

Soundchip-Musik

Computer- und Videospielemusik von 1977-1994

Magisterarbeit

Universität Lüneburg

Angewandte Kulturwissenschaften

Musik

Erstgutachter: Dr. Rolf Großmann

Zweitgutachter: Dr. Christian Bielefeld

Vorgelegt am 04.08.2005 von:

Nils Dittbrenner

Bertha-von-Suttner-Str. 10

21335 Lüneburg

Matrikelnummer 119690

nils@pingipung.de

04131-209080

Inhaltsverzeichnis

1 Einleitung	1
2 Technische Voraussetzungen: Soundchips	4
2.1 Die Frühzeit: Computerspiel-Sound ohne Soundchips	7
2.2 PSG-Soundchips (1977-1982)	8
2.2.1 Der Stella / TIA – Soundchip des Atari VCS.....	9
Der Stella / TIA unter musikalischen Gesichtspunkten.....	10
2.2.2 Atari Pokey.....	11
2.2.3 General Instruments AY-3-891x / Yamaha YM 2149 SSG.....	14
Programmierung des AY-Chips.....	16
2.2.4 Texas Instruments SN 76489 AN.....	19
2.3 Weiterentwickelte PSG-Soundchips	20
2.3.1 der SID-Chip des C64 (MOS 6581/ 8580 SID).....	21
2.3.1.1 Klangerzeugung und technische Übersicht.....	21
2.3.1.2 Beschreibung der Programmierung anhand der Registertabelle	23
2.3.2 Der 2A03 im Nintendo Famicom / Entertainment System.....	26
Funktionsweise der Klangerzeugung.....	27
2.3.3 PSG-Sound in den 1990ern: der Nintendo Game Boy	29
2.4 Die Bedeutung der PSGs für Soundchip-Musik	30
Exkurs: Die japanische Soundtrackkultur.....	33
2.5 FM-Synthese und Sampling: neue Möglichkeiten der Klangerzeugung	34
2.5.1 Das NEC PC-Engine: zwischen PSG-Sound und Sampling.....	35
2.5.2 FM-Synthese: die Yamaha YM-Soundchips.....	36
2.5.3 Homesampling: Der Commodore Amiga	37
Entwicklung der Tracker und MOD-Dateiformate.....	38
2.5.4 Das Sega Mega Drive.....	40
2.5.5 Nintendo Super Nes: DSP für ADPCM / Sample Playback.....	41
2.5.6 Die Bedeutung der Sampling-Soundchips für Soundchip-Musik.....	42
2.6 Vom PC-Speaker bis General MIDI: Klangerzeugung im PC (1981-1994)	44
2.6.1 Der PC-Speaker.....	44
2.6.2 IBM PC Junior / Tandy 1000.....	46
2.6.3 Die Verwendung von FM-Chips im PC.....	46
2.6.4 Die Wavetable-Synthesizerkarten von Roland.....	48
2.6.5 Die Entwicklung der PC-Soundkarten über 1991 hinaus.....	49
2.7 Zwischenfazit: Eigenheiten von Soundchips	51
3 Strategien der Gestaltung von Computerspielmusik für Soundchips	53
3.1 Musik in interaktiven Spielen	54
3.1.1 Eine Typisierung der verschiedenen Spielmusiken.....	55
3.1.1.1 Sekundäre Spielmusik.....	56
3.1.1.2 Ingame Spielmusik.....	58

3.1.2 Adaptive Audio / adaptive Musik für Soundchips.....	61
3.1.2.1 Adaptivität durch Synchronisation von Eingabe und Musikwiedergabe	62
3.1.2.2 Adaptivität durch Bruch / Wechsel der Musik.....	63
3.1.2.3 Adaptivität bei Fortschritt.....	64
3.1.2.4 Adaptive Soundeffekte: die Technik des Mickey Mousing.....	65
3.1.3 Zwischenfazit: Funktionale Aspekte von Computerspielmusik.....	68
3.1.4 Fallbeispiel: Bubble Bobble.....	69
3.2 Strategien der Gestaltung im Spiegel der technischen Limitierungen.....	74
3.2.1 Beschränkte Polyphonie.....	76
3.2.2 Limitierungen der Klangfarbe.....	79
3.2.3 Beschränkter Speicherplatz.....	81
3.2.4 CPU-Last und weitere Beschränkungen „von außen“	84
3.3 Zwischenfazit: Computerspielmusik im Spiegel ihrer Paradigmen.....	86
3.4 Plattformübergreifender Hörvergleich.....	88
Parodius.....	89
Bubble Bobble.....	90
The Secret of Monkey Island.....	91
4 Populäre Soundchip-Musik.....	92
4.1 Sound Culture.....	93
4.2 Verfahrensweisen und Tools.....	95
4.3 Genre-Bildung.....	98
4.4 Chiptunes.....	100
4.5 Micromusic.....	103
5 Fazit und Ausblick.....	105

Anhang

Bibliographie

Computerspielmusik.....	109
Computerspielhardware und -software.....	110
Scans von originalen Datenblättern / Application Manuals	112
FAQs / Anleitungen zu Spielkonsolen und Soundchips.....	113
Sonstige Sekundärliteratur.....	114
vom Autor geführte Interviews.....	115
Abbildungsverzeichnis.....	115
Tabellenverzeichnis.....	116
Erwähnte Computerspiele.....	117
Hörbeispiele.....	118
Internet-Linkverzeichnis.....	122
Erklärung nach § 25 (6) der Magisterprüfungsordnung.....	125

1 Einleitung

Computerspiele¹ sind in den letzten Jahren zunehmend in den Mittelpunkt wissenschaftlicher Betrachtung gerückt. Vor allem die Literatur- und Medienwissenschaften sowie ihnen verwandte Disziplinen (wie die Film- und Kommunikationswissenschaften) widmen diesem vergleichsweise jungen Medium immer mehr Aufmerksamkeit. Anders als etwa dem Bereich der narrativen Eigenheiten oder der grafischen Repräsentation ist dem Thema Musik und Klang in Computerspielen von der Wissenschaft bisher kaum Interesse zuteil geworden. Dennoch sind allein für die diese Arbeit betreffenden Computerspielplattformen von Ende der 1970er bis Mitte der 1990er Jahre tausende verschiedene Spiele² erschienen, ein Großteil von ihnen mit Musik unterlegt. Allein der Quantität wegen scheint eine qualitative, (musik-)wissenschaftliche Auseinandersetzung mit dem Thema daher überfällig. Die wenige verfügbare Fach-Literatur zu der Programmierung von Musik und Klang³ für Computerspiele ist entweder stark praxisorientiert oder stammt aus der Disziplin der Informatik und befasst sich mit der Programmierung der neueren Plattformen bzw. deren Entwicklungsumgebungen⁴. Eine umfassende geschichtliche Aufarbeitung ebenso wie eine Vorstellung der technischen Voraussetzungen für Musik in Computerspielen ist bislang nicht geleistet worden. Der Eigenklang von Computerspielmusik der 1980er Jahre erlebt seit Ende der 1990er Jahre ein regelrechtes 'Comeback' in der populären elektronischen Musik, die auf gesampelten oder emulierte Klängen von Spielen basieren. Im Internet hat sich eine lebendige Musikkultur rund um dieses Phänomen gebildet.

Der angesprochene Eigenklang von Soundchip-Musik markiert eines der Hauptinteressen dieser Arbeit, da scheinbar die Limitierungen der Klangerzeugung eine gewisse Simplizität bedingen, die den Soundchip-Klang im Vergleich zu anderen elektronischen Klangerzeugung wie Synthesizern deutlich abheben.

Für Soundchip-Musik⁵ gilt der Grundsatz, dass es sich nicht um klassische notierte, sondern vielmehr um Medienmusik handelt, die, wie der Musik- und Medienwissenschaftler Rolf Großmann erläutert, durch „die Zugehörigkeit zu einem durch spezifische technische Voraussetzungen geprägten Bereich musikalischen Handels“ (Großmann 1997a: 239) charakterisiert ist. Großmann erläutert den Begriff Medienmusik in starker Abhängigkeit zu den sog. Audio-Medien, auf denen Musik fixiert ist. Hier offenbart sich eine Besonderheit von Soundchip-Musik, da sie i.d.R. nicht in aufgenommener Form vorliegt, sondern während des Computerspielens als Ergebnis einer Prozessierung von Daten in den sog. Soundchips über Lautsprecher ausgegeben

¹ In dieser Arbeit soll aus Gründen der besseren Lesbarkeit der Begriff „Computerspiele“ für alle digitalen Bildschirmspiele (Computer-, Video-, Handheld- und Automaten Spiele) gebraucht werden.

² Den Übersichten Forsters (2005) zufolge sind es sogar über 15000 Titel für die Plattformen zwischen 1977 und 1993. Da jedoch nicht jedes Spiel Musik enthält und teilweise zahlreiche Konvertierungen der selben Spiele erschienen, wurde die Zahl an dieser Stelle relativiert. Deutlich wird jedoch, dass eine Menge für Computerspiele programmierte Soundchip-Musik existiert.

³ Der Begriff Klang wird in Abgrenzung zur Musik verwendet, um nicht mit den Begriff „Sound“ zu kollidieren, der in dieser Arbeit vor allem im Hinblick auf die „Sound Culture“ von Soundchip-Musik benutzt wird. Dennoch ist „Sound“ für ein einzelnes Klangobjekt in Computerspielen ein gängiger Begriff. In der vorliegenden Arbeit wird für ein einzelnes Klangobjekt daher der Begriff „Soundeffekt“ verwendet.

⁴ Der Programmierung von *Game Audio* widmen sich beispielsweise Boer (2003) sowie Turcan & Wasson (2004). Marks (2001) und Sanger (2003) liefern in ihren sehr praxisbezogenen Publikationen einen Einblick in das 'Berufsbild' des Computerspielmusik-Komponisten.

⁵ Unter Soundchip-Musik soll grundsätzlich jene Musik verstanden werden, die ganz oder teilweise von der Klangerzeugung von Soundchips (Kapitel 2) abhängt.

wird⁶. Ohne eine grundlegende Beschreibung der technischen Voraussetzungen wie der spezifischen Medienrealität von Soundchip-Musik kann eine Analyse des Phänomens also kaum zu einem Erkenntnisgewinn führen:

„Gerade für ein Verständnis, für systematische Beschreibung und Analyse von medienmusikalischen Vorgängen führt eine allein auf Klangstrukturen oder deren kompositorische Regeln ausgerichtete Theoriebildung in eine Sackgasse. Klangstrukturen müssen zwar auch dort untersucht werden, ein neuer Fokus kann jedoch erst dann sinnvoll auf diese gerichtet werden, wenn zumindest die Grundbedingungen ihrer spezifischen Existenz in der [...] 'Medienrealität' geklärt sind“ (Großmann 1997b: 65, Hervorh. i. O.).

Aus dem Antrieb heraus, die Grundbedingungen von Soundchip-Musik im Sinne der spezifischen Medienrealität umfassend darstellen zu wollen, soll die vorliegende Arbeit mit einer recht detaillierten Vorstellung der verschiedenen Soundchips beginnen. Diese Vorstellung setzt ein umfassendes Vorwissen aus dem Bereich der elektronischen Klangerzeugung voraus: musikwissenschaftliche Fachbegriffe insbesondere der elektronischen Klangsynthese wie Oszillator, LFO, Hüllkurve, Ringmodulation etc. werden nicht weiter erklärt⁷. Gleichzeitig wird auch im Hinblick auf den Umfang der Arbeit eine Selektion vorgenommen. So werden jene Soundchips sehr viel ausführlicher beschrieben, die einen größeren Einfluss auf die populäre Soundchip-Musik⁸ genommen haben, als jene, die den Übergang zu neueren Verfahren der Musik- und Klangerzeugung in Computerspielplattformen (FM-Synthese und Digital Audio) markieren. Den zeitlichen Rahmen markiert das Aufkommen der ersten programmierbaren Heimkonsole, das ATARI VCS (1977) sowie die breite Implementierung von Sampling in CD-Qualität durch die CD-ROM-Technologie, welche spätestens mit der SONY PLAYSTATION (1994) für Computerspiele zum Standard wurde. Durchgängige Computerspielmusik wird indes erst ab Anfang der 1980er Jahre in den Spielen implementiert.

Die *Medienrealität* von Soundchip-Musik für Computerspiele umfasst neben den technischen Voraussetzungen selbstverständlich auch ihre Anwendung im jeweiligen Spiel. Daher schließt sich eine Auseinandersetzung mit der funktionellen Einbindung von Musik und Klang im Rahmen interaktiver Spiele und mit den daraus folgenden formalen Besonderheiten an. Auch diese Auseinandersetzung hat bisher nicht stattgefunden, allein der Bereich *adaptiver* Musik und Soundeffekte wird anhand aktueller Beispiele mehr und mehr untersucht (vgl. Kapitel 3.1.2). Die Implikationen für Soundchip-Musik durch ihre Verwendung in Spielen werden im Kapitel 3 dieser Arbeit erläutert, indem die verschiedenen in Computerspielen zu findenden Anwendungen musikalischer Untermalung kategorisiert werden. Auf die bei den Spielern⁹ stattfindenden Rezeptionsprozesse wird nicht detailliert eingegangen.

⁶ Ebenso verhält es sich mit den relativ neuen, auf der Emulation von Soundchips basierenden Playern und PlugIns, welche die als Dateien gespeicherten, aus den Spielen extrahierten Musikdaten wiedergeben.

⁷ Wenig in der Materie der elektronischen Klangerzeugung versierten Lesern sei daher die Lektüre von z.B. Ruschkowski 1998 (insb. S.149ff) empfohlen

⁸ Unter „populärer Soundchip-Musik“ wird diejenige verstanden, die im Rahmen der populären elektronischen Musik auf Basis einer „Sound Culture“ von Soundchip-Musik entsteht (vgl. Kapitel 4).

⁹ In dieser Arbeit wird einer besseren Lesbarkeit zugute bei der unbestimmten Nennung von Akteuren die jeweils männliche Form bevorzugt.

Da neben den eigentlichen Klangerzeugern auch die Bedingungen ihrer Entstehung auf die erklingende Musik einen großen Einfluss nehmen, beschreibt das darauf folgende Kapitel die Beschränkungen und die Strategien der Gestaltung von Computerspielmusik. Wiederum ist die Literaturlage für diese Kapitel als dürftig zu klassifizieren. Über die Programmierung für Soundchips finden sich neben einigen Anwenderorientierten Ratgebern keine Quellen. Daher wurde hierfür die Methode der themenzentrierten, qualitativen Befragung gewählt, um Informationen zu erhalten. Hierfür wurden Telefoninterviews mit David Warhol, Michael Pummell und Chris Hülsbeck geführt, die als Komponisten und Programmierer von Computerspielmusik jahrelange Erfahrung gesammelt haben.

Um eine möglichst große Anschaulichkeit zu gewähren, liegt dieser Arbeit eine CD-ROM bei, auf der sich neben Hörbeispielen¹⁰ auch Emulatoren und sog. ROM-Images befinden, welche die besprochenen Spiele mit kleinen Abstrichen auch heute noch erfahrbar machen. Die in der Arbeit grundsätzlich zu konstatierende Nähe zum Gegenstand ist auf der einen Seite nötig, um einen teilbaren Erfahrungshorizont zu gewähren. Auf der anderen Seite folgt sie der noch mangelhaften Literaturlage, die eine distanziertere Betrachtung als Reaktion auf die vorliegende Arbeit in Kritik und Diskussion der hier beschriebenen Phänomene, vor allem der Zukunft überlässt. Aus den angesprochenen Aspekten lassen sich folgende Erkenntnis leitende Fragestellungen extrahieren:

- Welche Klangerzeugung bieten Soundchips, welche Unterschiede gibt es zwischen den verschiedenen Soundchips ?
- Wie gestaltet sich die Programmierung von Musik für Soundchips insbesondere im Rahmen von Computerspielen ?
- Kann die Hypothese, dass Musik für Soundchips vor allem durch die Limitierungen der Technik mit einer eigenen Klanglichkeit in Verbindung gebracht wird, bestätigt werden ?

¹⁰ Diese werden im Text durch das Zeichen  mit entsprechender Datei-Nummer gekennzeichnet.

2 Technische Voraussetzungen: Soundchips

Die Vorstellung der technischen Voraussetzungen von Soundchip-Musik ist in dieser Arbeit das zentrale Element, hat sie doch zum Ziel, herauszuarbeiten, in welchem Umfang Form und Klangästhetik der Computerspielmusik mit den technischen Grundlagen ihrer Klangerzeugung zusammenhängen, also in wie weit sich ihre Verwurzelung in der jeweiligen technischen Entwicklungsumgebung auf die musikalische Gestaltung auswirkt. Da Soundchip-Musik, wie in der Einleitung erläutert, als Medienmusik und damit als von ihren technischen Voraussetzungen wie ihrer Medienrealität abhängige Musik begriffen wird, markiert die Vorstellung der technischen Voraussetzungen von Soundchip-Musik den ersten Teil.

Es lässt sich tautologisch sagen, die technischen Voraussetzungen von Soundchip-Musik seien Soundchips. Einer weiter gefassten Perspektive folgend muss als technische Voraussetzung von Soundchip-Musik jedoch die *Gesamtheit* der zur Musik nötigen Technik betrachtet werden. Diese umfasst neben den Soundchips auch Speicherplatz und -medium, sowie Beschränkungen, die aus ihrer Funktion als Computerspielmusik folgen, und die Implikationen der zur Programmierung notwendigen Software. Eine umfassende und detaillierte Vorstellung dieser Punkte ist im Hinblick auf die erwähnte Literaturlage kaum zu leisten, dennoch soll eine erste, diese Bereiche berücksichtigende Aufarbeitung des Themas angeschoben werden.

Der für diese Arbeit zentrale Begriff der Soundchips¹¹ existiert in der Literatur bisher kaum, weshalb sich an dieser Stelle eine Definition anbietet: Soundchips sind integrierte Schaltkreise (ICs, Chips), die ab Ende der 1970er Jahre zur Erzeugung von Klang und Musik speziell für Spielautomaten und später auch Spielkonsolen und Heimcomputern entwickelt werden, um die teure und aufwendige, auf diskreten, analogen Schaltkreisen basierende Klangerzeugung früher Spielautomaten ersetzen zu können. Soundchips lassen sich am offensichtlichsten nach Art der Klangerzeugung gliedern. So lässt sich differenzieren zwischen

- den auf einfachen Wellenformen basierenden Programmable Sound Generators (PSGs¹²),
- auf dem Verfahren der Frequenz Modulation basierende (FM-)Chips,
- und Soundchips, die die nötigen Berechnungen für das Verfahren des Samplings¹³ leisten.

¹¹ Begriffe wie „Audio Device“ oder gar Instrument mögen als Oberbegriff für Hardwarelösungen, die Klang und Musik in digitalen Systemen erzeugen, sinnvoll erscheinen. In dieser Arbeit soll für solche jedoch allgemein der Begriff Soundchips benutzt werden, wobei anzumerken ist, dass die Klangerzeugung von FAMILICOM / NES und PC-ENGINE nicht auf speziellen Soundchips basieren, sondern in die jeweilige CPU integriert ist.

¹² PSG steht für „Programmable Sound Generator“. Die auf einfachen Wellenformen basierenden, frühen Soundchips werden so im Englischen bezeichnet.

¹³ Die verschiedenen, teilweise auch in FM-Soundchips integrierten Verfahren des Samplings werden allgemein durch die Abkürzungen PCM (Pulse Code Modulation) und PWM (Pulse Width Modulation) bezeichnet. Pulse Code Modulation bezeichnet das unkomprimierte Wandeln analogen Klangs durch Speichern von verschiedenen Werten in einem von der Bit-Tiefe auf der y-Achse und der Abtastfrequenz auf der x-Achse definierten Koordinatensystem. Bei der Pulse Width Modulation werden die gesampelten Informationen durch Modulation der Pulsbreite einer Pulswelle gespeichert. ADPCM (Adaptive Differenz Puls Code Modulation) ist ein Verfahren, in dem anstelle der einzelnen Sampling-Werte die relationale Abweichung dieser voneinander gespeichert wird. Zu den verschiedenen Kodierungsverfahren vgl. Ackermann (1991: 83, 227). In CD-Qualität (16 Bit-Auflösung bei einer Abtastfrequenz von 44,1 kHz) gesampeltes und wiedergegebenes Material wird in der Computerspielindustrie als „Digital Audio“ bezeichnet.

Die kanadische Musikwissenschaftlerin Karen Collins (2003a: 1) schlägt für die historische Unterscheidung der verschiedenen Soundchips eine Zeiteinteilung vor, die drei Abschnitte umfasst. Ihr erster beginnt mit den 8 Bit-Plattformen, in welcher sie neben den frühen Soundchips (ab 1977) auch die fest integrierten, analogen Schaltkreise (ab den frühen 1970er Jahren) einschließt¹⁴. Ihr zweiter Abschnitt wird „the 'in-between' years“ genannt, in denen FM-Synthese, das MIDI-Datenformat sowie Sampling-Fähigkeiten implementiert werden wodurch die Gestaltungsmöglichkeiten für Komponisten steigen. Der dritte Abschnitt beginnt mit Einzug der CD-ROM-Technologie¹⁵ und zeichnet sich durch den Wegfall vorher bestehender Einschränkungen und somit eine künstlerische Freiheit in Bezug auf die Wahl des Klangmaterial aus. Durch die Einbindung von *streamed* bzw. Redbook-Audio¹⁶ wird es möglich, mit kompositorischer Freiheit, ohne wesentliche, durch die Soundhardware auferlegte Beschränkungen, musikalische Ideen umzusetzen. Die technischen Limitierungen von Soundchip-Musik in Bezug auf Klangfarbe, Klanggestaltung, Speicherplatz und andere Ressourcen werden damit hinfällig. Mit dieser Entwicklung endet auch die den technischen Voraussetzungen stark untergeordnete Soundchip-Musik, um die es in dieser Arbeit geht.

Die folgenden Kapitel widmen sich den ersten beiden genannten Abschnitten, wobei Collins' ersterer im Rahmen dieser Arbeit in frühe PSG-Soundchips (Stella / VIA, AY-3891x, SN 76489 AN und Pokey) und weiterentwickelte, den früheren im Hinblick auf die Möglichkeiten der Klangerzeugung überlegene PSG-Soundchips (MOS SID & und 2A03) unterteilt wird. Hierbei beschränkt sich die Auswahl auf die am häufigsten verwendeten und für die Geschichte der Computerspiele wichtigsten Soundchips. Nicht eingegangen wird z.B. auf die Heimcomputer COMMODORE VC 20, TANDY TRS 80 und TEXAS INSTRUMENTS T1 99/4. Auch der APPLE II und verschiedene Sampling-Chips, die sich ab Mitte der 1980er Jahren in einer Reihe von Arcade-Spielautomaten finden, werden im Rahmen dieser Arbeit ausgeklammert, da ihre Verbreitung gegenüber der hier vorgestellten vergleichsweise gering ist und sich in ihnen ähnliche Verfahren der Klangerzeugung finden lassen.

Spielkonsolen und 8 Bit Arcade-Boards¹⁷ stellen für die Wiedergabe von Grafik und Sound optimierte Computer dar und werden daher mit speziellen Bauteilen für diese Zwecke ausgestattet. Der zur Verfügung stehende Arbeitsspeicher bleibt aufgrund hoher Kosten während des gesamten Untersuchungszeitraums limitiert. Die Entwicklung von Soundchips ist daher im Hinblick auf Kosten- und Speichereffizienz geschehen, musikalische Kriterien elektronischer Klangerzeugung wie Klangqualität (bzw. Güte der resultierenden

¹⁴ Die Aufteilung in die drei Zeitabschnitte erfolgt bei Collins nach technikhistorischen Aspekten und ist aus diesem Grund auch für die vorliegende Arbeit von Interesse. Andere Geschichtsschreibungen für Computerspielmusik wie z.B. von Brandon 2003, 197ff teilen die Entwicklung nach anderen Gesichtspunkten und damit auch in andere Zeiträume ein.

¹⁵ Lediglich der japanische Hersteller Nintendo verzichtet bei dem 1996 erschienenen NINTENDO 64 auf die Implementierung eines CD-basierten Speichermediums und zieht die in Bezug auf ihre Speicherkapazität eingeschränkten Festspeicher-Modulen vor.

¹⁶ Bei Redbook Audio wird ein unkomprimierter Audio-Track von CD wiedergegeben. Beim *Streaming* wird ein digitaler Datenstrom von der CD gelesen und für das Spielprogramm bereitgestellt (*streamed* Audio). Gestreamte Musikdaten können somit auch in komprimierten Formaten vorliegen, während Redbook-Audio zwar die Spiel-Engine entlastet, dafür aber mitunter viel Platz auf der CD verbraucht. (vgl. Boer 2003: 29, 575f)

¹⁷ Als „(Arcade-)Boards“ bezeichnet man die seit der Einführung des JAMMA-Standards (1986) auswechselbaren Hauptplatinen von Arcade-Spielautomaten.

Wellenformen) und die Bereitstellung verschiedener Klangfarben wurden bei der Entwicklung der ersten Soundchips daher kaum berücksichtigt. Weitere Möglichkeiten der Klangmanipulation durch LFOs, Filter oder Hüllkurven sind ebenso selten implementiert und werden teilweise durch Software verwirklicht. Dies wird bei den Beschreibungen der einzelnen Chips sowie den Strategien der Gestaltung näher ausgeführt.

Die neuen Verfahren der Klangerzeugung der sog. 16 Bit-Ära¹⁸ stellen FM-Synthese und Sampling dar. Ihre Wirkung auf die Ausgestaltung von Computerspielmusik ist jedoch weitgehend gering, da mit ihnen kompositorisch fortgeführt wird, was sich in der Zeit der 8 Bit-Plattformen an gestalterischen Grundsätzen entwickelt hat. Auf dem IBM-kompatiblen PC wird von Anfang der 1980er Jahre an, bis zu dem Aufkommen der ersten Soundkarten (vgl. Kapitel 2.7) vor allem der PC-Speaker zur Klang-, weniger zur Musikwiedergabe genutzt. Die von Collins als erste Phase gekennzeichnete Periode der frühen Soundchips geht mit Ausnahme des PC_{JR} (Kapitel 2.7.3) ohne Einfluss an dieser Plattform vorüber, auch weil Computerspiele in dieser Zeit weniger am PC denn auf Heimcomputern entwickelt und gespielt werden, was nicht zuletzt an den hohen Kosten der Hardware liegt. Die FM-Synthese wird ab 1987 mit der AD LIB-Soundkarte zum Standard für Computerspielmusik auf dem PC.

In der folgenden Vorstellung der verschiedenen Soundchips liefern kleine Übersichts-Tabellen grundlegende Informationen über Speicherplatz und Ausstattung der jeweiligen Hardwareplattformen. Diese bieten eine Orientierung, in welcher Umgebung der jeweilige Soundchip integriert ist. Eine Ausnahme stellen Arcade-Automaten dar, deren technische Details u.U. von Modell zu Modell differieren und daher nicht einzeln genannt werden¹⁹. Es muß darauf hingewiesen werden, dass die technischen Daten der jeweiligen Hardware keine direkten Rückschlüsse auf ihre Leistungsfähigkeit im Vergleich zu anderen zulassen, da sich diese fernab der quantifizierbaren Ebene in vielen anderen Punkten unterscheiden.

Sie beziehen sich grundsätzlich auf die Grundausstattung der jeweiligen Plattform und basieren auf dem umfassenden Übersichtswerk „Gameplan: Heimcomputer und Spielkonsolen 1972-2005“ des deutschen Publizisten Winnie Forster (2005: 202ff).

¹⁸ Diese beginnt ab 1985 mit den 68000er Heimcomputern COMMODORE AMIGA, ATARI ST und APPLE MACINTOSH . Die 16 Bit-Ära der Spielkonsolen wird von den Plattformen SEGA MEGA DRIVE (1988), NINTENDO SUPER NES (1990) und SNK NEO GEO (1990) markiert, welche Mitte der 1990er Jahre von den 32-Bit Konsolen wie SEGA SATURN (1994) und SONY PLAYSTATION (1994) abgelöst werden (vgl. Forster 2005: 144ff). Das MATTEL INTELLIVISION (1979) basiert bereits auf einem 16 Bit-Prozessor, ist jedoch von den Möglichkeiten der Grafik- und Soundwiedergabe am ehesten mit den anderen 8 Bit-Konsolen vergleichbar. Das NEC PC-ENGINE (1987) besitzt einen 16 Bit-Grafikchip, jedoch weiterhin eine 8 Bit-CPU.

¹⁹ In Arcade-Automaten werden bis Mitte der 1980er Jahre dieselben PSGs (Pokey, SN 76489, AY-Chips) benutzt, wie in Spielkonsolen und Heimcomputern, später dann vor allem FM-Chips der Firma YAMAHA, die in Kapitel 2.5.2 vorgestellt werden.

2.1 Die Frühzeit: Computerspiel-Sound ohne Soundchips

Die ersten Spielautomaten mit eigenen Soundeffekten enthalten keine Soundchips. Der erste erfolgreich kommerzialisierte Spielautomat, PONG (ATARI, 1972) erzeugt zwar Geräusche, diese sind jedoch analogen Ursprungs, da sie auf Spannungsspitzen der Schaltkreise basieren. Der tiefste von ihnen ist verstärkt und hörbar gemacht als „Pong-Sound“ (♩001) bekannt geworden. Al Acorn, der bei ATARI für PONG verantwortliche Ingenieur bemerkt:

„As in its predecessor, Computer Space, there were separate circuit boards for each of the game's functions – one for the paddles, one for the ball, and one for the scoring. The image seen on-screen was a reflection of the shifting ON/OFF patterns in the circuitry, drawn with a pulsing raster scan. As the patterns changed, they created a shift in voltage that, in turn, generated wave forms that – when amplified – would make the most fantastic sound. 'Pong.' It was perfect“ (zit. n. Burnham 2001: 87, Hervorh. i. O.).

Andere Spielautomaten dieser Frühzeit²⁰ waren mit eigens angefertigten, analogen Soundgeneratoren versehen. SPACE INVADERS (TAITO, 1978) besitzt neben Soundeffekten als erster Automat einen durchgängigen Soundtrack. Dieser besteht zwar nur aus einigen sich wiederholenden Bass-Noten, er markiert in seiner Abhängigkeit vom Vorankommen des Spielers jedoch eine frühe Integration von sog. adaptiver Musik²¹ in Computerspielen.

„Zu Beginn eines 'Space Invaders'-Spiels, wenn die ersten 55 [...] Bösen ihre Stellung am oberen Bildschirmrand einnehmen, gibt der Spielautomat einen sehr tiefen, sehr langsamen und noch recht leisen Bassrhythmus von sich. Tief genug, dass man den Boden ein wenig zittern spürt. Während die Außerirdischen vorrücken, wird der Bass lauter und schneller. Sind sie beim Spieler angekommen, ist der Rhythmus etwa so schnell wie sein Herzschlag“ (Lischka 2002: 48, Hervorh. i. O.).

Eine Veränderung der eingebauten Klänge ist in dieser Zeit nicht vorgesehen, eine Neu-Programmierung wäre nur durch Eingriffe in die Schaltkreise selbst möglich gewesen (vgl. Brandon 2003: 197).

²⁰ Für den Zeitraum von 1972 bis 1980 erwähnt die amerikanische Autorin Van Burnham gerade einmal rund 50 Spielautomaten als historisch relevant (vgl. Burnham 2001: 11). Insgesamt dürften es höchstens 200 gewesen sein (vgl. Forster 2002: 7), in Europa sind hiervon weitaus weniger erschienen. Für ihren Klang werden nur wenige hervorgehoben: Die Spiele RALLY (FOR-PLAY, 1973), QWAK (ATARI, 1974), SPEED RACE (TAITO, 1974), MANEATER (PSE, 1975) und GUNFIGHT (MIDWAY, 1975) werden aufgrund der „realistischen“ Soundeffekte genannt (Burnham 2001: 95-123). Musik ist zu dieser Zeit noch nicht implementiert.

²¹ Der Begriff „Adaptive Audio“ wird Brandon (2003: 197ff) zufolge benutzt, um den irreführenden Begriff von „Interactive Audio“ abzulösen. Nach Brandon ist in Computerspielen das Audio-Material bzw. die Musik nicht wirklich interaktiv, sie passt sich vielmehr den Handlungen des Spielers an, welche in Interaktion mit der digitalen Maschine über ein Interface abgefragt werden. Daher stellt „adaptiv“ einen passenderen Ausdruck für die nicht-statische klangliche Untermalung von Computer- und Videospielen dar. Die Diskussion um die verschiedenen Begriffe wird in Kapitel 3.1.2 resümiert.

Zur Zeit der erfolgreichen Einführung von ATARIS Spielautomaten PONG in den Vereinigten Staaten existiert bereits die erste für den eigenen Fernseher konzipierte Heimkonsole, das MAGNAVOX ODYSSEY (1972). Dieses basiert jedoch nicht auf einem Computer, sondern auf analogen Schaltkreisen und besitzt keine Klangerzeugung. Bis 1975 bleiben die Verkäufe mit geschätzten 200.000 Exemplaren weit unter den Erwartungen (vgl. Burnham 2001: 82), Lizenznehmern des ODYSSEY in Japan und Europa ist ebenfalls kein nennenswerter Erfolg beschert. 1975 veröffentlicht ATARI mit HOME PONG eine weitaus erfolgreichere Heimversion des berühmten Tennisspiels. In den Folgejahren erscheinen eine Vielzahl der sog. „Telespiele“, die i.d.R. unterschiedliche Varianten von PONG und auch Spielkonzepte des ODYSSEY, wie Tontaubenschießen enthalten²². All diesen ist gemein: Sie enthalten neben sog. Piezo-Beepern keine Hardware zur Klangerzeugung, Musik wird auf ihnen nicht wiedergegeben.

2.2 PSG-Soundchips (1977-1982)

Die frühen PSG-Soundchips, welche die kosten- und zeitintensive Praxis von speziellen Eigenentwicklungen analoger Sound-Schaltkreise für ein bestimmtes Spiel ablösen, erscheinen mit Ausnahme des bereits 1977 veröffentlichten Stella / TIA-Chips des ATARI VCS erst 1979 (ATARI Pokey, AY-3891X) bzw. 1982 (SN 76489 AN) auf dem Markt.

Alle verwenden dasselbe Prinzip der Tonhöhenberechnung: Ein Quotient der Taktfrequenz wird durch den Wert eines bestimmten Registers geteilt, wobei als Ergebnis mehr oder weniger musikalisch verwertbare Skalen einer mit der Registergröße korrelierenden Anzahl von möglichen Tonhöhen entstehen. Die Größe des Tonhöhen-Registers bestimmt somit die Auflösung der resultierenden Skalen, wodurch beim ATARI Stella / TIA gerade einmal 32 verschiedene, sehr grob aufgelöste und häufig nicht im musikalischen Kontext verwertbare Tonhöhen erzeugt werden. Die drei weiteren frühen PSG-Soundchips, der GENERAL INSTRUMENTS AY-3891X, der TEXAS INSTRUMENTS SN 76489 AN und der ATARI Pokey, besitzen als Standardeinstellung 8 Bit, 10 Bit bzw. 12 Bit-Register zur Berechnung der Tonhöhen, was zu 256, 1023 bzw. 4095 verschiedenen Tonhöhen führt. Deren Werte verhalten sich bei dieser Art von Frequenzberechnung umgekehrt proportional zu den Werten des Frequenzregisters.

Soundchips sind somit durch die diskrete Auflösung der Tonhöhen innerhalb der jeweiligen Umgebung *nicht exakt stimmbar*. Außerdem sind die PSG-Soundchips für Spielkonsolen wie auch Heimcomputer und Arcade-Automaten wegen der unterschiedlichen Wechselstrom- und Zeilenwiederholffrequenzen in den verschiedenen Territorien (PAL: 50 Hz, NTSC: 60 Hz) anders gestimmt und getimet: Aufgrund der unterschiedlichen Taktung erklingt in NTSC-Versionen die Computerspiel-Musik je nach technischer Plattform um bis zu einen Halbtonschritt höher und entsprechend schneller als ihr PAL-Pendant²³.

²² Im Rahmen der vorliegenden Arbeit, die sich explizit mit der Entwicklung von *Soundchips* beschäftigt, ist es nicht uninteressant, zu bemerken, dass sich die große Anzahl von Pong-Klonen ab 1975 laut Wolf (2001b: 52) vor allem auf die Entwicklung des GI AY-3-8500 *Grafikchips* zurückführen lässt. Die große Abhängigkeit der Entwicklung der Computertechnik von bestimmten Bauteilen ist indes keine neue Erkenntnis.

²³ Bei heutigen, auf Emulation der jeweiligen Soundchips basierenden Playern wie SIDPLAY lässt sich teilweise die Videonorm zwischen PAL und NTSC umstellen, womit diese Eigenschaft ohne größeren Aufwand erfahrbar gemacht werden kann. (SIDPLAY: <http://www.gsldata.se/c64/spw/sidplayw.html>; 20.07.2005)

Damit sind die frühen Soundchips in gewisser Hinsicht ein Kuriosum, bieten sie als digitale Bausteine im diskreten Computer die Möglichkeit, Tonhöhen eindeutig exakt zu berechnen, dieses jedoch allein auf die technisch determinierte Art und Weise. Eine Stimmung außerhalb ihrer Skalierung ist nicht möglich²⁴.

Neben der Klangerzeugung mittels digitaler Oszillatoren, die auf einfachen Zählern basieren, wurde in den PSGs – mit Ausnahme der Klangerzeugungen des ATARI VCS und des NES – bereits eine Technik angewendet, um kurze, 4 Bit-aufgelöste Samples wiedergeben zu können. Diese beruht auf der schnellen Modulation des Lautstärke-Registers zwischen Maximum und Minimum und wurde ab Mitte der 1980er Jahre verwendet.

Die nun folgende Vorstellung der technischen Daten der frühen Soundchips geschieht unter Berücksichtigung der folgenden Punkte:

- Art und Anzahl der Oszillatoren sowie Tonhöhenberechnung
- Steuerung des Klangverlaufs (Tondauern, Hüllkurven)
- Möglichkeiten der Klangfarbenmanipulation (Filter, PWM, Sweep, LFO etc.)

2.2.1 Der Stella / TIA – Soundchip des Atari VCS

Die Geschichte der Soundchips beginnt mit der Entwicklung des ATARI VIDEO COMPUTER SYSTEMS (ATARI VCS, später: ATARI 2600), welche als erste programmierbare, auf digitalen Bauteilen basierende Heimkonsole 1977 eingeführt wird. Für das VCS gibt es erstmals auf sog. Cartridges (Plastikkassetten mit integriertem ROM-Festspeicher-Modulen) gespeicherte Spiele (in der ersten Version mit einer maximalen Speicherkapazität von 4 kB), womit eine vorhandene Hardware für verschiedene Spiele nutzbar gemacht wird²⁵.

Atari VCS / 2600 (1977)	
CPU, Taktfrequenz	6507, 1,2 MHz
RAM	128 Bytes
Datenträger,	Modul
Kapazität	4kB – 32 kB

Der Stella-Chip ist im ATARI VCS vor allem für Grafik zuständig. Der Chip enthält zur Klangerzeugung zwei unabhängig programmierbare Audio-Schaltkreise, welche über je drei Register kontrolliert werden²⁶:

²⁴ Eine als ungerade wahrnehmbare Stimmung besitzen dadurch jedoch nur die ATARI-Chips Pokey und TIA. Da Soundchips ihrem Alltagsgebrauch als Computerspiel-Klangerzeuger entsprechend eher selten mit anderen Instrumenten zusammen erklingen, die z.B. wohltemperiert gestimmt sind, wird diese Eigenart bei den eine größere Auflösung bietenden Soundchips selten wahrgenommen: Die mathematisch-physikalischen Grundlagen der Musik (z.B. dass Intervalle als mathematische Verhältnissen berechnet werden können) bleiben erhalten.

²⁵ Bereits das erwähnte MAGNAVOX ODYSSEY benutzt verschiedene Steckkarten, diese stellen jedoch keine programmierten und gespeicherten Spiele dar, wie die auf Festspeicher basierenden VCS-Cartridges, sondern variieren lediglich die internen analogen Schaltkreise.

²⁶ Die recherchierten Daten über die Klangerzeugung des VCS entstammen keiner offiziellen Dokumentation des TIA-Chips, können jedoch als verlässlich eingestuft werden, da die verwendeten Quellen als Grundlage für die Programmierung von Emulatoren und auf die Klangerzeugung zugreifende Programme dienen.

- AUDC0 und AUDC1 sind 4-Bit-Register, die Zugriff auf 16 verschiedene Klangfarben bieten, welche Mischungsverhältnisse der Ausgänge von Pulswellengenerator und Rauschgenerator darstellen. Darüber hinaus stehen sie für eine jeweils andere Division des Frequenzwerts des entsprechenden Kanals, womit pro Klangfarbe andere Tonhöhen produziert werden²⁷. Laut Slocum (2003) sind nur 8 der 16 verschiedenen Einstellungen ihrer Klangfarbe her voneinander zu unterscheiden und damit zur musikalischen Verwendung geeignet.
- AUDF0 und AUDF1 sind 5 Bit-Register, durch deren Werte die am Chip anliegende Eingangstaktfrequenz dividiert wird, um eine von 32 möglichen Tonhöhen auszuwählen. Die resultierenden Tonhöhen differieren aufgrund ihrer Abhängigkeit von Eingangstaktfrequenz und Zeilenwiederholffrequenz je nach Klangfarbenregister-Einstellung und Territorium (PAL / NTSC) des verwendeten Systems.
- AUDV0 und AUDV1 kontrollieren die Lautstärke, abermals als 4 Bit-Register, also über 16 verschiedene Werte.

Weitere hardwareseitige Möglichkeiten zur Regelung der Klangparameter (wie Tondauern oder Klangfarbenmanipulation) existieren nicht. Das Problem der mangelhaften Stimbarkeit sowie die Limitiertheit aller auf dem VCS zur Verfügung stehenden Noten in Abhängigkeit von den verschiedenen Parametern arbeitet Stolberg (2000) heraus²⁸.

Der Stella / TIA unter musikalischen Gesichtspunkten

Slocum (2003) hat aus den Tabellen Stolbergs zwei Sets von für musikalische Zwecke verwendbaren Setups erstellt, bei denen auffällt, dass es unmöglich ist, für den Stella Soundchip Musik zu programmieren, die den uns bekannten Stimmungssystemen entspricht. Das erste Set enthält mehr verwendbare Bassnoten und ist exemplarisch als Auszug abgebildet (siehe Abb.1). Die Namen über den Tabellen (Bass, Pitfall, Square, Lead) kennzeichnen die von Slocum betitelten Wellenformen samt AUDC-Wert. Die unterlegten Spalten NTSC und PAL kennzeichnen die Verstimmung in Cents gegenüber der angegebenen Note in der ersten Spalte. Nicht sinnvolle (wie außerhalb des menschlichen Hörbereichs oder sehr nah beieinander liegende) Werte wurden ausgelassen.

Wie schon erwähnt, kann der Stella / TIA -Chip nur zwei Stimmen zur selben Zeit wiedergeben. Zwischen den verschiedenen Klangfarben muss im Falle des intendierten Wechsels jedes Mal mittels der Register AUDC0 und AUDC1 umgeschaltet werden. Slocum (2003: o.S.) geht auf verschie-

6 001 Bass				
7 010 Pitfall				
#	Note	NTSC	PAL	Bass Code
0	b5	+45	+32	[00100000]
1	b4	+45	+32	[00100001]
2	e4	+43	+30	[00100010]
3	h3	+45	+32	[00100011]
4	g3	+58	+45	[00100100]
5	e3	+43	+30	[00100101]
7	b2	+42	+32	[00100111]
8	a2	+42	+28	[00101000]
9	g2	+59	+45	[00101001]
11	e2	+43	+31	[00101011]
14	c2	+56	+44	[00101110]
15	b1	+46	+32	[00101111]
17	a1	+39	+27	[00110001]
19	g1	+59	+45	[00110011]
23	e1	+44	+28	[00110111]
26	d1	+41	+27	[00111010]
29	c1	+58	+42	[00111101]
31	h0	+44	+33	[00111111]

4 000 Square				
#	Note	NTSC	PAL	Driver Code
10	f6	+39	+25	[00001010]
12	d6	+49	+36	[00001100]
21	f5	+39	+26	[00010101]
22	e5	+62	+48	[00010110]
25	d5	+49	+36	[00011001]
28	c5	+60	+47	[00011100]
30	b4	+45	+32	[00011110]

12 101 Lead				
#	Note	NTSC	PAL	Driver Code
10	a#4	+39	+23	[10101010]
12	g4	+48	+34	[10101100]
21	a#3	+37	+24	[10110101]
22	a3	+60	+47	[10110110]
25	g3	+47	+34	[10111001]
28	f3	+59	+45	[10111100]
30	e3	+43	+30	[10111110]

Abbildung 1: Auszug des ersten Instrumenten-Sets von Slocum (2003), es fehlen das Saw-Instrument und der Pitfall-Binärcode.

²⁷ Eine genaue Auflistung der 16 verschiedenen Modi findet sich bei Israel (1993).

²⁸ Stolberg (2000) führt unter anderem aus, dass der 114te Anteil der Pixelclock oder der Systemclock als Basis der Frequenzdivision dienen kann. Bei NTSC-Geräten (262 Zeilen bei 60 Hz) entsprechen diese 31440 Hz und 10480 Hz, bei PAL-Geräten (312 Zeilen bei 50 Hz) 31200 Hz bzw. 10400 Hz. Die PAL-Version des VCS ist entsprechend tiefer gestimmt.

dene Effekte ein, mit denen trotz der limitierten Technologie musikalische Ideen umgesetzt werden können. So lassen sich quasi-Akkorde durch schnelle Arpeggien erzeugen, er nennt außerdem mögliche Percussion-Sounds sowie die Programmierung von Schlagzeug und Bass auf nur einem Kanal²⁹.

An dieser Stelle muss jedoch differenziert werden zwischen der zeitgenössischen Computerspielmusik auf dem VCS und der im Zuge des Phänomens der Chiptunes ab der zweiten Hälfte der 1990er Jahre komponierten populären Soundchip-Musik: In frühen Spielen können aufwendige Klangfarben-Schaltungen oder gar Percussion-Spuren nicht benutzt werden, auch aufgrund des limitierten Speicherplatzes³⁰ auf der Cart-ridge. Erst späte Spiele wie *ACID DROP* (SALU LTD, 1992; ♪002) nutzen verschiedene Klangfarben für die Melodie, um die in einer Klangfarbe nicht vorhandenen Noten von einer anderen spielen zu lassen.

Computerspielmusik für das VCS wurde in Assemblercode geschrieben und danach in die Spielen integriert. Collins (2004b) führt eine musikwissenschaftliche Inhalts- und Formanalyse dieser durch. Sie stellt fest, dass nur etwa jedes fünfte für das VCS erschienene Spiel überhaupt Musik enthält (Collins: 2004b: 4) und dass in denen, die sie untersucht hat, eine große Anzahl von kleinen Sekunden vorkommt. Kleine Sekunden werden seit jeher mit Affekten wie „Unheil“, „Hoffnungslosigkeit“ und „Verzweiflung“ assoziiert. Obwohl dies Intention der jeweiligen Komponisten gewesen sein kann, wird ihr gehäuftes Auftauchen von Collins in letzter Konsequenz auf die technisch bedingt stark limitierten Skalen zurückgeführt (ebd.: 9).

2.2.2 Atari Pokey

Der ATARI CO 12294 Pokey (**P**otentiometer and **K**eyboard Integrated Circuit) ist einer von drei speziell für die 1979 erscheinende ATARI Heimcomputer-Serie 400/800 von einem Team unter Leitung von Jay Miner bei ATARI entwickelten „Custom Chips“. Der Pokey ist neben der Abfrage des Keyboards, der Verwaltung von seriellen Daten und Potentiometer-Eingängen auch für die Klangerzeugung zuständig (vgl. Forster 2005: 34ff). Er stellt den historisch ersten, unter kompositorischen Gesichtspunkten ernst zu nehmenden Soundchip dar, verfügt er doch über eine vierstimmige Klangerzeugung (Rechteckwellen und Rauschen), die seinen flexiblen Einsatz als Klangerzeuger erlaubt. Der Soundchip wird in ATARI-Spielautomaten wie z.B. *BATTLEZONE*, *CENTIPEDE* oder *TEMPEST* (ATARI, 1980, 1980, 1981) sowie der Nachfolgekonsolle des VCS, dem ATARI 5200 (ab 1982) eingesetzt, in welchen er vor allem für Soundeffekte zuständig ist. Die Klangerzeugung klingt besonders in den Basslagen wenig überzeugend, wie Warhol im Interview bemerkt (Warhol 2005) und ein Hörtest verschiedener Spiele im Emulator bestätigt. Die nachfolgenden Details entstammen dem Pokey Data Manual, in dem der Chip von ATARI dokumentiert wurde, wenn auch in kryptischer Form. Laut Levy (1994: 316ff) ist

	Atari 400 / 800 (1979)	Atari 5200 (1982)
CPU; Taktfrequenz	6502A; 1,79 MHz	6502; 1,79 MHz
RAM	8 bis 48 kB	2 kB
ROM	10 kB	n.b.
Datenträger	Modul, Diskette, Kasette	Modul
Kapazität	2 kB - ~100 kB	bis 32 kB

²⁹ Diese Strategien gestalterischer Praxis werden generell für Soundchip-Musik benutzt, um trotz der begrenzten Polyphonie musikalische Ideen umzusetzen (vgl. Kapitel 3.2.1).

³⁰ In die anfangs zur Verfügung stehenden 4kB musste ein gesamtes Spiel passen, was das VCS zu einer großen Herausforderung für Assembler-Programmierung machte. Tod Frye benutzte in der Automatenumsetzung von *ASTEROIDS* (ATARI, 1981) erstmals die von Larry Wagner erfundene Technik des „bank switching“, die es ermöglicht, erst 8kB große, später auch noch größere Spiele zu entwickeln (vgl. Herman 2001: 151).

der ATARI 800 gegenüber anderen Heimcomputern der selben Zeit äußerst mangelhaft dokumentiert, vermutlich auch weil ATARI als Hardware- und Software-Hersteller viele der möglichen Features (z.B. mehr als die acht unter BASIC darstellbaren Farben) aus dem Gedanken heraus, die eigens entwickelte Software solle konkurrenzlos bleiben, nicht dokumentierte³¹.

