante Eigenschaften der Materialien wie z. B. mechanische und thermische Eigenschaften sind zwar nicht ein weiteres Mal in der MID-Materialdatenbank beschrieben, aber die MID-relevanten Eigenschaften wie z. B. Lötbarkeit, Metallisierbarkeit, chemische Beständigkeit usw. sollen in Zukunft aus dem MID-Handbuch abgeleitet und in der MID-Materialdatenbank beschrieben und gespeichert werden, um ein geeignetes Material für MID-Produkte auswählen zu können.

Bei der Auswahl geeigneter Materialkombinationen für die 2K-Spritzguss-Teile müssen einige prozessspezifische Kriterien besonderes beachtet werden, wie z. B., dass der Werkstoff des ersten Schusses der Schmelztemperatur und dem Werkzeuginnendruck des zweiten Schusses standhalten muss. Außerdem ist die partielle - nur auf den Leiterbahnen erfolgende - Beschichtung bzw. Metallisierung von 2K-Spritzguss-Teilen nur bei bestimmten Kunststoffkombinationen gut möglich, wie z. B. die Kombinationen PC/ABS // PC und PA66 // PA6/6T (metallis. K. // nicht met. K.); weiterhin sind Kombinationen aus LCP//LCP, PPA//PPS und sPS//sPS im Serieneinsatz möglich [86].

Zur Verwaltung der Daten der MID-spezifischen Materialbibliothek ist die Datenbanktechnologie anzuwenden. Um auf Daten in Datenbankmanagementsystemen (DBMS) zugreifen zu können, bietet Microsoft die Schnittstelle *Open Database Connectivity* (ODBC) an. Der ODBC-Standard besteht aus einer erweiterbaren Menge von DLLs, die eine Standardprogrammierschnittstelle für Datenbankanwendungen bereitstellen.

Im Rahmen dieser Arbeit ist es jedoch von Vorteil, hinsichtlich der Programmierung, ein möglichst einfaches Datenbankmanagementsystem zu verwenden. Die Verbindung zwischen MIDCAD und der MS ACCESS Datenbank wurde basierend auf den relevanten MFC-Klassen und der OLE-DB Technologie realisiert. Das *Frontend* zur Datenverwaltung ist ein eigenes Modul in MIDCAD und das *Backend*, d. h. die Datenverwaltung wird mit MS ACCESS durchgeführt (siehe Bild 80).

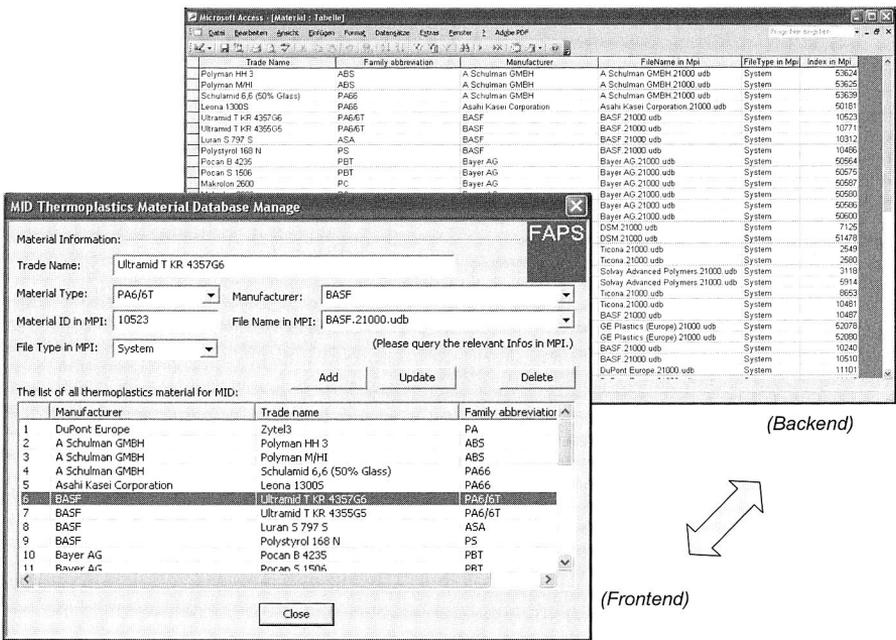


Bild 80: Datenverwaltung für MID-Materialbibliothek in MIDCAD (Frontend) und MS ACCESS Datenbank (Backend)

In der MS ACCESS Datenbank wurde für alle in MID angewandten Materialien eine Tabelle angelegt. In MIDCAD wurde auch eine graphische Benutzerschnittstelle entwickelt, um die Datenverwaltung direkt in MIDCAD durchführen zu können. Bild 80 zeigt die erzeugte Tabelle „Material“ in der ACCESS Datenbank und die in MIDCAD laufende Benutzerschnittstelle für die Datenverwaltung. In Bild 80 ist auch zu sehen, dass die MID-Materialbibliothek im jetzigen Entwicklungsstand nur eine Tabelle hat, in der die ther-

moplastischen Materialien gespeichert sind. Ein Ausblick auf die weitere Entwicklung liegt darin, dass in der Bibliothek eine zweite Tabelle eingeführt werden soll, in der die Materialien zur Metallisierung und Oberflächenveredelung für Leiterbahnen von MID-Produkten gespeichert werden.

#### 7.1.4 Integration der Spritzgussimulation im 3D-MID-Design

Bild 81 zeigt den Ablauf eines integrierten 3D-MID-Designs mit der Spritzgussimulation mit Hilfe des MIDCAD-Systems und des CAE-Systems (MPI). Der Ablauf besteht hauptsächlich aus den folgenden Schritten.

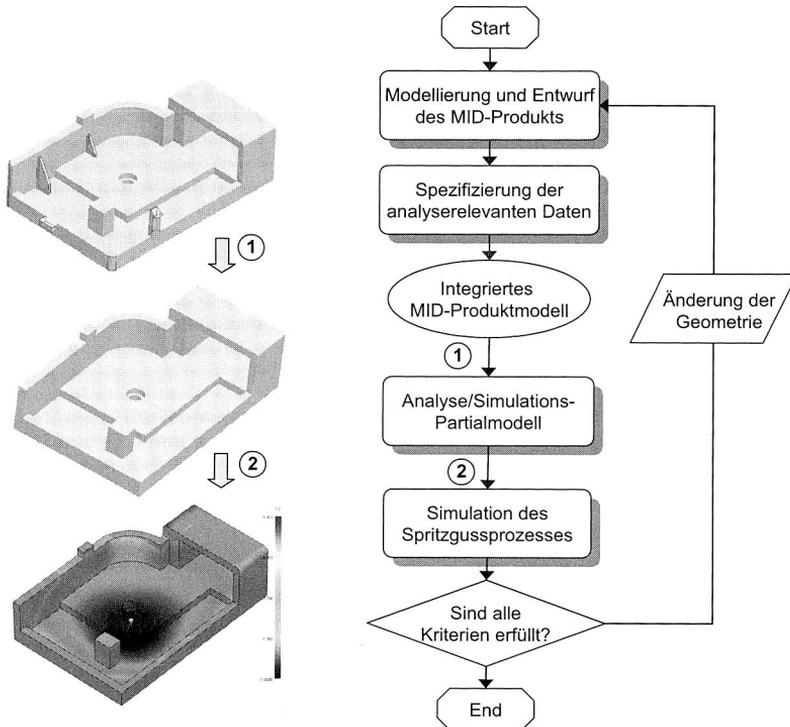


Bild 81: Integration der Spritzgussimulation im 3D-MID-Design

**Modellierung und Entwurf des MID-Produkts:** Der Schritt entspricht der hauptsächlichen Aufgabe des 3D-MID-Designs. Eingesetzt werden die wichtigen MID-spezifischen Funktionalitäten, z. B. 3D-Platzierung, -Routing, Erzeugung der 3D-Leiterbahnen usw.

**Spezifizierung der analyserelevanten Daten:** Wichtig ist hier, dass die nötigen analyserelevanten Informationen bei der Definition der *Part Feature* eingegeben sind. Für ein 2K-Spritzguss-Teil hat die Spezifizierung der Spritzgießparameter einen entscheidenden Einfluss.

den Einfluss auf die Haftfestigkeit der Metallschicht bei der Metallisierung. Es ist daher empfehlenswert, die Massetemperatur, die Werkzeugtemperatur und die Einspritzgeschwindigkeit dem Metallisierungsprozess anzupassen. Zum Beispiel wird bei den ABS-Kombinationen eine gute Haftfestigkeit der Metallisierung erzielt, wenn die metallisierbare Komponente mit einer höheren Masse- und Werkzeugtemperatur und einer kleinen Einspritzgeschwindigkeit spritzgegossen wird [86]. Außerdem muss die Definition der Angusspunkte beachtet werden, denn diese dürfen nicht auf der zu metallisierenden Leiterbahn liegen.

**Integriertes MID-Produktmodell:** Das Produktmodell muss die zur Durchführung der Spritzgussimulation benötigten Daten (z. B. Rand- bzw. Spritzgießparameter, Materialdaten) enthalten.

Die Methode der Realisierung der ersten drei Schritte ist schon in früheren Kapiteln diskutiert worden. Hier werden die nachfolgenden Schritte detailliert vorgestellt.

**Analyse- bzw. Simulationsmodell:** Der Schritt ist die Ableitung bzw. Abstraktion des Analyse- bzw. Simulationsmodells mit den geometrischen und nichtgeometrischen Informationen aus dem integrierten Produktmodell. Der Ablauf unterteilt sich in zwei Schritte: Zum einen in die Ableitung des Geometriemodells für die Simulation und zum anderen in die Abstraktion der nichtgeometrischen, analyserelevanten Informationen.

Das Geometriemodell im CAE-System ist in der Regel einfacher als im CAD-System. Für die Berechnung müssen die unerheblichen Geometrieelemente im Modell erkannt und ausgeblendet werden, was den Einsatz automatischer Vernetzer für eine FEM-Berechnung bzw. Simulation erheblich beschleunigt und Vernetzungsfehler vermeidet. In MIDCAD kann die Ableitung des vereinfachten Geometriemodells durch Feature-Technologie realisiert werden, weil die Ausblendbarkeit der Features schon definiert ist. Die Einstellung der Eigenschaft „Ausblendbarkeit“ eines Features ist abhängig von den Erfahrungen und Kenntnissen der Konstrukteure. Ein intelligentes Modul wurde zur Ableitung des Simulationsmodells entwickelt, indem die entsprechenden Features aus dem integrierten Produktmodell identifiziert und anschließend nach den geometrischen Eigenschaften der Features (Ausblendbarkeit) beurteilt werden, um die unerheblichen Features zu erkennen und auszublenden. In Bild 81 sind ein beispielhaftes Geometriemodell des Schaltungsträgers und das vereinfachte Geometriemodell für die Simulation gezeigt. Einige mit roter Farbe markierte Features (*Rib*, *Catch*, *Boss* usw.) sind im Simulationsmodell ausgeblendet, wie auch die spezifischen Features *3D-Circuit* und *Land* aufgrund der relativ geringen Dicke.

Die Abstraktion der nichtgeometrischen analyserelevanten Informationen ist leicht zu realisieren, weil diese Daten im integrierten Produktmodell gespeichert sind und als die Eigenschaften von *Part Feature* beschrieben werden.

**Simulation des Spritzgussprozesses:** Mit Hilfe der selbst entwickelten Programmierschnittstelle zwischen MIDCAD und MPI, können die geometrischen und nichtgeometrischen Daten vom Simulationsmodell direkt in MPI übertragen werden. Da MIDCAD auf

die Funktionalität von MPI direkt zugreifen kann, wird durch das Menü „Analyse Now“ in MIDCAD die Spritzgussimulation aktiviert. Anschließend findet der ganze Simulationsprozess automatisch in MPI statt (siehe Bild 82).

In MPI wird ein neues Projekt für die aktuelle Simulation erzeugt. Das vereinfachte Geometriemodell wird zuerst im neuen Projekt importiert, dann wird die Vernetzung des Geometriemodells durchgeführt. Danach werden andere Rand- und Spritzgießparameter schrittweise eingestellt. Schließlich wird der Simulationsprozess aktiviert. Alle Schritte in MPI laufen automatisch ab. Bild 82 zeigt das Vernetzungsergebnis und eines (*Fill time*) der vielseitigen Ergebnisse nach der Spritzgussimulation eines Beispiels in MPI. Die wichtigen analyserelevanten Daten des Beispiels sind in Bild 75 gezeigt.

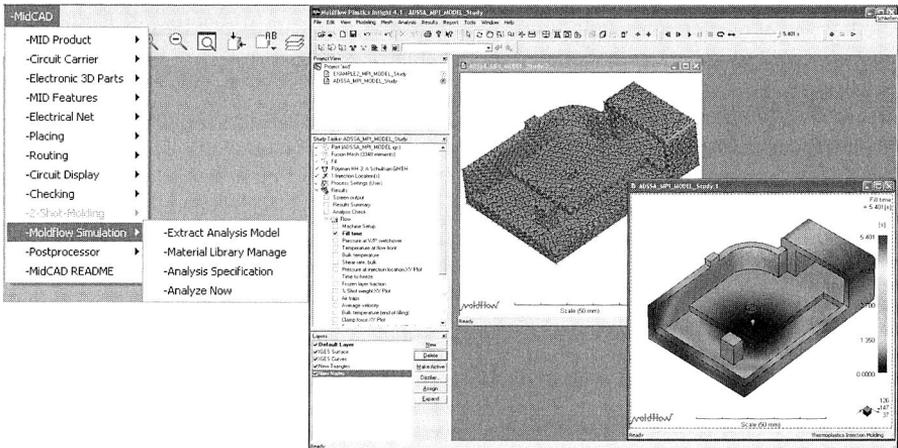


Bild 82: Simulation des Spritzgussprozesses in MPI

**Überprüfung der vordefinierten Kriterien:** Nach der Spritzgussimulation werden die Ergebnisse analysiert und verarbeitet sowie die vordefinierten Kriterien geprüft. Wenn einige Kriterien oder spritzguss- und metallisierungsgerechte Gestaltungsrichtlinien nicht erfüllt werden, sollen die Geometrie des Produktmodells oder die analyserelevanten Rand- bzw. Spritzgießparameter (z. B. Angusspunkte) modifiziert werden. Zum Beispiel verhindern die Bindenähte eine haftfeste Metallisierung und dürfen deswegen nicht in den Leiterbahnen entstehen (siehe auch Bild 46). Die Verlagerung von Bindenähten kann durch die Änderung der geometrischen Gestaltung und Angusspunkte (Lage oder Anzahl) realisiert werden. Die Regel-Schleifen der Optimierung müssen so lange durchgeführt werden, bis alle Kriterien und Gestaltungsrichtlinien erfüllt sind.

### 7.1.5 Automatische Gestaltoptimierung durch Spritzgussimulation

Durch die Realisierung der Integration der Spritzgussimulation im 3D-MID-Design können die geforderte Qualität und die Spritzgießbarkeit der MID-Schaltungsträger sicher-

gestellt werden. Zwar läuft der Simulationsprozess vollautomatisch, aber die Optimierung der geometrischen Struktur kann in MPI nicht durchgeführt werden, sondern muss in MIDCAD nach der Analyse der Simulationsergebnisse manuell durchgeführt werden. Der Vorgang ist zwar ganz normal und typisch, aber dafür passiv und er hat eine niedrige Effizienz. Es gibt auch eine andere Strategie bzw. Methode, die geometrische Strukturoptimierung automatisch und aktiv durchführen zu können. Das Konzept der Anwendung dieser Methode wird im Folgenden dargestellt.

In MIDCAD sollen das integrierte Produktmodell bzw. die Datenbeschreibung der MID-Features gemäß dieser Anforderung erweitert werden. Dazu werden nicht nur die nicht-geometrischen, analyserelevanten Daten, sondern auch die geometrischen Variablen zur Strukturoptimierung als Eigenschaften der MID-Features definiert. Dies geschieht, indem die relevanten Parameter und Dimensionen (z. B. Dicke) von den Features ausgewählt und durch die entsprechenden Variablen spezifiziert werden. Diese Variablen sind entweder form-relevant oder lage-relevant. Weiterhin müssen die Veränderungsstrategie (z. B. das Gewicht jeder Variablen) und Schrittweite der Variablen genau definiert werden. Diese Definition ist ein wichtiger Schritt und stark von der Erfahrung und Kenntnis des Konstrukteurs abhängig. Nachdem diese Informationen in das Produktmodell integriert sind, können die verschiedenen Varianten des Simulationsmodells abgeleitet werden. Aufgrund der Integration der Veränderungsdefinition im Produktmodell und der Verwendung des parametrischen Modellierungsansatzes, kann die Ableitung der verschiedenen Varianten des Simulationsmodells automatisch realisiert werden. Durch die Spritzgussimulation und die nachfolgende Überprüfung der vordefinierten Kriterien kann ein optimales Geometriemodell unter den abgeleiteten Varianten gefunden werden.

Der wichtige Vorteil dieser Methode ist, dass die Gestaltoptimierung automatisch und aktiv durchgeführt werden kann. Die Entwicklungseffizienz ist deutlich höher als bei der manuellen Optimierung. Jedoch gibt es auch einige Begrenzungen. So kann z. B. die Topologie der geometrischen Struktur nicht durch diese Methode verändert werden. Es ist auch schwierig, die Veränderungsstrategie und Schrittweite der Variablen zu definieren. Weiter gilt zu bedenken, dass die Ergebnisse der Platzierung und des Routings durch die Änderung der Geometrie beeinflusst werden, wenn die Geometrie der 3D-Leiterbahn im Simulationsmodell nicht ausgeblendet ist, besonders beim 2K-Verfahren. Trotzdem ist die Realisierung der automatischen Gestaltoptimierung zusammen mit der automatischen Überprüfung der Kriterien ein guter Ausblick auf die Integration der Spritzgussimulation.

## **7.2 Entwicklung einer CAD/CAM-Kette für ein MID-Bestücksystem**

### **7.2.1 Bestücksysteme für räumliche spritzgegossene Schaltungsträger**

Die 3D-MID-Technologie bietet die Chance, die Zahl der Prozessschritte und damit die Komplexität deutlich zu reduzieren, und eröffnet damit sowohl in der Montage als auch

in den indirekten Bereichen der Fertigung Kostensenkungspotentiale. Die 3D-MID-Technologie stellt andererseits jedoch erhöhte Anforderungen an die Beherrschung der Fertigungsprozesse und der Montagetechnik [31]. Die Entwicklung geeigneter Montageprozesse und -systeme für MID-Baugruppen wird stark von der geometrischen Gestaltung des Schaltungsträgers und von der Form der Bauelemente beeinflusst. Um diesen Anforderungen zu genügen, wurden am Lehrstuhl zwei alternative Bestücksysteme unter Berücksichtigung aller fertigungstechnischen Randbedingungen der Elektronikproduktion konzipiert und bisher prototypisch im Labor realisiert (siehe Bild 83).

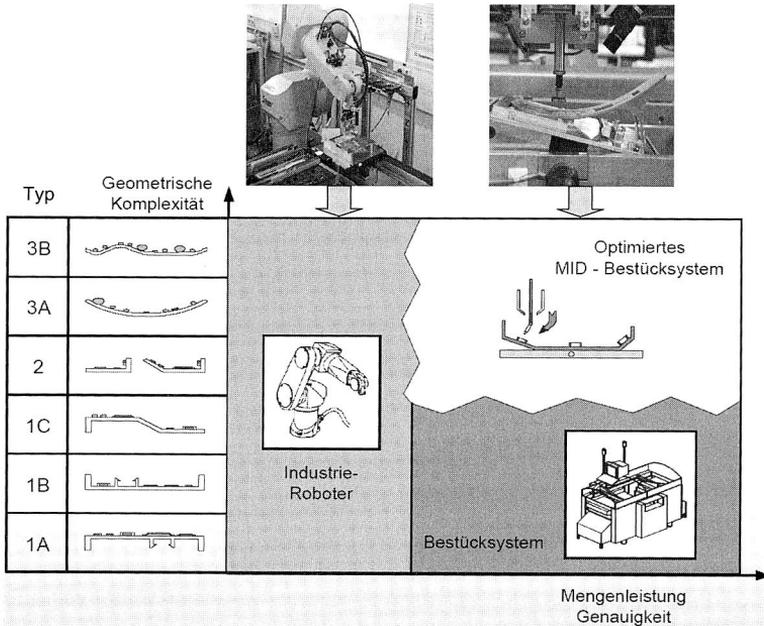


Bild 83: Systemkonzepte für das Bestücken räumlicher Schaltungsstrukturen in Abhängigkeit von der Produktstruktur [86]

Um den durch die Formgebung von 3D-MID-Produkten bedingten Anforderungen gerecht zu werden, wurde am Lehrstuhl ein Bestücksystem auf der Basis eines 6-achsigen Gelenkroboters entwickelt. Für räumliche Schaltungsstrukturen mit hoher Geometrie Komplexität ist diese Lösung geeignet, bei der durch die Integration geeigneter Werkzeuge zum Lotpastenauftrag und zum Bestücken ein hoher Grad an Bewegungsfreiheit erreicht wurde [57]. Im Vergleich zum Standardbestücksystem stehen die Mengenleistung und Arbeitsgenauigkeit dieses Systems auf relativ niedrigem Niveau. Alternativ dazu wurde ein so genannter *MIDplace* Bestückautomat auf der Basis des Standardbestücksystems durch die Entwicklung zusätzlicher Komponenten in seiner Flexibilität erweitert [8], wie z. B. der Bestückkopf mit zusätzlicher Rotationsachse im

Sauggreifer. Diese Lösung behält nicht nur die Vorteile des Standardbestücksystems (möglichst große Mengenleistung und relativ hohe Genauigkeit) bei, sondern zeichnet sich auch durch eine hohe Geometrieflexibilität aus. Dieses MID-Bestücksystem wurde modular aufgebaut und befindet sich derzeit noch in der Erprobung.

Die entwickelten Bestücksysteme eignen sich für eine wirtschaftliche Bestückung räumlicher Schaltungsträger. Auf diese Weise tragen sie zu einer Verbesserung der Rahmenbedingungen für die Nutzung der Rationalisierungspotentiale der MID-Technologie bei.

### **7.2.2 Konzept der Verbindung zwischen MIDCAD und dem MID-Bestücksystem**

Die Anzahl der Bauelemente, mit denen ein Schaltungsträger bestückt werden soll, ist in der Regel so groß, dass eine manuelle Eingabe der Bestückdaten unwirtschaftlich ist. Die meisten Bestücksysteme erlauben zwar eine manuelle Eingabe der Bestückdaten, eine automatische Generierung ist jedoch schon bei einer geringen Anzahl von Bauelementen deutlich effizienter. Ziel ist daher die Umsetzung einer solchen automatischen Funktion innerhalb des MIDCAD-Systems und MID-Bestücksystems, um die Programmierzeit für den Bestückvorgang zu reduzieren und Fehler durch eine falsche Dateneingabe zu vermeiden.

