

STEFAN HÖLTGEN

>OPEN HISTORY_

Archäologie des Retrocomputings

KULTURVERLAG KADMOS BERLIN

Disclaimer: Alle in diesem Buch dargestellten Programme und elektronischen Schaltungen sind geprüft. Verlag und Autor können dennoch keine Gewähr auf die Funktionalität geben und weisen darauf hin, dass die Implementierung auf eigene Verantwortung des Lesers geschieht.

Bibliografische Information der Deutschen Nationalbibliothek

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

Das Werk einschließlich aller seiner Teile ist urheberrechtlich geschützt. Jede Verwertung ist ohne Zustimmung des Verlages unzulässig. Das gilt insbesondere fürervielfältigungen, Übersetzungen, Mikroverfilmungen und die Einspeicherung und Verarbeitung in elektronischen Systemen.

Copyright © 2022

Kulturverlag Kadmos Berlin. Wolfram Burckhardt

Alle Rechte vorbehalten

Internet: www.kulturverlag-kadmos.de

Gestaltung und Satz: kaleidogramm, Berlin

Umschlagmotiv: Sebastian Bach

Korrektorat: Miriam M. Höltgen

Druck: Opolgraf

Printed in EU

ISBN 978-3-86599-475-2

INHALT

VORWORT	9
1. EINLEITUNG	13
1.1 Fragestellung	13
1.2 Gegenstand	15
1.3 Methoden-Überblick	17
1.4 Aufbau	21
2. FORSCHUNGSSTAND	25
2.1 Formen des Geschichtsbezugs	25
2.2 Preservation und Emulation	33
2.3 Szenen und Protagonisten	36
3. COMPUTERARCHÄOLOGIE	41
3.1 Methoden der Computer-Geschichtsschreibung	42
3.1.1 Zeit-, Sozial- und Mentalitätsgeschichte: <i>Hackers</i>	42
3.1.2 Unternehmer-Geschichte: <i>The Homecomputer Wars</i>	45
3.1.3 Computergeschichte als Wirtschaftsgeschichte: <i>A History of Modern Computing</i>	47
3.1.4 Interaktive Software-Geschichte(n)	51
3.2 Geschichtskritik	53
3.2.1 Die Poetologie der Computergeschichte(n)	54
3.2.2 Foucaults Archäologie der Diskurse	57
3.2.3 Medienarchäologie des Non-Diskursiven	60
3.3 Computerarchäologie als Methode	67
3.3.1 Informatikgeschichte und Retrocomputing	68
3.3.2 Archäologie der (Computer-)Gegenwart	71
3.3.3 Der (sogenannte) Computer	73
3.3.4 Fixing (the history of) E.T.	82
3.4 Zusammenfassung	85

4.	RETROCOMPUTING ALS ARCHÄOGRAPHIE	87
4.1	Demonstration, Simulation und Codierung.	88
4.1.1	Analogien	92
4.1.2	Metaphern	104
4.1.3	Mimikrys	108
4.1.4	Computerphilologie I: Menschen lesen Code	129
4.1.5	Computerphilologie II: Maschinen lesen Code	155
4.1.6	Zusammenfassung.	170
4.2	Computerspiele und Simulationen	175
4.2.1	Simulation.	175
4.2.2	Games of Life and Death	183
4.2.3	<i>Game of Memories</i>	190
4.2.4	Pixelspiele	198
4.2.5	Unconventional Computing	202
4.2.6	Toy Computing.	205
4.2.7	Zusammenfassung.	214
4.3	Software Preservation und Emulation.	216
4.3.1	Emulation	219
4.3.2	Retrogame Development.	227
4.3.3	Das VC4000MultiROM	238
4.3.4	Epistemologische Aspekte der Emulation	242
4.3.5	Retrofitting	252
4.3.6	Zusammenfassung.	256
4.4	Hardware und Knowledge Preservation	257
4.4.1	Computer als historische Artefakte.	259
4.4.2	Computer als Hardware	260
4.4.3	Werkstattbericht	264
4.4.4	Reparieren als Spiel.	276
4.4.5	Historische Computer als epistemische Dinge.	283
4.4.6	Der Computer als Diagramm	288
4.4.7	Zusammenfassung.	289
5.	RETROCOMPUTING ALS WISSENSPRAXIS	291
5.1	Die Archäologie des Wissen(wollen)s	291
5.2	Computer-Autodidaktik	295
5.2.1	Autodidaktik.	295
5.2.2	Selbstlernkompetenz	297
5.2.3	Methoden des Selbstlernens	299

5.3	Homecomputing	309
5.3.1	Das Handbuch des Schneider CPC6128.....	312
5.3.2	C16-BASIC-Kurs.....	314
5.3.3	Spielend Programmieren lernen.....	316
5.3.4	Mein Heimcomputer selbstgebaut.....	317
5.4	Retrocomputing	319
5.4.1	Das Computerspiel Rock	320
5.4.2	Das Betriebssystem SymbOS.....	321
5.4.3	Der Emulator JavaCPC	322
5.5	Zusammenfassung.....	323
6.	SCHLUSS.....	325
6.1	Zusammenfassung.....	325
6.2	Möglichkeiten und Grenzen	329
6.2.1	Interdisziplinarität.....	329
6.2.2	Retro-Didaktik	330
6.2.3	Reichweite.....	331
6.2.4	Beschreibungs(in)kompetenz	331
6.2.5	Software/Hardware Preservation.....	332
6.3	Ausblick	333
	BIBLIOGRAFIE	337
	ABBILDUNGSVERZEICHNIS	363
	ANHANG	
A.	Programmlisting »16KRA Long Memory Test Programm«	369
B.	Interview mit Marius Groth	374
C.	Sourcecode des Programms Game of Memories	378
D.	Interview mit Dr. Martin Wendt	388
E.	Interview mit James Jacobs.....	398
F.	Linkverzeichnis	401
G.	Register.....	406

1. EINLEITUNG

»Die Geschichte des Computers ist zu wichtig,
um sie allein den Historikern zu überlassen.«
[Raúl Rojas u.a. 2004: 229]

1.1 Fragestellung

Die Geschichte der Computer besitzt eine Eigentümlichkeit, die ihren Gegenstand von dem anderer historischer Gegenstände deutlich abhebt. Computer sind Medien, die aufgrund ihrer Operationen *speichern, übertragen und prozessieren* in der Lage sind alle anderen Medien zu simulieren. Um diese Eigenschaft zu erlangen, müssen Computer sich aber *im Vollzug* befinden, das heißt *operativ sein*. Erst im Verbund von prozessierender Hardware, prozessbeschreibenden Algorithmen und prozessierten Daten gelangen sie in den Medienstatus. Das bedeutet im Umkehrschluss, dass ein nicht-operativer Computer, sei er nur ausgeschaltet oder defekt, lediglich den Status von Hardware besitzt. Die Geschichte der Computer ist bislang notwendigerweise entweder als Hardware-Geschichte geschrieben worden (und zeigt sich in Technik- und Computermuseen als Ausstellung nicht-operativer Hardware) oder als Geschichte der um die Hardware, die Software und deren Nutzung etablierten Diskurse, Ingenieure, Ökonomien und Nutzer. Eine Geschichte der Computer (als operative Medien) steht bislang aus. Wie kann aber etwas, das sich im Prozess befinden muss, historisiert werden?

Die vorliegende Studie versucht diese Frage zu beantworten und das *Archiv der Computer* in Vollzug zu setzen, denn es scheint, als seien die Methoden akademischer Geschichtsschreibung kaum anwendbar auf einen Gegenstand, der doch lediglich *in der Gegenwart* existieren kann. Dies zeichnet Computer nämlich wie alle anderen technischen Medien aus: Dadurch, dass sie erst im Vollzug zu dem werden, was sie sind (Medien), befinden sie sich als solche stets in der Gegenwart. Das

hat auch Konsequenzen für ihren Status als historische Artefakte: Es ist beispielsweise unerheblich für das Medium Radio, wann es gebaut wurde; schaltet man es heute ein, dann hört man heutige Radioprogramme (nicht etwa die, die zurzeit der Entstehung des Apparates ausgestrahlt wurden). Ein operatives technisches Medium ist immer im Hier und Jetzt. Es scheint eng an das Reale gekoppelt zu sein: Es basiert auf Hardware, die Verschaltungen für Signalwege darstellt, um aus latenten Eingangs- evidente (sinnfällige) Ausgangssignale zu generieren. Die dabei ablaufenden Prozesse finden jenseits aller Symbole *im Realen* statt – das ist der Hauptgrund dafür, dass so verstandene Medien nicht (durch Symbole) historisierbar und nicht allein aus ihrer »diachronen Perspektive« [Hellige 2004: 4] verstehbar sind. Begriffe wie ›Geschichte‹, ›Historie‹, ›historisch‹, ›früh‹ und ähnliche sollen damit in der vorliegenden Arbeit kritisch reflektiert werden.³

Computer unterscheiden sich dadurch von anderen technischen Medien, dass sie das Symbolische der Schrift wieder integrieren – durch die Symbole ihrer Programmierung von der Ebene des Mikrocodes bis zu den höheren Programmiersprachen. Damit rufen sie neben den Hardware-Geschichten auch Schrift-Geschichten und -Theorien auf, die Diskursfelder von der diachronen und synchronen Sprachwissenschaft bis hin zur Theorie der formalen Sprachen und ihren Automaten öffnen. Diese Hybridität zwischen ›reiner‹ Hardware, formalisierter Logik, algorithmisch beschreibbarer Software und Daten erfordert besondere Methoden, die über den Methoden-Kanon von Elektrotechnik, Linguistik, formaler Philosophie, Informatik und Medienwissenschaft hinausgehen. Eine Geschichte des operativen Computers müsste sich daher *zwischen diesen Disziplinen* entwickeln. Damit würde sich ein solches Projekt erstmals auch die Möglichkeit einer *informatischen Computergeschichte* ermöglichen, die etwas anderes als die *Geschichte der Informatik* (als Disziplin) oder die *Geschichte des Computers* (als Technologie) sein könnte. Der methodische Ansatz, der dafür vorgestellt wird, erfordert jedoch sowohl eine scharfe Eingrenzung des untersuchten Gegenstands in Hinblick auf seinen Entstehungszeitraum als auch eine beispielhafte Auswahl der Einzelanalysen.

³ Dort, wo solche Begriffe zur Zuschreibung einer ›zeitlichen Situierung‹ verwendet werden, sollte dies in einem unemphatischen Sinne – etwa als »discontinued«/»nicht mehr lieferbar« – verstanden werden. In Retrocomputing-Szenen wird hierfür zuweilen das TM-Symbol zur ironischen Markierung des Vergangenheitsbezugs verwendet (»damalsTM«, vgl. [oh1]), bei Friedrich Kittler findet sich die markante Distanzierungsgeste des »sogenannten«.

1.2 Gegenstand

Unter der Zuschreibung »früher Mikrocomputer« werden Mikrocomputer (also Digitalcomputer auf Basis einer Ein-Chip-CPU) mit Datenbus-Breiten von 4, 8 und 16 Bit gefasst, die ab Mitte der 1970er-Jahre frei verkäuflich für Amateur-Anwender publiziert wurden. Dabei handelt es sich um Plattformen (zum Begriff der *Plattform* vgl. [Bogost/Montfort 2009a; Höltgen 2013a] sowie Kapitel 2.1 und 3.3.3), die notwendige Peripheriebestandteile (Eingabetechnologien, den Mikrocomputer selbst sowie Schnittstellen für Ausgabeperipherie) in einem Gehäuse vereinen. Ab 1975 verfügen diese Computer über Schnittstellen und Protokolle, die es ermöglichen, bereits in Privathaushalten vorhandene Medientechnologien wie Fernseher, Kassettenrecorder, Stereoanlagen, elektrische Schreibmaschinen und anderes direkt an sie anzuschließen. In diesem Fall soll von *Homecomputern* (oder Heimcomputern) in Abgrenzung zu *Personalcomputern* gesprochen werden, für deren Betrieb dedizierte Ein- und Ausgabe-Peripheriebausteine notwendig sind.