Der Pokey verfügt über vier Rechteckwellen generierende Tonkanäle, deren Tonhöhen via der Werte aus den 8 Bit-Registern AUDF1-4 errechnet werden³². Mit Hilfe des „Audio Control“-Register AUDCTL lassen sich globale Einstellungen für den Klangeinsatz steuern. Das Register bietet die Wahl der Periode des Rauschgenerators und alternative Einstellungen der Eingangstaktfrequenz³³. Des Weiteren lassen sich die Kanäle 2 und 4 mit den Kanälen 1 und 3 koppeln, wodurch die verbundenen AUDF Register wie ein 16 Bit-Register behandelt werden. Für den nun mehr zusammengefassten Tonkanal wird eine höhere Auflösung der verfügbaren Töne möglich, was z.B. für Glissandi eine reizvolle Option ist.

Der Pokey besitzt also verschiedene Modi der Polyphonie und Auflösung, unter denen je nach Einsatz gewählt werden kann: Es stehen entweder vier 8 Bit-, zwei 16 Bit- oder ein 16 Bit- und zwei 8 Bit-aufgelöste Kanäle zur Verfügung. Als Eingangstaktfrequenz kann zwischen 15 kHz, 64 kHz und 1,79 MHz gewählt werden³⁴. Zur weiteren Klangmanipulation besteht die Möglichkeit, die Klangkanäle 1 und 2 mit einem High-pass-Filter zu versehen, wobei hierfür die Frequenzen von Kanal 3 und 4 als Filtereckfrequenzen 'geborgt' werden.

Die „Audio Channel Control“-Register AUDC1-4 steuern sowohl Lautstärke des Kanals als auch Art und Intensität des Rauschens. Die Lautstärke kann in 16 Schritten geregelt werden, darüber hinaus stehen acht verschiedene Mischungs- und Stimmungsverhältnisse von Rauschen und Tonkanal zur Verfügung, womit eine bedingte Auswahl der Klangfarbe möglich ist. Weitere Editiermöglichkeiten der Klangparameter durch LFOs oder Hüllkurven sind hardwareseitig nicht vorgesehen.

Die grobe Auflösung des Chips in nur 256 diskrete Tonhöhen führt zu einer für geübte Ohren wahrnehmbaren, 'schiefen' Stimmung, die aufgrund der wählbaren Eingangstaktfrequenzen zwar abgemildert wird, jedoch selbst im 16 Bit-Modus erhalten bleibt, da die potenzierte Auflösung allein zu einer Mehrzahl verwendbarer Noten *unterhalb* des 8 Bit-Tonumfangs führt. So steht nur bei der 1,79 MHz-Eingangstaktfrequenz ein deutliches Plus an musikalisch verwertbaren Noten mit einer Grundfrequenz von unter ~3500 Hz zur Verfügung.

³¹ Levy beschreibt den Prozess des „Reverse Engineering“, welches es dem jungen Programmierer John Harris erlaubte, viele der undokumentierten Funktionen des ATARI 800 in seinen Spielen zu nutzen (vgl. Levy 1994: 317f).

³² Die Formel $F_{out}=(F_{in})/(2*N)$ erlaubt die Errechnung der klingenden Frequenz aus der Eingangstaktfrequenz und dem 8 Bit Wert aus AUDFx, wobei N dem in das jeweilige AUDF Register geschriebenen Wert zwischen 0 und 255 addiert mit 1 entspricht und damit zwischen 1 und 256 liegen kann.

³³ Der Standardwert für die Eingangstaktfrequenz beträgt 64 kHz, geringe Abweichungen gegenüber diesem Idealwert nennt das „Pokey Data Manual“ für NTSC-Geräte. Die Abweichung liegt mit unter 0,1% jedoch im tolerierbaren Bereich. Über die Register-Bits D5 und D6 wird die Frequenz der Tonkanäle 1 und 3 mit 1,79 MHz statt 64 kHz errechnet. Sofern das Bit D0 gesetzt ist, wird die Frequenz von 64 kHz durch 15kHz ausgetauscht, was jeweils zu anderen Skalen führt.

³⁴ Der Pokey erzeugt je nach Taktfrequenz (F_{in}) im 8 Bit-Modus 256 Frequenzen zwischen 29 Hz und 7500 Hz ($F_{in}=15$ kHz), 125 Hz und 16 kHz ($F_{in} = 64$ kHz) bzw. 3495 Hz und 89,5 kHz ($F_{in} = 1,79$ MHz). Im 16 Bit Modus sorgt die o.g. Umrechnungsformel für eine deutlich höhere Anzahl von Tönen *unterhalb* der tiefsten Frequenzen des zur jeweiligen Eingangstaktfrequenz gehörenden Tonumfangs.

Sowohl die unsaubere Stimmung, als auch das bereits erwähnte 4 Bit-Sampling durch Modulation des Lautstärke-Registers sind in der Titelmelodie von GHOSTBUSTERS (ACTIVISION, 1984; ♪053b) für ATARI 400 / 800 zu hören.

In Spielautomaten werden mehrere Pokeys parallel eingesetzt, um Soundeffekte wie Musik gleichzeitig und in Stereo wiedergeben zu können. Ein bekanntes Beispiel ist der auf Vektorgrafik basierende STAR WARS-Automat (ATARI, 1983) mit insgesamt vier Pokeys (♪003), welche neben der Musik auch die Soundeffekte generieren und von dem Sprachsynthesechip TI 5220 unterstützt werden (♪004) (vgl. Brandon & Fuller 1999). Der Pokey wird nach 1987³⁵ nicht mehr eingesetzt, ATARI-Spielautomaten ab MARBLE MADNESS (ATARI, 1984) mit FM-Chips der Firma YAMAHA versehen. Der mit dem Soundchip YM 2149 (vgl. nächstes Kapitel) ausgestattete ATARI ST löst ab 1985 die ATARI 8 Bit-Heimcomputer ab.

³⁵ In diesem Jahr erscheint die letzte Modellrevision der ATARI 800-Nachfolger, der ATARI XE GS (vgl. Forster 2005: 38)

2.2.3 General Instruments AY-3-891x / Yamaha YM 2149 SSG

Das MATTEL INTELLIVISION erscheint 1979 als direkter Konkurrent zum ATARI VCS. Neben besserer Grafikfähigkeiten ist das INTELLIVISION dem ATARI auch in Fragen der Klangerzeugung weit überlegen. In ihm kommt erstmals der GENERAL INSTRUMENTS AY-3-8914 Soundchip zum Einsatz. Dieser und die von

	Mattel Intellivision (1979)	GCE Vectrex (1982)	MSX (1982)	Amstrad CPC 464 (1984)	Atari ST (1985)
CPU; Taktfrequenz	GI CP 1610; 0,9MHz	MC 68A09; 1,6 MHz	Z80A; 3,58 MHz	Z80A; 4MHz	MC 68000; 7,9 MHz
RAM	2 bis 3 kB + 512 Byte VRAM	512 Byte	8 bis 64kB + 16 kB VRAM	64 kB	256 kB bis 1 MB
ROM	7 kB	-	32 kB	32 kB	192 bis 256 kB
Datenträger	Modul	Modul	Modul, Diskette, Kassette,	Kassette, Diskette	Diskette
Kapazität	bis 16 kB	bis 8 kB	bis 720 kB	360 kB	720 kB

der Klangerzeugung identischen Modelle AY-3-8910, AY-3-8912 und YM 2149 SSG³⁶, werden auch in einer Vielzahl von anderen Heimcomputern, Spielkonsolen und Spielautomaten verbaut³⁷.

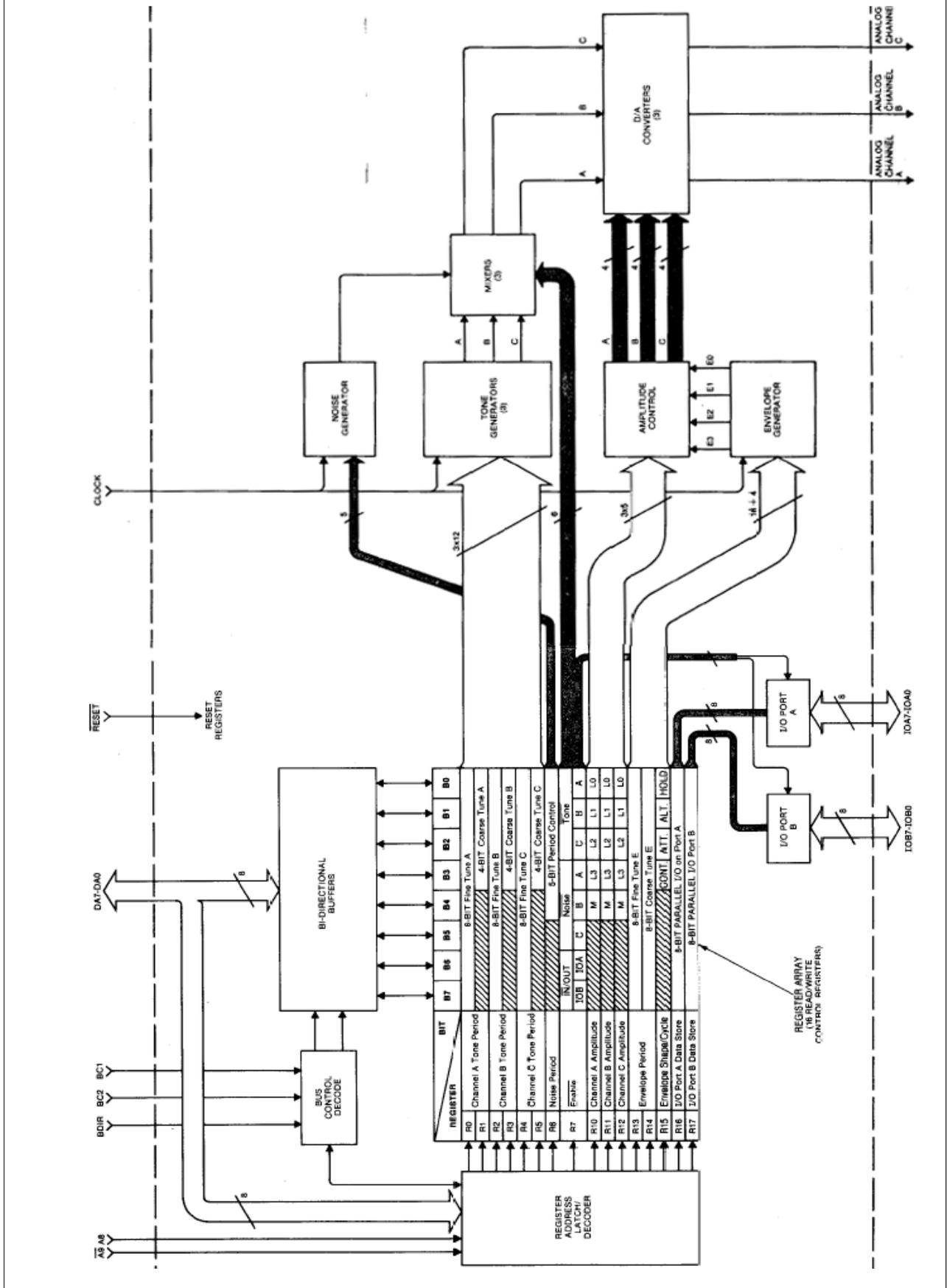
Anhand des Data Manuals³⁸ werden im Folgenden die hardwareseitigen Möglichkeiten der Klangerzeugung vorgestellt. Der AY-Chip verfügt über drei Tongeneratoren, die Rechteckwellen produzieren. Frequenz, Lautstärke und Hüllkurve sind pro Kanal regelbar, der Rauschgenerator erzeugt eine frequenzmodulierte Rechteckwelle, deren Pulsweite über einen Zufallsgenerator moduliert wird. Die Ausgänge der drei Tongeneratoren und des Rauschgenerators werden über den mittels einer Logikmatrix (Register 7) gesteuerten Mixer miteinander gemischt. Jedem Kanal kann hier der Ausgang des Rauschgenerators hinzu gemischt werden. Das Blockdiagramm (Abb. 2) bietet einen Überblick über Funktionsweise und Steuerung der Bauteile.

³⁶ Die verschiedenen AY-Baureihen unterscheiden sich v. a. durch Art und Anzahl der verfügbaren In/Out-Ports. Im Folgenden wird der Ausdruck „AY-Chips“ der besseren Lesbarkeit zugute synonym für die von der Klangerzeugung identischen Chips verwendet werden. Der YM 2149 SSG ist ein mit den AY-Chips baugleiches Modell der Firma YAMAHA..

³⁷ AY-Chips wurden außer im MATTEL INTELLIVISION auch in vielen Arcade-Automaten, in Heimcomputern wie dem ZX SPECTRUM 128, dem AMSTRAD bzw. SCHNEIDER CPC sowie in Spielkonsolen wie dem MB VECTREX eingesetzt. Außerdem ist der AY-38910 Teil der MSX-Spezifikationen (vgl. Fußnote 37). Viele japanische Personal Computer der 1980er Jahre wie der SHARP X1 (1982) besitzen ebenfalls einen AY-Chip.

³⁸ Die folgenden Informationen und Abbildungen entstammen dem „AY-3-8910/8912/8913 Programmable Sound Generator Datasheet“; URL: http://bulba.at.kz/AY-3-8910_2.rar (27.10.2004).

Abbildung 2 Blockdiagramm des AY-3-8910



Programmierung des AY-Chips

Zur Steuerung der Klangerzeugung des AY-Chips werden die Register R0-R15 beschrieben. Auf den verschiedenen Plattformen, in denen dieser verbaut worden ist, findet seine Programmierung in verschiedenen Programmiersprachen statt. Für Computerspiele erfolgt die Programmierung aus Gründen des Timings maschinennah, also meist in Assembler. Mit dem Einsatz des Chips in Heimcomputern (wie in Modellen, die dem MSX-Standard³⁹ folgen) wird die Programmierung

REGISTER		BIT								
		B7	B6	B5	B4	B3	B2	B1	B0	
R0	Channel A Tone Period	8-BIT Fine Tune A								
R1		/				4-BIT Coarse Tune A				
R2	Channel B Tone Period	8-BIT Fine Tune B								
R3		/				4-BIT Coarse Tune B				
R4	Channel C Tone Period	8-BIT Fine Tune C								
R5		/				4-BIT Coarse Tune C				
R6	Noise Period	/				5-BIT Period Control				
R7	Enable	IN/OUT		Noise			Tone			
		IOB	IOA	C	B	A	C	B	A	
R10	Channel A Amplitude	/				M	L3	L2	L1	L0
R11	Channel B Amplitude	/				M	L3	L2	L1	L0
R12	Channel C Amplitude	/				M	L3	L2	L1	L0
R13	Envelope Period	8-BIT Fine Tune E								
R14		8-BIT Coarse Tune E								
R15	Envelope Shape/Cycle	/				CONT	ATT	ALT	HOLD	
R16	I/O Port A Data Store	8-BIT PARALLEL I/O on Port A								
R17	I/O Port B Data Store	8-BIT PARALLEL I/O Port B								

Abbildung 3 Das Register-Array des AY-3-8910

auch von benutzerfreundlichen Sprachen wie BASIC unterstützt⁴⁰. Im Folgenden wird allein die plattformunabhängige, hardwareseitige Funktionsweise des Soundchips vorgestellt, wie sie auf Basis des Register-Arrays (Abb. 3) geschieht.

Die Register 0-5 des AY-Chips dienen zur Errechnung der Tonhöhen: Jeweils 12 Bit Informationen bilden die Tonhöhe der drei Tongeneratoren ab, wobei hierfür wie auch bei den anderen auf Frequenzdivision basierenden Soundchips eine Teilmenge aus Eingangstaktfrequenz und Tonhöhen-Wert gebildet wird⁴¹. Die resultierenden und damit für Soundeffekte und Musik zur Verfügung stehenden 4095 Tonhöhen⁴² stehen immer in Relation zur Eingangstaktfrequenz, die je nach Umgebung des Chips innerhalb der digitalen Maschine variieren kann. Das Data Manual des YM 2149 spricht von einem möglichen Wertebereich der Eingangstaktfrequenz von 1-4 MHz.

³⁹ Der MSX-Standard (Microsoft Extended Basic) wird von dem japanischen Unternehmer Kai Nishi in Zusammenarbeit mit den bedeutendsten Elektronikfirmen Japans sowie MICROSOFT 1983 festgelegt und kennzeichnet einen Heimcomputer-Standard, der die Interoperabilität zwischen Heimcomputern verschiedener Hersteller garantiert. Die Basis-Spezifikationen umfassen u.a. den Prozessor Z80, wenigstens 8kb RAM sowie den AY-38910. Für den MSX-Standard erscheinen ca. 1400 Spiele, v.a. in Japan ist der Standard populär. Diverse Erweiterungen (MSX 2, MSX 2+, MSX 2 Turbo R) werden bis Anfang der 1990er Jahre entwickelt (vgl. Forster 2005: 74ff).

⁴⁰ Enders stellt die Programmierung des YM 2149 am ATARI ST in den Sprachen GfA-Basic und Megamax-C vor (Enders 1988: 37ff), Höh / Pütz erläutern die Programmierung in MSX-Basic (Höh / Pütz 1986: 229ff).

⁴¹ $f_T = f_{\text{CLOCK}} / 16 * TP_{10}$. Hierbei entspricht f_T der gewünschten Tonfrequenz, f_{CLOCK} der Eingangstaktfrequenz und TP_{10} dem 12 Bit-Wert aus den Registern R0-R5. Die Frequenz des Rauschgenerators (f_N) wird auf die gleiche Art aus dem 5 Bit Wert in Register 6 (NP_{10}) errechnet. Die Gleichung für die Berechnung dieser lautet $f_N = f_{\text{CLOCK}} / 16 * NP_{10}$. Die Frequenzbreite des Rauschgenerators liegt bei einer Eingangstaktfrequenz von 2 MHz zwischen 4kHz und 125 kHz in 31 Schritten.

⁴² $10^{12} = 4096$ Werte, wobei der valide Wertebereich zwischen 00000000001 (Division durch 1) und 11111111111 (Division durch 4095) liegt.

Dem Data Manual des AY-38910 zufolge entspricht 2 MHz dem Standardwert des Chips, für diese ergibt sich ein möglicher Frequenzbereich von 30,5 Hz bis 125 kHz⁴³.

Der Umstand der diskreten Rasterung einer begrenzten Anzahl von möglichen Tonhöhen durch das Frequenzdivisionsverfahren führt bei den AY-Chips zu Abweichungen von der wohltemperierten Skala, wie auszugsweise in Abb. 4 zu sehen. Dennoch zeigt sich das Problem weniger auffällig als bei der extrem limitierten Tonhöhen-Auswahl des ATARI Stella / TIA oder dem 8 Bit-Modus des Pokey. Diese Verstimmungen sind für die Anwendungen des AY-Chips nicht als auffällig zu bewerten.

NOTE	OCTAVE	IDEAL FREQUENCY	ACTUAL FREQUENCY
C	1	32.703	32.898
C#	1	34.648	34.853
D	1	36.708	36.712
D#	1	38.891	38.895
E	1	41.203	41.201
F	1	43.654	43.662
F#	1	46.249	46.243
G	1	48.999	48.997
G#	1	51.913	51.908
A	1	55.000	54.995
A#	1	58.270	58.261
B	1	61.735	61.733
C	2	65.406	65.416
C#	2	69.296	69.307
D	2	73.416	73.399
D#	2	77.782	77.789
E	2	82.406	82.432
F	2	87.308	87.323
F#	2	92.498	92.523
G	2	97.998	98.037
G#	2	103.826	103.663
A	2	110.000	109.991
A#	2	116.540	116.522
B	2	123.470	123.467
C	3	130.812	130.831
C#	3	138.582	138.613
D	3	146.832	146.799
D#	3	155.564	155.578
E	3	164.812	164.743
F	3	174.616	174.510
F#	3	184.996	184.994
G	3	195.996	195.903
G#	3	207.652	207.634
A	3	220.000	220.198
A#	3	233.080	233.043
B	3	246.940	246.933

Abbildung 4 Differenz zwischen wohltemperierter Stimmung und Stimmung des AY-Soundchips bei einer Taktfrequenz von 2 MHz (Auszug)

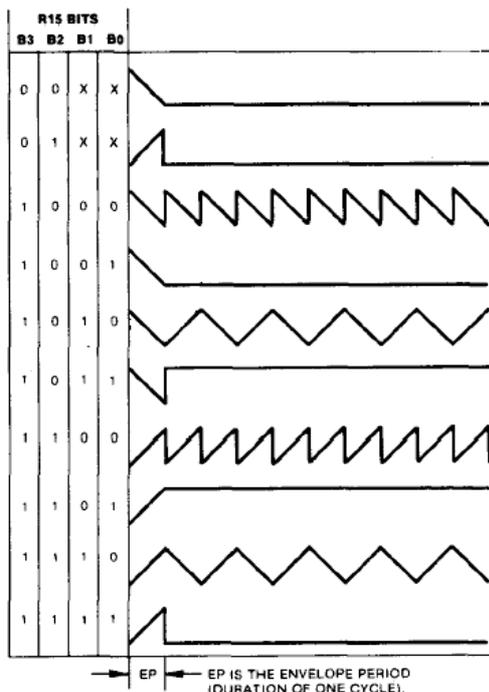


Abbildung 5
Die verschiedenen Hüllkurven-Typen des AY-Chips in Abhängigkeit zu den in R15 aktivierten Bits

Die Amplituden (Lautstärke)-Einstellungen des AY-Chips werden in den Registern R10-R12 vorgenommen. Hierbei bestimmt ein Schalter-Bit den Modus der Lautstärken-Regelung:

Ist das Bit (B4) deaktiviert, hat dies zur Folge, dass der 4 Bit-Wert von B0-B3 als fixes Lautstärken-Niveau für den jeweiligen Tongenerator genommen wird. Ist es gesetzt, wird der jeweilige Kanal mit variabler Lautstärke wiedergegeben: Die Register R13 und R14 bestimmen die Frequenz der Hüllkurve⁴⁴. Die Form der Hüllkurve (Shape / Cycle) nimmt eine von zehn (bzw. acht verschiedenen) Verlaufscharakteristiken an, welche durch das Register R15 bestimmt werden (s. Abb. 5).

Wie Enders (1988: 33) ausführt, kann eine Hüllkurven-Frequenz im hörbaren Bereich (>20 Hz) dazu benutzt werden, die sich wiederholenden Hüllkurven statt für Tremolo-Effekte auch zur Amplituden-Modulation und damit als Klangfarbe zu benutzen. Auf eine weitere Regelung der Lautstärke dieser Effekt-Klänge muss dann jedoch verzichtet werden, da das entsprechende

⁴³ Je geringer der Wert TP₁₀, desto höher ist die erzeugte Tonhöhe. Die Abstände zwischen den einzelnen Tonhöhen nehmen in den höheren Lagen exponentiell zu. So liefern die 10 Werte von 4085 bis 4096 die sehr tiefen, kaum unterscheidbaren Töne von 30,518 bis 30,600 Hz. Die 10 Werte von 1 bis 11 generieren vom Maximum 125 kHz bis 11363,636 Hz sehr weit voneinander entfernt liegende Töne, die jedoch kaum zum Einsatz kommen dürften, da sie am am Rande bzw. außerhalb des menschlichen Hörspektrums liegen.

⁴⁴ Die Hüllkurvenfrequenz (f_E) wird mit der Formel $f_E = f_{\text{CLOCK}} / 256 * EP_{10}$ ausgerechnet. EP₁₀ entspricht hierbei dem in R13 und R14 bestimmten 16 Bit-Wert. Eine Eingangstaktfrequenz von 2 MHz kann somit 65534 Periodendauern von 0,12 Hz bis 7812,5 Hz produzieren, die längste Dauer eines Zyklus entspricht also gut 8 Sekunden.

Schalter-Bit die Lautstärken-Steuerung an den nun zweckentfremdeten Hüllkurven-Generator weist. Der für alle drei Kanäle nur einmal vorhandene Hüllkurven-Generator wird bei der späteren Programmierung von Musik für den AY-Chip häufig durch Subroutinen wie Software-LFOs abgelöst.

Für musikalische Zwecke ließ sich der Hüllkurvengenerator sich verwenden, wie der Anfang der 1980er Jahre bei MATTEL ELECTRONICS beschäftigte Programmierer David Warhol im Interview bemerkt:

„It[der AY-Chip, N.D.] was good for sound effects but not so good for musical instruments. And it only had one envelope that could be applied to any of the three pitched channels, you would have to pick which one you wanted [...]. So for music with different notes going on around at different times you couldn't really use that“ (Warhol 2005).

Weitere Möglichkeiten der Manipulation von Klangparametern finden sich ebenso wie bei den anderen frühen Soundchips auch beim AY-Chip nicht. Aufgrund der gegenüber den direkten Konkurrenten flexiblen Möglichkeiten der Klanggestaltung sowie dem vor allem in den tieferen Lagen als angenehm empfundenen Klang werden AY-Chips jedoch über einen langen Zeitraum hinweg benutzt. Neben dem YM 2149 als AY-kompatibles Modell entwirft Yamaha mit dem YM 2203 und YM 2610 zwei FM-Soundchips, die die komplette Klangerzeugung des AY-Chips enthalten und in einigen japanischen Personal Computern ab Anfang der 80er Jahre (v.a. YM 2203) sowie vielen Arcade-Automaten verbaut werden⁴⁵. In Spielen früher Plattformen wie dem INTELLIVISION, wurde der Soundchip nur selten für durchlaufende Hintergrundmusik verwendet. Ein Beispiel für frühe INTELLIVISION-Musik ist in SHARK! SHARK! zu finden (MATTEL ELECTRONICS, 1982; 🎵005).

Da der Soundchip auch im Heimcomputer ATARI ST Verwendung findet, wird bis Anfang der 1990er Jahre viel Musik für den AY-Chip programmiert, durch die vorhandene Rechenleistung der CPU und entsprechenden Routinen des deutschen Programmierers und Musikers Jochen Hippel wird sogar die Möglichkeit, vierstimmig Samples wiedergeben zu können, mit dem Chip realisiert, wie Chris Hülsbeck im Interview erwähnt:

„Man konnte mit dem Prozessor per Software noch Samples mischen, das hat damals Jochen Hippel [herausbekommen], der sehr aktiv war auf dem Atari ST. Er hat einen 4-Stimmen Mischer geschrieben [...], um Amiga Sound aus dem Atari ST herauszukriegen“ (Hülsbeck 2005a).

Diese Technik der Sample-Wiedergabe geschieht wie bei anderen frühen Soundchips über die ausreichend schnelle Modulation des Lautstärke-Registers.

⁴⁵ Der YM-2203 befindet sich u.a. in Modellen der NEC Heimcomputer-Reihe, wie dem PC-6000, PC-8001, PC-8801, PC-9801 und dem Fujitsu FM77 (vgl. <http://gyusyabu.ddo.jp/MP3/MP3.html>; 15.06.2005). Der YM 2610 findet sich auf vielen TAITO Arcade-Boards ab Mitte der 1980er Jahre sowie im SNK NEO GEO.

Um Beschränkungen zu umgehen und mehrstimmige Musik und Soundeffekte gleichzeitig wiedergeben zu können, werden viele Arcade-Spielautomaten wie ELEVATOR ACTION, 1942, GYRUSS (♩006) und BOMB JACK (♩007)⁴⁶ mit mehreren AY-Chips bestückt. Der AY-Chip kann aufgrund des typischen Klangs und seiner großen Verbreitung als Standard-PSG des ersten Abschnitts von Soundchip-Musik bezeichnet werden.

2.2.4 Texas Instruments SN 76489 AN

Das CBS COLECOVISION kommt 1982 als dritte Spielkonsole der ersten Generation neben dem ATARI VCS und dem MATTEL INTELLIVISION auf den Markt. Es ist dem ATARI wie auch dem INTELLIVISION in grafischer Hinsicht überlegen und enthält ebenso wie das INTELLIVISION einen neuartigen Soundchip, den TEXAS INSTRUMENTS SN 76489 AN, welcher später

	CBS Colcovision (1982)	Sega Master System (1987)
CPU; Taktfrequenz	Z80A; 3,58 MHz	Z80A; 3,58 MHz
RAM	1 kB + 16 kB VRAM	8 kB + 16 kB VRAM
Datenträger	Modul	Modul, MyCard
Kapazität	bis 8 kB	bis 512 kB

zudem als SN 76494 bzw. 76496⁴⁷ in vielen Spielautomaten, im SEGA MASTER SYSTEM, dem SEGA MEGA DRIVE / GENESIS und dem PCJR / TANDY 1000 eingesetzt wird. Der Soundchip ist mit dem AY-Chip vergleichbar, jedoch weniger umfangreich ausgestattet und bietet ein weniger überzeugendes Klangbild⁴⁸. Auch er besitzt drei Rechteckwellen erzeugende Tongeneratoren sowie einen Rauschkanal. Die Art des Rauschen kann zwischen „White Noise“ und „Periodic Noise“ gewählt und vier verschiedenen Frequenzeinstellungen zugewiesen werden⁴⁹. Der Rauschkanal ist anders als bei AY- und Pokey-Chip ein eigenständiger Kanal, die Mischung von Rauschanteil und Pulsweite in einer Stimme ist nicht möglich. Bei der Frequenzdivision im SN 76489 wird die CPU-Taktfrequenz durch den 32-fachen Wert eines 10 Bit-Registers geteilt, was zu 1023 möglichen Tonhöhen führt.

Der Soundchip bietet keine Möglichkeit, hardwareseitig Hüllkurven auf die Kanäle zu legen. Die Verstärkung wird durch eine 4 Bit Matrix gesteuert, wodurch 15 in 2dB-Schritten unterteilte Werte zwischen 0 dB und 28 dB möglich sind, wie im SN 76489 AN Data Manual⁵⁰ beschrieben wird. Anders als beim AY-Chip, bei dem die drei Ausgänge einzeln an Pins des Chips anliegen, enthält der SN 76489 einen internen Mischer, welcher die Signale mischt und als Mono-Kanal ausgibt.

⁴⁶ TAITO, 1983; 4x AY-3-8910; CAPCOM, 1984; 2x AY-3-8910; KONAMI, 1983; 5x AY-3-8910; TEHKAN, 1984; 3x AY-3-8910.

⁴⁷ Das Data Manual des SN 76489 AN erwähnt den SN 76494 N als baugleichen Chip. Der SN 76496 ist von der Klangerzeugung ebenfalls identisch (Maxim 2003), wurde aber erst später verbaut, z.B. im SEGA MEGA DRIVE sowie in vielen Arcade-Automaten. Die folgenden Details gelten auch für diese baugleichen Chips.

⁴⁸ Einen Eindruck der frühen, auf dem Colecovision programmierten Musik ermöglicht die Titelmelodie des Spiels M.A.S.H (FOX VIDEO GAMES, 1982; ♩008)

⁴⁹ Mögliche Einstellungen für die Rauschfrequenz sind laut Data Manual : N/512, N/1024 oder N/2048 (N=Eingangstaktfrequenz). Optional kann dem Rauschen auch die Frequenz des dritten Tongenerators zugewiesen werden.

⁵⁰ „The Engineering Staff of TEXAS INSTRUMENTS: SN 76489 AN“ URL: <http://www.funet.fi/pub/msx/mirrors/msx2.com/vortexion/sn76489.zip> (14.03.2005)

Der SN-Soundchip bietet die bereits in Spielen benutzte Möglichkeit, PCM- oder PWM-kodierte Samples abzuspielen. Hierfür wird der für die Frequenzdivision verantwortliche Wert aller drei Soundkanäle auf 0 gesetzt (woraufhin die Kanäle eine kontinuierliche Spannung abgeben), um danach über schnelles, periodisches An- und Ausschalten der Lautstärke zwischen 0xf (maximale Lautstärke) und 0x0 (Lautstärke aus) den gewünschten Klang auszugeben (vgl. Maxim 2003). Dieses Feature wird von einigen Spielen für das MASTER SYSTEM genutzt, um digitalisierte Klänge wiederzugeben, wie SHOOTING GALLERY (SEGA, 1987) oder ALEX KIDD: THE LOST STARS (SEGA, 1989). Musikalische Anwendungen dieser Technik sind nicht dokumentiert. Laut Maxim (ebd.) ist die Lautstärke und Klangqualität von PWM-kodierten Samples größer, da für das PCM-Verfahren der Ausgangssound selbst als 4 Bit-Sample vorliegen muss, was von vornherein keine zufriedenstellende Qualität verspricht. Der SN-Soundchip ist in Bezug auf die Möglichkeiten der Klangerzeugung z.B. gegenüber dem AY-Chip stark beschnitten. Auch für die populäre Soundchip-Musik spielt er keine große Rolle. Auf die von den anderen Soundchips abweichende Programmierung (vgl. Talbot-Watkins 1998) soll daher an dieser Stelle nicht weiter eingegangen werden.

2.3 Weiterentwickelte PSG-Soundchips

In den folgenden Jahren werden die erwähnten PSG-Soundchips weiterhin in verschiedenen 8 Bit-Heimcomputern verwendet. Daneben sollten rückwirkend zwei neu entwickelte Plattformen für die Entwicklung und die Geschichte von Computerspielmusik von großer Bedeutung sein: Der COMMODORE C64 und das NINTENDO FAMICOM / NES.

Der C64 wurde vor allem durch die Implementierung eines speziell entwickelten Synthesizer-Soundchips zu einem reizvollen Musikcomputer der 1980er Jahre. Das NES besitzt durch seine große Verbreitung und der teilweise sehr hohen Qualität der meist in Japan entwickelten Spiele, bei denen auch der Musik besondere Beachtung geschenkt wurde, eine besondere Stellung in der Geschichte der Computerspielmusik. Die Klangerzeuger beider (MOS SID und 2A03) sollen daher detailliert vorgestellt werden.

In Spielautomaten werden ab 1984 in erster Linie auf FM-Synthese basierende Soundchips verwendet. Auch in den ab 1985 erscheinenden Nachfolgern der MSX-Heimcomputer finden sich in erster Linie FM-Soundchips⁵¹. Die Klangsynthese auf Basis von Frequenzmodulation erhält darüber hinaus in begrenztem Umfang bereits in späten FAMICOM / NES-Modulen Einzug. Auf dieser Plattform ist es möglich, die Klangerzeugung der Konsole durch speziell entwickelte Chips auf den Spielmodulen zu erweitern.

⁵¹ Namentlich die Chips YM 2203, YM 2413 (FM-Pac, MSX 2+, MSX-Music), YM 2151 (Yamaha SFG-01 FM Sound Synthesizer Unit), Y 8950 (Philips MSX-Audio, MSX 2 optional) und YM 2164 (Yamaha SFG-05 FM Sound Synthesizer Unit II) (vgl. Otten 2004, Forster 2005: 77ff). Vgl. Kapitel 2.5.2

2.3.1 der SID-Chip des C64 (MOS 6581/ 8580 SID)

Der SID-Chip des bereits 1982 in Amerika vorgestellten COMMODORE C64 gilt als der bekannteste Soundchip überhaupt. Die Klangfarbe des Sound Interface Device - Chips ist durch seine auf 4 Bit-Wellenformgeneratoren beruhende Klangerzeugung äußerst eigenständig und durchsetzungsfähig und hat unter anderem zu einer Reihe von Software auf Basis des emulierten Chips geführt. Der SID nimmt aufgrund seiner Klangeigenschaften und der großen Verbreitung eine zentrale Rolle unter den in dieser Arbeit vorgestellten Soundchips ein, weswegen eine detaillierte Betrachtung seiner Funktionsweise anhand des Datenblatts erfolgt. Der Entwickler des SID-Chips ist Robert Yannes, welcher für MOS TECHNOLOGIES schon den Soundchip des Commodore VIC 20⁵² mitentwickelt hat. Sein persönlicher Ehrgeiz während der Entwicklung des SIDs hing mit der Unzufriedenheit über damals verfügbare Soundchips zusammen:

Commodore C64 (1982)	
CPU; Taktfrequenz	6510; 1 MHz
RAM	64 kB bis 128 kB
ROM	20 kB
Datenträger	Kassette, Diskette, Modul
Kapazität	bis ~ 200 kB

„I thought the sound chips on the market [...] were primitive and obviously had been designed by people who knew nothing about music. [...] I was attempting to create a synthesizer chip which could be used in professional synthesizers.“ (zit. n. Weske 2000: o.S.)

Nach seiner Arbeit am SID gründet Yannes die Firma ENSONIQ, um sich der Entwicklung professioneller digitaler Synthesizersysteme zu widmen.

Die Unterschiede zwischen den beiden Modellrevisionen des SID, 6581 und 8580 sind marginal. Glashüttner erwähnt, dass der 8580 ein wenig klarer und schärfer klinge als der 6581, welcher wiederum „wärmere“, „analoger“ klingende Sounds produziere (vgl. Glashüttner 2000: 1f). Der SID-Chip markiert im Vergleich zu den anderen hier vorgestellten Soundchips den obertonreichsten und abwechslungsreichsten Grund-„Sound“. Er besitzt durch seinen genuinen Eigenklang mehr Charakter als andere Soundchips.

2.3.1.1 Klangerzeugung und technische Übersicht

Der SID stellt einen nach den Grundsätzen der subtraktiven Klangsynthese aufgebauten Synthesizer auf einem Chip dar, der mit drei programmierbaren Oszillatoren, jeweils vier wählbaren Wellenformen (Dreieck, Sägezahn, variable Pulswelle und Rauschen) sowie einem ADSR-Hüllkurvengenerator pro Kanal ausgestattet ist. Er besitzt zur weiteren Klangformung die Möglichkeit der Synchronisierung und der Ringmodulation zweier Oszillatoren sowie ein programmierbares, resonanzfähiges Filter mit drei verschiedenen, miteinander kombinierbaren Charakteristiken, welchem der Klang beliebiger Kanäle zugeführt werden kann. Der Chip

⁵² Der Rechner hieß in Europa VC 20 und in Japan VC 1001 (Wurster 2002: 303). Sein auch VIC genannter Soundchip, der MOS 6561 ist von den Fähigkeiten vergleichbar mit dem SN 76489 AN bzw. Pokey, klanglich ähnelt er dem SID. Ewing (o.J.) bietet eine technische Dokumentation des MOS 6561.

erlaubt die Abfrage zweier Potentiometer über seine 8 Bit-I/O-Ports und bietet die Möglichkeit der Beimischung eines externen Audiosignals. Darüber hinaus besitzt er einen Zufallsgenerator. Das Blockdiagramm (Abb. 6) bietet eine Übersicht der verschiedenen Elemente.

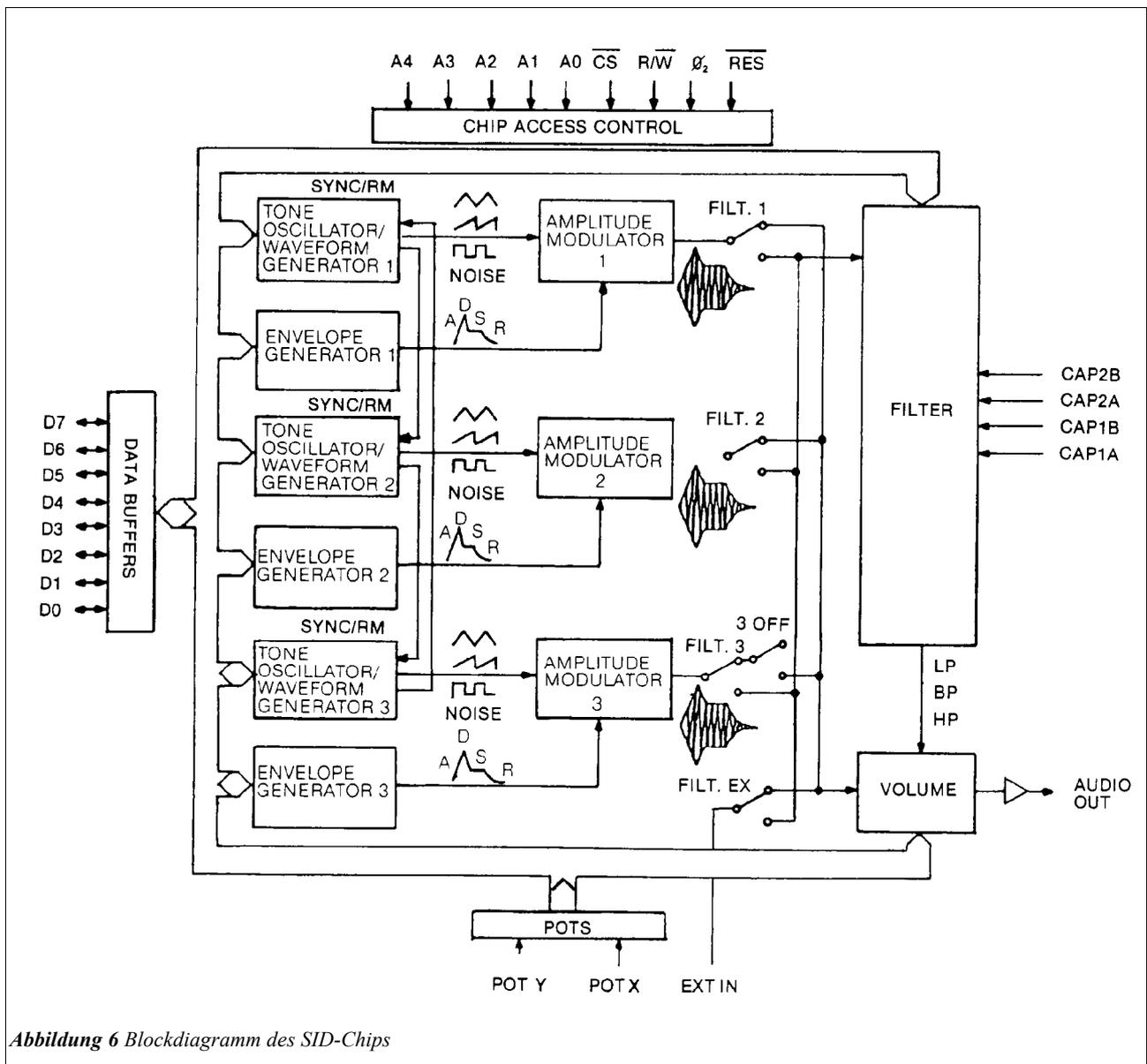


Abbildung 6 Blockdiagramm des SID-Chips

Der SID bietet somit im Vergleich zu den konkurrierenden Soundchips seiner Zeit eine geradezu opulente Ausstattung. Die Tonhöhen der Oszillatoren werden beim SID ebenso wie bei den anderen vorgestellten Soundchips durch die Methode der Frequenzdivision errechnet. Anders als bei diesen befindet sich beim SID jedoch der Wert des Frequenzregisters im Zähler und nicht im Nenner, was zu einer musikalisch sinnvollen, fein aufgelösten Skala von 65536 verschiedenen Tonhöhen führt⁵³.

⁵³ $F_{OUT} = (N_{(16)} * F_{clk} / 16777216)$ Hz, wobei $N_{(16)}$ dem 16 Bit-Wert der Frequenzregister entspricht, F_{clk} der Eingangstaktfrequenz. Bei der Standard-Taktfrequenz von 1MHz produziert der SID Grund-Frequenzen zwischen 0 Hz und 3906 Hz. Der höchste (Grund-)Ton liegt somit zwischen A#7 und B7, womit die Standard-Klavatur abgedeckt ist.

Wie bereits erwähnt enthält auch der Pokey-Chip einen 16 Bit-Modus mit gleich vielen unterschiedlichen Tonhöhen, er kann jedoch aufgrund der ungünstigen Umrechnungsformel nicht mit der Auflösung des SID konkurrieren: Hohe Frequenzen sind beim Pokey nur in weitaus größerer Rasterung als beim SID erzeugbar. Der SID besitzt drei Wellenformgeneratoren. Nach der D/A-Wandlung durchläuft das erzeugte Audio-Signal den vom Hüllkurvengenerator gesteuerten Amplitudenmodulator sowie Filter und Verstärker. Die Steuerdaten der letzteren Bauteile durchlaufen zum Zweck der (Audio-)Signalbearbeitung jeweils einen eigenen D/A-Wandler (vgl. Glasshüttner 2000: 3).

2.3.1.2 Beschreibung der Programmierung anhand der Registertabelle

Address					Reg #	Data								Reg Name	Reg Type
A4	A3	A2	A1	A0	(Hex)	D7	D6	D5	D4	D3	D2	D1	D0		
VOICE 1															
0	0	0	0	0	00	F7	F6	F5	F4	F3	F2	F1	F0	Freq Lo	Write-only
1	0	0	0	0	01	F15	F14	F13	F12	F11	F10	F9	F8	Freq Hi	Write-only
2	0	0	0	1	02	PW7	PW6	PW5	PW4	PW3	PW2	PW1	PW0	PW LO	Write-only
3	0	0	0	1	03	—	—	—	—	PW11	PW10	PW9	PW8	PW HI	Write-only
4	0	0	1	0	04	NOISE				TEST	RING MOD	SYNC	GATE	Control Reg	Write-only
5	0	0	1	0	05	ATK3	ATK2	ATK1	ATK0	DCY3	DCY2	DCY1	DCY0	Attack/Decay	Write-only
6	0	0	1	1	06	STN3	STN2	STN1	STN0	RLS3	RLS2	RLS1	RLS0	Sustain/Release	Write-only
VOICE 2															
7	0	0	1	1	07	F7	F6	F5	F4	F3	F2	F1	F0	Freq Lo	Write-only
8	0	1	0	0	08	F15	F14	F13	F12	F11	F10	F9	F8	Freq Hi	Write-only
9	0	1	0	0	09	PW7	PW6	PW5	PW4	PW3	PW2	PW1	PW0	PW LO	Write-only
10	0	1	0	1	0A	—	—	—	—	PW11	PW10	PW9	PW8	PW HI	Write-only
11	0	1	0	1	0B	NOISE				TEST	RING MOD	SYNC	GATE	Control Reg	Write-only
12	0	1	1	0	0C	ATK3	ATK2	ATK1	ATK0	DCY3	DCY2	DCY1	DCY0	Attack/Decay	Write-only
13	0	1	1	0	0D	STN3	STN2	STN1	STN0	RLS3	RLS2	RLS1	RLS0	Sustain/Release	Write-only
VOICE 3															
14	0	1	1	1	0E	F7	F6	F5	F4	F3	F2	F1	F0	Freq Lo	Write-only
15	0	1	1	1	0F	F15	F14	F13	F12	F11	F10	F9	F8	Freq Hi	Write-only
16	1	0	0	0	10	PW7	PW6	PW5	PW4	PW3	PW2	PW1	PW0	PW LO	Write-only
17	1	0	0	0	11	—	—	—	—	PW11	PW10	PW9	PW8	PW HI	Write-only
18	1	0	0	1	12	NOISE				TEST	RING MOD	SYNC	GATE	Control Reg	Write-only
19	1	0	0	1	13	ATK3	ATK2	ATK1	ATK0	DCY3	DCY2	DCY1	DCY0	Attack/Decay	Write-only
20	1	0	1	0	14	STN3	STN2	STN1	STN0	RLS3	RLS2	RLS1	RLS0	Sustain/Release	Write-only
Filter															
21	1	0	1	0	15	—	—	—	—	—	FC2	FC1	FC0	FC LO	Write-only
22	1	0	1	1	16	FC10	FC9	FC8	FC7	FC6	FC5	FC4	FC3	FC HI	Write-only
23	1	0	1	1	17	RES3	RES2	RES1	RES0	Filt EX	Filt 3	Filt 2	Filt 1	RES/Filt	Write-only
24	1	1	0	0	18	3 OFF	HP	BP	LP	VOL3	VOL2	VOL1	VOL0	Mode/Vol	Write-only
Misc															
25	1	1	0	0	19	PX7	PX6	PX5	PX4	PX3	PX2	PX1	PX0	POTX	Read-only
26	1	1	0	1	1A	PY7	PY6	PY5	PY4	PY3	PY2	PY1	PY0	POTY	Read-only
27	1	1	0	1	1B	07	06	05	04	03	02	01	00	OSC3/Random	Read-only
28	1	1	1	0	1C	E7	E6	E5	E4	E3	E2	E1	E0	ENV3	Read-only

Abbildung 7 Registertabelle des SID

Der Chip besitzt 29 Register, wobei die ersten 25 *write-only* sind und somit die Klangerzeugung des SID beeinflussen. Die letzten vier sind *read-only*, können also nicht verändert werden. Pro Stimme sind sieben Register (0-6, 7-13, 14-20) für die Klangerzeugung zuständig, vier Register steuern das Filter. Die Steuerung der Tonhöhen der Oszillatoren erfolgt über die 16 Bit großen Zahlenwerte der entsprechenden Register. Register 4, 11 und 18 sind logische Kontrollregister der jeweiligen Stimme. Sobald in diesen das entsprechende Gate-Bit auf 1 gesetzt ist, wird der Hüllkurvengenerator der jeweiligen Stimme getriggert und steuert den Lautstär-

kenverlauf (Attack, Decay, Sustain) des nun wiedergegebenen Klangs. Sobald das Gate-Bit auf 0 gesetzt wird, beginnt der Release-Abschnitt des Hüllkurvengenerators, womit der Klang ausklingt, wenn nicht während der Ausklingphase ein neuer Klang getriggert wird⁵⁴. Wenn im Kontrollregister das SYNC-Bit (D1) auf 1 gesetzt wird, synchronisiert dies den Nulldurchgang der Oszillatoren zueinander⁵⁵ was einen „Hard Sync“-Effekt ergibt, der komplexe Obertöne zu der Grundfrequenz hinzufügt. Die Ringmodulation kann in Bit D2 des Kontrollregisters aktiviert werden. Dann wird die Dreieckswelle des gewählten Oszillators durch eine Ringmodulation der beiden jeweiligen Oszillatorfrequenzen ersetzt. Durch Setzen des Test-Bits wird der Noise-Output zurückgesetzt und der Pulswellen-Output auf einem konstanten Spannungsniveau gehalten. Wie der Name schon sagt, wurde das Bit für Test-Zwecke implementiert, es ist jedoch auch als Synchronisationsmittel und zur experimentellen Klangerzeugung verwendet worden. Die restlichen 4 Bits der Kontrollregister fungieren als Wellenformselektoren: Der Oszillator erzeugt die dem jeweiligen Bit entsprechende Wellenform, wobei auch mehrere Wellenformen auf einmal gewählt werden können, was zu keiner Mischung der Wellenform wie bei einem klassischem Synthesizer führt, sondern geräuschhaftere Klänge erzeugt. Diese Eigenart führt zu einem interessanten Aufstocken der verfügbaren Wellenformen.

