

Inhaltsverzeichnis

Über den Autor	9	Unload (Form)	45
Einführung	10	Change (Textbox)	45
1. Hintergrundwissen über Visual Basic	13	Event-Aufruf	45
1.1 Windows	13	4.2 Methoden	47
1.2 Die objektorientierte Programmierung	14	5. Die Programmiersprache Visual Basic	48
1.3 Wo liegen die Vorteile von OOP?	15	5.1 Variablen	49
1.4 Definitionen im Überblick	15	Datentypen unter Visual Basic	50
1.5 Visual Basic.NET: Eine neue Gattung	17	Datentypen unter NET	52
2. Erkundung der Visual Basic Umgebung	18	5.2 Arrays	54
2.1 Ein Visual Basic Projekt starten	18	DIM	54
Projekt-Stile	18	ReDim	55
Zusätzliche Projektstile für NET-Framework	19	Ubound	56
2.2 Die Programmierumgebung	20	Lbound	57
Die Menüleiste	21	Array	58
Funktionszugang über die Toolleiste	21	Arrays im NET-Framework	58
Der Objektbrowser (Toolbox)	22	Sortierung (Nur NET)	59
Der Projekt-Navigator	22	Binärsuche (Nur NET)	60
Der Eigenschaften-Navigator oder Inspektor	22	5.3 Typen	60
Das Layout-Fenster	22	5.4 Konstanten	62
Die Form-Ansicht	22	5.5 Aufzählungen	62
Die Code-Ansicht	23	5.6 Sammlungen	64
Das Hilfesystem	25	5.7 Der Aufgabenbereich von Variablen	66
3. Objekte und Steuerelemente	26	Public / Global	66
3.1 Die Form	26	Private	68
3.2 Die Steuerelemente	27	Static	68
3.3 Die Standard-Steuerelemente von Visual Basic 5.0 und 6.0	29	5.8 Arbeitsebenen	68
3.4 Gebräuchliche Steuerelemente aus COMCTRL unter VB 5.0 und 6.0	30	5.9 Subroutinen und Funktionen	71
3.5 Gebräuchliche Dialog-Steuerelemente aus CMDLG unter VB 5.0 und 6.0	31	Subroutinen	71
3.6 Zusätzliche Steuerelemente, die nur für die NET-Umgebung existieren	32	Funktionen	72
3.7 Serielle Kommunikation mit Control MSCOMM	34	5.10 Der Geltungsbereich von Prozeduren	73
3.8 Menüs	35	5.11 Numerische Operatoren	74
Der Menü-Editor von NET	36	5.12 Einfache Konvertierungen	75
3.9 Eigenschaften im Detail	38	5.13 Logische Operatoren	76
Die Eigenschaft Name	38	5.14 Shift-Operatoren (Nur NET)	76
Die Eigenschaften Oben, Links, Höhe und Breite	39	5.15 Flusssteuerung	77
Die Eigenschaften Hintergrund-, Vordergrund- und Textfarbe	39	If – Then – Else	77
Titel und Text	39	Select Case	79
Aktiviert und Sichtbar	40	5.16 Schleifen	80
Index	40	For – Next	80
Tabindex	40	While – Wend	81
Tooltip Text	40	Do Until / While	82
4. Events und Methoden	42	Die For-Each-Konstruktion	83
4.1 Das Belauschen der Events	42	5.17 String-Manipulationen Left\$ – Right\$ – Ltrim\$ – Rtrim\$ Left\$	84
Click (Bestandteil der meisten Steuerungen)	43	Right\$	84
DbClick (Bestandteil der meisten Steuerungen)	43	Mid\$	84
KeyPress (Bestandteil der meisten Steuerungen)	43	Ltrim\$ / Rtrim\$ /Trim\$	84
MouseMove (Bestandteil der meisten Steuerungen)	44	Ucase\$ und Lcase\$	85
Activate (Form)	44	VAL und STR\$	85
Deactivate (Form)	44	LEN	85
Load (Form)	44	INSTR	86
		String-Verkettung	86
		Der Umgang mit Strings in der NET-Umgebung	87
		Vergleichbare Stringmethoden unter NET	87
		Die Methoden Padleft und Padright (Nur NET)	87
		Insert und Remove (Nur NET)	88
		Join und Split (Nur NET)	88
		5.18 Datei-Manipulationen unter Visual Basic Classic	89

Dateihandle	89
Strukturen zum Öffnen und Schließen von Dateien	92