Ab 1977 entwickelt sich eine Industrie, die Homecomputer herstellt und sich dabei bis circa Mitte der 1980er-Jahre immer stärker diversifiziert [vgl. Ceruzzi 2003a: 311 ff.] Insbesondere der Wechsel von 8- auf 16-Bit-Architekturen, das Wiedererstarken der Spielkonsolen-Industrie und die stärkere Verbreitung günstiger IBM-kompatibler Personalcomputer auch für Amateur-Anwendungen (insbesondere für Computerspiele) hat ab Mitte der 1980er-Jahre die Vielfalt an Herstellern und Gerätetypen stark dezimiert, so dass zum Ende des von mir betrachteten Zeitraums nur noch wenige Hersteller mit Homecomputer-Modellen am Markt verblieben sind: Apple, Commodore, Atari und Acorn. Letztere drei veröffentlichten mit den Modellen Acorn Archimedes A3010, Commodore Amiga 1200 und Atari Falcon 030 1992 die letzten Homecomputer, die Tastatur, Computer und Diskettenlaufwerk in einem Gehäuse vereinten und Anschlüsse für Atari-kompatible Joysticks verfügt. Zu dieser Zeit erscheinen neue Modelle der Konkurrenz bereits als Desktop-Computer mit abgesetzter Tastatur und Monitoranschluss. 1993 stellte Atari und 1994 Commodore die Herstellung dieser Rechnermodelle ein.⁴ Acorn entwickelt die Prozessortechnik (ARM) sowie – nach verschiedenen Ausgründungen und Umfirmierungen [vgl. Mohr 2014] – das Betriebssystem RISC OS weiter. Schließlich konsolidiert sich der Markt zwischen

⁴ Die genauen historischen Entwicklungen stellen sich komplexer als hier skizziert dar [vgl. Bagnall 2010; Hertzfeld 2004].

Apple (mit seinen Macintosh-Computern) und IBM-kompatiblen Personalcomputern (zumeist mit *MS-DOS*- und *Windows*-Betriebssystemen). Die Homecomputer-Ära ist damit zu Ende.

Die hier vorzunehmende Eingrenzung der zu betrachtenden Plattformen auf die Jahre 1974–1984 hat noch einen zweiten Grund. Es sind insbesondere die in diesem Zeitraum erschienenen Computer, mit denen sich das *Retrocomputing* heute beschäftigt. Für historische Homecomputer existiert heute ein lebendiger Markt mit gebrauchten Modellen (die teilweise hohe Sammler-Werte produzieren, [vgl. Brückner 2011]), sowie für alte und neue Hardware und Software, neue Zeitschriften und Bücher sowie zahlreiche Online-Plattformen (Blogs, Diskussionsforen, Facebook-Seiten und -Gruppen, Archive für Software und Dokumente). Seit einigen Jahren finden sich Nutzergruppen bestimmter Homecomputer-Plattformen oder zu Themen wie Computerspielen in Vereinen und locker organisierten Stammtischen zusammen, veranstalten Ausstellungen, Messen und Festivals, auf denen Privatsammlungen, neue Produkte (insbesondere neue Software und Hardware-Erweiterungen) vorgestellt werden, Workshops, Seminare und Vorträge stattfinden. Allein in Deutschland ereignen sich monatlich bis zu zehn Retrocomputing-Veranstaltungen [vgl. N.N. 2015a].

Bemerkenswert ist hierbei die starke Konzentration auf die Softwaregattung *Computerspiel*. Nicht nur, weil diese für die Retrocomputing-Gruppen und -Enthusiasten und innerhalb der Software Preservation die zentralen Gegenstände sind, nehmen Computerspiele in der vorliegenden Untersuchung eine wichtige Rolle ein; an diesen Programmen lassen sich überdies besonders viele unterschiedliche technische Phänomene kondensieren. Computerspiele stellen immer schon einen ›Motor‹ der Entwicklung sowohl privater Computerhard- und -software [Levy 1984: 37–57, 277 ff.] als auch computertheoretischer Diskurse (wie sich am Schachspiel zeigt [Turing 1987a; Shannon 1950]) dar. Sie sind die »Drosophila der Computerentwicklung«, wie Alexander Reinefeld [2006: 252] schreibt.

Die Eingrenzung verfolgt jedoch auch forschungspragmatische Gründe: Eine Theorie mittlerer Reichweite, wie sie im Folgenden formuliert wird, verlangt einen überschaubaren Gegenstandsbereich, der in ausreichender Detailtiefe wissenschaftlich analysiert werden kann. Dieser soll weder das Bild einer historischen ›Epoche‹ formen, noch sollen damit (ohnehin nicht mögliche) Eingrenzungen und Ausgrenzungen erreicht werden. Dasjenige, was Homecomputer als solche auszeichnet, hat es in Teilaspekten sowohl vor als auch nach dem hier eingegrenzten Zeit-

raum in der Computertechnik und -kultur gegeben. Die Homecomputer zeigen sich so, trotz meiner Eingrenzung auf die vergleichsweise kurze Existenzphase innerhalb der sogenannten ›Geschichte des Computers‹, als Gegenstand mit vielfältiger Ausprägung und langer Nachwirkung. Im Folgenden sollen überblicksartig die Methoden und ihre Herkünfte skizziert werden, mit denen dieser Gegenstand untersucht werden soll. (Im weiteren Verlauf der Studie wird detaillierter auf die einzelnen Methoden eingegangen.)

1.3 Methoden-Überblick

Eine Untersuchung der frühen Mikrocomputer und ihrer Programmierung unter den eingangs genannten Einschränkungen und Erweiterungen wird einen sorgsam abgestimmten Methoden-Kanon erfordern. Dieser muss sich einerseits an geschichtswissenschaftlichen, didaktischen, informatischen und medienwissenschaftlichen Methoden orientieren und andererseits auf einzelne Diskurse und Arbeitsfelder der jeweiligen Disziplinen zurückgreifen.

Zunächst wäre – nicht nur in Abgrenzung zu deren tradierten und für den Gegenstand als defizitär ausgewiesenen Methoden – die *Technik-geschichtsschreibung* selbst zu fokussieren: Nach welcher Methodologie verfährt sie, insbesondere, wenn ›der Computer‹ ihr Gegenstand ist? Welche *Historeme*⁵ fasst sie dabei vor allem ins Auge (Technik, Ökonomie, Personen, Praktiken, ...)? Die Quellen der Computergeschichtsschreibung sind vielfältig. Neben akademischen Studien in Monographien und Zeitschriften (etwa den *Annals of the History of Computing*) existieren zahlreiche populärwissenschaftliche und semi-professionelle Publikationen. Die Erlebnisberichte von Zeitzeugen und Mitwirkenden der Computerindustrie des frühen Mikrocomputers finden sich in zahlreichen Quellen, welche der *Oral History* zugerechnet werden müssen und damit eigene Interpretationen [vgl. Stöckle 1990; Vorländer 1990] verlangen. Zudem zeigen sich sowohl zeitgenössische technische Manuals und Handbücher (sogenannte *graue Literatur*) als auch Lehrbücher, Zeitschriftenpublikationen, Dokumentar- und Spielfilme und andere Medien als informative Quellen für die Geschichte früher Mikrocomputer.

⁵ Zum Begriff *Historem* als diskursivierter historischer Fakt [vgl. Neumann 1999]. Im Folgenden wird der Begriff im Sinne Whites verwendet: als ein Element einer »gegebenen Menge von zufällig überlieferten Ereignissen [, ... die von Historikern] zu einer Geschichte gemacht« [White 1991a: 104] werden.

Markant an sowohl der zeitgenössischen Nutzung der frühen Mikrocomputer (Homecomputing) als auch an ihren heutigen Verwendungsweisen im Retrocomputing ist eine *spezifische Didaktik*, die sich als *Autodidaktik* darstellen lässt. Neben den vielfältigen und abwechslungsreichen Ansätzen zur curriculären Computerdidaktik entsteht erstmals mit der Privatisierung von Computertechnik in den 1970er-Jahren ein Zugang zu dieser Technologie, der freiwillige Lektüre (Lehrbücher, Handbücher, Zeitschriften) voraussetzt und allenfalls von Szene-internen Diskursen begleitet und geleitet wird. Die Emergenz der Homecomputer scheint eine Form des *Willens zum Wissen* in den – zumeist kindlichen und jugendlichen! – Nutzern provoziert zu haben, die eng mit ihren damaligen wie heutigen Nutzungsweisen verwoben ist. Neben der Tatsache, dass die Lehliteratur für Homecomputernutzer eine historisch reichhaltige Quelle darstellt, lässt sich aus ihr – im Kontrast zu schul- und universitätsdidaktischen Publikationen – der Modus Operandi der *Hackerkultur* [vgl. Levy 1984: 26–30] extrahieren. Ihr zugleich »kalter Blick« [vgl. Ernst 2005/6] und »respektloser« Zugriff auf Technik bildet auch die Basis für heutige Praktiken im Umgang mit »historischer Hardware«.

Zur adäquaten technischen Beschreibung eines Computers (seiner Operation, Programmierung und nicht zuletzt seiner Didaktik und seinen gesellschaftlichen Wechselwirkungen) stellt die *Informatik* etablierte Methoden bereit. Hiervon können für die vorliegende Untersuchung vor allem methodologische Aspekte der *Theoretischen Informatik* (etwa über den Aufbau formaler Sprachen und zur Automatentheorie), der *Technischen Informatik* (zur Beschreibung und Analyse logischer Gatter, von Mikroprozessoren, Computerarchitekturen und Protokollen), der *Praktischen Informatik* (in Hinblick auf Betriebssysteme, Compiler und Interpreter, Programmiersprachen und Algorithmen-Entwicklung sowie des Software-Engineering) und nicht zuletzt des Gebiets *Informatik und Gesellschaft* (in dem Diskurse zur Software-Archivierung, Informatikgeschichte, Kultur und Didaktik der Informatik, zur Emulation sowie der Geschichte und Entwicklung von Mensch-Computer-Interaktionen stattfinden) genutzt werden. Für eine Studie im Fachgebiet der Informatik müssen diese Perspektiven privilegiert eingenommen werden. Aus diesem Grund finden sich im Verlauf der Arbeit zur Argumentation meiner Thesen sowohl technische Schaltpläne als auch Programmcode. Dort, wo auf informatische Spezialdiskurse zurückgegriffen wird, findet eine Definition von Fachtermini nur cursorisch und mit Verweis auf die einschlägige Fachliteratur statt.

Schließlich liefert die *Medienwissenschaft* in der Ausprägung der sogenannten ›Berliner Schule‹ die dann noch fehlenden methodischen Bausteine für die vorliegende Studie. Die eingangs bereits skizzierte Sicht der Medienwissenschaft (im Singular) auf *Medien als Medienapparate im Vollzug* und nicht etwa als Produzenten bestimmter Ästhetiken, Verständnis- und Nutzungsweisen, Wirkungen, Märkte und anderer Felder (mit denen sich die Medienwissenschaften – im Plural – beschäftigen) hat unter Friedrich Kittler und Wolfgang Ernst eine fruchtbare Methodik hervorgebracht: die *Medienarchäologie*. Sie beschäftigt sich, knapp gesagt, mit dem *medientechnischen Apriori des Wissens* und geht auf eine Lektüre des französischen Philosophen Michel Foucault zurück. Wie dieser untersucht die Medienwissenschaft die Bedingungen der Möglichkeit des Wissens. War es bei Foucault aber vor allem die (institutionelle, diskursive, biologische, ...) *Macht*, welche die Bedingungen des Wissens darstellte, sind es nach Kittler und Ernst vorrangig *Medientechnologien*, mit denen Information (und damit Wissen) gespeichert, übertragen und verarbeitet [Ernst 2012b; Kittler 1993c: 64] werden.