Notwendige Voraussetzung auf Seiten des MIDCAD-Systems ist hierfür, dass in der 3D-Bauelementebibliothek jedes Bauelement die notwendigen bestückungsrelevanten Informationen enthält. Neben der Beschreibung der Bauelemente mit dem erforderlichen Detailgrad sind ein Modul zur Ableitung der 3D-Bestückposition sowie ein weiteres zur statischen Kollisionsprüfung zu entwickeln. Durch die Kombination der Funktion zur 3D-Platzierung und den oben genannten Modulen kann ein kollisionsfreier Bestückvorgang sichergestellt werden (siehe Bild 84).

Zur Umsetzung der CAD/CAM-Kopplung soll der am Lehrstuhl entwickelte Standard IPC-D-355 (*Interconnecting and Packaging Electronic Circuits*) [35] auf seine Eignung für den Datentransfer der Bestückposition für MID-Produkte überprüft werden. Der Standard IPC-D-355 behandelt den Transfer von Bestückdaten zwischen dem Design im ECAD-System und der Fertigung im CAM und ist für die automatische Generierung von Anweisungslisten erforderlich. Die Bestückdaten werden aus dem MIDCAD-System im IPC-D-355 Format bereitgestellt und mit Hilfe der Steuerung des MID-Bestücksystems in eine Anweisungsliste umgewandelt. Zur Generierung der Anweisungsliste sind auch die bestücksystembezogenen Informationen, wie Feederpositionen, Schaltungsträgerorientierung im Bestücksystem sowie Mappingdaten, notwendig.

Die statische Kollisionsprüfung kann die potentiellen Kollisionen beim Verfahren des Werkzeugs zwischen Bereitstellungsposition und Fügeort nicht vorhersagen. Deswegen muss die Möglichkeit geschaffen werden, den gesamten Bestückungsablauf auf die Kollisionsfreiheit zu prüfen und die Anweisungsliste entsprechend zu verändern. Damit erhält man eine kollisionsfreie Anweisungsliste für die Bestückungsaufgabe. Zur dynamischen

schen Überprüfung der Kollisionsfreiheit soll dabei die dreidimensionale kinematische Simulation zum Einsatz gebracht werden. Die so erzeugte, maschinenneutrale Beschreibung der Bestückungsaufgabe kann durch systemspezifische Postprozessoren weiterverarbeitet werden [8].

Die Technik und entsprechende Module auf Seiten des MID-Bestücksystems sind schon in Dissertationen [8], [57] viel diskutiert und realisiert worden. Außerdem ist auch die 3D-Kinematiksimulation am Beispiel der Roboterbestückungszelle realisiert worden [57]. Im Rahmen der vorliegenden Forschungsarbeit sind die im Folgenden dargestellten bestückungsrelevanten Funktionen bzw. Module seitens des MIDCAD-Systems zu entwickeln, um eine erfolgreiche Verbindung zwischen MIDCAD und dem MID-Bestücksystem zu realisieren.

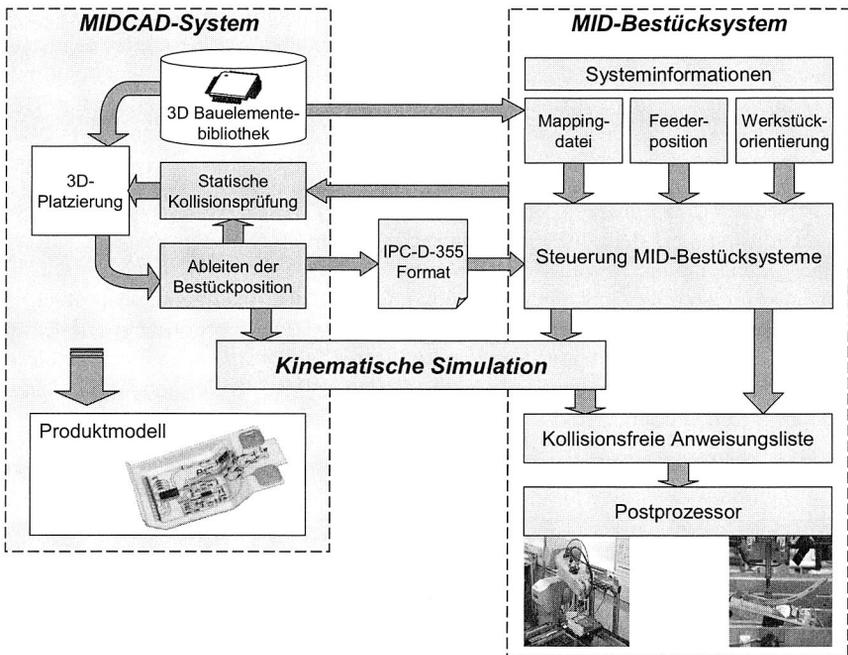


Bild 84: CAD/CAM-Kette zwischen MIDCAD und dem MID-Bestücksystem

### 7.2.3 Definition der bestückungsrelevanten Informationen

Wie in Kapitel 6.3.1 dargestellt, muss die Bestückfläche für die Bauelemente bzw. Gehäuse in der 3D-Bauelementebibliothek definiert werden, um die 3D-Platzierung der Bauelemente auf dem Schaltungsträger durchführen zu können. Die Bestückfläche wird in Pro/ENGINEER-Modellen der Bauelemente bzw. Gehäuse mit einer *Datum Plane* und dem einheitlichen Namen *SEAT* repräsentiert und im entsprechenden Datenmodell

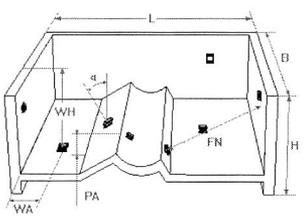
durch das Objekt *Seating\_plane* dargestellt. Weiterhin sind zur Ableitung der Bestückposition der Bauelemente, der Greif- oder Ansaugpunkt und das entsprechende Koordinatensystem für Bauelemente in der Bibliothek zu definieren. Der Greif- oder Ansaugpunkt und das entsprechende Koordinatensystem werden in Pro/ENGINEER-Modellen der Bauelemente bzw. Gehäuse mit einem *Datum Coordinate System* und dem einheitlichen Namen *GREIF* repräsentiert und im entsprechenden Datenmodell durch das Objekt *Package\_XYZ* dargestellt (siehe auch Kapitel 6.3.2).

Die oben genannten bestückungsrelevanten Daten stehen nur im Zusammenhang mit einem einzelnen Bauelement. Die benötigten, bestückungsrelevanten Informationen müssen weiter definiert werden, um die statische Kollisionsprüfung durchführen und kollisionsfreie Bestückpositionsdaten ableiten zu können.

Das bestückbezogene Koordinatensystem des Schaltungsträgers muss vor der Ableitung der Bestückpositionsdaten der Bauelemente im Pro/ENGINEER-Modell des Schaltungsträgers definiert werden. Zur Realisierung der statischen Kollisionsprüfung müssen neben den abgeleiteten Bestückpositionen der Bauelemente die relevanten Geometriemodelle der verschiedenen Bestückköpfe der MID-Bestücksysteme in MIDCAD definiert oder bereitgestellt werden.

In der Regel wird die dreidimensionale geometrische Struktur eines Schaltungsträgers im CAD-System (z.B. MIDCAD) unter funktionalen Gesichtspunkten gestaltet und modelliert. Dann kann die Struktur im Rahmen der Möglichkeiten nach hierarchischer Vorgehensweise durch die Spritzgussimulation und DRC/MRC (inkl. Kollisionsprüfung) an die Belange der Herstellung und Montage optimiert werden. Dabei sollten in MIDCAD einige wichtige bestückbezogene Kenngrößen (siehe Tabelle 5) erhalten und kontrolliert werden, die die Bestückungsaufgabe besonders beeinflussen. Die detaillierte Darstellung dieser Kenngrößen ist in der Dissertation [8] vorgestellt.

Tabelle 5: Spektrum geometrischer Merkmale mit dem System MIDplace bestückbarer Schaltungsträger [8]

	Merkmal	Wert	Beispiel
1	Neigungswinkel $\alpha$	beliebig	
2	Länge der Fügennormalen $FN$	>40	
3	Wandabstand $WA$	>3	
4	Anzahl Prozessflächen $n$	beliebig	
5	Abstand Prozessflächen $PA$	<150	
6	Hauptmaße $LxBxH$	400x400x150	
7	Wandhöhe $WH$	<150	

Die Definition dieser bestückungsrelevanten Daten kann wie die Definition der analyserelevanten Daten für *Part Feature* mit der Feature-Technologie realisiert werden (siehe Bild 73). Dadurch können die CAM-relevanten Informationen zusammen mit den CAE-relevanten Informationen im Produktmodell integriert werden.

### 7.2.4 Ableitung der Bestückposition der Bauelemente

Zur Erstellung der IPC-D-355-Datei müssen die abgeleiteten Bestückpositionsdaten für jedes Bauelement drei Koordinaten und drei Winkel enthalten. Die Koordinaten  $x$ ,  $y$  und  $z$  bestimmen die Position des Bauelements auf dem Schaltungsträger, bezogen auf das bestückungsrelevante Koordinatensystem des Schaltungsträgers, sowie die Winkel  $\alpha$ ,  $\beta$  und  $\gamma$  seine Orientierung. Da das Koordinatensystem *GREIF* des Bauelements für die Bestückung definiert ist, entspricht die Bestückung der Koordinatentransformation des Bauelements vom eigenen Koordinatensystem *GREIF* in das bestückungsrelevante Koordinatensystem des Schaltungsträgers und weiterhin in das Koordinatensystem der Bestückmaschine.

Die Koordinatentransformation wird nach der Euler-Transformation durchgeführt, die bei Robotern häufig angewandt wird. Die Winkel  $\alpha$ ,  $\beta$  und  $\gamma$  heißen Euler-Winkel, d. h. die Orientierung wird festgelegt, indem das Bezugskordinatensystem, an der Stelle des bestückungsrelevanten Koordinatensystems des Schaltungsträgers, durch die folgende Reihenfolge der Drehungen verdreht wird: Zuerst eine Drehung um die Z'-Achse um den Winkel  $\alpha$ , dann eine Drehung um die neue Y'-Achse um den Winkel  $\beta$  und schließlich eine Drehung um die neue X'-Achse um den Winkel  $\gamma$ . Nach einer Verschiebung des verdrehten Zwischenkoordinatensystems steht das endgültige Bezugskordinatensystem an der Stelle des Koordinatensystems *GREIF* des jeweiligen Bauelements auf dem Schaltungsträger. Dabei wurde eine Koordinatentransformationsmatrix  $M_{transf}$  für die Euler-Transformation hergeleitet, die in folgender Formel dargestellt wird. Bild 85 zeigt die in MIDCAD abgeleiteten Bestückpositionsdaten der Bauelemente eines Beispiels.

No.	Reference	Part Name	x	y	z	alfa(z)	beta(y)	gama(x)
1	C1	MK316-D	13.16	6.73	3.84	-90.00	0.00	0.00
2	C2	MK325-D	50.43	7.28	3.84	-90.00	0.00	0.00
3	D3	SOT23DIO_MAX	43.15	28.32	7.21	180.00	0.00	0.00
4	D4	SOD323DIO_MAX	16.84	45.90	12.89	0.00	90.00	90.00
5	Q1	SOT23TRAN_MAX	43.30	22.27	7.21	180.00	0.00	0.00
6	R1	RD41-2D	40.18	45.63	11.73	180.00	0.00	-90.00
7	R2	RK73-2B	11.45	7.60	3.60	-0.51	0.00	0.00
8	c3	RD41-2D	67.71	26.13	13.77	0.00	90.00	90.00
9	u1	SOPIC10	11.94	32.66	4.50	180.00	0.00	0.00
10	u2	SOPIC10	42.73	37.98	7.50	180.00	0.00	0.00
11	c4	SOT23DIO_MAX	65.13	15.85	7.21	180.00	0.00	0.00
12	r4	MK316-D	28.50	39.94	6.84	-90.00	0.00	0.00

important:  
The value of angles is the Euler angles of the rotation around the three axes.

Bild 85: Ergebnis der abgeleiteten Bestückpositionen der Bauelemente

$$M_{transf} = \begin{bmatrix} \cos \beta \cos \alpha & \cos \beta \sin \alpha & -\sin \beta & 0 \\ \sin \gamma \sin \beta \cos \alpha - \cos \gamma \sin \alpha & \cos \gamma \cos \alpha + \sin \beta \sin \gamma \sin \alpha & \sin \gamma \cos \beta & 0 \\ \cos \gamma \sin \beta \cos \alpha + \sin \gamma \sin \alpha & \cos \gamma \sin \beta \sin \alpha - \sin \gamma \cos \alpha & \cos \gamma \cos \beta & 0 \\ x & y & z & 1 \end{bmatrix} \quad (11)$$

### 7.2.5 Statische Kollisionsprüfung mit verschiedenen Bestückköpfen

Basierend auf den abgeleiteten Daten der Bestückposition und dem 3D-Geometriemodell des Bestückkopfs können die Kollisionsprobleme zwischen dem Bestückkopf und den MID-Komponenten (dem Schaltungsträger oder den bestückten Bauelementen) beim Bestückvorgang frühzeitig erkannt werden. Die erkannten Kollisionsprobleme können durch die Änderung der Bestückposition der Bauelemente oder durch den Einsatz anderer Bestückköpfe gelöst werden.

In MIDCAD können verschiedene Typen von Bestückköpfen jeweils modelliert werden. Das Geometriemodell ist kein detailliertes CAD-Modell, sondern ein vereinfachtes und mit besonderen Anforderungen an Bestückung parametrisch modelliertes Modell, um die richtige Bestückposition des Bestückkopfs in MIDCAD zu simulieren. Die Anforderungen und die Modellierungsmethode werden im nachfolgenden Abschnitt dargestellt.

Für jedes vereinfachte Geometriemodell des Bestückkopfs muss auf dem Saugnapf ein zusätzliches Koordinatensystem definiert werden, das zusammen mit dem Koordinatensystem *GREIF* die Position des Bestückkopfs bei der Überprüfung bestimmt. Eine Besonderheit bei der Modellierung des Geometriemodells für den fünffachsignen Bestückkopf ist, dass nicht nur die Länge und der Winkel der zusätzlichen Rotationsachse sondern auch die Position des zusätzlichen Koordinatensystems auf dem Saugnapf parametrisch definiert werden müssen. Bei der Bestückung jedes Bauelements ändert sich der Winkel der Rotationsachse automatisch in Abhängigkeit von den Bestückdaten. Dadurch kann die Hauptachse des Oberteils, wie in Bild 86(c) gezeigt, immer senkrecht (Z-Richtung) bleiben.

Bild 86 zeigt die wichtigen Schritte bei der statischen Kollisionsprüfung in MIDCAD. In Bild 86(a) werden der Typ und die Parameter der Bestückköpfe eingestellt. Gleichzeitig können auch die wichtigen bestückbezogenen Kenngrößen (siehe Tabelle 5) gemäß der Definition des Bestückkopfs definiert werden. Bild 86(b) zeigt den Zwischenzustand bei der Kollisionsprüfung an einem Beispiel. Durch die automatische Platzierung des Geometriemodells des Bestückkopfs auf jede Bestückposition des Bauelements und mit Hilfe der integrierten Funktionen zur Detektion von Volumendurchdringungen werden mögliche Kollisionen festgestellt. Das Ergebnis der statischen Kollisionsprüfung des Beispiels ist in Bild 86(c) gezeigt.

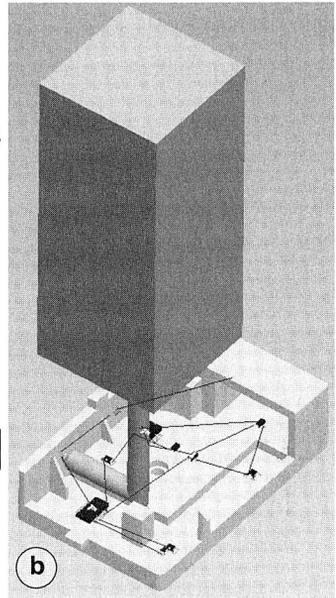
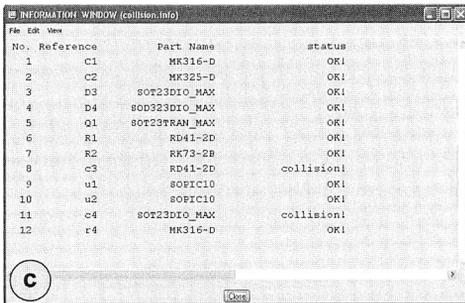
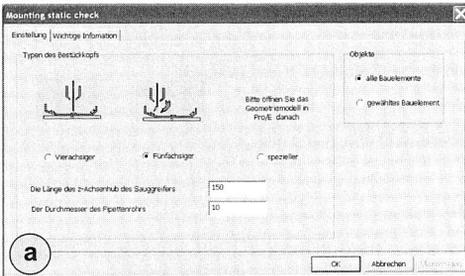


Bild 86: Statische Kollisionsprüfung in MIDCAD

# 8 Realisierung eines integrierten CAD-Systems für räumliche spritzgegossene Schaltungsträger

## 8.1 Prototyp eines integrierten MIDCAD-Systems

### 8.1.1 Architektur des Software-Prototyps

Bild 87 vermittelt einen Überblick über die dem integrierten MIDCAD-System zugrunde liegenden Komponenten der gesamten Systemarchitektur, die der Konzeption und dem Aufbau des Systems in Kapitel 3 entspricht. Die Architektur bietet eine benutzer- und aufgabenangepasste Unterstützung für die MID-Konstruktion an. Auf Basis der in Kapitel 4 bis 7 vorgestellten Arbeiten und Ergebnissen wird ein integriertes MIDCAD-System entsprechend der Architektur aufgebaut. Das MIDCAD-System läuft auf Basis von Pro/ENGINEER als Plugin, die Benutzerinteraktion erfolgt über eine Erweiterung der Benutzungsschnittstelle von Pro/ENGINEER und damit in der für den Konstrukteur vertrauten Arbeitsumgebung. Die grafische Benutzeroberfläche wurde für MIDCAD um ein zusätzliches Menü in Pro/ENGINEER erweitert, aus dem alle Funktionen von MIDCAD gestartet werden können (siehe Bild 88).

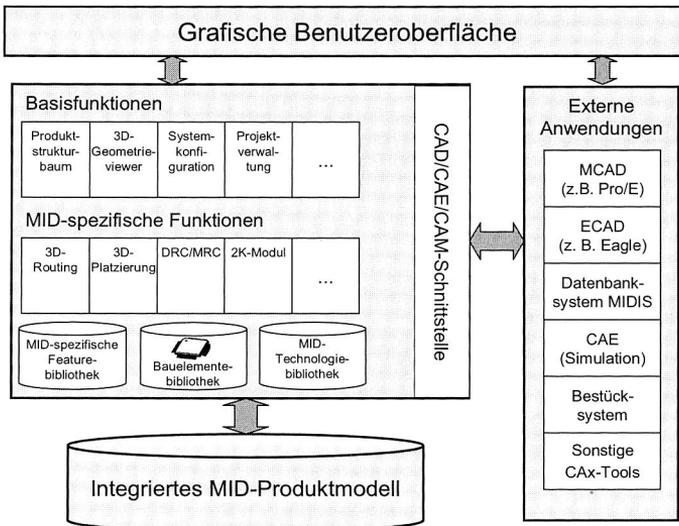


Bild 87: Architektur des integrierten MIDCAD-Systems

Das MIDCAD-System begleitet den Anwender während der MID-Konstruktion und stellt ihm eine hohe Gestaltungsfreiheit sowie ein spezielles 3D-MID-Design in der Phase der Systemintegration zur Verfügung. Die Bereitstellung der Basisfunktionen, MID-spezifischer Funktionen sowie einer Reihe der relevanten Bibliotheken im MIDCAD-

System bieten dem Entwickler eine effiziente Unterstützung bei der Konstruktion räumlicher spritzgegossener Schaltungsträger.

Das integrierte, MID-spezifische Produktmodell bietet gegenüber dem klassischen dokumentenorientierten Ansatz von PDM-Systemen eine weit feinere Verknüpfung von Daten und fachbereichsspezifischen Partialmodellen. Darüber hinaus bildet es die enge Verbindung der eingesetzten Werkzeuge im Entwicklungsablauf ab.

Externe Anwendungen bezeichnen dabei bestehende, eigenständige Programme in Bezug auf die MID-Entwicklung. Die Kopplung mit externen Anwendungen wird durch die entsprechenden Datenschnittstellen ermöglicht. Die von externen Anwendungen erzeugten Daten und Dateien können dadurch in das MIDCAD-System eingelesen und verarbeitet werden. Andererseits ermöglichen die Datenschnittstellen den Datenzugriff von außen auf Daten und Dateien im Systemverbund.

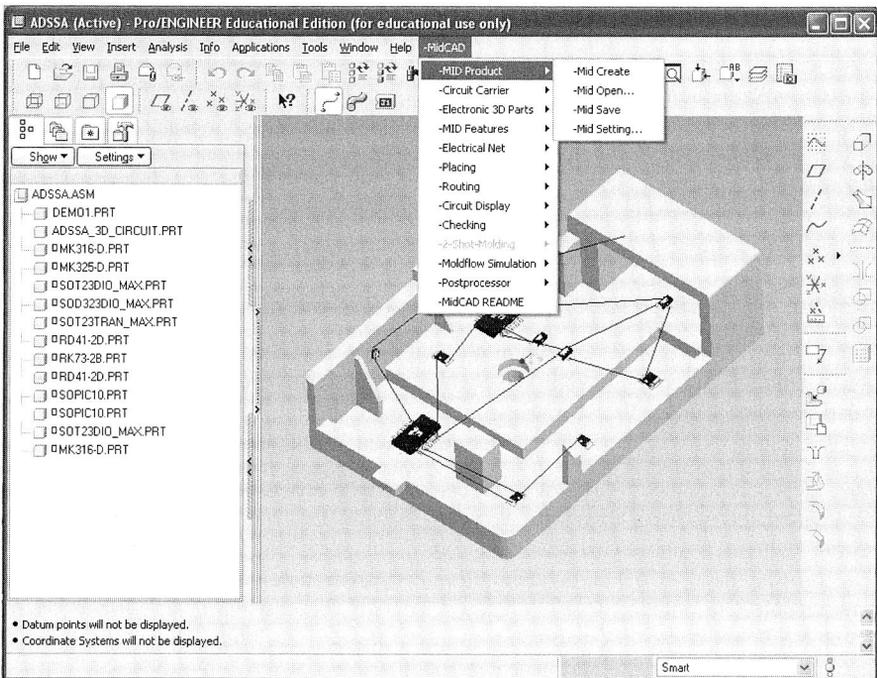


Bild 88: Das neu erzeugte Menü in Pro/ENGINEER als grafische Benutzeroberfläche von MIDCAD

### 8.1.2 Verknüpfungen zwischen MIDCAD und externen Anwendungen

Die externen Anwendungen können nach der Art der Verknüpfung mit MIDCAD unterschieden werden. Man unterscheidet zwischen einer direkten Verknüpfung für integrier-

te Anwendungen und einer losen Verknüpfung für separate Anwendungen. Die integrierten externen Anwendungen können über die neutrale, objektorientierte Schnittstellenbeschreibung gekapselt und in das MIDCAD-System integriert werden. Die Kommunikation mit separaten externen Anwendungen erfolgt ausschließlich über Austauschdateien, an die die entsprechenden Schnittstellen in MIDCAD gekoppelt werden.