Ausgabe-Modus	93
Einfüge-Modus	94
Eingabe-Modus	94
Binär-Modus	94
Random-Modus	95
Etwas in eine Datei speichern	95
PRINT-Konstruktionen für Objekt- und Dateiausgabe	95
Line Input	97
Write	97
Input	98
Get und Put	99
Die Such-Operation	100
Bestimmen von Länge und Ende einer Datei	101
Dateinamen	102
Der Befehl DIR und die Existenz einer Datei	102
Dateien mit der Kill-Funktion entfernen	104
Verzeichnisse und Laufwerke wechseln	104
Das Erstellen und Entfernen von Verzeichnissen	105
Den aktuellen Ort bestimmen	106
5.19 FileSystemObject	106
FileSystemObject einrichten	107
Die Laufwerkssammlung	107
Die Methoden von FileSystemObject	109
Textstream-Ein-/Ausgabe mit FileSystemObject	110
5.20 Datei-Ein-/Ausgabe im NET-Framework	111
Die Verwendung der FileStream-Klasse	111
Schreib- und Lesezugriff auf einen Stream	112
StreamWriter und StreamReader im Detail	113
Details des StringReaders und StringWriters	120
Details des BinaryReaders und BinaryWriters	120
Änderungen am Dateisystem überwachen (NET)	122
Beispiel: Dateisystemüberwachung unter NET	124
Dateimanipulationen (Nur NET)	125
Verzeichnisänderungen (Nur NET)	128
6. Details über die NET-Plattform	132
6.1 Namensräume	132
Den eigenen Namensraum erstellen	133
6.2 Threads	133
Einen Thread erstellen	134
Beenden eines Threads und Rückkehr zum Hauptprogramm	135
Überwachen und Verändern der Thread-Ausführung	136
Hintergrund/Vordergrund und Priorität	139
Die Sicherheit des Threads	140
7. Der Aufbau eines User-Interfaces	142
7.1 Die Form entwerfen	142
7.2 Arrays aus Objekten und Steuerungen	144
Die Verwendung von Steuerungsarrays spart Speicherplatz	145
NET kennt keine Steuerungsarrays	145
Steuerungen während des Programmablaufs in VB Classic erstellen	146
Steuerungen während des Programmablaufs in NET erstellen	149
7.3 Konsolen-Anwendungen (NET)	149
In die Kommandozeile schreiben	150
Eingaben aus der Kommandozeile empfangen	150
8. Code an Events anknüpfen	151
8.1 Code an Events in VB Classic anhängen	151
Code anknüpfen	152
8.2 Event-Handler unter NET angliedern	153
Die Handlekonstruktion	154
Die Methode AddHandler	155
Die Methode RemoveHandler	155
9. Ein Programm laufen lassen und von Fehlern befreien	156
9.1 Ein Programm starten	156
Start, Pause, Stopp	156
9.2 Ein Programm debuggen	157
9.3 Variablen untersuchen	158
9.4 Fortgeschrittenes Debuggen: Das Überwachungsfenster	158
Elemente des Überwachungsfensters	159
Der Befehl Add Watch	159
Die Überwachungs-Dialogbox hinzufügen	159
Der Befehl Quick Watch (Shift F9)	160
Der Befehl Edit Watch	160
Das Edit Watch Fenster	160
9.5 Die Verwendung von Breakpoints	161
9.6 Das Objekt Debug	162
9.7 Debuggen unter NET	163
Debug.Print und der Debugger unter NET	163
10. Die Programmverteilung	164
10.1 Die ersten Schritte	164
10.2 Ausgabemedium angeben	165
11. Projekte mit mehreren Modulen	168
11.1 Mehrere Formen	168
Die Formen von Visual Basic Classic	168
Formen unter Visual Basic NET	169
11.2 Module	170
11.3 Zugriffe aus anderen Bereichen des Programms	170
11.4 Die Analogie zwischen Verzeichnisstruktur und einem Projekt	171
12. Mehrere Fallstudien	173