Die Register 2 und 3 (bzw. 9 und 10 sowie 16 und 17) steuern die Pulsweitenmodulation⁵⁶, bleiben also bei anderen Wellenformen als der Pulswelle ohne Auswirkungen auf den Klang. Die modulierte Pulswelle ist als Bass-Sound einer der bekanntesten und am häufigsten verwendeten Klänge des SID-Chips. Die Hüllkurvengeneratoren werden über die in die entsprechenden Register⁵⁷ geschriebenen Werte gesteuert. Dabei steht für jeden der vier Hüllkurvenwerte ein 4 Bit großes Wort. Die Werte für Attack, Decay und Release generieren eine bestimmte Dauer für die jeweilige Phase der Hüllkurve, welche von der Eingangstaktfrequenz abhängig ist. Tabelle 1 zeigt die hierzu korrelierenden Dauern. Ein Attack-Wert von 2 würde den Hüllkurvengenerator dementsprechend veranlassen, die Lautstärke des generierten Klangs innerhalb von 16 ms vom Minimum zum Maximum zu steuern. Der Decay-Wert regelt die Dauer vom Maximum zum Sustain-Level, welches einem linearen Wert zwischen 0 und 15 entspricht.

Die Register 21 und 22 enthalten einen 11 Bit großen Wert, der die Filtereckfrequenz auf lineare Weise in 2048 Schritten steuert. Die Eckfrequenz des Filters ist von der Kapazität der verbauten Kondensatoren abhängig, laut des Data Manuals liegt die Eckfrequenz zwischen 30 Hz und 20 kHz, in späteren Revisionen des C64

Wert	Attack-Zeit	Decay-/Release-Zeit
(Eingangstaktfrequenz = 1 MHz)		
0	2 ms	6 ms
1	8 ms	24 ms
2	16 ms	48 ms
3	24 ms	72 ms
4	38 ms	114 ms
5	56 ms	168 ms
6	68 ms	204 ms
7	80 ms	240 ms
8	100 ms	300 ms
9	250 ms	750 ms
10	500 ms	1,5 s
11	800 ms	2,4 s
12	1 s	3 s
13	3 s	9 s
14	5 s	15 s
15	8 s	24 s

Tabelle 1 Die verschiedenen Dauern der Hüllkurven-Phasen in Relation zu den Registerwerten (vgl. Commodore o.J.:5)

⁵⁴ Das Setzen des GATE-Bits auf 0 entspricht also dem Loslassen der Taste auf einer Klaviatur beim analogen Synthesizer. Ein erneutes Triggern des Hüllkurvengenerators durch Setzen des GATE-Bits auf 1 wird diesen dazu veranlassen, wiederum mit der Attack-Phase einen neuen Durchlauf zu beginnen.

⁵⁵ Je nach Kanal wird die Frequenz des ersten Oszillators zu der des dritten (R4) bzw. die des zweiten zum ersten (R11) bzw. die des dritten zum zweiten (R18) synchronisiert. Dieselbe Richtung der Einflussnahme (1→3, 2→1, 3→2) gilt auch für den Ringmodulator.

⁵⁶ Die Pulsbreite wird nach folgender Formel errechnet: $PW_{out} = (PW_n / 40,95) \%$, wobei PW_n dem 12-Bit-Wert der o.g. Register entspricht. Steht der Wert auf 0 (\$000) oder 4095 (\$FFF), wird eine konstante Spannung abgegeben, der Mittelwert von 2048 (\$800) produziert dementsprechend eine Rechteckwelle.

⁵⁷ Also je nach Kanal 5 und 6 bzw. 12 und 13 sowie 19 und 20.

wurden Kondensatoren mit abweichenden Kapazitäten verwendet, wodurch auch die Eckfrequenz einer Veränderung unterlag. Daher führte die Benutzung des SID-Filters auf verschiedenen Exemplaren des C64 zu unterschiedlichen klanglichen Ergebnissen:

„[T]here's a three-state filter which would doubtless[ly] come in very handy if only you were allowed to use it. Trouble is, the filters on old 64s are calibrated so differently from those on newer machines, that a piece of filter-inspired commercial software might sound great on one person's 64, but utterly appalling on others“ (Vosburgh o.J.: o.S.).

Auch Chris Hülsbeck berichtet von Erfahrungen mit seinem ersten Stück „Shades“ (♫009, ♫010), nach denen er in der Benutzung des Filters sehr vorsichtig geworden sei⁵⁸. Die Resonanz des Filters, welche die Frequenzen um die Filtereckfrequenz hervorhebt, kann durch einen 4 Bit-Wert linear zwischen 0 (keine Resonanz) und 16 (maximale Resonanz) geregelt werden. Die restlichen Bits des Registers 23 legen fest, ob das Filter das Signal der jeweilige Stimme oder des externen Audio-Eingang beeinflusst. Register 24 enthält u.a. die Schaltung der Filtercharakteristik und die Regelung der Gesamtlautstärke. Erstere lässt sich zwischen Lowpass (LP), Bandpass (BP) und Highpass (HP) umschalten, die Gesamtlautstärke des SID kann linear in 16 Schritten geregelt werden.

Register 25 und 26 sind die ersten beiden Read-Only-Register des SID, in welche die A/D-Wandler der optional angeschlossenen Potentiometer alle 512 Eingangstaktzyklen einen 8 Bit-Wert schreiben. Der C64 besitzt standardmäßig keine angeschlossenen Potentiometer, findige Bastler erweiterten den Computer jedoch, um mit Hilfe von Drehreglern mehr Kontrolle über die Klangverläufe (v.a. Filtereckfrequenz und Resonanzfrequenz) des SID zu bekommen. Auch die auf dem SID-Chip basierende SID-STATION, ein Mitte der 1990er Jahre entworfener, kommerzieller digitaler Synthesizer⁵⁹, bedient sich dieser Möglichkeit.

Ein Abbild der von Oszillator 3 generierten Wellenform findet sich in Register 27 wieder, welches als LFO oder (im Falle eines Rauschens) für Sample & Hold - Effekte genutzt werden kann, wofür der Audio-Ausgang des dritten Oszillators ausgeschaltet werden sollte (indem das „3OFF“-Bit in Register 24 gesetzt wird). Register 28 enthält die Werte des dritten Hüllkurvengenerators, die zum Beispiel als Filter- oder PWM-Hüllkurve fungieren können. Für eine Beschreibung des ENV3-Registers (29) muss das Gate-Bit des dritten Klangkanals (R18) auf 0 gesetzt werden. Das Auslesen der Read-Only Register und das Routen der Werte auf Modulationsziele wie Tonhöhen- oder Filterfrequenz geschieht softwareseitig, da der SID keine entsprechende interne Modulationsmatrix bietet.

⁵⁸ *„Ich wusste zu dem Zeitpunkt nicht, dass es unterschiedliche C64 Revisionen gegeben hat. Ich hatte ja nur einen zu Hause und ging davon aus, dass der Filter auf jedem C64 gleich klingt. Dann habe ich das Stück mit Filterprogrammierung eingeschickt und hab den ersten Platz gemacht [♫009]. Im Nachhinein habe ich den Track auf mehreren anderen C64s gehört und war ganz erstaunt, wie unterschiedlich das klingt. Auf manchen war der Filter sogar so zu, dass man die Melodie gar nicht mehr hören konnte. Also habe ich für die nächste Ausgabe dieser Zeitschrift ein Programm geschrieben, mit dem man diesen Filter an verschiedene C64 anpassen konnte. Dadurch konnte man den Filter für alle Sounds in dem Stück höher oder tiefer schrauben und dann das Stück als neue Version [♫010]abspeichern. [...] Deswegen war ich später immer relativ vorsichtig mit dem Filter“ (Hülsbeck 2005a).*

⁵⁹ <http://www.sidstation.com/> (15.06.2005).

2.3.2 Der 2A03 im Nintendo Famicom / Entertainment System

Das 1983 in Japan erscheinende NINTENDO FAMICOM bzw. 1985 in anderem Gehäuse als NINTENDO ENTERTAINMENT SYSTEM (NES) in den USA und Europa eingeführte Videospielsystem ist mit weltweit über 60 Millionen verkauften Exemplaren eine der erfolgreichsten Spielkonsolen. Erst im September 2003 wird die letzte Produktionsstraße der Konsole in Japan auf-

Nintendo Famicom / Nes (1983 / 1985)	
CPU; Taktfrequenz	Ricoh 2A03; 1,79 MHz
RAM	2 kB + 2,25 kB VRAM
Datenträger	Modul, Kasette (jp.), Diskette (jp.)
Kapazität	bis 512 kB

gegeben, da wichtige Bauteile am Markt nicht mehr erhältlich sind (vgl. Yamauchi 2003). Als CPU besitzt sie eine um Soundgeneratoren erweiterte, von RICOH produzierte Version des MOS 6502, den 2A03. Dieser wird ab Markteinführung in überaus großer Stückzahl gefertigt, um die Kosten gering zu halten (vgl. Sheff 1993: 41f). Ein Data Manual des Prozessors liegt im Gegensatz zu anderen Soundchips nicht vor. Dies mag auch daran liegen, dass es sich nicht um einen eigenen Soundchip handelt, sondern digitale Oszillatoren über CPU-Register angesteuert werden. Außerdem existieren für Spielkonsolen gemeinhin weniger verfügbare Detail-Informationen, als für die Heimcomputer, da diese als 'Blackboxes' konzipiert sind und für den Benutzer bzw. den Spieler die interne Funktionsweise i.d.R. nicht von Interesse ist. Womöglich existiert eine japanische Dokumentation der Klangerzeugung, da dort anders als in Europa und Amerika ein BASIC-KIT für das FAMICOM erschienen ist (NINTENDO, 1984), welches die Programmierung der Konsole ermöglicht. Die folgenden technischen Daten basieren daher primär auf dem „NES Sound Channel Guide“ von Brad Taylor (Taylor 2000b).

Die Klangerzeugung des 2A03 umfasst 4 Kanäle. Die Oszillatoren sind im Vergleich zu denen des SID weitaus beschränkter: Zwei Pulswellen mit je vier wählbaren Pulsbreiten, eine Dreieckswelle und ein Rauschkanal stehen zur Verfügung. Außerdem existiert in den europäischen und amerikanischen Modellen ein fünfter Kanal, der „Delta Modulation Channel“, welcher PCM-kodierte Samples abspielen kann, wozu über DMA (Direct Memory Access) ein 1 Bit breiter Datenstrom direkt vom Speicher gelesen wird. Aufgrund des limitierten Modul-Speicherplatzes und des Fehlens dieses Kanals im japanischen FAMICOM kam diese Technik jedoch kaum zum Einsatz⁶⁰.

⁶⁰ Genaue Zahlen über den Sample-Kanal unterstützenden Spiele liegen nicht vor. Der Willkommensatz des Spiels GAUNTLET II (MINDSCAPE, 1990; ♪104) stellt ein Beispiel für die Qualität der Sample-Wiedergabe mittels DMC dar.

Funktionsweise der Klangerzeugung

Für die Tonhöhenberechnung der Oszillatoren wird die interne Chip-Taktfrequenz von 1,79 MHz durch den 11 Bit-großen Tonhöhen-Wert des jeweiligen Kanals, addiert mit eins, dividiert, was zu 2048 möglichen Werten führt. Dieser Ausgangs-Wert dient dem jeweiligen Oszillator als Grundlage für die resultierende Tonhöhe. Die Dreieckswelle wird durch einen 5 Bit Dreieckswellengenerator anhand der auf Basis des 11 Bit-Wertes ausgerechneten Frequenz generiert. Die beiden Pulswellen werden durch 4 Bit Zähler erzeugt⁶¹. Die Aufteilung der generierten Werte der Pulswellengeneratoren ist je nach Pulsbreiten-Bits (in Klammern): 2:14 (00), 4:12 (01), 8:8 (10) oder 12:4 (11). Die Pulsbreite entspricht also 12,5 %, 25 %, 50 % oder 75 %, was in verschiedenen Obertonanteilen und damit unterschiedlich vollen Klangfarben resultiert.

Register	Kanal	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
\$4000	Pulswelle 1	Pulsbreite		Tondauer-Zähler aus /Decay loopen	Hüllkurve aktivieren	Lautstärke / Hüllkurve Anfangswert				
\$4001		Sweep an	Sweep Update Rate			Sweep Richtung	Right Shift Amount			
\$4002		8 LSB der Tonhöhe								
\$4003		Tondauern-Zähler Anfangswert					3 MSB der Tonhöhe			
\$4004	Pulswelle 2	Pulsbreite		Tondauer-Zähler aus /Decay loopen	Hüllkurve aktivieren	Lautstärke / Hüllkurve Anfangswert				
\$4005		Sweep an	Sweep Update Rate			Sweep Richtung	Right Shift Amount			
\$4006		8 LSB der Tonhöhe								
\$4007		Tondauern-Zähler Anfangswert					3 MSB der Tonhöhe			
\$4008	Dreieckswelle	Tondauer-Zähler aus	Linear Counter Load							
\$4009		unbenutzt								
\$400A		8 LSB der Tonhöhe								
\$400B		Tondauern-Zähler Anfangswert					3 MSB der Tonhöhe			
\$400C	Rauschen	unbenutzt		Tondauer-Zähler aus /Decay loopen	Hüllkurve aktivieren	Lautstärke / Hüllkurve Anfangswert				
\$400D		unbenutzt								
\$400E		Zufallszahlen-generator Typ	unbenutzt			Playback Sample Rate				
\$400F		Tondauern-Zähler Anfangswert					unbenutzt			
\$4015 (Read)	Tondauer-Zähler Status	IRQ Status von DMC	unbenutzt		DMC	Rauschen	Dreieck	Puls 1	Puls 2	
\$4015 (Write)	Kanal aktivieren	unbenutzt			DMC	Rauschen	Dreieck	Puls 1	Puls 2	

Table 2 Die für die Klangerzeugung zuständigen Register des 2A03 (nach Taylor 2000b)

Die Berechnung der Tondauern wird für alle Kanäle anhand von Zählern realisiert. Diese zählen von einem 7 Bit-Wert aus mit einer Frequenz von 60 Hz (mutmaßlich 50 Hz in der PAL-Version⁶²) abwärts, bis der Wert 0 erreicht wird. Dann wird die Klanguausgabe auf dem entsprechenden Kanal gestoppt und der Zähler verbleibt

⁶¹ Die Frequenz der Dreieckswelle (32 Taktzyklen zur Beschreibung eines Wellendurchgangs) entspricht also einem 32tel des aus der Frequenzdivision gewonnen Wertes, die der Rechteckwellen (16 Taktzyklen pro Durchgang) einem 16tel. Damit liegt der Tonumfang der Dreieckswelle insgesamt eine Oktave unter den Rechteckwellen.

Bits 7-3		Startwert (Dez)	Dauer (s)
00011		1	0,017
00101		2	0,033
00111		3	0,05
01001		4	0,067
00111	00000	5	0,083
01101	10000	6	0,1
01111	01100	7	0,117
10001	11100	8	0,133
10011		9	0,15
10101	11110	10	0,167
10111		11	0,183
11001	10010	12	0,2
11011		13	0,217
11101	01110	14	0,233
11111		15	0,25
11110		16	0,267
00100		20	0,333
10100		24	0,4
01010		30	0,5
11010		36	0,6
00110		40	0,667
10110		48	0,8
01000		80	1,333
11000		96	1,6
00001		127	2,117

Tabelle 3 Die verschiedenen Tondauern, die der jeweilige Zähler bei einer Frequenz von 60 Hz aus den Bits 7-3 generiert (nach Taylor 2000b).

auf 0. Über das „Tondauer-Zähler aus“-Bit kann der jeweils gespielte Ton auch länger gehalten werden, da der Zähler dann beim aktuellen Wert verbleibt. Der 7 Bit-Startwert des jeweiligen Zählers wird aus einem 5 Bit-Wert (Register \$4003, \$4007, \$400B, \$400F) erzeugt und kann 25 verschiedene Tondauern generieren, die in Tabelle 3 exemplarisch abgebildet sind.

Wenn die Hüllkurve mit dem entsprechenden Bit (der Register \$4000, \$4004 und \$400C) deaktiviert wurde, wird der Ausgang der jeweiligen Stimme mit dem in Bits 0-3 geschriebenen Lautstärkewert (von 0-16) an den jeweiligen DAC (Digital-Analog Converter) übergeben. Die Signale der beiden Pulswellen sowie der Ausgang von Dreieckswelle und Rauschen liegen jeweils an einem gemeinsamen Pin des Chips an, werden also hardwareseitig gemischt.

Ist die Hüllkurve aktiviert, regelt sie die Amplitude von Pulswellen und Rauschkanal linear auf 0. Dieses Decay (die Ausklingzeit) wird von einem abwärts zählenden 4 Bit-Zähler⁶³. Wenn der Zähler 0 erreicht, wird je nach Status des „Decay loopen“ Bits entweder die Decay-Phase wiederholt (was in einem Tremolo-Effekt resultiert) oder der Kanal bleibt still. Nur die Pulswellenkanäle und der Rauschkanal besitzen diese Art der Hüllkurve. Die Ausklingzeit des Dreieckswellenkanals generiert ein linearer Zähler, mit dem eine Ausklingzeit von maximal einer knappen halben Sekunde möglich wird⁶⁴.

Die Klangerzeugung der Pulswellen besitzt darüber hinaus eine Sweep-Funktion, also einen Portamento-Effekt, bei dem der betreffende Kanal von dem bisher wiedergegebenen Wert auf die nächste Tonhöhe gleitet. Dabei wird der für die Tonhöhe zuständige 11 Bit-Wert kontinuierlich ausgelesen und neu geschrieben. Der Effekt wird über das Register \$4001 bzw. \$4005 gesteuert. Hierfür gibt es ein Aktivierungs-Bit, ein Bit legt darüber hinaus die Richtung des Sweep-Effekts fest. Die Sweep-Rate bestimmt, wie häufig neue Werte abgelegt werden und regelt damit die Länge des Effekts⁶⁵.

Wie erwähnt, entstammen diese Informationen auf der FAQ von Brad Taylor (2000b), über diese Quelle hinaus war keine Dokumentation der Klangerzeugung verfügbar. Die Dreieckswelle des 2A03 klingt relativ analog und bildet einen angenehmen Kontrast zu den beiden Rechteckwellen, die Klangerzeugung bietet im Vergleich zum SID jedoch nur limitierte Möglichkeiten.

⁶² Nicht geklärt ist, ob die in den Formeln angegebenen Frequenzen auch in der PAL-Version dieselben sind. Da sich in den bei Taylor genannten Formeln häufig durch 60 teilbare Werte befinden, liegt die Vermutung nahe, dass für die PAL-Version des Gerätes diese durch das jeweils durch 50 teilbare Produkt ersetzt werden müssen.

⁶³ Die Dauer dieser wird wie folgt berechnet: $240\text{Hz} / (N+1)$, wobei N dem 4 Bit Wert aus den Registern \$4000, \$4004 bzw. \$400C entspricht. Die Ausklingzeit für diese drei Klangkanäle kann somit zwischen 0,133 und 1,067 Sekunden liegen.

⁶⁴ Dieser Zähler zählt vom 7 Bit-Wert des nur bei der Dreieckswelle vorhandenen „Linear Counter Load“ Registers aus mit einer Frequenz von 240 Hz bis auf 0.

⁶⁵ Die Erneuerungsrate ist abhängig vom Prozessortakt und entspricht $120\text{ Hz} / N+1$, wobei N dem 3 Bit-großen Sweep-Rate-Wert entspricht. Je höher der Wert, desto länger also der Effekt.

Die Beschränkung auf die verfügbaren Stimmen und Klangfarben wurden bei einigen wenigen Spielen der japanischen Hersteller KONAMI, NINTENDO, NAMCO und SUNSOFT ab Ende der 1980er Jahre durch in den Spielmodulen implementierte Zusatz-Soundchips umgangen⁶⁶

2.3.3 PSG-Sound in den 1990ern: der Nintendo Game Boy

Als der NINTENDO GAME BOY 1989 auf den Markt kommt, gilt der auf einem ZILOG Z 80 basierende und nur vier Graustufen darstellende 8 Bit-Handheld bereits als veraltet. Die Konkurrenzprodukte SEGA GAME GEAR (SEGA, 1990) und ATARI LYNX (ATARI, 1989) bieten Farbdisplays mit Hintergrundbeleuchtung, haben dadurch jedoch den Nachteil eines hohen Batterieverbrauchs.

	Game Boy (1989)	Game Boy Color (1996)	Game Boy Advance (2001)
CPU; Taktfrequenz	Sharp Z80; 4,2 MHz	Sharp Z80; 4,2 / 8,4 MHz	ARM7-Variante; 16,7 MHz
RAM	8 kB + 8 kB	8 kB + 16 kB VRAM	32 kB + 96 kB VRAM + 256 kB WRAM + 16 kB Sound
Modulkapazität	0,25 bis 4 MBit	0,25 bis 8 MBit	bis 256 MBit

Der preislich günstigere GAME BOY wird zusammen mit einem Link-Kabel und dem Spiel TETRIS (NINTENDO, 1989) ausgeliefert und verkauft sich bis 1995 hervorragend. Vor allem durch die Popularität des Spiels POKÉMON (NINTENDO, 1995) wird der bereits in Ablösung begriffene GAME BOY in den Jahren 1996 (POCKET) und 1998 (COLOR) zwei Überarbeitungen unterzogen und bis über das Erscheinen des Nachfolgers GAME BOY ADVANCE (2001) hinaus erfolgreich weiterverkauft. Weltweit werden bis 2002 über 120 Millionen Geräte der verschiedenen Revisionen abgesetzt (vgl. Forster 2005: 128f). Die in dem Hauptprozessor integrierte Klangerzeugung des GAME BOYS (auch als RICOH DMG SYSTEM bekannt), die über die verschiedenen Gerätegenerationen beibehalten wird, stellt vier PSG-Klangkanäle zur Verfügung: Zwei Pulswellenkanäle und einen Wavetable-Kanal, der mit einer 32 Werte langen Wellenform mit einer Auflösung von 4 Bit beschrieben werden kann⁶⁷. Darüber hinaus ist ein Rauschkanal integriert⁶⁸.

Die ersten drei Kanäle besitzen demzufolge einen Tonumfang von 64 Hz bis 131072 Hz. Der Rauschkanal kann Frequenzen von 2 Hz bis 1048576 Hz produzieren, wobei beide Extreme weit außerhalb des menschlichen Hörbereichs liegen. Lediglich der erste Kanal besitzt einen Sweep-Effekt (Portamento), die Pulsbreite der beiden Pulswellen lässt sich wie beim NES zwischen 12,5 %, 25%, 50 % und 75 % wählen. Auch in den Berechnungen von Tonhöhe und Tonlängen ist die Klangerzeugung des GAME BOYS der des NES sehr ähnlich. Auch der GAME BOY errechnet die 1024 möglichen Tonhöhen aus einem 11 Bit-Wert, die möglichen Tondauern sind auf 32 verschiedene beschränkt, neben den vier verschiedenen Pulsbreiten sind weder Filter noch

⁶⁶ Diese Soundchips tragen die Namen VRCVI, VRCVII, MMC5 audio, Namco 106 und Sunsoft FME-07 und finden sich z.B. in CASTLEVANIA 3, AKUMAJOU DENSETSU (KONAMI, 1989), sowie den Titeln MOURYOU SENKI MADARA (KONAMI, 1990), JUST BREED (ENIX, 1992), SHIN 4-JIN UCHI MAHJONG: YAKUMAN TENGOKU (NINTENDO, 1991), KING OF KINGS (NAMCO, 1988), GIMMICK (SUNSOFT, 1992) und LAGRANGE POINT (KONAMI, 1991) (vgl. Horton 1999a, 1999b und Mamiya o.J.). Die Zusatz-Soundchips basieren teilweise auf FM-Synthese (FAMICOM DISC SYSTEM Sound, Namco 106 und VRCVII, vgl. Horton 1999b und Horton 2000) oder schon bekannten Technologien wie dem AY-Chip (Sunsoft FME-07, vgl. Horton 2000). Die o.g. Spiele sind ebenso wie das FAMICOM DISC SYSTEM nur in Japan erschienen.

⁶⁷ Ein Beispiel für eine Wellenform mit 5 Bit großen Hex-Werten findet sich in Fußnote 85 bei der Beschreibung der Klangerzeugung des PC-ENGINE

⁶⁸ Diese und die folgenden Daten zur Klangerzeugung des GAME BOY stammen von Gevanyahu (2004).

Hüllkurven zur weiteren Klangbearbeitung enthalten. Da der GAME BOY batteriebetrieben ist und der eingebaute Bildschirm weltweit mit der gleichen Bildwiederholfrequenz arbeitet, ist hier der bei anderen Plattformen wichtige Unterschied zwischen PAL- und NTSC-Hardware (50 Hz / 60 Hz) irrelevant.

Der GAME BOY ist für die Soundchip-Musik von Computerspielen aufgrund der großen Verbreitung und dem langen Lebenszyklus von über zehn Jahren als überaus wichtig einzustufen. Mit der großen Zahl von Konvertierungen klassischer Computerspiele wurde auch die klassische Soundchip-Musik weiter portiert, wobei die Eigenheiten der beschränkten Klangerzeugung und damit sein „Sound“ einer ebenso großen Verbreitung unterliegen. Der Game Boy hat darüber hinaus ab Ende der 1990er Jahre durch spezielle Software für die populäre Soundchip-Musik eine große Bedeutung als Klangerzeuger erlangt.

2.4 Die Bedeutung der PSGs für Soundchip-Musik

Die besprochenen PSG-Soundchips haben in Abgrenzung zu den ihnen folgenden, auf den Technologien FM-Synthese und Sampling basierenden Chips ab Mitte der 1980er Jahre einen durch die reduzierten technischen Möglichkeiten der Klangerzeugung bedingten Eigenklang, der sich durch die Verwendung einfacher Wellenformen und das Fehlen von aufwendigen Möglichkeiten der Klangmanipulation auszeichnet. Für die in der „Sound Culture“ von Computerspielmusik entstehenden Musikgenres populärer Soundchip-Musik (vgl. Kapitel 4) sind aus heutiger Sicht vor allem NES, C64 und GAME BOY als Klangerzeuger von Bedeutung. Auch der AY-Chip bzw. YM 2149 ist durch seinen massenhaften Einsatz in zahlreichen Computerspielsystemen und Heimcomputern wie dem ATARI ST für Soundchip-Musik ein häufig genutzter Klangerzeuger. AY bzw. YM-Musik ist jedoch weniger bekannt als z.B. Musik für den SID-Chip. Einige Musiker verwenden auch den COMMODORE VIC und selbst das ATARI VCS für die sog. Chiptunes. C64, NES und GAME BOY besitzen jedoch den größten 'Kultstatus', was auch an ihrer großen Verbreitung und der breiten Rezeption der für sie produzierten Software liegen mag.

Dem SID wird sowohl bei der Betrachtung von Computerspielmusik als auch dem Phänomen der Chiptunes besondere Aufmerksamkeit geschenkt, da der C64 für eine Generation von Computernutzern in Deutschland und Europa den ersten Kontakt mit dem Computer markiert. Eine große Zahl an Fan-Webseiten und die noch bis in die zweite Hälfte der 1990er Jahre aktive Demo-Szene⁶⁹ belegen dies. Gerade die weite Verbreitung des C64 verbunden mit den für damalige Verhältnisse überragenden Möglichkeiten der zu Experimenten einladenden Klangerzeugung führten dazu, dass die Komposition von Musikdateien für Soundchips nicht allein den Ingenieuren bzw. Programmierern der Spieleherstellern vorbehalten blieb.

⁶⁹ Unter Demo-Szene versteht man eine ab Mitte der 1980er Jahre aus der sog. Hacking- und Cracking-Szene hervorgegangene Bewegung, die in Gruppen organisiert möglichst eindrucksvolle Demos (audiovisuelle Demonstrations-Programme) für Computer programmiert (vgl. Tasajärvi 2004a). Über diese wird auf Partys mit anderen Demo-Groups in Konkurrenz getreten, eine bestimmte Datei-Größe darf dabei je nach „Disziplin“ nicht überschritten werden. Für Demo-Programmierung ist das Ziel also ebenso wie bei der Programmierung von Computerspielen ein im Rahmen der technischen Möglichkeiten „optimales“ audiovisuelles Ergebnis zu erreichen.

Viele an Synthesizern interessierte Nutzer erstehen einen C64 gerade wegen seines Soundchips. COMMODORE bewirbt von Anfang an offensiv die Möglichkeiten des integrierten Synthesizers und eine Reihe von spezialisierter Literatur zur Programmierung von Sound und Musik in BASIC erscheint in den Folgejahren⁷⁰. Dem Anwender stehen bald eine breite Palette spezieller Soundeditoren für den SID-Chip zur Verfügung⁷¹.

Das NES stellt durch zahlreiche Titel mit qualitativ hochwertiger Spielmusik sowie die große Verbreitung der Konsole für die Entwicklung von Computerspielmusik eine wichtige Plattform dar.

In den Hoch-Zeiten von NES und COMMODORE C64 ab Mitte der 1980er Jahre haben Komponisten von Spielmusik in dem subkulturellen Milieu der 'Spielefreaks' erstmals einen gewissen Bekanntheitsgrad erlangt⁷². Aus Europa seien für die erste Generation der Heimcomputer-Komponisten der Deutsche Chris Hülsbeck, die Engländer Rob Hubbard und Martin Galway, sowie der Niederländer Jeroen Tel genannt, die allesamt durch ihre Kompositionen auf dem C64 bei Computerspielern berühmt wurden. Auf dem NES sind es vor allem japanische Komponisten, die bis heute für ihre Musik bewundert werden und einen Kultstatus erlangt haben, wie der FINAL FANTASY⁷³ - Komponist Nobuo Uematsu, der Nintendo-Komponist Koji Kondo, der unter anderem die Spielserien SUPER MARIO BROS. (NINTENDO, ab 1985) und THE LEGEND OF ZELDA (NINTENDO, ab 1986) musikalisch untermalte und Koichi Sugiyama, der die vor allem in Japan populäre Rollenspiel-Serie DRAGON QUEST⁷⁴ (ENIX, ab 1986) vertonte. Die genannten Komponisten haben mit ihren Kompositionen qualitative Pionierarbeit geleistet. Die massenhafte Verbreitung bestimmter Spiele⁷⁵ führte zu einer ebenso großen Verbreitung der in ihnen enthaltenen Musikstücke, so dass diese in das kollektive Gedächtnis der ersten computerspielenden Generation einzogen.

Ein als solcher wahrgenommener Eigenklang von Computerspielmusik konnte sich also erst durch die weite Verbreitung der Hardware, verbunden mit der fortschreitenden musikalischen Untermalung der Spiele bilden, wofür in Computerspiel-Plattformen abseits der Spielhalle erst ab 1982 die Voraussetzung in Form des C64 geschaffen wurden⁷⁶. Auch Belinkie klammert die frühen Anwendungen von Soundchip-Musik aus seiner

⁷⁰ In der musikwissenschaftlichen Literatur steht bei der Betrachtung der Heimcomputer ihr Einsatz als MIDI-Steuerzentrale im Vordergrund. Eine Ausnahme stellt der 1984 in Europa eingeführte MSX-Musik Computer YAMAHA CX5M dar, der die FM-Synthese für den Musiker erstmals anschaulich macht.

⁷¹ Der Programmierung in BASIC widmen sich Dachsel 1984, Barrett / Colwill 1985, Vogel / Scrimshaw 1984 u.a.. Aichner (1986: 198ff) gibt eine umfangreiche Übersicht der bis dahin erschienenen Programme zur Klangbearbeitung.

⁷² Vor der Heimcomputer-Zeit fanden die für die Musik in Computerspielen zuständigen Personen nur selten Erwähnung in den spielbegleitenden Credits. Musikprogrammierung wurde in der Frühzeit der Computerspiele kaum als kreative, künstlerische Tätigkeit angesehen.

⁷³ Die FINAL FANTASY-Serie des japanischen Herstellers SQUARE wird seit 1987 fortgesetzt und ist mittlerweile (2005) bei über fünfzehn verschiedenen Veröffentlichungen auf diversen Systemen angelangt. Nobuo Uematsu hat seit dem ersten Teil die Musik für die meisten Teile komponiert.

⁷⁴ In Amerika sind der erste Teil der Serie sowie die Fortsetzungen unter dem Namen DRAGON WARRIOR erschienen, in Europa wurden die NES-Versionen nicht veröffentlicht.

⁷⁵ Sheff (1993: 72) zufolge wurden von den zwischen 1985 und 1991 produzierten acht Super Mario-Spielen in jenem Zeitraum weltweit über 60 Millionen Exemplare verkauft. Laut dem Webzine „gamecubicle“ wurden bis 2001 insgesamt 120 Millionen Mario-Spiele für die verschiedenen NINTENDO-Plattformen abgesetzt, die allesamt neben jeweils neuen Musikstücken auch Neubearbeitungen der alten Themen enthalten (http://www.gamecubicle.com/features-mario-units_sold_sales.htm; 15.06.2005).

⁷⁶ Die Konsolen der ersten Generation, ATARI VCS, ATARI 5200, INTELLIVISION und COLECOVISION boten einerseits nicht genügend Ressourcen, um aufwendige Musik-Routinen zu implementieren, andererseits sind die Klangerzeuger im Vergleich zum SID und 2A03 weitaus limitierter.

Betrachtung von Computerspielmusik aus: „*The history of game music begins in earnest with the Nintendo Entertainment System*“ (Belinkie 1999: o.S.).

Durch die genannten Faktoren gelten die Jahre der Plattformen C64 und Famicom / NES sowie der weniger erfolgreichen Konkurrenzprodukte von Mitte bis Ende der 1980er Jahre als 'goldene Jahre' (vgl. Tozuka 1999: 53) der Soundchip-Computerspielmusik, die vor allem durch den kreativen Umgang mit Limitierungen charakterisiert wird.

Als Abschluss des Kapitels über die auf einfachen Wellenformgeneratoren basierenden PSG-Soundchips, gibt die folgende Tabelle⁷⁷ eine Übersicht über die hardwareseitig implementierten Fähigkeiten.

	Atari TIA	Atari CO12294 Pokey	GI AY-3891x (YM 2149, YM 2203, YM 2610)	TI SN 76489 AN	MOS 6581 SID	Ricoh 2A03	Ricoh DMG System
Verwendet ab	1977	1979	1979	1982	1982	1984	1989
Verwendet in	Atari VCS 2600, 7800	Atari 400 / 800, XL / XE, 5200, Arcade	Intellivision, MSX, Vectrex, Amstrad CPC 464, Atari ST, Arcade	Colecovision, SEGA Master System & Mega Drive, IBM PCjr, Arcade,	Commodore C64 und Nachfolger / Kompatible	Nintendo Entertainment System / Famicom	Game Boy, GB Color, GB Advance
Klang-erzeugungs-verfahren	DOs	DOs	DOs	DOs	DOs / subtraktive Synthese	DOs, PCM	DOs, Wavetable
Anzahl und Art d. Kanäle	2, div. WF	2-4, Pulse o. Noise, auch gemischt	3: 3 Pulse + Noiseanteil (Noise wird Kanälen hinzu gemischt)	4: 3 Pulse, 1 Noise	3 (wählbar): Sine, Tri, Saw var. Pulse, Noise	4/5: 1 Tri, 2 Pulse, 1 Noise, 1 PCM (nicht in Famicom)	2 Pulse, 1 Wavetable (32 Werte, 4 Bit), 1 Noise
Tonhöhen pro Kanal	32	256 (4 Kanäle) oder 65536 (2-3 Kanäle)	4095	1023	65536	2048	1024
Hüllkurven, -dauern	nicht vorhanden; 16 versch. Lautstärke-werte	nicht vorhanden; 16 versch. Lautstärke-werte	10 versch. aus Saw- und Tri – LFO wählbar. Dauer eines Intervalls: 0,128 ms - 8,33s (Clock = 2MHz)	nicht vorhanden; 14 verschiedene Lautstärke-Werte	1 ADSR-Hüllkurve pro Kanal, Attack, Decay / Release: jew. 16 Werte, 16 Sustainwerte, 16 Lautstärke-werte	25 versch. Tondauern. Pulse & Noise: 16 Releasezeiten, Tri: Release zw. 4ms u. 0,5 s. Sustain möglich.	nicht vorhanden
Effekte	nicht vorhanden	8 Mischungsverhältnisse von Pulse und Noise wählbar, Highpass-Filter optional Tweak: 4 Bit-Sampling	Per Hüllkurve: Tremolo, Vibrato, AM Tweak: 4 Bit-Sampling	Tweak: 4 Bit Sampling	Ringmodulation, Hard Sync, Lowpass, Bandpass o. Highpass -Filter, Wellenform und Hüllkurve von Kanal 3 auslesbar Tweak: 4 Bit-Sampling	Pulse: 4 Pulsbreiten (12,5, 25, 50, 75 %), Sweep-Effekt; Pulse & Noise: Decay-Phase loopbar	Pulse: 4 Pulsbreiten (12,5, 25, 50, 75 %), Sweep-Effekt

Tabelle 4 Die verschiedenen PSG-Soundchips im Vergleich

⁷⁷ Verwendete Abkürzungen: DO = Digitaler Oszillator, WF = Wellenform, PWM = Pulse Width Modulation, PCM = Pulse Code Modulation, AM= Amplitudenmodulation, LFO= Low Frequency Oscillator,

Exkurs: Die japanische Soundtrackkultur

Einen besonderen Stellenwert hat die Computerspielmusik schon früh auf dem japanischen Markt erlangt. Weder in Europa noch in Amerika ist das Phänomen in vergleichbarer Weise in die Populärkultur integriert. Bereits 1984 wurde mit dem von Haruomi Hosono produzierten Tonträger „Video Game Music“ auf dem Label „Yen“⁷⁸ die Musik früher Arcade-Spiele des japanischen Herstellers NAMCO (der Jahre 1980-1984) erstmals einer breiten Öffentlichkeit zugänglich gemacht (vgl. Tozuka 1999: 10). Die auf dem Tonträger fixierten Musikstücke sind die aufgenommenen Spielmusiken von NAMCO-Spielen wie PAC-MAN (NAMCO, 1980), DIG DUG, XEVIOUS und POLE POSITION (alle drei: NAMCO, 1982) als Original-Version. Die beiden 1986 erscheinenden Schallplatten „Famicom Music“⁷⁹ und „Kumikyoku Dragon Quest“⁸⁰ stellen weitere wichtige Entwicklungen für die japanische Rezeption von Spielmusik dar. „Kumikyoku Dragon Quest“ verkauft sich im Erscheinungsjahr bereits 260.000 mal (vgl. Tozuka 1999: 30). Laut dem „Dictionary of Computer and Videogames“⁸¹ geht GMO 1989 in die Firma Scitron über, einige Hersteller wie KONAMI und NAMCO wechseln daraufhin zu anderen Labels⁸².

Soundtracks zu Videospielen werden in Japan somit seit gut 20 Jahren ganz selbstverständlich auf herkömmlichen Tonträgern veröffentlicht und gekauft. Sie erscheinen dabei entweder für Orchester arrangiert oder als „Original Soundtrack Version“. Letztere besitzen häufig den Untertitel „Memorial Album“, um die Erinnerungsfunktion der Musik an das erlebte Spiel zu unterstreichen (vgl. Belinkie 1999, Leonard 2001). Auch in Europa werden seit Ende der 1990er Jahre immer mehr Soundtracks zu Video- und Computerspielen veröffentlicht. Ihre Marktpräsenz ist verglichen mit Japan jedoch marginal, häufig erscheinen sie z.B. im Rahmen von sog. „Special Edition“-Angeboten.

⁷⁸ „Video Game Music“, Yen / Alfa: YLR 20003, VÖ: 1984.

⁷⁹ „Famicom Music“ enthält Stücke aus FAMILICOM (NES)-Spielen, GMO / Alfa: 28XA-69, VÖ: 1986.

⁸⁰ auch „Dragon Quest Suite“, Bandai Music: BY30-5121, VÖ: 1986.

⁸¹ http://www.explore-games.com/computer_and_video_games/v/video_game_music.html (08.04.2005).

⁸² Eine Übersicht mit Suchfunktion der (vor allem in Japan veröffentlichten) Spielmusik-Tonträger findet sich auf der „Game Music Revolution“ - Webseite: <http://gmronline.com> (15.06.2005).

2.5 FM-Synthese und Sampling: neue Möglichkeiten der Klangerzeugung

Der technischen Entwicklung folgend kommen ab 1984 die auf dem MOTOROLA 68000er Mikroprozessor basierende Heimcomputer auf den Markt: APPLE MACINTOSH, COMMODORE AMIGA und ATARI ST⁸³.

Die erfolgreichsten Spielkonsolen dieses als '16 Bit-Ära' bezeichneten Zeitraums sind das SEGA MEGADRIVE (US/JP:GENESIS) von 1988 und das NINTENDO SUPER NES (JP: SUPER FAMICOM) von 1990. Neben diesen erscheinen auf dem japanischen wie amerikanischen Markt das PC-ENGINE von NEC (1987 JP / 1989 US: TURBO GRAFX 16), welches zwar einen 16 Bit-Grafikchip, jedoch weiterhin eine 8 Bit-CPU besitzt, sowie eine für damalige Verhältnisse hochwertige Automatenhardware für den Heimgebrauch, das SNK NEO-GEO (1990)⁸⁴.

Neben einer den damaligen Konkurrenten (also vor allem dem NES, SEGA MASTER SYSTEM und den 8 Bit-Heimcomputern) weitaus überlegenen Grafikhardware besitzt die Mehrheit dieser neuen Plattformen auch leistungsfähigere Verfahren der Klangerzeugung, die auf Weiterentwicklungen im Bereich der Musikelektronik beruhen. Allein der APPLE MACINTOSH besitzt keinen Soundchip und der ATARI ST ist 'nur' mit dem YM 2149-Chip ausgestattet. Die Zeit der 16 Bit-Plattformen markiert die zweite größere Epoche der Computerspielmusik. Nach Collins (2003a) ist diese Zeit vor allem durch das Ausprobieren verschiedener Konzepte der über einfache PSGs hinausgehenden Klangerzeugung geprägt. Das MEGA DRIVE besitzt den SN 76489 AN für Soundeffekte und seine Abwärtskompatibilität zum MASTER SYSTEM. Für die Musik enthält es einen FM-Chip der Firma YAMAHA, wie er ähnlich auch in vielen der Spielautomaten der 16 Bit-Ära oder dem SNK NEO GEO zu finden ist. Die Klangerzeugung des SUPER NES basiert ebenso wie die des AMIGAS auf Sample Playback. Es enthält dafür einen 16 Bit-DSP (Digitalen Signal Prozessor), der eine große Freiheit der Klanggestaltung mit einer aufwendigen Programmierung über einen eigens dafür enthaltenen Prozessor verbindet. Der AMIGA selbst bietet als erster Heimcomputer serienmäßig die mehrkanalige Wiedergabemöglichkeit von digitalisiertem Audiomaterial.

Im Folgenden wird die Klangerzeugung des in Europa kaum erhältlichen PC-ENGINE vorgestellt, die durch die gleichzeitige Integration der verschiedenen Verfahren den Umbruch von 8 Bit- zu 16 Bit-Generation bei der Soundhardware markiert. Die FM-Chips von YAMAHA werden Erwähnung finden, sie lösen die meist noch aus der ersten Generation stammenden, PSG-Soundchips in den Spielautomaten ab. Danach wird auf den AMIGA-Heimcomputer und seine Fähigkeiten im Bereich des Sampling ebenso eingegangen, wie auf die beiden großen Konkurrenten im Bereich der 16 Bit-Spielkonsolen: Das NINTENDO SUPER NES und das SEGA MEGADRIVE / GENESIS. Auf eine vergleichbar detaillierte Vorstellung der Technik wie bei den PSG-Soundchips wird jedoch verzichtet, weil die Klangerzeugungsverfahren an anderer Stelle bereits ausreichend dokumentiert sind und weniger Eigenarten besitzen als die PSGs mit ihren deutlichen Beschränkungen.

⁸³ Der APPLE MACINTOSH als erstes auf dem 68000er Prozessor basierendes Computersystem ist bereits 1984 eingeführt worden. Die auch aufgrund des weitaus niedrigeren Preises für Computerspiele wichtigeren Plattformen COMMODORE AMIGA und ATARI ST erscheinen 1985.

⁸⁴ Einen detaillierten Überblick der verschiedenen 16 Bit-Hardwareplattformen im Hinblick auf die Spieleentwicklung bietet Forster 2005: 96ff.

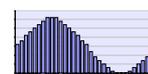
2.5.1 Das NEC PC-Engine: zwischen PSG-Sound und Sampling

Die Soundhardware des PC-ENGINE markiert im Kanon der verschiedenen Soundchips einen Zwischenschritt, bietet es doch erstmals die Möglichkeit, auf sechs Kanälen 32 Byte lange Wellenformen mit einer 5 Bit-Auflösung abzuspielen⁸⁵, ein Verfahren, welches dem Wavetable-Kanal des GAME BOYS ähnelt. Die Klangerzeugung erfolgt dabei nicht in einem Soundchip, sondern ist Teil des HuC6280-Custom Chips (vgl. Forster 2002: 129), dem Hauptprozessor der Konsole. Sowohl Sampling als auch FM-Synthese bietet die Klangerzeugung optional an, wodurch der Soundchip aus technikhistorischer Perspektive interessant wird: Die Kanäle 0 und 1 lassen sich zu einem FM-Kanal zusammenfassen, außerdem darf jeder einzelne Kanal in den „Direct D/A“-Modus geschaltet werden, wodurch nicht der 32 Byte große Sample Buffer ausgelesen wird, sondern die Klangdaten direkt vom Speichermodul zum Soundmixer geschickt werden. Dieses Verfahren wird aufgrund der beschränkten Speicherkapazität der Spielmodule vor allem für kurze Sounds, in größerem Rahmen erst auf den CD-ROM basierten Ausbauoptionen und Nachfolgekonsolen⁸⁶ des PC-ENGINE eingesetzt. Die Kanäle 4 und 5 können optional weißes Rauschen wiedergeben. Wie auch bei den in den vorigen Kapiteln behandelten Soundchips werden die Tonhöhen des HuC6280 nach dem Frequenzdivisionsverfahren⁸⁷ berechnet, welches nach Clifford (o.J.) zu 4095 verschiedenen Tonhöhen zwischen 27,3 Hz und 111,871 kHz führt.

NEC PC-Engine (1987)	
CPU; Taktfrequenz	HuC6280; 7,16 MHz
RAM	8 kB + 64 kB VRAM
ROM	-
Datenträger	HuCards
Kapazität	bis 2,5 MB

Die Klangcharakteristik der auf den Wavetable-Wellenformen basierenden Musik auf dem PC-ENGINE ähnelt der mit denselben Klangerzeugungsverfahren arbeitenden digitaler Synthesizer der 1980er Jahre⁸⁸. Viele der in Spielmusiken auf dem PC-ENGINE verwendeten Wellenformen sind von ihrem Charakter her auch den von FM erzeugten relativ nah. So werden z.B. häufig glockenartige Klangfarben benutzt, ohne dass diese jedoch ähnlich komplexe Klangverläufe wie die FM-Klänge besitzen. Es werden auch mit den PSGs vergleichbare, simple Wellenformen benutzt. In online Spielmusik-Archiven finden sich eine Reihe von Stücken von für das PC-ENGINE portierten Arcade-Spielen, bei denen sich die z.B. gegenüber den NES-Versionen erweiterten Möglichkeiten der Klanggestaltung deutlich hörbar sind. Ein gutes Beispiel für einen derartigen Vergleich von Spielmusik auf verschiedenen technischen Plattformen sind die Adaptionen berühmter Themen der klassischen Musik in den auf mehreren Konsolen erschienenen Umsetzungen des Spielautomaten-Titels PARODIUS (KONAMI, 1990), die deshalb im Hörvergleich (Kapitel 3.4) verwendet werden.

⁸⁵ Clifford führt folgendes Beispiel für Hexadezimal-Werte an, die eine quasi-Sinuswelle beschreiben würden: 0f,12,15,17,19,1b,1d,1e,1e,1e,1d,1b,19,17,15,12,0f,0c,09,07,05,03,01,00,00,00,01,03,05,07,09,0c In Dezimalzahlen übersetzt und grafisch repräsentiert: 16,18,21,23,25,27,29,31,31,31,29,27,25,23,21,18,16,12,09,07,05,03,01,00,00,00,01,03,05,07,09,12.



⁸⁶ Nachfolger wie das PC-ENGINE SUPER GRAFX (1989), die CD-ROM-Erweiterung PC-ENGINE CD-ROM 2 (1988) bis zum PC-ENGINE DUO-RX (1994) erscheinen allesamt nur in Japan (vgl. Forster 2005 S.119).

⁸⁷ Laut Clifford (o.J.) lautet die Formel für die Tonhöhenberechnung: $F_{out} = 3580000 / 32 * N$, wobei N dem 12 Bit-Wert des Tonhöhen-Registers entspricht. Eventuell ist die halbierte Taktfrequenz des PC-ENGINEs von Clifford auf 3,58 MHz gerundet worden, womit die Werte nicht genau wären.

⁸⁸ Wie z.B. dem KORG DW-8000. Ähnlich umfangreiche Möglichkeiten zur Klangbearbeitung durch Filter etc. bietet die Klangerzeugung des PC-ENGINE jedoch nicht.

2.5.2 FM-Synthese: die Yamaha YM-Soundchips

1983 wird der erste professionelle Synthesizer mit Frequenzmodulations-Technologie⁸⁹, der YAMAHA DX 7, auf dem Markt eingeführt, welcher bis heute zu den meistverkauftesten elektronischen Musikinstrumenten zählt (Ruschkowski 1998: 304f)⁹⁰. Bereits Ende der 1960er Jahre entwickelt John Chowning das Verfahren an der Stanford University. Die Lizenzrechte werden an YAMAHA verkauft, wo das Verfahren zur Marktreife gebracht wird. Neben ihrem Hauptanwendungsgebiet in programmierbaren, digitalen Synthesizern werden die von YAMAHA gefertigten FM-Chips bald auch in Spielautomaten verbaut.