### **Schnittstelle zwischen MIDCAD und ECAD**

Gemäß der Vorstellung in Kapitel 3.2.3 und dem V-Modell wird das ECAD-System in der domänenspezifischen Phase der MID-Konstruktion eingesetzt. Aufgrund der 3D-Layoutfunktionen in MIDCAD ist das ECAD-System hauptsächlich für den Schaltplangentwurf zuständig. Der Layoutentwurf erfolgt stattdessen später im Dreidimensionalen in MIDCAD. Daher konzentriert sich der Datenaustausch zwischen MIDCAD und ECAD nur auf die Informationen, die für den Schaltplangentwurf relevant sind.

Als eine separate externe Anwendung ist ECAD im jetzigen Entwicklungsstand mit MIDCAD durch Austauschdateien verbunden. Die aus dem ECAD-System EAGLE exportierten Komponenten- und Netzlisten können durch die entsprechenden Schnittstellen in MIDCAD eingelesen bzw. importiert werden. Wegen des nativen Dateiformats sind diese systemspezifischen Schnittstellen für andere ECAD-Systeme nicht geeignet. Da sich die relevanten Datenstrukturen im Produktmodell an AP210 orientieren, bietet MIDCAD eine weitere systemneutrale Schnittstelle. Dadurch werden die relevanten, elektronischen Informationen aus der nach AP210 erzeugten Datei in MIDCAD eingelesen.

### **Schnittstelle zwischen MIDCAD und anderen externen Anwendungen**

Neben dem ECAD-System können in MIDCAD noch eine Reihe weiterer externer Anwendungen eingebunden werden, so etwa zu Simulations- bzw. Berechnungszwecken oder weitere MID-spezifische Anwendungen. Auch hier besteht die Möglichkeit, diese entweder in MIDCAD zu integrieren oder durch Austauschdateien zu verbinden.

MPI stellt ein wichtiges CAE-Werkzeug in der MID-Entwicklung zur Simulation des Spritzgussprozesses für räumliche Schaltungsträger dar. Wie in Kapitel 7 dargestellt, wurde MPI durch eine selbst entwickelte Programmierschnittstelle auf Basis der OLE-Technologie mit MIDCAD verbunden. MPI zählt daher zu den integrierten externen Anwendungen. Die anderen CAE-Werkzeuge können voraussichtlich auf gleiche Weise durch die Entwicklung der entsprechenden Schnittstellen als integrierte externe Anwendungen mit MIDCAD systemtechnisch verknüpft werden.

Zum jetzigen Entwicklungsstand werden die anderen externen Anwendungen über ihre Ein- und Ausgabedateien angesprochen, andere MCAD-Systeme wie z. B. CATIA hauptsächlich über deren STEP-Datenschnittstellen. Das MID-Bestücksystem wird nach der Vorstellung in Kapitel 7.2 nur als separate externe Anwendung mit MIDCAD verbunden.

### 8.1.3 Integration von CAx-Systemen zu einer MID-Konstruktionsumgebung

Basierend auf den Resultaten der bisherigen Arbeit wird eine grundlegende integrierte Konstruktionsumgebung für MID-Produkte durch Integration von CAx-Systemen entwickelt, in der ein ECAD-, ein MCAD- und ein CAE-System sowie ein MID-Bestücksystem bereits mit dem MIDCAD-System verknüpft sind. Die Verknüpfung mit weiteren beteiligten Systemen, speziell mit Systemen aus Gestaltung und Simulation, soll bidirektional und systemneutral erfolgen, so dass eine möglichst große Anzahl von CAx-Systemen integriert werden kann. Dies können beispielsweise das Datenbanksystem MIDIS oder Anwendungen zur Kinematik- und Thermoanalyse sein.

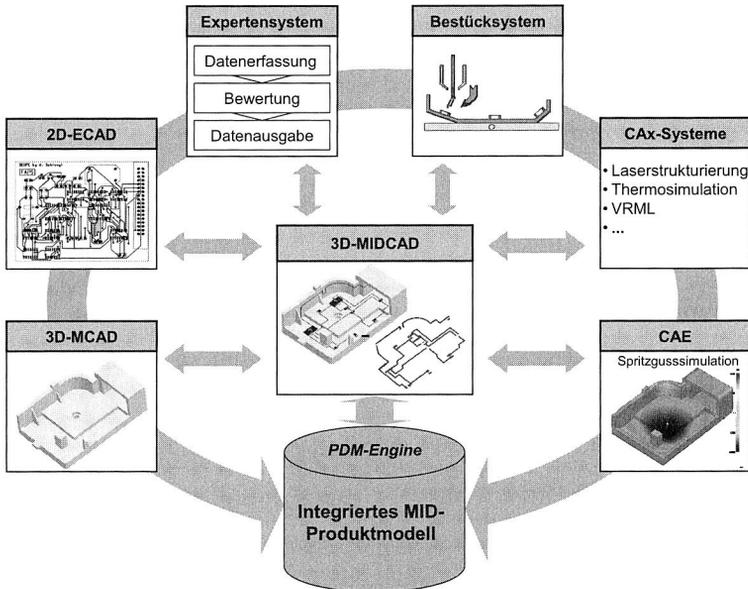


Bild 89: Eine MID-Konstruktionsumgebung durch Integration von CAx-Systemen

## 8.2 Vorgehensweise des 3D-MID-Designs im MIDCAD-System

### 8.2.1 MID-Konstruktion in der frühen Konzeptphase und der domänenspezifischen Entwurfsphase

In der frühen Konzeptphase einer Produktentwicklung wird ein wissenbasiertes MID-Expertensystem [72] zur Entscheidung der Anwendung der MID-Technologie eingesetzt. Zur Unterstützung bei der Auswahl eines geeigneten MID-Herstellungsverfahrens wurde das Expertensystem in das bestehende Datenbanksystem MIDIS integriert, in dem umfangreiche Informationen über MID-Applikationen, MID-Prozessketten sowie Werkstoffdaten abgelegt sind.

Auf der Basis dieses domänenübergreifenden Lösungskonzepts erfolgt der weitere domänenspezifische Entwurf meist in getrennten MCAD- und ECAD-Systemen. Während die rein mechanisch-geometrischen Strukturen des 3D-Schaltungsträgers in einem MCAD-System (z. B. Pro/ENGINEER) entworfen und modelliert (siehe Bild 90(1)) werden, erfolgt der elektronische Entwurf (z. B. Schaltplanentwurf) im ECAD-System. Der Layoutentwurf erfolgt später im Dreidimensionalen im MIDCAD-System.

### 8.2.2 3D-MID-Design in der Phase der Systemintegration in MIDCAD

Die Phase der Integration von mechanischen und elektronischen Funktionen ist während des 3D-MID-Designs eine wichtige und entscheidende Phase. Gemäß der Konzeption des Aufbaus von MIDCAD unterstützt MIDCAD in dieser Phase hauptsächlich die Konstruktion, Modellbildung und -analyse der MID-Produkte. Bild 90 zeigt, wie einige wichtige Funktionen in MIDCAD den Konstrukteur bei der Entwicklung einer MID-Komponente unterstützen. Im Folgenden wird die allgemeine Vorgehensweise des MID-Designs in MIDCAD vorgestellt.

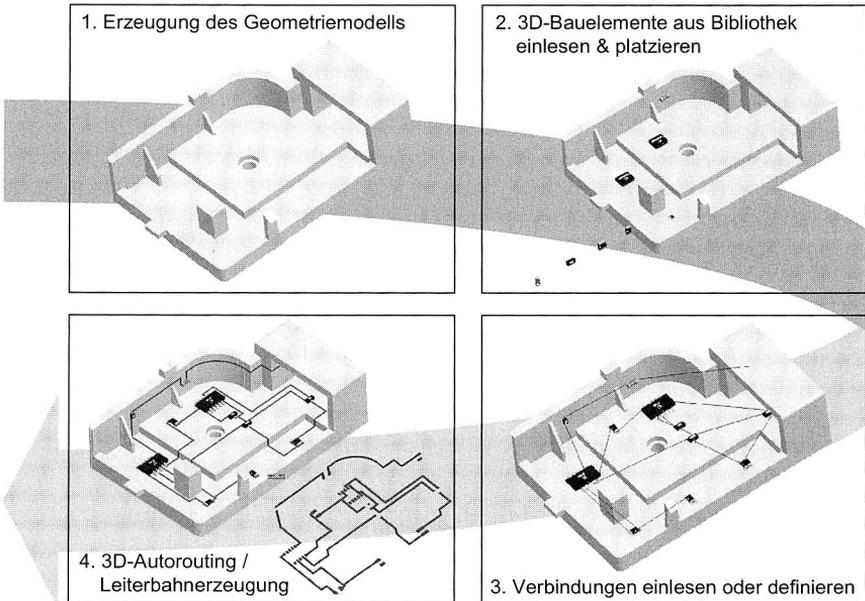


Bild 90: Konstruktion eines MID-Produkts in MIDCAD [33]

#### Schritt 1: Initialisierung eines neuen MID-Designs in MIDCAD

Zu Beginn des MID-Designs wird ein neues Projekt in MIDCAD angelegt. Mit Hilfe eines Eingabefensters können der Projektname und die relevanten Informationen des MID-Produkts (z. B. Herstellungsverfahren usw.) definiert werden. Gleichzeitig wird automa-

tisch in Pro/ENGINEER eine neue entsprechende Montagedatei erzeugt, in der das 3D-MID-Design stattfindet.

#### *Schritt 2: Einlesen der nötigen Daten aus den vorhergehenden Entwicklungsphasen*

Die grundlegenden, von MIDCAD benötigten Daten sind die Geometriedaten des räumlichen Schaltungsträgers sowie die aus der Elektronikentwicklung hervorgegangenen Netz- und Komponentenlisten des Schaltplans. Die Modellierung der originalen, mechanisch-geometrischen Struktur des Schaltungsträgers wird direkt in Pro/ENGINEER vorgenommen. Daher sind keine zusätzlichen Datenaustauschvorgänge zwischen Pro/ENGINEER und MIDCAD nötig. Die in ECAD definierten Netz- und Komponentenlisten werden durch die entwickelte Schnittstelle in MIDCAD importiert. Über die Importmöglichkeit hinaus bietet auch MIDCAD selbst einige grundlegende Funktionen zur Definition und Bearbeitung von Netz- und Komponentenlisten.

Nach dem Einlesen der Komponentenliste werden die entsprechenden Bauelemente automatisch aus der 3D-Bauelementebibliothek ausgewählt und ins Produkt eingefügt (siehe Bild 90(2)). Nach dem Einlesen der Netzliste werden die logischen Verbindungen zwischen den Anschlusspunkten der Bauelemente durch Luftlinien dargestellt (siehe Bild 90(3)).

#### *Schritt 3: Weitere geometrische Gestaltung durch Einfügen der MID-spezifischen Features und Veränderung der entsprechenden Netzliste*

Auf Basis des originalen mechanisch-geometrischen Modells des Schaltungsträgers werden die besonderen Strukturen, die nur in MID-Produkten existieren, durch Einfügen der MID-spezifischen Features (*Typ I*, z. B. Durchkontaktierung) gestaltet, um die stoffliche elektronische Funktionsintegration auf dem Schaltungsträger zu realisieren. Auf Basis des selbst entwickelten Feature-Baums können nicht nur die MID-spezifischen Features und die allgemeinen MID-Features definiert und übersichtlich verwaltet, sondern auch deren nichtgeometrische Eigenschaften spezifiziert werden.

Da diese MID-spezifischen Features wie herkömmliche elektronische Bauelemente funktionieren, ist eine Änderung der relevanten Verbindungen nach dem Einfügen der MID-spezifischen Features nötig.

#### *Schritt 4: 3D-Platzierung, -Routing und -Leiterbahnerzeugung*

Zwei wichtige Aufgaben für die MID-Konstruktion stellen die 3D-Platzierung und das 3D-Routing dar. Zuerst erfolgt die 3D-Platzierung der Bauelemente interaktiv mit Hilfe der Maus. Nach der Platzierung der Bauelemente erfolgt das 3D-Routing mit Hilfe der automatischen und manuellen Routingfunktionen. Auf Basis des Routingergebnisses wird das 3D-Geometriemodell der Leiterbahnen nach Vorgabe der Leiterbahnquerschnitte automatisch erzeugt (siehe Bild 90(4)). Die Positionen der Bauelemente können jederzeit nach den Anforderungen des Routings oder der DRG/MRC manuell verändert werden. Bei Bedarf können auch die Leiterbahnen manuell geändert bzw. durch nochmaliges Autorouting erneut verlegt werden. Die konkreten Vorgehensweisen bei

3D-Platzierung, -Routing und -Leiterbahnerzeugung wurden in den vorhergehenden Kapiteln erläutert.

*Schritt 5: Weiterverarbeitung und Übergabe der relevanten Daten für integrierte oder separate, externe Anwendungen*

- Ableitung des Simulationsmodells zur Spritzgussimulation

Für räumliche spritzgegossene Schaltungsträger ist die Spritzgussimulation in der fertigungsorientierten Konstruktion von großer Bedeutung. Die Ableitung des Simulationsmodells sowie die Integration der Spritzgussimulation im 3D-MID-Design wurden bereits in früheren Kapiteln im Detail dargestellt.

Wird das Produkt im 2K-Verfahren hergestellt, können die Geometriemodelle des ersten und zweiten Schusses durch die hierfür entwickelte, spezielle Funktion auf Basis der Geometrie des Schaltungsträgers und der 3D-Leiterbahnen automatisch erzeugt werden. Die Geometriemodelle liefern somit nicht nur bereits grundlegende Informationen für das nachfolgende Molddesign, sondern auch die geometrischen Daten der entsprechenden Simulationsmodelle.

- Ableitung der Bestückdaten für das MID-Bestücksystem

Auf Basis des Platzierungsergebnisses werden die dreidimensionalen Bestückpositionen der Bauelemente abgeleitet. Darauf aufbauend kann eine statische Kollisionsprüfung durchgeführt werden. Zusammen mit der 3D-Platzierung können kollisionsfreie Bestückdaten für das MID-Bestücksystem abgeleitet werden.

**Folgende Anwendungsmöglichkeiten sind denkbar (siehe auch Bild 89):**

- Einlesen der Daten aus der Datenbank MIDIS

Eine Weiterverwendung der relevanten Daten aus dem Beschreibungsmodell des MID-Expertensystems für MIDCAD ist anzustreben. Hierzu bedarf es noch der Entwicklung einer entsprechenden Schnittstelle zwischen beiden Systemen. Diese Daten sollen nicht manuell, sondern automatisch durch die Schnittstelle in MIDCAD eingelesen werden, um Fehler durch eine falsche Dateneingabe zu vermeiden und die Einheitlichkeit der Daten sicherzustellen.

- Ableitung der Daten für die Laserstrukturierung

Eine weitere Anwendungsmöglichkeit ist die Laserstrukturierung. Hier werden die Leiterbahngeometrien durch die Ansteuerung des Scanners erzeugt und nicht durch die Geometrie eines Spritzgießwerkzeuges vorgegeben. Deswegen ist die Ableitung der geometrischen Daten der 3D-Leiterbahnen für die Laserstrukturierung von großer Bedeutung, damit die entsprechenden Steuerprogramme erzeugt werden können.

- Verwaltung der Produktdaten im PDM-System bzw. Editor

Das integrierte Produktmodell und die darin enthaltenen Daten können an ein PDM-System übergeben werden, mit dem die Daten während des gesamten MID-Entwicklungsprozesses angelegt, editiert und gelöscht werden können. Das PDM-System bzw. der Editor lassen sich als eine integrierte, externe Anwendung (z. B. auf Basis von *Windchill* und *PDMLink*<sup>8</sup>) aus dem Menü von MIDCAD starten und sind für die Verwaltung der Produktdaten und Dateien vorgesehen.

### 8.2.3 Integrierte Strukturoptimierung beim 3D-MID-Design in MIDCAD

Die direkte stoffliche Integration elektronischer und mechanischer Funktionen auf räumlichen spritzgegossenen Schaltungsträgern bietet erhebliches Potenzial zur Produktgestaltung und Rationalisierung. Durch die Anwendung der MID-Technologie können während des 3D-MID-Designs mit neuen Möglichkeiten innovativer Produktgestaltung die Miniaturisierung der MID-Komponente und die Einsparung mechanischer Bauteile erzielt werden.

Deswegen konzentriert sich das 3D-MID-Design in der Phase der Systemintegration nicht einfach nur auf die Definition der Bauelemente und Leiterbahnverläufe auf bzw. innerhalb des 3D-Schaltungsträgers, sondern auch, unter der Berücksichtigung verschiedener interaktiver Einflussfaktoren, auf die Optimierung der gesamten Struktur der MID-Komponente. Die originale geometrische Struktur des Schaltungsträgers wird nicht nur durch die stoffliche, elektronische Funktionsintegration geändert, sondern auch entsprechend der Belange der Berechnung, Herstellung und Montage optimiert. Solche Optimierungen brauchen starke Modellierungsfunktionen und sollten daher basierend auf einem parametrischen Geometriemodell mit den vollkommenen historischen Modellierungsinformationen durchgeführt werden. Aus diesem Grund hat das auf der Basis von Pro/ENGINEER entwickelte MIDCAD-System, das eine integrierte Strukturoptimierung im 3D-MID-Design erlaubt, einen großen Vorteil. Die Strukturoptimierung trägt sowohl zur Verbesserung der Produktqualität und zur Sicherstellung der Fertigbarkeit, als auch zur Miniaturisierung der gesamten, mechatronischen Struktur der MID-Komponente bei.

Während des 3D-MID-Designs werden verschiedene Optimierungsvorgänge in aufeinander folgenden Schritten durchlaufen, um zusammen eine Optimierung der Gesamtstruktur zu erreichen. Im Folgenden werden diese Optimierungsmöglichkeiten beim 3D-MID-Design zusammengefasst.

- Bei dem Einfügen bzw. der Definition der MID-spezifischen Features (*Typ 1*, z. B. Durchkontaktierung) auf dem Schaltungsträger wird gleichzeitig die geometrische Struktur des Schaltungsträgers durch die stoffliche Integration elektronischer Funkti-

---

<sup>8</sup> Das PDM-System *Windchill* und die Entwicklungsschnittstelle *PDMLink 8.0* von PTC stellen Datenverwaltungslösungen zur Verwaltung von Produktentwicklungsprozessen auf Unternehmensebene zur Verfügung.

onen verändert. Die Lage und die Dimensionen der speziellen Features sind zwar hauptsächlich von den Designregeln der entsprechenden elektronischen Funktion abhängig, werden aber auch von der originalen Struktur des Schaltungsträgers beeinflusst. Umgekehrt soll sich die geometrische Struktur des Schaltungsträgers an das Einfügen dieser spezifischen Features anpassen. Es besteht eine Optimierungsmöglichkeit der Strukturen von beiden Seiten.

- Während die 3D-Platzierung stark von der geometrischen Struktur des Schaltungsträgers bzw. des Gehäuses abhängig ist, kann die 3D-Platzierung wiederum die geometrische Gestalt des Schaltungsträgers beeinflussen. Die Strukturoptimierung des Schaltungsträgers ist auch bei der 3D-Platzierung möglich.
- Das Ergebnis des 3D-Routings und die entsprechenden 3D-Leiterbahnen hängen auch von der geometrischen Struktur des Schaltungsträgers ab. Der Schaltungsträger muss genügend freien Raum für das erfolgreiche 3D-Routing anbieten. Als wichtiges MID-spezifisches Feature sind die 3D-Leiterbahnen in die Struktur des Schaltungsträgers stofflich integriert. Es besteht ein Optimierungspotential der Strukturen der 3D-Leiterbahnen und des Schaltungsträgers.
- Durch die Integration der Spritzgusssimulation können einerseits die analyserelevanten Randbedingungen (z. B. Angusspunkte) optimiert werden, andererseits kann die geometrische Struktur des Schaltungsträger nach spritzguss- und metallisierungsgerechten Gestaltungsrichtlinien optimiert werden (siehe Kapitel 7.1.4 und 7.1.5). Beim 2K-Verfahren ist die Optimierung der geometrischen Struktur des ersten Schusses durchzuführen, um genügend Haftung zwischen beiden Schüssen zu erreichen (siehe Kapitel 5.4.3).
- Durch die Entwicklung der CAD/CAM-Kette zwischen MIDCAD und dem MID-Bestücksystem kann einerseits das 3D-Platzierungsergebnis nach der statischen Kollisionsprüfung optimiert werden, um die Bestückbarkeit der platzierten, elektronischen Bauelemente sicherzustellen. Andererseits kann die Struktur des Schaltungsträgers nach bestückbezogenen Kenngrößen überprüft und optimiert werden, z. B. durch die Optimierung der Hauptmaße und der Lage der Hauptbestückfläche.

Diese Optimierungsvorgänge unterteilen sich in zwei Typen: Zum einen in die elektronisch-mechanisch integrierte Strukturoptimierung, die MID-spezifisch und an den Anforderungen der stofflichen Integration mechanischer und elektronischer Funktionen auf dem Schaltungsträger orientiert ist; zum anderen in die mechanisch-geometrische Strukturoptimierung, die sich an den Belangen der Berechnung, Herstellung und Montage orientiert. Das Integrationsdesign in MIDCAD ist gleichsam ein Optimierungsdesign, das sowohl das mechatronisch-orientierte Layout als auch die geometrische Struktur optimiert.

Gemäß der Vorstellung der Feature-Technologie in Kapitel 6.5 und der automatischen Optimierungsmethode in Kapitel 7.1.5 spielt die Feature-Operation zur Realisierung der oben genannten Optimierungsvorgänge in MIDCAD eine wichtige Rolle (siehe Bild 91).

Da jedes definierte MID-Feature, ob allgemeines Feature oder MID-spezifisches Feature, eine bestimmte geometrische Struktur des Schaltungsträgers repräsentiert, entspricht die Strukturoptimierung den Änderungen der MID-Features. Nach der Spezifizierung der wichtigen formrelevanten und lagerelevanten Dimensionen und Parameter des Features als dessen Eigenschaften kann die Strukturoptimierung durch die Verwendung des parametrischen Modellierungsansatzes durchgeführt werden, wenn die Topologie bei der Optimierung nicht verändert ist. Durch die zusätzliche Definition der Veränderungsstrategie und der Schrittweite kann die Strukturoptimierung nach der Vorstellung in Kapitel 7.1.5 automatisch durchgeführt werden. Die topologische Strukturoptimierung ist durch das Einfügen eines neuen MID-Features oder durch eine Löschung eines bereits definierten MID-Features gekennzeichnet.