12.1 Fallstudie 1: Ein kleiner Texteditor	173
Der Entwurf des User-Interfaces	173
Code hinzufügen	176
12.2 Fallstudie 2: Ein Rechner	179
Der Entwurf des User-Interfaces	180
Programmcode schreiben	182
Programmcode für das User-Interface	183
12.3 Der Rechner als NET-Anwendung	188
Einführung in Teil II	191
13. Einen Schritt weiter	192
13.1 Veränderungen an erweiterten Formen	192
Laden	192
Entladen	193
Zeigen	193
Verstecken	194
Formen, modal und nicht modal	194
MDI-Formen	195
13.2 Zusätzliche Form-Eigenschaften unter NET	196
Lichtdurchlässigkeit	196
Die Tasten Akzeptieren und Abbrechen	196
Größenziehpunkt	197
Autoscroll	197
13.3 Menüs	198

Popup-Menüs (Classic) und Kontext-Menüs (NET)	198	Farben	239
13.4 Menüs über den Programmcode verändern	202	Brushes	240
Menübefehle aktivieren oder deaktivieren	203	Stifte	241
Prüfmarkierungen in einer Menüsteuerung anzeigen	203	Linien ziehen	242
Menüsteuerungen unsichtbar machen	204	Zeichenfiguren	242
Menüsteuerungen während der Laufzeit		Figuren füllen	246
hinzufügen (VB Classic)	205	Grafiken zeichnen	247
Menüsteuerungen während der Laufzeit		Text zeichnen	247
hinzufügen (VB NET)	205	Pfade	248
13.5 Spezielle Menüfunktionen	207	Regionen	249
Die Fensterliste	207		
Menüs übertragen	208	15. Programme mit Eingabeaufforderung (NET) 250	
13.6 Optionswahl	208	15.1 Standard I/O	250
Checkboxes	209	In die Kommandozeile schreiben	250
Options- oder Radio-Buttons	209	Input aus der Kommandozeile	250
Radio-Buttons gruppieren (VB Classic)	210	Input/Output umleiten	251
Radio-Buttons gruppieren (VB NET)	210	Kommandozeilen-Parameter empfangen	251
Listboxen	211	16. Kommunizieren mit der Welt da draußen. 253	
13.7 Timerobjekte	212	16.1 Sendkeys – ein einfacher Kommunikationsweg	253
Genauere Zeitsteuerung	213	AppActivate	255
Zeitmessung	213	Shell 256	
13.8 Objekte für die User-Eingabe	215	16.2 DDE – eine andere Art der internen Programm-	
Textboxen	215	kommunikation	257
Die Combobox	216	Verbindungsmodus	257
13.9 Drucken	218	LinkTopic	258
Das Objekt Printer	218	LinkItem	259
Etwas zum Drucker schicken	219	16.3 Serielle I/O: Sprechen mit der Welt hinter dem Port.	260
Ein Vorschaufenster unter VB Classic	219	Ein Objekt einfügen	260
Die Druckvorschau (NET)	220	PortOpen	261
Text für den Ausdruck	220	Handshake	262
Grafik für den Ausdruck	221	Einstellungen	263
EndDoc	221	Größe des Eingangs- und Ausgangspuffers	264
13.10 Vorteile der Windows 95-Oberfläche	221	Zählung im Eingangs- und Ausgangspuffer	264
13.11 Das Aussehen von XP	222	Ersetzen des Paritätscodes	265
Die Manifestdatei	223	DTR-Bereitschaft	265
Die richtige Bibliothek unter Visual Basic Classic	224	Das Event OnComm	266
XP-Stil unter NET	225	CommEvent	266
14. Grafiken 226		Rthreshold, Sthreshold	267
14.1 Einfache Arbeiten mit Koordinaten	226	16.4 Serieller Input/Output unter NET	268
CurrentX, CurrentY	226	Die Definition der Steuerungsstruktur	268
14.2 Zeichnungseinstellungen	227	I/O-Prototypen	269
Linienbreite	227	Ein Device-Handle für den seriellen Port	271
Zeichenmodus	227	Die Steuerung der Port-Einstellungen	271
Zeichenstil	228	Timeout	272