Die relevantesten Modelle hiervon sind die Typen YM 2151 OPM, YM 2203 OPN, YM 2413 OPLL, YM 2610 OPNB, YM 3526 OPL sowie YM 3812 OPL-II⁹¹. Die Chips unterscheiden sich vor allem in der Anzahl erzeugbarer Stimmen und im Grad der Komplexität der Klangerzeugung (Anzahl der Operatoren pro Stimme und verfügbare Algorithmen). Unterschiede bestehen außerdem beim Umfang der Eingriffsmöglichkeiten und den integrierten Preset-Instrumenten.

Der YM 2203 ist als dreistimmiger FM-Soundchip der technisch einfachste unter besitzt sowohl eine FM-Klangerzeugung als auch jene des YM 2149. Der YM 2612 besitzt ebenfalls die SSG-Klangerzeugung, beide sind damit abwärtskompatibel zu für die AY-Chips programmierter Software.

Der YM 2151 wird ab 1984 vor allem in Arcade-Automaten eingesetzt und bietet später durch die von Earl Vickers entwickelte „Sinusoidal Modeling“-Algorithmen die Möglichkeit der Sprachsynthese; ein Gebiet, das in vielen Automaten der 1980er Jahre von speziell entwickelten Speech-Synthesis Chips wie dem TI 5520, dem SSI 263 oder dem VOTRAX SC-01 abgedeckt wird (vgl. Brandon / Fuller 1999: o.S.).

Neben der Verwendung in Arcade-Automaten basiert auch der ab Mitte der 1980er Jahre entstehende Markt der PC-Soundkarten in erster Linie auf FM-Soundchips der Firma YAMAHA (vgl. Kapitel 2.7.4) Auch 16 Bit-Spielkonsolen werden mit den YM-Chips ausgestattet, so enthält das SEGA MEGA DRIVE den FM-Soundchip YM 2612 OPN (2). Dieser besitzt die Möglichkeit, auf einem Kanal PCM-kodierte Samples in 8 Bit-Auflösung abzuspielen. Das mit der gleichnamigen Automaten-Hardware identische SNK NEO-GEO enthält wie auch viele TAITO-Boards der späten 1980er Jahre den Soundchip YM 2610, welcher eine sechskanalige FM-Klangerzeugung (à vier Operatoren), die o.g. AY-Kompatibilität sowie die Möglichkeit der zweikanaligen Wiedergabe ADPCM-kodierter 4 Bit-Samples besitzt. Eine Übersicht der verschiedenen auf FM-Synthese basierenden Chips der Firma Yamaha bietet folgende Tabelle⁹²:

⁸⁹ Die Funktionsweise der FM-Technologie wird anschaulich erklärt in Ruschkowski 1998 S.295ff. Für detailliertere Informationen empfiehlt sich die Lektüre von Bristow / Chowning 1986.

⁹⁰ Vor dem DX 7 erscheinen 1982 die ebenfalls auf FM-Synthese basierenden YAMAHA-Synthesizer GS1 und GS2.

⁹¹ Laut der „Classic Arcade Database“ (http://ie.ath.cx/cad/index_audio.htm) gibt es allein ~150 verschiedene Boards mit dem YM 2151, ~ 125 mit dem YM 2203, ~ 173 mit dem YM 2610, ~ 39 mit dem YM 3526 und mehr als 48 Boards mit dem YM 3812, wobei Mehrfachnennungen der Datenbank durch verschiedene Revisionen und Bootlegs in die o.g. Zahlen nicht mit einbezogen wurden.

	YM 2151 OPM	YM 2203 OPN	YM 2413 OPLL	YM 2610 OPNB	YM 2612 OPN(2)
Verwendet ab	1984	1984	1987 (?)	1987 (?)	1988
Verwendet in	Arcade, div. jap. Personal Computer, Yamaha SFG-01 FM (MSX)	Arcade, div. jap. Personal Computer	Arcade, div. MSX-Nachfolger	SNK Neo-Geo, Arcade (v.a. Taito-Boards)	SEGA Mega Drive
Klang-erzeugung	FM	PSG (AY- / SSG kompatibel), FM	FM	PSG (AY- / SSG kompatibel), FM & PCM	FM & PCM
Art und Anzahl der Kanäle	8 FM (4 OP),	3 Pulse & Noise, 3 FM (4OP)	9 FM-Kanäle (à 2 OP) o. 6 FM (2 OP) + 5 Percussion (BD, SD, HH, TC, TOM)	3 Pulse & Noise, 6 FM-Kanäle à 4 OP, 2 ADPCM-Kanäle (4 Bit)	6 FM-Kanäle à 4 OP, Kanal 6 kann optional 8 Bit Samples wiedergeben
FM-Algorithmen	8	8	2	8	8
Effekte	Vibrato, AM, De-Harmonize, Feedback	Feedback	Feedback, Vibrato, AM	Feedback, Vibrato, AM	Feedback, Vibrato, AM
Hüllkurven	ADDR	PSG: SSG FM: ADSR o. SSG	ADSR / ADR	PSG: SSG FM: ADSR / SSG; ADPCM: ADDR	ADDR
Besonderheiten	Operator 4 von Kanal 8 kann Rauschen produzieren; durch „Sinusoidal Modeling“ auch Sprachwiedergabe möglich	Für Kanal 3 können Frequenzen der Operatoren frei zugewiesen werden (mehr Möglichkeiten zur Generierung dissonanter Klänge)	Zwei Wellenformen: Sine und Half-Sine. Nur Platz für einen programmierten Sound; dieser oder einer von 15 Preset-Klängen muss von jedem Kanal verwendet werden.		Für Kanal 3 können Frequenzen der Operatoren frei zugewiesen werden (mehr Möglichkeiten zur Generierung dissonanter Klänge)

Tabelle 5 Die verschiedenen YM-Chips (ausgenommen die OPL-Reihe, welche in Kapitel 2.6.3 verglichen wird)

2.5.3 Homesampling: Der Commodore Amiga

Der COMMODORE AMIGA kommt 1985 nach dem APPLE MACINTOSH und dem ATARI ST als dritter 16 Bit-Heimcomputer auf den Markt. Wie seine direkten Konkurrenten basiert auch der AMIGA auf der 68000er CPU von MOTOROLA. Er besitzt mehrere von dem Ingenieur Jay Miner und seinem Team entwickelte Custom-Chips⁹³. Miner hatte bereits die in den ATARI Heimcomputern 400 / 800 verbauten Spezial-Chips, zu denen auch der Pokey gehört, mitentwickelt. Von den Custom-Chips ist der Paula 8364 bzw. schlicht Paula genannte Chip für den Sound und die Controller-Verwaltung (Abfrage der Joystickports sowie I/O des Diskettenlaufwerks) zuständig. Die Sound-Hardware des AMIGA besitzt von Haus eine Fähigkeit, die in voriger Spielhardware nur beim PC-ENGINE in vergleichbarer und beim NES in rudimentärer Form implementiert gewesen ist: Sample-

Commodore Amiga (1985)	
CPU; Taktfrequenz	MC 68000; 7,2 MHz
RAM	256 kB bis 1 MB
ROM	bis 512 kB
Datenträger	Diskette
Kapazität	880 kB

⁹² Die Klangerzeugung der PSG-Parts von YM 2203 und YM 2610 (Pulse & Noise) sind identisch mit dem AY-Chip bzw. YM-2149 (= SSG). Die Angaben in den Zeilen „Algorithmen“ und „Effekte“ beziehen sich nur auf die FM-Klangerzeugung. Optional kann die Hüllkurve des SSG auch für die FM-Kanäle benutzt werden. Bei der Hüllkurve des YM 2413 wird per Status Bit zwischen „Percussive Tone“ (ADR) und „Sustained Tone“ (ADSR) unterschieden. Die Informationen zum YM 2610 lagen nur in japanischer Sprache vor. Daher basieren die vorliegenden Informationen teilweise auf Schlussfolgerungen aus den Romanji-Anteilen des japanischen „Application Manuals“ und den gesicherten Informationen der anderen YM-Chips.

⁹³ Die Custom-Chips „Agnes“ und „Denise“ sorgen für die der Konkurrenz überlegenen Grafik-Fähigkeiten des AMIGAS, „Paula“ ist für die Klangerzeugung zuständig. Der sog. „Blitter“ dient als Beschleuniger für Vektoren und Polygone (vgl. Forster 2005: 107f).

Playback⁹⁴. So unterstützt der AMIGA die vierstimmige Wiedergabe von gesampelten, digitalen Wellenformen (in 8 Bit-Auflösung mit variabler Samplingfrequenz von 20 bis 28867 Hz) ohne eine auf Tongeneratoren basierende Klangerzeugung zu besitzen. Die Samplingfrequenzen und Loopingpunkte lassen sich laut Hülbeck recht flexibel programmieren, wodurch bereits auf dem AMIGA Time-Stretching ähnliche Effektklänge möglich waren⁹⁵.

Für jeden Samplekanal können Tonhöhe, Lautstärke und Panning bestimmt werden, des Weiteren gibt es ein für alle Kanäle zuschaltbares Tiefpassfilter mit einer Eckfrequenz von 7 kHz, um ungewollte Frequenzen zu eliminieren. Hardwareseitig sind zudem Modulationen möglich, um Tremolo- und Vibrato-Effekte zu erzeugen (vgl. Collins 2003b). Bis zur breiten Verwendung des MOD-Formats werden für AMIGA-Musik wie auch beim PC-ENGINE oder SNES vor allem kurze, geloopte Teile von periodischen Wellenformen genutzt, um verschiedene Klangfarben zu erzeugen, eine Technik, die auch in den Chiptunes für AMIGA (s. Kapitel 4.4) verwendet wird. Spätere Musik für AMIGA-Spiele wird aufgrund des häufigen Einsatzes von gesampeltem Audiomaterial für Schlagzeugsounds oder verzerrte Gitarren, der bei den bisherigen Klangerzeugern nicht oder nur schwer möglich war, als die den Konkurrenzplattformen (v.a. ATARI ST, C64 und PC) klanglich überlegene angesehen⁹⁶.

Entwicklung der Tracker und MOD-Dateiformate

Ein wahrer Boom der AMIGA-Musik wird durch das Programm ULTIMATE SOUNDTRACKER ausgelöst. Die erste Version und das dafür entwickelte MOD- (von „Module“) Dateiformat werden im August 1987 von dem bei der Spielefirma EIDOS angestellten Karsten Obarski veröffentlicht. Der SOUNDTRACKER legt als tabellarische Kompositionshilfe, in welcher sich vierstimmige Arrangements aus Samples erstellen lassen, den Grundstein für eine ganze Szene (die sog. Tracker-Szene), die in den 1990er Jahren für eine erste „Welle der Euphorie“ im Bereich der digitalen Musikproduktion sorgt (Tasajärvi 2004b: 35) bemerkt. Das MOD-Format enthält neben den einzelnen Samples die Informationen, wann und wie diese mit welchen Parametern gespielt werden sollen. Die Samples der jeweiligen MODs liegen ebenso wie die musikalischen Strukturen und Kniffe in der Programmierung der vier Kanäle und der Patterns grundsätzlich der Verwendung durch andere offen. Der Remix eines geöffneten MODs im Tracker, der somit gleichzeitig Produktions- und Abspielsoftware

⁹⁴ Wie erwähnt können sowohl der SID des C64 als auch der Pokey, AY-Chip und SN 76489 AN über ein ausreichend schnelles An- und Ausschalten des 4 Bit D/A-Lautstärke-Registers einen de-facto Sample-Kanal bilden. Die Soundqualität dieses Verfahrens blieb jedoch meist unbefriedigend. Auch gibt es für die meisten 8 Bit-Heimcomputer wie den C 64 und den SINCLAIR SPECTRUM bereits Erweiterungen, die Sampling im kleinen Rahmen ermöglichen. Allen Verfahrensweisen ist gemein, dass sie nicht von den Entwicklern der Basishardware vorgesehen waren.

⁹⁵ Er verweist im Interview auf die Titelmusik zu R-TYPE (IREM, 1987; AMIGA: 1989; ♪102), in der diese Effektklänge deutlich hörbar sind (Hülbeck 2005b).

⁹⁶ Die von Chris Hülbeck komponierte Hintergrundmusik zu TURRICAN II (RAINBOW ARTS, 1991; ♪103) ist eine bis heute viel beachtete Spielmusik. Hülbeck benutzt auf dem AMIGA ein eigenes Musikformat namens TFMX-Format. Die Komposition geschieht ähnlich wie in seinem SOUNDMONITOR-Programm für C64, indem einzelne arrangierte Patterns den vorhandenen Kanälen zugeteilt werden können. Bei normalen Trackern sind in einem Pattern immer alle Spuren gemeinsam enthalten, was weniger Flexibilität und einen höheren Speicherverbrauch bedingt, da die Musikdaten dadurch mehr Redundanzen enthalten (Hülbeck 2005b).

darstellt⁹⁷, liegt nur einen Mausklick entfernt, die Grenze zwischen Produzenten und Konsumenten wird aufgeweicht. MODs werden Standard in der Hacking- und Demo-Szene, um u.a. Demos⁹⁸ der jeweiligen Gruppe klanglich zu untermalen. Auch der Begriff Chiptunes stammt ursprünglich aus der Demo-Szene und kennzeichnet die MODs, die auf kurzen, gesampelten Wellenformen beruhen⁹⁹. Ab Mitte der 1990er Jahre werden Tracker mehr und mehr auf dem PC verwendet, wodurch die Limitierungen der Amiga-Hardware für Tracker-Musik hinfällig werden. Durch die Entwicklung der Tracker (vielfältige Unterformate und Konkurrenz-Tracker erschienen in den Folgejahren¹⁰⁰) lernt eine ganze Generation an Computern und Musik interessierter Anwender die Verfahrensweisen elektronischer, samplebasierter Musik kennen. Außerdem entsteht durch die PC-Portierungen und die Möglichkeiten des WWW eine – wenn auch auf subkulturelle Kreise begrenzte – OpenSource-Gemeinschaft, deren Mitglieder das Interesse an elektronischer Musik verbindet und die laut Tasajärvi unabhängig von der Demo-Szene ist (vgl. Tasajärvi 2004b: 35)¹⁰¹. Neben Chiptunes und einer großen Zahl an Techno-Stilen orientierten Stücken finden sich im Fundus dieser Tracker-Szene auch solche, die die Möglichkeiten des Samplings äußerst kreativ ausnutzen, wie das Stück „Catch the Goblin!“¹⁰² des Künstlers „Skaven“ (♫011). Viele bekannte Komponisten für Computerspielmusik wie Jeroen Tel, Bjorn Lynne, Olof Gustaffson, Andrew Barnabas, Jochen Hippel, Jesper Kyd und Frederic Motte gelangen darüber hinaus über die Demo- oder Tracker-Szene in die Spiele-Industrie. Verschiedene Datei-Formate, Inkompatibilitäten zwischen diesen, sowie eine mangelnde Unterstützung durch die Industrie lassen Tracker-Musik jedoch nicht zu einer breiten Lösung für Endnutzer werden. Mit dem Aufkommen und nachhaltigem Erfolg des MP3-Dateiformats als quasi-Standard für digitale Musik wird mit Trackern produzierte Musik immer häufiger im MP3-Format verbreitet. Dabei gehen jedoch wichtige Eigenschaften des offenen Formats wie Editierbarkeit und geringer Speicherplatzverbrauch verloren.

Bei der Entwicklung von Computerspielmusik werden z.B. für das NINTENDO 64 (NINTENDO, 1996) in großem Maße MOD ähnliche Dateiformate und Audio-Engines verwendet, da der geringe Modul-Speicherplatz kaum eine andere Möglichkeit für Digital Audio-Dateien zulässt. Auch auf dem PC wurden bis Ende der 1990er Jahre MOD-Dateien z.B. von EPIC MEGAGAMES (in UNREAL, 1998) oder dem aus der Demo-Szene stammenden Programmiererteam DIGITAL ILLUSIONS (PINBALL DREAMS, 1993 (PC)) verwendet. Tracker werden für den AMIGA bis weit in die 1990er Jahre hinein weiterentwickelt, beispielsweise durch Sample-Editoren oder software-

⁹⁷ In der Folgezeit wurden auch spezialisierte Player ohne Eingriffsmöglichkeit entwickelt, vor allem, um den vielen Formaten handhabbar zu werden. Der DELI PLAYER (<http://www.deliplayer.com>; 14.03.2005) ermöglicht die Wiedergabe verschiedener MOD-Formate und spielt ebenso Musikdateien für SID, NES, SNES, AY-Chip, AD LIB etc. auf Windows-Plattformen ab. Für den populären Player WINAMP existiert ein die MOD-Wiedergabe ermöglichendes PlugIn namens OLDSK00 (<http://www.winamp.com/plugins/details.php?id=2266>; 18.07.2005).

⁹⁸ Unter Demos versteht man audiovisuelle Demonstrationsprogramme die programmiert werden, um das Renommee der eigenen Crew bzw. Group zu steigern und auf speziellen Partys der Demo-Szene in Wettbewerben andere Crews zu schlagen (vgl. Tasajärvi 2004a). Eine Sammlung von Demos für den AMIGA findet sich beim A.D.A. Amiga Demoscene Archive (<http://www.ada.planet-d.net>; 14.03.2005).

⁹⁹ MOD-Archive finden sich auf <http://www.chiptunes.org> und <http://www.modarchive.com> (07.07.2005)

¹⁰⁰ Vor allem die Programme NOISE TRACKER und PRO TRACKER sowie -für PC- IMPULSE TRACKER, FAST TRACKER sowie SCREAM TRACKER 3 sind als relevant zu erachten, für die Spieleproduktion ist eine Reihe von anderen Formaten entstanden (vgl. z.B. <http://exotica.fix.no/tunes/unexotica/formats/index.html>; 20.07.2005).

¹⁰¹ Auf den Servern von scene.org (<http://www.scene.org>; 20.04.2005) finden sich neben vielen MP3-basierten Netzlabeln auch die Archive der damaligen, auf dem MOD-Format basierenden Music Groups, die Vorläufer der heutigen Netzlabel-Szene sind.

¹⁰² Das Stück wurde 1995 mit dem SCREAM TRACKER 3 (auf PC) komponiert. „Skaven“ ist Mitglied der „Future Crew“: <http://www.futurecrew.com/skaven/> (25.07.2005).

basierte Synthesizer ergänzt oder schlicht in der Menge der speicherbaren Patterns erweitert. Auch gibt es Tracker, die bis zu acht Stimmen wiedergeben können (wobei Arbeit an die CPU abgegeben und so softwareseitig gemischt wird). Die Exotica!-Webseite¹⁰³ nennt auf ihrer „File Formats und Replays“-Übersicht 39 verschiedene Formate inklusive der Entwickler und verfügbaren Player.

Seit Anfang der 1990er Jahre existieren den AMIGA-Trackern technisch weit überlegene Versionen für den PC. Mit der PC-Soundkarte GRAVIS ULTRASOUND erscheint 1991 eine auf das Arbeiten mit Trackern spezialisierte Hardware (Göhler 2004). Mit der Entwicklung virtueller Studio-Umgebungen und dem Aufkommen immer leistungsfähigerer Software in der digitalen Musikproduktion ab Ende der 1990er Jahre verliert die Verwendung von Trackern für die Musikproduktion mit dem Computer allgemein an Bedeutung, auch wenn weiterhin einige Netzlabels und Chiptune-Künstler weiterhin in gängigen MOD-Formaten (MOD, XM, S3M etc.) Musik publizieren. Der aktuelle Tracker RENOISE¹⁰⁴ für WINDOWS XP bietet eine zeitgemäße Adaption der Tracker-Arbeitsweise: Er ist kompatibel mit einer Reihe älterer MOD-Formate und verfügt über eine VST-Schnittstelle, mit der sich virtuelle Instrumente und Studio-Effekte nutzen lassen.

2.5.4 Das Sega Mega Drive

Das MEGA DRIVE kommt 1988 als erster Vertreter einer neuen Generation von 16 Bit-Spielkonsolen auf den Markt¹⁰⁵. Mit dem MASTER SYSTEM war es dem japanischen Hersteller SEGA nicht gelungen, dem Konkurrenten NINTENDO ENTERTAINMENT SYSTEM ernsthaft Marktanteile streitig zu ma-

Sega Mega Drive (1988)	
CPU; Taktfrequenz	MC 68000; 3,58 MHz
RAM	64 kB + 64 kB VRAM
Datenträger	Modul
Kapazität	bis 4 MB

chen. SEGA ist jedoch auf dem Arcade-Automaten-Markt äußerst erfolgreich und veröffentlicht sein MEGA DRIVE (in Japan und Amerika „GENESIS“) gut anderthalb Jahre bevor Nintendo seinerseits eine vergleichbare Konsole (das SNES) auf den Markt bringt. Wie die im Rahmen dieser Arbeit besprochenen 16 Bit-Heimcomputer basiert das MEGA DRIVE auf dem 68000er Prozessor und ist damit dem NES weit überlegen. Nicht nur Grafik und Geschwindigkeit der Spiele, auch die musikalische Untermalung kommt den SEGA-Originalen der Spielhalle¹⁰⁶ sehr nahe. Es besitzt für diesen Zweck neben einen die Audiowiedergabe steuernden Co-Prozessor (Z80) den vor allem für Soundeffekte genutzten SN 76489 AN und einen modernen, sechs-stimmigen YAMAHA YM 2612 FM-Chip¹⁰⁷ mit der Möglichkeit, 8 Bit PCM-Samples wiederzugeben. Die FM-Kanäle besitzen je 4 Operatoren, die sich in acht verschiedenen Algorithmen schalten lassen. Der Kanal für die Sample-Wiedergabe lässt sich anstelle des Kanals 6 einschalten. Das SEGA-Dokument (Sega o.J.) weist für den PCM-Kanal darauf hin, dass, wenn die Software nicht sehr clever geschrieben sei, um ein ausreichendes Timing zu garantieren, eine Benutzung der FM-Kanäle während des Sample-Playbacks unmöglich sei (Sega

¹⁰³ Die Webseite (<http://exotica.fix.no/frames.html>; 13.08.2004) kümmert sich vor allem um die Archivierung der exotischen, an MOD angelehnten Tracker-Formate und -Dateien.

¹⁰⁴ Auf der Webseite (<http://www.renoise.com>; 25.06.2005) befindet sich neben einer kostenlosen Demo-Version auch ein umfangreiches Tutorial für die verschiedenen Funktionen.

¹⁰⁵ Das MATTEL INTELLIVISION besitzt wie erwähnt zwar eine 16 Bit-CPU, sowohl Grafik- als auch Soundhardware sind jedoch mit der von 8 Bit-Computern vergleichbar. Das PC-ENGINE besitzt zwar einen 16 Bit-Grafikchip, basiert jedoch auf einer 8 Bit-CPU.

¹⁰⁶ Einige diese Titel sind ALTERED BEAST (SEGA, 1988), SUPER HANG-ON (SEGA, 1989), GOLDEN AXE (SEGA, 1989) und AFTER BURNER II (SEGA, 1990).

o.J.: 92). Das lässt darauf schließen, dass in Spielen mit FM-Musik und gleichzeitig wiedergegebenen Samples wie bei dem Michael Jackson-Computerspiel MOONWALKER (SEGA, 1990; ♪012) oder TOEJAM & EARL (SEGA, 1991) entweder die Programmierung der Soundchiptreiber sehr gut in dem Programm integriert wurde oder der SN 76489 für die Samplewiedergabe zweckentfremdet wurde, wie in Kapitel 2.2.4 beschrieben. Hülsbeck (2005b) weist jedoch darauf hin, dass das MEGA DRIVE einen vom Soundchip losgelösten D/A-Wandler besitze, der für die Wiedergabe von niedrig aufgelösten Samples verwendet werden kann. Darüber hinaus stellt er heraus, dass die Integration der verschiedenen Technologien im Schaffensprozess äußerst motivierend gewesen sei:

„Die Kombination von FM-, PSG- und Sample-Stimmen ergab eine sehr interessante klangliche Mischung, weil die Verfahren alle ihre Vor- und Nachteile hatten. Das war schon interessant, aber eine Heidenarbeit“ (Hülsbeck 2005a).

Die 1991 und 1994 erscheinenden Erweiterungen, das SEGA MEGA CD und das SEGA 32X ergänzen die Konsole u.a. mit einer überarbeiteten Klangerzeugung (MEGA CD: 8 Kanal PCM-Sample Playback (bis 16 Bit) plus 10 Stimmen FM-Synthese, SEGA 32X: Stereo-PCM; vgl. Collins 2003b: 21). Beide haben am Markt jedoch keinen nennenswerten Erfolg.

2.5.5 Nintendo Super Nes: DSP für ADPCM / Sample Playback

Das SUPER NINTENDO ENTERTAINMENT SYSTEM (in Japan: SUPER FAMICOM) kommt 1990 mit anderthalb Jahren Verzug gegenüber dem direkten Konkurrenzprodukt von SEGA MEGA DRIVE auf den Markt. Die Soundhardware besteht aus dem 8 Bit-Prozessor SONY SPC 700 und einem

Nintendo Super Nes (1990)	
CPU; Taktfrequenz	65C816; 3,58 MHz
RAM	128 kB + 64 kB VRAM
Datenträger	Modul
Kapazität	bis 6 MB

ihm angeschlossenen 16 Bit-DSP (Digitaler Signal Prozessor), die zusammen de facto einen Wavetable-Synthesizer mit der Möglichkeit des Sample-Playbacks darstellen. Der DSP kann acht Stimmen ADPCM-kodierter Samples gleichzeitig wiedergeben, wobei pro Stimme Tonhöhe, Lautstärke, Panning sowie weitere Abspielparameter wie Looping, Sample Interpolation, ein Echo-Effekt, ein Lowpass-Filter und eine ADSR-Lautstärke-Hüllkurve definiert werden können. Außerdem befindet sich im ROM ein vorgefertigtes Set von Klängen, welches standardmäßig in den Speicher geladen wird. Ein Programm im SPC 700 instruiert den DSP, damit die richtigen Samples vom Modul geladen und wiedergegeben werden. Die Samples werden in einem „Bit Rate Reduction“ (BRR) genannten Kompressionsverfahren mit einer Kompressionsrate von 32:9 gespeichert (vgl. Collins: 2003b: 22), um den Speicherbedarf gering zu halten. Der für Musik reservierte

¹⁰⁷ Derselbe Chip wurde laut Collins (2003b: 21) und Weske (2001) in den FM-Keyboard-Synthesizern DX 27 und DX 100 verwendet. Beide Synthesizer sind jedoch achttimmig spielbar. Das „Sega Technical Overview“-Dokument erwähnt eine Ähnlichkeit des Chips zum YM 2151 (der ebenfalls 4 Operatoren pro Stimme besitzt) und derer, die in den Yamaha-Synthesizern verbaut wurden (Sega o.J.: 92). Otten (2004: o.S.) verweist darauf, dass der YM 2612 zweimal die FM-Klangerzeugung des YM 2203 besitze. Aufgrund dieser gegensätzlichen Angaben wird an dieser Stelle auf einen Vergleich mit anderen Soundchip verzichtet. Dennoch sei darauf hingewiesen, dass im bereits genannten Technical Overview - Dokument ein [YM 2149] SSG-Enable Bit erwähnt wird, das jedoch keine Verwendung habe. Das heißt eine Verwandtschaft mit dem die SSG-Klangerzeugung enthaltenden YM 2610 liegt nahe.

Speicher im Soundchip ist laut Hülsbeck (2005b) 64 kB groß, ihm zufolge werden z.B. zwischen unterschiedlichen Levels die Samples für das nächste Stück über einen DMA-Kanal nachgeladen, was weniger als eine Sekunde dauert.

Das Verfahren der Konvertierung eines Samples ins BRR-Format schildert Of the Veldt detailliert¹⁰⁸. Laut ihm und Collins (a.a.O.) sind auf den Spiel-Cartridges 24 MBit für Sound-Daten reserviert. Hülsbeck (2005b) verweist jedoch darauf, dass dieser Wert schlicht die maximale Kapazität von Spielmodulen für das SUPER NINTENDO darstelle und die Entwickler grundsätzlich die Freiheit hatten, den Platz nach Bedarf auch für Samples zu nutzen. Die Wellenformen werden in 16 Bit-Auflösung wiedergegeben, angesichts der Abtastrate von 14,4 kHz bleibt im Hauptspeicher somit gerade einmal Platz für rund vier Sekunden Sample-Playback. (Of the Veldt o.J.). Die Musikstücke werden im Entwicklungsprozess der Spiele i.d.R. in einer MIDI-Umgebung komponiert und danach in ein Programm¹⁰⁹ übersetzt, das der SPC-700 verarbeiten kann. Das SUPER NES ist ebenso wie der Vorgänger nur mangelhaft dokumentiert. Laut Hülsbeck (2005b) stand die Funktionsweise der Klangerzeugung Pate für die in den Folgejahren von SONY entwickelten Konsolen PLAYSTATION 1 und 2, wobei in diesen zahlreiche Beschränkungen der SUPER NINTENDO-Hardware entfallen. In Computerspielmusik für das SNES wurden ebenso wie für den AMIGA häufig gesampelte Instrumental- und Schlagzeug-Klänge verwendet.

2.5.6 Die Bedeutung der Sampling-Soundchips für Soundchip-Musik

Einer auf die Technik der Klangerzeugung fokussierenden Einteilung folgend, fällt auf, dass in den 16 Bit-Plattformen neben FM-Synthese vor allem Sampling als neue Verfahren der Klangerzeugung implementiert werden. Auch in Spielautomaten werden beide Verfahren verwendet. Im Hinblick auf die folgenden Kapitel wird daher abermals darauf hingewiesen, dass vor allem die 8 Bit-Plattformen und ihre Soundchips im Zuge der Chiptunes-Bewegung aufgrund ihres spezifischen Klangs im Sinne einer „Sound Culture“ wieder entdeckt wurden und bis heute in der elektronische Musik benutzt werden. Die neuen Verfahrensweisen FM-Synthese und Sampling besitzen für die „Sound Culture“ von Soundchipmusik ebenso wie für die Chiptunes-Szene weniger Bedeutung, da mit der Möglichkeit der Verwendung von Instrumental-Klängen (FM) bzw. beliebigen digitalisierten Klangquellen die eigene Klanglichkeit von Computerspielmusik nicht weiter unterstützt wird.

¹⁰⁸ Bei der Methode werden jeweils 16 zusammenhängende Samples zusammengefasst, wobei in einem 9 Byte großem Block sechzehn 16 Bit-Samples des Raw-Formats gespeichert werden. vgl. Of the Veldt (o.J.)

¹⁰⁹ Da der SPC-700 ein eigener Prozessor und kein einfacher Soundchip ist, muss ein Assembler-Programm für ihn geschrieben werden, welches die Ansteuerung des DSPs durch den SPC-700 regelt.

Für die Entwicklung der Computerspielmusik ist diese Entwicklung jedoch umso mehr von Bedeutung, da Sampling seit Mitte der 1990er Jahre zum Standard in Computerspielplattformen geworden ist und die auf Sampling basierenden Soundchips somit Wegbereiter darstellen. Eine Übersicht dieser zeitlich neben den YM-Chips von YAMAHA erscheinenden Sampling-Chips findet sich in der nachfolgenden Tabelle, die im Vergleich mit der Übersichtstabelle der frühen PSGs und Soundchips mehr Freiheiten im Hinblick auf die Klanggestaltung offenbart:

	Paula 8364	NEC HuC6280	Sony SPC-700 und DSP
Verwendet ab	1985	1987	1990
Hardware	Commodore Amiga	PC-Engine und Nachfolger	Nintendo Super NES / Famicom
Klangerzeugung	PCM (8 Bit, bis zu ~29 kHz Sample-Rate)	5 Bit Wavetable (32 Werte), PCM, FM, Noise	Wavetable / ADPCM
Art und Anzahl der Kanäle	4 Hardware-Kanäle, später teilweise durch Software-gemischte Kanäle unterstützt.	6 Wavetable / PCM; optional: 1 FM & 4 Wavetable / PCM; Kanäle 5+6 können auch weisses Rauschen wiedergeben.	8
Tonhöhen pro Kanal		4095	
Effekte	Looping, Tremolo, Vibrato	k.A.	Looping, Sample Interpolation, Echo, Lowpass-Filter
Hüllkurven	Software	k.A.	ADSR pro Kanal
Besonderheiten	Sample Rate pro Kanal zw. 20 und 28867 Hz einstellbar, Loop-Punkte und Interrupts sehr flexibel nutzbar.	Neben dem 32 Byte großen Wavetable-Sample Buffer können die Kanäle auf „Direct-DMA“ gesetzt werden. Dadurch werden Sample-Daten direkt von der CPU geschickt und wiedergegeben.	SPC-700 instruiert den 16 Bit DSP, der die eigentliche Klangerzeugung darstellt, ADPCM-Samples werden im BRR-Verfahren komprimiert gespeichert. Keine Sample-genauen Loop-Punkte möglich (Hülsbeck 2005b).

Tabelle 6 Die auf Sampling basierenden Soundchips der 16 Bit-Ära im Vergleich

Im Folgenden wird auf die parallel zur Entwicklung von Spielkonsolen und Arcade-Automaten verlaufende Historie von technischen Voraussetzungen der Computerspielmusik für (IBM-kompatible) Personal Computer eingegangen. Wie Brandon (2003: 197) bemerkt, muss eine ernsthafte Auseinandersetzung mit Musik in Computerspielen die drei Hauptbereiche Spielkonsolen, PCs und Arcade-Automaten trennen, Heimcomputer werden von ihm nicht explizit erwähnt. Diese Trennung ist auch Anliegen dieser Arbeit, wobei die Spielautomaten nur am Rande behandelt werden, bieten sie doch einerseits aufgrund der sehr unterschiedlichen Ausstattung der verschiedenen Boards wenig Vergleichsmöglichkeiten, basieren andererseits jedoch auf denselben Technologien und Soundchips wie Spielkonsolen und Heimcomputer¹¹⁰.

¹¹⁰ In Kapitel 3.1 werden daher primär solche Spiele als Beispiele genutzt, die zuerst als Spielautomaten erschienen sind und später auf zahlreiche andere Plattformen konvertiert wurden.

2.6 Vom PC-Speaker bis General MIDI: Klangerzeugung im PC (1981-1994)

Mit dem Standard des IBM-PCs und der damit sukzessive erfolgten Umwälzung nahezu des gesamten Marktes der Personal Computer durch miteinander kompatible Komponenten verschiedener Anbieter auf Basis einheitlicher Industriestandards, stellt sich den Spieleproduzenten im Hinblick auf die Implementierung von Musik ein grundlegendes Problem: In

IBM PC-AT (1984)	
CPU; Taktfrequenz	Intel 80286; 8 bis 12 MHz
RAM	640 kB
ROM	n.b.
Datenträger	Disketten
Kapazität	360 kB, 1,2 MB

den Anfangsjahren können IBM-PCs über den eingebauten Lautsprecher (PC-Speaker) allein eine einstimmige Rechteckwelle ohne über Tonhöhe und Tondauer hinausgehende Beeinflussungsmöglichkeiten wie Hüllkurven o.ä. wiedergeben, was die Plattform gegenüber den Konkurrenten in Gestalt von Heimcomputern und Spielkonsolen für Computerspiele unattraktiv macht. Erst 1986/87 erscheinen die ersten „Soundkarten“ genannten Erweiterungskarten für den PC. Dieser Entwicklung folgend müssen Spielehersteller neben den verschiedenen Grafik-Standards auch die zunehmenden Möglichkeiten der Musik- und Klangwiedergabe berücksichtigen. Der Zustand einer ausbleibenden einheitlichen Hardware-Lösung für Musik und Sound in Computerspielen auf dem PC dauert bis Ende der 1990er Jahre an. Erst dann wird mit verschiedenen Standards wie DirectX¹¹¹ oder der AC '97-Norm¹¹² dem Problem des Nebeneinanders Abhilfe geschaffen. In diesem Kapitel werden die wichtigsten dieser technischen Standards vorgestellt. Auf die Verfahren der FM-Synthese und des Samplings, auf welchen die meisten Soundkarten basieren, wird nicht detailliert eingegangen, da sie ausreichend an anderer Stelle dokumentiert sind¹¹³.

2.6.1 Der PC-Speaker

Der erste als Heimrechner konzipierte IBM-PC, das Modell 5150, kommt 1981 auf den Markt und begründet damit den neuen Standard von kostengünstigen Personal Computern auf Basis der 8086er Prozessor-Familie von INTEL (vgl. Forster 2005: 98f). Die offene Architektur erlaubt es auch anderen Herstellern, PCs und Erweiterungen nach den IBM-Spezifikationen zu bauen. Diese PCs und ihre Klone enthalten als Sound-Hardware nur einen „PC-Speaker“ genannten kleinen Lautsprecher, welcher eine Pulswelle wiedergibt. Standardmäßig sind die Computer mit dem von IBM entwickelten BASICA auf der Basis von PC-DOS ausgestattet, in welchem ähnlich wie in BASIC-Versionen für viele Heimcomputer die Kommandos BEEP, SOUND und PLAY für die Steuerung der Klangwiedergabe zur Verfügung stehen. Der BEEP-Befehl bewirkt nicht mehr als einen normalen Hinweisklang ohne Modifikationsmöglichkeiten, weswegen er musikalisch ohne Nutzen bleibt. Mit dem SOUND-Befehl können immerhin Frequenz und Dauer des gespielten Tons festgelegt

¹¹¹ Auch wenn DirectX eine einheitliche Schnittstelle darstellt, unterscheiden sich aktuelle Soundkarten weiterhin durch die Unterstützung verschiedener EAX-Versionen, Surround-Formaten o.ä. (vgl. Boer 2003: 6ff, 27f)

¹¹² Die genauen Spezifikationen des AC'97-Codecs finden sich auf der Homepage von INTEL (www.intel.com/design/chipsets/audio/ac97r22.pdf; 14.03.2005). Die Verwendung dieser und ähnlicher Chipsätze auf den Mainboards sowie die Integration des „MICROSOFT GS Wavetable Synthesizers“ in das Betriebssystem WINDOWS XP ließen die meisten Soundkarten obsolet werden

¹¹³ Wiederum sei zu der FM-Synthese auf Ruschkowski 1998: 295ff sowie für detailliertere Informationen auf Bristow / Chowning 1986 verwiesen. Zum Thema Sampling empfiehlt sich z.B. Ruschkowski 1998: 336ff.

werden. Erst der PLAY-Effekt bietet die Möglichkeit, kurze, simple Melodien wiederzugeben, bei denen über Dauer und Frequenz der Töne hinaus Pausen, verschiedene Einschwing- bzw. Attacktypen und ein Tempo definiert werden können. Die ersten acht (Melodie-)Töne der fünften Symphonie Beethovens können beispielsweise mit folgendem PLAY-Befehl programmiert werden: PLAY="mBt180o2p2p8l8gggl2e-p8l8ff-f12d"¹¹⁴

Den beschränkten Möglichkeiten zum Trotz werden auf dem PC-Speaker schon ab Anfang der 1980er Jahre durch einen Tweak¹¹⁵ auch gesampelte Klänge wiedergegeben, wobei das Verfahren dem des SID oder SN 76489 AN sehr ähnelt. Feldman (1999) erläutert ausführlich, in welcher Form dem Speaker eine lange Folge von Nullen und Einsen gesendet werden kann, um die Lautsprecher-Membran in die gewünschten Zustände zu bringen. Er nennt auch einige Probleme: Eine Abtastrate von beispielsweise 11 kHz führt aufgrund der im hörbaren Bereich liegenden Sampling-Frequenz zu einer konstanten Schwingung von eben 11 kHz, welche als Pfeifton hörbar wird. Eine Abtastrate von 22 kHz, wie von Feldman empfohlen, führt dazu, dass aufgrund der Taktfrequenz der internen Bauteile (Interrupts) nur 54 Zyklen bleiben, um einen bestimmten Sample-Wert wiederzugeben. Somit wird die eigentlich in 256 Schritte unterteilte 8 Bit-Wellenform auf 54 diskrete Werte begrenzt (statt einer 8 Bit- wäre also nur noch weniger als eine 6 Bit-Auflösung vorhanden), was eine größere Umrechnung voraussetzt, bis die Wellenform-Daten im gewünschten Format verfügbar sind.

Viele Spiele, die die Wiedergabe von digitalisierten Klängen über PC-Speaker unterstützen, benutzen aufgrund des notwendigen, akribischen Timings derartige Samples nicht als konstante Hintergrundmusik sondern nur als Soundeffekte oder im Intro: z.B. SIM CITY (MAXIS, 1989) oder die Golfsimulation LINKS bzw. deren Vorläufer WORLD CLASS LEADER BOARD (ACCES, 1990, 1988). In Spielen finden sich nur wenige Beispiele der erfolgreichen Implementierung durchgängiger Hintergrundmusiken, z.B. die von AMIGA auf PC portierten Flipper-Simulationen der schwedischen Programmierer DIGITAL ILLUSIONS (PINBALL DREAMS, 1993 und PINBALL FANTASIES, 1994; ♪013). Diese Technik der Sample-Wiedergabe ist auf frühen PCs aufgrund der relativ hohen Anforderungen an die Ressourcen nicht möglich. Auch funktioniert sie nicht mit Piezo-Beepern, wie sie anstelle von Membran-Lautsprechern aus Platzgründen in frühen Laptops und aus Kostengründen auch in vielen Desktop-PCs verbaut wurden. Trixter (o.J.) liefert auf der „IBM PC-Ramblings“- Seite¹¹⁶ ein Dutzend Hörbeispiele für das Verfahren.

Der PC-Speaker wird aufgrund seiner überaus großen Verbreitung als Standard-Device von einem Großteil der erscheinenden Spiele unterstützt¹¹⁷. Auch im „normalen“ Modus, also ohne Sample-Playback wird teilweise auf sehr intelligente Weise für den einstimmigen PC-Speaker komponiert, z.B. die Adaption der Titelmusik zu THE SECRET OF MONKEY ISLAND (LUCASFILM GAMES, 1990; ♪014).

¹¹⁴ Zit. nach Collins (2003b). Die Anweisung kann wie folgt übersetzt werden: „*Spiele im Hintergrund mit einem Tempo von 180 Bpm in Oktave 2: Erst eine halbe, dann eine achte Pause; dann drei achte Noten G und eine halbe Note Es; nach einer weiteren achte Pause spiele drei achte Noten F und eine halbe Note D*“.

¹¹⁵ Als „Tweak“ bezeichnet man vor allem im Computerbereich eine technische Bastelei, die bestimmte Fähigkeiten ermöglicht (von engl.: to tweak = justieren, optimieren).

¹¹⁶ <http://www.oldschool.org/sound/pc> (28.10.2004).

¹¹⁷ Die Computerspiele-Datenbank 'Mobygames' enthält 1210 Spiele mit Unterstützung des PC-Speaker und 161 Spiele, die den PC Speaker „tweaked“ verwenden, also zur Sample-Wiedergabe (vgl. die Übersichtsseiten www.mobygames.com/attribute/sheet/attributeId.25/ und www.mobygames.com/attribute/sheet/attributeId.34/ (15.03.2005)).

2.6.2 IBM PC Junior / Tandy 1000

1983 kommt der IBM PC_{JR} auf den Markt, vor allem um in Anbetracht der Heimcomputer-Konkurrenz das bisherige Image des IBM PCs als Business-Rechner zu verbessern. Aufgrund einiger Fehler im Design wie eine schlecht funktionierende Infrarot-Tastatur oder die Inkompatibilität zum damals bereits verbreiteten ISA-Standard für Erweiterungskarten (die

IBM PC jr. (1983)	
CPU; Taktfrequenz	8088; 4,77 MHz
RAM	64 kB bis 128 kB
ROM	64 kB
Datenträger	Modul, Diskette
Kapazität	bis ~ 200 kB

Aufrüstung war nur durch proprietäre Adapter möglich), bleibt dem IBM PC_{JR} größerer Erfolg verwehrt. Der Bürogerätehersteller RADIO SHACK entwirft einen zum PC_{JR} kompatiblen Computer namens TANDY 1000, der mit der Peripherie des in Amerika weit verbreiteten Heimcomputer TRS-80 kompatibel ist, wodurch nur ein Austausch der Zentraleinheit nötig ist, um das neue Modell nutzen zu können. Der TANDY 1000 wird erfolgreicher als der PC_{JR} und besitzt sowohl den 16 Farben unterstützenden Grafikmodus als auch den SN 76496 Soundchip des IBM-Modells (vgl. Forster: 2005: 98). Dieser ist mit dem bereits in Kapitel 2.2.4 behandelten SN 76489 kompatibel. Die Plattform wird durch die für damalige Verhältnisse gute technische Ausstattung für Spieledesigner interessant und es erscheinen annähernd 300 Spiele mit Unterstützung der Klangerzeugung des PC_{JR} / TANDY 1000. IBM schließt zum Start des PC_{JR} einen Exklusivvertrag mit der amerikanischen Firma SIERRA ON-LINE, ein Spiel zu programmieren, welches die Möglichkeiten von Sound und Grafik des neuen PC-Modells demonstriert. Als Ergebnis präsentiert SIERRA mit KING'S QUEST (1984) das erste (pseudo-) 3D-Adventure der Firma. In den folgenden Jahren erscheinen eine Reihe Spiele, die auf derselben, AGI (Adventure Game Interpreter) genannten Spiele-Engine basieren. Neben der KING'S QUEST-Serie sind dies die ersten Teile der Erfolgsserien SPACE QUEST (ab 1986), POLICE QUEST (ab 1987) und LEISURE SUIT LARRY (ab 1987). Die Hintergrundmusik dieser Spiele nutzt die vier verfügbaren Kanäle des SN-Soundchips, auf normalen IBM-PCs gibt der PC-Speaker nur die Melodiestimme wieder (vgl. Collins 2003b: 10).

2.6.3 Die Verwendung von FM-Chips im PC

Mitte der 1980er Jahre beginnen immer mehr Spielehersteller den PC/AT zu unterstützen, der mit seiner steigenden Verbreitung und seit der Einführung des 80286-Mikroprozessors und des EGA-Grafikstandards auch für Konvertierungen von Spielen anderer Plattformen interessant geworden ist. Die Spieler verlangen daher eine bessere klangliche Unterstützung ihrer Plattform, was zu der Entwicklung von meist auf FM-Synthese-Chips der Firma YAMAHA basierenden Soundkarten führt. Die meisten dieser besitzen einen Joystick-Port und sind somit auf die Bedürfnisse von Spielern zugeschnitten.

1987 erscheint in Gestalt der IBM MUSIC FEATURE CARD die erste auf einem FM-Soundchip basierende Soundkarte. Als Soundchip auf dieser dient der YM 2164, welcher in Form des YAMAHA FM-Synthesizermodul FB01 und des KORG Synthesizers 707 auch in professionellen Musikstudios zu finden ist (vgl. Göhler 2004, Otten 2004). Die achttimmige Synthesizerkarte ist außerdem mit einem vollwertigen MPU-401 kompatiblen MIDI-Interface ausgestattet und richtet sich damit sowie angesichts des hohen Preises vor allem an

professionelle Anwender. Neben den Firmen SIERRA und MICROPROSE unterstützt nur ein weiterer Spielehersteller diese Karte (vgl. Stolz 1992: 12 oder Göhler 2004). Der Datenbank Mobygames.com¹¹⁸ zufolge gibt es nur 18 erschienene Spiele mit MUSIC FEATURE CARD - Unterstützung.

Die erste kommerziell erfolgreiche PC-Soundkarte wird 1987 von der kanadischen Firma AD LIB MULTIMEDIA entwickelt und basiert auf dem FM-Soundchip YM 3812, welcher eine Weiterentwicklung des in vielen Spielautomaten verwendeten YM 3526 OPL. Der YM 3812 wird auch als OPL2 bezeichnet und definiert den sog. AD LIB-Standard. Er bietet die Möglichkeit, für jeden der 18 Operatoren eine von vier Wellenformen zu wählen, wobei diese aus Teilen von Sinuswellen generiert werden, wie Abb. 8 illustriert. Ein OPL2-Enable-Bit bewirkt, dass für den OPL programmierte Software nicht irrtümlich auf die neuen OPL2-Register zugreift. Alle OPL-Chips bieten die Möglichkeit, zwischen zwei Modi auszuwählen: Entweder es klingen alle FM-Stimmen à zwei Operatoren oder eine geringere Anzahl FM-Stimmen à zwei Operatoren mit fünf vordefinierten Perkussionsinstrumenten gemeinsam¹¹⁹.

Viele Spielehersteller unterstützen die AD LIB-Karte ab 1988, wobei abermals SIERRA eine der ersten dieser Firmen darstellt (vgl. Wing o.J.). Die AD LIB-Soundkarte wird im Paket mit einem Jukebox-Programm für MIDI-Files, einem Sequenzer und einem FM-Synthese-Programm verkauft, besitzt jedoch keinen Joystick-Port.

D1	D0	Waveform
0	0	
0	1	
1	0	
1	1	

Abbildung 8
Die verschiedenen Wellenformen des YM 3812

Die Singapurische Firma CREATIVE TECHNOLOGY veröffentlicht 1988 eine eigene, GAME BLASTER getaufte, zwölfstimmige Soundkarte, die von der Klangqualität mit klassischen PSGs vergleichbar ist¹²⁰ und daher mit den auf FM-Synthese basierenden AD LIB-Karte nicht konkurrieren kann. Diese in Deutschland auch als CMS-CARD angebotene Erweiterungskarte wird daher bald wieder vom Markt genommen, auch wenn der GAME BLASTER - Standard zwischen 1989 und 1991 in gut 70 Titeln Unterstützung von Spieleherstellern erfährt. 1989/90 bringt CREATIVE mit der SOUND BLASTER-Karte ein neues Modell heraus, welche auf demselben FM-Chip basiert wie die AD LIB-Karte, mit dieser also Software-kompatibel ist. Darüber hinaus bietet sie die Wiedergabe eines 8 Bit-Sampling-Kanals und ist in der 1.0 Version zudem GAME BLASTER-kompatibel¹²¹. SOUND BLASTER bzw. AD LIB (bzw. OPL 2) werden in den Folgejahren zum de facto-Standard für Musik in PC-

¹¹⁸ <http://www.mobygames.com/attribute/sheet/attributeId,22/> (15.03.2005).