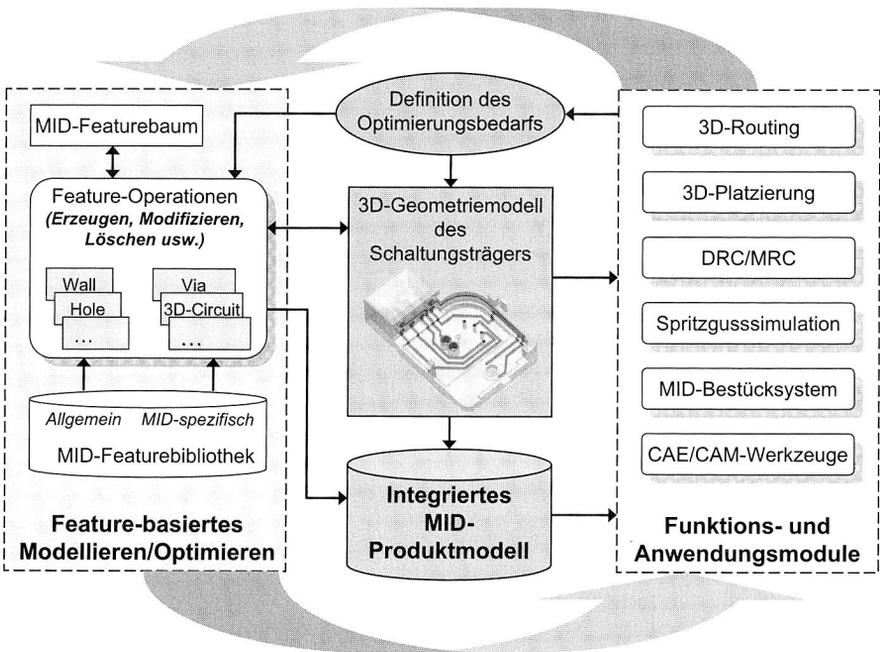


Bild 91: Strukturoptimierung des Schaltungsträgers auf Basis der Feature-Technologie

Eine weitere Entwicklungsmöglichkeit ist, dass Features mit Wissen in Form relevanter Gestaltungsrichtlinien verknüpft werden. Dadurch können die Features selbst Vorschläge zur Strukturoptimierung machen („Design Advisor“), wobei erweiterte Wissenverarbeitungsmechanismen erforderlich sind.

## 8.3 Technischer und wirtschaftlicher Nutzen des integrierten MID-CAD-Systems

Im Folgenden werden die wissenschaftlichen Erkenntnisse und entwickelten Methoden bzw. Funktionen der vorliegenden Arbeit bewertet und gleichzeitig das MIDCAD-System im gesamten MID-Entwicklungsprozess positioniert.

### 8.3.1 Bewertung der technischen Vorteile

#### *Erfüllung der neuen Anforderungen aus der MID-Technologie*

Das integrierte MIDCAD-System in der vorliegenden Arbeit wurde gemäß den neuen Anforderungen an die MID-Konstruktion entwickelt. Es basiert auf dem kommerziellen 3D-MCAD-System Pro/ENGINEER und bietet gleichzeitig auf 3D erweiterte elektronikrelevante Funktionen (z. B. 3D-Layoutfunktionen) sowie eine Reihe MID-spezifischer Funktionen. Mit diesem Entwicklungskonzept gewährleistet das MIDCAD-System eine hohe geometrische Gestaltungsfreiheit bei gleichzeitiger Integration der elektronikrelevanten Konstruktion direkt in einer 3D-MCAD-Umgebung (im Vergleich zur konventionellen Lösung, Mechanik und Elektronik getrennt aufzubauen und dann das Board in das Gerät einzufügen). Durch die Integration bzw. Verknüpfung mit anderen CAX-Systemen, wie z. B. die Integration der Spritzgussimulation, ist die Anforderung nach der Berücksichtigung von fertigungsgerechten Konstruktionslinien erfüllt.

#### *Schwerpunkt auf bestimmte MID-Konstruktionsphase*

Gemäß der Konzeption von MIDCAD konzentriert sich MIDCAD hauptsächlich auf die Konstruktion, Modellbildung und -analyse von MID-Produkten in der Phase der Systemintegration. Die stoffliche Integration der elektronischen und mechanischen Funktionen durch die Definition der Bauelemente und 3D-Leiterbahnverläufe auf bzw. innerhalb des Schaltungsträgers und die Beschreibung solcher speziellen Strukturen durch ein integriertes MID-Produktmodell stehen im Vordergrund des entwickelten MIDCAD-Systems. Weitere Ziele waren, dass in den späten Phasen der Systemintegration durch die Integration bzw. Verknüpfung der relevanten externen Anwendungen wie z. B. Spritzgussimulation und Bestücksystem, eine fertigungsorientierte Strukturoptimierung einer MID-Komponente durchgeführt werden kann. Zur Unterstützung der gesamten MID-Konstruktionsphase sollen für den Aufbau einer integrierten Konstruktionsumgebung die externen Anwendungen aus der Montage, der Fertigung sowie der Simulationen bei verschiedenen Herstellungsverfahren weiter integriert bzw. verknüpft werden.

#### *Algorithmen für 3D-Leiterbahnverlegung bzw. -erzeugung*

Die Realisierung der dreidimensionalen routingrelevanten Funktionen, insbesondere der 3D-Autoroutingfunktion, ist einer der Schwerpunkte in der vorliegenden Arbeit und ein wesentliches Vorteil von MIDCAD. Dadurch können die komplexen und zeitaufwändigen Vorgänge der 3D-Leiterbahnverlegung und -erzeugung auf bzw. innerhalb des 3D-Schaltungsträgers deutlich leichter realisiert werden. Alternative 3D-Routingalgorithmen

bzw. -methoden wurden hierzu für das 3D-MID-Routing entwickelt und erforscht. Diese innovative Arbeit eröffnet nicht nur eine neue Dimension für das traditionelle 2D oder 2.5D Routingproblem im ECAD-Bereich, sondern bietet auch verschiedene Lösungsmöglichkeiten für das Routingproblem im realen dreidimensionalen Raum.

#### *Beschreibung des MID-Produktmodells*

Ein integriertes und einheitliches Produktmodell ist für eine qualitativ hochwertige, objektorientierte Strukturierung und Verwaltung der anfallenden Daten im Entwicklungsprozess von großer Bedeutung. Da die STEP-Norm universell einsetzbar ist, bietet sie viele Möglichkeiten, darauf aufbauend eigene Produktmodelle zu entwickeln. In der vorliegenden Arbeit wurde gezeigt, dass die Schemen von AP210 als Grundlage für die Beschreibung des MID-Produktmodells dienen können. Die Beschreibungen der speziellen Struktur der MID-Produkte und der Integration der mechanischen und elektronischen Daten wurden durch die Erweiterung und direkte Anwendung der Schemen von AP210 im Produktmodell realisiert. Die Modellierungstechnik Feature-Technologie wurde nicht nur für die Bereitstellung und Definition der MID-spezifischen Features, sondern auch für die Integration der CAE/CAM-Informationen im Produktmodell eingesetzt. Die Zusammenhänge zwischen AP210, Feature, den relevanten Partialmodellen und den Sichten wurden bei der Beschreibung des integrierten, domänenübergreifenden MID-Produktmodells dargestellt.

#### *Integration der Spritzgussimulation auf Basis einer systemtechnischen Verknüpfung*

Eine Verknüpfung zwischen einem CAD- und CAE-System hat in der Regel eine enge und eine lose Kopplung. Im Vergleich zur normalen losen Kopplung über Schnittstellendateien durch Modelltransformation hängt die enge Verknüpfung immer mit den folgenden Mechanismen zusammen: CORBA, OLE oder *Distributed Component Object Model* (DCOM). In der vorliegenden Arbeit wurde das Simulationswerkzeug MPI mit MIDCAD durch die Entwicklung der Programmierschnittstelle auf Basis der OLE-Technologie systemtechnisch eng verknüpft. Dabei erfolgt keine Datenübersetzung. Außerdem kann MIDCAD auf die Analysefunktion, die Einstellungsfunktionen der Randbedingung und die weiteren Funktionen von MPI zugreifen und sie direkt benutzen.

#### *Beschränkungen auf bestimmte Anwendungssoftware*

Da das MIDCAD-System auf dem kommerziellen 3D-MCAD-System Pro/ENGINEER basiert, muss die originale Geometrie des Schaltungsträgers über Datenschnittstellen (z. B. STEP-Datei) in MIDCAD importiert werden, wenn der Schaltungsträger in einem anderen 3D-MCAD-System (z. B. CATIA) modelliert und entworfen wurde. Wegen der prototypischen Entwicklung bietet MIDCAD nur eine beispielhafte Verknüpfung mit dem ECAD-System EAGLE an; dadurch kann eine aus EAGLE exportierte Komponenten- und Netzlisten direkt eingelesen werden. Bei Verwendung einer anderen systemneutralen Schnittstelle gilt diese Beschränkung nicht, wenn die Informationen über den Schaltplan aus anderen ECAD-Systemen durch AP210-Konvertierung in eine STEP-Datei konvertiert werden.

### 8.3.2 Wirtschaftliche Bewertung

Die Aufwand-/Nutzen-Abschätzung ist für das entwickelte MIDCAD-System nicht allgemeingültig durchführbar, weil das prototypische Werkzeug nicht als kommerzielles Produkt vorliegt. Im Folgenden werden das erzielbare Nutzenpotenzial und der zu erwartende Aufwand bei Verwendung des integrierten MIDCAD-Systems gegenübergestellt.

Der Hauptaufwand ergibt sich insbesondere durch die Anschaffungskosten von MIDCAD bzw. Pro/ENGINEER und eine entsprechende Mitarbeiterschulung. Da es sich bei MIDCAD um ein Plugin von Pro/ENGINEER handelt, sind zum einen die Kosten für MIDCAD deutlich geringer als für ein getrenntes System; dies ist gerade für mittelständische Unternehmen von Vorteil. Zum anderen ist der Schulungsaufwand bei Verwendung von MIDCAD deutlich geringer, da die Arbeitsumgebung von Pro/ENGINEER vielen Anwendern bereits bekannt ist. Weitere, über die gesamte Nutzungszeit anfallende Aufwendungen sind z. B. die Kosten für die Entwicklung neuer Funktionen und Lösungen im System, die Anpassung des Systems an verschiedene Nutzungswünsche sowie Datenbankpflege und Erweiterung der verschiedenen Bibliotheken im System.

Das Nutzenpotenzial ist im 3D-MID-Design selbst zu sehen. MIDCAD bietet eine hohe Gestaltungsfreiheit bei gleichzeitiger Integration mechanischer und elektronischer Funktionen aus ECAD und MCAD-Umgebungen an. Durch die Entwicklung der MID-spezifischen Funktionen (z. B. Autorouting) und eines integrierten Produktmodells werden die neuen Anforderungen der MID-Technologie erfüllt, die Entwicklungszeit verkürzt und der Entwicklungsaufwand gesenkt. Durch die Verknüpfung mit ECAD-, CAE- und Bestückssystemen wird außerdem die Kommunikation im Entwicklungsprozess verbessert und es werden Fehler beim Datenaustausch vermieden. Die Zusammenführung der bisher isoliert verwendeten Funktionalitäten aus ECAD und MCAD in MIDCAD liefert einen positiven Beitrag für eine domänenübergreifende MID-Konstruktion.

Abschließend lässt sich sagen, dass die Nutzeneffekte die Aufwendungen deutlich übertreffen und somit aus wirtschaftlichen Gesichtspunkten die Umsetzung bzw. Entwicklung des MIDCAD-Systems sinnvoll ist.

## 9 Zusammenfassung und Ausblick

Die Besonderheit der MID-Technologie – die stoffliche Integration mechanischer und elektronischer Funktionen – erfordert ein neuartiges CAD-System, um einen effizienten Konstruktionsprozess sicherstellen zu können. Im Rahmen der vorliegenden Arbeit wurde ein integriertes 3D-MIDCAD-System vorgestellt, welches die besonderen Anforderungen der MID-Produktentwicklung berücksichtigt und die bisher isoliert verwendeten Funktionalitäten aus dem ECAD- und MCAD-Bereich zusammenführt. Das Ziel der vorliegenden Arbeit war es, mit dem MIDCAD-System einen Beitrag für einen innovativen MID-Produktentwicklungsprozess in der Phase der Systemintegration zu liefern, um die Entwicklungszeit und damit die Kosten zu senken und gleichzeitig die Produktstruktur zu optimieren und damit die Produktqualität zu erhöhen.

Die Grundbestandteile des MIDCAD-Systems sind die beiden Kernfunktionen (3D-Platzierung und 3D-Routing) zur Definition der Bauelemente und Leiterbahnverläufe auf bzw. innerhalb eines 3D-Grundkörpers. Da das manuelle 3D-Routing zwischen verschiedenen Anschlusspunkten ein komplexer und zeitaufwändiger Vorgang ist, ist die Realisierung des 3D-Autoroutings ein wesentlicher Vorteil des MIDCAD-Systems. Durch die Entwicklung weiterer MID-spezifischer Funktionen, wie z. B. der automatisierte Erzeugung der Geometriemodelle des ersten und zweiten Schusses für das 2K-Verfahren, kann MIDCAD aufgrund seiner Besonderheit für die MID-Technologie die Konstruktion von MID-Produkten deutlich effektiver als andere Systeme unterstützen.

Die Grundlage zur Beschreibung von MID-Produkten und zur Speicherung der Produktdaten stellt ein integriertes Produktmodell dar. Feature-Technologie, STEP-Norm, Partialmodelle und verschiedene „Sichten“ werden für die Entwicklung des integrierten Produktmodells eingesetzt. Die Beschreibung der horizontalen Integration elektronischer und mechanischer Daten im Produktmodell basiert auf dem AP210 Schema. Zur vertikalen Integration analyse- und montagerelevanter Informationen im integrierten MID-Produktmodell wird die Feature-Technologie angewendet. In Vergleich zu den allgemeinen Features zeichnen sich die neu definierten MID-spezifischen Features durch zusätzlich enthaltene elektronische Funktionen aus. Weiterhin ist die Bereitstellung einer 3D-Bauelementebibliothek eine wichtige Kernfunktion von MIDCAD. Die Entwicklung der Bibliothek und die Beschreibung der Bauelemente orientieren sich an AP210. Dafür wurden auch ein Bauelementemodellgenerator und eine 3D-Gehäuse- und Funktionsbibliothek entwickelt.

In MIDCAD sind externe Anwendungen in verschiedenen MID-Entwicklungsphasen auf Basis des integrierten Produktmodells über entsprechende Schnittstellen integrierbar. In der vorliegenden Arbeit wurden in MIDCAD nicht nur die bisher getrennten MCAD- und ECAD-Umgebungen zusammengeführt, sondern auch das CAE-System MPI und das MID-Bestücksystem gemäß den Konzeptionen „*Design for Manufacturing*“ und „*Design for Mouldability*“ durch die entwickelten Schnittstellen mit MIDCAD verbunden, um eine optimale Konstruktion und Fertigbarkeit des Produkts sicherzustellen [121].

Die Vorteile des integrierten MIDCAD-Systems werden zum Schluss an einer beispielhaften Anwendung präsentiert. Es wurde gezeigt, wie ein 3D-MID-Design in der Phase der Systemintegration schnell, effektiv, mit einer hohen Qualität und einer optimierten, mechanisch-elektronischen Struktur durchgeführt werden kann.

Die vorliegenden Ergebnisse, die mit der Erstellung des integrierten Produktmodells und MIDCAD-Systems erzielt worden sind, bieten eine gute Grundlage für den Aufbau einer integrierten Konstruktionsumgebung für den gesamten MID-Entwicklungsprozess. In dieser Konstruktionsumgebung sind ein ECAD-, ein MCAD- und das CAE-System MPI sowie das MID-Bestücksystem bereits mit dem MIDCAD-System verknüpft. Ein Ziel weitergehender Forschungsarbeiten wird es deshalb sein, alle MID-relevanten Systeme, die über den bisherigen Funktionsumfang des MIDCAD-Systems hinausgehen, mit ihren entstehenden Daten und Dateien eng oder lose zu integrieren. Dies können beispielsweise ein PDM-System, ein Laserstrukturierungssystem oder Kinematik- und Thermosimulationen sein.

Eine Funktionserweiterung des MIDCAD-Systems in verschiedene Richtungen ist ebenfalls denkbar. So könnte zum Beispiel eine Funktion zur automatischen 3D-Platzierung der Bauelemente entwickelt werden. Zur kommerziellen Anwendung sollten auch eine Reihe weiterer grundlegender Funktionen und eine robuste Bauelemente-, Feature-, MID-Technologie- bzw. MID-Materialbibliothek angeboten werden. Zwar konzentriert sich das MIDCAD-System im gegenwärtigen Entwicklungsstand vor allem auf die Konstruktion von MID-Komponenten, ein weiteres Anwendungsgebiet könnte aber z. B. das Leitungsverlegungsproblem auf bzw. innerhalb elektrischer Geräte und Anlagen sein, so wie etwa bei der MakroMID-Technologie [28] als eine neue elektrische Verbindungslösung für Bordnetze in der Automobilindustrie.

Mit der zunehmenden Verbreitung des Internets bzw. dem darauf aufbauenden Mehrwertdienst *World Wide Web* (WWW) bietet ein CAD-System großes Potenzial und eine hohe Leistungsfähigkeit aufgrund von Funktionalitäten, die durch die Vernetzung von Rechnern angeboten werden. Durch die neuen Datenaustauschstandards (z. B. UML, XML [67]) und die heutige Internet- bzw. Programmieretechnik (z. B. COM, CORBA, JAVA usw.), ergeben sich interessante Möglichkeiten für MIDCAD zur Verknüpfung mit externen CAx-Anwendungen auf verschiedenen Rechnern an unterschiedlichen Orten über das Internet. Zum Beispiel kann das CAE-System MPI zur Spritzgussimulation auch über eine Netzwerkverbindung angesprochen werden. In diesem Fall ist eine zusätzliche Software für die Abwicklung der Kommunikation zwischen den Systemen erforderlich. Die Entwicklung des so genannten kollaborativen CAD-Systems ist einer der aktuellen Forschungsschwerpunkte im Bereich der CAD-Entwicklung [64]. Die hierarchische Entwicklungsmethode eines kollaborativen CAD-Systems, mit der auf Basis der Erweiterung der Prinzipien und Methoden von *Concurrent Engineering* die verschiedenen Werkzeuge dynamisch miteinander verknüpft werden können, wird in der Entwicklung des integrierten MIDCAD-Systems weiterhin eine wichtige Rolle spielen.

## Summary

The special characteristic of the MID technology, which is the direct material integration of mechanical and electronic functions, requires a new creative CAD system. In the context of the present work an integrated 3D-MIDCAD system has been introduced, which satisfies the special requirements of MID development and provides an integration of electronic and mechanical design in one system. The main aims of the research work are providing a high efficiency in the MID product development process in the phase of the system integration through the MIDCAD system, so that the development period can be shortened to cut prices. In addition, the structure of the product can be optimised, increasing the product quality at the same time.

The basic elements of the MIDCAD system are the two core functions 3D-placement and 3D-routing, for the definition of the electronic components and conductor path respectively on the surfaces of the 3D circuit carrier. Therefore the 3D autorouting function was developed. Since the 3D-routing is a complex and time-consuming procedure, the realisation of the 3D autorouting is an important advantage in the MIDCAD system. Furthermore some MID specific functions have been developed, e.g. the automatic generation of geometry models of the first and second shot to support the MID design using 2-shot-molding. Therefore the MIDCAD system can support the construction of MID products more effectively than other systems.

The basis for describing the MID products and the storage of the product information is an integrated product model. Some important techniques, such as multiple views, feature modeling, STEP and partial models, were used in the development of this integrated product model. The application objects and scheme of AP210 and their extensions were used to describe horizontal integration and form the basis of the whole integrated product model. In order to realize the vertical integration of CAE/CAM information, the feature technology is applied in the product model. In comparison to the common mechanical features, the new defined MID specific features contain additional electronic functions. Furthermore, some libraries, e.g. 3D electronic parts library, are compulsory in MIDCAD. The development of the 3D electronic parts library and the corresponding part data model were also oriented on AP210. In addition a part & package model generator as well as a 3D package library and a function library were developed.

In MIDCAD, external applications for the different MID development phases can be integrated on the basis of the integrated product model through the corresponding interfaces. In this research work not only the separated MCAD and ECAD design environment are merged in MIDCAD, but also the CAE system MPI and the MID placement equipment were integrated or connected successfully into MIDCAD to support the concept of design for mouldability and design for manufacturing. This enables the user to accomplish a manufacturing-oriented optimization and to guarantee the manufacturability of MID-products.

Finally, the advantages of the integrated MIDCAD systems were exemplified on an application. It has been presented how a 3D-MID design can be operated more effectively in the phase of the system integration and at the same time to ensure a high product quality and an optimised electromechanical structure.

The research results, which were achieved with the development of an integrated product model and the MIDCAD system, provide a good basis for the construction of an integrated design environment for the complete MID development process. In this design environment, the ECAD, MCAD, MPI as well as the MID specific placement system are already linked with the MIDCAD system. It is therefore a further target to integrate closely or loosely all MID relevant systems with their emerging data and files. This could be, for instance, the PDM system, the laser structuring system or kinematics and thermo simulations.

The extension of the functionality of the MIDCAD system in different aspects is also possible. For example, an automatic 3D-placement of the components could be developed. For the commercial application, the necessary basic functions and a robust part-, feature-, MID technology- and MID-material libraries have to be offered. The presented MIDCAD system concentrates now especially on the construction of the MID products, but further application areas could be the electronic layout problem within electrical instruments and systems, e.g. the Macro-MID technology as a new electrical connect method for cable and harness design in the automobile industry.

As the Internet is spreading all over the world, the CAD system has huge potential and a high capability because of the functions which can be offered through the network. There are a lot of possibilities for MIDCAD to link the external CAx applications on different computers at different places via the Internet, due to the new data exchange standards (e.g. UML, XML) and the latest Internet- and programming techniques (e.g. COM, CORBA, JAVA etc.). For example, the CAE system MPI for injection molding simulation can also be connected to MIDCAD via network. In this case, additional software development is needed to handle the communication between the systems. The development of so called collaborated CAD systems is one of the latest main research topics in the field of CAD. The hierarchic methodology of the collaborated CAD system, which is based on the extension of principles and methodologies of the concurrent engineering, will be used in the development of upcoming integrated MIDCAD systems.