Der in dieser Arbeit verwendete Begriff *Archäologie* unterscheidet sich von dem der akademischen Facharchäologie [vgl. Schneider 2004: 93–96], indem er etymologisch (wieder) näher an die ursprüngliche Bedeutung des Wortes heranrückt: »Für den neuzeitlichen Begriff von Archäologie ist es kennzeichnend, grabend zu suchen. Genau dies aber kennzeichnet nicht den klassisch-griechischen Gebrauch des Wortes; in der gleichnamigen Schrift des Dionysius von Halikarnaß meint *archaiologia* schlicht die Redaktion, das Bearbeiten alter Nachrichten.« [Ernst 2004: 253 f.] Archäologie in diesem Sinne ist zugleich Theorie und Methode: Das *theoretische* Wesen dieser Redaktionstätigkeit in Bezug auf die Historiographie wird im Kapitel 3 erläutert; das manipulative *Hands-on-Imperativ*, mit dem sich der Computerarchäologe *methodisch* den Artefakten annähert, wird an den Beispielen des Kapitels 4 exerziert. Wo Facharchäologie das (stets historische) Artefakt von den Sedimenten der Vergangenheit freilegt, um es derartig herauspräpariert der Beschreibung der Historiker und der Bewahrung der Museen zu überantworten, legt die Computerarchäologie das medientechnische Artefakt (das nicht notwendig ein vergangenes sein muss [vgl. Ebeling 2006: 12]) von den es überlagernden Diskursschichten der Historiographie frei, indem es sich durch seine Oberflächen und Schichten ›gräbt‹ bis hinab auf die Ebene von Hardware und Software. Dort manipuliert der Medienarchäologe die Funde, um ihnen verborgenes Wissen zu entlocken; die so verstandene Archäologie ist also keine Konservierungspraxis. (Zeitweise kann

diese Form der redaktionellen Archäologie mit facharchäologischen »Grabungspraktiken« zusammenfallen, wie etwa bei der Freilegung der LSI-Strukturen historischer integrierter Schaltkreise [vgl. Swaminathan 2011], um Informationen verloren gegangener Entwicklerdokumente durch Prozesse von Reverse Engineering zu ersetzen [vgl. oh2]⁶.)

Die Analysen dieser Archäologie verfahren im Wesentlichen *techno-mathematisch* (und nicht *diskursanalytisch* wie bei Foucault) und sind bereits von daher zum Methodenkanon der Informatik »kompatibel«. Zudem hat den »Computer als Medium« [vgl. Bolz/Kittler/Tholen 1994; Warnke/Coy/Tholen 1997 u. a.] die Berliner Medienwissenschaft bereits sehr früh als Forschungsgegenstand entdeckt und betreibt nach wie vor Grundlagenforschung zu dem, was sich auf Arbeitsfeldern wie den »Digital Humanities« [Jones 2014] oder der »Kulturtechnikforschung« [Holl 2015] ereignet.

So eng die Zusammenhänge zwischen dieser Medienwissenschaft und den Gegenständen des Gebiets Informatik und Gesellschaft in der Vergangenheit gewesen sind, hat sich in den vergangenen Jahren jedoch eine merkliche Trennung beider Forschungsdiskurse ereignet (die zuletzt in der Entscheidung der GI-Arbeitsgruppe »Computer und Gesellschaft« die langjährige, stark medienwissenschaftlich geprägte Konferenz-Reihe »Hyperkult« zu beenden und sich aufzulösen, kulminierte). Wo die akademische Informatik beständig (auch als Folge des Bologna-Prozesses) anwendungsbezogene(re) Blicke auf den Computer eingenommen hat, hat sich die Medienwissenschaft des Computers zuletzt zu einer Art »Kulturwissenschaft des Computers« [vgl. Holl 2015] gewandelt, die nun nicht mehr die technischen Medien als solche, sondern deren Kultur, Geschichte und Ökonomie untersucht und lehrt. Dort werden Computer beispielsweise vollständig auf »Dienste« [Pias 2015: 33], also (je nach Verständnis des Begriffs) auf Angebote für Anwender oder einzelne technische Funktionen reduziert. Es ist nicht zuletzt ein Anliegen dieser Arbeit, neue Möglichkeiten der Interaktion zwischen Informatik und Medienwissenschaft aufzuzeigen und so Fäden eines ehemals fruchtbaren interdisziplinären Diskurses wieder aufzunehmen.

⁶ Dort werden unterschiedliche historische Schaltkreise mechanisch und chemisch geöffnet, um ihre Strukturen zu analysieren und darauf basierende Softwaresimulationen zu entwickeln.

1.4 Aufbau

Die vorliegende Studie verfährt im wesentlichen deduktiv. Im zweiten Kapitel findet die Diskussion des Forschungsstandes statt. Dabei werden Beiträge ab Mitte der 1960er-Jahre bis in die Gegenwart mit Blick darauf vorgestellt, in welche Beziehung sie Technik zur ihrer Geschichte setzen und dabei spezifische Begriffe in den Forschungsdiskurs einschreiben, welche Auseinandersetzungen es zur Bewahrung historischer Hardware und Software gegeben hat und gibt und welche Szenen und Protagonisten sich mit der Computergeschichte und dem Retrocomputing befassen und wie sie dies tun. Es wurde hierfür versucht einen Großteil der Forschungsbeiträge zu sichten und die relevant(est)en Beiträge daraus überblicksartig vorzustellen. Einige an dieser Stelle nicht berücksichtigte Texte werden im weiteren Verlauf der Studie diskutiert.

Im dritten Kapitel wird der theoretisch-methodische Ansatz der *Computerarchäologie* vorgestellt. Hierzu müssen (auf Grundlage eines Beispiels diskursarchäologischer Praxis) die Möglichkeiten und Probleme aktueller Computergeschichtsschreibung benannt werden: Mit welchen Methoden und durch welche Quellen werden Computer als historische Artefakte dargestellt? Zur Beantwortung dieser Frage wird die methodologische Auseinandersetzung zwischen Ereignis- und Strukturgeschichte, wie sie der Technikhistoriker Wolfgang König [König 2009: 40 f.] dargestellt hat, erörtert. Ausgewählte, besonders diskursmächtige⁷ computerhistorische Monographien sollen hernach vor dem Hintergrund ihres methodischen Ansatzes, der von ihnen verwendeten Quellen, ihrer Argumentationsstrategien und ihres thematischen Fokus' analysiert werden. Die Ergebnisse dieser Lektüre werden mit dem geschichtskritischen Modell der *Archäologie des Wissens* Michel Foucaults [vgl. Foucault 1981] konfrontiert, die im Anschluss eine Erweiterung durch die Theorien der Medienarchäologie und schließlich der Formulierung der Methoden einer Computerarchäologie erfährt. Es soll sich zeigen, dass die Praxis des Retrocomputings unter diesem Aspekt auch als *operative Geschichtskritik* verstanden werden kann – insbesondere dann, wenn sie sich den durch die publizierte Geschichtsschreibung marginalisierten Objekten (Plattformen, Programmiersprachen, Betriebssystemen, Programmen, Literaturen u. a.) zuwendet.

Das vierte Kapitel diskutiert vier Retrocomputing-Projekte detailliert, die zwischen 2012 und 2017 im Rahmen der universitären Lehre

⁷ Grundlage für diese Zuschreibung stellen die Auflagenzahlen dar.

und Forschung durchgeführt wurden. Diese sind auf den Hauptbetätigungsfeldern der Szenen – Demo-Coding, Computerspielentwicklung, Software Preservation und Emulation sowie Hardware Preservation – angesiedelt. Einige dieser Projekte fanden im Rahmen universitärer Lehrveranstaltungen statt, die sich damit neben der Anwendung einer »Retro-Didaktik« [Höltgen 2020a] auch zugleich einer computerhistorischen Wissensvermittlung als »knowledge preservation« [Agrifoglio 2015: 15–20] widmeten. In Kapitel 4.1 wird die Demoprogrammierung anhand eines physikalischen Beispiels (Ball sprung-Simulation) auf unterschiedlichen Plattformen (Analogrechner, Spielkonsole, Homecomputer) und in unterschiedlichen Programmiersprachen/-systemen vorgestellt. Mithilfe eines hierfür entwickelten computerphilologischen Methodensets werden diese Implementierungen miteinander verglichen und zueinander in eine »rezeptionsgeschichtliche« Beziehung gesetzt. Kapitel 4.2 diskutiert die Entwicklung eines Simulationsspiels, dessen epistemologische Deutungsmöglichkeiten sowie die grundsätzliche Beziehung zwischen Spiel(en) und Computern. In Kapitel 4.3 wird ein Projekt zur Software Preservation durchgeführt: die Entwicklung eines modernen Speichersystems für eine historische Computerspielplattform. Dieses wird kontrastiert mit der zentralen Bewahrungsstrategie für Software, der Emulation. Die Frage, welche Möglichkeiten, Grenzen aber auch welche Mehrwerte ein Emulator mit sich bringt, bildet hierbei den argumentativen Fluchtpunkt. Das letzte Projekt in Kapitel 4.4 befasst sich mit der Reparatur eines historischen Computers. Die Tools, die hierfür zum Einsatz kommen, entstammen einer autodidaktischen Beschäftigung mit historischer Hardware und rekrutieren Methoden des Retro-Hackings, die sich deutlich von konservatorischen Praktiken der Hardware Preservation unterscheiden. Der Wissenserwerb, der – neben einem wieder funktionsfähigen Computer – aus diesem Projekt resultiert, wird als dialektischer Prozess dargestellt, der sich in wechselnder Perspektive auf Werkzeuge als Hilfsmittel und Objekte des Wissens, ergibt.

Das fünfte Kapitel wertet die vier zuvor diskutierten Projekte aus, indem hier die darin zur Anwendung gekommenen Wissens-Erwerbsprozesse in eine Didaktik überführt werden. Diese befasst sich mit den Methoden, Praktiken und Quellen der Aneignung autodidaktischen Wissens, welches eine Bedingung der Möglichkeit von Retrocomputing darstellt. Den schulischen und universitären Konzepten der Informatik-Ausbildung werden seit dem ersten Erscheinen der Homecomputer in der zweiten Hälfte der 1970er-Jahre Materialien und Methoden autodidaktischer Informatik-Selbstausbildung zur Seite gestellt. Letztere bilden die Wissens-

basis für die private ›Aneignung‹ der Computertechnologie, womit alle Formen der Verstehbarmachung, Modifikation und des ›Missbrauchs‹ [vgl. Kittler 1986: 149; Pias 2015: 32–34] von Computerhardware durch Hacker gemeint sind. Den Kulminationspunkt bildet hier ein Überblick über Formen der informatischen Selbstausbildung. Auf diese Weise soll ein Verständnis der *Computernutzung als Spiel* evoziert werden, womit nicht das Spielen von Computerspielen, sondern das Programmieren, die Hardware-Entwicklung und die produktive Auseinandersetzung mit der publizierten Computergeschichtsschreibung (etwa in Form von Rezensionen, Internet-Kommentaren, Zeitschriftenartikel und anderem mehr) *mit spielerischen Vorgehensweisen* gemeint ist. *Gamification*, so verstanden, meint also den Wissenserwerb als Spiel – bei dem der zu beherrschende Computer zugleich als Spielmaschine, Gegenspieler und als Gegenstand des Spiels in Erscheinung tritt.

Im Schlusskapitel werden die Untersuchungen noch einmal zusammengefasst und ihre Ergebnisse kritisch reflektiert. Zwei Anliegen sollen hier als Ausblick vorgetragen werden: Zum einen wird das zugrunde liegende Forschungsprojekt in seinen interdisziplinären Konsequenzen dargestellt. Zum anderen werden – in Form eines Ausblicks – Anschlussmöglichkeiten für Forschung und Praxis (hier vor allem zu Programmiersprachen und Computerspielen) vorgeschlagen.