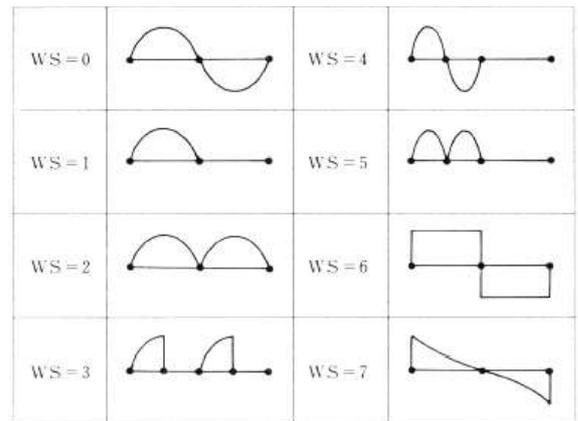
¹¹⁹ Die fünf Perkussionsinstrumente heißen Bass Drum, Snare Drum, Hi-Hat, Tom, und (Top) Cymbal, wovon allein die Bass Drum zwei Operatoren benutzt, der Rest einen. Hi-Hat und Cymbal sind als einzige auf eine Frequenz festgelegt. Der OPL3 (YM 262) besitzt darüber hinaus zwei weitere Möglichkeiten, wenn das OPL3-Enable Bit gesetzt ist: sechs Stimmen à vier Operatoren und sechs Stimmen à zwei Operatoren oder sechs Stimmen à vier Operatoren, drei Stimmen à zwei Operatoren und die fünf Percussion-Sounds. Die auf vier Operatoren basierenden Algorithmen des OPL 3 bieten aufgrund redundanter Schaltungen weniger komplexe Möglichkeiten der Klangsynthese als z.B. der YM 2151 (vgl. Arnost 2000: o.S.). Für die Entwicklung der Musik von Videospiele weniger von Bedeutung sind die später erscheinenden Soundchips des OPL4-Standards, die nicht mehr allein auf FM-Synthese beschränkt bleiben, sondern wie spätere YAMAHA-Soundchips ein GM-kompatibles Set von Wavetable-Sounds enthalten.

¹²⁰ Der Kommentar Göhlers auf seiner Webseite zu dem GAME BLASTER: „Man nehme 12 PC Speaker, erlaube ihnen einen Lautstärkeregelung in 16 Stufen, gebe dazu noch einen Noisegenerator, der 3 verschiedene Drums nachbilden soll, und erlaube zudem die Wiedergabe in Stereo“ (Göhler 2004). Die Webseite Mobygames vergleicht den GAME BLASTER mit der Klangerzeugung des PCJR (<http://www.mobygames.com/attribute/sheet/attributeId,42/>; 18.07.2005).

¹²¹ Die SOUND BLASTER 1.5 und Teile der 2.0 Version konnten mit den sog. CMS-Chips nachgerüstet werden, um sie GAME BLASTER-kompatibel zu machen.

Computerspielen, der sich bis Ende der 1990er Jahre halten kann. Die SOUND BLASTER-Karte verdrängt dabei das AD LIB Modell relativ schnell, da sie die trotz der Möglichkeit des Sample-Playbacks nur wenig teurer ist, dieses Feature jedoch von vielen Anwendern gewünscht wird. Von der SOUND BLASTER-Karte erscheinen verschiedene Revisionen. Bald nach Einführung der 2.0 Version wird auf die Sockel für die Nachrüstung von CMS-Chips verzichtet. Die SOUND BLASTER PRO erscheint 1991, mit der Möglichkeit Stereo-Sound wiederzugeben, was durch zwei OPL2-Chips bzw. später einem OPL3 (YMF 262) ermöglicht wird.

Abbildung 9
Die verschiedenen Wellenformen des YMF 262 / OPL3



Außerdem kommt eine 16 Bit-Version mit demselben FM-Chip auf den Markt. Sie besitzt eine Schnittstelle für verschiedene CD-ROM-Laufwerke, die es ermöglicht, 16 Bit-aufgelöste Audio-Signale (Redbook-Audio) direkt von der CD wiederzugeben. Die SOUND BLASTER 16 löst SOUND BLASTER und SOUND BLASTER PRO ab 1992 als Standard für den PC-Bereich ab. Folgende Tabelle vergleicht die Daten der drei ersten OPL-Chips:

	YM 3526 OPL	YM 3812 OPL2	YMF 262 OPL3
Verwendet ab	1986	1987	1991
Verwendet in	Arcade	Arcade, Ad Lib-, Sound Blaster- Soundkarten & kompatible	Sound Blaster Pro
Art und Anzahl der Kanäle	9 FM-Kanäle (à 2 OP) oder 6 FM (2 OP) + 5 Percussion (BD, SD, HH, TC, TOM)	9 FM-Kanäle (à 2 OP) oder 6 FM (2 OP) + 5 Percussion (BD, SD, HH, TC, TOM)	18 FM (à 2 OP) oder 15 FM (à 2 OP) + 5 Percussion oder 6 FM (à 4 OP) + 6 FM (à 2 OP) oder 6 FM (à 4 OP) + 3 FM (à 2 OP) + 5 Percussion
FM-Algorithmen	2	2	4
Effekte	Feedback, Vibrato, Tremolo		
Hüllkurven	ADSR / ADR (kann über Status-Bit geändert werden)		
Besonderheiten	-	Gegenüber YM 3526 neben Sinus drei neue Wellenformen, pro Generator wählbar.	Stereo; Gegenüber vorigen OPL-Modellen acht versch. Wellenformen, pro Generator wählbar.

Tabelle 7 Die verschiedenen OPL-Chips (mit Ausnahme des YM 2413 und späterer OPL-Versionen)

2.6.4 Die Wavetable-Synthesizerkarten von Roland

Der japanische Synthesizer-Produzent ROLAND veröffentlicht 1988/89 mit der LAPC-1 eine Soundkarten-Version der in populären ROLAND Synthesizern (z.B. dem D50) verwendeten LA (Linear Arithmetic)-Synthese. Diese basiert auf 8 Bit PCM-Samples, von denen jeweils bis zu vier in verschiedenen „Structures“ zu einem Patch zusammengefasst werden. Die LAPC-1 bietet 32 verfügbare Stimmen, 256 Preset-Instrumente, die Möglichkeit, eigene Patches zu erstellen sowie einen integrierten Hall-Effekt, womit diese Karte den auf FM-Synthese basierenden Konkurrenzprodukten klanglich weit voraus ist. Eine 12 Bit D/A-Wandlung und 32 kHz Sampling-Frequenz sind zudem technische Werte, die den vergleichsweise hohen Preis rechtfertigen.

Das Soundmodul ROLAND MT-32 basiert wie die LAPC auf der LA-Synthese, besitzt jedoch nur 128 Preset-Instrumente und keine Möglichkeit, eigene Patches zu erstellen. Abermals ist es ein Abkommen mit dem amerikanischen Spielehersteller SIERRA, der diesen Karten zu einem großen Bekanntheitsgrad verhilft. So werden LAPC-1 und MT-32 ab dem Spiel KING'S QUEST IV (SIERRA, 1988) zum Standard für die Musik-Wiedergabe des Spieleherstellers, was für die ersten orchestralen Soundtracks in PC-Spielen sorgt¹²².

Die Spielmusik-Versionen für OPL 2, SN 76489 oder PC-Speaker stellen bei entsprechenden Spielen somit zumeist Umsetzungen der 'originalen' LAPC-Versionen dar. SIERRA wird darüber hinaus Vertriebspartner für das MT-32 in Amerika. Die der Konkurrenz überlegene Klangqualität führt dazu, dass beide Modelle in der Folge auch von vielen anderen Herstellern unterstützt werden. Gerade in der Anfangszeit entsteht dabei ein großer Unterschied zwischen der klanglichen Untermalung und der rudimentären grafischen Repräsentation, die bei den ersten Titeln mit ROLAND-Unterstützung wie SILPHEED (SIERRA, 1988) allenfalls im CGA bzw. EGA Standard (max. 16 Farben bei max. 640 x 400 Pixeln Auflösung) möglich ist. Das Verfahren der LA-Synthese verwendet verschiedene Samples für die einzelnen Hüllkurven-Phasen eines Sounds. Für die Attack- und Decay-Phase werden beispielsweise Anblas-Geräusche eingespielt, für die Release-Phase ein Nachhall o.ä.. Zwischen den bis zu vier Einzel-Samples, die einen Patch ergeben, wird zudem teilweise interpoliert, was jedoch die Anzahl der verfügbaren, gleichzeitig klingenden Stimmen von 32 auf bis zu acht reduziert. Berühmt geworden ist neben dem für damalige Verhältnisse realistischen Eigenklang die häufige Verwendung des Hall-Effekts in Computerspielmusik für ROLAND-Soundkarten.

2.6.5 Die Entwicklung der PC-Soundkarten über 1991 hinaus

Bis 1991 bestimmen die vorgestellten Modelle von AD LIB, SOUND BLASTER und ROLAND den Markt der Soundkarten für PCs, wobei durch eine Fülle von kompatiblen Karten der SOUND BLASTER-Standard mit Abstand am erfolgreichsten bleibt. Die ROLAND-Soundkarten erfahren aufgrund ihres relativ hohen Preises nur wenig Absatz, werden jedoch von vielen Spieleherstellern unterstützt. 1991 erscheinen gleich mehrere Nachfolge- bzw. Konkurrenzprodukte, wie die AD LIB GOLD, die GRAVIS ULTRASOUND oder die MEDIA VISION PRO AUDIO SPECTRUM. Ebenso erscheint 1992 die bereits erwähnte CREATIVE SOUND BLASTER 16, die neben der CD-Audio Wiedergabe-Möglichkeit eine Schnittstelle für die Aufrüstung um ein Wavetable-Board, der sog. WAVE BLASTER bietet. Diese stellt eine Wavetable-Erweiterung dar, ähnliche Boards werden auch bald von anderen Herstellern angeboten, z.B. die DB50XG von YAMAHA.

Sowohl die AD LIB GOLD als auch die GRAVIS ULTRASOUND haben kommerziell unter der fehlenden SOUND BLASTER-Kompatibilität zu leiden. Die GRAVIS ULTRASOUND wird aufgrund ihrer flexiblen Sample-Verwaltung und des hochwertigen integrierten Mixers in der Tracker-Szene zu einem 'Geheimtipp'. Mit der SCC-1 bringt ROLAND 1992 die erste GM (General MIDI)-kompatible Soundkarte heraus, die auf dem Soundmodul SC-55 (SOUND CANVAS) basiert. In den Folgejahren erscheinen in erster Linie mit Wavetable-Synthese ausgestattete Soundkarten, wie die SOUND BLASTER AWE 32 / 64 (1993) oder die auf den YAMAHA YMF278 / OPL4-Chip

¹²² Als Beispiel sei die Titelmelodie von STAR WARS: X-WING (LUCAS ARTS, 1993) erwähnt (♩015).

(mit integriertem, General MIDI-kompatiblen Wavetable) basierende MAGIC MIDIA AUDIOTRIX PRO (1994). Der GM-Standard führt ab 1993 zu einer Vielzahl von Soundkarten, die mit GM- oder LAPC-1-Kompatibilität werben, klanglich aber weit hinter den Erweiterungen zurückbleiben, wie beispielsweise die SOUND BLASTER AWE-Karten. Sanger beschreibt das Aufkommen der ersten GM-kompatiblen Soundchips, die zwar kompatibel sein sollen, aber die für den SOUND CANVAS komponierten MIDI-Daten nicht korrekt interpretierten:

„Every one of the GM cards I got for the next year was awful. I mean, it sounded good if you wrote music especially for it but if you played back stuff that was written on a sound canvas, it was dreadful. Entire passages would be lost. Instruments would be an octave off. Some sounds would blare out suddenly at five times the volume of any other instrument playing. The problem turned out to be that there had been no specification for General MIDI, just a set of suggested guidelines, and these guidelines had no indication of how loud the instruments should be, compared to each other, nor how slow or fast the instrument's attack should be“ (Sanger 2003: 189).

Neben den Komponisten waren auch die Hersteller der Karten wenig erfreut über diese Entwicklung. Sanger beschreibt weiter (2003: 190f), wie seine Firma zahlreiche Karten testet und Algorithmen für die Emulation von General MIDI auf Soundkarten wie der MT-32 / LAPC-1 sowie den OPL-Chips entwickelt. Ab Ende der 1990er Jahre wird durch die bereits erwähnte Integration von Standards für die Wiedergabe von Digital Audio in Form von *onboard*-Soundkarten sowie dem Bereitstellen von Software-Synthesizern für die GM-Wiedergabe die Anschaffung spezieller Soundkarten für die meisten Anwendungen überflüssig. Gleichzeitig unterstützen moderne Spiele wie MAX PAYNE 2: THE FALL OF MAX PAYNE (ROCKSTAR GAMES, 2003) für aufwendige Hall- und Effekt-Berechnungen in der dreidimensionalen Spielumgebung Soundkarten, die einen Dolby 5.1-Encodingchip oder auch hardwarebasierte Algorithmen für die Berechnung von Hall-Effekten im Rahmen des sog. EAX-Standard¹²³ bieten.

Das Wegfallen der Ansteuerung einer genuine Klangerzeugung (mittels Soundchips) zugunsten der Wiedergabe von Redbook oder *streamed* Audio ist für Computerspiele auf dem PC ebenso wie auf den Spielkonsolen seit der Verbreitung der CD-ROM-Technologie zur Regel geworden, Soundchip-Musik scheint obsolet. Dennoch lässt sich in aktuellen Spielen vereinzelt eine Renaissance alter Technologien wahrnehmen. So verwendet das 2005 erschienene Spiel DARWINIA der englischen Firma INTROSPECTIVE SOFTWARE¹²⁴ einen emulierten Pokey-Soundchip in Verbindung mit einer zeitgemäßen Effektsektion, was zu einer interessanten klanglichen Untermalung des Spielgeschehens führt.

¹²³ Die ENVIRONMENTAL AUDIO EXTENSIONS VON CREATIVE LABS stellen den Programmierern heutiger Spiele eine sog. „statistical reverbation engine“ zur Verfügung, in der modellhafte Räume über eine geringe Anzahl von Parametern beschrieben und berechnet werden können. Die Berechnungen der Halleffekte werden hierbei zwischen den einzelnen Klangobjekten in Relation zu dem sog. „listener object“ vorgenommen. (Boer 2003: 505f)

¹²⁴ Eine Demo-Version des bisher nicht in Deutschland veröffentlichten, ästhetisch lohnenswerten Spiels kann unter <http://www.darwinia.co.uk/> (01.08.2005) heruntergeladen werden.

2.7 Zwischenfazit: Eigenheiten von Soundchips

Wie aus der quantitativen Schilderung von technischen Daten und Fähigkeiten der verschiedenen Soundchips ersichtlich wird, handelt es sich bei ihnen um Klangerzeuger, die vor allem durch *Limitierungen* erklärt werden können.

Diese Eigenart grenzt sie deutlich von den historischen elektronischen Musikinstrumenten ab, welche entlang ihrer Entwicklungsgeschichte seit den 1920er Jahren vor allem im Hinblick auf die Überwindung bisheriger Limitierungen und in Betrachtung der durch sie möglichen Potentiale in Bezug auf neue Klangfarben, Mikrotonalität oder ähnliche Aspekte diskutiert wurden¹²⁵. Das seit den 1950er Jahren stetig wachsende Feld der Computermusik thematisiert dagegen vor allem die Formalisierbarkeit von musikalischen wie klanglichen Strukturen¹²⁶.

Die Determiniertheit von Soundchips durch Limitierungen wie mangelhafte Stimmbarkeit und Beschränktheit der möglichen Ausgestaltung von musikalischen Parametern wie Klangfarbe, Lautstärke und Tonhöhe sowie ihre grundsätzliche Abhängigkeit von der jeweiligen Hardwareumgebung machen Soundchips daher zu einem Kuriosum innerhalb der elektronischen Klangerzeugung.

Der wissenschaftliche Diskurs um elektronische und von Computern erzeugte bzw. berechnete Musik klammert Heimcomputer, Spielkonsolen und die dazugehörigen Soundchips bisher jedoch noch aus. Daher lässt sich kaum auf wissenschaftliche Aufsätze zurückgreifen und die qualitative Bewertung der zuvor erläuterten Fakten muss ebenso wie die im ersten Abschnitt erfolgte 'Archäologie' größtenteils ohne Rückgriffe auf externe Literatur geschehen.

Es sei an diesem Punkt abermals darauf hingewiesen, dass Computerspielsysteme industriell gefertigte Massenprodukte darstellen, die mit möglichst preiswerten Mitteln ein Höchstmaß an multimedialer Unterhaltung garantieren sollen. Vor allem Spielkonsolen basieren auf dem Geschäftsmodell, eine kostengünstige Hardware durch attraktive Software zu verkaufen.

Die vorgestellten technischen Details der unter diesen Umständen entwickelten Soundchips lassen einige qualitative Rückschlüsse und Thesen zu, die für den Aspekt der Programmierung / Komposition und damit im Endeffekt auch für die medienmusikalische Aufarbeitung von Wichtigkeit sind:

- Die technischen Voraussetzungen determinieren die Computerspielmusik im starken Maße. Neben den Soundchips als Wiedergabemedium bzw. Instrument muss das jeweilige Speichermedium als Informationsträger für die Musikdaten sowie die Struktur der Musikdaten selbst Berücksichtigung finden. Diese unterliegen den v.a. ökonomisch bedingten, technischen Limitierungen und wirken auf die formale und klangästhetische Ausgestaltung ein.
- Es lassen sich verschiedene Epochen in der Entwicklung von Computerspielmusik herausarbeiten, die stark mit den technischen Voraussetzungen verbunden sind. Kritisch hierfür ist die technische Weiterentwicklung der Klangerzeugung (FM, Sampling, mehr Stimmen etc.).

¹²⁵ Eine Übersicht der in den Anfängen der elektronischen Klangerzeugung entwickelten Instrumente und der mit diesen Entdeckungen verbundenen Gedanken bieten Berrisch 1996 und Ruschkowski 1998.

¹²⁶ Ruschkowski 1998 bietet einen Überblick über die Hauptbereiche der „ersten“ Computermusik, der Partitur- (266ff) und Klangsynthese (285ff).

- Auch die massenhafte Verbreitung bestimmter Plattformen und durch geschickte Programmierung als „Meilensteine“ berühmt gewordene Soundtracks von Computerspielen sind wichtig für die Kanon-Bildung in der Computerspielmusik. Der kommerzielle Erfolg einer bestimmten Hard- oder Software darf also nicht außer acht gelassen werden.
- Neben der Berücksichtigung der verwendeten Technik muss der Charakter von Computerspielmusik als auditives Beiwerk eines multimedialen, digitalen Kulturguts reflektiert werden. Als solches unterliegt die Programmierung der Computerspielmusik eigenen, vor allem funktionalen Kriterien, welche die Medienrealität von Soundchip-Musik für Computerspiele bilden.

Sich diesen Punkten in der Betrachtung der Strategien der Gestaltung von Computerspielmusik zu nähern, ist Anliegen des nächsten Kapitels, welches nach einer formalen Katalogisierung der Musik in Computerspielen und einem Exkurs zu ihrem mitunter adaptiven Charakter auf die Eigenheiten ihrer Komposition / Programmierung im Spiegel der technischen Limitierungen eingeht.

3 Strategien der Gestaltung von Computerspielmusik für Soundchips

Der erste Teil der vorliegenden Arbeit hat sich in einer deskriptiven Form den technischen Voraussetzungen der Computerspielmusik in den 1980er und 1990er Jahren gewidmet. Diese rein phänomenologische Betrachtung der technischen Voraussetzungen von Soundchip-Musik soll im Folgenden erweitert werden, um ein umfassendes medienmusikalisches Bild der Computerspielmusik für Soundchips zeichnen zu können. Daher werden der funktionale Charakter von Spielmusik, sowie weitere Einflüsse aus der das auditive Material eines Computerspiels berechnenden Technik vorgestellt.

Die Näherung an die Gestaltung der Computerspielmusik geschieht in einem ersten Teil „von außen“. Dieser erläutert, welche Typen von Musik in interaktiven Spielen vorkommen, wo die Anfänge dieser liegen und welchen Formen der Weiterentwicklung sie unterliegen.

In einem zweiten Teil wird die Perspektive „von innen“, also stark von dem Schaffen der in der jeweiligen Zeit tätigen Komponisten / Programmierern ausgehend, erläutert. Diese Innensicht fußt in erster Linie auf vom Autor telefonisch geführten, themenzentrierten Interviews und deduktiven Schlussfolgerungen aus diesen. Die zu gewinnenden Erkenntnisse ermöglichen eine Thematisierung der Frage, welche kompositorischen bzw. programmiertechnischen Verfahren angewendet worden sind, um den technischen Unzulänglichkeiten der Soundchips zum Trotz musikalische Ideen umzusetzen. Diese Unzulänglichkeiten umfassen sowohl die limitierten Möglichkeiten der Soundchips selbst (Klangerzeugungsverfahren), als auch durch die damalige Computertechnik auferlegte Restriktionen (Speicherplatz, Rechenleistung,..).

3.1 Musik in interaktiven Spielen

Bei der Betrachtung von Computerspielen stellt sich das grundsätzliche Problem der Trennung zwischen dem eigentlichen Output des Spiels und dem Spiel als Computerprogramm. Betrachten lässt sich nur das Gezeigte, also das audiovisuelle Resultat eines im Hintergrund laufenden Programms. Das Programm besitzt somit den Charakter einer „Blackbox“ (vgl. Stockburger 2003: 3).

Der dänische Wissenschaftler Jesper Juul unterscheidet in seinem triadischen Modell zwischen *Material* (Sound, Grafik, Text), *Programm* (Regeln der Kombination des Materials) und dem daraus resultierenden *Output* (vgl. Juul 2001: 35).

Die als Programm implementierte Musik von Computerspielen weist jedoch in zweierlei Hinsicht über den bloßen Materialcharakter (im Sinne Juuls) hinaus. Wie Stockburger anhand des aktuellen Spiels METAL GEAR SOLID 2 erläutert, ist das Programm in Computerspielen immer schon die *Ordnungsinstanz* der eingespielten Klangobjekte, weswegen der Fokus auf die Relation zwischen Programm und Material bei der Analyse von Musik und Klang in Computerspielen zu jeder Zeit berücksichtigt werden muss (Stockburger 2003: 3). Im Fokus der vorliegenden Arbeit wird die Musik von Computerspielen darüberhinaus als *Ergebnis einer Programmierung* begriffen, da Musik aus Soundchips nur durch die Programmierung dieser möglich ist. Da Juul Programm und Material als grundsätzlich trennbar definiert (2001: 36), tritt hier eine Leerstelle seiner eher auf narrative Strukturen in Spielen fokussierten Theorie zutage.

Im Hinblick auf die Ordnungsinstanz des Programms für Soundchip-Musik fällt auf, dass anders als bei den modernen Beispielen Stockburgers die Musikroutinen der Soundchip-Zeit weniger die Organisation von Klangmaterial oder Soundeffekt-Dateien samt der ihnen je nach Spielgeschehen zugeordneten Parameter bezüglich ihrer räumlichen Verortung und entsprechende Mischungsverhältnisse zur Aufgabe haben. Vielmehr erfolgt die Organisation auf einer unmittelbareren Ebene: Musik-Daten müssen gelesen, in der Musikroutine zusammengesetzt und danach an den Soundchip geschickt werden. Bei den Beispielen der Soundchip-Musik ist das Programm somit weniger ordnende Instanz auf Mikro-Ebene als in modernen Audio-Engines.

Nicht die ständige Anpassung des auditiv wahrgenommenen Spielgeschehens¹²⁷ steht im Vordergrund, vielmehr gliedern auf der organisatorischen Ebene die Strukturen des Spielinhalts die Computerspielmusik für Soundchips. Die als Ergebnis der Prozessierung von Daten eingespielte Musik ist damit ebenso eingebunden in einen funktionalen Rahmen, auch wenn dieser nicht so vielschichtig kategorisierbar ist, wie im Katalog von Stockburger.

Soundchip-Musik in frühen Spielen besetzt bereits erste Schlüsseleigenschaften als *Informationsgeber* in interaktiven Spielen: Sie schafft Aufmerksamkeit, indiziert Gefahren oder Chancen. Nicht zuletzt ist sie auch ein genereller *Stimmungsträger*, mit dessen Hilfe das Geschehen auf dem Bildschirm in der Wahrnehmung der Spieler in bestimmten Affekten verstärkt wird. Der erste Teil dieses Abschnittes widmet sich einer Systematisierung der verschiedenen Typen von Computerspielmusik, unabhängig von ihrer Informationshaftigkeit. Letztere soll später unter der Überschrift *adaptiver* Musik erläutert werden.

¹²⁷ In aktuellen Spielen wie METAL GEAR SOLID 2 : SONS OF LIBERTY (KONAMI, 2002) oder MAX PAYNE 2 findet aufgrund der dreidimensionalen Beschaffenheit der Spielwelt eine ständige Berechnung der Klangobjekte in Relation zum Listener-Objekt statt (vgl. Boer 2003: 55).

3.1.1 Eine Typisierung der verschiedenen Spielmusiken

Computerspiele als Produkt bestehen nicht allein aus dem Spiel als solchem, welches an dieser Stelle als während und durch die Eingaben des Spielers stattfindende multimediale Repräsentation einer Spielhandlung verstanden wird, sondern ebenso aus anderen Elementen, die letztere umrahmen.

Spielhandlungen werden generell eingerahmt von mindestens zwei Elementen: Ein Intro- oder Start-Bildschirm und ein Game Over-Bildschirm¹²⁸. Es gibt – vor allem in Action-Spielen¹²⁹ bis Anfang der 1990er Jahre – High Score-Listen¹³⁰ oder die sog. „Endings“, in denen die Rahmenhandlung des Computerspiels nach erfolgreicher Beendigung zu Ende erzählt wird. Des Weiteren existieren Menüs, in denen verschiedene, das Spiel betreffende Optionen geändert werden können.

All diesen Elementen ist gemein, dass sie zunehmend musikalisch untermalt wurden, aber nicht zum Spielgeschehen zählen. Diese Elemente sind ebenso Teile des als Computerspiel verkauften Produkts, die Produktionsparadigmen der sie untermalenden Musik sind jedoch andere, als jene der mit den Spielhandlungen synchron laufenden Musik. Ich spreche bei dieser Form von Spielmusik daher von *Sekundärer* Spielmusik.

Im eigentlichen Spielgeschehen gibt es je nach Genre verschiedene Level und Welten sowie mitunter versteckte, zu entdeckende Bereiche. Es gibt Bonusrunden, die erst nach einer erbrachten Leistung des Spielers spielbar sind, häufig existieren Zeitlimits und manchmal sogar Gegenstände, die selbst musikalisch sind. Also nach erfolgter Interaktion eine Melodie von sich geben oder eine im Hintergrund laufende Musik beeinflussen. Da solche Musik zeitlich mit den Spielhandlungen des Spielers zusammen fällt, spreche ich hier, den Konventionen der online verfügbaren Archive¹³¹ entsprechend, von *Ingame* Spielmusik. Eine das Spielgeschehen einleitende Melodie, genau so wie etwa die Melodie beim Ableben der Spielfigur, markieren demzufolge den Übergang von der *Sekundären* in die *Ingame* Spielmusik und umgekehrt.

Dem historischen Fokus dieser Arbeit entsprechend stehen bei der folgenden Beschreibung die Spiele der 8 und 16 Bit-Epoche im Mittelpunkt, wobei hier sowohl Spielautomaten, als auch Spielkonsolen und Heimcomputer bzw. PCs grundsätzlich zusammen gedacht werden. Die folgenden Kategorien beziehen sich v.a. auf Spiele dieser Zeit, finden jedoch je nach Genre bis heute Verwendung. Die als Beispiele verwendeten Titel können dem Action-Genre zugeordnet werden. Sowohl bei Strategie- als auch Adventure-Spielen des

¹²⁸ Selbstverständlich existieren Spiele, die diese beiden Elemente nicht enthalten, wie z.B. PONG, das gänzlich ohne Intro-/Start- und Game Over-Bildschirm auskommt. Viele Strategie-Spiele wie SIM CITY enthalten außerdem keinen Game Over-Bildschirm

¹²⁹ Bei der Genre-Einteilung möchte ich der Differenzierung von Claus Pias folgen, der Computerspiele in drei basale Genres einteilt. Strategie-Spiele bieten dem Spieler ein *konfigurationskritisches* Modell, an dem er im Spiel verschiedene Parameter ändert. Adventure-Spiele leiten den Spieler *entscheidungskritisch* durch eine in einer Datenbank abgelegte Geschichte, die von ihm auf diese Art nacherzählt wird. Action-Spiele fragen die Eingaben des Spielers zeitkritisch ab, wobei seine Aktionen möglichst im richtigen Takt zu erfolgen haben (vgl. Pias 2002: 10f).

¹³⁰ Der erste Automat mit zwischenspeicherbarer High Score - Liste ist ASTEROIDS (ATARI, 1979). Mit zunehmenden Umfang der Spiele und dem verstärkten Implementieren explorativer Spielinhalte in nahezu allen Genres wurde der High Score in Computerspielen zunehmend überflüssiger und ist heutzutage nur noch selten implementiert. Action-Genres wie Renn- und Sportspiele führen ihrem Sujet entsprechend weiterhin Bestenlisten.

¹³¹ In Archiven von Webseiten für gerippte oder in MIDI-Dateien arrangierte Computerspielmusik, wie z.B. www.vgmusic.com, www.mirsoft.info oder www.zophar.net (alle URLs:25.06.2005) wird der Begriff „ingame Music“ häufig verwendet (engl.: in game = innerhalb des Spiels).

betreffenden Zeitraums ist die musikalische Untermalung weniger abwechslungsreich, vor allem aufgrund der längeren Zeitbögen im Spielgeschehen wurde von einer beispielhaften Thematisierung der in ihnen enthaltenen Musiken abgesehen¹³², obwohl sich auch in ihnen eine Vielzahl der Elemente der folgenden Einteilung finden, die sich primär auf klassische Action-Spiele¹³³ bezieht. Ein Vertreter dieser stellt das Spiel BUBBLE BOBBLE (TAITO, 1986) dar, welches in Kapitel 3.1.4 als Fallbeispiel für die erarbeiteten Kategorien dient. Es ist bis heute ein beliebtes Spiel und in zahlreichen Umsetzungen für die meisten Hardware-Plattformen erhältlich¹³⁴. Neben der großen Verfügbarkeit eignet es sich aus dem Grunde der schnellen Erlernbarkeit gut zur Demonstration, außerdem ist es zu zweit, in einem kooperativen Modus spielbar und stellt keine besonders hohen Anforderungen an die Spieler¹³⁵. Des Weiteren sollen zu Illustrationszwecken der Automatenklassiker DIG DUG (NAMCO, 1982) und SUPER MARIO BROS. (NINTENDO, 1985) für NES dienen.

3.1.1.1 Sekundäre Spielmusik

Sekundäre Spielmusik umfasst allein die Musiken, die das Spielgeschehen „einrahmen“, also die Musik, die nicht während der eigentlichen Spielhandlungen erklingt. Je nach technischer Plattform, Genre und Erscheinungsjahr¹³⁶ sind weniger oder mehr der folgenden Musiktypen implementiert.

Als **Loader-Musik** bezeichnet man die primär auf Heimcomputern wie dem COMMODORE C64 oder COMMODORE AMIGA während des Ladevorgangs abgespielten Stücke. Bis alle Daten von Diskette gelesen sind und das Spiel bzw. ein Teil davon sich im Arbeitsspeicher befindet, vergehen teilweise mehrere Minuten, weswegen einige Firmen beginnen, diese Wartezeit mit Musik zu unterlegen. Bei gecrackten (also ihres Kopierschutzes entledigten) und danach raubkopierten Spielen, die teilweise mit Hacks wie den sog. „Trainern“ (Modifikationen, die es erlauben, ein Spiel z.B. mit einer unendlichen Anzahl Leben oder ohne Kollisionsabfrage zu spielen) versehen wurden (vgl. Levy 1994: 373ff), programmierten die entsprechenden, sog. Cracking- und Hacking-Groups kleine Introfilme (sog. Hacktros bzw. Cracktros¹³⁷).

¹³² Viele der dem Adventure-Genre zugeordneten „Role Playing Games“ (RPGs) auf dem NES wie DRAGON QUEST oder FINAL FANTASY gelten für die Entwicklung der Soundchip-Musik in Computerspielen als Klassiker, vgl. Kapitel 2.4.

¹³³ Diese bieten außerdem einen schnelleren Zugang als z.B. RPGs oder Adventure-Spiele, die ihrer langen Spieldauern wegen nicht als Beispiele genommen werden sollen. Um eine Kanonisierung zu gewährleisten, finden sich auf der dieser Arbeit beiliegenden CD ROM-Abbilder der meisten im Folgenden benutzten Beispiele für den MAME-Emulator (<http://www.mame.net>; 13.06.2005).

¹³⁴ Von der originalen Arcade-Version sind laut der Datenbank mobygames bis dato Portierungen für PC, APPLE II, C64, NES, SEGA MASTER SYSTEM, ATARI ST, AMIGA, ZX SPECTRUM, GAME BOY, GAME BOY ADVANCE und SEGA SATURN erschienen (<http://www.mobygames.com/search/quick/p,-1/q,bubble:20bobble/showOnly,9/>; 11.04.2005).

¹³⁵ Eine ausführliche Erklärung des Spiels und viele Informationen finden sich in der Bubble Bobble FAQ Version 2.0 (Hall o.J.)

¹³⁶ Es lässt sich grundsätzlich konstatieren, dass mit der Zeit immer aufwendigere Spielmusiken produziert werden. Dies erscheint plausibel, da immer leistungsfähigere Tools, Treiber und Tweaks mit der Zeit zu einer immer effizienteren Nutzung der Ressourcen und damit zu einer effizienteren Nutzung der Bandbreite für kreative Anwendungen führt.

¹³⁷ Cracktros und Hacktros gelten als Ursprung der Demo-Szene (vgl. Tasajärvi 2004a).

In diesen bildet ein mit der Graffiti-Kultur des Hip Hop vergleichbarer, häufig animierter Schriftzug die Basis, Grüße, Beschimpfungen, sowie die Lobhudelei des eigenen Schaffens werden in einem meist scrollenden Text angezeigt (vgl. Tasajärvi 2004a: 12f). Im Falle eines vorhandenen Trainers können Einstellungen an diesem vorgenommen werden. I.d.R. ersetzen solche Cracktros den eigentlichen beim Laden angezeigten Bildschirm und eine entsprechende Cracktro-Musik die u.U. vom Hersteller implementierte Loader-Musik. In Spielen für Spielkonsolen und Arcade-Automaten ist wegen der kaum vorhandenen Ladezeiten und des schwierigen Raubkopierbarkeit selten Loader- oder Cracktro-Musik implementiert worden. Erst bei den auf CD-ROM-Datenträgern basierenden Spielkonsolen nach 1994 wird während der relativ langen Ladezeiten wieder häufig Musik gespielt.

Intro-Musik: Der Titelschirm der entsprechenden Computerspiele besteht i.d.R. aus einer Grafik mit dem Schriftzug des Spiels und einem kleinen Menü, das den Benutzer zwischen Punkten wie „Spiel starten“ und „Optionen“ wählen lässt. Die diesen Intro-Bildschirm untermalende Intro-Musik findet sich vor allem bei den Heimcomputern und -konsolen. Auch für sie gilt die Tatsache, von rigidem Timing und anderen Paradigmen der *Ingame*-Musik losgelöst zu sein. Intro-Bildschirme auf Arcade-Automaten besitzen i.d.R. keine musikalische Untermalung, da sie schnell zwischen Demo-Modus (Darstellung des Spielablaufs), High Score-Tabelle und Titelschirm wechseln. Bei den Spielszenen im Demo-Modus und den dabei teilweise stattfindenden Erklärungen der grundsätzlichen Spielregeln wird die Hintergrundmusik des Spiels ebenfalls meistens nicht abgespielt. Eine häufig verwendete Technik ist die, sporadisch Soundeffekte und/oder Musik abzuspielen, um die Aufmerksamkeit der Besucher der Spielhalle zu gewinnen und diese zum Spiel zu animieren.

Ending-Musik: Nach erfolgreichem Meistern eines Computerspiels, wenn vorhanden auch eines bestimmten Spielmodus (Arcade, Story) in diesen, bekommt der Spieler als Belohnung den Ausgang der Geschichte und in einer Art Abspann alle am Spiel beteiligten Personen (Credits / Staff Roll) zu sehen. Diese Sequenz nennt man „Ending“. Ending-Musiken stellen häufig das aufwendigste Arrangement des betreffenden Computerspiels dar. Auch wird in ihnen teilweise die Leistung des Spielers reflektiert, indem je nach Leistung z.B. ein Happy- oder Sad- bzw. Good- oder Normal-Ending mit entsprechender Musik abgespielt wird.

Die **Game Over-Musik** ist ebenso wie die häufig anzutreffende, den Tod einer Spielfigur markierende **Schlusskadenz** jene Musik, welche mit musikalischen Mitteln z.B. das Ableben der Spielfigur oder den endgültigen Verbrauch der verbleibenden Credits reflektiert. In wettkampfbasierten Spielen kann teilweise zwischen **Game Over Winner** und **Game Over Loser**-Themen unterschieden werden, Diese meist kurzen Phrasen greifen i.d.R. die Hintergrundmusik auf bzw. führen diese zu einem tonalen Ende. Diesen Melodien schließt sich, wenn vorhanden, der Continue-Bildschirm samt **Continue-Musik** an, um das Spiel mit einem neuen Kredit im vorher erreichten Level fortzusetzen. Ansonsten springt das Programm zum Intro-Bildschirm zurück und wartet auf einen erneuten Start durch den Benutzer.

Andere Elemente, die *Sekundäre* Spielmusik enthalten können sind die auch Cutscenes genannten Zwischensequenzen (**Cutscene-Musik**), ein Optionsmenü oder Spieler- bzw. Fahrzeug-Auswahlmenü (**Menü-Musik**), die Gratulation / Mitteilung über einen High Score-Platz sowie die Namenseingabe für die High Score - Tabelle (**Belohnungs-Musik, High Score Entry-Musik, High Score-Musik**¹³⁸).

Zu beachten ist, dass frei anwählbare Optionsmenüs i.d.R. nur bei Heimcomputern und -konsolen enthalten sind, ein Continue-Bildschirm andererseits standardmäßig nur bei Arcade-Automaten und deren Konvertierungen für Videokonsolen oder Heimcomputer auftauchen. Die anderen Elemente können je nach Spiel und Genre¹³⁹ gar nicht oder in wechselndem Umfang vorhanden sein.

Sekundäre Spielmusik ist grundsätzlich von einigen Einschränkungen befreit, denen *Ingame*-Musiken unterliegen, wie z.B. Prozessorlast oder der Reservierung einer eigenen Stimme für Soundeffekte. Somit ist sie einerseits häufig aufwendiger gestaltet, gleichzeitig je nach Position und damit Funktion im Spielprogramm auch simpler Natur: Intro-Musiken haben aufwendige Arrangements, Menü-Musiken bestehen aus relativ kurzen, einfachen Loops.

3.1.1.2 Ingame Spielmusik

Ingame Spielmusik umfasst alle Arten von Musik, die zeitlich direkt mit dem Spielgeschehen zusammenliegen, also die Interaktion von Spieler und Maschine untermalen. Wenn die Musik das Spielgeschehen dabei als Informationsgeber unterstützt, also im engen Sinne auf die Aktionen des Spielers eingeht und sich diesen anpasst, spricht man auch von *adaptiver* Musik, auf die im nächsten Kapitel näher eingegangen wird. Brian Schmidt entwirft ein archetypisches Modell für die Organisation der *Ingame* Spielmusik. Er versteht digitale Spiele vor allem als verschiedene Zustände des Spielgeschehens (*game states*), zwischen denen verschiedenen Überleitungen (*transitions*) und Bögen (*arcs*) bestehen:

„Any game can be thought of as consisting of a number of states and arcs connecting those states. A typical video game [...] machine can have dozens [of] these states and connecting arcs. As a player plays a game [...], arcs are taken depending on the actions taken by the player, and the current game state changes accordingly. These changes in game states are accompanied by corresponding changes in the background music. This is done by composing a musical score for each state node and connecting arc. Scores for game states are usually full and complete musical compositions, while scores for connecting arcs are typically short transition pieces that lead directly to the 'next game state' music. The arcs can be as simple as a drum fill

¹³⁸ In dem Spiel DIG DUG ist die High Score-Tabelle selbst nicht mit Musik unterlegt, dafür erklingt eine fröhliche Belohnungsmusik (♩016) für die höchste Punktezah, gefolgt von der hektischen High Score-Entry-Musik (♩017).

¹³⁹ Viele Spiele verzichten z.B. auf ein Spielfiguren-Auswahlmenü. Bei Prügelspielen wie STREET FIGHTER II (CAPCOM, 1991) ist dieses jedoch essentiell, genau so wie bei Rennspielen mit verschiedenen Fahrzeugen und unterschiedlichen Fahreigenschaften oder Sportspielen, die pro Spielcharakter eine andere Gewichtung der verschiedenen Fertigkeiten bieten.

or as complex as a harmonic cadence setting up a new lay and theme. When the game state changes, the music for the arc corresponding to the reason for the state change is played, usually a transition leading into the music for the next game state¹⁴⁰.

Nach dem Drücken der Start-Taste im Titel- oder Player Select-Bildschirm erklingt i.d.R. eine kurze **Vorspann-** oder **Vorspiel-Musik**, die je nach Länge des Vorspanns, Umfang und Genre des Spiels von einigen, einleitenden Takten (BUBBLE BOBBLE) bis zu einem eigenen, den grafischen Vorspann untermalenden Soundtrack (v.a. in neueren, auf ausführlich erzählten Hintergrundgeschichten basierenden Spielen) reichen kann. Vorspann-Musik ist seit Anfang der Arcade-Spiele auch dafür da, den Spieler auf das Spielgeschehen vorzubereiten, bei ihm für die nötige Aufmerksamkeit zu sorgen.

Sie ist die erste Art von auf breiter Basis implementierter Musik in Videospiele und markiert den Wechsel von *Sekundärer* zu *Ingame* Spielmusik. Ein frühes Beispiel stellt die einleitende Musik (♩018) für den Arcade-Hit PAC-MAN dar, die auch als kurzes Zwischenspiel nach einer bestimmten Levelanzahl (als eine Art früher Cutscene-Musik) auftaucht:



Abbildung 10 Die einleitende Melodie von Pac-Man

Im darauf folgenden Spielgeschehen fehlt jegliche Hintergrundmusik. Allein der lautmalerisch als „Wakka-Wakka“ berühmt gewordene Klang beim Auffressen der Punkte, verbunden mit einem Sirenen-Klang, der die von den Geistern ausgehende Gefahr symbolisiert, untermalt das Spiel. Auch SUPER MARIO BROS. und DIG DUG enthalten musikalische Vorspiele. Bei SUPER MARIO BROS. ist dieses allein ein Auftakt, der Teil der wiederholten Hintergrundmusik (♩019) bleibt, bei DIG DUG eine viertaktige, musikalische Phrase, die während des Abstiegs der Spielfigur in das unterirdische Spielfeld erklingt (♩020).

Hintergrundmusik (aus dem Englischen hat sich auch die Abkürzung **BGM** (für **Background Music** etabliert) ist die wohl gängigste Form von Musik in Computer- und Videospiele, wurde aber erst ab den 1980er Jahren zum Standard¹⁴¹. Der erste größere Hit in den Spielhallen, SPACE INVADERS (TAITO, 1978) besitzt bereits einen Soundtrack. Dieser besteht nur aus vier alternierenden, chromatisch auf- und absteigenden Bass-tönen, welche mit fortschreitendem Spiel immer schneller wiederholt werden (vgl. Collins 2003a: 2). Diese spezielle Anwendung von Klang markiert einen dritten Ort zwischen Soundeffekt (die Bewegung der angreifenden Aliens) und Hintergrundmusik (die eine unheilvolle Stimmung erzeugt)¹⁴².

¹⁴⁰ Schmidt, Brian: „Designing Sound Tracks for Coin-Op Games“ in: „Proceedings: 1989 Computer Music Conference“, 1989, S. 237-276. zit. n. Belinkie 1999: o.S., Hervorh. i. O.

¹⁴¹ Zwei der ersten Arcade-Automaten mit durchgängiger Hintergrundmusik sind CARNIVAL (SEGA, 1980; ♩021) und FROGGER (KONAMI, 1981; ♩022).

¹⁴² Leider fehlt dieser Klang in der MAME-Emulation von SPACE INVADERS.

Hintergrundmusik wird in der Zeit der Soundchips neben den Soundeffekten zum wichtigsten auditiven, vor allem aber identitätsstiftenden Element eines Computerspiels. Die Soundeffekte fordern meist ihren Tribut in Form eines der limitierten Klangkanäle. Chris Hülsbeck (2005a) erwähnt, dass Hintergrundmusik für die C64-Spiele i.d.R. nur zwei der drei Klangkanäle des SID-Chips benutzt. Das Spiel SUPER MARIO BROS. benutzt eine andere häufig zu findende Technik: Die Melodiestimme wird ausgeblendet, sobald ein Soundeffekt wiedergegeben wird.

Da ein Musikstück als Hintergrundmusik mangels größeren Speicherplatzes mitunter das ganze Spiel über als Endlosschleife zu hören ist, wird ihr zwangsläufig Aufmerksamkeit zuteil¹⁴³. Viele Spiele operieren mit Loops von unter einer halben Minute Länge, wobei im Verlauf der Jahre sowohl die Länge der Loops als auch die Anzahl und die Komplexität des Arrangements der verschiedenen verwendeten Liedsätze zunimmt. Wie Belinkie bemerkt, ist das Bestehen der Musik in den Ohren der Konsumenten angesichts einer häufig stundenlangen Repetition ein Hauptmerkmal für die Qualität einer Hintergrundmusik: „*The true test of game music is, whether a player can still stand a simple theme after hearing it repeated for an hour or more*“ (Belinkie 1999: o.S.).

Dass daher solche Stücke mit 'Ohrwurmqualität' immer wieder als „gute“ Spielmusik zitiert werden, erscheint naheliegend. Zwischen Qualität des Spiels und Qualität der Hintergrundmusik lässt sich jedoch keine zwingende Verbindung herstellen. Gleichzeitig konnte das musikalische Thema von SUPER MARIO BROS. (♩019) auf dem NES bei einer unrepräsentativen Befragung Belinkies unter amerikanischen College-Studenten von 66% der Studenten gesummt werden, auch wenn sie das Spiel seit Jahren nicht mehr gespielt hatten. Pidkameny bemerkt: „[I]f *Koji Kondo* [der Komponist des Super Mario Bros. Themas, N.D.] *had written the 'Mario Theme' for a game of poor quality, the music would never have been remembered*“ (Pidkameny 2002: o.S.). Diese Feststellung unterstreicht den großen Wiedererkennungswert eines Spieleklassikers über die gehörte Musik¹⁴⁴.

Ab Mitte der 1980er Jahre werden für verschiedene Level bzw. Welten in Computerspielen distinktive Hintergrundmusiken komponiert. SUPER MARIO BROS. ist eines der ersten und bekanntesten Beispiele dieser Entwicklung. Neben dem Thema der Oberwelt (♩019) gibt es ein Thema für die durch Röhren erreichbare Unterwelt (♩024), ein Thema für die Unterwasser-Welt (♩025) und eines für das finale Level (♩026), der Burg des Endgegners (vgl. Whalen 2004). All diese Themen stellen also Hintergrundmusiken dar, die im Spiel einer bestimmten Örtlichkeit entsprechen. Mit einer eigenen Hintergrundmusik gekennzeichnet werden auch die Boss-Stages / -Level genannten Kämpfe mit im Spiel vorhandenen (Zwischen-) Endgegnern (**Boss-Musik**).

¹⁴³ Eine Ausnahme im Kanon der Soundchipmusik markiert die über 16 Minuten lange, von Rob Hubbard für den SID komponierte Hintergrundmusik zu KNUCKLE BUSTERS (MELBOURNE HOUSE, 1986; ♩023).

¹⁴⁴ Ähnlich populär sind je nach Alter und Herkunft der Rezipienten Spielmusiken wie die Themen von BUBBLE BOBBLE, RAINBOW ISLANDS (TAITO, 1987), GIANA SISTERS oder TURRICAN II.

Die Musik, welche bei Erreichen eines Bonuslevels oder durch Aufsammeln bestimmter Items¹⁴⁵ mit entsprechenden Belohnungen für den Spieler abgespielt wird, stellt die **Bonusrunden-Musik** dar. Diese erklingt für eine gewisse Zeit (häufig verbunden mit einer temporären Unverwundbarkeit der Spielfigur) anstelle der Hintergrundmusik. Je nach Umfang des jeweiligen Titels gibt es verschiedene Bonusrunden pro Spiel, die teilweise mit Ortswechseln der Spielfigur verbunden sind. Häufig erklingen auch kurze musikalische Boni beim erfolgreichen Abschließen eines Levels, wie eine triumphale Fanfare oder eine die ständige Wiederholung des Themas der Hintergrundmusik beendende Schlusskadenz (**Level Complete-Musik**).

Für einen Wechsel der Hintergrundmusik werden in Spielen also die *game states* in Bezug auf Örtlichkeit und Fortschritt im Spiel abgefragt. Variationen der Hintergrundmusik können z.B. von *states* wie Zeitlimit oder Gesundheitszustand der Spielfigur abhängen. Durch die genannten Abhängigkeiten dieser *Ingame-Spiel*musik von den entsprechenden *game states* erhält diese ihren adaptiven Charakter.

3.1.2 Adaptive Audio / adaptive Musik für Soundchips

Der noch relativ junge Begriff *Adaptive (Game) Audio* bzw. adaptive Musik kennzeichnet nach Brandon (2003) die Beeinflussung des zu hörenden Audio-Materials durch die Eingaben des Nutzers / Spielers sowie in einer erweiterten Definition auch den umgekehrten Weg. Gleichwohl bevorzugen einige Autoren wie Ross (2001) für dasselbe Phänomen weiterhin den Begriff *interaktiver* Musik, um Gewicht auf die Wirkung der Musik auf den Spieler zu legen:

„Some might say that Adaptive Audio or even Reactive Audio is a better term for what I'm talking about but I think that these labels do not properly convey the idea that we want audio not to only react to a given situation or adapt to the changes in the environment but to also give a portent of things to come“ (Ross 2001: o.S.).