## Abkürzungen

2D/3D	two- /three-dimensional
2K	Zweikomponentenspritzgussverfahren
AP	Application Protocol
API	Application Programming Interface
ARM	Application Reference Model
CAD	Computer Aided Design
CAE	Computer Aided Engineering
CAM	Computer Aided Manufacturing
CAX	Oberbegriff für rechnerunterstützte Systeme oder Gesamtheit aller CA-Systeme (CAD, CAE, CAM, usw.)
CE	Concurrent Engineering
CIM	Computer Integrated Manufacturing
CORBA	Common Object Request Broker Architecture
COM	Component Object Model
CPU	Central Processing Unit
DLL	Dynamic Link Library
DRC	Design Rule Check
DXF	Drawing Exchange Format
ECAD	Electronic-CAD
EDIF	Electronic Design Interchange Format
FEM	Finite-Elemente-Methode
ID	Identity
IDF	Intermediate Data Format
IGES	Initial Graphics Exchange Specification
IPC	Interprocess Communication
ISO	International Standard Organisation

MCAD	Mechanic-CAD
MCM	Multi-Chip Module
MFC	Microsoft Foundation Class
MID	Molded Interconnect Devices
MKS	Mehrkörpersimulation
MRC	Manufacturing Rule Check
MST	Minimum Spanning Tree
NC	Numerical Control
N. N.	unbekannter Autor (lateinisch: nomen nescio)
NP	Non-deterministic Polynomial time
OLE	Object Linking and Embedding
PCA/PCB	Printed Circuit Assembly/Board
PCK	Printed Circuit Board Kollmorgen
PDM	Product Data Management
RAM	Random Access Memory
SKW	Sankyo Kasei Wiring Board
SMD	Surface Mounted Devices
SMT	Steiner Minimum Tree or Surface Mount Technology
STEP	Standard for the Exchange of Product Model Data
STL	Stereolithography (Main use: 3D printers and rapid prototyping machines)
UML	Unified Modeling Language
VDI	Verein Deutscher Ingenieure
VRML	Virtual Reality Modelling Language
XML	Extensible Markup Language
WWW	World Wide Web

## Literaturverzeichnis

- [1] Adler, T.; Barke, E.: Single Step Current Driven Routing of Multiterminal Signal Nets for Analog Applications, in: Proceedings of the conference on Design, automation and test in Europe, Paris, France, 2000, S. 446-450.
- [2] Al-Ashaab, A.; Rodriguez, K.; Molina, A.; Aca, M.C.; Saeed, M.; Abdalla, H.: Internet-based Collaborative Design for an Injection-moulding System, in: Concurrent Engineering: Research and Application, Vol. 11 (2003), S. 289-297.
- [3] Anton, O.; Lercher, B.: Mechatronik: Integrator von CAx- und PDM-Tools, FORSOFT Publications, *iwb*, München, 2001.
- [4] Arnold, F.: MACAO-a journey into CAx interoperability and collaborative design, in: IEEE Conference on Information Visualization, London, GB, 2000, S. 557-562.
- [5] Arnold, F.; Janocha, T.; Kilb, T.; Swienczek, B.: Rendezvous der Monolithen: Integration heterogener CAx-Systeme, in: OBJEKTSpektrum, Vol. 6 (1999), S. 22-27.
- [6] Bajaj, M.; Peak, R.; Wilson, M.; Kim, I.; Thurman, T.; M.C.Jothishankar; Benda, M.; Ferreira, P.; Stori, J.: Towards Next-Generation Design-for-Manufacturability (DFM) Frameworks for Electronics Product Realization, in: 28th IEEE/CPMT/SEMI International Electronics Manufacturing Technology Symposium, San Jose, USA, 2003, S. 359-367.
- [7] Bentley, J.L.: Multidimensional Binary Search Trees Used for Associative Searching, in: Communications of the ACM, Vol. 18 (1975), S. 509-517.
- [8] Brand, A.: Prozesse und Systeme zur Bestückung räumlicher elektronischer Baugruppen (3D-MID), Dissertation, FAPS, Universität Erlangen-Nürnberg, 1996.
- [9] Bronsvort, W.F.; Noort, A.: Multiple-view feature modelling for integral product development, in: Computer-Aided Design, Vol. 34 (2004), S. 929-946.
- [10] Cage, W.G.; Smith II, R.J.: A Rectangle-Probe Router for Multilayer P. C. Boards, in: Proceedings of the 14th design Automation Conference, New Orleans, 1977, S. 13-23.
- [11] Chew, L.P.: Constrained delaunay triangulations, in: Algorithmica, Vol. 4 (1989), S. 97-108.
- [12] Clow, G.W.: A Global routing Algorithm for General Cells, in: Proc. of the ACM IEEE 21st Design Automation Conference, 1984, S. 45-51.

- [13] Cong, J.; Fang, J.; Khoo, K.-Y.: DUNE--A Multilayer Gridless Routing System, in: IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 20 (No. 5) (2001), S. 633-647.
- [14] Cong, J.; Fang, J.; Khoo, K.-Y.: DUNE: A Multi-layer Gridless Routing System with Wire Planning, in: Proc. Int. Symp. Physical Design, San Diego, CA., 2000, S. 12-18.
- [15] Cong, J.; Fang, J.; Khoo, K.-Y.: An implicit connection graph maze routing algorithm for ECO routing, in: Proc. Int. Conf. Computer-Aided Design, 1999, S. 163-167.
- [16] Cong, J.; Fang, J.; Khoo, K.-Y.: Via Design Rule Consideration in Multilayer Maze Routing Algorithms, in: IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 19 (No. 2) (2000), S. 215-223.
- [17] Cormen, T.H.; Leiserson, C.E.; Rivest, R.L.; Stein, C.: Introduction to Algorithms Second ed., The MIT Press, 2001.
- [18] Croxford, G.: Mastering the ECAD to MCAD Process, in: PTC/USER World Event, Orlando, Florida, USA, 2003.
- [19] Dai, W.W.-M.; Dayan, T.; Staepelaere, D.: Topological Routing in SURF: Generating a Rubber-Band Sketch, in: Proceedings of the 28th Design Automation Conference, 1991, S. 39-44.
- [20] Dayan, T.: Rubber-Band Based Topological Router, Dissertation, Computer Engineering, University of California Santa Cruz, 1997.
- [21] Deng, Y.-M.; Britton, G.A.; Lam, Y.C.; Tor, S.B.; MA, Y.S.: Feature-based CAD-CAE integration model for injection-moulded product design, in: International Journal of Production Research, Vol. 40 (2002), S. 3737-3750.
- [22] Deng, Y.-M.; Lam, Y.C.; Tor, S.B.; Britton, G.A.: A CAD-CAE Integrated Injection Molding Design System, in: Engineering with Computers, Vol. 18 (2002), S. 80-92.
- [23] Dohmen, W.: Interdisziplinäre Methoden für die integrierte Entwicklung komplexer mechatronischer Systeme, Dissertation, *iwb*, TU München, 2002.
- [24] Dyla, A.: Modell einer durchgängig rechnerbasierten Produktentwicklung, Dissertation, Institut für Maschinen- und Fahrzeugtechnik - Lehrstuhl für Maschinenelemente, TU München, 2002.
- [25] Feldmann, K.: Advance of MID by Innovative Products and Production Technologies, in: Proceedings of the 4th International Congress on Molded Interconnect Devices, Research Association Molded Interconnect Devices 3-D MID e.V., Erlangen, 2000, S. 1-12.

- [26] Feldmann, K.: Improving MID-Potentials by Optimised Product Design and Stable Manufacturing Processes, in: Proceedings of the 5th International Congress on Molded Interconnect Devices, Research Association Molded Interconnect Devices 3-D MID e.V., Erlangen, 2002, S. 1-13.
- [27] Feldmann, K.; Pfeffer, M.; Reihardt, A.: Creative Developments and innovative Technologies for the Further Success of MID, in: Proceedings of the 7th International Congress on Molded Interconnect Devices, Research Association Molded Interconnect Devices 3-D MID e.V., Fürth, 2006, S. 1-15.
- [28] Feldmann, K.; Schüßler, F.; Wölflick, P.: Welcome Address and Introduction: MID and Foil – Technologies for Flexible Circuit Carriers, in: Proceedings of the 6th International Congress on Molded Interconnect Devices, Research Association Molded Interconnect Devices 3-D MID e.V., Erlangen, 2004, S. 1-15.
- [29] Feldmann, K.; Shi, Y.; Zhuo, Y.: Three-Dimensional Automatic Routing in Design of Molded Interconnect Devices (MID), in: Production Engineering, Vol. X/1 (2003), S. 65-68.
- [30] Feldmann, K.; Shi, Y.; Zhuo, Y.; Meier, R.: An Automatic Routing System for 3-D MID Design, in: Proceedings of the 5th International Congress on Molded Interconnect Devices, Research Association Molded Interconnect Devices 3-D MID e.V., Erlangen, 2002, S. 143-153.
- [31] Feldmann, K.; Slama, S.; Kunze, A.: Rationalisierungspotenziale und Montagekonzepte für mechatronische Systemlösungen, Maschinenbau in Bayern – Profile, Porträts, Perspektiven, München, 2003, S. 24-29.
- [32] Feldmann, K.; Zhuo, Y.; Alvarez, C.: 3D Gridless Routing for the Design of Molded Interconnect Devices, in: Production Engineering, Vol. XII/2 (2005).
- [33] Feldmann, K.; Zhuo, Y.; Alvarez, C.: Entwicklung eines integrierten Konstruktionsystems für mechatronische Produkte mit räumlichen Schaltungsstrukturen (3D-MID), in: 4. Paderborner Workshop - Entwurf mechatronischer Systeme, Vol. 189, HNI-Verlagsschriftenreihe, Paderborn, Deutschland, 2006, S. 371-383.
- [34] Frank, M.: Gehört STEP-AP 210 die Zukunft?, in: Productronic, Vol. 1/2 (2002).
- [35] Franke, J.: Integrierte Entwicklung neuer Produkt- und Produktionstechnologien für räumliche spritzgegossene Schaltungsträger (3D-MID), Dissertation, FAPS, Universität Erlangen-Nürnberg, 1995.
- [36] Gausemeier, J.; Feldmann, K.: Integrative Entwicklung räumlicher elektronischer Baugruppen, Carl Hanser Verlag, München, 2006.
- [37] Gausemeier, J.; Frank, U.: Stand und Perspektiven der Entwicklung mechatronischer Systeme, in: 4. Paderborner Workshop - Entwurf mechatronischer

- scher Systeme, Vol. 189, HNI-Verlagsschriftenreihe, Paderborn, Deutschland, 2006, S. 1-24.
- [38] Gausemeier, J.; Kespohl, H.D.; Möhringer, S.: Entwicklungsumgebungen Mechatronik. Integration von Methoden und Werkzeugen in den frühen Phasen der Entwicklung mechatronischer Systeme, in: Industrie Management, Vol. 17/3 (2001), S. 29-33.
- [39] Gausemeier, J.; Moehringer, S.: VDI 2206 - A new guideline for the design of mechatronic systems, in: Proceeding of the 2nd IFAC Conference on Mechatronic Systems, Vol. 2, 2002, S. 785-790.
- [40] Glendenning, P.; Debowski, M.: Factors influencing the quality of two-shot, selectively plated MID's, in: Proceedings of the 5th International Congress on Molded Interconnect Device, Research Association Molded Interconnect Devices 3-D MID e.V., Erlangen, 2002, S. 59-69.
- [41] Grabowski, H.; Anderl, R.; Polly, A.: Integriertes Produktmodell, Beuth Verlag Berlin, Wien, Zürich, 1993.
- [42] Haderer, G.: Integration von Gestaltung und Berechnung mittels CORBA, Dissertation, Fachbereich 11 -- Maschinenbau und Produktionstechnik, TU Berlin, 2000.
- [43] Hadlock, F.O.: A shortest path algorithm for grid graphs, in: Networks, Vol. 7 (4) (1977), S. 323-334.
- [44] Hama, T.; Etoh, H.: Topological Routing Path Search Algorithm with Incremental Routability Test, in: IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 18 (1999), S. 142-150.
- [45] Hare, R.M.; Julstrom, B.A.: A Spanning-Tree-Based Genetic Algorithm for Some Instances of the Rectilinear Steiner Problem with Obstacles, in: SAC, Melbourne, FL, USA, 2003, S. 725-729.
- [46] Hightower, D.W.: A solution to line-routing problem on the continuous plane, in: Proc. 6th Design Automation Workshop, 1969, S. 1-24.
- [47] Hingston, J.; Loughhead, P.; Irwin, R.: Topological Autorouting, Altium Limited, 2002.
- [48] Vojin Jovanović: Pro/TOOLKIT: Setting Up Your Environment in VC++ .NET, [http://www.caddigest.com/subjects/pro\\_engineer/select/tutorials/jovanovic\\_toolkit\\_environment.htm](http://www.caddigest.com/subjects/pro_engineer/select/tutorials/jovanovic_toolkit_environment.htm), Stand: 2003
- [49] Kallmann, M.: Path Planning in Triangulations, in: Proceedings of the Workshop on Reasoning, Representation, and Learning in Computer Games, International

- Joint Conference on Artificial Intelligence (IJCAI), Edinburgh, Scotland, 2005, S. 49-54.
- [50] Kallmann, M.; Bieri, H.; Thalmann, D.: Fully Dynamic Constrained Delaunay Triangulations, *Geometric Modelling for Scientific Visualization*, Springer-Verlag, Heidelberg, 2003, S. 241-257.
- [51] Klein, L.: STEP for Electronics, in: 6th NASA-ESA Workshop on Product Data Exchange – Open Standards for Model-Based Development, 2004.
- [52] Kramer, M.R.; Leeuwen, J.v.: Wire-routing is NP-complete, Technical Report, Computer Science Dept., University of Utrecht, 1982.
- [53] Krause, F.-L.; Ciesla, M.; Rieger, E.; Stephan, M.; Ulbrich, A.: Featureverarbeitung – Kernkomponente integrierter CAE-Systeme, in: *CAD' 94, Features als Träger semantischer Objekte integrierter Prozessketten*, Springer Verlag, 1994, S. 407-424.
- [54] Krause, F.-L.; Tang, T.; Ahle, U.: Abschlußbericht zum BMBF-Leitprojekt "Integrierte Virtuelle Produktentstehung (iViP)", Fraunhofer IRB-Verlag, Stuttgart, 2002.
- [55] Krebs, T.: Applications of the NEXTR - Microline 3D-Link for the Design and Manufacturing of MID products, in: *Proceedings of the 7th International Congress on Molded Interconnect Devices*, Research Association Molded Interconnect Devices 3-D MID e.V., Fürth, 2006, S. 237-243.
- [56] Krebs, T.: Integration elektromechanischer CA-Anwendungssysteme über einem STEP-Produktmodell, Dissertation, *FAPS*, Universität Erlangen-Nürnberg, 1996.
- [57] Krimi, S.: Analyse und Optimierung von Montagesystemen in der Elektronikproduktion, Dissertation, *FAPS*, Universität Erlangen-Nürnberg, 2001.
- [58] Kruskal, J.B.: On the shortest spanning subtree of a graph and the traveling salesman problem, in: *Proceedings of the American Mathematical Society*, Vol. 7 (1956), S. 48-50.
- [59] Lauther, U.: A Data Structure for Gridless Routing, in: *Proceedings of the 17th Design Automation Conference*, 1980, S. 603-609.
- [60] Lee, C.Y.: An algorithm for path connections and its application, in: *IRE Trans. on Electronic Computers*, Vol. EC-10 (1961), S. 346-365.
- [61] Lee, J.H.; Bose, N.K.; Hwang, F.K.: Use of Steiner's problem in suboptimal routing in rectilinear metric, in: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. cas-23 (1976), S. 470-476.

- [62] Lee, S.H.: A CAD-CAE integration approach using feature-based multi-resolution and multi-abstraction modelling techniques, in: Computer-Aided Design, Vol. 37 (2005), S. 941-955.
- [63] Li, S.G.: Pro/TOOLKIT Programmierung (Chinesisch), China Machine Press, Beijing, 2003.
- [64] Li, W.D.; Lu, W.F.; Fuh, J.Y.H.; Wong, Y.S.: Collaborative computer-aided design--Research and development status, in: Computer-Aided Design, Vol. 37 (2005), S. 931-940.
- [65] Lienig, J.: Layoutsynthese elektronischer Schaltungen - Grundlegende Algorithmen für die Entwurfsautomatisierung, Springer-Verlag Berlin Heidelberg, 2006.
- [66] Löffel, C.: Integration von Berechnungswerkzeugen in den rechnerunterstützten Konstruktionsprozess, Dissertation, Lehrstuhl für Konstruktionstechnik, Universität Erlangen-Nürnberg, 1997.
- [67] Lubell, J.; Peak, R.S.; Srinivasan, V.; Waterbury, S.C.: STEP, XML, AND UML: COMPLEMENTARY TECHNOLOGIES, in: Proceedings of DETC 2004, ASME 2004 Design Engineering technical Conferences and Computers and Information in Engineering Conference, Salt Lake City, Utah USA, 2004.
- [68] Maaßen, E.; Gleixner, J.: Connector-PCB for Anti-Blocking-Systems, in: Proceedings of the 3rd International Congress on Molded Interconnect Devices, Research Association Molded Interconnect Devices 3-D MID e.V., Erlangen, 1998, S. 51-60.
- [69] Malonnek, C.: Ein leitbahnorientiertes Verfahren für den physikalischen Entwurf integrierter digitaler Schaltungen, Dissertation, Informatik, Universität Hannover, 2004.
- [70] MECADTRON GmbH: Nextra Produktinformation, <http://www.mecadtron.de/produkte/nextra.php.de> Stand: 2006
- [71] Meerkamm, H.; Hochmuth, R.: Integrated Product Development based on the Design System mfk, in: International Conference DESIGN 98, Dubrovnik, Croatia, 1998.
- [72] Meier, R.: Strategien für einen produktorientierten Einsatz räumlicher spritzgossener Schaltungsträger (3-D MID), Dissertation, FAPS, Universität Erlangen-Nürnberg, 2002.
- [73] Moldflow Corporation: Moldflow Plastic Insight Release 4.1 Online Document: MPI Application Programming Interface, Stand: 2003

- [74] Moldflow Corporation: Moldflow Produktinformation, [http://www.moldflow.com/stp/deutsch/products/des\\_opt\\_sol.htm](http://www.moldflow.com/stp/deutsch/products/des_opt_sol.htm), Stand: 2006
- [75] Mössner-Beigel, M.; Rühle, R.; Ammon, D.; Rauh, J.: Datenintegration in Berechnung und Simulation mittels MechaSTEP, in: VDI-Berichte, Vol. 1755 (2003), S. 205-225.
- [76] N.N.: AP210 tutorial for experts in the electro-mechanical domain, <http://www.ap210.org/tiki/tiki-index.php?page=Tutorial>, Stand: 2006
- [77] N.N.: Automating Microsoft Office 97 and Office 2000, Microsoft Corporation, 2000.
- [78] N.N.: Concept of Operations for Application Protocol 210, PDES, Inc., 2001.
- [79] N.N.: Defining a CAD Library -- What does a CAD library need to meet manufacturing requirements?, <http://www.pcbstandards.com>, Stand: 2002
- [80] N.N.: ISO 10303-1, Industrial automation systems and integration -- Product data representation and exchange -- Part 1: Overview and fundamental principles, International Organization for Standardization, Geneva, Switzerland, 1994.
- [81] N.N.: ISO 10303-203, Industrial automation systems and integration -- Product data representation and exchange -- Part 203: Application protocol: Configuration controlled 3D designs of mechanical parts and assemblies, International Organization for Standardization, Geneva, Switzerland, 2005.
- [82] N.N.: ISO 10303-210, Industrial automation systems and integration -- Product data representation and exchange -- Part 210: Application protocol: Electronic assembly, interconnection, and packaging design, International Organization for Standardization, Geneva, Switzerland, 2001.
- [83] N.N.: ISO 10303-212, Industrial automation systems and integration -- Product data representation and exchange -- Part 212: Application protocol: Electro-technical design and installation, International Organization for Standardization, Geneva, Switzerland, 2001.
- [84] N.N.: ISO 10303-214, Industrial automation systems and integration -- Product data representation and exchange -- Part 214: Application protocol: Core data for automotive mechanical design processes International Organization for Standardization, Geneva, Switzerland, 2003.
- [85] N.N.: MECHASOFT -- Integrierte Entwicklung mechatronischer Systeme, Newsletter 10, *iwb*, 2002.
- [86] N.N.: Technologie 3D-MID -- Räumliche elektronische Baugruppen -- Herstellungsverfahren, Gebrauchsanforderungen, Materialkennwerte, Carl Hanser Verlag, München, 2004.

- [87] N.N.: VDI-Richtlinie 2206 Entwicklungsmethodik für mechatronische Systeme, VDI-Verlag, Düsseldorf, 2004.
- [88] N.N.: VDI-Richtlinie 2218 Informationsverarbeitung in der Produktentwicklung - Feature-Technologie, VDI-Verlag, Düsseldorf, 2003.
- [89] N.N.: VDI-Richtlinie 2221 Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte, VDI-Verlag, Düsseldorf, 1993.
- [90] Nell, J.: STEP on a page, National Institute of Standards and Technology (NIST) <http://www.mel.nist.gov/sc5/soap/>, 2003.
- [91] Ohtsuki, T.: Layout Design and Verification, Elsevier Science Publishers B.V., North-Holland, 1986.
- [92] Prim, R.C.: Shortest Connection Networks and some Generalizations, in: Bell System Technical Journal, Vol. 36 (1957), S. 1389-1401.
- [93] ProSTEP iViP Association: Was ist STEP?, <http://www.prostep.org/de/standards/was/>, Stand: 2006
- [94] Parametric Technology Corporation: Pro/ENGINEER Produktinformation, <http://www.ptc.com>, Stand: 2006
- [95] Parametric Technology Corporation: Pro/ENGINEER Wildfire 2.0 -- Pro/TOOLKIT User's Guide, Stand: 2005
- [96] Reinhart, G.; Anton, O.; Lercher, B.: Funktionsorientiertes Sichtenmodell für die Entwicklung mechatronischer Systeme, in: VDI-Z Integrierte Produktion, (2001).
- [97] Russell, S.J.; Norvig, P.: Artificial Intelligence -- A Modern Approach, Second ed., Pearson Education, 2003.
- [98] Schlögl, W.: Integriertes Simulationsdaten-Management für Maschinenentwicklung und Anlagenplanung, Dissertation, FAPS, Universität Erlangen-Nürnberg, 2000.
- [99] Schubert, M.: 3D-Design elektro-mechanischer Baugruppen als Heißspräge-MID, Ergebnisse und Leistungen 2001, HSG-IMAT, Universität Stuttgart, 2001, S. 12-13.
- [100] Shepherd, G.; Kruglinski, D.: Programming with Microsoft Visual C++ .NET, Sixth Edition, Microsoft Press, 2003.
- [101] Smith, G.L.: Utilization of STEP AP210 at the Boeing Company, in: Computer Aided Design, Vol. 34 (2002), S. 1055-1062.
- [102] Soukup, J.: Fast maze router, in: Proceedings 15th Design Automation Conference, 1978, S. 100-102.