G. Register

1. Namen und Firmen

- Acorn: 15
Alcorn, Alan: 90
AMD: 27 [FN]
Apple: 15f.
Aristoteles: 41, 62
Atari: 15, 46, 82, 90, 103, 109f., 118, 166
Babbage, Charles: 34, 41, 289 [FN]
Baer, Ralph: 90, 103f.
Baudrillard, Jean: 26
Boole, George: 62, 71, 192, 329, 331
Brookhaven National Laboratory: 90
Bushnell, Nolan: 90
Ceruzzi, Paul: 15, 41, 44, 47–51, 53f., 74
Church, Alonzo: 203, 248
Collingwood, Robin G.: 80f., 85, 169, 325
Commodore: 15, 46, 108–110, 314
Conway, John H.: 53, 181f., 187, 195, 378
Cotes, Roger: 135
Dionysius von Halikarnaß: 19
Douglas, Alexander: 199
Erhard, Ludwig: 207
Exidy: 77f.
Felsenstein, Lee: 44, 262, 265
Foucault, Michel: 10, 19–21, 31, 57–61, 65, 72, 85, 135, 291–294, 311, 323, 325
Gardener, Martin: 131, 181
Genette, Gerard: 139–141, 148
Gosper, William: 182
Grimm, Jacob: 295
Grimm, Wilhelm: 295
Hall, Chris: 144
Haraway, Donna: 31
Heidegger, Martin: 284f., 288, 304, 328
Hewlett-Packard: 49f.
Homebrew Computer Club: 50
Hopper, Grace: 140, 170, 266 [FN]
IBM: 15f., 49f., 223f.
Intel: 49, 262
Jameson, Fredic: 26
Jauß, Hans Robert: 133f.
Kelvin, William Thomson (Lord): 203
Kemeny, John G.: 140, 170
Kilburn, Tom: 199
Kittler, Friedrich: 14, 19, 59 [FN], 73, 130f., 216, 240, 292–294
Kurtz, Thomas E.: 140, 170f., 301, 312
Levy, Steven: 43–47, 50, 53f., 309
Lincoln, Abraham: 295
LMI: 44
Lovelace, Ada: 41, 62
Luck, Dale: 108, 114, 116, 137f.
Lyotard, Jean-François: 89f.
McLuhan, Marshall: 25f.
Merton, Robert K.: 79
Mical, Robert J.: 108, 114, 116, 137f.
Microsoft: 50, 82, 271
Miner, Jay: 109f.
MIT: 44, 90, 103, 182, 309f.
MOS Technology Inc.: 124

- National Semiconductor: 252f.
Peirce, Charles Sanders: 227, 331
Plateau, Joseph: 203
Processor Technology: 273, 276
Radio Electronics: 49
Rheinberger, Hans-Jörg: 243, 285f., 328
Riemann, Bernhard: 203
Rousseau, Jean-Jacques: 285
Schickard, Wilhelm: 41, 62
Shannon, Claude: 41, 62, 71, 208
Sierra On-Line: 44
Signetics: 105, 317, 398f.
Sinclair: 109, 324
Solomon, Les: 262
Spectravideo: 52
Stallman, Richard: 44
Symbolics: 44
Texas Instruments: 249
Tomczyk, Michael S.: 46, 54
Tramiel, Jack: 46
Turing, Alan M.: 34, 70f., 73, 170, 180, 186, 203, 207, 209, 223, 248, 329
Turkle, Sherry: 296f., 304, 309f.
Twain, Marc: 48, 51, 54
Ulam, Stanley: 180
Völz, Horst: 11, 77f.
von Neumann, John: 71, 180, 186, 248, 258, 288, 329
White, Hayden: 17 [FN], 54–57, 62, 65 [FN], 80, 85, 90, 325
Wilkes, Maurice: 62
Williams, Fredric C.: 199
Wolfram, Stephen: 180f.
Zilog: 63
Zuse, Konrad: 12, 41, 62, 69f., 180, 259, 289 [FN]

2. Plattformen

- Acorn: Archimedes A3010: 15; Archimedes A3020: 52; BBC Micro: 27, 307
- Altair 8800: 49f., 105, 223
- Amstrad: CPC: 91, 109, 111, 118, 121f., 136, 165, 322; CPC464: 119; CPC664:119f., 136; CPC6128: 119f., 123 [FN], 136, 144, 312f., 316
- AN/FSQ-7: 90, 199
- Apple: II: 27, 31, 50, 52; Macintosh: 224
- Arduino: 201f., 219
- Atari: Falcon 030: 15; Lynx: 228f., 388, 392, 394; VCS: 27, 74–76, 83f., 91, 109f., 124–128, 136, 141f., 166f., 228, 249 [FN], 388–391, 394f.; ST: 118 [FN], 224, 228; XL: 375, 388
- BBC Micro Bit: 201f.
- Calliope: 201f.
- CBM PET 2001: 46, 49, 52
- Colossus: 223
- Commodore: C128: 33, 224, 254; C16: 314f.; C116: 314; Plus 4: 314; C64; VC-20/VIC-20: 46, 54; Amiga 27, 46, 91, 108–119, 122–124, 126f., 136, 138, 142, 161f., 224, 230, 250; Amiga 500: 111; Amiga 1000: 112 [FN]; 1200: 15
- DEC: PDP-10: 50, 223; PDP-11: 256
- Difference Engine No. 2: 34, 289 [FN]
- EDSAC: 199f.
- ENIAC: 77
- Exidy Sorcerer: 77f.
- Ferranti Pegasus: 34
- Harvard Mark I: 77
- HP-65: 49
- IBM: 704: 90, 97, 104f., 107, 159f., 198; 70XX: 223; /360: 223; PC-Kompatibel: 15f., 224, 254
- Intellec-4: 50
- Interton VC-4000: 218f., 228, 232, 234–238, 240, 242, 256f.
- Jupiter Ace: 307
- KC-85: 77
- Leybold-Heraeus-Analogcomputer: 103
- Lorrain: 108 [FN], 109, 162
- Magnavox Odyssey: 90, 103f.
- Manchester Mark I: 199
- Mark-8: 49f.
- Micral: 50
- MSX: 28
- Nintendo: Entertainment System (NES): 27, 167 [FN]; Super Entertainment System (SNES): 27; GameBoy: 213; GameBoy Advance: 228, 389
- Oric: 375
- PLATO: 27
- Raspberry Pi: 202
- Scelbi-8H: 50
- Schneider CPC s. Amstrad
- SDS-940: 43
- Signetics Instructor 50: 91, 105, 108, 136, 160, 160f., 186–198, 244, 327, 378
- Sinclair: QL: 109, 118 [FN]; ZX Spectrum: 118, 224, 228, 254, 375, 388f., 393
- Sirius 1: 253f.
- Sol-20: 261–266, 278, 283f., 288f., 327, 369–373
- Telefunken: RA-742: 91, 93, 95, 97, 236; RAT-700: 93 [FN]; RA-741: 93 [FN]; RA-770: 93 [FN]; RA-800-HYBRID: 93 [FN]
- Texas Instruments: TI-99/4 / TI99/4a: 375
- Tandy/Radio Shack TRS-80: 49, 74, 398; Model I Level I: 76f.; Color Computer: 118
- TX-0: 43
- Vectrex: 103
- Victor 9000: 253f.
- Watara Supervision: 228, 388, 390, 392f.
- Zuse Z3: 77, 289 [FN]
- ZX Pentagon: 374

3. Schlagworte

- 16 Bit: 15, 109, 118, 127, 136, 164, 223, 227, 254, 280, 318, 388, 391
 1940er: 199; 1941: 82, 223; 1948: 199, 205
 1950er: 27, 34, 43, 69, 295, 309; 1950: 90, 208; 1951: 208; 1952: 180, 199; 1955: 207; 1958: 44, 90, 93; 1959: 34
 1960er: 21, 25, 33, 57, 104, 133, 138, 181, 205, 211, 223, 262; 1964: 25; 1966: 209, 210
 1970er: 15, 18, 22, 43, 50, 90, 116, 130, 158, 182, 198, 252, 262, 294, 299, 306, 318, 328, 333; 1971: 63, 234; 1972: 44, 49, 51, 90, 103, 162, 289 [FN]; 1973: 28, 44; 1974: 16, 49, 181, 252, 335; 1975: 15, 49, 105, 223; 1976: 46, 261f.; 1977: 15, 27, 51f., 74, 124, 252, 253, 369; 1978: 218
 1980er 15, 43, 45, 63, 82, 130, 223, 235, 262, 271, 320, 333; 1980: 44, 46; 1983: 27, 44, 82, 299; 1984: 16, 43, 46, 49, 77, 91, 108f., 138, 224, 309, 319; 1985: 27, 48, 63, 71, 91, 109, 138, 162, 180, 224, 289 [FN]; 1987: 224; 1989: 34, 224
 1990er: 11, 116, 228; 1991: 289 [FN]; 1992: 27, 49; 1993: 15, 28, 68, 73
 32 Bit: 224, 322
 8 Bit: 44, 50, 105, 110, 118, 122, 124, 160, 189, 192, 202, 223f., 228, 231, 244, 252, 254, 262, 268, 320f., 330, 333f., 375, 389–391, 396
 Abakus: 61, 64
 Absturz: 191, 225, 294 [FN], 301
 Abtippen: s. Type-in-Listing
 Abtippfehler: 145
 accuracy: 27, 221, 223f., 400
 Adaption: 50, 92, 109f., 113, 116–118, 126f., 132, 166, 173, 225, 227f., 231, 388, 392
 Adapter: 110, 173 [FN], 238f., 267f., 270 [FN]
 Addieren: 61, 127, 387
 Addierwerk: 64
 Adressarithmetik: 152, 190, 194, 279, 379, 380–383, 385
 Adressierungsart: 152f., 187f.
 Adressbus: 124
 ahistorisch: 325
 Algorithmus: 70, 114, 127, 129, 134f., 142, 147, 180, 182f., 185, 187, 204, 209, 238, 279, 300; A.theorie 69; Bresenham-A. 121, 135f.; Dijkstra-A. 329; Standard-A.: 137
 Alphabetismus: 131; Ana.: 131
 alphanumerisch: 43, 110, 132
 ALU: 213
Amiga Boing: 108–115; 117–119; 122, 124, 127f., 135–138, 141f., 154, 161f., 166
Amiga-Ball: s. *Amiga Boing*
 AMSDOS: 119, 312
 Anachronismus: 71, 244
 analog: 36, 104
 Analogrechner/Analogcomputer: 22, 62, 68, 74, 91, 93 [FN], 95–99, 100, 103, 135f., 153, 157f., 171, 175f., 180 [FN], 203, 259, 332
 Anekdote: 47, 71, 207, 309
 Animation: 27, 34, 113, 117–119; 121f., 126–128, 136, 234
 Anspielung: 139f., 209
 ANTIC: 110
 Apfelmännchen: s. Fraktal
 APL (Programmiersprache): 308
 Approximation: 114f., 160
 Apriori: 19, 37, 61, 85, 131, 311, 325, 335
 Arcade: 33, 104, 245
 Archäographie: 66, 69f., 85, 87, 89, 91, 147, 326–328
 Archäologie: 10f., 19–21, 29, 58–61, 65–68, 71, 82f., 85, 128, 141, 156, 200, 249, 251, 291f.; Facha.: 19f., 29, 59, 71
 Architektur: 15, 75, 87, 180, 198, 223f., 232, 252, 26'54, 258, 283, 288, 295, 319, 322, 326, 330; Computera.: 18, 186, 203, 215, 242, 302; Prozessora.: 27 [FN], 33, 187, 188 [FN], 242; von-Neumann-A.: 72, 186, 247, 250, 257, 283, 329, 332