Füllfarbe	229	Etwas an den Port schreiben	272
Die Füllung	229	Daten am Port lesen	273
14.3 Einfache Zeichnungen	230	Das Schließen des Ports	273
Pset	230	Beispielprogramm	273
Linie	230	16.5 Serieller Input/Output unter Visual Basic 2005	
Kreis	232	(NET 2.0)	274
14.4 Grafiken speichern und laden	232	16.6 TCP/IP für VB Classic – Windows Anschlüsse mit Winsock	274
Grafiken speichern	232	TCP-Grundlagen	274
Grafiken laden	233	UDP-Grundlagen	275
14.5 Das Koordinatensystem	235	RemoteHost und RemoteHostIP	276
Skalieren	235	Lokaler Hostname und lokale IP	276
Skalierungsmodus	236	Protokoll	276
Skalierungsgrößen	236	Status	277
ScaleLeft und ScaleTop	237	Verbindungsanfrage und Bestätigung	277
14.6 Die Grafikumgebung unter NET	237	Ankommende Daten und deren Weiterverarbeitung	278
Das Objekt Punkt	238	16.7 TCP/IP unter NET	280
Das Objekt Größe	238	Zugriff auf einen Webserver	280
Das Objekt Rechteck	238	Der Download einer Datei von einem Webserver	280

Zugriff auf einen auf TCP/IP aufsetzenden Dienst	281
Daten im Netzwerk verteilen	281
17. Weitere Fallstudien	284
17.1 Fallstudie 3: Doodle, ein Grafikprogramm	284
17.2 Fallstudie 4: Das Datenterminal	289
17.3 Fallstudie 5: Alpha Server – ein Telnet-Server	294
Ein ausgefeilterer Server unterstützt mehrere Sessions	295
17.4 Fallstudie 6: AlphaServer unter NET.	302
17.5 Fallstudie 7: LoanCalc – der Einsatz von Exel in Ihrer Anwendung	302
Einführung in Teil III: Master-Programmierung mit Visual Basic	304
18. Windows Schatzsuche	305
18.1 DLL's	305
Der Zugriff auf DLL-Routinen	306
18.2 Parameterübergabe an Prozeduren und Funktionen	307
18.3 API-Programmierung.	309
Ein einfaches API-Beispiel	310
18.4 Die API-Programmierung unter NET	312
19. Der Aufbau von ActiveX-Steuerungen	313
19.1 Ein ActiveX-Objekt erstellen.	314
19.2 Eigenschaften und Events ergänzen	317
19.3 Die Wizard-Hilfen	322
19.4 Ein genauer Blick auf den endgültigen Code	324
19.5 Eine eigene Steuerung im NET-Framework	327
Ein Komponenten-Projekt	327
Eigenschaften hinzufügen	327
Methoden einfügen	328
Events einbinden	329
Variablen und Aufzählungen einflechten	329
Zusammenarbeit mit der VB IDE	329
20. Bessere Programme schreiben	332
20.1 Das KISS-Prinzip	332
20.2 Atom-Programmierung.	335
20.3 Objektnamen	336
20.4 Fehlerbehandlung in Visual Basic Classic	337
Die Bedingung On Error Goto	337
Das Objekt Err	339
Wiederaufnahme der Programmausführung nach einer Fehlerbehandlung	340
Erkennbare Fehler	341
Syntaxfehler (Verstöße gegen Syntaxregeln).	341
Laufzeitfehler	342
Fehler durch fehlerhafte Programmierlogik	342
Dateiverarbeitungsfehler	342
20.5 Fehlerbehandlung im NET-Framework	343
21. Die Registry von Windows	345
21.1 In die Registry eindringen	345
21.2 Datenrecherche in der Registry	346
GetSetting	347
SaveSetting	347
DeleteSetting	348
GetAllSettings.	349
21.3 Die Registry nutzen.	349
21.4 Die Registry und NET.	350
Der Namensraum Microsoft.Win32	350
22. Skript-Interpreter	352
22.1 Der Bau eines einfachen Skript-Interpreters	352
Das Skript laufen lassen	353
Der Skript-Parser	354
Parameterentnahme.	355
22.2 MSScript: Ein echter Skript-Interpreter.	357
Skriptsprache	357