- [103] Tamburini, D.R.; Peak, R.S.; Fulton, R.E.: Driving PWA Thermomechanical Analysis from STEP AP210 Product Models, in: ASME Intl. Mech. Engr. Congress and exposition, Dallas, TX, 1997.
- [104] Thurman, T.: AP 210 Foundations for Electrical-Mechanical Integration, in: NASA STEP for Aerospace Workshop at Jet Propulsion Laboratory, 2000.
- [105] Thurman, T.: A Detailed Review of Some Aspects of AP 210 Useful for Electro-Technical Designs and Libraries, in: NASA STEP for Aerospace Workshop at Jet Propulsion Laboratory, 2000.
- [106] Thurman, T.: PDES Inc. Electro-mechanical Pilot, in: NASA STEP for Aerospace Workshop at Jet Propulsion Laboratory, 2000.
- [107] Thurman, T.; Smith, G.: Overview & Tutorial of STEP AP 210, Standard for Electronic Assembly Interconnect and Packaging Design, PDES, Inc., 1999.
- [108] Vajna, S.; Podehl, G.: Durchgängige Produktmodellierung mit Features, in: CAD-CAM REPORT, (Nr. 3) (1998).
- [109] Wartzack, S.; Meerkamm, H.: Durchgängige Rechnerunterstützung in der Produktentwicklung durch den Einsatz von semantisch hochwertigen Features. In: Beschleunigung der Produktentwicklung durch EDM/PDM- und Feature-Technologie, in: VDI-Berichte, Vol. 1497 (VDI-Verlag) (1999), S. 369-391.
- [110] Welp, E.G.; Braun, P.: Semantische und systemtechnische Kopplung kommerzieller CAD und Berechnungssysteme durch einen objektorientierten Integrationsprozessor., in: VDI-Berichte, Vol. 1357 (1997), S. 125-141.
- [111] Wölflé, F.: Virtuelle Berechnungskompetenzzentren als Dienstleister zur Integration von Gestaltung und Berechnung., in: VDI Fortschr. -Ber., (VDI-Verlag (Reihe 20 Nr. 274)) (1998).
- [112] Wu, Y.: Describing Integrated Power Electronics Modules using STEP AP210, Master, Center for Power Electronics Systems, Virginia Polytechnic Institute and State University, 2003.
- [113] Wu, Y.F.; Widmayer, P.; Schlag, M.D.F.; Wong, C.K.: Rectilinear Shortest Paths and Minimum Spanning Trees in the Presence of Rectilinear Obstacles, in: IEEE Transactions on Computers, Vol. C-36 (1987), S. 321-331.
- [114] Xing, Z.Y.; Kao, R.: A Minimum Cost Path Search Algorithm Through Tile Obstacles, in: Proc. Int. Symp. Physical Design, Sonoma, California, USA, 2001, S. 192-197.
- [115] Xu, X.Y.; Wang, Y.Y.: Multi-model technology and its application in the integration of CAD/CAM/CAE, in: Journal of Materials Processing Technology, Vol. 129 (2002), S. 563-567.

- [116] Zäh, M.F.; Ehrenstraßer, M.; Guserle, R.; Lercher, B.; Pörnbacher, C.; Wunsch, G.: Integration von Simulationsmodellen im Werkzeugmaschinenbau, in: VDI-Z, Düsseldorf, (2003).
- [117] Zäh, M.F.; Lercher, B.; Pörnbacher, C.; Wunsch, G.: Anforderungen an das Datenmanagement in der Mechatronik, in: Industriemanager, (Berlin) (2003).
- [118] Zheng, S.Q.; Lim, J.S.; Iyengar, S.: Finding obstacle-avoiding shortest paths using implicit connection graphs, in: IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 15 (1996), S. 103-110.
- [119] Zhou, H.: Efficient Steiner Tree Construction Based on Spanning Graphs, in: Proc. Int. Symp. Physical Design, Monterey, California, USA, 2003, S. 152-157.
- [120] Zhuo, Y.: Rechnergestützter Entwurf von räumlichen elektronischen Baugruppen (3D-MID), Bayerisches Kompetenznetzwerk für Mechatronik Abschlussbericht 2005, FAPS, 2005.
- [121] Zhuo, Y.; Alvarez, C.; Feldmann, K.: AN INTEGRATED DESIGN SYSTEM FOR MOLDED INTERCONNECT DEVICES (3D-MID), in: Proceedings of the 3rd CIRP Sponsored Conference on Digital Enterprise Technology, Setubal, Portugal, 2006.
- [122] Zhuo, Y.; Alvarez, C.; Feldmann, K.: MIDCAD – Ein rechnergestütztes System für den Entwurf räumlicher spritzgegossener Schaltungsträger, in: VDI-Berichte - Mechatronik 2005, Vol. 1892.2, VDI/VDE-Gesellschaft, Wiesloch, Deutschland, 2005, S. 1117-1135.
- [123] Zuken GmbH: EM-Designer Product Information, [http://www.zuken.com/electromechanical/em\\_designer.asp](http://www.zuken.com/electromechanical/em_designer.asp), Stand: 2006
- [124] Zwicker, E.: Unterstützung der unternehmensübergreifenden Produktentwicklung durch den Einsatz moderner Informationstechnologien, Dissertation, Institut für Konstruktion und Bauweisen, Eidgenössische Technische Hochschule (ETH) Zürich, 1998.

# Anhang

## Übersicht über die verschiedenen Bestandteile von STEP

ISO TC184 SC4	STEP on a Page	ISO 10303
<b>APPLICATION PROTOCOLS AND ASSOCIATED ABSTRACT TEST SUITES</b>		
<ul style="list-style-type: none"> <li>201 Explicit draughting [ATS 301 = X]</li> <li>202 Associative draughting [X]</li> <li>203 Configuration-controlled design (c2=I, a1=I) [X]</li> <li>204 Mechanical design using boundary rep [I]</li> <li>X 205 Mechanical design using surface rep [W]</li> <li>X 206 Mechanical design using wireframe [X]</li> <li>207 Sheet metal die planning and design [I]</li> <li>X 208 Life-cycle product change process [X]</li> <li>209 Composite &amp; metal structural anal &amp; related design [X]</li> <li>210 Electronic assy, interconnection &amp; packaging design [X]</li> <li>X 211 Electronic P-C assy test, diag, &amp; remanuf [X]</li> <li>X 212 Electrotechnical design and installation [C]</li> <li>X 213 Num control (NC) process plans for mach'd parts [X]</li> <li>214 Core data for automatic mech design processes (e2=E) [F]</li> <li>HE 215 Ship arrangement [X]</li> <li>216 Ship moulded forms [X]</li> <li>X 217 Ship piping [X]</li> <li>218 Ship structures [X]</li> <li>X 219 Dimension inspection [X]</li> <li>O 220 Proc. plg, mfg, assy of layered electrical products [X]</li> </ul>	<ul style="list-style-type: none"> <li>C 221 Functional data &amp; their schem rep for process plant [X]</li> <li>X 222 Design-manuf for composite structures [W]</li> <li>X 223 Exch of design &amp; mfg product info for cast parts [O]</li> <li>I 224 Mech pdt del for p. plg using mach'g feat (e2=X, e3=A)</li> <li>I 225 Building elements using explicit shape rep [C] [X, I]</li> <li>X 226 Ship mechanical systems [C]</li> <li>I 227 Plant spatial configuration (e2=C) [X]</li> <li>X 228 Building services: HVAC [X]</li> <li>X 229 Design &amp; mfg product info for forged parts [X]</li> <li>X 230 Building structural frame: steelwork [X]</li> <li>X 231 Process-engineering data [X]</li> <li>I 232 Technical data packaging: core info &amp; exch [I]</li> <li>W 233 Systems engineering data repr (to be PAS 20342) [X]</li> <li>X 234 Ship operational logs, records, and messages [X]</li> <li>W 235 Materials info for des and verif of products [X]</li> <li>W 236 Furniture product and project data [W]</li> <li>W 237 Computational Fluid Dynamics</li> <li>A 238 Computer numerical controllers</li> <li>X 239 Product life-cycle support</li> <li>W 240 Process plans for machined products</li> </ul>	
<b>COMMON RESOURCES (with 13584-20 logx. model of expr.(I) and 15531-42 Time (W))</b>		
<b>APPLICATION MODULES (Technical specifications)</b>		<b>Legend: TS Status</b>
For status of the modules see the file via the SOAP home page.		0-10 =O=Prop. → app for ballot 10-20=A=NP bit circ. → NFP app 20-60=D=DTS dev. → reg as TS >60 =T=TS Published
<b>INTEGRATED-APPLICATION RESOURCES</b>		
<ul style="list-style-type: none"> <li>I 101 Draughting (c1=I)</li> <li>X 102 Ship structures</li> <li>X 103 E/E connectivity</li> <li>I 104 Finite element analysis</li> <li>I 105 Kinematics (c1=I, c2=I)</li> </ul>	<ul style="list-style-type: none"> <li>X 106 Building core model</li> <li>C 107 Finite-element analysis definition relationships</li> <li>C 108 Prmetizat'n &amp; Constraints for expl geom prod mds</li> <li>C 109 Assembly mode for products</li> <li>W 110 Mesh-based computational fluid dynamics</li> </ul>	
<b>INTEGRATED-GENERIC RESOURCES</b>		
<ul style="list-style-type: none"> <li>41 Fund of pdct desc &amp; spt (e2=I, c1=I)</li> <li>42 Geom &amp; top rep (c3=I, e2c1=I, e3=F)</li> <li>43 Repres specialization (e2=I, c1=I, c2=I)</li> <li>44 Product struct config (e2=I, c1=I)</li> <li>45 Materials (c1=I)</li> <li>46 Visual presentation (c1=I, c2=I)</li> <li>47 Tolerances (c1=I)</li> <li>X 48 Form features</li> <li>49 Process structure &amp; properties</li> </ul>	<ul style="list-style-type: none"> <li>I 50 Mathematical construct</li> <li>E 51 Mathematical description</li> <li>W 52 Mesh-based topology</li> <li>W 53 Numerical Analysis</li> <li>C 54 Classification Set theory</li> <li>A 55 Procedural and hybrid represent.</li> <li>W 56 State</li> <li>W 57 Expression extensions</li> <li>A 58 Risk</li> </ul>	
<b>APPLICATION-INTERPRETED CONSTRUCTS</b>		
<ul style="list-style-type: none"> <li>I 501 Edge-based wireframe</li> <li>I 502 Shell-based wireframe</li> <li>I 503 Geom-bounded 2D wireframe</li> <li>I 504 Draughting annotation</li> <li>I 505 Drawing structure &amp; admin.</li> <li>I 506 Draughting elements</li> <li>I 507 Geom-bounded surface</li> <li>I 508 Non-manifold surface</li> <li>I 509 Manifold surface</li> <li>I 510 Geom-bounded wireframe</li> <li>I 511 Topological-bounded surface</li> </ul>	<ul style="list-style-type: none"> <li>I 512 Faceted B-representation</li> <li>I 513 Elementary B-rep</li> <li>I 514 Advanced B-rep</li> <li>I 515 Constructive solid geometry</li> <li>X 516 Mechanical-design context</li> <li>I 517 Mech-design geom representation (c1=I)</li> <li>I 518 Mech-design shaded presentation</li> <li>I 519 Geometric tolerances (c1=I)</li> <li>I 520 Assoc draughting elements</li> <li>O 521 Manifold subspaces</li> <li>E 522 Machining features</li> <li>A 523 Curve swept solid</li> </ul>	
<b>IMPLEMENTATION METHODS</b>		
<ul style="list-style-type: none"> <li>I 21 Clear-text encoding exch str (c1=I, e2=I)</li> <li>I 22 Standard data access interface</li> <li>I 23 C++ language binding (to #22)</li> <li>I 24 C language binding (to #22)</li> </ul>	<ul style="list-style-type: none"> <li>C 25 EXPRESS to OMO XML</li> <li>X 26 IDL language binding (to #22)</li> <li>I 27 JAV &amp; language binding (to #22)</li> <li>@28 XML rep for EXPRESS-sc: hemata &amp; data</li> <li>X 29 Lwt Java binding (to #22) (DTS)</li> </ul>	
<p><b>Legend: Part Status (E, F, I safe to implement)</b>          (E)=F= Preliminary Stage (Proposal → app for NP ballot)          10=A= Proposal Stage (NP ballot circ. → NFP approval)          20=W= Preparatory Stage (Wkg Draft devel. → CD regis)          30=C= Committee Stage (CD circulation → DIS regis)</p>		
<p><b>Legend: TS Status</b>          0-10 =O=Prop. → app for ballot          10-20=A=NP bit circ. → NFP app          20-60=D=DTS dev. → reg as TS          &gt;60 =T=TS Published</p>		
<b>CONFERENCE TESTING METHODOLOGY &amp; FRAMEWORK</b>		
<ul style="list-style-type: none"> <li>1-31 General overview</li> <li>X 32 Requirements on testing plans and plans</li> <li>X 33 Structure and use of abstract test suites</li> <li>I 34 Abstract test methods for Part 21 implementation</li> <li>C 35 Abstract test methods for Part 22 implementation</li> </ul>		

Bild 92: Übersicht über die verschiedenen Bestandteile von STEP [90]

## Beispielschema in EXPRESS, EXPRESS-G und Part 21 Format

Ein Beispiel eines einfachen EXPRESS Informationsmodells, angelehnt an ISO10303-11 und 21, ist in folgenden Bildern zu sehen.

### Express

```

SCHEMA simple_drawings;

ENTITY drawing;
  name : STRING;
  elements : SET [1:?] OF shape;
END_ENTITY;

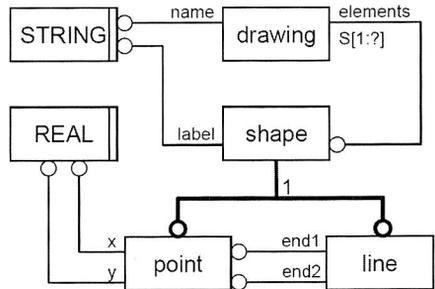
ENTITY shape;
  label : STRING;
END_ENTITY;

ENTITY point SUBTYPE OF (shape);
  x : REAL;
  y : REAL;
END_ENTITY;

ENTITY line SUBTYPE OF (shape);
  end1 : point;
  end2 : point;
END_ENTITY;

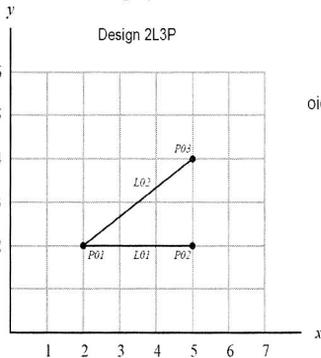
END_SCHEMA;
    
```

### Express-G



5

Beispielhaftes Modell  
Klassisches graphisches Format



Beispielhaftes Modell  
STEP Part 21 Format

```

ISO-10303-21;
...
FILE_SCHEMA (('simple_drawings'));
...
oid = #10 = point ('P01', 2.0, 2.0);
      #20 = point ('P02', 5.0, 2.0);
      #30 = point ('P03', 5.0, 4.0);
      #110 = line ('L01', #10, #20);
      #150 = line ('L02', #10, #30);
      #200 = drawing ('Design 2L3P',
                    (#10, #20, #30, #110, #150));
...
END-ISO-10303-21;
    
```

Verknüpfte Schema  
Erbte Attribute  
Beziehung zu anderen Instanzen  
Mitglied der gesamten Attributen

oid = Objektidentität

## Verbindung zwischen Pro/TOOLKIT und Microsoft MFC

Die in der vorliegenden Arbeit eingesetzte Konfigurationsmethode, mit der im Pro/TOOLKIT-Projekt die MFC-Bibliothek direkt benutzt werden kann, ist besser als die Methode, die im offiziellen Dokument „Pro/TOOLKIT User's Guide“ von PTC gegeben ist und unter der Entwicklungsumgebung VC++ von Visual Studio 2005 realisiert wurde. Im Folgenden werden einige wichtige Schritte der Methode vorgestellt. Die anderen gleichen Einstellungen können in der Literatur [48], [63] im Detail nachgeschlagen werden.

Zuerst wird in VC++ ein neues Projekt als *MFC DLL* erzeugt (d. h. das zu erzeugende Projekt ist eine DLL, die die MFC-Bibliothek benutzen wird), aber nicht als normal Windows DLL Applikation (siehe Bild 93).

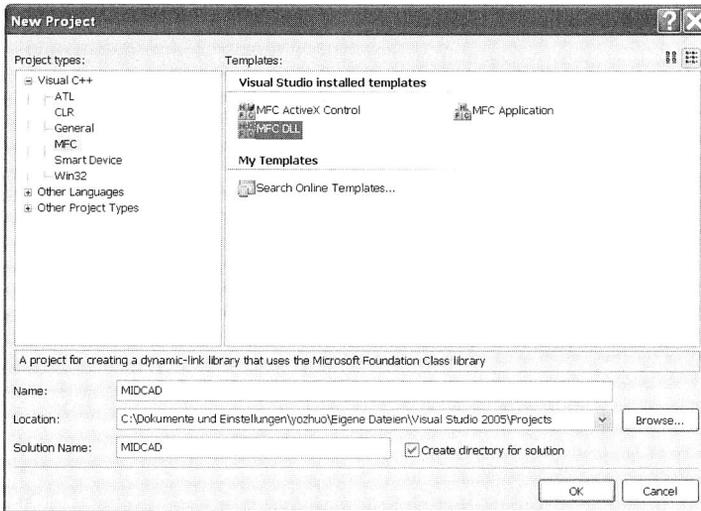


Bild 93: Erzeugung des Pro/TOOLKIT-Projektes als MFC DLL in VC++

Anschließend wird die Konfiguration des Projektes durchgeführt. Der Schritt ist am wichtigsten, um eine richtige Verbindung mit Pro/TOOLKIT einzustellen. Die gesamte Konfiguration ist im Vergleich zur anderen Methoden relativ einfach. Sie steht nur im Zusammenhang mit den Eigenschaften „C/C++“ und „Linker“. Der Unterschied ist, dass die MFC DLL-Applikation als *Multi-threaded Debug DLL (/MDd)* behandelt wird und deshalb die Bibliotheken *protk\_dllmd.lib* und *prodev\_dllmd.lib* von Pro/TOOLKIT benutzt werden.

Die Einstellung der Eigenschaft „C/C++“ des Projektes ist in Bild 94 dargestellt. Die Einstellung der Eigenschaft „Linker“ ist in Bild 95 gezeigt.

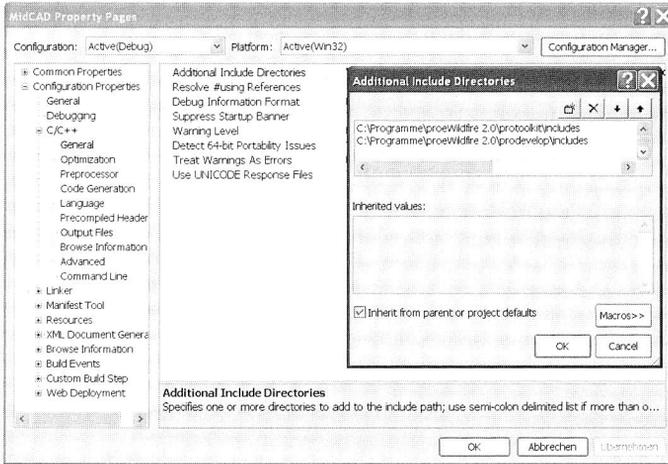


Bild 94: Einstellung der Eigenschaft „C/C++“

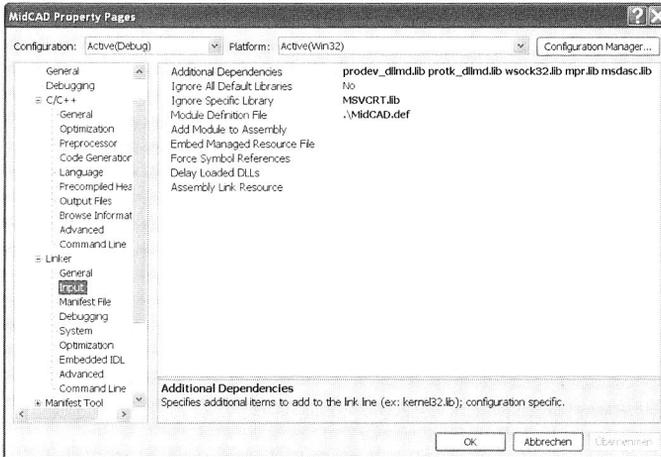


Bild 95: Einstellung der Eigenschaft „Linker“

Schließlich werden die zu realisierenden Funktionen in verschiedenen C++ Dateien in Rahmen des Projektes programmiert. Beispielsweise werden die in Bild 96 dargestellten Quelltexte, zur Erzeugung des eigenen Menüs in Pro/ENGINEER als Benutzeroberfläche vom MIDCAD-System, in die Hauptdatei eingefügt. Da die originalen Funktionen `user_terminate()` und `user_initialize()` zu Funktionen der Programmiersprache C gehören, muss die zusätzliche Erklärung „extern „C““ vor den Titeln der beiden Funktionen geschrieben werden.

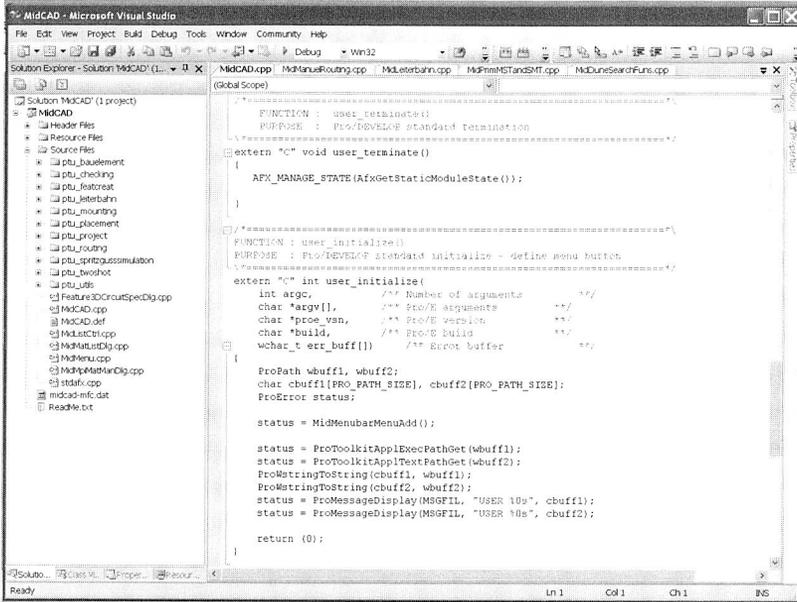


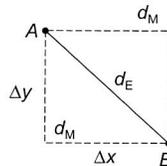
Bild 96: Die wichtigen Quelltexte zur Initialisierung in Hauptdatei

## Manhattan- und euklidische Metrik

Die Abstandsfunktion in der Ebene für zwei Punkte  $(x_1, y_1)$  und  $(x_2, y_2)$  ist definiert mit

$$d = \sqrt[n]{|x_2 - x_1|^n + |y_2 - y_1|^n}, \text{ wobei}$$

$$n \begin{cases} = 1 & \text{Manhattan-Metrik} \\ & \text{(Rectilinear/Manhattan metric)} & d_M = |\Delta x| + |\Delta y|, \\ = 2 & \text{Euklidische Metrik} \\ & \text{(Euclidian/Boston metric).} & d_E = \sqrt{\Delta x^2 + \Delta y^2}. \end{cases}$$



Die Manhattan- und die euklidische Metrik zeichnen sich durch einige spezifische Eigenschaften aus. Die konkrete Darstellung der Eigenschaften kann weiter in relevanter Literatur wie z. B. [65] nachgeschlagen werden.