- Archiv: 9, 13, 16, 32, 35, 59–61, 66, 68–70, 134, 146, 156, 170, 173f., 216, 234, 238, 251, 276, 292
- Archivierung: 10, 59, 146, 276; Software-A.: 18, 170, 238
- Artificial Intelligence (AI): s. künstliche Intelligenz (KI)
- ASCII-Zeichen: 185, 374
- Assembler (Programmiersprache): 52, 105, 109, 116–118, 129, 136f., 142, 149–153, 155, 162–164, 186–188, 192, 198, 218f., 225, 228, 233, 247, 273, 277f., 293f., 310, 320f., 334, 375, 378, 385, 389, 391
- Assemblierer: 141 [FN], 142 [FN], 187, 233, 247, 278f., 320–322, 398
- Ästhetik: 19, 25f., 28, 30, 32, 37f., 72, 76, 102, 126, 133, 189, 227, 230, 232, 261, 325
- Auflösung: 35, 98, 107, 111f., 114, 122f.
- Aufschreibesystem: 9
- Authentizität: 31, 36, 39, 174, 220, 226f., 243, 245, 250, 289, 320, 323, 327, 333
- Autodidaktik: 18, 22, 31, 39, 43, 45, 90, 145, 205, 218f., 257, 260, 287, 290f., 295–309, 311, 314, 316f., 320, 323, 328, 334
- Automat (Informatik): 14, 66, 72, 179; A.entheorie: 18, 69, 178, 186, 215; zellulärer A. 175, 178–183, 186, 188f., 191, 200, 202–206, 212, 214f., 327, 331; Zustandsa.: 179, 211
- Autor: 54–56, 74, 109, 122, 128, 133, 135–138, 140, 144, 147f.
- AY-3-8912: 119
- Backus-Naur-Form: 132
- Ball im Kasten*: 91, 93–97, 100, 108, 110, 114, 129, 135f., 149, 153–155, 157f.
- Bank Switching: 119
- bare metal: 141f.
- Barfußhistoriker: 39, 54
- Barock: 62, 135, 293
- BASIC (Programmiersprache): 27, 50, 52, 118f., 121–124, 129, 136, 140, 144–147, 150, 153, 155, 162–167, 170f., 247, 271, 278, 301, 305, 307f., 310, 312, 314–318, 330 [FN], 334; BBC-B.: 308; Microsoft-B.: 273; Altair-B.: 223; Chipmunk-B.: 308; TrueB.: 184, 187; Locomotive-B.: 109, 119, 136, 143, 150–153, 164, 312
- Batch Processing: 43
- Berechenbarkeit: 69, 203, 215, 238
- Betriebssystem: 15f., 18, 21, 33, 44, 40f., 71, 88, 109, 116, 118f., 121, 123, 131, 137 [FN], 141, 143–145, 147, 173 [FN], 201, 220, 228, 244, 254, 264f., 277, 293f., 309, 312, 321, 330
- Bewahrung: s. Preservation; B.sstrategie: 22, 30f., 35, 217, 250, 334
- Bibliothek: 141f., 389
- Big Data: 169, 181
- BiggerBoing26*: 124
- Billard-Kugel-Computer: 204, 331
- Binary-Datei: 115–117, 233 [FN], 245
- Binärzahl: 164, 189, 206
- Biologie: 19, 178, 186, 202; B.smus: 186
- BIOS: 77, 307, 317, 398, 400
- Bit: B. Board: 212, 281; B.-Maske: 121; B.map: 116, 136f.; b. plane: 112–114
- Black Box/Blackboxing: 36, 189f., 245, 258, 286, 288, 322, 326, 331
- Blended Learning: 218, 328
- Blitter: 110, 117–119, 122f., 135f., 141, 143–145, 154f., 164–166; B.-Objekt: 113
- Brute Force: 223, 302
- Buch: 16, 45, 53, 60, 139, 156, 305, 334; Lehrb.: 17f., 147, 161, 167, 187, 306, 311f., 314f., 317f., 320, 323, 328; Handb.: 9, 17f., 66, 77, 93 [FN], 95, 137, 142, 144, 165, 171, 187, 211, 213, 270, 276–278, 282f., 307, 312–314, 316, 323, 394
- Byte Magazine*: 183
- C (Programmiersprache): 109, 113–117, 129, 138, 141–143, 150, 152f., 155, 162–164, 189 [FN], 190, 219, 225, 237, 334, 389; Lattice C: 114, 136, 138; Visual C: 398
- C++ (Programmiersprache): 225
- CA65: 142 [FN]
- Caren and the Tangled Tentacles*: 229 [FN], 389f., 394, 396f.

- Cartridge: s. ROM-Modul
 Casual Game: 29, 230
 Chemie: 79, 256
 Chomsky-Hierarchie: 132
 Chronologie: 44f., 47f., 51, 62, 134, 169
 Chronik: 29, 54f., 134
 Chronos: 9, 72
 Circuit Bending: 274
 COBOL (Programmiersprache): 140, 170, 256
 Codierung: 9, 63, 88f., 98, 117, 130, 133, 157, 169, 176, 187, 190, 193, 212, 220, 225, 238, 247, 281, 301f., 332
 COMAL (Programmiersprache): 278
 Compiler 18, 69f., 117, 138, 142, 155, 163f., 247f., 305, 307f.; C.bau: 69
 Computer: C.club: 311; C.kultur: 41, 261f., 310f., 334; C.spiel: 15f., 22f., 26–32, 35f., 42, 44, 51f., 76, 82f., 87–90, 93, 96, 104, 175, 194, 199f., 212f., 215, 217, 227, 229, 238, 245, 259, 288, 303, 306, 311f., 316f., 320, 327, 332, 334, 388f., 394; C.zeitschrift: 16–18, 23, 38 [FN], 49, 63, 119, 145, 174, 183, 261f., 318, 334
 Controller (Eingabegerät): 15, 75, 84, 110, 125, 196, 232, 234, 244, 321
 Copper: 110, 137
 CORDIC-Algorithmus: 114
 CPM: 50, 119, 312
 CPU s. Mikroprozessor
 Cracker: 37
 Critical Code Studies: 29
 Cross Platform Development: 83, 109, 196, 240, 256, 320
 Crossassembler: 196
 CRTC: 119–121, 123, 144
 CTIA: 110
 Cultural Studies: 28, 38, 76
 Daten: 13f., 61, 65, 79, 97, 115, 124, 128, 130, 155, 160, 162f., 192, 198, 217, 239 [FN], 277–279, 283, 311 [FN], 332, 391; D.bus: 16, 74, 105, 136, 222, 224, 226, 243, 254; D.träger: 36, 156, 173, 217, 226, 239f., 247, 333, 288; D.typ: 192
 DDR: 11, 245
 Debugger: 79, 84, 128, 156, 165f., 168, 233
 Debugging: 83f., 165, 172, 228, 302f., 306, 389–391
 Decompiler: 117 [FN]
 Deixis: 148, 152–154
 Deklaration: 140, 165, 178, 194
 Demo Coding: 22, 38, 87–89
 Demonstration: 65 [FN], 72, 75, 80f., 88–91, 93, 96f., 110, 134, 136, 155, 157f., 160f., 172–174, 258, 288, 326, 331
 Demoscene: s. Szene
 DENISE: 110
 Determinismus: 56, 177, 238, 332
 Dezimalübertrag: 61, 64
 diachron: 14, 89, 129, 131–134, 141, 147, 172, 242, 249, 256, 326, 332
 Diagramm: 66, 69, 100, 132, 153, 179, 288f., 317, 321; Flussd.: 66, 72; Datenflussd.: 69; Baumd.: 69
 Diagrammatik: 85, 101, 156, 229, 258, 288, 326, 331, 334
 Didaktik: 9, 17f., 22, 37, 53, 69f., 85, 87, 116, 121, 159, 165f., 170f., 186f., 202, 204, 206f., 211, 215, 251, 255, 257, 276, 283, 289f., 294f., 298–300, 303, 306, 308f., 314, 316f., 323, 327f., 330
 Digital Humanities: 20, 130f., 170, 330
 Digitalcomputer: 15, 41, 54, 62, 71, 74, 93, 97f., 175f., 180, 186, 191, 198, 203, 212, 248
 Digitalisierung: 216, 271
 Diode: Zener-D.: 98 [FN], 102, 154, 158; Leuchtd.: 105, 107, 159f., 189, 191, 198, 268
 Direct Memory Access (DMA): 254
 Disassembler: 79, 84, 116f., 165, 398
 disassemblieren: 84, 109, 113f., 115f., 137f., 143f., 149, 155, 161f., 237, 243, 398, 400
 Diskette: 10, 53, 119, 156, 238f.; D.laufwerk: 15, 110, 119, 238f., 264
 Diskontinuität: 57, 61
 Diskurs: 9, 13f., 16–21, 25, 29, 36–37, 42, 45, 49, 51, 53, 57–62, 64–67, 71–73, 79–82, 87–89, 113, 119, 130,

- 132, 135, 138, 161, 175, 180, 186, 200, 203f., 215, 217, 223, 240, 245, 248, 255, 261, 289, 292, 294, 323, 325–328; D.analyse: 10, 20, 38, 59f., 64, 71, 287, 326; D.archäologie: 21, 31, 36, 54, 57, 60f., 311; D.macht: 21, 134, 180
- Dispositiv: 9, 11, 28, 57, 59–61, 65, 71, 294, 323, 325
- Dissimulation: s. Skeuomorphismus
- Doom*: 26
- Downgrade: 227
- Dr. NIM*: 207 [FN], 209–215
- Drucker: 264, 270
- Dualzahl s. Binärzahl
- E-Learning: 218, 276, 283, 289, 299f., 306–309, 323, 328
- E.T. – The Extraterrestrial*: 82–84, 303f.
- Echtzeit: 38, 84, 93, 100, 125f., 144, 157f., 165, 172, 196, 198, 244, 248, 389, 391
- Editor: 70, 198, 244, 322; Texte.: 113 [FN], 164, 228, 233, 289
- Ego-Shooter: 26
- Eigenzeit: 38, 80, 93, 130, 156
- Einplatinencomputer: 198, 202, 253
- Elektronenröhre: 34, 199
- Elektronik: 10, 14, 34, 60, 66, 72, 74, 76, 79, 97, 100–103, 119, 131 [FN], 132, 157, 193, 202f., 207, 232, 240, 242f., 247–249, 253, 256–262, 267f., 270, 275, 284, 295, 302, 306, 317, 322, 326, 329, 275; Mikroe.: 215, 219; Rande.: 193
- Elektromechanik: 101
- Elektrophysik: 65f.
- Elektrotechnik: s. Elektronik
- Emergenz: 18, 61, 75, 79, 245, 261
- Emulator/Emulation 11, 18, 22, 26, 30–36, 38, 52f., 70, 81, 83f., 93, 103, 122, 128, 144, 161, 165f., 168, 173–175, 177, 188f., 192 [FN], 195–200, 213f., 216–238, 242–252, 256f., 266, 271, 274, 308, 320, 322f., 327, 332f., 378, 391–395, 398–400
- Enigma: 223
- Episteme: 87, 329; e.isches Objekt/Ding: 64, 243, 256f., 283, 285, 287f., 299;
- E. Spielzeug: 304
- Epistemologie: 22, 28, 32, 36, 38, 48, 63, 68, 70, 88, 92, 110, 117, 132, 174, 176, 178, 186, 216f., 220, 226, 242f., 259, 257f., 283f., 286, 289, 291–293, 301f., 307, 328–330
- Epoche: 9, 16, 31, 43f., 58, 109, 131, 133, 215, 242, 256, 259
- EPROM: 216, 228, 232–234, 238, 392–394
- Ereignisgeschichte: 51
- Erweiterung: 16f., 21, 74, 77, 83, 109, 153, 196 [FN], 198, 214, 217, 226, 234, 239, 244, 254, 262–264, 271, 273, 276f., 283, 295, 321f., 394
- Erzählung/Narration: 28, 42, 46–48, 51, 54, 56, 62, 66, 69, 139
- Eschatologie: 56, 62
- Europa: 49, 64, 264
- Evokation: 23, 26, 91, 112, 245 [FN], 296f.
- Evolution: 58, 132–134, 169, 186
- Exklusiv-Oder (XOR): 145, 192, 206, 386
- Experiment: 10, 70, 80f., 84, 91, 136, 158, 160f., 171 [FN], 174f., 178, 186, 188, 191, 198, 201–204, 211–213, 216, 218, 264, 282f., 285–287, 299, 301–305, 210, 328, 331, 400; e.elle Archäologie: 81, 287;
- Farbregister: 84, 110, 112f., 390
- Farbtiefe: 111f.
- Fehler/Error/Glitch: 45, 82f., 138, 145, 161f., 166, 168, 170, 172, 196, 225f., 228, 230, 234–237, 243f., 247, 249–251, 256, 265–267, 270–273, 276, 280–282, 297, 302f., 305, 308, 210, 327f., 333, 370, 373, 375, 394, 397, 399f.; F.quelle: 265, 267
- Fernseher: 15, 26, 61, 262; F.bild: 60
- Festplatte: 10, 156, 321f.
- Festverdrahtung: 62f.
- Film: 26, 93, 172f.; Spielf.: 17, 26; Dokumentarf.: 82
- Filter: 51, 55, 62, 65, 222
- Firmware: 141, 144, 214
- First Draft of a Report on the EDVAC*: 186, 258, 288