Die Eigenschaften von MSScript	358
Steuermethoden für Skripte	358
Code-Erweiterung für die Skriptengine	359
Objektfreigabe	359
23. Visual Basic für Applikationen	360
23.1 Was bedeutet Visual Basic für Applikationen?	360
23.2 Der Einsatz von VBA zur Codeentwicklung unter Word oder Exel	360
24. Klassen	361
24.1 Das Klassen-Konzept.	361
24.2 Eine Klasse erzeugen.	361
24.3 Ein Objekt aus einer Klasse erstellen	363
24.4 Ein praktisches Beispiel.	363
24.5 Erstellen einer Klasse in NET	364
25. Weitere Fallstudien	365
25.1 Fallstudie 7: Windows via API-Aufruf beenden.	365
25.2 Fallstudie 8: Die ActiveX-Steuerung LED.	366
25.3 Fallstudie 9: MiniBasic – Eine Programmier- umgebung für MSScript.	371
25.4 Fallstudie 10: Anmerkungen zum Einsatz von Klassen	374
Einführung in Teil IV: Visual Basic und Hardware	377
26. Der Computer	378
26.1 Der PC: Ein historischer Überblick	378
26.2 Der PC: Beschreibung der Hardware	379
26.3 Die I/O-PC-Komponenten	381
Der parallele Port	381
Die serielle Schnittstelle	381
Der USB-Port	382
FireWire	383
Local Area Network (LAN) und Wide Area Network (Internet).	383
Feldbusse (CAN, VAN u.a.)	384
Der GPIB-Bus	384
VXI / PXI / SCXI etc.	384
SCSI	385
PS/2	385
26.4 Interne Bussysteme.	385
ISA-Bus	386
EISA-Bus	386
Microchannel-Bus	386
VESA-Bus	387
PCI	387
PCI-Express	387
AGP-Port	388
PCMCIA – PC-Card – CardBus	388
I2C-Bus / SM-Bus	388
PC104	389
CompactPCI	389
27. Die Steuerung externer PC-Ports	390
27.1 Die Suche nach I/O-Ports	390
Das BIOS-System	391

Mit dem Debugger herumschnüffeln	392	Subnetz-Masken	441
27.2 Hardwarezugang	394	32.3 Ordnung halten im Netzwerk.	442
28. Ein genauer Blick auf den Drucker-Port.	395	Address Resolution Protokoll.	442
28.1 Funktionsdiagramm	395	Protokoll zur dynamischen Host-Konfiguration	442
28.2 Beschreibung der Registerebenen.	396	Automatische IP.	442
28.3 Einfache Operationen	397	32.4 Sockets und Ports für den Datentransport	442
28.4 Schnittstellen für Bit Banging	398	Telnet.	443
Die Steuerung einer Leitung	399	ICMP	443
Emulation eines seriellen Protokolls	400	TFTP	443
28.5 Druckerportsteuerung mit ClassWork	402	HTTP	443
28.6 Besondere Modi für den Druckerport	403	SMTP	444
Bi-direktionaler Parallel-Port	403	32.5 Xport-Device.	444
Der Standard IEEE 1284.	403	32.6 Ein Beispielsystem mit dem Xport	445
28.7 Erweiterter Parallel-Port	405	33. GPIB	446
Kommunikation über das EPP-Protokoll	405	33.1 Die Struktur des GPIB-Busses	446
Schnittstellensteuerung mit einem EPP-System.	406	33.2 IEEE-488.2	446
Interface-Elektronik	408	Gemeinsamer Befehlssatz	447
28.8 Der ECP-Port	409	33.3 SCPI.	447
29. Details zum seriellen Port.	410	34. Vision.	449
29.1 Systembeschreibung	410	34.1 GPIBcore	450
Handshake und Flusssteuerung	411	Die Eigenschaften von GPIBcore	451
Flusssteuerung per Software	411	GPIBcore installieren	451
Welche Flusssteuerung soll man verwenden?	412	34.2 Die GPIBcore-Programmieranleitung	451
29.2 Der Schnittstellenanschluss	412	GPIB Funktionen	451
Die Elektronik hinter dem Anschluss	414	GPIBinit.	451
29.3 Der UART	415	GPIBbye	452
29.4 RS232 und andere serielle Standards	416	GPIBopen.	452