Bei dem Manhattan-Routing sind nur waagrechte und senkrechte Verbindungswege möglich, d. h. die einzelnen Netzsegmente sind orthogonal zueinander angeordnet, während bei dem euklidischen Routing (all angle routing) diese Beschränkung nicht besteht.



## Lebenslauf

Yong ZHUO

geboren am 14. November 1970 in Putian, VR China

verheiratet

- 09/76 – 07/81      Grundschule in Jianou, Provinz Fujian, VR China
- 09/81 – 07/87      Mittelschule in Jianou, Provinz Fujian, VR China
- 09/87 – 07/91      Studium an der Zhejiang Universität, Hangzhou, VR China  
Hauptfach: Mechanik, Nebenfach: Maschinenbau  
Abschluss: Bachelor of Engineering
- 07/91 – 09/95      Assistent an der forstwissenschaftlichen Universität Fujian,  
Nanping, VR China
- 09/95 – 03/98      Masterstudium an der Zhejiang Universität, Hangzhou,  
Studienrichtung: Engineering Drawing Science and CAD  
Abschluss: M.Sc.
- 03/98 – 03/02      Dozent am Institut für Maschinenbau an der Xiamen Universi-  
tät, Xiamen, VR China
- 03/02 – 02/04      Stipendiat der Hanns-Seidel-Stiftung, Gastwissenschaftler am  
Lehrstuhl für Fertigungsautomatisierung und Produktionssys-  
tematik der Universität Erlangen-Nürnberg  
Leiter: Prof. Dr.-Ing. K. Feldmann
- 03/04 – 02/07      Stipendiat der Bayerischen Forschungsstiftung, Gastwissen-  
schaftler am Lehrstuhl für Fertigungsautomatisierung und Pro-  
duktionssystematik der Universität Erlangen-Nürnberg  
Leiter: Prof. Dr.-Ing. K. Feldmann



# Reihe Fertigungstechnik - Erlangen

[www.mb.uni-erlangen.de/diss](http://www.mb.uni-erlangen.de/diss)

**Band 1 - 52**  
Carl Hanser Verlag, München

**ab Band 53**  
Meisenbach Verlag, Bamberg  
45,- Euro

Band 1: Andreas Hemberger  
**Innovationspotentiale in der rechnerintegrierten Produktion durch wissensbasierte Systeme**  
208 Seiten, 107 Bilder. 1988.

Band 2: Detlef Classe  
**Beitrag zur Steigerung der Flexibilität automatisierter Montagesysteme durch Sensorintegration und erweiterte Steuerungskonzepte**  
194 Seiten, 70 Bilder. 1988.

Band 3: Friedrich-Wilhelm Nolting  
**Projektiertung von Montagesystemen**  
201 Seiten, 107 Bilder, 1 Tabelle. 1989.

Band 4: Karsten Schlüter  
**Nutzungsgradsteigerung von Montagesystemen durch den Einsatz der Simulationstechnik**  
177 Seiten, 97 Bilder. 1989.

Band 5: Shir-Kuan Lin  
**Aufbau von Modellen zur Lageregelung von Industrierobotern**  
168 Seiten, 46 Bilder. 1989.

Band 6: Rudolf Nuss  
**Untersuchungen zur Bearbeitungsqualität im Fertigungssystem Laserstrahlschneiden**  
206 Seiten, 115 Bilder, 6 Tabellen. 1989.

Band 7: Wolfgang Scholz  
**Modell zur datenbankgestützten Planung automatisierter Montageanlagen**  
194 Seiten, 89 Bilder. 1989.

Band 8: Hans-Jürgen Wißmeier  
**Beitrag zur Beurteilung des Bruchverhaltens von Hartmetall-Fließpreßmatrizen**  
179 Seiten, 99 Bilder, 9 Tabellen. 1989.

Band 9: Rainer Eisele  
**Konzeption und Wirtschaftlichkeit von Planungssystemen in der Produktion**  
183 Seiten, 86 Bilder. 1990.

Band 10: Rolf Pfeiffer  
**Technologisch orientierte Montageplanung am Beispiel der Schraubtechnik**  
216 Seiten, 102 Bilder, 16 Tabellen. 1990.

Band 11: Herbert Fischer  
**Verteilte Planungssysteme zur Flexibilitätssteigerung der rechnerintegrierten Teilefertigung**  
201 Seiten, 82 Bilder. 1990.

- Band 12: Gerhard Kleineidam  
**CAD/CAP: Rechnergestützte Montagefeinplanung**  
203 Seiten, 107 Bilder. 1990.
- Band 13: Frank Vollertsen  
**Pulvermetallurgische Verarbeitung  
eines übereutektoiden verschleißfesten Stahls**  
XIII u. 217 Seiten, 67 Bilder, 34 Tabellen. 1990.
- Band 14: Stephan Biermann  
**Untersuchungen zur Anlagen- und Prozeßdiagnostik  
für das Schneiden mit CO<sub>2</sub>-Hochleistungslasern**  
VIII u. 170 Seiten, 93 Bilder, 4 Tabellen. 1991.
- Band 15: Uwe Geißler  
**Material- und Datenfluß  
in einer flexiblen Blechbearbeitungszelle**  
124 Seiten, 41 Bilder, 7 Tabellen. 1991.
- Band 16: Frank Oswald Hake  
**Entwicklung eines rechnergestützten Diagnosesystems  
für automatisierte Montagezellen**  
XIV u. 166 Seiten, 77 Bilder. 1991.
- Band 17: Herbert Reichel  
**Optimierung der Werkzeugbereitstellung  
durch rechnergestützte Arbeitsfolgenbestimmung**  
198 Seiten, 73 Bilder, 2 Tabellen. 1991.
- Band 18: Josef Scheller  
**Modellierung und Einsatz von Softwaresystemen  
für rechnergeführte Montagezellen**  
198 Seiten, 65 Bilder. 1991.
- Band 19: Arnold vom Ende  
**Untersuchungen zum Biegeumformen mit elastischer Matrize**  
166 Seiten, 55 Bilder, 13 Tabellen. 1991.
- Band 20: Joachim Schmid  
**Beitrag zum automatisierten Bearbeiten  
von Keramikguß mit Industrierobotern**  
XIV u. 176 Seiten, 111 Bilder, 6 Tabellen. 1991.
- Band 21: Egon Sommer  
**Multiprozessorsteuerung für kooperierende  
Industrieroboter in Montagezellen**  
188 Seiten, 102 Bilder. 1991.
- Band 22: Georg Geyer  
**Entwicklung problemspezifischer Verfahrensketten  
in der Montage**  
192 Seiten, 112 Bilder. 1991.
- Band 23: Rainer Flohr  
**Beitrag zur optimalen Verbindungstechnik  
in der Oberflächenmontage (SMT)**  
186 Seiten, 79 Bilder. 1991.
- Band 24: Alfons Rief  
**Untersuchungen zur Verfahrensfolge Laserstrahlschneiden  
und -schweißen in der Rohkarosseriefertigung**  
VI u. 145 Seiten, 58 Bilder, 5 Tabellen. 1991.
- Band 25: Christoph Thim  
**Rechnerunterstützte Optimierung von Materialflußstrukturen  
in der Elektronikmontage durch Simulation**  
188 Seiten, 74 Bilder. 1992.
- Band 26: Roland Müller  
**CO<sub>2</sub>-Laserstrahlschneiden  
von kurzglasverstärkten Verbundwerkstoffen**  
141 Seiten, 107 Bilder, 4 Tabellen. 1992.
- Band 27: Günther Schäfer  
**Integrierte Informationsverarbeitung bei der Montageplanung**  
195 Seiten, 76 Bilder. 1992.

- Band 28: Martin Hoffmann  
**Entwicklung einer CAD/CAM-Prozeßkette für die Herstellung von Blechbiegeteilen**  
149 Seiten, 89 Bilder. 1992.
- Band 29: Peter Hoffmann  
**Verfahrensfolge Laserstrahlschneiden und –schweißen : Prozeßführung und Systemtechnik in der 3D–Laserstrahlbearbeitung von Blechformteilen**  
186 Seiten, 92 Bilder, 10 Tabellen. 1992.
- Band 30: Olaf Schrödel  
**Flexible Werkstattsteuerung mit objektorientierten Softwarestrukturen**  
180 Seiten, 84 Bilder. 1992.
- Band 31: Hubert Reinisch  
**Planungs- und Steuerungswerkzeuge zur impliziten Geräteprogrammierung in Roboterzellen**  
XI u. 212 Seiten, 112 Bilder. 1992.
- Band 32: Brigitte Bärnreuther  
**Ein Beitrag zur Bewertung des Kommunikationsverhaltens von Automatisierungsgeräten in flexiblen Produktionszellen**  
XI u. 179 Seiten, 71 Bilder. 1992.
- Band 33: Joachim Hutfless  
**Laserstrahlregelung und Optikiagnostik in der Strahlführung einer CO<sub>2</sub>-Hochleistungslaseranlage**  
175 Seiten, 70 Bilder, 17 Tabellen. 1993.
- Band 34: Uwe Günzel  
**Entwicklung und Einsatz eines Simulationsverfahrens für operative und strategische Probleme der Produktionsplanung und –steuerung**  
XIV u. 170 Seiten, 66 Bilder, 5 Tabellen. 1993.
- Band 35: Bertram Ehmann  
**Operatives Fertigungscontrolling durch Optimierung auftragsbezogener Bearbeitungsabläufe in der Elektronikfertigung**  
XV u. 167 Seiten, 114 Bilder. 1993.
- Band 36: Harald Kolléra  
**Entwicklung eines benutzerorientierten Werkstattprogrammiersystems für das Laserstrahlschneiden**  
129 Seiten, 66 Bilder, 1 Tabelle. 1993.
- Band 37: Stephanie Abels  
**Modellierung und Optimierung von Montageanlagen in einem integrierten Simulationssystem**  
188 Seiten, 88 Bilder. 1993.
- Band 38: Robert Schmidt–Hebbel  
**Laserstrahlbohren durchflußbestimmender Durchgangslöcher**  
145 Seiten, 63 Bilder, 11 Tabellen. 1993.
- Band 39: Norbert Lutz  
**Oberflächenfeinbearbeitung keramischer Werkstoffe mit XeCl–Excimerlaserstrahlung**  
187 Seiten, 98 Bilder, 29 Tabellen. 1994.
- Band 40: Konrad Grampp  
**Rechnerunterstützung bei Test und Schulung an Steuerungssoftware von SMD–Bestücklinien**  
178 Seiten, 88 Bilder. 1995.
- Band 41: Martin Koch  
**Wissensbasierte Unterstützung der Angebotsbearbeitung in der Investitionsgüterindustrie**  
169 Seiten, 68 Bilder. 1995.
- Band 42: Armin Gropp  
**Anlagen- und Prozeßdiagnostik beim Schneiden mit einem gepulsten Nd:YAG–Laser**  
160 Seiten, 88 Bilder, 7 Tabellen. 1995.

Band 43: Werner Heckel  
**Optische 3D-Konturerfassung und on-line Biegewinkelmessung  
mit dem Lichtschnittverfahren**  
149 Seiten, 43 Bilder, 11 Tabellen. 1995.

Band 44: Armin Rothhaupt  
**Modulares Planungssystem  
zur Optimierung der Elektronikfertigung**  
180 Seiten, 101 Bilder. 1995.

Band 45: Bernd Zöllner  
**Adaptive Diagnose in der Elektronikproduktion**  
195 Seiten, 74 Bilder, 3 Tabellen. 1995.

Band 46: Bodo Vormann  
**Beitrag zur automatisierten Handhabungsplanung  
komplexer Blechbiegeteile**  
126 Seiten, 89 Bilder, 3 Tabellen. 1995.

Band 47: Peter Schnepf  
**Zielkostenorientierte Montageplanung**  
144 Seiten, 75 Bilder. 1995.

Band 48: Rainer Klotzbücher  
**Konzept zur rechnerintegrierten Materialversorgung  
in flexiblen Fertigungssystemen**  
156 Seiten, 62 Bilder. 1995.

Band 49: Wolfgang Greska  
**Wissensbasierte Analyse und Klassifizierung von Blechteilen**  
144 Seiten, 96 Bilder. 1995.

Band 50: Jörg Franke  
**Integrierte Entwicklung neuer Produkt- und Produktionstechnologien  
für räumliche spritzgegossene Schaltungsträger (3-D MID)**  
196 Seiten, 86 Bilder, 4 Tabellen. 1995.

Band 51: Franz-Josef Zeller  
**Sensorplanung und schnelle Sensorregelung für Industrieroboter**  
190 Seiten, 102 Bilder, 9 Tabellen. 1995.

Band 52: Michael Solvie  
**Zeitbehandlung und Multimedia-Unterstützung  
in Feldkommunikationssystemen**  
200 Seiten, 87 Bilder, 35 Tabellen. 1996.

Band 53: Robert Hopperditzel  
**Reengineering in der Elektro- und Elektronikindustrie**  
180 Seiten, 109 Bilder, 1 Tabelle. 1996.  
ISBN 3-87525-070-2

Band 54: Thomas Rebhan  
**Beitrag zur Mikromaterialbearbeitung mit Excimerlasern –  
Systemkomponenten und Verfahrensoptimierungen**  
148 Seiten, 61 Bilder, 10 Tabellen. 1996.  
ISBN 3-87525-075-3

Band 55: Henning Hanebuth  
**Laserstrahlhartlöten mit Zweistrahltechnik**  
157 Seiten, 58 Bilder, 11 Tabellen. 1996.  
ISBN 3-87525-074-5

Band 56: Uwe Schönherr  
**Steuerung und Sensordatenintegration für flexible Fertigungszellen  
mit kooperierenden Robotern**  
188 Seiten, 116 Bilder, 3 Tabellen. 1996.  
ISBN 3-87525-076-1

Band 57: Stefan Holzer  
**Berührungslose Formgebung mit Laserstrahlung**  
162 Seiten, 69 Bilder, 11 Tabellen. 1996.  
ISBN 3-87525-079-6

- Band 58: Markus Schultz  
**Fertigungsqualität beim 3D-Laserstrahlschweißen von Blechformteilen**  
165 Seiten, 88 Bilder, 9 Tabellen. 1997.  
ISBN 3-87525-080-X
- Band 59: Thomas Krebs  
**Integration elektromechanischer CA-Anwendungen über einem STEP-Produktmodell**  
198 Seiten, 58 Bilder, 8 Tabellen. 1997.  
ISBN 3-87525-081-8
- Band 60: Jürgen Sturm  
**Prozeßintegrierte Qualitätssicherung in der Elektronikproduktion**  
167 Seiten, 112 Bilder, 5 Tabellen. 1997.  
ISBN 3-87525-082-6
- Band 61: Andreas Brand  
**Prozesse und Systeme zur Bestückung räumlicher elektronischer Baugruppen (3D-MID)**  
182 Seiten, 100 Bilder. 1997.  
ISBN 3-87525-087-7
- Band 62: Michael Kauf  
**Regelung der Laserstrahlleistung und der Fokusparameter einer CO<sub>2</sub>-Hochleistungslaseranlage**  
140 Seiten, 70 Bilder, 5 Tabellen. 1997.  
ISBN 3-87525-083-4
- Band 63: Peter Steinwasser  
**Modulares Informationsmanagement in der integrierten Produkt- und Prozeßplanung**  
190 Seiten, 87 Bilder. 1997.  
ISBN 3-87525-084-2
- Band 64: Georg Liedl  
**Integriertes Automatisierungskonzept für den flexiblen Materialfluß in der Elektronikproduktion**  
196 Seiten, 96 Bilder, 3 Tabellen. 1997.  
ISBN 3-87525-086-9
- Band 65: Andreas Otto  
**Transiente Prozesse beim Laserstrahlschweißen**  
132 Seiten, 62 Bilder, 1 Tabelle. 1997.  
ISBN 3-87525-089-3
- Band 66: Wolfgang Blöchl  
**Erweiterte Informationsbereitstellung an offenen CNC-Steuerungen zur Prozeß- und Programmoptimierung**  
168 Seiten, 96 Bilder. 1997.  
ISBN 3-87525-091-5
- Band 67: Klaus-Uwe Wolf  
**Verbesserte Prozeßführung und Prozeßplanung zur Leistungs- und Qualitätssteigerung beim Spulenwickeln**  
186 Seiten, 125 Bilder. 1997.  
ISBN 3-87525-092-3
- Band 68: Frank Backes  
**Technologieorientierte Bahnplanung für die 3D-Laserstrahlbearbeitung**  
138 Seiten, 71 Bilder, 2 Tabellen. 1997.  
ISBN 3-87525-093-1
- Band 69: Jürgen Kraus  
**Laserstrahlumformen von Profilen**  
137 Seiten, 72 Bilder, 8 Tabellen. 1997.  
ISBN 3-87525-094-X
- Band 70: Norbert Neubauer  
**Adaptive Strahlführungen für CO<sub>2</sub>-Laseranlagen**  
120 Seiten, 50 Bilder, 3 Tabellen. 1997.  
ISBN 3-87525-095-8

- Band 71: Michael Steber  
**Prozeßoptimierter Betrieb flexibler Schraubstationen in der automatisierten Montage**  
168 Seiten, 78 Bilder, 3 Tabellen. 1997.  
ISBN 3-87525-096-6
- Band 72: Markus Pfestorf  
**Funktionale 3D-Oberflächenkenngrößen in der Umformtechnik**  
162 Seiten, 84 Bilder, 15 Tabellen. 1997.  
ISBN 3-87525-097-4
- Band 73: Volker Franke  
**Integrierte Planung und Konstruktion von Werkzeugen für die Biegebearbeitung**  
143 Seiten, 81 Bilder. 1998.  
ISBN 3-87525-098-2
- Band 74: Herbert Scheller  
**Automatisierte Demontagesysteme und recyclinggerechte Produktgestaltung elektronischer Baugruppen**  
184 Seiten, 104 Bilder, 17 Tabellen. 1998.  
ISBN 3-87525-099-0
- Band 75: Arthur Meßner  
**Kaltmassivumformung metallischer Kleinstteile – Werkstoffverhalten, Wirkflächenreibung, Prozeßauslegung**  
164 Seiten, 92 Bilder, 14 Tabellen. 1998.  
ISBN 3-87525-100-8
- Band 76: Mathias Glasmacher  
**Prozeß- und Systemtechnik zum Laserstrahl-Mikroschweißen**  
184 Seiten, 104 Bilder, 12 Tabellen. 1998.  
ISBN 3-87525-101-6
- Band 77: Michael Schwind  
**Zerstörungsfreie Ermittlung mechanischer Eigenschaften von Feinblechen mit dem Wirbelstromverfahren**  
124 Seiten, 68 Bilder, 8 Tabellen. 1998.  
ISBN 3-87525-102-4
- Band 78: Manfred Gerhard  
**Qualitätssteigerung in der Elektronikproduktion durch Optimierung der Prozeßführung beim Löten komplexer Baugruppen**  
179 Seiten, 113 Bilder, 7 Tabellen. 1998.  
ISBN 3-87525-103-2
- Band 79: Elke Rauh  
**Methodische Einbindung der Simulation in die betrieblichen Planungs- und Entscheidungsabläufe**  
192 Seiten, 114 Bilder, 4 Tabellen. 1998.  
ISBN 3-87525-104-0
- Band 80: Sorin Niederkorn  
**Meßeinrichtung zur Untersuchung der Wirkflächenreibung bei umformtechnischen Prozessen**  
99 Seiten, 46 Bilder, 6 Tabellen. 1998.  
ISBN 3-87525-105-9
- Band 81: Stefan Schubert  
**Regelung der Fokusbildung beim Schweißen mit CO<sub>2</sub>-Hochleistungslasern unter Einsatz von adaptiven Optiken**  
140 Seiten, 64 Bilder, 3 Tabellen. 1998.  
ISBN 3-87525-106-7
- Band 82: Armando Walter Colombo  
**Development and Implementation of Hierarchical Control Structures of Flexible Production Systems Using High Level Petri Nets**  
216 Seiten, 86 Bilder. 1998.  
ISBN 3-87525-109-1
- Band 83: Otto Meedt  
**Effizienzsteigerung bei Demontage und Recycling durch flexible Demontagetechnologien und optimierte Produktgestaltung**  
186 Seiten, 103 Bilder. 1998.  
ISBN 3-87525-108-3

- Band 84: Knuth Götz  
**Modelle und effiziente Modellbildung zur Qualitätssicherung in der Elektronikproduktion**  
212 Seiten, 129 Bilder, 24 Tabellen. 1998.  
ISBN 3-87525-112-1
- Band 85: Ralf Luchs  
**Einsatzmöglichkeiten leitender Klebstoffe zur zuverlässigen Kontaktierung elektronischer Bauelemente in der SMT**  
176 Seiten, 126 Bilder, 30 Tabellen. 1998.  
ISBN 3-87525-113-7
- Band 86: Frank Pöhlau  
**Entscheidungsgrundlagen zur Einführung räumlicher spritzgegossener Schaltungsträger (3-D MID)**  
144 Seiten, 99 Bilder. 1999.  
ISBN 3-87525-114-8
- Band 87: Roland T. A. Kals  
**Fundamentals on the miniaturization of sheet metal working processes**  
128 Seiten, 58 Bilder, 11 Tabellen. 1999.  
ISBN 3-87525-115-6
- Band 88: Gerhard Luhn  
**Implizites Wissen und technisches Handeln am Beispiel der Elektronikproduktion**  
252 Seiten, 61 Bilder, 1 Tabelle. 1999.  
ISBN 3-87525-116-4
- Band 89: Axel Sprenger  
**Adaptives Streckbiegen von Aluminium-Strangpreßprofilen**  
114 Seiten, 63 Bilder, 4 Tabellen. 1999.  
ISBN 3-87525-117-2
- Band 90: Hans-Jörg Pucher  
**Untersuchungen zur Prozeßfolge Umformen, Bestücken und Laserstrahlöten von Mikrokontakten**  
158 Seiten, 69 Bilder, 9 Tabellen. 1999.  
ISBN 3-87525-119-9
- Band 91: Horst Arnet  
**Profilbiegen mit kinematischer Gestalterzeugung**  
128 Seiten, 67 Bilder, 7 Tabellen. 1999.  
ISBN 3-87525-120-2
- Band 92: Doris Schubart  
**Prozeßmodellierung und Technologieentwicklung beim Abtragen mit CO<sub>2</sub>-Laserstrahlung**  
133 Seiten, 57 Bilder, 13 Tabellen. 1999.  
ISBN 3-87525-122-9
- Band 93: Adrianus L. P. Coremans  
**Laserstrahlsintern von Metallpulver – Prozeßmodellierung, Systemtechnik, Eigenschaften laserstrahlgesinterter Metallkörper**  
184 Seiten, 108 Bilder, 12 Tabellen. 1999.  
ISBN 3-87525-124-5
- Band 94: Hans-Martin Biehler  
**Optimierungskonzepte für Qualitätsdatenverarbeitung und Informationsbereitstellung in der Elektronikfertigung**  
194 Seiten, 105 Bilder. 1999.  
ISBN 3-87525-126-1
- Band 95: Wolfgang Becker  
**Oberflächenausbildung und tribologische Eigenschaften excimerlaserstrahlbearbeiteter Hochleistungskeramiken**  
175 Seiten, 71 Bilder, 3 Tabellen. 1999.  
ISBN 3-87525-127-X
- Band 96: Philipp Hein  
**Innenhochdruck-Umformen von Blechpaaren: Modellierung, Prozeßauslegung und Prozeßführung**  
129 Seiten, 57 Bilder, 7 Tabellen. 1999.  
ISBN 3-87525-128-8