- Flappy Bird*: 219, 227f., 230–237, 240 [FN], 256
Flash-ROM: 218, 228, 234, 238, 240, 257
Fließkomma/FPU: 63, 116, 121, 136, 162, 165
Flip-Flop: 199, 209, 214, 236
FLOW-MATIC (Programmiersprache): 140, 170
formale Sprache: 14, 18, 69, 87, 132, 149, 170, 220, 247
Formalwissenschaft: 10, 85, 326
Format: 9, 25, 34f., 224 [FN], 238, 239 [FN], 247, 254, 269, 276, 370f.
Formel: 66, 69
Forth (Programmiersprache): 307, 334
Fortschritt: 30, 48, 57f.; F.sgeschichte: 25, 258, 325
FPGA: 216
Frequenz: 60, 96, 232, 391, 395; F.teiler: 10, 391; F.analysator: 79;
Funktionserweiterung: 226, 244
Game Engine: 26
Game of Death: 188f., 191
Game of Life: 53, 177–179, 181–187, 189–192, 195, 200, 202–205, 212f., 215, 227, 280, 327, 378
Game of Memories: 179 [FN], 188–195, 197f., 200, 216, 244 [FN], 378–387
Game Studies: 29, 31, 334
Gameplay: 28, 230f.
Gamification: 23, 32, 37, 118, 205, 258, 289, 299, 303–305, 307, 309, 316, 323, 328, 334
Gate Array/ASIC/ULA: 322
Geisteswissenschaften: 10, 25, 77, 243
Genre: 45, 48, 56, 66, 88, 134, 172, 396
Geschichtskritik: 21, 61, 65, 88
Geschichtswissenschaft: 333
Geschwindigkeit: 95, 98, 167 [FN], 171, 222, 250, 304, 327; Recheng.: 34, 155f., 161, 176
Gestalt: 96, 200, 332
Gleitkomma s. Fließkomma
Glossierung: 137
GNU: 44
GOTO: 120, 151f., 154, 162–164, 184f.
Grafik: 82, 102f., 103, 110, 112, 119, 122, 126, 136, 138, 158, 167, 236, 394, 396; Schwarz-weiß-G. 200; G.modus 121, 145; G.-RAM/Speicher 110f., 121, 123. 166, 201f.; G.adventure: 396
Graue Literatur s. Literatur
GTIA: 110
Hacker: 11, 18, 23, 36f., 42–45, 50, 43, 90f., 124, 141, 182, 205, 229, 262, 299, 309–311, 321f., 335
Halbaddierer: 64, 186
Halbleiter: 50, 240, 249, 252
Hands-on: 12, 19, 259, 261, 276, 286, 299, 309
Heimcomputer s. Homecomputer
Hermeneutik: 29, 42, 71, 116, 173, 295, 300, 329
Heuristik: 97, 287
Historiographie: 9–11, 19, 30f., 38, 41, 44–46, 49, 53f., 57–62, 65, 67, 71, 85, 88–90, 134, 157f., 161, 217, 249, 259, 288, 325, 327, 333
Hobbyist: 9–11, 29, 31f., 36, 39, 82, 87–89, 91, 117, 124, 215–219, 224f., 227, 229, 242, 251, 256f., 260, 262, 274, 276, 288, 295, 298f., 302, 304, 308, 311, 321–323, 326, 328, 333f., 398
Homebrew: 228, 240, 392
Homecomputer: 15–18, 22, 27, 33, 42, 44–47, 49f., 52f., 64, 71, 74f., 108–111, 116, 137 [FN], 143, 182, 198, 219, 224, 227–230, 239, 262, 276, 293, 303, 307–319, 320, 323, 327, 330, 334, 388
HTML: 35, 255, 322
Human Computer Interaction (HCI): 132, 303, 332
Humanwissenschaften: 56
hybrid: 14, 32, 103, 121, 178, 283, 301, 320; H.rechner: 74, 93 [FN]
I/O: 26, 34, 105, 110, 124, 141, 201 [FN], 202, 220, 262, 277, 399, 400
Ideengeschichte: 41, 58
Idiosynkrasie: 34, 53, 77, 166, 174f., 196, 238, 245 [FN], 247f., 327
Ikon: 100, 129, 132, 153, 161, 332

- Image (Daten): 156, 188 [FN], 189 [FN], 226, 238f., 247, 249, 392
- Implementierung: 22, 27f., 36, 41, 50, 62f., 69–71, 75, 89, 90–109, 113f., 118, 121f., 124, 128f., 134, 136, 140f., 143, 153, 156, 163f., 171, 174, 180–187, 191f., 196, 198, 203–209, 211 [FN], 212–216, 220–230, 232f., 235f., 242, 251, 254, 270, 308, 326f., 330f., 397, 398, 400
- implizites Wissen: 136, 297, 300
- Index: 89, 160; i.ikalisch: 129, 132, 153f., 227
- Informatik: Theoretische I.: 18, 71, 73, 132, 223, 329; Technische I.: 18, 79, 242, 293, 329; Praktische I.: 18, 242, 293, 302, 312, 329; Angewandte I.: 33, 175, 177, 183, 220, 252, 254f.; I. und Gesellschaft: 9, 18, 20, 68; I.didaktik: 69, 171, 215, 257, 294, 299, 303, 330; I.system: 74, 255; I.lehre: 298f., 323; I.unterricht: 207, 298f., 309, 319
- Informationstheorie: 69, 132
- Ingenieurwissenschaften: 10, 66
- IDE (Integrierted Development Environment): 83, 228, 308, 321, 389, 398
- Integrierter Schaltkreis (IC): 20, 233, 272, 277, 281f., 393
- Intel 8080: 49, 105, 105, 223, 262; I. 8088: 254; I. 8086: 254; I. x86: 188 [FN], 224; I. 80286: 292; I. Atom: 196
- Interdisziplinarität: 11f., 20, 23, 257, 326, 329, 333
- Interface: s. Schnittstelle
- Internet: 23, 50f., 72, 126, 174, 211, 229, 238, 239, 244, 255, 266, 268, 271, 274–276, 307f., 320, 375
- Interpreter: 18, 71, 84, 115, 118, 119, 121, 135, 143, 150, 153, 155, 164f., 169, 174, 213, 248, 271, 278, 305, 307f., 321
- Interrupt: 107, 144, 167, 232, 312, 332, 391
- Intertextualität: 129, 132, 138f., 141, 143, 147
- ISA-Ebene: 109, 242, 292
- Java (Programmiersprache): 225, 322
- JavaCPC*: 122, 144, 166, 222 [FN], 225, 322, 333
- JavaScript (Programmiersprache): 225, 322, 333
- Joystick: s. Controller
- Kairos: 10, 72
- Kanon: 30f., 35, 42, 110, 134, 135, 284, 293
- Kassette: 238f., 270f., 274–277, 314; Mikro.: 239; K.nrecorder: 15, 119, 262, 264, 271, 274f.
- Kathodenstrahl: 96, 98, 100–103, 125, 135, 155, 157, 167, 199; K.röhre: 96, 199
- Kausalität: 59, 176
- Kill the Bit*: 105f.
- Kippschalter/Binärschalter: 105, 189f.
- Koeffizientenpotentiometer: 98, 102
- Kohärenz: 47, 55, 148, 152
- Kohäsion: 130, 132, 145, 147–155, 159; K.marker: 145, 148f., 154; K.arten: 149, 152
- Kommentar: 105f., 136f., 139
- Kommunikationstheorie: 69
- Komparatistik: 132
- Kompetenz: 66, 71, 79, 131, 133, 171 [FN], 291, 294, 297, 308, 331, 376
- Komplexität: 28, 36, 58, 64, 77, 92, 108, 118, 141 [FN], 146, 165, 178, 187, 205f., 212, 213, 215, 218, 222, 247, 251, 253, 258, 292, 317, 327, 330, 334, 375, 389–391, 395; Schaltungsk. 96f., 330; Zeitk.: 159; K.sk. klasse: 179
- Kondensator: 76f., 100, 157, 199, 221
- Konnektor: 148–151, 153f., 159
- Konstruktivismus/konstruktivistisch: 31, 58, 65, 80, 325; K. Lerntheorie: 298f., 323, 328
- Kontinuität: 36, 57–59, 62, 222
- Kopierschutz: 29
- Korrektheit: 45, 92, 161, 270, 305
- Kreis: 95f., 98 [FN], 101, 113, 121f., 135f., 158, 166f.; K.funktion: 99, 101, 114; K.algorithmus: 121, 136
- Krieg: 26, 46, 68
- Kryptographie: 68
- Kulturtechnik: 20, 170
- Kulturtheorie: 2, 32

- Kunst: 26, 54, 56, 65 [FN], 132f., 310; Computerk.: 68, 74, 261; K.wissenschaft: 37, 132, 135f., 255
 Künstler: 10, 54, 133
 Künstliche Intelligenz (KI): 49, 51, 181, 207, 209, 212, 214, 300
 Kybernetik: 69, 172, 178, 186, 283, 300 [FN], 302, 332
La-Mulana: 28, 220, 227
 Langzeiterhaltung: 35
 Lautsprecher: 191, 391
 Learning by doing: 89, 299f., 323, 328
 LED/Leuchtdiode: s. Diode
 Lektüre: 114, 130, 137f., 147, 155–157, 159, 161, 164, 166, 171–174, 257
 Lerncomputer: 91, 105, 187, 196, 253, 317
 Library: s. Bibliothek
Life in Life: 184
 Lingua Universalis: 170
 Linguistik: 14, 130, 132, 139, 148f., 170, 312, 333, 138, 245, 247; C.linguistik: 130; Textl.: 139, 148
 Linux: 173 [FN], 391f.
 literacy: s. Kompetenz
 Literatur: 10, 18, 21, 42, 60, 82, 133f., 169, 187, 215, 242, 312, 317, 321; L.wissenschaft: 42, 130, 132–134, 138f., 171, 326, 332; L.theorie 138; Sekundärl.: 35, 134, 137, 143, 160, 165, 307, 314, 316, 334; graue L.: 17, 66, 134, 256, 334
 Lochband: 239
 Lochkarte: 43, 239
 Logik: 14, 62, 66, 150, 206f., 215, 316, 326, 329; Aussagenl.: 41, 62, 69; Schaltl.: 69, 135, 393; L.gatter: 63, 183, 206, 214, 258; L.analysator: 79, 156, 392
 Logo (Programmiersprache): 119, 126, 309f., 312
 Lookup Table: 114f., 126–128, 136
 Lötcolben: 274
 LSI: 20
 LUA (Programmiersprache): 308
machina arithmeticae dyadicae: 62, 64, 289 [FN]
 Macht: 19, 30f., 57, 85, 291–294, 311, 323, 325, 328
 Mächtigkeit: 212, 214
 Magnetkernspeicher: 201 [FN]
 Mainframe: 41, 43, 74, 104, 160, 199, 223
 MAME: 33
 Manual: s. Handbuch
 Maschine: m.lles Lernen: 170; m.nnah: 77, 121, 129, 163, 173, 295; M.sprache: 118, 121, 135, 142f., 153, 163–166, 188, 225, 247, 273, 277, 294; M.zeit: 98
 Mathematica (Programmiersprache): 181
 Mathematik: 9, 11, 41, 62, 64–68, 70, 72, 92, 98–100, 102, 110, 114, 129, 135f., 141, 150, 153, 160f., 170, 176, 178, 180f., 192, 206f., 211, 215, 220, 292, 302, 316, 326
 Maus: s. Controller
 Medien: M.archäologie: 10, 12, 19, 21, 29, 36, 54, 60f., 65, 67f., 80f., 85, 156, 240, 249, 286f., 292, 294, 325; M.epistemologie: 10, 64f., 85, 292, 328, 424; M.geschichte: 66f., 183; M.theater: 81, 172; M.theorie: 10–12, 87, 178; M.wissenschaft: 10–12, 14, 17, 19f., 25, 29, 42, 59f., 64, 70f., 77, 80f., 87f., 132, 171 [FN], 183, 216–218, 240, 242, 249, 260, 286, 303, 328–330, 333
 Medizin: 178, 186
 memory map: 116, 193, 400; M.ed I/O: 201 [FN]
 Mentalitätsgeschichte: 41–43, 58, 325
 messen: 10, 175, 169, 199, 317f.; M.technik: 79, 257, 326; M.daten: 66
 Metahistory/Metahistorie: 54, 56f.
 Metaphorik: 66, 104, 186, 318, 332
 Microcode: s. Mikroprogramm
 Mikrocontroller: 74
 middle level language: 129
 Migration: 35f., 173, 224
 Mikrocomputer: 16–18, 33, 35, 41, 43f., 50f., 53, 59, 63f., 70f., 74, 77, 82, 87, 198, 201f., 206, 211, 216, 223, 242, 252 257 260, 262, 291, 306f., 312 323, 333, 375
 Mikroprogramm: 14, 62f., 169, 188, 214