RS232	416	GPIBclose.	453
RS422	416	GPIBtimeout	453
RS485	416	GPIBdefer.	453
Current Loop	416	GPIBsinglestep	454
29.5 Tipps und Tricks	417	GPIBtroff	455
29.6 Einfache serielle Operationen mit MSCOMM	417	GPIBtron	455
30. Details zum ISA-Bus	418	GPIBwrite	455
30.1 Der ISA-Bus	418	GPIBread	456
Interface-Chips	419	GPIBfind	456
Einfache Interfaceschaltungen mit dem 8255		GPIBFindall	457
I/O-Controller	421	GPIBIdentifyAll	457
Einfache Interfaceschaltungen mit klassischer Logik	422	34.3 Die I/O-Funktionen von GPIBcore	457
30.2 Eine Adresse für unsere Karte.	423	OUT	458
Zugriff auf die Karte.	424	INP	458
30.3 PC104	425	34.4 Support-Funktionen für GPIBcore	459
31. USB	427	DELAY	459
31.1 Details zum USB-Port	427	Microdelay	459
USB-Nachrichten	428	SStr\$	459
USB-Transfer	429	Bin\$	460
Transferarten	430	vVal	460
31.2 USB aus Sicht der Hardware	430	Logentry	461
Hardware bauen	432	Log10	461
31.3 Umsetzer von USB auf RS232	432	34.5 ClassWork.	461
31.4 Ein USB/GPIO-Umsetzer mit dem FTDI245	434	Das ClassWork-Konzept.	461
An den Ausgang schreiben	436	Die ClassWork-Lösung	462
31.5 USB/GPIO-Umsetzer mit dem FTDI2232	437	Programmierung mit ClassWork	464
32. Ethernet	439	Ein ClassWork-Programm als Beispiel	465
32.1 Die Verkabelung eines Netzwerks.	439	Die Entwicklung von ClassWork-Modulen	466
Direkte Kabel	439	Modul-Header	466
Cross-Over-Kabel.	440	Interne ClassWork-Variablen	466
32.2 Konfiguration eines Netzwerks.	440	Events initialisieren und beenden.	467
Die IP-Adresse	440	Adresszuordnung	467
		AssignTo Zuordnung	468
		Überblick über den globalen Einführungscode	468

34.6 Regeln für die Entwicklung von ClassWork-Modulen	469	37. Der Aufbau von Anwenderschnittstellen	484
Eigenschaften	469	37.1 Entwurf von Begrüßungsbildschirm, Logo und Icon	484
Methoden (Sub)	469	37.2 Die Hauptform	485
Methoden (Funktionen)	470	Die Arbeitsfläche des Programms	485
Sonderfälle	470	Der Entwurf eines Menüs	486
Der Treiber HP34401 unter ClassWork	471	Tooltips	487
34.7 TestBench	472	Tooleisten	487
35. Der Entwurf von Testprogrammen	474	37.3 Objekte und Steuerungen anordnen	487
35.1 Sauberer Code	474	37.4 Tool-Formen für die Konfiguration	488
Modulares Programmieren	474	37.5 Hilfe-Dateien	488
Code-Dokumentierung	475	38. Weitere Fallstudien	489
Ablaufmarkierungen und CamelWriting	475	38.1 Fallstudie 11: SPI-Stack auf dem Druckerport	489
35.2 Zugriff auf Instrumente und Hardware	475	38.2 Fallstudie 12: Datenexport in eine Datei	493
Zugriff auf Instrumente	476	38.3 Eine CSV-Datei erzeugen	495
Hardwarezugriff im Computer	476	38.4 Fallstudie 13: Ein U/I-Plotter unter Verwendung der GPIBcore-Operationen	496
35.3 Datensammlung gegen Analyse	476	38.5 Fall 13: Ein U/I-Plotter mit ClassWork-Operationen	497
35.4 Log-Dateien erstellen	476	38.6 Fall 13: Ein U/I-Plotter mit TestBench-Operationen	498
35.5 Die Anatomie eines gut strukturierten Testprogramms	477	39. Schlussgedanken	500
36. Besondere Programmieretechniken	479	Downloads zum Buch	500
36.1 Stream Interpreting	479	Stichwortverzeichnis	501
Ein monolithisches Programm	479		
Ein modulares Programm	481		
Einen Stream erzeugen	482		