- Band 97: Gunter Beifinger  
**Herstellungs- und Prüfverfahren  
für thermoplastische Schaltungsträger**  
169 Seiten, 92 Bilder, 20 Tabellen. 1999.  
ISBN 3-87525-129-6
- Band 98: Jürgen Knoblach  
**Beitrag zur rechnerunterstützten verursachungsgerechten Angebotskalkulation von  
Blechteilen mit Hilfe wissensbasierter Methoden**  
155 Seiten, 53 Bilder, 26 Tabellen. 1999.  
ISBN 3-87525-130-X
- Band 99: Frank Breitenbach  
**Bildverarbeitungssystem zur Erfassung der Anschlußgeometrie  
elektronischer SMT-Bauelemente**  
147 Seiten, 92 Bilder, 12 Tabellen. 2000.  
ISBN 3-87525-131-8
- Band 100: Bernd Falk  
**Simulationsbasierte Lebensdauervorhersage  
für Werkzeuge der Kaltmassivumformung**  
134 Seiten, 44 Bilder, 15 Tabellen. 2000.  
ISBN 3-87525-136-9
- Band 101: Wolfgang Schlögl  
**Integriertes Simulationsdaten-Management  
für Maschinenentwicklung und Anlagenplanung**  
169 Seiten, 101 Bilder, 20 Tabellen. 2000.  
ISBN 3-87525-137-7
- Band 102: Christian Hinsel  
**Ermüdungsbruchversagen hartstoffbeschichteter  
Werkzeugstähle in der Kaltmassivumformung**  
130 Seiten, 80 Bilder, 14 Tabellen. 2000.  
ISBN 3-87525-138-5
- Band 103: Stefan Bobbert  
**Simulationsgestützte Prozessauslegung  
für das Innenhochdruck-Umformen von Blechpaaren**  
123 Seiten, 77 Bilder. 2000.  
ISBN 3-87525-145-8
- Band 104: Harald Rottbauer  
**Modulares Planungswerkzeug  
zum Produktionsmanagement in der Elektronikproduktion**  
166 Seiten, 106 Bilder. 2001.  
ISBN 3-87525-139-3
- Band 105: Thomas Hennige  
**Flexible Formgebung von Blechen durch Laserstrahlumformen**  
119 Seiten, 50 Bilder. 2001.  
ISBN 3-87525-140-7
- Band 106: Thomas Menzel  
**Wissensbasierte Methoden für die rechnergestützte Charakterisierung  
und Bewertung innovativer Fertigungsprozesse**  
152 Seiten, 71 Bilder. 2001.  
ISBN 3-87525-142-3
- Band 107: Thomas Stöckel  
**Kommunikationstechnische Integration der Prozeßebene  
in Produktionssysteme durch Middleware-Frameworks**  
147 Seiten, 65 Bilder, 5 Tabellen. 2001.  
ISBN 3-87525-143-1
- Band 108: Frank Pitter  
**Verfügbarkeitssteigerung von Werkzeugmaschinen  
durch Einsatz mechatronischer Sensorlösungen**  
158 Seiten, 131 Bilder, 8 Tabellen. 2001.  
ISBN 3-87525-144-X
- Band 109: Markus Korneli  
**Integration lokaler CAP-Systeme  
in einen globalen Fertigungsdatenverbund**  
121 Seiten, 53 Bilder, 11 Tabellen. 2001.  
ISBN 3-87525-146-6

Band 110: Burkhard Müller  
**Laserstrahljustieren mit Excimer-Lasern – Prozeßparameter und Modelle zur Aktorkonstruktion**  
128 Seiten, 36 Bilder, 9 Tabellen. 2001  
ISBN 3-87525-159-8

Band 111: Jürgen Göhringer  
**Integrierte Telediagnose via Internet zum effizienten Service von Produktionssystemen**  
178 Seiten, 98 Bilder, 5 Tabellen. 2001.  
ISBN 3-87525-147-4

Band 112: Robert Feuerstein  
**Qualitäts- und kosteneffiziente Integration neuer Bauelementetechnologien in die Flachbaugruppenfertigung**  
161 Seiten, 99 Bilder, 10 Tabellen. 2001.  
ISBN 3-87525-151-2

Band 113: Marcus Reichenberger  
**Eigenschaften und Einsatzmöglichkeiten alternativer Elektroniklote in der Oberflächenmontage (SMT)**  
165 Seiten, 97 Bilder, 18 Tabellen. 2001.  
ISBN 3-87525-152-0

Band 114: Alexander Huber  
**Justieren vormontierter Systeme mit dem Nd:YAG-Laser unter Einsatz von Aktoren**  
122 Seiten, 58 Bilder, 5 Tabellen. 2001.  
ISBN 3-87525-153-9

Band 115: Sami Krimi  
**Analyse und Optimierung von Montagesystemen in der Elektronikproduktion**  
155 Seiten, 88 Bilder, 3 Tabellen. 2001.  
ISBN 3-87525-157-1

Band 116: Marion Merklein  
**Laserstrahlumformen von Aluminiumwerkstoffen - Beeinflussung der Mikrostruktur und der mechanischen Eigenschaften**  
122 Seiten, 65 Bilder, 15 Tabellen. 2001.  
ISBN 3-87525-156-3

Band 117: Thomas Collisi  
**Ein informationslogistisches Architekturkonzept zur Akquisition simulationsrelevanter Daten**  
181 Seiten, 105 Bilder, 7 Tabellen. 2002.  
ISBN 3-87525-164-4

Band 118: Markus Koch  
**Rationalisierung und ergonomische Optimierung im Innenausbau durch den Einsatz moderner Automatisierungstechnik**  
176 Seiten, 98 Bilder, 9 Tabellen. 2002.  
ISBN 3-87525-165-2

Band 119: Michael Schmidt  
**Prozeßregelung für das Laserstrahl-Punktschweißen in der Elektronikproduktion**  
152 Seiten, 71 Bilder, 3 Tabellen. 2002.  
ISBN 3-87525-166-0

Band 120: Nicolas Tiesler  
**Grundlegende Untersuchungen zum Fließpressen metallischer Kleinstteile**  
126 Seiten, 78 Bilder, 12 Tabellen. 2002.  
ISBN 3-87525-175-X

Band 121: Lars Pursche  
**Methoden zur technologieorientierten Programmierung für die 3D-Lasermikrobearbeitung**  
111 Seiten, 39 Bilder, 0 Tabellen. 2002.  
ISBN 3-87525-183-0

Band 122: Jan-Oliver Brassel  
**Prozeßkontrolle beim Laserstrahl-Mikroschweißen**  
148 Seiten, 72 Bilder, 12 Tabellen. 2002.  
ISBN 3-87525-181-4

Band 123: Mark Geisel

**Prozeßkontrolle und –steuerung beim Laserstrahlschweißen mit den Methoden der nichtlinearen Dynamik**

135 Seiten, 46 Bilder, 2 Tabellen. 2002.  
ISBN 3-87525-180-6

Band 124: Gerd Eßer

**Laserstrahlunterstützte Erzeugung metallischer Leiterstrukturen auf Thermoplastsubstraten für die MID-Technik**

148 Seiten, 60 Bilder, 6 Tabellen. 2002.  
ISBN 3-87525-171-7

Band 125: Marc Fleckenstein

**Qualität laserstrahl-gefügter Mikroverbindungen elektronischer Kontakte**

159 Seiten, 77 Bilder, 7 Tabellen. 2002.  
ISBN 3-87525-170-9

Band 126: Stefan Kaufmann

**Grundlegende Untersuchungen zum Nd:YAG- Laserstrahlfügen von Silizium für Komponenten der Optoelektronik**

159 Seiten, 100 Bilder, 6 Tabellen. 2002.  
ISBN 3-87525-172-5

Band 127: Thomas Fröhlich

**Simultanes Löten von Anschlußkontakten elektronischer Bauelemente mit Diodenlaserstrahlung**

143 Seiten, 75 Bilder, 6 Tabellen. 2002.  
ISBN 3-87525-186-5

Band 128: Achim Hofmann

**Erweiterung der Formgebungsgrenzen beim Umformen von Aluminiumwerkstoffen durch den Einsatz prozessangepasster Platinen**

113 Seiten, 58 Bilder, 4 Tabellen  
ISBN 3-87525-182-2

Band 129: Ingo Kriebitzsch

**3 - D MID Technologie in der Automobilelektronik**

129 Seiten, 102 Bilder, 10 Tabellen. 2002.  
ISBN 3-87525-169-5

Band 130: Thomas Pohl

**Fertigungsqualität und Umformbarkeit laserstrahlgeschweißter Formplatinen aus Aluminiumlegierungen**

133 Seiten, 93 Bilder, 12 Tabellen. 2002  
ISBN 3-87525-173-3

Band 131: Matthias Wenk

**Entwicklung eines konfigurierbaren Steuerungssystems für die flexible Sensorführung von Industrierobotern**

167 Seiten, 85 Bilder, 1 Tabelle. 2002.  
ISBN 3-87525-174-1

Band 132: Matthias Negendanck

**Neue Sensorik und Aktorik für Bearbeitungsköpfe zum Laserstrahlschweißen**

116 Seiten, 60 Bilder, 14 Tabellen  
ISBN 3-87525-184-9

Band 133: Oliver Kreis

**Integrierte Fertigung –**

**Verfahrensintegration durch Innenhochdruck-Umformen, Trennen und Laserstrahlschweißen in einem Werkzeug sowie ihre tele- und multimediale Präsentation**

167 Seiten, 90 Bilder, 43 Tabellen  
ISBN 3-87525-176-8

Band 134: Stefan Trautner

**Technische Umsetzung produktbezogener Instrumente der Umweltpolitik bei Elektro- und Elektronikgeräten**

179 Seiten, 92 Bilder, 11 Tabellen. 2002.  
ISBN 3-87525-177-6

Band 135: Roland Meier

**Strategien für einen produktorientierten Einsatz räumlicher  
spritzgegossener Schaltungsträger (3-D MID)**

155 Seiten, 88 Bilder, 14 Tabellen. 2002.  
ISBN 3-87525-178-4

Band 136: Jürgen Wunderlich

**Kostensimulation – Simulationsbasierte Wirtschaftlichkeitsregelung  
komplexer Produktionssysteme**

202 Seiten, 119 Bilder, 17 Tabellen. 2002.  
ISBN 3-87525-179-2

Band 137: Stefan Novotny

**Innenhochdruck-Umformen von Blechen aus Aluminium- und  
Magnesiumlegierungen bei erhöhter Temperatur**

132 Seiten, 82 Bilder, 6 Tabellen. 2002.  
ISBN 3-87525-185-7

Band 138: Andreas Licha

**Flexible Montageautomatisierung zur Komplettmontage flächenhafter  
Produktstrukturen durch kooperierende Industrieroboter**

158 Seiten, 87 Bilder, 8 Tabellen. 2003.  
ISBN 3-87525-189-X

Band 139: Michael Eisenbarth

**Beitrag zur Optimierung der Aufbau- und Verbindungstechnik  
für mechatronische Baugruppen**

207 Seiten, 141 Bilder, 9 Tabellen. 2003.  
ISBN 3-87525-190-3

Band 140: Frank Christoph

**Durchgängige simulationsgestützte Planung von  
Fertigungseinrichtungen der Elektronikproduktion**

187 Seiten, 107 Bilder, 9 Tabellen. 2003.  
ISBN 3-87525-191-1

Band 141: Hinnerk Hagenah

**Simulationsbasierte Bestimmung der zu erwartenden  
Maßhaltigkeit für das Blechbiegen**

131 Seiten, 36 Bilder, 26 Tabellen. 2003.  
ISBN 3-87525-192-X

Band 142: Ralf Eckstein

**Scherschneiden und Biegen metallischer Kleinstteile –  
Materialeinfluss und Materialverhalten**

148 Seiten, 71 Bilder, 19 Tabellen. 2003.  
ISBN 3-87525-193-8

Band 143: Frank H. Meyer-Pitroff

**Excimerlaserstrahlbiegen dünner metallischer Folien  
mit homogener Lichtlinie**

138 Seiten, 60 Bilder, 16 Tabellen. 2003.  
ISBN 3-87525-196-2

Band 144: Andreas Kach

**Rechnergestützte Anpassung von Laserstrahlschneidbahnen  
an Bauteilabweichungen**

139 Seiten, 69 Bilder, 11 Tabellen. 2004.  
ISBN 3-87525-197-0

Band 145: Stefan Hierl

**System- und Prozeßtechnik für das simultane Löten mit  
Diodenlaserstrahlung von elektronischen Bauelementen**

124 Seiten, 66 Bilder, 4 Tabellen. 2004.  
ISBN 3-87525-198-9

Band 146: Thomas Neudecker

**Tribologische Eigenschaften keramischer Blechumformwerkzeuge-  
Einfluss einer Oberflächenendbearbeitung mittels Excimerlaserstrahlung**

166 Seiten, 75 Bilder, 26 Tabellen. 2004.  
ISBN 3-87525-200-4

Band 147: Ulrich Wenger

**Prozessoptimierung in der Wickeltechnik durch innovative maschinenbauliche und regelungstechnische Ansätze**

132 Seiten, 88 Bilder, 0 Tabellen. 2004.

ISBN 3-87525-203-9

Band 148: Stefan Slama

**Effizienzsteigerung in der Montage durch marktorientierte Montagestrukturen und erweiterte Mitarbeiterkompetenz**

188 Seiten, 125 Bilder, 0 Tabellen. 2004.

ISBN 3-87525-204-7

Band 149: Thomas Wurm

**Laserstrahljustieren mittels Aktoren – Entwicklung von Konzepten und Methoden für die rechnerunterstützte Modellierung und Optimierung von komplexen Aktorsystemen in der Mikrotechnik**

122 Seiten, 51 Bilder, 9 Tabellen. 2004.

ISBN 3-87525-206-3

Band 150: Martino Celeghini

**Wirkmedienbasierte Blechumformung: Grundlagenuntersuchungen zum Einfluss von Werkstoff und Bauteilgeometrie**

146 Seiten, 77 Bilder, 6 Tabellen. 2004.

ISBN 3-87525-207-1

Band 151: Ralph Hohenstein

**Entwurf hochdynamischer Sensor- und Regelsysteme für die adaptive Laserbearbeitung**

282 Seiten, 63 Bilder, 16 Tabellen. 2004.

ISBN 3-87525-210-1

Band 152: Angelika Hutterer

**Entwicklung prozessüberwachender Regelkreise für flexible Formgebungsprozesse**

149 Seiten, 57 Bilder, 2 Tabellen. 2005.

ISBN 3-87525-212-8

Band 153: Emil Egerer

**Massivumformen metallischer Kleinsteile bei erhöhter Prozesstemperatur**

158 Seiten, 87 Bilder, 10 Tabellen. 2005.

ISBN 3-87525-213-6

Band 154: Rüdiger Holzmann

**Strategien zur nachhaltigen Optimierung von Qualität und Zuverlässigkeit in der Fertigung hochintegrierter Flachbaugruppen**

186 Seiten, 99 Bilder, 19 Tabellen. 2005.

ISBN 3-87525-217-9

Band 155: Marco Nock

**Biegeumformen mit Elastomerwerkzeugen**  
Modellierung, Prozessauslegung und Abgrenzung des Verfahrens am Beispiel des Rohrbiegens

164 Seiten, 85 Bilder, 13 Tabellen. 2005.

ISBN 3-87525-218-7

Band 156: Frank Niebling

**Qualifizierung einer Prozesskette zum Laserstrahlsintern metallischer Bauteile**

148 Seiten, 89 Bilder, 3 Tabellen. 2005.

ISBN 3-87525-219-5

Band 157: Markus Meiler

**Großserientauglichkeit trockenschmierstoffbeschichteter Aluminiumbleche im Presswerk**

Grundlegende Untersuchungen zur Tribologie, zum Umformverhalten und Bauteilversuche

104 Seiten, 57 Bilder, 21 Tabellen. 2005.

ISBN 3-87525-221-7

- Band 158: Agus Sutanto  
**Solution Approaches for Planning of Assembly Systems in Three-Dimensional Virtual Environments**  
169 Seiten, 98 Bilder, 3 Tabellen. 2005.  
ISBN 3-87525-220-9
- Band 159: Matthias Boiger  
**Hochleistungssysteme für die Fertigung elektronischer Baugruppen auf der Basis flexibler Schaltungsträger**  
175 Seiten, 111 Bilder, 8 Tabellen. 2005.  
ISBN 3-87525-222-5
- Band 160: Matthias Pitz  
**Laserunterstütztes Biegen höchstfester Mehrphasenstähle**  
120 Seiten, 73 Bilder, 11 Tabellen. 2005.  
ISBN 3-87525-223-3
- Band 161: Meik Vahl  
**Beitrag zur gezielten Beeinflussung des Werkstoffflusses beim Innenhochdruck-Umformen von Blechen**  
165 Seiten, 94 Bilder, 15 Tabellen. 2005.  
ISBN 3-87525-224-1
- Band 162: Peter K. Kraus  
**Plattformstrategien – Realisierung einer varianz- und kostenoptimierten Wertschöpfung**  
181 Seiten, 95 Bilder, 0 Tabellen. 2005.  
ISBN 3-87525-226-8
- Band 163: Adrienn Cser  
**Laserstrahlschmelzabtrag – Prozessanalyse und -modellierung**  
146 Seiten, 79 Bilder, 3 Tabellen. 2005.  
ISBN 3-87525-227-6
- Band 164: Markus C. Hahn  
**Grundlegende Untersuchungen zur Herstellung von Leichtbauverbundstrukturen mit Aluminiumschaumkern**  
143 Seiten, 60 Bilder, 16 Tabellen. 2005.  
ISBN 3-87525-228-4
- Band 165: Gordana Michos  
**Mechatronische Ansätze zur Optimierung von Vorschubachsen**  
146 Seiten, 87 Bilder, 17 Tabellen. 2005.  
ISBN 3-87525-230-6
- Band 166: Markus Stark  
**Auslegung und Fertigung hochpräziser Faser-Kollimator-Arrays**  
158 Seiten, 115 Bilder, 11 Tabellen. 2005.  
ISBN 3-87525-231-4
- Band 167: Yurong Zhou  
**Kollaboratives Engineering Management in der integrierten virtuellen Entwicklung der Anlagen für die Elektronikproduktion**  
156 Seiten, 84 Bilder, 6 Tabellen. 2005.  
ISBN 3-87525-232-2
- Band 168: Werner Enser  
**Neue Formen permanenter und lösbarer elektrischer Kontaktierungen für mechatronische Baugruppen**  
190 Seiten, 112 Bilder, 5 Tabellen. 2005.  
ISBN 3-87525-233-0
- Band 169: Katrin Melzer  
**Integrierte Produktpolitik bei elektrischen und elektronischen Geräten zur Optimierung des Product-Life-Cycle**  
155 Seiten, 91 Bilder, 17 Tabellen. 2005.  
ISBN 3-87525-234-9
- Band 170: Alexander Putz  
**Grundlegende Untersuchungen zur Erfassung der realen Vorspannung von armierten Kaltfließpresswerkzeugen mittels Ultraschall**  
137 Seiten, 71 Bilder, 15 Tabellen. 2006.  
ISBN 3-87525-237-3

Band 171: Martin Prechtl  
**Automatisiertes Schichtverfahren für metallische Folien –  
System- und Prozesstechnik**  
154 Seiten, 45 Bilder, 7 Tabellen. 2006.  
ISBN 3-87525-238-1

Band 172: Markus Meidert  
**Beitrag zur deterministischen Lebensdauerabschätzung  
von Werkzeugen der Kaltmassivumformung**  
131 Seiten, 78 Bilder, 9 Tabellen. 2006.  
ISBN 3-87525-239-X

Band 173: Bernd Müller  
**Robuste, automatisierte Montagesysteme durch adaptive Prozessführung  
und montageübergreifende Fehlerprävention am Beispiel flächiger Leichtbauteile**  
147 Seiten, 77 Bilder, 0 Tabellen. 2006.  
ISBN 3-87525-240-3

Band 174: Alexander Hofmann  
**Hybrides Laserdurchstrahlschweißen von Kunststoffen**  
136 Seiten, 72 Bilder, 4 Tabellen. 2006.  
ISBN 978-3-87525-243-9  
ISBN 3-87525-243-8

Band 175: Peter Wölflick  
**Innovative Substrate und Prozesse mit feinsten Strukturen  
für bleifreie Mechatronik-Anwendungen**  
177 Seiten, 148 Bilder, 24 Tabellen. 2006.  
ISBN 978-3-87525-246-0  
ISBN 3-87525-246-2

Band 176: Attila Komlódi  
**Detection and Prevention of Hot Cracks during Laser Welding  
of Aluminium Alloys Using Advanced Simulation Methods**  
155 Seiten, 89 Bilder, 14 Tabellen. 2006.  
ISBN 978-3-87525-248-4  
ISBN 3-87525-248-9

Band 177: Uwe Popp  
**Grundlegende Untersuchungen zum Laserstrahlstrukturieren  
von Kaltmassivumformwerkzeugen**  
140 Seiten, 67 Bilder, 16 Tabellen. 2006.  
ISBN 978-3-87525-249-1  
ISBN 3-87525-249-7

Band 178: Veit Rückel  
**Rechnergestützte Ablaufplanung und Bahngenerierung  
Für kooperierende Industrieroboter**  
148 Seiten, 75 Bilder, 7 Tabellen. 2006.  
ISBN 978-3-87525-250-7  
ISBN 3-87525-250-0

Band 179: Manfred Dirscherl  
**Nicht-thermische Mikrojustiertechnik mittels ultrakurzer Laserpulse**  
154 Seiten, 69 Bilder, 10 Tabellen. 2007.  
ISBN 978-3-87525-251-4  
ISBN 3-87525-251-9

Band 180: Yong Zhuo  
**Entwurf eines rechnergestützten integrierten Systems für Konstruktion und  
Fertigungsplanung räumlicher spritzgegossener Schaltungsträger (3D-MID)**  
181 Seiten, 95 Bilder, 5 Tabellen. 2007.  
ISBN 978-3-87525-253-8