- Mikroprozessor: 15, 18, 33, 41, 44, 50, 63f., 77, 106, 109f., 117–119, 124f., 144, 152, 160f., 163, 166–169, 176 187f., 218, 223f., 244, 252–254, 262, 281, 292–294, 317, 320, 333, 390, 395, 400
 Militär: 42, 48f., 178
 Mimikry: s. Nachahmung
Minecraft: 207, 212–215, 227, 306
 Minicomputer: 41, 59, 74, 199
 Mischperspektive: 82f.
 Missbrauch: 23, 42, 90, 215, 292, 334, 376
 Mnemonic: 105f., 187, 188 [FN], 191, 278f., 379
 Modell: 72, 92, 176–178, 183, 216, 219
 Modifikation: 11, 23, 74, 77f., 91, 136f., 159, 211, 214, 252–254, 283, 299, 312, 322, 369
 Monitor (Peripherie): 15, 111f., 145, 266, 287, 276; Farbm.: 119, 245; LED-M.: 202; CRT-M. 199, 222, 399; Grünm.: 119; LCD-M.: 202, 376, 393; Plasmam.: 202
 Monitorprogramm: 165, 187, 192 [FN], 244, 270f., 273, 278, 307, 327, 400
 Moore'sches Gesetz: 203, 222
 Morphogenese: 180, 186
 MOS 6502: 124, 228, 320, 333, 389; M. 65C02: 391; M. 6507; M. 6510
 Motorola 68000: 109f., 117, 118 [FN], 136, 151f., 163
 MP3: 33, 244, 271, 275, 287
 MS-DOS: 16, 33, 50, 224, 254
 Multimedengerät: 79, 266, 269, 274
 Museum: 30, 33f., 200 [FN], 226, 245f., 259, 276, 289 [FN], 328, 333
Myst: 26
 Nachahmung: 36, 108, 118, 196, 216, 220, 222f., 318
 Nachbildung: 33f., 222, 243, 245 [FN]
 Nachvollzug: s. Re-enactment
 Naturwissenschaft(en): 9, 11, 56, 71, 73, 80f., 326, 331
 Nim (Spiel): 205–215
 NIMROD: 207f.
Nimwit: 208
 Nostalgie: 25f., 29–31, 240, 251, 323, 376
 Notationssystem: 65, 72
 Oberfläche: 19, 61, 64, 84, 90, 134, 155f., 226, 245, 282, 292, 294, 334, 289, 391
 Obfuskierung: 70, 116
 Obsoleszens: 27, 34, 223, 250, 252 284
 Ökonomie: 13, 19f., 36, 42, 44f., 47, 72, 82, 118, 170, 292, 325
 Opcode: 63f., 105f., 115, 127, 143, 145, 151, 155, 162 [FN], 169, 187f., 191, 273, 277–279, 318, 379; Pseudo-O.: 141f. [FN], 150f., 153; illegaler O.: 294 [FN], 394
 Open Source: 136
 Operationsverstärker: 93 [FN], 97, 100–103, 153, 157–159
 operativ: 10, 12, 13f., 23, 30–32, 39, 42 51, 53, 65f., 69–72, 77, 81, 85, 89, 130, 132 156f., 169, 226 251, 255f., 258–260, 287–289, 292, 323, 326, 331f., 334
 Oral History: 17, 43f., 74 [FN]
 Oszilloskop: 79, 95f., 100–104, 155–158
 OXO: 199f., 213
 Pädagogik: 37, 297, 299f., 323, 333
 Paddle: s. Controller
 page: 192–194, 198, 279, 282, 369–373
 Papier: 60, 68, 169, 181, 187, 189, 275, 289 [FN]; Metallschichtp.: 270, 275
 Paratext: 29, 129, 139–141, 145, 153, 211
 Pascal (Programmiersprache): 334
 PAULA: 110
 PC: s. Personalcomputer
 Performace: 93, 113, 117f., 122, 141, 224
 Performanz: 32, 174
 Peripheriebaustein: 15, 116
 PERL (Programmiersprache): 308
 Personalcomputer: 15f., 50, 52f., 84, 110, 116, 196, 219, 228, 245, 253f., 256, 270, 320f., 322, 388, 392f., 394
 Phänomenologie: 90, 110, 282
 Philologie: 22, 85, 89, 129, 132–134, 155f., 170; Computerp.: 22, 89, 129f., 132, 155–157, 172–174, 243, 245, 257, 326, 332; Editionsnp.: 131, 278; Medienp.: 131f., 156

- Philosophie: 14, 26, 61, 261; Geschichtsp.: 80f.
- Phorik: 152, 154; Anap.: 152; Katap.: 152
- PHP (Programmiersprache): 308
- Physik: 10, 22, 34, 61, 65f., 76, 79, 89, 91f.f., 96–98, 100, 107f., 114, 129, 134, 136, 158, 174, 175, 202, 213, 247f., 261, 326, 329, 332, 375; Spiel-P.: 102, 231f.
- Physiologie: 132
- PIO: 144
- Pixel: 60, 111f., 122, 124–126, 166, 200, 202, 400; P.grafik: 199, 201
- Plankalkül (Programmiersprache): 69f.
- Platform Studies: 27–29, 38, 74–76
- Platine: 102, 233, 240, 245 [FN], 265–269, 275, 278, 317, 321, 393; P.nen-Revision: 77
- Plattform: 15f., 21f., 27f., 32, 38, 42, 44, 52f., 71, 74–76, 81, 88, 91f., 146, 162, 173, 188, 196, 202, 212, 216–218, 221, 223, 226–228, 231f., 236, 238f., 250, 252f., 260, 305f., 312, 316, 319–323, 326f., 330, 394, 396, 398; Original-P.: 196, 198, 218, 222, 224, 230, 233f., 236, 250, 322, 392
- Poetologie: 54, 57, 85
- Pointer: 112, 114, 137, 372, 391; P.-Arithmetik: 112f.
- POKEY: 110
- Polling: 107
- polynomisch: 111
- Pong: 90, 103, 105 [FN], 213
- Portierung: s. Adaption
- Präprozessor-Direktive: 141f., 147, 153
- Preservation: 19, 21, 25, 31, 34–36, 38f., 146, 216f., 224, 250, 260; Hardware-P.: 22, 87, 220, 255, 257, 276, 327, 332; Software-P.: 16, 22, 35f., 38, 87, 130, 134, 137, 146, 173, 216f., 220, 222, 224, 226, 239, 252, 257, 322, 326f., 332f.; Knowledge P.: 22, 257, 260, 276, 284
- Prince of Persia*: 31, 227
- Privatisierung: 18, 41, 44, 50
- Program Bumping: 83, 304
- Programm: P.analyse 129; P.text/code: 18, 71, 83f., 116, 128–130, 135–137, 140f., 145, 150, 152–155, 161, 163f., 169, 171f., 190, 236, 247, 279, 283, 326, 379
- Programmieren: P.fehler: 82f., 138, 168, 170, 196, 225, 228, 302f.; P.lehre 87, 171, 297, 303f., 307f., 312, 314, 316, 322; P.trick: 88, 95, 396
- Programmiersprache 14, 18, 21–23, 49–51, 68f., 71, 91, 109, 114, 116, 119, 129, 140f., 145, 149–152, 162, 164, 170f., 173, 181–184, 191, 202, 206, 211, 219, 225, 248, 264, 278, 293, 301f., 306f., 309f., 312, 314f., 317, 320–322, 326f., 330, 334, 389; höhere P.: 109, 151, 307
- Programmierung (P.paradigma): 162f., 232; imperative P.: 162f.; prozedurale P.: 162f.; objektorientierte P.: 129, 140, 173, 301; strukturierte P.: 115, 117, 127, 137, 144, 149f., 155, 162f., 167, 229, 278; modulare P.: 173, 232, 242, 305, 320
- programming sheet: 187f.
- Protected Mode: 292, 294
- Protokoll: 15, 18
- Prototyp: 45f., 49, 108 [FN], 109, 138, 161f., 203, 219, 309, 388
- Prozedur: 69, 123, 152, 163, 304
- Prozessortakt: s. Taktrate
- Pseudozufall: s. Zufall
- Python (Programmiersprache): 213, 308, 391
- Quelle: 9, 17f., 21f., 31, 42, 44, 49, 51, 52 [FN], 60, 87, 104, 142, 156, 160f., 174, 232, 242, 251, 255, 256, 269, 274–276, 295, 299, 328, 334, 376
- Racing the Beam: 125f., 168
- RAM 107, 111–113, 119, 123–125, 127f., 142 [FN], 144f., 152, 164, 189f., 192f., 196 [FN], 198, 201f., 232, 236f., 243, 262, 265, 270–274, 279–283, 321f., 327, 369, 373, 374, 398; DR.: 199, 274, 277; SR.: 236f., 251, 281, 392f.; R.-Image: 226; Scratchpad-R.: 124, 189, 236