Aufgrund der Hervorhebung des Einspielens bestimmter Musik in Relation zu sich ändernden *game states*, wird hier dennoch der Begriff „adaptiv“ benutzt. Die Gedanken von Ross werden dadurch nicht hinfällig, zielen jedoch mehr auf die Rezeptionsprozesse und Affekte beim Spieler. Der Begriff *reaktiver* Musik wird ebenfalls synonym mit den anderen verwendet. Aktuelle Diskussion über adaptive Musik verwenden vor allem neuere Computerspiele, die nicht in den Zeitraum der Soundchips fallen. So bei Brandon (2003) oder Stockburger (2003) die beide auch darauf eingehen, in wie weit adaptive Soundeffekte das Spielgeschehen unterstützen und Informationsträger sind¹⁴⁶. Stockburgers Ansatz, Klangobjekte in Bezug auf ihre räumliche

¹⁴⁵ „Items“ bezeichnen aufzusammelnde Bonusgegenstände in Computerspielen.

¹⁴⁶ Brandon geht nach einer informativen Geschichtsschreibung (2003: 195-204) vor allem auf die Anwendungen von adaptiven Elementen in verschiedenen Genres wie Renn- und Musikspielen sowie Charakter-orientierten Action-Adventures ein (ebd.: 205ff). Spätestens seit *METAL GEAR SOLID 2* und *SPLINTER CELL* (UBI SOFT, 2002) ist die musikalische Reflektion verschiedener „Stress-Level“ durch zunehmend spannungsvolle Musik in modernen Spielen implementiert.

Verortung in METAL GEAR SOLID 2 (KONAMI, 2002) zu untersuchen, basiert auf die durch moderne Mehrkanal- und Halltechniken möglich gewordene, gestiegene Informationshaftigkeit von Klangobjekten, die den Spieler über Räumlichkeit und Beschaffenheit einer dreidimensionalen Spielumgebung aufklärt¹⁴⁷.

Für die Betrachtung von Soundchip-Musik sind diese nur bedingt relevant, da mit den damaligen klanglichen Möglichkeiten nur wenige der mittlerweile diskutierten Anwendungen umgesetzt werden konnten. Im Folgenden sollen die Möglichkeiten der adaptiven Anwendungen von Soundchip-Musik in Spielen erarbeitet werden. Diese sind wegweisend für spätere Entwicklungen. Es finden sich in den Spielen dieser Zeit nur partiell adaptive Elemente, die auf einige wenige Techniken zurückgreifen, welche im Folgenden vorgestellt werden.

3.1.2.1 Adaptivität durch Synchronisation von Eingabe und Musikwiedergabe

Als frühe adaptive Anwendungen von Audio-Material in Computerspielen nennt Brandon (2003: 202f) die mit Handlungen des Spielers synchronisierten *Soundeffekte* wie Explosionen bei ASTEROIDS oder die bereits erwähnten Geräusche der Punkte fressenden Spielfigur in PAC-MAN. Die Synchronisation von Benutzereingabe, ihrer Prozessierung und der Ausgabe des Ergebnisses gekoppelt mit akustischen Signalen findet sich bereits bei frühen Großrechnern und ist bis heute in allen Betriebssystemen implementiert. Für Computerspiele setzt sich diese Koppelung von grafischem und klanglichem Output vom Klang des PONG-Automats und der Soundeffekte beliebiger Action-Spiele der frühen Jahre bis heute fort.

Direkt an die Eingaben des Spielers gekoppelte bzw. mit dieser synchronisierte *Musik* findet sich indes bei auf Soundchips basierender Spielmusik sehr selten. In dem Spiel DIG DUG wurde die Hintergrundmusik (♫027) jedoch auf eine sehr direkte Art und Weise adaptiv implementiert, indem das Abspielen der Musik und die Bewegung der Spielfigur miteinander synchronisiert sind: Sobald der Spieler sich nicht mehr bewegt, stoppt auch die Musikwiedergabe, um danach am selben Punkt des Musikstückes fortzusetzen (♫028). DIG DUG scheint das einzige Spiel mit derart direktem Feedback in der Verwendung von adaptiver Musik zu sein.

Erst in neueren, auf der Einspielung von *streamed* Audio¹⁴⁸ basierenden Spielen für PLAYSTATION 1 und 2, wie den Titeln PARAPPA THE RAPPER 1 und 2 (SONY, 1996, 2001) oder FREQUENCY und AMPLITUDE (SONY, 2001, 2003) findet sich das Konzept der Koppelung von Eingaben des Spielers mit der Hintergrundmusik wieder. Diese Spiele markieren die auch „Bemani“ oder „Rhythm Action Games“ genannten Musikspiele, welche die rhythmischen Eingaben des Spielers abfragen und ein Musikstück diesen entsprechend ausschmücken bzw. anpassen (vgl. Edge 2003).

¹⁴⁷ Die Modellierung entsprechender spatialer Audio-Engines erläutert Boer (2003: 537ff).

¹⁴⁸ Wie erwähnt können mittels Streaming auch komprimierte Formate in die Engines eingebunden werden. Die SONY PLAYSTATION bietet ebenso wie das SEGA SATURN die Möglichkeit, dem MOD-Format ähnelnde XA-Files abzuspielen, bei denen z.B. Loops verwendet werden können.

Ein früher Vertreter¹⁴⁹ dieser ist BEATMANIA (KONAMI, 1997), welches japanisch abgekürzt „be ma ni“ ergibt. Einige „Rhythm Action Games“ verwenden nachgebildete Perkussionsinstrumente als Controller wie SAMBA DE AMIGO (SEGA DREAMCAST; SEGA, 2000) oder DONKEY KONGA (NINTENDO GAMECUBE; NINTENDO, 2004).

3.1.2.2 Adaptivität durch Bruch / Wechsel der Musik

Der Wechsel der Hintergrundmusik aufgrund einer Änderung der *game states* kennzeichnet grundsätzlich ein adaptives Moment, wie es Brandon für das Spiel VANGUARD (SNK, 1981) schildert¹⁵⁰ und kommt in vielen Spielen vor. Diese Brüche in der Musik kennzeichnen auch den Wechsel zwischen verschiedenen *game states* und sind somit für den Spieler als Orientierung wichtig. Die mittels von Brüchen erzeugte Adaptivität ist laut Brandon (2003: 204) für 90% aller Spiele, die Adaptive Audio verwenden, die bevorzugte Technik. Dabei wird der Bruch zwischen den beiden aufeinander folgenden Musikstücken durch die von Schmidt beschriebenen Bögen erwirkt. In Spielen der 1980er Jahre sind diese Bögen statt längerer melodischer Phrasen häufig nur kurze Soundeffekte, die zum nächsten *game state* samt neuer Musik überleiten.

In vielen Fällen beginnt die nächste Melodie auch ohne überleitende Bögen. Der harte Bruch kann durch die Verwendung eines gemeinsamen tonalen Zentrums bzw. sinnvoller Kadenzen gemildert werden. Er kann jedoch auch intendiert sein, um Aufmerksamkeit zu erzeugen, wie die Sirene in dem Spiel BOMB JACK (TEKHAN, 1984) zeigt: Hier wird nach einer bestimmten Anzahl aufgesammlter Bomben die Hintergrundmusik durch eine Sirene ausgetauscht. Währenddessen erscheint ein Item, das sich über den Bildschirm bewegt. Nach dessen Aufsammeln kann der Spieler in einer zeitlich begrenzten Bonusrunde die vorher unbesiegbaren Monster auffressen. Die Sirene deckt dabei den Tonumfang von über einer Oktave ab. Die sich nach dem Aufsammeln anschließende Bonusrunde dauert genau einen viertaktigen Turnaround von Tonika, Moll-Parallele, Subdominante und Dominante (♩029), wonach die dem jeweiligen Level entsprechende Hintergrundmusik (auf der Tonika) von vorne beginnt und das Spiel wieder zum Ausgangs-*state* (keine Unverwundbarkeit, dafür tauchen die Monster wieder auf) zurückspringt (♩007/♩007b).

Ein Beispiel für die durch einen harten Bruch der Musik erzeugten Adaptivität ist die in vielen Spielen implementierte Schlusskadenz beim Tod der Spielfigur etwa in SUPER MARIO BROS. (♩030)(NINTENDO, 1985), welche Whalen (2004) zusammen mit der Game Over-Melodie (♩031) mit bekannten Jingles von Gewinn-Shows im Privatfernsehen vergleicht:

¹⁴⁹ Die britische Zeitschrift EDGE arbeitet einige Vorläufer der Rhythm Action Games heraus, die jedoch noch nicht den Rhythmus als Spielelement verwenden. PARAPPA THE RAPPER 1, ein Spiel, in dem der Spieler mittels rhythmischer Kombinationen auf dem Joypad ein Hund namens „Parappa“ rappen lässt, kann daher als erstes Rhythm Action Game bezeichnet werden (vgl. Edge 2003: 58).

¹⁵⁰ Wenn der Spieler in VANGUARD durch ein Treibstofflager fliegt, wird die Hintergrundmusik durch ein Musikstück triumphierenden Charakters ausgetauscht, solange die dadurch erlangte Unverwundbarkeit anhält (~15 Sekunden) (vgl. Brandon 2003: 203f).

„*'Dying' in Super Mario Bros. produces a staccato pulse followed by a conciliatory musical cadence reminiscent of the music one hears upon a contestant's mis-estimating the value of a vacuum cleaner or dish set on The Price is Right*“ (Whalen 2004: o.S. Hervorh. i. O.).

Auch das Spiel PARODIUS (KONAMI, 1990) verwendet eine ähnliche Schlussphrase beim Ableben (♩032). Es lassen sich viele weitere Beispiele für *game state* - Änderungen begleitende, musikalische Elemente finden: In DIG DUG ebenso wie bei BOMB JACK und SUPER MARIO BROS. existieren kurze Level-Complete-Musiken (♩033, ♩034, ♩035), in denen eine kleine Melodie das vorher penetrant sich wiederholende Thema abschließt. Gerade bei der musikalischen Simplizität des DIG DUG-Themas ist diese musikalische „Erlösung“ als unterstützendes Moment für das erleichternde Gefühl, ein Level gelöst zu haben, zu werten.

Adaptivität durch Wechsel der Hintergrundmusik lässt sich auch im Adventure-Genre häufig finden, wenn Orte und Ereignisse mit bestimmten Hintergrundmusiken untermalt sind. Beispielsweise finden sich in Rollenspielen musikalische Themen für Kampfszenen oder Orte wie Shops, Gasthäuser, das eigene Heim, Kirchen, Tempel etc.. Der informative Charakter dieser an Orte gekoppelten Musik ist hingegen gering, verglichen mit den schneller aufeinander folgenden *game states* in Actionspielen. Die Adaptivität auf technischer Ebene bedingt an dieser Stelle eher eine durch die jeweilige Musik assoziierte Stimmung als Affekt beim Spieler hervorrufen, der z.B. innerhalb des Rollenspiels nach einem Abenteuer wieder in Sicherheit zu Hause speichern kann, neue Informationen sammeln kann, gewisse Items im Shop kauft o.ä.. Dafür können auch komponierte Leitmotive von Nutzen sein., wie Pidkameny für FINAL FANTASY III (SQUARE, 1994)¹⁵¹ auf dem SNES erwähnt (Pidkameny 2002, o.S.).

3.1.2.3 Adaptivität bei Fortschritt

Die Technik, das Vorankommen des Spielers in der Fülle des Arrangements der Hintergrundmusik zu reflektieren, ist spätestens seit modernen Rhythm Action Games wie AMPLITUDE oder REZ (SONY 2001, SEGA, 1999)¹⁵² eine diskutierte Anwendung von Musik in Computerspielen¹⁵³. Einige Spiele, die bereits in den 1980er Jahren eine Abhängigkeit zwischen Spielfortschritt und Hintergrundmusik besitzen, werden hier

¹⁵¹ Pidkameny weist darauf hin, dass das Spiel in Japan als FINAL FANTASY VI bekannt ist..

¹⁵² Bei AMPLITUDE muss der Spieler bestimmte Elemente der jeweiligen Patterns durch zeitlich exaktes, rhythmisches Drücken der Controllerknöpfe „aktivieren“. REZ ist ein sog. „Rail-Shooter“, bei dem der Spieler durch einen abstrakten, aus Drahtgitter-Modellen bestehenden Raum fliegt und mit dem Zeiger Gegner „einloggen“ muss, die danach effektiv in ihre Polygone zerfallen und dabei jeweils eigene Soundeffekte ansteuern. Jedes der 5 Level im Spiel enthält 10 Sub-Level, die bei Erreichen das Arrangement des aus Techno-Musik bestehenden Soundtrack aufbauen.

¹⁵³ Pidkameny geht vor allem auf den Punkt der synchronisierten Soundeffekte in REZ ausgiebiger ein, vernachlässigt aber den Aspekt der aufeinander folgenden Level der Musik. Für ihn ist die u.a. aus Cartoon-Filmen stammende Technik des „Mickey Mousing“, die das zeitliche Zusammenfallen von Bildschirmgeschehen und Soundeffekten bezeichnet, bei REZ offensichtlich. (vgl. Pidkameny 2002: o.S.).

vorgestellt. Die meisten Anwendungen sind dabei noch recht simpler Natur: Um die gewünschten Effekte als Spieler zu hören, muss dieser gut genug spielen, um in dem Spiel möglichst weit ohne Scheitern voranzukommen.

So enthält das Spiel *AZTEC CHALLENGE* (COSMI, 1984) auf dem *COMMODORE C64* eine Hintergrundmusik (♩036), die mit der Zeit immer weiter ausgeschmückt wird, wobei das Ergebnis nach heutigen Gesichtspunkten unspektakulär klingt, aber für damalige Verhältnisse ein außergewöhnliches Konzept darstellte¹⁵⁴.

Neben dem sukzessiven Aufbau einer Hintergrundmusik durch Hinzufügen musikalischer Elemente wird das Vorankommen in einem Spiel auch anders musikalisch reflektiert. So verwendet das Spiel *CARNIVAL* (SEGA, 1980; ♩021) ähnlich wie das bereits erwähnte *SPACE INVADERS* dafür die Technik der Tempo- und Tonhöhensteigerung. Bei *SPACE INVADERS* nimmt die Geschwindigkeit des pulsierenden Klangs dabei mit geringerer Entfernung der angreifenden Aliens zu, die Hintergrundmusik von *CARNIVAL* wird pro Wiederholung des gesamten Musikstücks schneller abgespielt und um ca. einen Halbtonschritt nach oben transponiert.

Die Technik der zunehmenden Abspielgeschwindigkeit einer Hintergrundmusik fungiert darüber hinaus in vielen Spielen als unterstützendes Zeichen für ein nahendes Ende der zur Verfügung stehenden Zeit und somit als Indikator für gebotene Eile bzw. drohende Gefahr. So soll der Spieler die restlichen Gegner töten, bevor die Gefahr der bösen Geister naht, wie in *BUBBLE BOBBLE*, oder er wird daran erinnert, vor Ablauf der zur Verfügung stehenden Zeit das Ende des betreffenden Levels zu erreichen (vgl. Whalen 2004: o.S.).

Eine eingeblendete Aufforderung (i.d.R. „Hurry Up!“) unterstützt häufig den musikalischen Appell an den Spieler, sich zu beeilen. Zumeist wird auch ein passender Soundeffekt eingespielt, wie z.B. in *RAINBOW ISLANDS* (♩037) und *SUPER MARIO BROS.* (♩038).

3.1.2.4 Adaptive Soundeffekte: die Technik des Mickey Mousing

Whalen (2004) weist darauf hin, dass im Gegensatz zum häufig geführten Vergleich des Mediums der Computerspiele mit dem Medium Film¹⁵⁵ ein Vergleich zwischen Animationsfilmen (Cartoons) und Computerspielen angemessener erscheine. Gerade für die Betrachtung von Computerspielen vor der CD-ROM-Technologie gilt dieses in besonderem Maße, da erst durch die auf der CD vorhandenen Speicherkapazität die verstärkte Produktion von grafisch opulenteren Spielen einsetzt¹⁵⁶. Er bemerkt, dass sowohl

¹⁵⁴ Dieser Effekt führte zu einer großen Faszination, die noch heute in Äußerungen von damals aktiven Spielern über Spielmusik erhalten bleibt. Der Autor und Wissenschaftler Mathias Mertens z.B. spielte das Spiel mit zwei Schulfreunden vor allem der Musik wegen. So trafen sich die drei vor dem C64, wobei jeweils einer im Spiel vorankommen musste ohne zu versagen (da dadurch die Musik abgebrochen worden wäre), während die anderen beiden „moschen“ durften, also den Kopf rhythmisch hin und her bewegten. (persönliches Gespräch mit dem Autor, 26.08.2003)

¹⁵⁵ Zu der Diskussion von Film und Computerspielen vgl. z.B. den Sammelband „ScreenPlay“ (King / Krzywinska 2002)

¹⁵⁶ Erst durch die zunehmende Implementierung filmischer Elemente entstehen denn auch die „filmischen“ Unter-Genres der Computerspiele. Diese arbeiten in den sog. Cutscenes und auch während des Spielgeschehens in 3D-Umgebungen mit Kameraeinstellungen, Schnitten und nicht zuletzt einer den Szenen entsprechenden (Film-)Musik, um Spannung zu erzeugen. Als Paradebeispiel darf das „Survival-Horror“-Genre gelten (z.B. die Spielserien *RESIDENT EVIL* (CAPCOM, ab 1996) oder *SILENT HILL* (KONAMI, ab 1999)).

Spiele als auch Animationsfilme reduktive Elemente enthalten, die eine Wirklichkeit eher emulieren denn simulieren, d.h. die Welten in Animationsfilmen wie auch in Computerspielen entsprechen weniger einer Nachbildung der Realität, in beiden Fällen wird vielmehr eine Welt konstruiert, in der viele Elemente aus unserem Alltag entweder fehlen, in ihrem Detailgrad reduziert oder den Fähigkeiten der Spielfigur angepasst werden. Mittels dieser Behandlung wird ein Gewicht auf die vorhandenen Elemente gelegt.

Ebenso werden vor allem beim Cartoon synchronisierte Soundeffekte und musikalische Elemente verwendet, um bestimmte Bewegungen auf dem Bildschirm bzw. im Spiel stattfindende Handlungen zu untermalen. Whalen nennt als Beispiel die musikalische Untermalung von Gewalt in der Filmserie „The three Stooges“, durch welche die dargestellten Aktionen mittels des akustischen Kontrapunkts (z.B. ein Paukenschlag bei einem Schlag in die Magenkuhle) ihren Realismus verlieren und die Zuschauer zum Lachen bringen. Danach zitiert Whalen Studien, denen zufolge die klangliche und musikalische Untermalung von bewegten Figuren dafür sorgt, dass diese lebendiger wahrgenommen werden als ohne Untermalung. Er benutzt hierfür den aus der Filmmusik bekannten Begriff des *Mickey Mousing* für die Synchronisation von Bewegung und auditiven Elementen.

In Computerspielen ist die Synchronisation von Spielgeschehen und Soundeffekt notwendig für eine kohärente Präsentation des auf dem Bildschirm Gezeigten, als Unterstützung des Feedbacks, dass der Spieler etwas am Output des Spiels ändert. Daher sind synchronisierte Soundeffekte meist implementiert¹⁵⁷. Gänzlich „stumme“ Computerspiele finden sich wenige, und wenn, dann vor allem auf Plattformen mit einer mangelhaften Klangerzeugung (wie dem ATARI VCS, SINCLAIR ZX SPECTRUM, APPLE II oder IBM PC), sowie in Genres, die wenig direktes Feedback benötigen, wie Adventure- oder Strategiespiele.

Schon PONG gab, wie erwähnt, ein akustisches Feedback für jede Schlägerberührung des Balls ab. In späteren Spielen werden Aktionen wie Springen, Laufen, Schießen etc. i.d.R. akustisch untermalt: Diese Synchronisation durch Soundeffekte ist in digitalen Spielen allgegenwärtig. Die sonische Thematisierung von Bewegung im Sinne des *Mickey Mousing* wird meist mit Glissandi innerhalb der Soundeffekte erreicht. So zum Beispiel beim Springen in BUBBLE BOBBLE, durch Motorgeräusche in frühen Autorennspielen, die Geschwindigkeit reflektieren (POLE POSITION, NAMCO 1982) oder hinunterfallenden Gefahren wie bei den Monstern in BOMB JACK (diese werden durch absteigende Glissandi markiert ♪007). Auch werden musikalische Parameter in Soundeffekte integriert. So liegt der Hüpf-Klang beim „großen“ Mario (in SUPER MARIO BROS.) eine Oktave tiefer als beim „kleinen“ Mario (vgl. Whalen 2004). Bei MR. DO! (UNIVERSAL, 1982) werden die nacheinander aufgesammelten Kirschen mit einer aufsteigenden Tonleiter charakterisiert, wodurch die Zahl von eingesammelten Boni und damit die steigenden Punkte reflektiert werden.

¹⁵⁷ TENNIS FOR TWO (1958, vgl. Burnham 2001: 28) sowie SPACE WAR ! (1961) gelten als Vorläufer der modernen digitalen Spiele. Sie waren gänzlich ohne Soundeffekte programmiert, auch das bereits erwähnte MAGNAVOX ODYSSEY (1972) enthält keine Klangerzeugung. Frühe Beispiele für kleine, die Klanguausgabe von Großrechnern nutzende Anwendungen stellt ein bereits in den 1960ern geschriebenes Demonstrationsprogramm für den Whirlwind Computer namens BOUNCING BALL dar. Es bildet einen hüpfenden Ball auf dem Kathodenstrahlmonitor ab, wobei das Hüpfen mit einem lauten „thok“ aus dem Lautsprecher synchronisiert ist (vgl. Graetz 2001: 44f) (vgl. auch Fußnote 174)

Ganz anders sieht es mit der *musikalischen* Untermalung von Computerspielen. Die Nähe der Technik zum *Mickey Mousing* bei musikalischen Einspielungen mit starker Nähe zu Soundeffekten, wie den erwähnten Schlusskadenzen von gescheiterten Spielfiguren liegt auf der Hand. Durchkomponierte Hintergrundmusiken für Soundchips, die synchron zum Spielgeschehen laufen, sind dem Autor nicht bekannt. Auch gestaltet sich ihre Umsetzung technisch schwierig, da sich das konkrete Spielgeschehen eines Computerspiels vom zeitlichen Verlauf her aufgrund seiner Abhängigkeit vom Input des Spielers nicht genau hervorsagen lässt. Vor allem sind es daher z.B. bestimmte Stilistiken der Hintergrundmusik, die den Affekt zur Bewegung unterstützen oder Bewegung thematisieren.

Ein diese Überlegung illustrierendes Beispiel ist das 'fetzige', mit Sambarhythmus unterlegte Thema (♩039), wenn Mario in SUPER MARIO BROS. einen Stern auffängt und danach für einige Zeit Unsterblichkeit erlangt¹⁵⁸. Die durch eine entsprechende Musik beim Spieler hervorgerufene Stimmung kann also zu einer dem *Mickey Mousing* nahen Erfahrung führen¹⁵⁹, die jedoch von der Rezeption bedingt wird¹⁶⁰.

¹⁵⁸ Dieses Beispiel kann ebenso als Adaptivität durch Bruch bzw. Wechsel der Musik gewertet werden (Kapitel 3.1.2.2).

¹⁵⁹ Laut Whalen ermutigt das *beschwingte* SUPER MARIO BROS. - Thema (♩019) den Spieler, in der Welt voran zu kommen. Das *mysteriöse* „Unterwelt“-Thema (♩024) bzw. die dem finalen Level unterlegte *bedrohliche* Musik (♩026) sorgen ebenfalls für eine Verstärkung dieser jeweiligen Stimmung (vgl. Whalen 2004).

¹⁶⁰ Das erwähnte Beispiel einer ortsabhängigen Hintergrundmusik in einem Rollenspiel, welche im Spieler z.B. ein Gefühl der Heimkehr hervorruft, basiert auf ähnlichen Affekten. Ebenfalls die in Japan verbreitete Soundtrackkultur (Stichwort: „Memorial Albums“, vgl. Kap. 2.4., S.34).

3.1.3 Zwischenfazit: Funktionale Aspekte von Computerspielmusik

Soundchip-Musik in Computerspielen zeigt sich nach der erfolgten Aufgliederung differenzierter als auf den ersten Blick anzunehmen. Vor allem legen die verschiedenen Verwendungen von Musik und Klang mit adaptiven Hintergrund nahe, dass eine dem Medium gerecht werdende Aufarbeitung des Phänomens auch eine qualitative Analyse der in einem Computerspiel integrierten Musik benötigt. Die folgende Grafik bietet eine modellhafte Übersicht der vorgestellten, während der Spielhandlungen mitunter adaptiv verwendeten, musikalischen Objekte:

Als Computerspiel verkaufte Produkt													
Sekundäre Spielmusik			Ingame Spielmusik					Sekundäre Spielmusik					
Hacktro-/Cracktro-Musik	Intro-Musik	Menü-Musik	Vorspiel / Auftakt	Hintergrundmusik Level n ¹⁶¹	Level Complete-Musik	Bonusrunde zw. Levels	Hintergrundmusik Level n+1 ¹⁶¹	Schlusskadenz bei Tod	Continue-Musik	High Score Entry-Musik	High Score-Musik	Game Over-Musik	
				evtl. adaptiv oder unterbrochen v.: -Bonusrunden-Musik -Hintergrundmusik			evtl. adaptiv oder unterbrochen v.: -Bonusrunden-Musik -Hintergrundmusik						
Loader-Musik	Titel-Musik	Player Select-Musik											
<i>Nach erfolgreichem Durchspielen folgen auf das Spielgeschehen:</i>									Gratulations-Musik	Ending-Musik	High Score Entry-Musik	High Score-Musik	Game Over-Musik

Tabelle 8 Klassifizierung der verschiedenen Musiktypen in Computerspielen

Die x-Achse markiert die zeitliche Abfolge des dem Spiels zugrunde liegenden Programms, die verschiedenen *Ingame* Musiken werden den *game states* entsprechend abgewechselt. Sofern vorhanden sind Cracktro-, Loader- und Intro-Musik nur einmal am Anfang (beim Laden) des Spiels zu hören, der danach folgende Ablauf ist zirkulär: Bei einem vorzeitigen Scheitern und darauf folgenden Neustart des Spiels bzw. Levels wird die Abfolge unterbrochen und beginnt wieder am Anfang des Levels oder dem Titel-Bildschirm. Sie ist daher nicht linear zu sehen. Kurze, nur einige Takte lange und nicht wiederholte musikalische Phrasen sind in der Übersicht vertikal beschriftet und mit entsprechend geringer Breite gekennzeichnet.

Aufgrund der unterschiedlichen Genres und differierenden Spielelemente sind nicht in jedem Spiel alle Musiktypen enthalten (z.B. Bonusrunden-Musik). Ebenso ist die Abfolge von Game Over-Musik und den beiden High Score-Musiken in vielen Spielen umgekehrt. Cracktro- oder Hacktro-Musik tauchen meist nur in entsprechenden Programmen für Heimcomputer auf¹⁶², sind also nicht Teil der verkauften Produkte, die jedoch häufig Loader-Musik besitzen. Auch bestimmte Spielelemente wie der Continue-Bildschirm tauchen in einigen Spielgenres nur selten auf.

¹⁶¹ Bei Spielstart: n=1. Neben Hintergrundmusik für die verschiedenen Level werden hierunter auch die *n* verschiedenen Hintergrundmusiken von Endgegner-Kämpfen oder bestimmten Örtlichkeiten verstanden.

¹⁶² vgl. S.57f, Kap. 3.1.1.1.

Die oben erläuterte, modellhafte Übersicht wurde entwickelt, um die funktionale Medienrealität von Soundchip-Musik fassbar zu machen. Des Weiteren kann sie dem sich in Gründung befindlichen Diskurs über Musik in Computerspielen eine mögliche Grundlage zur Klassifizierung bieten. Im Folgenden wird nach der Anwendung der erarbeiteten Kategorien auf das Beispiel BUBBLE BOBBLE von weiteren formellen und kategorialen Einteilungen Abstand genommen werden und stattdessen die Gestaltung von Musik für Soundchips im Spiegel der technischen Beschränkungen in den Mittelpunkt der Betrachtung rücken.

3.1.4 Fallbeispiel: Bubble Bobble

Am Beispiel des Plattformspiels¹⁶³ BUBBLE BOBBLE sollen die verschiedenen Musiken in Abhängigkeiten zu den sich ändernden *game states* erläutert werden¹⁶⁴. Wenn der Stecker des Automaten mit dem Stromnetz verbunden wird, meldet sich das Spiel mit einer Tonleiter (♩040), noch bevor ein Bild auf dem Bildschirm erschienen ist. Diese akustische Rückkoppelung zu Beginn eines Spiels ist ein häufiges Phänomen bei Automaten ab Mitte der 1980er Jahre¹⁶⁵, sie ist in Emulationen jedoch nicht immer vorhanden.

Nach Münzeinwurf und Spielstart ertönt ein kurzes, in der gewählten Notation (Abb. 11) achttaktiges Vorspiel (♩041), das den Vorspann des Spiels untermalt. Mit dem Auftakt (ab Takt 9) beginnt die Wiederholung der Hintergrundmusik (Takt 9 bis Ende; ♩042), die durch ihre zweiteilige Liedform und die sehr eingängige Melodie gekennzeichnet ist.

¹⁶³ Als Plattformspiel oder auch Plattformer wird ein Vorläufer bzw. Untergenre des Jump'n Run verstanden. Das Spielgeschehen ist i.d.R. von dem Springen mit der Spielfigur auf verschiedene Plattformen geprägt, das gesamte Level dabei auf einem Bildschirm sichtbar. Der Spieler versucht verschiedene Gegenstände aufzusammeln ohne die Feinde zu berühren und kann diese ausschalten (vgl. Wirsig 2003: 363). Neben BUBBLE BOBBLE zählt auch das bereits erwähnte BOMB JACK zu den Plattformern.

¹⁶⁴ Die „Bubble Bobble FAQ“ (Hall o.J.) benennt die im Spiel enthaltenen Zähler, welche die verschiedenen *game states* beeinflussen. Die meisten hiervon bedingen jedoch das Auftauchen bestimmter Items, nur wenige führen zu einem Wechsel der eingespielten Musik.

¹⁶⁵ Der Spielautomaten-Spezialist Tobias Jachmann erläutert dazu in einer Email: „*Es war durchaus üblich dass die Automaten als Statusmeldung eine Tonfolge spielen. [...] Der Soundchip muß erst mit Daten versorgt werden, beim Start kann er einfach irgendwas abspielen, bis der ganze Rechner läuft und er sein Programm bekommt. Andere Games spielen einfach als korrekte Statusmeldung einen Ton ab. [...] Dabei gibt es verschiedene Ansätze. Einige spielen bloß einen Ton, um zu zeigen, dass der Soundchip funktioniert, andere melden mehr. [...] Atari z.B. zeigt durch die Tonhöhe und Folge an, welches ROM defekt ist.*“ (Email an den Autor, 28.06.2005).

The image displays a musical score for the background music of the game 'Bubble Bobble'. It consists of six systems of music, each with a melody line (treble clef) and a bass line (bass clef). The score begins with a tempo marking of $\text{♩} = 280$. The music is written in a key signature of one flat (B-flat) and a 2/2 time signature. The score includes various musical notations such as rests, notes, and slurs. There are two first endings (marked '1.') and two second endings (marked '2.'). The piece concludes with the instruction 'D.S.' (Da Capo).

Abbildung 11 Hintergrundmusik von Bubble Bobble mit Vorspiel (nur Melodie- und Bassstimme)

Diese wird das gesamte Spiel über, außer bei Unterbrechungen durch Bonusrunden oder Erreichen eines versteckten Levels, wiederholt. Bei Ablauf eines je nach Level variablen Timers folgt eine „Hurry up!“-Einblendung mit Sirenen-Soundeffekt, woraufhin das Thema schneller abgespielt wird ($\text{♩}043$). Diese Anwendung der Musik ist in zweierlei Hinsicht als adaptiv zu bezeichnen: Einerseits die beschriebene „Adaptivität bei Fortschritt“, bei der nach Ablauf des Zeitlimits ein Tempowechsel stattfindet, um den Spieler auf die drohende Gefahr der Geister hinzuweisen. Andererseits handelt es sich bei der Sirene um „Adaptivität durch Bruch“, die Aufmerksamkeit hervorruft.

Das Erreichen der Aufmerksamkeit wird dadurch verstärkt, dass vor Einspielung der Sirene die Musikausgabe zusätzlich für ca. eine halbe Sekunde aussetzt ($\text{♩}044$). Gleichzeitig stellt diese Pause eine musikalisch elegante Lösung der Überleitung zwischen Hintergrundmusik, Soundeffekt und Tempowechsel dar.

Im Fall der folgenden drei *game state*-Änderungen wird das sich wiederholende Thema durch die aus einem viertaktigen Turnaround bestehende Bonusrundenmusik (♩045, Abb. 12) unterbrochen:

- Alle Buchstabenblasen, die das Wort E-X-T-E-N-D bilden, sind vom Spieler aufgesammelt worden, woraufhin das Level beendet wird und ein Bildschirm voll Blumen mit der Aufschrift „Nice 1P !“¹⁶⁶ erscheint. Als Belohnung erhält der Spieler ein Extra-Leben. Dieses freudige Ereignis wird außerdem durch einen längeren Soundeffekt eingeleitet (♩046).
- Das Item „Trank“ wird aufgesammelt, wodurch eine ungerade Anzahl von Items (je nach Farbe des Tranks Regenbögen, Blumen, Noten oder Kleeblätter) in dem Level erscheint, die während eines bestimmten Zeitlimits aufgesammelt werden muss, um 100.000 Extrapunkte zu erhalten.
- Das Item „blinkendes Herz“ wird aufgesammelt, wodurch die Spielfigur beschleunigt wird und alle Feinde eine bestimmte Zeit lang wehrlos auf ihrer Stelle verharren. Solange die Bonusrundenmusik erklingt, können die Feinde durch einfache Berührung ausgeschaltet werden.



Abbildung 12 Bonusrundenmusik von Bubble Bobble (nur Melodie- und Bassstimme)

Nur bei den ersten beiden *game state*-Änderungen wird der Wechsel von Hintergrund zu Bonusrundenmusik durch einen Soundeffekt eingeleitet. Nach Verlust des letzten verbleibenden Lebens kommt ein High Score-Bildschirm, auf dem sich der Spieler ggf. in die Bestenliste eintragen darf, abgelöst durch den sog. „How far did you reach?“-Bildschirm, in welchem die eigene Leistung mit der besten in der High Score gespeicherten verglichen wird. Beide sind unterlegt mit der High Score-Musik (♩047):

¹⁶⁶ Wenn der zweite Spieler im 2-Spieler-Modus dasselbe erreicht, ist die Einblendung dieselbe, außer dass ein „Nice 2P !“ erscheint.



Abbildung 13 High Score-Musik von Bubble Bobble (nur Melodie- und Bassstimme)

Wenn der Spieler auf diesem Bildschirm einen der Feuerknöpfe drückt oder einige Wiederholungen der High Score-Musik erfolgt sind, erscheint der Game Over-Bildschirm, in dem eine Kadenz die Highscore-Musik abschließt (♩048):



Abbildung 14 Schlusskadenz von Bubble Bobble (nur Melodiestimme)

Neben den genannten Musiktypen existieren für die verschiedenen Konvertierungen von BUBBLE BOBBLE teilweise weitere, wie Loader- und Titel-Musik (z.B. auf dem C64, auf welchem andererseits die Bonusrundenmusik nicht bei dem aufgesammelten Item „Trank“ eingespielt wird). Da das sog. Continue auch bei der Arcade-Version nur im laufenden 2-Spieler-Spiel möglich ist und dieses ohne eigenen Continue-Bildschirm durch Drücken der Start-Taste bestätigt wird, existiert keine derartige Melodie.

Bei erfolgreichem Erreichen des Levels 20, 30 oder 40 ohne Verlust eines einzigen Lebens wird der Spieler mit einem versteckten Level belohnt, in dem eine große Zahl Edelsteine einzusammeln sind. Auch dieses ist mit einer eigenen Melodie versehen:

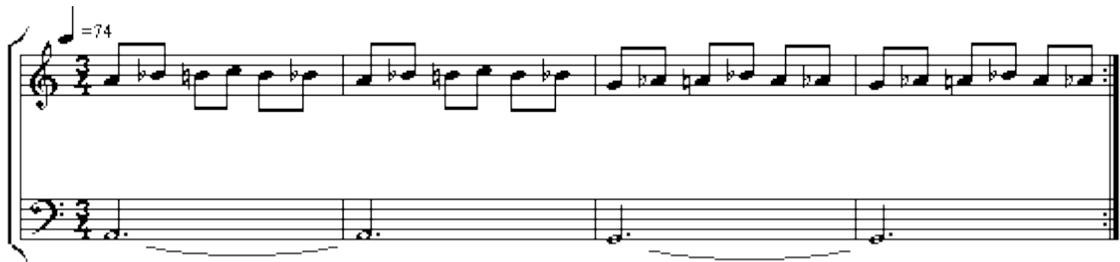


Abbildung 15 Hintergrundmusik des versteckten Levels in Bubble Bobble (nur Melodie- und Bassstimme)

Diese Hintergrundmusik basiert ebenso wie die den Endkampf untermalende Boss-Musik von BUBBLE BOBBLE auf alternierenden, chromatischen Läufen zwischen Grundton und Terz (auf ein Notenbeispiel der Boss-Musik wird daher verzichtet). Beide haben durch die Verwendung von kleinen Sekunden einen unheilvollen Charakter, der etwas Bedrohliches ausstrahlt. Außerdem existieren zwei verschiedene Ending-Musiken.

Auffallend ist, dass während des gesamten Spielgeschehens von BUBBLE BOBBLE (von Endgegner-Kampf und verstecktem Level abgesehen) lediglich zwei verschiedene *Ingame*-Spielmusiken (Hintergrundmusik und Bonusrundenmusik) erklingen. Durch die mit dem Spielgeschehen synchronisierten Soundeffekte sowie der Aufmerksamkeit, die der Spieler diesem entgegenbringen muss, um ein erfolgreiches Spiel spielen zu können, ist diese Monotonie jedoch kaum auffällig. Diese Eigenart ihrer Rezeption verweist auf einen interessanten Aspekt für die Bewertung von Computerspielmusik: Scheinbar wird die Hintergrundmusik nicht das gesamte Spiel über mit einer gleichbleibenden Aufmerksamkeit wahrgenommen, verbleibt aber nach beendtem Spiel längere Zeit im Gedächtnis. Gleichzeitig sorgt die bei positiv wahrgenommenen Ereignissen eingespielte Bonusrundenmusik für eine Abwechslung, die mit einer kurzen visuellen wie auditiven 'Verschnaufpause' verbunden ist¹⁶⁷. Das Phänomen der besonderen Wahrnehmung von Soundchip-Musik im audiovisuellen Verbund der Computerspiele, in welcher die Musik weniger, wie bspw. Filmmusik, bestimmte Affekte unterstreicht, sondern im gewissen Sinne in Konkurrenz zu den visuellen Informationen steht, gestaltet sich daher als interessanter Anknüpfungspunkt für eine auf die Rezeption von Computerspielmusik fokussierte Forschung.

¹⁶⁷ Die Bonusrunden sind jedoch relativ selten und der Autor hat trotz häufigem und ausgiebigem Spiel von BUBBLE BOBBLE erst im Rahmen dieser Arbeit bemerkt, dass alle drei Bonusrunden mit der identischen Musik unterlegt sind.

3.2 Strategien der Gestaltung im Spiegel der technischen Limitierungen

Nach der erfolgten Unterscheidung von Computerspielmusik entsprechend ihrer funktionalen Anwendungen im Spiel wird nun auf die Rolle von Komposition und Programmierung unter Berücksichtigung der Eigenheiten von Soundchips eingegangen (vgl. Kap. 2.7). Die verschiedenen technischen Limitierungen werden erläutert und die aus ihnen folgenden Gestaltungsstrategien erarbeitet.

Die den Chips innewohnenden Limitierungen wurden im Rahmen der Vorstellung der technischen Voraussetzungen thematisiert und umfassen Beschränkungen bei

- *Anzahl, Tonumfang und Stimmung der Klangkanäle,*
- *den verfügbaren Klangfarben und*
- *Möglichkeiten ihrer klanglichen Ausgestaltung.*

Aus der Hardwareumgebung des jeweiligen Soundchips wirken außerdem Beschränkungen aus den Bereichen

- *des verfügbaren Speicherplatz,*
- *der Rechenleistung sowie*
- *des für Computerspiele wie Demos wichtigen Timings*

auf die Gestaltung ein.

Die Komposition von Musik für Soundchips ist in den frühen Jahren gleichzusetzen mit deren *Programmierung*. Maschinennah, also meist in Assembler programmierte Instruktionen werden an die Soundchips geschickt. Hierfür werden von Komponisten / Programmierern der ersten Generation i.d.R. selbst entwickelte Software-Routinen benutzt. Diese entsprechen folglich soweit wie möglich den eigenen Anforderungen und werden stetig weiterentwickelt. Ihre Entwicklung findet selbstredend unter musikalischen Gesichtspunkten statt, wenn auch das Selbstverständnis der interviewten Komponisten auch dem eines Programmierers entspricht¹⁶⁸. So sagt Chris Hülsbeck:

„Also auf jeden Fall [habe ich] kein Notenpapier für die Komposition benutzt, da ich keine Noten lesen kann. [...] Damals war das alles autodidaktisch und die Art und Weise wie ich das komponiert habe war [...] mit sogenannten Hex-Zahlen [...]. [I]ch war damals auch mehr Programmierer als Musiker“ (Hülsbeck 2005a).

Eine nach kreativen Maßstäben stattfindende, maschinennahe Programmierung ist nicht nur für Soundchip-Musik, sondern auch für Computerspiele allgemein und im besonderen Maße die Demo-Szene¹⁶⁹ Grundlage der Code-Produktion. Statt des sperrigen Begriffs der „kreativen Programmierung“ soll an dieser Stelle auf

¹⁶⁸ In den geführten Interview ist deutlich geworden, dass alle drei in erster Linie programmiert haben und viel Zeit darauf verwendet wurde, konkret auftauchende technische Probleme zu lösen. Die Komposition nimmt somit für die Produktion von Computerspielmusik in den 1980er Jahren mutmaßlich die geringste Zeit in Anspruch (vgl. Pummell (2005), Warhol (2005), Hülsbeck (2005a)).

¹⁶⁹ Vgl. die Fußnoten 69 und 98 bzw. Kapitel 2.5.3 und 4.4.

den von Levy geprägten Begriff des „Hacking“ zurückgegriffen werden, welches von einem Ausprobieren des Möglichen auf einer gegebenen digitalen Maschine geprägt ist. Neben der grundsätzlich auf Effizienz ausgelegten Programmierung sind es die sog. Tweaks, also programmiertechnische Tricks¹⁷⁰, die über die Spezifikationen der Chips hinausgehen, die als Hacking verstanden werden. Die programmiertechnische Praxis, Strategien bereitzustellen um im Angesicht der technischen Limitierungen der Soundchips die jeweilige musikalische Idee umzusetzen, ist besonders aufgrund der *low level*-Programmierung als Hacking zu sehen.

Der Begriff der „Hacker Ethik“ (vgl. Levy 1994: 39ff) geht auf die frühen Experimente von MIT-Studenten¹⁷¹ mit dem TX-0 und PDP-1 Computer zurück. Levy spricht im Anschluss an diese erste und einer zweiten Generation von Hackern¹⁷², von einer dritten Generation, die auf Heimcomputern gelernt hat zu programmieren und keine Probleme mehr mit der Ökonomisierung der u.a. auf freien Informationsfluss basierenden Ethik hat¹⁷³. Andere Elemente dieser lassen sich weiterhin in dem Handeln der hier im Fokus stehenden Komponisten / Programmierer finden. Hacker sind grundsätzlich von der Realisierung des bestmöglichen Ergebnisses im Rahmen der jeweiligen Hardwareumgebung besessen. Die kulturelle Praxis des Hacking von den frühen MIT-Hackern bis zu den Spieleprogrammierern der 1980er Jahre oder den 'Kids' der Demo-Szene ist also als Handeln zu verstehen, das grundsätzlich von technischen Limitierungen in seine Schranken gewiesen wird. In diesen sehen Hacker jedoch in erster Linie eine Herausforderung, die weiteren Ansporn für die mitunter beeindruckenden Ergebnisse ihrer Tätigkeit liefert. Bereits Anfang der 1960er Jahre wurde am MIT beispielsweise ein erster, nicht-akademischer Musik-Compiler für den TX-0-Computer geschrieben¹⁷⁴, welcher dem MIT Anfang der 1960er Jahre zur Verfügung gestellt wurde (Levy 1994: 27f).

Im Folgenden werden in den Kapitelüberschriften kritische Limitierungen für Soundchip-Musik genannt um danach auf die in ihrem Spiegel angewendeten Gestaltungsstrategien zu erläutern.

¹⁷⁰ Darunter fällt z.B. die Technik, über die schnelle Modulation der Lautstärke-Register (zwischen Maximal- und Minimalwert) der frühen PSGs oder des PC-Speakers gesampelte Audio-Daten wiederzugeben.

¹⁷¹ MIT = Massachusetts Institute of Technology.

¹⁷² Die erste Generation der Hacker lernten ihr Handwerk auf Großrechnern in den 1960er Jahren am MIT, die zweite Generation war maßgeblich an der Entwicklung der ersten Heimcomputer beteiligt. Levy überschreibt diese mit „True Hackers“ und „Hardware Hackers“ (Levy 1994: 15ff, 153ff)

¹⁷³ Levy schreibt über diese dritte Generation von „Game Hackers“, die Anfang der 1980er Jahre in der Spieleprogrammierung den Erfolg suchten: „*They were the [...] third Generation of hackers who had learned their programming artistry on small computers, who had never bootstrapped themselves up by way of a community. Who dreamed not only of the ultimate hack, but of fame, and big royalty checks.*“ (Levy 1994: 284)

¹⁷⁴ Das von Peter Samson entwickelte Programm, um auf dem Konsolenlautsprecher des frühen Transistor-Computers ein Bach-Stück wiedergeben zu können (Levy 1994: 33ff) kann als erste, nicht aus dem Bereich der akademischen Computermusik stammende Musik-Routine gewertet werden und stellt somit einen frühen Vorläufer der Soundchip-Musik dar. Für den eine dreistimmige Tonwiedergabe bietenden PDP-1 wurden die Routinen von Samson überarbeitet, der das Programm dem Hersteller DEC kostenlos zur Verfügung stellt (vgl. Levy 1994: 56). Ein anderes Programm für den TX-0 Computer, HAX errechnete Zufallsgrafiken und -sounds auf Basis zweier Register (vgl. Graetz 2001: 44f) (vgl. Fußnote 157).

3.2.1 Beschränkte Polyphonie

Alle Soundchips vor den reichhaltiger ausgestatteten Plattformen ab Mitte der 1980er Jahre, die erstmals sechs Stimmen (PC-ENGINE) und mehr bieten, beschränken sich auf eine drei- bzw. vier-stimmige Klangwiedergabe. Eine Ausnahme stellt das ATARI VCS mit seinen zwei Stimmen dar. Der AY-Chip besitzt als einziger PSG-Soundchip Einzelausgänge für die vorhandenen Kanäle¹⁷⁵. Die anderen PSGs (bis zur Einführung der Stereo-Chips von AMIGA und PC-ENGINE) haben gemeinsam, dass sie nur Mono-Signale ausgeben, in welchen die Anteile der Kanäle miteinander gemischt sind. Die Beschränkung der Mehrstimmigkeit kann wie die anderen Beschränkungen auch als Ursache für bestimmte Eigenheiten in der ästhetischen Gestaltung von Musik für Computerspielmusik gesehen werden. So spielt die Beschränkung eine große Rolle als Herausforderung an den Komponisten, wie Michael Pummell bemerkt:

„It's [...] like writing for a string quartet. Imagine having only four instruments [...] [a]nd you're supposed to keep it interesting. That's the challenge“ (zit. n. Belinkie 1999: o.S.).

Chris Hülsbeck antwortet pragmatischer auf die Frage nach der beschränkten Polyphonie und den Folgen für sein kompositorisches Vorgehen:

„Die Art, wie ich mir damals Musik in meinem Kopf vorgestellt habe, war schon prädestiniert für den SID-Chip. Ich hätte mich natürlich gefreut, wenn ich mehr Stimmen gehabt hätte aber im Prinzip war meine Denkweise damals 'Bass, Melodie und Drums'. Schon dreistimmig sozusagen. Und das hat mir für lange Zeit gereicht“ (Hülsbeck 2005a).

Computerspielmusik für PSGs basiert i.d.R. auf diesen drei Haupt-Stimmen: Neben einer Schlagzeug- und der Melodiespur steht noch eine begleitende (Bass-) Spur zur Verfügung. Häufig wird ein Kanal des Chips dabei für Soundeffekte reserviert. Ansonsten wird die Technik angewandt, eine Melodiestimme temporär stummzuschalten, um auf dieser Soundeffekte abzuspielen, wie es z.B. in SUPER MARIO BROS. der Fall ist.

Auf Arcade-Boards werden schon früh mehrere Soundchips verbaut, um Soundeffekte und Musik gleichzeitig wiedergeben zu können, wie in den bereits erwähnten Automaten STAR WARS (ATARI, 1983; 4 x Pokey), BOMB JACK (TEKHAN, 1984; 3 x AY-38910) oder MR. DO'S CASTLE (UNIVERSAL, 1983; 3 x SN76496AN).

Bei Computerspielsystemen mit nur einem integrierten Soundchip werden schon bald schnelle Schaltungen zwischen verschiedenen Klangeinstellungen vorgenommen, um die limitierte Polyphonie zu umgehen: Nimmt der Hörer mehr Instrumente wahr, als die verfügbaren Kanäle eigentlich bieten, ist dies auf eben diese Technik zurückzuführen, durch die in Pausen des einen musikalischen Elements auf dem gleichen Kanal ein anderes erklingt. Dieses grenzt sich dann durch Tonhöhe, Klangfarbe oder einen anderen musikalischen Sinnzusammenhang (als Teil eines anderen musikalischen Motivs) von dem ersten ab.