- RAM *Pong*: 124 [FN]
 Re-enactment: 70, 80–82, 88, 93, 101, 105, 134, 155, 158–160, 169, 172f., 174, 187, 198, 202, 231, 249, 287, 325, 327, 334
 Real/Realisierung: 14, 26, 34, 56, 64, 71, 72f., 76, 92, 98, 113, 114, 126, 132, 149, 153, 158, 169, 174, 176f., 205, 213, 216, 223, 227, 237f., 242, 244, 248, 250, 257, 301, 327, 333, 398, 400; r. mode: 293
 Rechenmaschine: 41, 61f., 64, 68, 203, 213; Kugelr.: 64
 Rechenschaltung: 96f., 158
 Rechenwerk: 62–64, 259
 Red Stone: 212–215
 Redaktion: 19f., 83, 115
 Referenz 26, 138, 140, 142, 144, 147, 157, 278; implizite R.: 138, 143, 147, 152; Selbstr.: 147, 157, 419
 Rekursion: 38, 65, 70, 300
 Relais: 49, 74
 Remake: 32
 Retro-Didaktik: 22, 87, 330
 Retrofitting: 252–25, 327, 330
 Retrogame/Retrogaming: 28f., 30, 32, 38, 227
 Reverse Engineering: 20, 114f., 149, 218, 242, 322, 333, 392
 Revival: 26
 Rezeption: 29, 129, 130, 132–138, 159, 169, 183, 247; R.ästhetik/R. stheorie: 132–135, 138, 169, 242; R.geschichte: 22, 137, 146; produktive R.: 133
 Rezipient: 26, 55, 132, 146, 247, 318
 RIOT: 124
 Rock: 320f.
 ROM: 33, 77, 83f., 115 [FN], 119, 126–128, 141, 143–145, 147, 165, 216, 232f., 237; R.-Datei: 33, 83, 218; R.-Modul: 124, 218, 232 [FN], 239, 273, 307, 314, 398f.; R.-Listing: 143
 Routine: 83, 88, 107, 121, 126–128, 135, 142–145, 147f., 163–165, 167–169, 188, 190, 192, 201, 236, 244 [FN], 277, 285, 304, 371, 389–391, 396; Betriebssystemr.: 116, 121, 137 [FN], 141, 145, 147; Subr.: 106f., 144, 163, 165, 190, 304, 379, 382–384, 386
 Ruby (Programmiersprache): 308
 S-100: 262–264, 272
 Sammler: 11, 16, 260, 264
 Sammlung: 9, 11, 16, 76f., 259f.
 SC/MP: 252f.; S. II: 252f.; S. *Retrofit Kit*: 252–254
 Schach: 16; S.computer: 69f., 72, 146; S.programm: 44, 69, 194
 Schalt: S.algebra: 41; S.er: 84, 268 [FN], 273, 281f.; S.kreis: 20, 64, 67f., 73, 135, 267; S.plan: 9, 18, 59 [FN], 61, 65f., 69, 94f., 102, 132, 153, 156–158, 242, 266, 275f., 294, 375; S.symbol: 153
 Schieberegister: 64
 Schleife: 66, 106f., 115, 121, 148, 150f., 161, 165, 190f., 236, 378f., 386; Endloss.: 122, 198; Wartes.: 106f., 159f., 168, 191, 378, 386
 Schlüsselwort: 164
 Schnittstelle (Hardware): 15, 84, 88, 134, 153, 191, 226, 238, 244, 247, 258, 262, 264f., 318, 332; serielle S.: 109f., 270
 Schnittstelle (Software): 26, 71, 173, 220 [FN]321, 392, 397
 Schreibmaschine: 15, 262
 Schrift: 14, 25, 34, 53, 59, 60, 65 [FN], 169f., 262, 332; Hands.: 146; Vors.: 62, 302, 311
 Schwierigkeitsgrad: 26, 28, 83, 211, 231
Scientific American: 181
 Scrolling: 123, 230f.; Hardware-S.: 120, 123
 SD-Speicherkarte: 218, 238, 239, 271, 394
 Selbstreflexivität: 65 [FN], 250
 Semantik/Semantisierung: 34f., 103f., 115, 132, 148, 150, 168, 221, 228, 245, 247, 305
 Sensor: 199, 215, 268 [FN]
 Shareware: 388
 SID: 390
 Signal: 14, 105, 107, 125, 129, 132, 138f., 144, 154, 157–159, 169, 182, 199, 203, 209, 212–214, 249, 261,

- 268; S.labor: 10f., 87, 260, 265, 270; S.verarbeitung: 65, 72, 110, 203
- Signetics 2650: 105, 107, 136, 160, 161 [FN], 187f., 192, 196, 218, 232, 244, 317, 319, 378, 398f.; S. 2636 (PVI): 232, 243, 400
- Simulation 13, 20 [FN], 22, 34, 36, 81, 88, 91–102, 107f., 128, 155f., 175–178, 182f., 186f., 195, 200, 203, 205, 213–216, 219–221, 223, 303, 322, 326f.; physikalische S.: 92, 97, 114, 134f., 137; S.theorie: 26, 176 [FN]; kontinuierliche S.: 176; diskrete S.: 176, 178, 183, 202, 214, 327; Universalitätss.: 34, 177, 181, 223, 248, 288; Monte-Carlo-S.: 177
- Sinus: 99, 114f., 121, 128f., 135f.
- Skeuomorphismus: 222, 226, 244–246, 322
- Software: S.emulation: 35, 174, 213f., 216, 220–223, 243, 247, 257, 322, 332f.; S. Engineering: 18, 53, 225, 228f., 320; S.gattung: 16, 52, 93, 172, 219, 230, 295, 317, 334; S.geschichte: 51, 88, 129, 134, 141, 147, 169, 172f., 326; S. Preservation: 16, 22, 35f., 38, 87, 130, 137, 146, 173, 217, 220, 222, 225f., 239, 251, 257, 322, 326f., 332; S. Studies: 29
- Sound: 26, 110, 119–121, 138, 222, 224, 226, 229, 231–233, 271, 312, 320, 392, 399; S.prozessor: 109f., 126, 400
- Sourcecode: 31, 66, 69, 109 [FN], 113 [FN], 114, 117, 119, 136, 141, 142f., 155, 162–164, 168, 171, 188–190, 196, 228, 232f., 236 [FN], 258, 304, 322, 332, 378–387
- Sowjetunion: 49, 77
- Soziologie: 48, 59, 79, 178, 271; Mediens.: 37
- Spaghetti-Code: 163, 166
- Speicher: 13, 19, 22, 38, 60f., 65f., 90, 105f., 110–113, 117, 119, 121, 123f., 127, 131, 135, 141, 142 [FN], 143f., 152f., 164–165, 169, 173, 188 [FN], 189–196, 198–203, 209, 211f., 216, 218, 226, 228, 230, 232, 236f., 238f., 243f., 248, 258f., 272–277, 279–283, 288, 291, 294, 307, 321f., 326, 327f., 379, 381–387, 389f., 383–395; S.erweiterung: 198, 232 [FN], 244, 272, 276f.; S.kapazität: 34, 111, 170; S.Matrix: 193, 198; S.platz: 83, 128, 163; S.zelle: 152, 192–194, 201, 237, 243, 279f., 387; Funktionsss.: 63; Massens.: 228, 238–240, 264, 321, 327
- Spiel: S. des Lebens: s. Game of Life; S.konsole: 15, 22, 27, 33, 75f., 84, 89f., 103, 198f., 124, 166f., 217–219, 228, 232, 235–237; S.regel: 103, 185, 385–387; S.theorie: 178, 203, 206, 215, 334
- Sprachspiel: 65, 90
- Sprachwissenschaft: s. Linguistik
- Sprite: 110f., 119, 125–127, 310, 390, 384; S.-Kollision: 82
- Stella*: 83, 128, 168, 391
- Stil: 9, 26, 32, 62, 66, 73, 83, 230, 311, 316; Programmiers.: 141, 161, 163, 311
- stochastisch: 179, 192
- Strukturgeschichte: 21, 48
- Subkultur: 27, 45, 389
- Sukzession: s. Fortschritt
- Summierer: 97, 99, 153, 157
- symbolisch: 14, 37, 44, 53, 58, 62f., 98, 129, 132, 149, 153, 171, 176, 178, 216, 223, 225, 237f., 245, 247, 239, 257, 278f., 301, 327, 332
- SymbOS*: 321
- synchron (ling.): 14, 129, 132f., 141, 172, 249, 251, 326, 332
- System: S.variable: 115f.; Hosts.: 93, 174, 196, 219, 222–226, 244, 247–252; Ziel-S.: 107, 174, 244, 248, 256
- Szene/Scene: 11, 14 [FN], 18, 21, 22, 25, 32, 36, 45, 87, 118, 217, 228f., 238, 259f., 262, 276, 289, 333f., 395; Demos.: 37f., 89, 93, 117f., 172, 376; Hackers.: 11, 45, 50, 182
- T42: 102f.
- Table Tennis*: 90, 103
- Takt: 107, 110, 118, 122, 160f., 166, 252, 262, 266, 395; T.geber: 10; T.zyklen/frequenz: 160, 168;

- T.perioden: 160; T.rate: 107, 160, 222, 244
 Taschenrechner: 50, 376
 Tastatur: 15, 109f., 187, 196–198, 262, 264, 266–271, 274f., 318
 technical report: 52, 66f., 69, 83, 87, 287
 Techno-Mathematik: 20, 61, 65, 67, 70, 72, 292
 Tennis: 90, 93, 97, 101, 103–105, 107; *T. 2650*: 107f., 141, 399; *T. for Two*: 90f., 93, 101, 103, 158, 177; *T. für Drei*: 101f., 158f.
 Terminal: 43, 109
Tetris: 204, 213
Texor: 77
 Textadventures: 388
 Textsorte: 83, 134
 Textualität: 132, 148, 156
 Thema-Rhema: 148, 152
 Theorie mittlerer Reichweite: 16, 69, 79f., 85, 300, 326
 TIA: 109f., 125f., 166f., 249
 Time Shring: 50
 Timer: 124, 167, 234f., 389, 391
 Token: 164f.
TOPS-10: 50
 Toy Computing: 175, 205f., 215f., 327
 Tradition: 57f., 71, 117, 262, 307, 322
 Transistor: 49, 74, 221
 translation: s. Übersetzung
 Trial and Error: 172, 242, 276, 287, 299, 301f., 305, 323, 328
 TTL: 242, 252, 375
 Turingmaschine: 34, 37, 41, 72, 181–183, 215, 223, 238, 248, 288, 327, 330, 332
 turingvollständig: 62, 181, 188, 205, 211f.
TuroChamp: 70
 Type-in-Listing: 119, 121, 136, 145f., 187, 282, 312, 334
 Übertrag: 61f., 64
 Übersetzung: 172, 211, 226, 236, 245, 247–249
 Unconventional Computing: 175, 202f., 205, 207 [FN], 215, 216, 327, 331
 Universalrecher: 62, 72, 177
 Unix: 131, 392
 Unterfläche: 28, 64, 84, 134, 155, 214, 332, 334
 Urheberrecht: 33, 174, 294
 Urtext: 135
 USB: 226, 238f., 270 [FN], 393
 VACS: 196, 233, 398
VCS Boing!: 124, 126, 128, 141f., 151, 154, 166, 168
 Vektor: 126, 155; V.bildschirm: 103; Einsprungv.: 137
 Vernetzung: 26, 68, 166, 255, 260, 322
 Verweis: 130, 131, 141, 145, 167, 149, 152f., 312
 Videospiel-Crash; 82
 viewport: 110, 112–114, 119, 123
 Virtualisierung: 34, 84, 217f., 220, 226, 237f., 248f., 256f.
 Volladdierer: 64
 Von-Neumann-Architektur: s. Architektur
 Werkzeug/Tool: 22, 29, 31, 80, 83f., 115f., 126, 156, 170, 174, 216, 226, 238, 242, 244, 249, 258f., 261, 274–276, 282–289, 284f., 299, 307–310, 314, 320, 323, 327, 332, 376, 389, 391, 398, 400
 Williams-Kilburn-Tube: 111, 199f., 215, 327
WinArcadia: 188f. [FN], 195–197, 219, 233, 237, 243f., 259, 251, 327, 378, 398–400
Windows: 16, 28, 33, 131, 238f., 294, 392, 398; *W. XP*: 196, 250 [FN]; *W. 7*: 233, 250
Wolfram Alpha: 181
 Wort (Informatik): 160, 164
 Wort (ling.): 148, 152
 Wrapper: 110
 Zahlensystem: 41, 62, 206, 310
 Zeichensystem: 129, 132, 176, 248, 332
 Zeichentheorie: 85, 216
 Zeile: 64, 120, 132, 137, 149–152, 164f., 193, 277–279, 294, 387; Bildschirmz.: 60, 125, 166f., 236, 394; Z.nummer: 113, 120 [FN], 164f., 278
 zeitkritisch: 36, 72, 93, 131, 156, 167f., 223, 225, 234, 251, 304f., 391

Zeitschrift: 16–18, 23, 49, 63, 118, 145, 174, 261f., 318, 334
Zilog: Z180: 63; Z80: 121, 321; Z80A: 77, 118f.
Zitat: 139f., 142, 169

Zufall: 55, 190–192, 237f., 269f.; Z.zahl: 177, 192, 196, 238, 244, 318, 332, 374, 379, 287; Z.zahlengenerator: 105, 177, 237, 244; Pseudoz.: 126, 185, 192, 237f., 332