¹⁷⁵ Einzelne Pinouts der SSG-Tonkanäle bleiben auch bei den Chips YM 2149, 2203 und 2610 erhalten.

Die Technik, durch Änderung von Klangparametern mehrere „Instrumente“ auf einem technischen Kanal wiederzugeben, wird zum Beispiel verwendet, um gesamte Schlagzeugspuren (Bassdrum, Snaredrum, Hi-Hat) mit Bassspur auf einem Kanal wiederzugeben, wie Hülsbeck im Interview erläutert:

„Später haben wir innerhalb [...] eines Klangs die Wellenform geändert, also bestimmte Sachen aus- und eingeschaltet. Du konntest zum Beispiel ein Frame lang ein Rauschen anschalten und danach die Rechteckwelle, das hat dann einen perkussiveren Anschlag gegeben, als würde noch eine Drum mitspielen. Später war dass dann praktisch eine Sound Makro-Programmierung [...]: In dem Augenblick, wo du eine Note anspielst, wird ein kleines Programm abgearbeitet, mit einer ganz simplen Befehlsstruktur. Dadurch konnte man die gesamten Klangparameter des SID-Chips beeinflussen. [...] [Eine Technik die dadurch möglich wurde, basiert auf drei Bass-Sounds]: Einer war so programmiert, dass er in der ersten halben Sekunde eine Art Bass-Drum spielt und danach dann sofort der Basston kommt. Das ging so schnell und war so effizient, dass man nicht mehr unterscheiden konnte zwischen dem normalen Bass-Sound und dem mit der Drum davor. Die sind auf eine Art verschmolzen, weil das Gehör nicht schnell genug ist bzw. die Drum den Bass an der Stelle sowieso überdeckt hätte. Und genau so konnte man das mit einer kurzen Snare Drum machen [, wie auch mit der Hi-Hat]. Auf diese Art konnte man Drum- und Bass-Spur auf einer Stimme realisieren.“ (Hülsbeck 2005a).

Ein anderer kompositorischer Trick zur Umgehung der Limitierung der verfügbaren Polyphonie stellt das Verwenden schneller Arpeggien dar, die durch die Wahl von sehr kurzen Tondauern als Akkorde wahrgenommen werden. So benutzt Hülsbecks Hintergrundmusik von THE GREAT GIANA SISTERS (RAINBOW ARTS, 1987; ♪049), neben der eben beschriebenen Technik der Drum- und Bass-Spur auf einem Kanal auch eine durch schnelle Modulation des Lautstärke-Registers erzeugte Melodie-Stimme in dem eigentlich zwei-kanaligen Stück¹⁷⁶. Darüber hinaus finden die erwähnte quasi-Akkorde für die Reggae-Nachschläge des Themas Verwendung. Er selbst bemerkt zu dieser Technik:

„[...] Synthi-Akkorde habe ich mit diesen Arpeggien gemacht, die waren nicht unbedingt meine Erfindung. Ich habe damals ein Stück gehört von Martin Galway [Ocean Loader 1] und der hatte eine langsamere Version von diesen Arpeggien benutzt. Ich hatte mir vorher schon mal überlegt, ob man so was machen könnte, hab es dann aber wieder verworfen, weil ich dachte, das klingt bestimmt komisch, wenn man die Tonhöhe derart schnell abwechselt. Doch wie ich das dann in dem Stück von Galway gehört habe, dachte ich: 'jetzt musst du das auch mal probieren'“ (ebd.).

¹⁷⁶ Dieser Tweak, der zur Wiedergabe von 4 Bit-Samples auf dem C64 angewendet wird, beeinflusst die Wiedergabe der Klangkanäle des SID-Chips nicht, die trotz der Modulation des Lautstärke-Registers weiterhin deutlich wahrnehmbar bleiben (Hülsbeck 2005a).

Neben den offensichtlichen Beschränkungen der Polyphonie bei auf frühen Soundchips basierender Musik ist die Komposition von Musik für die in ihrer Polyphonie unterschiedlichen Soundlösungen für PC (von PC-Speaker bis ROLAND LAPC-1) eine äußerst vielschichtige Angelegenheit. Wie Sanger konstatiert, war das Abdecken der verschiedenen Soundplattformen mit einem erheblichen Arbeitsaufwand verbunden:

„A composer who wanted to be at the cutting edge of PC audio would have to create a different file for each piece of music times the number of sound cards supported. Since most people did not know about, much less have, what we now might call 'WAV playback capability', sound effects, too, had to be created multiple ways for the various possible playback devices that might be out there“ (Sanger 2003: 177, Hervorh. i. O.).

Bezüglich des hierbei anfallenden Arbeits- und Organisationsaufwand merkt Sanger für das von ihm vertonte Spiel 688 ATTACK SUB (ELECTRONIC ARTS, 1989) an:

„[...]o deliver 24 pieces of music, we had to deliver 216 different files, via mail or phone-to-phone 300-baud modem. Despite our best efforts, I must confess that more energy was expended on organizing and converting files for this game than was spent composing“ (Sanger 2003: 178).

Ein Hörvergleich der Adaptionen der Titelmelodie des Grafik-Adventures THE SECRET OF MONKEY ISLAND (LUCASFILM GAMES, 1990) illustriert die Unterschiede der verschiedenen Soundkarten auf dem PC im Rahmen des Kapitels 3.4. Adaptionen einer Spielmusik auf verschiedene Plattformen wurden auch an selbstständige Programmierer abgegeben, wie z.B. David Warhol. Dieser äußert sich zu dem frustrierenden Aspekt seiner Arbeit, die Musik für das LUCASFILM-Adventure ZAK MCKRACKEN (LUCASFILM GAMES, 1988) für den einstimmigen PC-Speaker umzusetzen, im Interview:

„Doing those projects was about the same time as the Ad Lib card was coming out, or Sound Blaster. And even the MT-32, the Roland synthesizer. So you'd be working very very hard to make this tiny tiny, limited technology sound good, and than the guy who was doing the Ad Lib version would just pretty much do his beautifully crafted arrangements with six part things with fancy, rich soundwaves. So I kind of lost interest in doing that“ (Warhol 2005).

Für weitere Untersuchungen erscheint die Frage interessant, nach welchen Kriterien aus einem viel-stimmig arrangierten Musikstück für LAPC-1 eine einstimmige PC-Speaker-Version gefertigt wurde. Bei den frühen für PCJR optimierten SIERRA-Adventures (vgl. Kapitel 2.6.2) wurden beispielsweise einfach die Melodiestimmen behalten. Diese Technik führte jedoch zu musikalisch nur wenig überzeugenden Ergebnissen.

Die limitierte Polyphonie hat zwei für Soundchip-Musik genrebildende Strategien der Gestaltung hervorgebracht, die auch auf ihre „Sound Culture“ Einfluss nehmen: Die schnellen Arpeggien sowie die erwähnte Technik, die Musikroutinen zwischen verschiedenen Klangeinstellungen der Soundchips springen zu lassen,

um auf einem technischen Kanal mehrere Stimmen des Arrangements in den jeweiligen Pausen abspielen zu lassen. Diese Strategien werden aufgrund ihrer Klangästhetik auch in der populären Soundchip-Musik (Kapitel 4) häufig aufgegriffen.

3.2.2 Limitierungen der Klangfarbe

Bei der Vorstellung der verschiedenen Soundchips sind die Limitierungen der Klangerzeugung bereits offensichtlich geworden. Diese ist bei den frühen PSGs auf Rauschen und Rechteckwellen ohne weitere Manipulationsmöglichkeit beschränkt. Die zur Verfügung stehenden Klangfarben sind, mit Ausnahme des SID, bis zur Einführung der FM-Synthese und des Sampling in Arcade-Automaten und dem COMMODORE AMIGA Mitte der 1980er Jahre äußerst monoton.

So bietet das ATARI VCS zwar 16 verschiedene Klangeinstellungen, die jedoch wenig Variation bieten, der Pokey Soundchip ist ebenso wie der SN76489 für seinen „trockenen“, wenig durchsetzungsfähigen Klang vor allem in den Basslagen bekannt. Die verglichen mit dem AY-Chip minderwertige Klanglichkeit der letzteren ist auch darauf zurückzuführen, dass die zur Verfügung stehenden Rechteckwellen keiner weiteren Manipulation unterworfen werden können, wie es der AY-Chip durch die mögliche Mischung von Rechteckwelle und Rauschanteil sowie der vorhandenen Möglichkeit einer rudimentären Gestaltung des Klangverlaufs bietet.

Der MOS SID und die Klangerzeugung des NINTENDO ENTERTAINMENT SYSTEMS markieren einerseits den nächsten historischen Schritt, andererseits zwei Pole der Klanglichkeit von Soundchips: Der SID klingt relativ hart und scharf, das NES bietet weitaus rundere, weichere Klangfarben mit analogerem Charakter. Neben dem Sweep-Effekt, den vier einstellbaren Pulsbreiten und der Option, die Hüllkurve zu wiederholen, bietet es keine weiteren Möglichkeiten der Klangbearbeitung. Der SID dagegen ist ein einzigartiger Soundchip, der gegenüber den anderen durch das regelbare Filter, verschiedene wählbare Wellenformen pro Kanal, der Implementierung von ADSR-Hüllkurven sowie der Möglichkeit von Ringmodulation und Synchronisierung der Oszillatoren untereinander eine Vielzahl von Optionen der Klangmanipulation bietet, die weit über die Fähigkeiten der anderen Soundchips hinaus gehen. Vor allem die Möglichkeit der Pulse Width Modulation wurde häufig und gerne benutzt, diese Klangfarbe stellt eine Spezialität des SID-Chips dar, wie Hülsbeck erwähnt:

„[...] Auf dem C64 [...] konnte man mit einem selbst programmierten Software-LFO die Pulsbreite modulieren. Das führte zu einem ziemlich fetten Sound aus einem Oszillator. Der klingt dann wie zwei oder drei gegeneinander verstimmt Sägezahn-Oszillatoren. [...] Das war eigentlich der Haupt-Klang, den alle C64-Musiker benutzt haben und auch eine der ersten Sachen, die ich rausgetüftelt hatte mit meiner Soundroutine.“ (Hülsbeck 2005a).

Die frühen, über die beschriebene Modulation der Lautstärke-Register erreichten Tweaks, um Samples wiedergeben zu können¹⁷⁷ stellen erste Ansätze dar, die Beschränkungen der Klangfarbe zu überwinden. Dieses Verfahren des 4 Bit-Sample-Playbacks besitzt eine eigene Klangcharakteristik, die auf der nach heutigen Maßstäben mangelhaften Auflösung und Quantisierung basiert.

Sowohl FM-Synthese als auch die zeitlich parallel stattfindende Integration des Samplings in Computerspiel-systemen (vor allem in SNES & AMIGA) sorgen für die Aufweichung der genuinen Klanglichkeit von Soundchip-Musik, da beide Verfahren die Verwendung einer breiten Palette von verschiedenen Klangfarben ermöglichen. Die Integration von FM-Chips ab Mitte der 1980er sowohl in Arcade-Automaten, als auch in Videokonsolen wie dem SEGA MEGA DRIVE und den PC-Soundkarten bietet eine neue, eigene Klanglichkeit, die sich vom PSG-Sound abhebt und erstmals Klänge erzeugt, bei denen eine gewisse Instrumentengruppe als Referenz dienen kann (wie z.B. Streicher, Bläser, Piano etc.). Die Einführung des Samplings auf AMIGA und SNES läutet die Befreiung der Computerspielmusik von den vorher streng limitierten Klangfarben ein.

Die genuinen Klangfarben der frühen PSGs¹⁷⁸ und die aus ihnen entstehende Klangästhetik werden somit als klangliche Referenzen im Sinne einer „Sound Culture“¹⁷⁹ der Soundchip- oder Computerspielmusik wahrgenommen. Mit Aufkommen der auf Sampling und FM-Synthese basierenden Plattformen bleiben die gestalterischen Paradigmen von Computerspielmusik (kurze, wiederholte, melodiebetonte Stücke mit Ohrwurmqualität sowie deren Aufteilung in formale Sinnzusammenhänge, den verschiedenen *game states* entsprechend) zwar zunächst erhalten, dieses Phänomen kann jedoch auch im Spiegel der vorerst gleichbleibenden gestalterischen Paradigmen der jeweiligen Spielkonzepte erklärt werden. In der 16 Bit-Ära bleiben die Spielkonzepte grundsätzlich dieselben, auch auf den FM-Soundchips erklingen daher weiterhin kurze, sich wiederholende Themen. Ein Beispiel für eine solche Adaption ist die auf FM-Synthese basierende Hintergrundmusik des Plattformspiels BUBBLE BOBBLE (YM 2203 & YM 3526; ♪042) sowie dessen Nachfolger RAINBOW ISLANDS (YM 2151; ♪050). Ebenfalls interessant im Hinblick auf die fortschreitende Entwicklung der Klangerzeugung ist das Aufgreifen von Melodien der Vorgänger in populären Spielserien. Ein Beispiel stellen die Konvertierungen des SUPER MARIO BROS.-Themas auf das SNES (SUPER MARIO ALLSTARS (NINTENDO, 1993); ♪051) sowie auf GAMECUBE (als Hintergrundmusik der Plattform-Level in SUPER MARIO SUNSHINE (NINTENDO, 2002; ♪052) dar. In beiden wird die Melodie des NES-Originals neu instrumentiert und interpretiert.

¹⁷⁷ Diese Technik wurde bei allen PSGs angewendet. Allein für SID- und AY-Chip sind jedoch Beispiele bekannt, bei der während der Wiedergabe von Samples ebenso die Klangkanäle klingen.

¹⁷⁸ Die Wellenformen der PSGs klingen verglichen mit denen von analogen Synthesizern deutlich „digital“.

¹⁷⁹ Der Begriff „Sound Culture“ bezeichnet an dieser Stelle die Identifikation eines gesamten Musikgenres durch eine bestimmte Klanglichkeit, die jedoch nicht allein auf Klangfarben beschränkt bleiben muss, sondern auch von formalen oder strukturellen Merkmalen innerhalb der Musik ausgehen kann. Wichtig hierbei ist die Bedeutung des Klangs als formal-ästhetisches Charakteristikum (vgl. Büsser 1996: 6f). Vgl. auch Kapitel 4.1.

3.2.3 Beschränkter Speicherplatz

Die Entwicklung von Soundchips hängt unmittelbar mit der teuren und daher nicht verfügbaren Bandbreite für andere Verfahren digitaler Klangerzeugung bzw. Musikwiedergabe wie Sampling im entsprechenden Entwicklungszeitraum zusammen. Soundchips stellen einen Kompromiss aus technischer Machbarkeit und klanglicher Qualität dar, um Musik und Klang in digitalen Umgebungen kostengünstig produzieren zu können. Die Software für Arcade-Automaten wie auch Spielkonsolen wurde von Anfang an in maskenprogrammierten ROM-Modulen gespeichert, deren Speicherchips den dominanten Kostenfaktor bei der Vervielfältigung darstellen. Für die vorliegende Arbeit trifft diese Feststellung uneingeschränkt zu, da Konsolen mit optischen Speichermedien nicht in den Zeitraum der Untersuchung fallen bzw. optional erhältliche CD-ROM-Laufwerke (für PC-ENGINE und MEGA DRIVE) keine nennenswerte Rolle für die Entwicklung von *Soundchip*-Musik spielen. Für Heim- und Personalcomputer ist währenddessen – von frühen Vertretern wie dem ATARI 800 und COMMODORE C64 abgesehen, die optional auch Module lesen können – die magnetische Speicherung auf Kassetten¹⁸⁰ oder Disketten von Anfang an Standard.

Elektronische Festspeicher-Medien wie auch magnetische Speicher-Medien haben unterschiedliche Vor- und Nachteile. So sind erstere vor allem ein sicheres Medium. Durch die statische Speicherung („solid state“) sind sie gegenüber Staub und Wärme recht unempfindlich und lange haltbar. Außerdem ist die Geschwindigkeit mit über 5 MByte/s Datenübertragungsrate und Zugriffszeiten von unter einer Millisekunde sehr hoch (vgl. Forster 2005: 5ff). Ein Nachteil der Festspeicherung ist neben dem hohen Herstellungspreis vor allem die nicht vorhandene Wiederbeschreibbarkeit. Magnetische Medien hingegen lassen sich mehrmals beschreiben und sind in der Herstellung vergleichsweise günstig. Aufgrund der mechanischen Abtastung im Laufwerk erweisen sie sich jedoch als störanfällig, außerdem reagieren sie empfindlich auf Wärme, Staub und magnetische Einflüsse. Die Geschwindigkeit ist bei einer Datenübertragungsrate von weit unter einem kB/s für die Audio-Kassette und der von Medium und Controller abhängigen maximalen Übertragungsrate von wenigen kB/s für Floppydisks weitaus niedriger als bei Festspeichern. Aus der Datenübertragungsgeschwindigkeit wurde bereits eine Form von Spielmusik abgeleitet: die der Loader-Musik. Bei Heimcomputern war die Zeit, während der ein Spiel lud, lang genug, um sie mit Musik zu unterlegen, weswegen diese Kategorie überhaupt entstehen konnte.

Weiterhin ist der verfügbare Arbeitsspeicher (RAM) für Computerspielmusik von herausragender Bedeutung, da der Hauptprozessor hieraus die zu verarbeitenden Daten lesen muss. Die vergleichsweise hohen Preise für Speicherbausteine sind der Grund dafür, dass in Spielplattformen bis in die 1990er Jahre hinein der Hauptspeicher großen Beschränkungen unterliegt, wie die in Kapitel 2 vorhandenen Informationskästen zu den jeweiligen Systemen illustrieren.

Der für die Musikinstruktionen für Soundchips reservierte Speicherplatz in frühen Computerspielsystemen liegt laut Collins (2004a: 3f), die INTELLIVISION und NES als Beispiele nennt, bei ungefähr 10 % des Datenträger-Speicherplatzes. Leonard (2001: o.S.) erwähnt für das NES einen Speicherplatz von bis zu 40 kB, bei

¹⁸⁰ Die Daten werden dafür in analoge Audio-Signale umgewandelt und auf Kassette aufgezeichnet.

einer maximalen Modulgröße von 512 kB. In der High Voltage SID Collection¹⁸¹, die einen Großteil der für den COMMODORE C64 - Soundchip programmierten Musik enthält, finden sich nur wenige von Spielen extrahierte SID-Dateien, die größer als 25 kB sind. Diese enthalten häufig gesampelte Klänge wie z.B. die SID-Dateien die der Spiele GHOSTBUSTERS (ACTIVISION, 1984; ♪053), LEMMINGS (PSYGNOSIS, 1993) und GHOULS 'N' GHOSTS (CAPCOM, 1989; ♪053)¹⁸². Im Vergleich dazu ist das bereits erwähnte, mit 16 Minuten äußerst lange KNUCKLE BUSTERS (♪023) als SID-Datei nur 7 kB groß. Vosburgh sagt im Hinblick auf die Entwicklung von Computerspielmusik für den COMMODORE C64:

„[P]laying the second fiddle to the graphics means you get allocated only a tiny share of the computer's memory, typically between 4K and 6K. Not only do you have to fit your tune into that space, you also have to find room to define all your sounds – not just for the music, but also for the game-activated sound effects“ (vgl. Vosburgh o.J.).

Chris Hülsbeck erläutert im Interview detailliert, welche Speicherbereiche des C64 er mit der Zeit nutzen konnte und deutet darüber hinaus an, welche anderen maschinennahen Limitierungen auf die Art der Speicherverwaltung und der Zwischenspeicherung von Programm und Daten Einfluss genommen haben:

„Zu dem Zeitpunkt, als ich angefangen habe [1984-1985], haben noch recht wenig Spiele alle Datenbereiche des C64 ausgenutzt. Die meisten Spiele haben erstmal nur in den RAM-Bereichen gearbeitet, die nicht überlappt waren von irgendwelchen ROM-Bereichen. Nach diesem ganzen Zero Page- und Bildschirmspeicher gab es einen Bereich, der normalerweise offen war, wo man sein BASIC-Programm abgelegt hätte. Der Bereich war relativ groß und die meisten Spiele haben den belegt für ihren Programmcode. Dann gab es einen Speicherbereich, der war nur 4 kB groß, gerade über dem BASIC-ROM. Am Anfang habe ich natürlich erstmal diesen 4K-Bereich ausgenutzt, weil der außerhalb vom BASIC-Speicher war. Das heißt, man konnte den direkt beschreiben und lesen, ohne dass man irgendwelche ROM-Bereiche ausblenden musste. [...]

Die 64 K Speicherstellen [der Hauptspeicher des C64] stellten theoretisch RAM dar. Der C64 hat aber auch ein BASIC-ROM und ein Betriebssystem, die alle irgendwo laufen müssen in diesem 16 Bit Adressbereich. Mehrere dieser Speicherbereiche wurden von einem ROM überlagert. [...] Wenn man von dem Maschinenprogramm aus das entsprechende Bit ausschaltet, dann sieht der Prozessor das RAM, und wenn man das Bit einschaltet, dann sieht er das ROM. [...] Wenn meine [spätere] Musikroutine aufgerufen wurde, dann hat die als erstes dieses Bit umgeschlagen, damit man auf das RAM unter dem BASIC-ROM zugreifen konnte [da bei kommerziellen C64-Spielen i.d.R. kein BASIC verwendet wurde, konnte dieser Platz genutzt werden, N.D.], und da habe ich dann meine Musikdaten abgelegt“ (Hülsbeck 2005a).

¹⁸¹ <http://www.hvsc.c64.org/> (26.07.2005). Die „High Voltage SID Collection“ umfasst zum Zeitpunkt dieser Arbeit (Sommer 2005) über 17000 verschiedene SID-Dateien.

¹⁸² Komponisten / Programmierer der genannten Titel sind laut File-Tags der Reihenfolge nach Adam Bellin, Jeroen Tel und Tim Follin.

Auch angesichts des beschränkten zur Verfügung stehenden Speicherplatzes für Musik-Daten sind die einzelnen Elemente der Musikroutinen darauf zugeschnitten, möglichst sparsam mit den Ressourcen umzugehen. Der bereits ab Anfang der 1980er Jahre als Programmierer bei MATTEL ELECTRONICS beschäftigte David Warhol entwickelt angesichts des begrenzten Speicherplatzes möglichst kleine Instruktionen für seine Programme zur Steuerung des AY-Soundchips. Um in den maximal 16 kB verfügbaren Modul-Speicherplatz des INTELLIVISION neben dem Spiel überhaupt Musik unterbringen zu können, optimiert er die Musikdaten, indem er sie in Subroutinen ihrer größten Redundanzen befreit:

“I would write the music out on manuscript paper just to get a song, completely doing the arrangement on paper, then typing it in note by note, using assembly language macros. Those macros would [occupy] typically one byte per note and [...] I would even have music subroutines, where, if I knew I was going to use a certain bassline repeatedly 8 times a phrase, I'd have a command that would take it [in] a subroutine just to play that one bar [8 times]. So the pieces were turned in a jigsaw puzzle. Each line of, each measure or any size of musical unit could be broken into a musical subroutine and ultimately the song was just saying 'oh, call this subroutine on this channel and this subroutine on this channel'. [This techniques were developed] to be very conservative with the amount of space that we had“ (Warhol 2005).

Beschränkungen im Speicherplatz haben auf die Programmierung der Instruktionen und somit in letzter Konsequenz auch auf die formalen Aspekte der Chipmusik einen großen Einfluss. So bedingen sie die häufigen Wiederholungen von Themen und die insgesamt kurzen Loops, die zur Wiederholung stehen. Der Programmierer und Komponist Michael Pummell (2005) berichtet von Restriktionen der Publisher und Main Programmer in späteren Jahren, die in dem Fall von MISSION: IMPOSSIBLE (ACCLAIM, 1998) auf dem NINTENDO 64 dazu führten, dass mit 125 kB weniger als 2% der Modulkapazität von 8 MB für den Kernel der gesamten Musik des Spiels zur Verfügung stand, auch wenn dieses für ihn die „worst case situation“ darstellt (Pummell 2005).

3.2.4 CPU-Last und weitere Beschränkungen „von außen“

Als Element eines Computerprogramms, das vor allem auf die Timing-genaue Darstellung von grafischen Elementen und der Abfrage der Eingaben des Spielers basiert, ist Computerspielmusik weiteren wichtigen Beschränkungen unterlegen, die sich auch auf den Kompositions- und Programmierungsprozess auswirken. Der Musik wurde vor allem in der Anfangszeit innerhalb des Prozesses der Spieleentwicklung nur wenig Aufmerksamkeit zuteil. In den frühen Jahren war die Implementierung einer funktionierenden Musikroutine für die am Spiel beteiligten Programmierer eine Art Beiwerk, ein für den Markterfolg notwendigerweise zu integrierender Code. Daraus folgt, dass anstelle ästhetischer Kriterien vor allem technische für die Implementierung von Musik wichtig waren. Ihr Code musste den Rest des Spiels unbeeinflusst lassen, ansonsten waren die Komponisten relativ frei von ästhetischen Ansprüchen¹⁸³. Ein Indiz für die damalige Haltung gegenüber Musik ist die häufige Verwendung von Adaptionen bekannten Liedguts in den frühen Spielen der 1980er Jahre wie zum Beispiel in den Automaten FROGGER (KONAMI, 1981; ♪022) oder CARNIVAL (SEGA, 1980; ♪021)¹⁸⁴. Pummell bemerkt im Interview:

„[I]n all honesty, they [die Entwickler, N.D.] were happy to have music that didn't crash. Or didn't cause the system to slow down. [If it fits] in a few methods to initial requirements, then your music is wonderful because it works and it's not embitting the other things [...]. That's the first criteria“ (Pummell 2005).

Dass den Audio-Entwicklern neben dem geringen Speicherplatz auch nur ein bestimmtes Maß an CPU-Last zugesprochen wird („5% to audio, 95 % is graphics“), ist für Pummell (ebd.) zwar bedauerlich, aber bei einem primär visuellen Medium wie den Computerspielen verständlich.

Auch die für Computerspielsysteme überaus wichtige Zeilenwiederholffrequenz hat großen Einfluss auf die Musik für Soundchips. In den USA wie in Japan (NTSC-Norm) werden 60 Bilder pro Sekunde errechnet, in Europa (PAL-Norm) jedoch nur 50. Laut Warhol ist daher z.B. eine Tempo-Änderung relativ schwierig zu programmieren, da die Zeilenwiederholffrequenz von 50 bzw. 60 Hz den Rahmen für mögliche Änderungen des Tempos diktiert:

„The games were all based on 60 Frames a second, I guess in Europe this would be 50 ? So there wasn't a lot of resolution that allowed you to speed up a song or slow it down. Well, now I can probably think of some techniques where you'd use counters that were modified by the 60 Frames a second clock that you might be able to subtly speed something up. But the timing

¹⁸³ Ein gutes Beispiel bietet das 16-minütige Stück zu KNUCKLE BUSTERS (♪023) von Rob Hubbard, das mit dem Spielgeschehen in keiner Weise adaptiv verbunden ist.

¹⁸⁴ Das Spiel FROGGER verwendet in der Hintergrundmusik unter anderem „Yankee Doodle“ und „Camp Town Races“, die Hintergrundmusik von CARNIVAL basiert auf dem populären Lied „The loveliest Night of the Year“. Viele andere Beispiele existieren, wie die Verwendung von „Louie Louie“ (♪055) in dem Spiel CALIFORNIA GAMES (EPYX, 1987), von Rimsky-Korsakows „Hummelflug“ in BUZZ BOMBER (MATTEL, 1983) oder des Beatles-Songs „Lady Madonna“ in BOMB JACK (♪007b). Das Spiel PARODIUS verwendet viele berühmte Themen der klassischen Musik (vgl. Kapitel 3.4)

between 8th notes and 16th notes wouldn't be consistent just based on the fact that everything ultimately was timed to a 60th in a second. If you sped something up by 108 % or 106 % I think you might notice that some of the faster runs might not actually be on [time]“ (Warhol 2005).

Die Abhängigkeit der Musik vom Raster der Zeilenwiederholffrequenz ist aufgrund der entsprechenden Interrupts zu verstehen, da in Computerspielen in erster Linie die reibungslose, bildliche Darstellung von Wichtigkeit ist. Chris Hülsbeck bemerkt für die Programmierung auf dem C64:

„Wenn meine Musikroutine am Laufen war, dann war da noch ein anderer Prozess im Gange, der sog. Interrupt [...]. [M]eine Musikroutinen hatten normalerweise [ihren Platz]; zum Beispiel beim Raster-Rücklauf: Der Bildschirm wird gezeichnet und in dem Moment, wo der Raster wieder hochgeht in die linke obere Position, um ein neues Bild zu zeichnen, was ja 50 mal die Sekunde passiert bei einem PAL C64, konnte man den sog. Vertical Blank Interrupt ausnutzen. [...] Die Interrupt-Routine rettet alle Prozessor-Register [...] auf einen sogenannten Stack. Dann wird das Programm [...] abgearbeitet. Und wenn das fertig ist, wird der ganze Kram wieder zurückgeholt, es springt praktisch zurück dahin, wo es unterbrochen wurde. Der Prozessor in dem Moment weiß gar nicht, dass er unterbrochen wurde, [...] alle Register sind ja wieder so, wie sie waren.“ (Hülsbeck 2005a)

Nach der Abhängigkeit der musikalischen Gestaltung von der Interruptfrequenz gefragt, erläutert er:

„Wenn man sich in diesem Vertical Blank Interrupt eingeklinkt hat, musste man mit den 50 Hz oder 60 Hz leben. Dann waren natürlich die Teiler [entscheidend]. Die Maximalgeschwindigkeit wären ja 50 [bzw. 60] Beats pro Sekunde, [wenn] bei jedem Vertical Blank-Aufruf der Sequenzer einen Schritt weitergehen würde. Das ist natürlich viel zu schnell, da kann man keine sinnvolle Musik mit machen. Wenn man das teilt, z.B. jeden vierten oder jeden fünften [Interrupt], dann hat man schon eine vernünftige musikalische Geschwindigkeit. [...] Ich erinnere mich noch, die Standard-Geschwindigkeiten, für den 4/4 Takt waren 90, 126 oder 150 BPM [...] in denen ich meistens für die C64-Musik gearbeitet habe. [...] [Z]u dem Zeitpunkt haben wir überhaupt nicht für die 60 Hz-Variante entwickelt, sondern hauptsächlich für Deutschland und Europa [als Hauptmärkte des Herstellers Rainbow Arts, N.D.] [...]. Später hat man es entweder in Kauf genommen, dass das auf dem amerikanischen Gerät einfach schneller läuft und einen anderen Pitch hat, es hat sich ja auch ausgewirkt auf die Tonhöhe, weil der Soundchip anders getaktet war [...], oder es gab die Methode, alle sechs Durchläufe einen auszulassen, was dann allerdings ein bisschen hakelig war [...]. Freie Timer Interrupts konnte man im Spiel nicht benutzen, das Problem bestand darin: Es war alles so genau getimet. Beim Spiel hängt ja noch viel mehr am Vertical Blank Interrupt. Zum Beispiel wenn das smooth scrollen soll, muss alles in einem bestimmten Timing von statten gehen und man kann das nicht irgendwo unterbrechen. Die Spieleprogrammierer haben immer gefordert, dass man sich in den Vertical Blank einhängt und nicht mit einem Timer-Interrupt daherkommt. Der [...] Timer-Interrupt besitzt eine höhere

Priorität als der Vertical Blank [...] und die Musikroutine braucht auch eine gewisse Rechenzeit, um ihren einen Zyklus abzuarbeiten. Das konnte man sich also nicht erlauben“ (Hülsbeck 2005a).

Die Abhängigkeit der Musik-Routinen von bestimmten Interrupts ist grundsätzlich für Spielmusik als gegeben anzusehen, auch wenn diese angesichts der Technologien in heutigen Computerspielen keinen großen Einfluss mehr auf die erklingende Musik nehmen dürfte. Auch die in Kapitel 2.6.1 erläuterten Programmierung des PC-Speakers, um Samples wiederzugeben, ist im starken Maße von den Zyklen der internen Bauteile abhängig. Neben den offensichtlichen Beschränkungen durch Architektur der Soundchips und ihrer Umgebung kommen also weitere technische Aspekte hinzu, welche die Programmierung / Komposition von Computerspielmusik für Soundchips nachhaltig determinieren.

3.3 Zwischenfazit: Computerspielmusik im Spiegel ihrer Paradigmen

Musik für Computerspiele zwischen 1977 und 1994 ist bei ihrer Komposition und Programmierung nicht nur von den technischen Voraussetzungen abhängig, sondern muss ebenso den formal-funktionalen Anforderungen ihrer Verwendung entsprechen. Diese werden einerseits vom Spielinhalt bedingt (Anzahl, Art und Abfolge der integrierten Stücke). Andererseits dürfen die Musikdaten, v.a. auf den PSG-Soundchip-Plattformen, einen gewissen Speicherplatz und die entsprechenden Musikroutinen die ihnen zugestandenen Ressourcen nicht überschreiten. Diese für Soundchip-Musik kritischen Beschränkungen müssen bei der Programmierung ständig berücksichtigt werden, womit sie großen Einfluss auf die Musik nehmen. Das klingende Musikstück muss also immer als Ergebnis der für eine bestimmte technische Plattform mit eigenen Einschränkungen und Möglichkeiten erfolgten Programmierung begriffen werden. Sie ist damit in ihren medialen Nutzungszusammenhang integriert und von diesem nicht ohne weiteres trennbar¹⁸⁵.

Die Thematisierung der kreativen Programmierung bzw. der Komposition unter besonderer Berücksichtigung der technischen Einschränkungen scheint lohnenswert, verweist sie doch auf die Arbeitsweise klassischer Komponisten, wie Pummell anmerkt:

„We just had so many restrictions placed on us at that time. From a compositional standpoint [...] that limited power usually led to better compositions because [it's] much of the same way the classical composers [had to work]. Take Bach for an example: He knew that: 'all I got to work with is this small chamber orchestra. I have 8 players and I know my Cello player is not

¹⁸⁵ Erst seit der Entwicklung der sog. Emulatoren, also Programmen, die sich als Software genau so verhalten, *als ob* sie eine entsprechende Hardware-Plattform seien und entsprechende Instruktionen verarbeiten können, und der damit verbundenen Archivierung von aus Spielen extrahierter („gerippter“) Computerspielmusik ist diese von ihrem Nutzungszusammenhang befreit. Dasselbe gilt selbstredend auch für Computerspiel-Soundtracks (vgl. Kapitel 2.4).

very good. So I don't write anything difficult for the Cello-player. Sonically I know what I have to work with.' [...] [Y]ou deal with the talent that you're given and you make the best out of it. From a compositional standpoint that tends to lead to good compositions.“ (Pummell 2005)

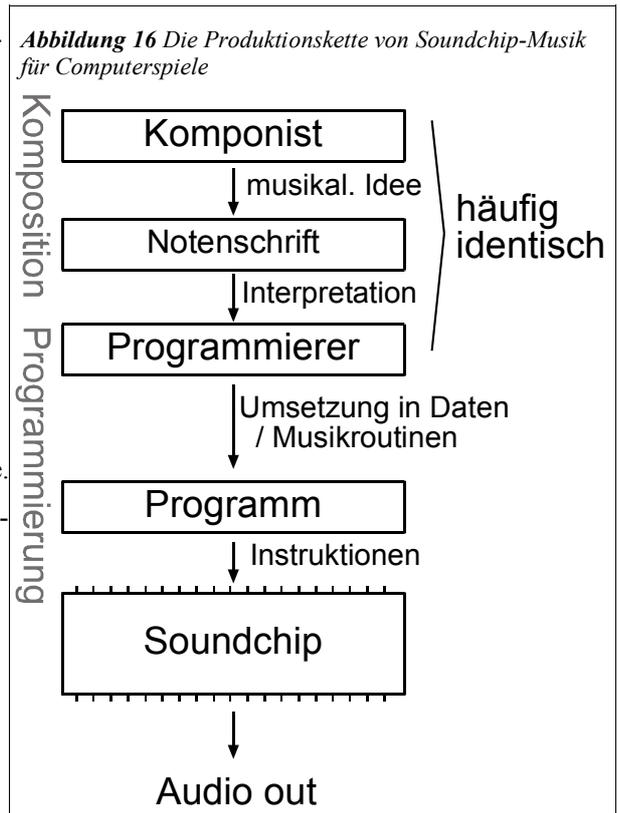
Dass bei der Komposition von Soundchip-Musik mit den Möglichkeiten und Begrenzungen auch der jeweiligen konkreten Hardware-Umgebung umgegangen werden muss, lässt aus einer detaillierten Auseinandersetzung mit dem Kompositions- und Produktionsprozess von Computerspielmusik schnell ein ausuferndes Unterfangen der wechselseitigen Betrachtung von Musik und (Computer-)Technik werden, wie die durch die Interviews gewonnenen Einblicke in die Komposition für Soundchips offenbaren.

Abb. 16 thematisiert die Produktionskette von Soundchip-Musik für Computerspiele. Zu beachten ist, dass bei der Programmierung für Soundchips eine Komposition in Notenschrift nicht bei jedem Komponisten stattfand. Die vorgestellten technischen Limitierungen nehmen wie erläutert Einfluss auf die Musik. Dies kann schon bei der Komposition geschehen, wenn sie vom entsprechenden Komponisten mitgedacht werden. Spätestens jedoch nehmen sie bei der Programmierung großen Einfluss, also dem Umsetzen der musikalischen Ideen in Programmcode. Dieser Aspekt spielt auch bei Adaptionen bekannter Melodien wie Titelsongs oder Jingles bekannter Lizenzen aus der Unterhaltungsindustrie¹⁸⁶ für Computerspiele eine große Rolle.

Der „Ausgang“ wurde in der Abbildung bewusst mit „Audio Out“ benannt, da dieser wiederum mit einem „Audio In“ verbunden werden kann, wodurch die dann aufgenommene Soundchip-Musik ihrerseits zu in den

Audio-Medien fixierten Medienmusik wird. Dieser Aspekt ist vor allem im Hinblick auf das vierte Kapitel von Interesse.

Interessanterweise hat der musikwissenschaftliche Diskurs die während einer kurzen Periode (~1981-1994) massenhaft im Verbund von Computerspielen und der sich darum entwickelten Kultur entstandene Soundchip-Musik bisher nicht thematisiert. Heimcomputer werden in der zu dieser Zeit veröffentlichten musikwissenschaftlichen Literatur vor allem im Hinblick auf ihre Steuerungsfähigkeiten mittels des noch neue MIDI-Standards wahrgenommen. Die interne Klangerzeugung der Heimcomputer scheint für eine Thematisierung nicht wertvoll genug. So wird sich vor allem mit den Implikationen der neuartigen Verfahren wie Sampling und FM-Synthese sowie der sich langsam abzeichnenden Digitalisierung der Produktionsmittel auseinander gesetzt¹⁸⁷.



¹⁸⁶ Beispiele stellen die bereits erwähnte Computerspiele MOONWALKER, M.A.S.H und STAR WARS dar.

¹⁸⁷ Exemplarisch seien an dieser Stelle Ackermann 1991, Ruschkowski 1998 und Harenberg 1989 genannt.

Am Ende dieses Zwischenfazits sollen einige, aus den genannten Aspekten hergeleiteten Anknüpfungspunkte für weitergehende musik- und medienwissenschaftliche Auseinandersetzungen benannt werden.

Musikalische Begriffe wie Werk und Material dürften im Spiegel der stark funktionalisierten Komposition sowie der Prozessierung von Daten, um diese als Instruktionen an die Klangerzeugung zu schicken, in weiterführenden Diskussionen von Interesse sein. Hierfür können sowohl die Sound-Routinen selbst als auch die programmiertechnische Praxis beleuchtet werden. Offensichtlich werden auch Anschlusspunkte für weitergehende Untersuchungen des Instrument-Charakters von Soundchips im Spiegel ihrer funktionellen Einbindung in ein der Musik übergeordnetem Werk und ihrer Mitwirkung an diesem. Eine Thematisierung der kompositorischen Praxis und das Verhältnis des Komponisten / Programmierers zu seinem Werk im Hinblick auf den stark anwendungsorientierten Charakter stellen ebenfalls interessante Untersuchungsfelder dar. Die nähere Betrachtung der Spannungsfelder Quellcode / Partitur sowie Programmierung / Komposition im Hinblick auf die erklingende Musik bietet sich daher unter Berücksichtigung der erläuterten Aspekte an.

3.4 Plattformübergreifender Hörvergleich

Die verschiedenen Soundchips und ihre Klanglichkeit werden nun in einigen Beispielen vergleichbar gemacht, um die Offensichtlichkeiten ihrer Klangerzeugung zu illustrieren. Dabei steht weniger deren funktionale Einbindung ins Spielgeschehen im Vordergrund sondern allein das klingende Ergebnis der vorangegangenen Prozessierung. Wie ersichtlich geworden ist, sind die Klangerzeugungsverfahren der unterschiedlichen Plattformen einander sehr unterschiedlich.

Pummell und Hülsbeck verweisen im Rahmen der geführten Interviews darauf, dass die Daten einer bestimmten Musik im Rahmen der Konvertierung für die „Zielplattform“ i.d.R. nutzlos sind, da die Soundchips und Hardware-Plattformen spezifische Eigenheiten besitzen und ein einheitlicher Standard für Musik-Daten wie MIDI für Soundchips nicht existiert (Hülsbeck 2005a, Pummell 2005).

Die Unterschiede der Soundchips im Hinblick auf Anzahl der Stimmen, Klangfarben und Umfang der einzelnen Musikstücke werden im Hörvergleich anschaulich. Die Auswahl ist dabei auf drei Spiele gefallen, die auf eine relativ große Zahl verschiedener Plattformen konvertiert wurden und sich daher für den Hörvergleich anbieten. Außerdem verdeutlichen vor allem die Adaptionen aus PARODIUS und THE SECRET OF MONKEY ISLAND den kreativen Umgang mit den limitierteren Klangerzeugern.

Parodius

Der seitlich scrollende Weltraum-Shooter PARODIUS ist als Parodie auf die erfolgreiche Konami-Serie GRADIUS (in Amerika NEMESIS, KONAMI 1985) erstmals 1988 für den MSX-Standard erschienen¹⁸⁸. Erst die 1990 erscheinende Automatenversion PARODIUS DA! wird konvertiert auch in Europa und Amerika verkauft¹⁸⁹.

Die Klangerzeugung auf dem Arcade-Board basiert ebenso wie der japanische Spielcomputer SHARP X68000¹⁹⁰ auf den YM 2151 Soundchip.



Abbildung 17 Screenshot aus Parodius Da! (Arcade-Version)

Die Musik des Spiels und seiner Nachfolger enthält Motive aus Tschaikowskys „Tanz der Kosaken“, „Marsch der Zinnsoldaten“ und dem „Blumenwalzer“ (alle aus der Nussknacker-Suite); Auszüge des Allegros aus Tschaikowsky Klavierkonzert Nr. 1 in b-Moll (op.23), Bizets „Farandole“ aus der Suite „L'Arlésienne“ no.2 und weitere Adaptionen von bekannten Themen der orchestralen Musik wie dem „Säbeltanz“ von Chatschaturjan, dem „Hummelflug“ von Rimsky-Korsakow, dem „Can-Can“ aus „Orpheus in der Unterwelt“ von Offenbach und Griegs „In der Halle des Bergkönigs“ (Peer Gynt op.23 Nr.7).

Hierbei wurden nicht für alle Plattformen sämtliche Musikstücke konvertiert, was unter anderem auf begrenzten Speicherplatz zurückzuführen ist. Auf der beiliegenden CD wird die Musik der Arcade-Version¹⁹¹ denen der Konvertierungen auf PC-ENGINE, SUPER NES, NES sowie GAME BOY gegenübergestellt.

Die Adaption von Bizets „Farandole“ (♩056, ♪057, ♪058, ♪059, ♪060) untermalt ebenso wie Rimsky-Korsakows „Hummelflug“-Thema (♩061, ♪062, ♪063, ♪064, ♪065) einen Bosskampf. Die Adaption des Allegros aus Tschaikowsky Klavierkonzert Nr. 1 in b-Moll (op.23) (♩066, ♪067, ♪068, ♪069, ♪070) dient in PARODIUS als Hintergrundmusik des zweiten Levels, der „Tanz der Kosaken“ (♩071, ♪072, ♪073, ♪074, ♪075) als Hintergrundmusik in Level 3. Alle Themen sind relativ kurz, nach etwas mehr als 20 Sekunden beginnt die Wiederholung.

Die Hörbeispiele lassen die verschiedenen Verfahren der Klangerzeugung der Soundchips deutlich erkennen. So klingt die PC-ENGINE-Version dem Automaten-Original am ähnlichsten, bis auf die Schlagzeug-Klänge, die beim PC-ENGINE fehlen, dafür in der SNES-Version aber deutlich wahrnehmbar sind. Die begrenzte Poly-

¹⁸⁸ Vgl. <http://www.msxnet.org/konami/parodius/> (25.06.2005); zum MSX-Standard: Fußnote 39.

¹⁸⁹ Umsetzungen existieren für NES (KONAMI, 1990), PC-ENGINE: (KONAMI, 1992), GAME BOY (KONAMI, 1991) und SUPER NES (KONAMI, 1992). Außerdem wurde das Spiel im Rahmen des PARODIUS DELUXE PACK auf der SONY PLAYSTATION (KONAMI, 1994) und dem SEGA SATURN (KONAMI, 1995) veröffentlicht.

¹⁹⁰ Die Version für X68000 unterstützt laut dem japanischen „Retro PC Music Streaming Radio“ auch das GM-Modul ROLAND MT-32. vgl. <http://gyusyabu.ddo.jp/MP3/1991/PARO1.html> bzw. <http://gyusyabu.ddo.jp/MP3/1991/1991.html> (25.06.2005).

¹⁹¹ Das Board von PARODIUS DA! enthält neben einem YM 2151 auch den Soundchip 053260 zum Abspielen von Samples, wie Stimmen und Schlagzeug-Klängen. Die Hörbeispiele der Automatenversion wurden während des Spiels im Emulator aufgenommen, was die Soundeffekte und im Vergleich zu ihnen relativ leise Musik erklärt.

phonie der GAME BOY und NES -Versionen ist den Beispielen gut zu entnehmen, auch die anderen Eigenheiten der PSG-Soundchips wie ihre Klangfarben und Klanglichkeit sind deutlich hörbar. Bei den Adaptionen für GAME BOY fällt auf, dass die digital erzeugten Einzel-Klänge keinen graduellen Hüllkurvenverlauf besitzen.

Bubble Bobble

Als weiteres Beispiel dient abermals das Spiel BUBBLE BOBBLE, wobei Hintergrundmusik samt Einleitung sowie Bonusrundenmusik verglichen werden sollen, deren Notenbilder sich auf den Seiten 70 und 71 dieser Arbeit finden.

Die Musik des mit den FM-Chips YM 2203 und YM 3526 bestückten Original-Automaten (♫041, ♫042, ♫045) wird ihren Konvertierungen auf

- COMMODORE C64 (♫076, ♫077),
- NES (♫078, ♫079),
- AMIGA (♫080, ♫081),
- SINCLAIR SPECTRUM 128 (bestückt mit einem AY-38910; ♫082, ♫083)
- und ATARI ST (♫084¹⁹²) gegenüber gestellt.

Die SID-Version für C64 offenbart hierbei den obertonreichen und damit vollen Eigenklang des Chips, die Versionen für den AY- bzw. YM-Chip in SPECTRUM und ATARI illustrieren die auf Rechteck-Wellen und Rauschen basierende Klangerzeugung.

Die AMIGA-Version verwendet kurze, gesampelte Wellenformen, klingt jedoch insgesamt wenig überzeugend, auch weil der Adaption der Musik auf die Bassstimme verzichtet wurde. Auffällig bei allen Beispielen ist die reduzierte Polyphonie gegenüber der originalen Version.

So verwenden alle Versionen für den *Ingame*-Teil der Hintergrundmusik nur 2 Stimmen. Die ATARI ST-Version greift hierbei auf die bereits erwähnten schnellen Arpeggien zurück, um trotz der beschränkten Polyphonie ein relativ volles Klangbild zu bieten.

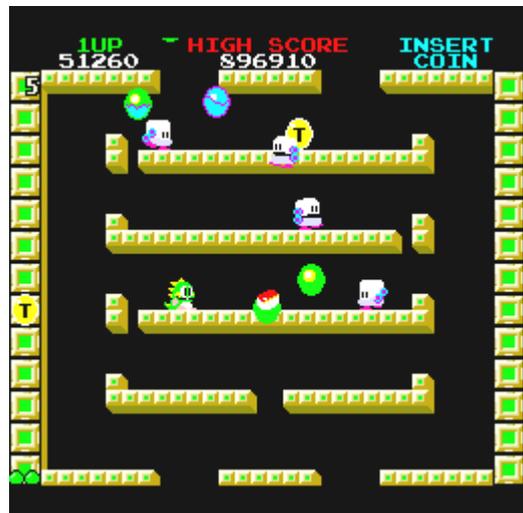


Abbildung 18 Screenshot aus Bubble Bobble (Arcade-Version)

¹⁹² Die Bonusrundenmusik der ATARI ST-Version war im WWW nicht zu finden.

The Secret of Monkey Island

Für einen Überblick der klanglichen Fähigkeiten der verschiedenen PC-Soundkarten soll die Titelmelodie des berühmten Grafikadventures THE SECRET OF MONKEY ISLAND (LUCASFILM GAMES, 1990) dienen, die in insgesamt zehn verschiedenen Varianten, von der einstimmigen PC-Speaker- bis zur LAPC-1-Version vorliegt.

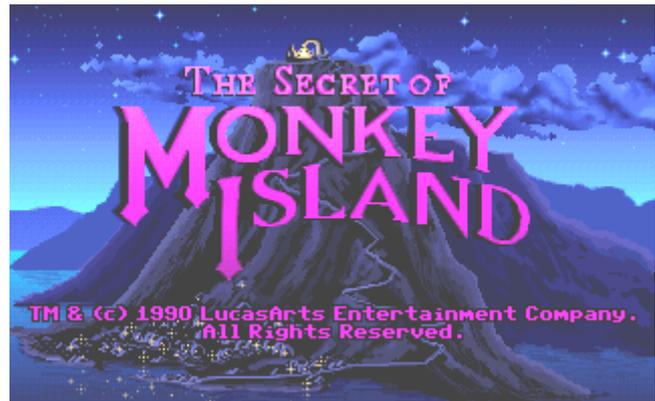


Abbildung 19 Titelschirm von *The Secret of Monkey Island* (PC-Version, VGA)

Die Arrangements für den PC-Speaker (♫014), den vierstimmigen SN 76489 des PC_{IR} (♫085) und den GAME BLASTER (♫086) illustrieren diese in Kapitel 2.6 vorgestellten Soundlösungen in ihrer Einfachheit in Relation zu den anderen, weiter entwickelten Soundkarten.

Die AD LIB-Version (♫087) benutzt typische, perkussive FM-Klänge, die das karibisch beeinflusste Thema passend instrumentieren.

Die klangliche Überlegenheit der LAPC-1- (♫088) und MT-32- (♫089) Versionen tritt ebenso deutlich hervor wie die Unterschiedlichkeit der das Instrumenten-Kit der ROLAND-Karten emulierenden GRAVIS ULTRASOUND (♫090) und SOUND BLASTER AWE 32 (im MT-32 Emulationsmodus; ♫091).

Letztere, wie die Versionen für die auf dem SOUND CANVAS SC55 basierenden Roland SCC-1 (♫092) und für den YAMAHA-Wavetable DB50XG (♫093) demonstrieren die in 2.6.5 angesprochenen Probleme der auf General MIDI basierenden Soundkarten:

So klingt die AWE 32-Version sehr 'dünn' und kann nicht mit den realistischer klingenden Samples der anderen Karten konkurrieren. Bei der SCC-1-Version ist die Melodiestimme auffällig leise. Von den auf Wavetable-Technologie basierenden Soundkarten klingen also nur die Versionen für LAPC-1, MT-32, GRAVIS ULTRASOUND und DB50XG in sich stimmig: Die einzelnen Spuren sind im Vergleich zueinander transparent gemischt und ausreichend hörbar. Deutlich zu hören ist außerdem der Hall-Effekt bei den ROLAND-Soundkarten LAPC-1 und MT-32.

4 Populäre Soundchip-Musik

Seit Ende der 1990er Jahre wird Soundchip-Musik auch als eigenständiges Genre innerhalb der populären elektronischen Musik wahrgenommen und bleibt somit nicht allein auf ihre Einbindung im Rahmen der Computerspiele oder Demo-Szene beschränkt.

Diese wird im Folgenden „populäre Soundchip-Musik“ genannt und teilt eine „Sound Culture“ mit der bisher vorgestellten Soundchip-Musik.

Für populäre elektronische Musik ab Anfang der 1990er Jahre¹⁹³ stellt Kleiner fest, dass sie „*immer eingeschrieben [bleibt] in die Epochalität des Technischen [und damit Medienmusik ist]*“ (Kleiner 2003: 9).

Für populäre Soundchip-Musik ist diese Feststellung freilich zutreffend, da neben der Klangerzeugung auch die digitale Umgebung in großem Maße technologisch determiniert sind. Im Hinblick auf kompositorische Strukturen entspricht populäre Soundchip-Musik ebenso eindeutig der von Großmann festgestellten 'Armut' von Medienmusik:

„Die Medienmusik der vergangenen 15-20 Jahre bringt kaum eine Innovation der musikalischen Struktur im Sinne neuer kompositorischer Prinzipien. Von der Geräuschkultur der Futuristen über die freie Tonalität, die Experimente des Bauhauses, die Serialität, die musique concrète bis zur Elektronischen Musik und Free Jazz (um nur einige Strömungen zu nennen) sind bis in die 70er Jahre die Felder für diese Art von Neuerung bereits abgesteckt. Statt dessen verändert sich die Relation von Musikpraxis, -rezeption und Material. Die spezifische Innovation der letzten Dekaden findet m. E. im Reibungsprozeß vom medienmusikalischer Produktion und der oben beschriebenen Diversifikation von Mediennutzung und Nutzergruppen statt.“
(Großmann 1997a: S.244)

Der Aspekt der originären Mediennutzung von bestimmten Nutzergruppen eröffnet einen interessanten Blick auf die herausgearbeitete Funktionalität besonders auch im Hinblick auf die zeitliche Abfolge:

So adressiert sich die frühe, für Computerspiele programmierte Soundchip-Musik in erster Linie an Rezipienten dieser, also an Computerspieler. Als solche ist sie unter den in Kapitel 3.1 vorgestellten Kriterien programmiert worden und wurde kaum von anderen Nutzergruppen rezipiert. Bis Mitte der 1980er Jahre bleibt Soundchip-Musik also vor allem auf ihre Anwendung in Computerspielen beschränkt, von Experimenten einzelner Hobby-Programmierer einmal abgesehen.

Die ihr zeitlich folgenden, vorerst im kulturellen Rahmen der Demo-Szene¹⁹⁴ entstehenden Chiptunes sind ebenso stark an eine eigene Mediennutzung und Nutzergruppe gebunden, nämlich die Demo-Szene selbst, die im Verbund mit gecrackter Spielesoftware verbreiteten Hacktros / Cracktros und ihre Musik sind eben-

¹⁹³ Abkürzend soll an dieser Stelle unter populärer elektronischer Musik all die Musik gefasst werden, die in den 1980er Jahren mit „House“ beginnt und durch die Digitalisierung der Produktionsumgebungen und einer hierdurch häufig beschworenen Demokratisierung der Produktionsmittel (Stichwort: „Bedroom Producer“) zu einem bunten Mit- und Nebeneinander vielfacher Stilistiken und Genrebezeichnungen geführt hat.

¹⁹⁴ Vgl. Tasajärvi (2004a: 12f) oder Fußnote 69 und 98.

falls nur von einer entsprechenden Audienz rezipiert worden. Gleichzeitig entsteht mit dieser Szene ein erster, von ihrer Nutzung in Computerspielen emanzipierter soziokultureller Rahmen für Soundchip-Musik. In diesem wird sie jedoch ebenfalls unter stark funktionalistischen Aspekten verwendet, nämlich um die selbstgeschriebenen Demos möglichst ressourcensparend musikalisch zu untermalen¹⁹⁵. Parallel wird mit den sog. *Music Disks*¹⁹⁶ eine erste eigenständige Form der Distribution von Soundchip-Musik gefunden.

Auch die populäre Soundchip-Musik, setzt auf Hörerseite eine gewisse Affinität mit dieser bzw. den entsprechenden Klängen voraus, was ihre Hörerschaft bzw. Nutzergruppe ebenfalls begrenzt. Ihre Wurzeln als Element eines digitalen Unterhaltungsproduktes mit eingeschränktem Rezipientenkreis spielen somit eine entscheidende Rolle in ihrer historischen Entwicklung, auch wenn ihre „Sound Culture“ mittlerweile in einem breiteren Rahmen wahrgenommen wird¹⁹⁷.

Im Folgenden wird das Konzept der „Sound Culture“ vorgestellt und für Soundchip-Musik erläutert. Ihr folgt die Benennung der technischen Verfahrensweisen und Tools, welche die Grundlage der populären Soundchip-Musik bilden. Im Anschluss sollen die beiden ästhetischen Haupt-Konzepte, Chiptunes und Micromusic voneinander abgegrenzt und beschrieben werden.

4.1 Sound Culture

Die Konzeption einer „Sound Culture“ der auf Soundchips basierenden Computerspielmusik folgt der Annahme, dass Soundchip-Musik vor allem aufgrund ihres „Sounds“ wegen von anderen Stilen elektronischer Musik abgegrenzt werden kann.

„Sound“ ist nach Büsser (1996: 6) für Popmusik mit Zunahme der technischen Möglichkeiten ein immer umfassenderes, für Stücke, Künstler und selbst Genres identitätsstiftendes Charakteristikum geworden. Der Begriff „Sound Culture“ kennzeichnet ihm zufolge den stattgefundenen Paradigmenwechsel in Musikproduktion wie -rezeption ab Anfang der 1960er Jahre und umfasst die Suche nach neuen Klangquellen abseits des traditionellen Instrumentariums sowie die Abwendung von der herkömmlichen Notation (Büsser 1996 : 7f). Gleichzeitig gewinnt „Sound“ gegenüber anderen musikalischen Parametern an Bedeutung. Da der „Sound“ von Musik vor allem von einer technologischen (Auf-)Bereitung eines Eingangssignal und die Fixierung dessen auf einem Audio-Medium bedingt wird, ist die Nähe der „Sound Culture“ zum Konzept der Medienmusik ist somit augenscheinlich.

¹⁹⁵ Anders als die später entstehende Tracker-Szene (vgl. Kapitel 2.5.3, S.38f), bei der die Komposition von samplebasierter Musik weniger von einem möglichst sparsamen Umgang mit Ressourcen geprägt ist.

¹⁹⁶ Auf diesen Floppydisketten begleitet eine grafische Player-Software den jeweiligen „Release“ der Music Group.

¹⁹⁷ Die „Sound Culture“ von Soundchip-Musik hat mittlerweile auch im Mainstream-Pop-Business Fuß gefasst: Das Lied „Girl“ des amerikanischen Musikers Beck (Auf dem Album „Guero“, erschienen bei INTERSCOPE, 2005) benutzt den GAME BOY als einleitendes Instrument sowie für Rhythmus-Elemente im Hintergrund. Auf der Maxi „Hell Yes“ befinden sich außerdem zwei Micromusic-Remixe von Beck-Titeln der Formation „8Bit“ sowie zwei des schwedischen Künstlers „Paza“ (ebenfalls erschienen bei INTERSCOPE, 2005).

Um das Verbinden von „Sounds“ mit beliebigen Bedeutungen zu umschreiben, bietet sich der von Jochen Bonz geprägte Begriff der „Sound Signature“ an, der von ihm wie folgt charakterisiert wird:

„Die Sound Signature ist ein Konzept der verbindlichen Bezugnahme. Indem sie massiv auf einen Kontext verweist, trifft sie zunächst keine Aussage, sondern gibt Auskunft über den Ort, an dem die Aussage entsteht. Im Sound der Signatur liegen die Informationen über den Rahmen der unmittelbaren Relevanz ihrer Rede; er zeigt die Landschaft, die Umgebung des Ortes. Dieses Eingeständnis ihrer Abhängigkeit löst die Aussage sogleich ein Stück weit aus ihrer Abhängigkeit heraus. Denn die als kontextabhängig gekennzeichnete Vorstellung gewinnt nun jenseits ihres eigentlichen Kontextes an Verständlichkeit.“ (Bonz 2001: 15f)

Die Signatur eines Sounds verweist also bei der Rezeption auf dessen Herkunft und sorgt gleichzeitig dafür, dass sein eigener Ursprungsort besser verstanden werden kann. Daher sind „Sound Signatures“ grundsätzlich vom Hören abhängig, da nur der jeweilige Hörer individuell den Verweis auf die Herkunft des Klangs deuten kann oder nicht. Aufgrund der eher auf Produktion und Produktionskultur, weniger auf Rezeption fokussierten Analyse soll an dieser Stelle primär die „Sound Culture“ von Soundchip-Musik betrachtet werden. Ihre „Sound Signatures“ sind dennoch für die hier beschriebenen Phänomene nicht uninteressant, verweisen sie doch auch nach der erfolgten 'Emanzipation' der Soundchip-Musik von ihrer Verwendung in Computerspielen weiterhin auf ihren Herkunftsort. Dieser liegt in 8 Bit-Hardwareumgebungen und damit in letzter Konsequenz im Klang der entsprechenden Computerspiele. „Sound Signatures“ sind somit für die Rezipienten und Mediennutzergruppen von großer Bedeutung.

Die Klanglichkeit von Soundchip-Musik ist, wie in den vorigen Kapiteln erläutert, nicht nur mit den Klangerzeugern sondern auch mit anderen technischen Beschränkungen der jeweiligen Hardwareplattform eng verwoben. Formale Eigenarten schreiben den „Sound“ von Soundchip-Musik ebenso mit wie die Verfahren ihrer Klangerzeugung. Häufig verwendete, formale musikalische Figuren in Computerspielmusik sind simple, oktavierte Bassläufe und die auf schnellen Arpeggien basierenden quasi-Akkorde. Auch diese sind vor allem von Beschränkungen bedingt. In einem größeren Maßstab betrachtet dürfen auch die einfachen Liedformen mit simplen Aufbau (vgl. Altmann 1968: 31ff) als typische formale Eigenart von Soundchip-Musik betrachtet werden, deren Instrumentierung generell mit musikalisch einfachen Mitteln geschieht¹⁹⁸.

¹⁹⁸ Oktavierte Bassläufe finden sich z.B. in BUBBLE BOBBLE (♩042), als Beispiel für die überaus typischen Arpeggien sei abermals auf das Thema von THE GREAT GIANA SISTERS (♩049) verwiesen. Beide Techniken sind auch in der ATARI ST – Version des BUBBLE BOBBLE-Themas zu hören (♩084). Viele Hintergrundmusiken verwenden außerdem klassische Liedformen (z.B. BUBBLE BOBBLE: AA'BB'). Das SUPER MARIO BROS -Thema stellt mit der 'Bridge' (D) und der Reprise ihrer zweiten Wiederholung (D') bereits ein aufwendigeres Arrangement dar: AABBC'CAADD'CC'D' (♩019).

4.2 Verfahrensweisen und Tools

Soundchip-Musik wie auch populäre elektronische Musik im Allgemeinen sind von der ihnen zu Grunde liegenden Technik determiniert, wären ohne ihre technischen Voraussetzungen also nicht denkbar. Die rasante Entwicklung der elektronischen, nun digitalen Musikproduktion seit Mitte der 1990er Jahre hat zu einer Reihe von neuen Phänomenen und einer rasanten Aufgliederung der Genres und Stilstiken im Bereich der populären Musik geführt¹⁹⁹. Für virtuelle Studioumgebungen ist mittlerweile eine große Zahl von verschiedenen Klangerzeugungs-PlugIns erschienen, die auf der Emulation von historischen elektronischen Klangerzeugern basieren. Diese werden auch als virtuelle Musikinstrumente bezeichnet²⁰⁰.

Emulation ist auch für Soundchip-Musik ein überaus wichtiges Verfahren, da erst sie es ermöglicht, die Klanglichkeit von Soundchip von ihren ehemaligen Beschränkungen emanzipiert zu nutzen. Auf emulierten Hardwareplattformen ist es außerdem möglich, für sie programmierte Kompositionstools zu nutzen. Darüber hinaus existieren verschiedene, die Klangerzeugung von Soundchips emulierenden Tools und VST-PlugIns. PlugIns, welche die Klangerzeugung der Soundchips emulieren, kokettieren dabei im Unterschied zu jenen, die klassische Synthesizer und anderen Studio-Gerätschaften emulierenden nicht mit der fotorealistischen Repräsentation einer Arbeitsoberfläche²⁰¹, auch da diese für Soundchips nicht existiert. Allein die möglichst exakte Nachbildung der ihrer Klangerzeugung zugrunde liegenden digitalen Schaltkreise ist bei ihrer Programmierung Anliegen. Auch die zahlreichen Wiedergabe-PlugIns für z.B. WINAMP, die das Abspielen diverser Dateiformate²⁰² von Soundchip-Musik ermöglichen, basieren auf Emulationen.

Die meisten Kompositionstools für Soundchips arbeiten mit einer Tracker-Oberfläche, die sich in grundlegenden Punkten von der eines MIDI-Sequenzers²⁰³ unterscheidet. Für die mit Soundchips ausgestatteten Heimcomputern (v.a. C64 und AMIGA) finden sich eine Vielzahl von Trackern, auf die nicht detailliert eingegangen werden soll. Dennoch sollen einige der neueren, teilweise auf Emulationen beruhenden Programme genannt werden.

¹⁹⁹ Diese Entwicklung kennzeichnende Stichworte sind unter anderem: virtuelle Studioumgebung, Emulation von Klangerzeugern, Software-Sampler und Bedroom Producer. Die Aufgliederung der Genres und Stilstiken zeichnen Bogdanov et al. (2001) nach.

²⁰⁰ Harenberg (2003: 69ff) setzt sich kritisch mit Aspekten der Entwicklung virtueller Instrumente auseinander, wie mit der bloß im Ab- und Nachbildcharakter verhafteten Interface-Metapher.

²⁰¹ Ein illustratives Beispiel für fotorealistische Nachempfndungen einer historischen Arbeitsoberflächen / Interfaces findet sich auf der Herstellerseite der Software-Firma ARTURIA, die eine Emulation des MOOG V-Modularsystems verkaufen: <http://www.arturia.com/en/moog/mmscreenshots.php> (25.07.2005).

²⁰² Es existieren z.B. das SID-, AY- und YM-Format für die entsprechenden Soundchips (SID, AY-Chip (v.a. MSX) und YM 2149 (ATARI ST)). NSF-, SPC-, GYM-, HES- und GBS-Format werden vor allem für die Speicherung gerippter Musik aus entsprechenden NES-, SNES-, MEGA DRIVE-, PC-ENGINE- und GAME BOY-Spielen benutzt.

²⁰³ Tracker stellen die verschiedenen Spuren als Spalten dar, in denen entlang der vertikalen Zeitachse in die als Hexadezimal-Zahlen kodierte Anweisungen geschrieben werden. Pro Schritt können pro Spur ein Klangereignis getriggert werden und außerdem verschiedene weitere Parameter-Werte definiert werden. MIDI-Sequencer wie CUBASE (STEINBERG) oder LOGIC (APPLE) repräsentieren die verschiedenen Events entlang einer horizontalen Zeitachse, die grafisch repräsentiert werden und feiner quantisiert werden können.

Selbst der Stella / TIA des ATARI VCS als technisch einfachster Soundchip wird von Programmierern für die Produktion von populärer Soundchip-Musik benutzt. Er wird dabei von den Musikern durch spezielle, selbst programmierte Musik-Software²⁰⁴ nutzbar gemacht. Für den AY/ YM 2149 existiert ein den Chip emulierendes VST-PlugIn²⁰⁵. Der SID-Chip ist durch seinen eigenständigen Klang, der Flexibilität sowie der weiten Verbreitung als Hauptauslöser für die Entstehung des Phänomens der Chiptunes zu werten. Auch die Herkunft der Demo-Szene und der damit verbundene kreative Output ist stark mit dem C64 verbunden. Für die heutige Computergeneration existieren verschiedene, auf der Emulation des Chips basierende Programme, wie das vier SIDs gleichzeitig emulierende VST-PlugIn QUADRASID oder der GOAT TRACKER²⁰⁶. Der SID ist auch der erste Soundchip, der als Basis für neue Hardware verwendet wurde: So der professionelle Synthesizer SID-STATION und die Soundkarte HARDSID²⁰⁷.

Das NES ist ebenfalls ab Ende der 1990er in den Fokus der Szene gelangt. Spielmusik wird seitdem öffentlich zugänglich gemacht, wobei diese Entwicklung vor allem mit dem NSF (Nintendo Sound File)-Format zusammenhängt. Der NERD TRACKER II (1998) stellt ein DOS-basiertes Tracker-Programm für die emulierte Soundhardware des NES dar. NERD TRACKER II bietet auch die Möglichkeit, Musikdaten als 6502-Maschinencode zu exportieren²⁰⁸. Die emulierte Soundhardware des NES ist außerdem Basis des VST PlugIns PEACH²⁰⁹. Neben der Komposition im Tracker oder per emulierter Klangerzeugung ist vor allem in Japan eine andere Methode des Zugriffs auf die Klangerzeugung des 2A03 weit verbreitet: Die FAMICOM-Soundhardware wird mittels CMK-Skriptsprache für die Musikwiedergabe programmiert. Eine Einführung in diese für Europa und USA ungewöhnliche Verfahrensweise gibt Johnson (o.J.). Durch ein MIDINES getauftes Modul für das NINTENDO ENTERTAINMENT SYSTEM kann man dieses als MIDI-Expander verwenden²¹⁰.

Der NINTENDO GAME BOY ist erst relativ spät für die Musikproduktion nutzbar gemacht worden. Ab dem Erscheinen der als Hobbyprojekte entstandenen Cartridges NANOLOOP (1998²¹¹) von Oliver Wittchow und LITTLE SOUND DJ (2000²¹²) von Johan Kotlinski setzt ein wahrer 'Boom' in der Micromusic-Szene ein. NANOLOOP ist

²⁰⁴ Beispielsweise das Programm 'TIAseq' von Noah Vawter, (<http://www.gweep.net/%7Eshifty/portfolio/tiasequencer/index.html>; 25.07.2004) oder auch die Programme 'loopcart' und 'synthcart' von Paul Slocum (<http://www.qotile.net/loopcart.html> bzw. <http://www.qotile.net/synth.html>; 23.07.2004).

²⁰⁵ In der Demo-Szene bekannt geworden ist der YAMAHA Nachbau YM 2149 des AY-Chip als Soundchip des ATARI ST. Daher wird er häufig als YM-Chip bezeichnet. Das diesen emulierende PlugIn YMVST ist als Freeware unter <http://www.preromainbritain.com/ymvst> (14.03.2005) verfügbar.

²⁰⁶ Das QUADRASID-PlugIn ist eines der wenigen kostenpflichtigen PlugIns aus dem Bereich der emulierten Soundchips, die Webseite des Herstellers findet sich unter http://www.refx.net/pro_QuadraSID.htm (15.06.2005). Der GOAT TRACKER kann unter <http://cadaver.homeftp.net/tools/goatrk2.zip> (18.07.2005) heruntergeladen werden.

²⁰⁷ Die Soundkarten HARDSID PCI und HARDSID QUATTRO PCI bieten bis zu vier originale SID-Chips auf einer Steckkarte, die über MIDI angesteuert werden (<http://www.hardsid.com/>; 01.08.2005).

²⁰⁸ Die Homepage des NERD TRACKERS findet sich unter <http://nesdev.parodius.com/nt2/index.html> (16.05.2005). Das NSF-Format enthält 6502er- Maschinencode und ist dem SID-Format ähnlich. Archivseiten für Musik im NSF-Format finden sich beispielsweise unter <http://www.zophar.net/nsf/> und <http://www.2a03.org> (14.07.2005).

²⁰⁹ PEACH ist erhältlich unter <http://www.tweakbench.com/instruments.php?id=9> (18.07.2005).

²¹⁰ Die Homepage von MIDINES lautet http://www.wayfar.net/0xf00000_overview.php (22.07.2005).

²¹¹ Eine erste Version des dreistimmigen NANOLOOP war bereits 1998 einsatzbereit, ab 1999 startete der Vertrieb mit der Version 0.8, im Jahr 2000 war die Version 1.0 fertig (Email von Oliver Wittchow an den Autor. 16.06.2005). NANOLOOP 2.0 für GAME BOY ADVANCE bietet sogar acht Stimmen (vgl. <http://www.nanoloop.de>; 14.07.2005).

²¹² Nach dem Beginn an dem Programm im Sommer 2000, war bereits im Herbst 2000 eine erste lauffähige Version fertiggestellt. Der Vertrieb der 1.0 Version startete im Frühjahr 2001 (vgl. Tasajärvi 2004c).

ein dreikanaliger 16-step-sequenzer, LITTLE SOUND DJ ein vierstimmiger Tracker mit vielfältigen Möglichkeiten. Beide erweitern die Klangerzeugung des GAME BOYS mittels einiger Tweaks²¹³ und lassen sich mittels eines MIDI-Interfaces zur MIDI-Clock synchronisieren²¹⁴.

Für die zeitlich nach den 8 Bit-Konsolen erscheinenden Soundchips gilt der in Kapitel 2.5.6 erläuterte Grundsatz, dass sie im Hinblick auf die ihnen zugrunde liegenden Verfahren (FM-Synthese und v.a. Sampling) weniger für die „Sound Culture“ von Soundchips stehen, sie also auch für populäre Soundchip-Musik seltener benutzt werden. Diese Feststellung ist leicht nachzuvollziehen, stellen doch sowohl FM-Synthese als auch Sampling für die allgemeine elektronische populäre Musik gängige Verfahren dar. Durch ihre vielfältige Klanglichkeit fehlen ihnen gleichzeitig die auf PSG-Soundchips und Computerspiele verweisende „Sound Signatures“. Dennoch existieren auch für die in AD LIB und SOUND BLASTER eingesetzten OPL-Chips eine große Anzahl von im Rahmen der Demo-Szene programmierter Musik. Auch auf dem AMIGA wurde für die Demo-Produktion eine große Anzahl von Chiptunes produziert, die eher auf kurzen, gesampelten Wellenformen als auf Samples von Instrumental-Klängen beruhen²¹⁵.

Im Bereich der Micromusic ist außerdem eine gänzlich neue Verfahrensweise des Umgangs mit Hardware entstanden: die Praxis des sogenannten *Circuit-Bendings*. Auf der Anti-theory-Webseite heißt es:

*„Circuit-bending is an electronic art which implements creative audio short-circuiting. This renegade path of electrons represents a catalytic force capable of exploding new experimental musical forms forward at a velocity previously unknown. Anyone at all can do it; no prior knowledge of electronics is needed. The technique is, without a doubt, the easiest electronic audio design process in existence“*²¹⁶.

Circuit-Bending (auch *Toy-Bending* genannt) umfasst die Modifikation von modernem elektronischem Spielzeugs mit integrierter Klangerzeugung. Deren Schaltkreise und meist auch ihr Aussehen werden modifiziert bzw. erweitert und -letztendlich- wird auch Musik mit ihnen gemacht. *Circuit-Bending* (übersetzt: „Schaltkreise Krümmen / Verbiegen“) ist ebenso wie Micromusic und Chiptunes ein Phänomen, das in einer kleinen Szene über Mailinglisten und Webseiten²¹⁷ von Enthusiasten verfolgt wird. Es ist mit dem Phänomen der Micromusic eng verbunden, in der weniger die Auseinandersetzung mit den tiefen des Computers dominiert, als bei den Chiptunes. Auch Modifikationen für C64, ATARI ST und GAME BOY existieren²¹⁸.

²¹³ So bieten beide eine Art von Frequenzmodulation, die auf der Technik beruht, die Tonhöhen-Register möglichst schnell (>50 Hz) mit entsprechenden Tonhöhen-Werten zu beschreiben. LSDJ enthält darüber hinaus ein Sample-ROM, das verschiedene Schlagzeug-Samples sowie die 59 Phoneme des Speech-Synthesis-Chips SP0256-AL2 zur Verfügung stellt.

²¹⁴ Das MIDI-Interface für GAME BOY wird auf der Seite <http://www.firestarter-music.de/lsdj/> dokumentiert, neben dem noch ein zweites Interface für Nanloop angeboten wird, vgl. <http://www.nanoloop.de> (alle URLs: 22.07.2005).

²¹⁵ OPL-Dateien lassen sich z.B. unter <http://www.chiptune.de> oder über die Portalseite <http://www.mirsoft.info>, entsprechende MOD-Dateien z.B. unter <http://www.chiptunes.org/files/> finden (alle URLs: 18.07.2005).

²¹⁶ <http://www.anti-theory.com/soundart/circuitbend/> (31.07.2005).

²¹⁷ Webseiten mit teilweise detaillierten Informationen, Abbildungen, Klangbeispielen und Anleitungen: http://homepage.mac.com/kaseo/circuit_bending/, <http://www.oscillateur.com/bending.php>, <http://hypatia.club.fr/circuit.html>, <http://www.anti-theory.com/soundart/> (alle URLs: 31.07.2005).

4.3 Genre-Bildung

Laut Toynbee²¹⁹ lässt sich die Genre-Bildung in der populären Musik durch die Analyse von vier sich ergänzenden Bereichen beschreiben:

1. Eine gemeinsame ökonomische Einstellung zum Markt,
2. das musikalische und darstellerische Handeln der Musiker,
3. die soziokulturelle Funktion des jeweiligen Genres, sowie
4. eine geteilte Produktionskultur der betreffenden Musiker.

Die Bildung von Genres ist ihm zufolge sowohl identitätsstiftendes Moment für die Zugehörigen einer bestimmten Produktionskultur, als auch ein ordnendes, Orientierung bietendes für die Rezipienten der populären Musik und damit den Nutzergruppen. Für die Charakterisierung der populären Soundchip-Musik sind die ersten beiden Punkte Toynbees weniger Genre-spezifisch als die letzten beiden, weswegen auf sie nur kurz eingegangen wird.

Die ökonomische Einstellung der Akteure basiert größtenteils auf dem soziokulturellen „Miteinander“ (s.u.): Ihren historischen Wurzeln entsprechend ist populäre Soundchip-Musik i.d.R. frei verfügbar. Auch die zur Produktion von Soundchip-Musik benötigte Software wird überwiegend kostenlos angeboten. Dennoch existieren einige Kleinst-Labels²²⁰, die populäre Soundchip-Musik auf physischen Tonträgern veröffentlichen. Auftretende Künstler lassen sich bei populärer Soundchip-Musik vor allem im Bereich der Micromusic ausmachen, die mit Chiptunes die Produktionskultur im Hinblick auf die verwendeten Klangerzeuger teilt.

Die soziokulturelle Dimension stellt für Soundchip-Musik allgemein ein wichtiges Charakteristikum dar, wie die Vorstellung der Nutzergruppen gezeigt hat. Toynbee zufolge können populärmusikalische Genres auf bestimmte Gruppierungen, Gemeinschaften oder Bewegungen beschränkt bleiben, die einen ähnlichen kulturellen Hintergrund teilen (vgl. Toynbee 2000: 110f). Die Szene der populären Soundchip-Musik ist vor allem über das Internet organisiert, wo sie in verschiedenen Portalen / Communities rund um das jeweilige geteilte Haupt-Interesse zum Thema (Chiptunes, Demo-Szene, Emulation, *Video Game Music* als MIDI, Rip oder MP3) einen regen Austausch pflegt.

Die Produktionskultur von Soundchip-Musik hat sich seit den Produktionsbedingungen im Rahmen ihrer Verwendung in Computerspielen stark verändert. Sowohl die vorgestellten, auf der Emulation von Soundchips basierenden PlugIns und Tracker, als auch die in den vergangenen Jahren allgemein erfolgte Entwicklung im Bereich der elektronischen, nun mehr digitalen Musikproduktion stellen für sie eine Grundvoraus-

²¹⁸ Kratzberg (2002) schildert detailliert den Einbau eines zweiten SIDs im C64, um ein Pseudo-Stereo-Signal zu produzieren, die Micromusic-Band „Dropdabomb“ dokumentieren den Umbau eines ATARI MEGA STS, um die einzelnen Kanäle des YM 2149 mischen zu können (<http://www.dropdabomb.org/stu/ymixer/>; 01.08.2005). Der GAME BOY kann umgebaut werden, um seinen Klang basslastiger zu machen.

²¹⁹ Einen Überblick über seine Methode mit Nennung der folgenden Kategorien bietet Toynbee auf den Seiten xx-xxiii der Einleitung (Toynbee 2000: xxxff).

²²⁰ Wie BLEEPSTREET, REBEL PET SET, JENKA MUSIC, RELAX BEAT, PEN PAL RECORDINGS etc.. Der Begriff „Kleinst-Labels“ markiert hierbei den wenig kommerziellen Charakter der meist als Hobby-Projekte geführten Labels.

setzung dar. Einen wichtigen Aspekt im Hinblick auf die Verfügbarkeit der Produktionsmittel als auch den Charakter der Soundchip-Musik-Szene als soziokulturelle Bewegung markiert die meist freie Verfügbarkeit dieser Tools.

Diese Haupt-Charakteristiken ziehen auch Implikationen für die erklingende Musik nach sich: Der örtlich ungebundenen, vor allem durch das WWW organisierten und wenig hierarchischen Struktur dieser Szene entsprechend, ist grundsätzlich eine große Bandbreite verschiedener musikalischer Stilstiken in Soundchip-Musik vertreten. So finden sich sowohl Polkas, Reggae- oder Country-Adaptionen, als auch dementsprechend instrumentierte Techno- oder Drum 'n Bass-Tracks sowie experimentelle Musik und Cover-Versionen von Klassikern der Popmusik. Selbst zu kleinen Sparten der populären Musik wie „Death Metal“ finden sich z.B. in der High Voltage SID-Collection entsprechende Adaptionen.

Außer den in Kapitel 3.2 erwähnten kompositorischen Folgen aus den Beschränkungen für Computerspiel-musik, die vor allem in Form einer auf musikalischen Formen basierenden „Sound Culture“ in die populäre Soundchip-Musik einfließen (so basieren Polka- als auch Reggae-Akkorde i.d.R. auf den schnellen Arpeggien), sowie einem häufig zu konstatierenden Aufgreifen von musikalischen Mitteln der jeweils aktuellen Pop-musik (also vor allem der 1980er und frühen 1990er Jahre), existieren für populäre Soundchip-Musik wenig stilistische Konventionen.

Für die populäre Soundchip-Musik sollen im Folgenden zwei Pole benannt werden, welchen die Künstler graduell zugeordnet werden können. Beide Pole tragen die Bezeichnung von großen, sich der jeweiligen Form widmenden Portalseiten im Internet, nämlich Chiptunes und Micromusic. Die beiden Begriffen werden in Texten über das Phänomen (vgl. z.B. Collins 2003a :2) nicht deutlich umrissen und voneinander abgegrenzt gebraucht. Beim Gebrauch innerhalb der sich mit den Genres gebildeten Gemeinschaften werden beide Begriffe häufig synonym gebraucht, auch Bezeichnungen für Sub-Genres, wie „Chipcore“ oder „Chip-Hop“ werden benutzt, um eigene Stilstiken zu benennen.

Die online verfügbare Stil- und Genre-'Landkarte' elektronischer Musik, „Ishkur's Guide to Electronic Music“, siedelt die beiden Genres unter den Namen „VGM“ (Video Game Music) und „Casiocore“ (entsprechend Chiptunes und Micromusic) als Subgenres von „Techno“ an²²¹. Ein weiterer erwähnenswerter Bereich sind vor allem aus den USA stammende Cover-Bands, die bekannte Computerspielmelodien als klassische Rock-Band (Gitarren, Schlagzeug, Bass) neu interpretieren²²².

²²¹ „Ishkur's Guide to Electronic Music V.2.5“: <http://www.di.fm/edmguide/edmguide.html> (19.07.2005). Diese Ressource stellt nur eine von vielen Versuchen dar, Genres und Stile populärer elektronischer Musik zu klassifizieren und kategorisieren. Der bisher umfassendste Versuch einer solchen Kanonisierung, die Publikation „All Music's Guide to Electronica“ (Bogdanov et al. 2001) erwähnt die hier unter Micromusic und Chiptunes gefassten Genres gar nicht.

²²² Z.B. „the minibosses“ (<http://www.minibosses.com>) oder „the advantage“ (<http://theadvantageband.com>; beide URLs: 07.07.2005).

4.4 Chiptunes

Der Begriff Chiptunes hat ebenso die Soundchips als Klangerzeuger wie auch das Musikstück (engl. „tune“) bereits in sich vereinigt. Im Rahmen dieser Arbeit und dem Angebot von Portalseiten wie Chiptunes.org oder Chiptune.com²²³ entsprechend, kennzeichnen die sog. Chiptunes Musik, die *ausschließlich für einen bestimmten Soundchip* bzw. eine bestimmte Plattform programmiert worden ist. Ansporn des Komponisten / Programmierers ist es dabei, zwischen Fähigkeiten und Limitierungen dieses Soundchips ein möglichst aufregendes Musikstück hervorbringen.

Sowohl die Verfahrensweisen als auch die Wurzeln von Chiptunes hängen also neben der Produktion von Computerspielmusik stark mit der sog. Demo-Szene zusammen, welche seit Mitte der 1980er Jahre vor allem auf den Heimcomputer-Plattformen COMMODORE C64, ATARI ST und AMIGA aktiv ist.

Diese soziokulturelle Bewegung entzieht sich bis heute weitestgehend der Wahrnehmung der breiten Öffentlichkeit, ist jedoch über das WWW sehr gut organisiert und bis in die heutige Zeit aktiv. Wie der unter seinem Pseudonym „Goto 80“ selbst in der Demo-Szene aktive Musiker Anders Carlsson²²⁴ im Gespräch bestätigt, handelt es sich bei Demos primär um „Geek Art for Geeks“²²⁵, bei der sich sowohl die Produzenten, als auch die meist auf die Demo-Szene beschränkten Rezipienten kaum um eine weitere Verbreitung kümmern. Der Begriff Chiptunes wurde von dieser Szene geprägt und kennzeichnete am Anfang die auf AMIGA entstandenen, MOD-Dateien, die auf kurzen Wellenformen basieren und geloopt abgespielt werden²²⁶. Der Grundsatz, auf einer bestimmten, limitierten Hardware-Basis ein möglichst spektakuläres Ergebnis zu erzielen und dabei Ressourcen schonend zu programmieren, ist bei Demos von großer Wichtigkeit. Neben der kulturellen Praxis des Hacking (vgl. Kapitel 3.2, S.84f) als Grundlage für den gewünschte audiovisuellen Output und den damit verbundenen technischen Aspekten ist für die in der Szene Aktiven vor allem eine mit die mit der Kultur des Hip Hop vergleichbare Distinktionskultur²²⁷ von Wichtigkeit.

²²³ <http://www.chiptune.com> (12.05.2005). Die Webseite bietet Chiptunes für verschiedene Plattformen an, wobei Micromusic nicht darunter fällt. Es handelt sich also um Stücke für einen einzelnen Soundchip bzw. eine bestimmte Plattform. Hinter den Internet-Seiten <http://www.chiptune.de> und <http://www.chiptunes.org> (beide 07.07.2005) stehen Archive für AD LIB, ATARI ST und AMIGA-Chiptunes in den Dateiformaten YM, BIN, HSC, RAW, SA2 und verschiedenen MOD-Formaten, die u.a. mit dem DELIPLAYER 2 abspielbar sind.

²²⁴ Die Demo-Group von Carlsson trägt den Namen „Hack 'n Trade“, in der er selbst seit 1993 (dreizehnjährig) aktiv ist. In der HVSID-Collection (<http://www.hvsc.c64.org/>; 26.07.2005) finden sich 81 Dateien von ihm, er tritt seit 2001 auch live auf. Seine Musik wurde bisher u.a. auf folgenden Tonträgern / Alben veröffentlicht: „Commodore Grooves“ (REBEL PET SET 2005, CD), „Bravor“ (CANDYMIND 2005, MP3), „Contech“ (8BITPEOPLES 2005, MP3/MCD), „Copyslave EP“ (20KBPS RECORDS 2005, MP3), „Bushrunner EP“ (PENPAL 2002, 7"), „Papaya EP“ (BLEEPSTREET 2001, 7"), „Philemon Arthur and the VIC“ (GOTO80.COM 2002, MP3/SID) (vgl. <http://www.goto80.com>; 25.07.2005).

²²⁵ Mit dem englischen Ausdruck „Geeks“ (engl. (slang) für: Streber, Stubengelehrter) werden i.d.R. Computer-Nerds (Nerds = engl. (slang) für Fachidiot, Sonderling etc.) bezeichnet. Das Zitat entstammt einem Gespräch mit dem Autor, 21.06.2005.

²²⁶ Die MODs des Künstlers „4-Mat“ der Demo-Group „Arnachy“ verwenden beispielsweise einzelne Rechteckwellenformen, die als verschiedene Samples mit verschiedenen Offset-Einstellungen abgespielt werden, was zu einem der Pulse Width Modulation entsprechenden, lebendigeren Klangbild führt (♩094, 095).

²²⁷ Im Hip Hop „batteln“ die verschiedenen Crews mit Worten, „dissen“ sich gegenseitig, während Graffitis, Kleidung und Musik die Identität der eigenen Crew festigen. Bei Demos vergleichen sich die verschiedenen Demo-Groups durch die in Teamarbeit programmierten Demos, die auf entsprechenden Partys miteinander im Wettbewerb stehen.

Der beim *Hacking* geltende Grundsatz einer Herausforderung durch Technik besteht auch bei Chiptunes fort: Die technisch limitierte Soundhardware wird als Voraussetzung der gesamten Musik genommen, womit eine Vergleichbarkeit zu anderen Künstlern entsteht und die 'Skills' des Programmierers, also seine Fertigkeiten im Umgang mit der bestimmten Hardware, objektiviert werden können. „Beste C64-, AMIGA- oder ATARI ST-Musik einer maximalen kB-Größe“ sind bis heute anzutreffende Kategorien bei Wettbewerben auf den (Partys oder Conventions genannten) Veranstaltungen der Demo-Szene. Dennoch darf die Chiptunes-Szene nicht allein der Demo-Szene zugerechnet werden. Vor allem nicht-europäische und jüngere Künstler kennen den sozialen Rahmen der Demo-Szene kaum.

Der amerikanische Musiker „Nullsleep“ (Jeremiah Johnson)²²⁸ sieht neben der von der 8 Bit-Technik ausgehenden Faszination auch ohne biografischen Demo-Szene-Hintergrund in der technischen Limitiertheit der Plattformen einen wichtigen Faktor für sein kreatives Schaffen:

“[A]lthough there is some degree of nostalgia involved as these were the games that I grew up with, the real attraction of making music in this way is the challenge of working within the limits of these pieces of hardware. In a game boy you have 4 channels to work with [...]. With an NES you basically have the same setup [...]. Now composing interesting tunes under these constraints makes one squeeze everything they can out of these little soundchips, tweaking wave properties on every frame, or using hyper arpeggios to 'fake' chords ... In my opinion, this fosters creativity and imagination [...].” (zit. n. Zaremba 2002, Hervorh. i. O.)

Die Herausforderung durch Technik treibt manche Programmierer zu noch älteren Plattformen: So benutzt ein kleiner Kreis von Musikern das ATARI VCS mit seinen 32 Stimmen pro Klangkanal, um Chiptunes zu produzieren. Es wird dafür aufgrund seiner mangelnden Polyphonie auch doppelt verwendet, wie in Paul Slocums „Up on the Housetop“ (♫097).

Chiptunes-Künstler pflegen einen regen Austausch über das Internet wobei neben Demo-Szene-Portalen wie pouet.net auch Seiten wie micromusic.net oder vorc.org²²⁹ eine wichtige Rolle spielen.

Chiptunes können offensichtlich gegliedert werden nach den verschiedenen Hardwareplattformen / Soundchips, für die sie komponiert werden. Außerdem lassen sich nostalgische, konservative oder auch 'retro' Chiptunes von denen trennen, die modernere Musikstile reflektieren. Die Künstler ersterer bleiben in einer ge-

²²⁸ „Nullsleep“ ist ein überaus respektierter Chiptunes-Künstler und Mitglied des Kollektivs 8bitpeople (<http://www.8bitpeople.com>; 01.08.2005). Sein die 256 möglichen Speicherstellen für verschiedene Patterns im GAME BOY-Modul LITTLE SOUND DJ komplett ausnutzender „Megamix“ berühmter „Depeche Mode“-Songs (♫096) ist in der Szene legendär.

²²⁹ Die japanische Portalseite <http://www.vorc.org> (07.06.2005) widmet sich neben Rezensionen von im Internet veröffentlichten Chiptunes bzw. Micromusic auch dem Austausch zwischen japanischer Soundchip-Musik-Kultur und denen Amerikas und Europas. Die Seite <http://www.pouet.net> (13.07.2005) ist eine Plattform rund um die aktuelle Demo-Szene. <http://www.micromusic.net> (12.06.2005) stellt eine vielbesuchte Plattform für Micromusic und Chiptunes dar.

wissen Nostalgie mit den 1980er Jahren bzw. der 8 Bit-Zeit verhaftet und wollen diese mit denselben Maschinen, Klängen und / oder Musikstücken zelebrieren. So finden sich häufig Remakes oder Konvertierungen bekannter Themen von Computerspiel-Klassikern von einem Chip zum anderen²³⁰.

Auch lassen sich die für AMIGA entstandenen Chiptunes der Demo-Szene von jenen für PSGs differenzieren. Chiptunes für den AMIGA benutzen wie erwähnt verschiedene Tracker als Hauptkompositionstools und damit Tracker-Formate als Dateiformat, in welchen jedoch in erster Linie kurze, geloopte Wellenformen als Samples verwendet werden, die eine den PSGs ähnliche, simple Klanglichkeit bei entsprechend geringem Speicherplatzverbrauch ermöglicht und die in Trackern entstandenen Chiptunes deutlich von der sog. Tracker- oder MOD-Musik (vgl. Kapitel 2.5.3) abgrenzt.

Ein eigenes Format für Chiptunes der Demo-Szene wurde ab Anfang der 1990er Jahre mit den sog. „Music Disks“ gefunden, auf welchen eine bestimmte Anzahl von Chiptunes von einer eigens programmierten Player-Software präsentiert werden²³¹. Chiptunes werden häufig in eigenen Dateiformaten für die jeweiligen Soundchips bzw. den auf Emulation beruhenden Playern publiziert²³².

Die Künstler der sich mittlerweile entwickelten „New School“ von Chiptunes verwenden die Soundchips in einem weniger auf authentische Hardware und 8 Bit-Programmierung fokussierten Kontext, um moderne Musik, oder solche, die nicht unbedingt zeitgenössisch auf Heimcomputern zu finden war, zu komponieren. So finden sich zum Beispiel dem Reggae ähnliche oder 'jazzige' Arrangements bei Künstlern wie „Goto 80“ („Ajvar Relish“; 🎵098) und „YMCK“ („Pow*Pow“; 🎵099), die in ihrer musikalischen Offenheit und der Verwendung von auch anderen Klangquellen nicht mehr den rigiden Anforderungen der genuinen Chiptunes entsprechen. So verwundert es nicht, dass derartige Projekte aufgrund des häufig verwendeten Gesangs und den 'fremden' Klangquellen von rigorosen Chiptunes-Anhängern als „Lamer“ abgelehnt werden.

²³⁰ Beispielsweise die Adaptionen verschiedener C64-Musikstücke wie „Giana Sisters“ und „Shades“ für den YM2149 von Jochen Hippel, die sich in dem Archiv http://www.chiptune.de/ym/c64_conversions.zip (19.07.2005) befinden.

²³¹ Ein Archiv für Amiga-Musicdisks findet sich unter <http://ada.planet-d.net/indexdemo.php?category=musicdisk> (19.07.2005).

²³² Neben den verschiedenen MOD-Formaten existieren z.B. das SID-, AY-, YM- und NSF-Format für die Musikdaten der jeweiligen entsprechenden Soundchips (SID, AY-Chip, YM 2149 und 2A03; s. auch Fußnote 197).

4.5 Micromusic

„Low tech music for high tech people“ heißt es bereits im Untertitel der 1999 vom ehemaligen ETOY-Mitglied Carl gegründete Micromusic.net-Webseite. Der Fokus bei Micromusic liegt weniger auf der Produktion möglichst 'authentischer' Soundchip-Musik, sondern vielmehr auf der Klangästhetik von 8 Bit- und anderen 'lofi'²³³-Klangquellen. Es werden also ebenso Heimcomputer und Spielkonsolen wie auch Personal Keyboards oder Plastikspielzeug²³⁴ als Klangerzeuger verwendet. Sie ist somit grundsätzlich von einem Nebeneinander verschiedener Klangerzeuger geprägt. Ihre Nähe zu Computerspielmusik ist damit weniger offensichtlich als bei Chiptunes.

Auf der Micromusic-Webseite ist die verfügbare Musik mittels der Attribute *funky*, *sporty*, *relaxy*, *lovely*, *worky*, *spooky*, *fucky*, *sexy* und *no style* sortiert, das Micro-Level des Users (ein mit Rollenspielen oder Internet-Foren vergleichbares Erfahrungspunkte-System) bestimmt, wie umfassend der Zugriff auf die in der Micromusic-Datenbank gespeicherten MP3-Dateien ist. Damit die auf der Seite verfügbare Musik einer gewissen Klammer in Bezug auf Klanglichkeit und Qualität entspricht, werden hochgeladene Tracks in einem „quality filtering system“ auf ihre „Micromusic-Qualität“ hin untersucht, sprich: selektiert.

Historisch gesehen lässt sich Micromusic u.a. aus den Chiptunes ableiten, deren Klangerzeuger erst durch die vorgestellte Entwicklung in der digitalen Musikproduktion verfügbar wurden. Chiptunes-inspirierte Micromusic stellt aus der Perspektive der rigorosen, sehr auf die Technikkultur erpichten Anhänger der Chiptunes in gewisser Art „eine Musik der Unbefugten“ dar²³⁵, woran sich jedoch angesichts der rasanten technologischen Entwicklung nicht mehr wirklich gestört wird.

Daneben wirken verschiedene andere Einflüsse auf sie ein. So ist grundsätzlich die Popmusik der 1980er und 1990er Jahre zu nennen, deren musikalischen Formen auch in vielen Chiptunes-Coverversionen aufgegriffen werden, darüber hinaus können auf elektronischen Klangerzeugern basierende Musikrichtungen wie (für Deutschland) die „Neue Deutsche Welle“ als historische Vorläufer der Micromusic gelten. Ein gewisser DIY (Do it yourself)-Geist ist außerdem häufig erkennbar, wie der Bereich des *Circuit-Bending* eindrucksvoll zeigt.

Die Portalseite micromusic.net hat sich über ihre Datenbankfunktion für frei verfügbare Musik im MP3-Format hinaus als Kommunikationsmittel für populäre Soundchip-Musiker weltweit etabliert. Konzertanfragen und Organisation der „Micro Eventz“ genannten Partys werden über die Seite geregelt. Eine umfangreiche Links-Rubrik stellt ferner die neuesten PlugIns, Labels oder Artikel zu den Themen Soundchip- und Lofi-Musik vor.

²³³ Der Begriff „lofi“ bedeutet low fidelity und ist somit vor allem als Abgrenzung zum Hifi (high fidelity)-Sound zu sehen.

²³⁴ Eine interessante Sammlung dieser mit Bewertung für die Verwendung in Micromusic findet sich unter http://users.informatik.haw-hamburg.de/~windle_c/TableHooters/instruments.html (18.07.2005).

²³⁵ Dieses Zitat ist eine Anspielung auf die technische Entwicklung der elektronischen Klangerzeugung, die ab dem Ende der 1960er Jahre den Zugriff der Popkultur auf die bisher der Ernsten Musik vorbehaltenen Technologie ermöglichte, womit in der Folge eine von Brus und Meifert als „Musik der Unbefugten“ klassifizierte „Sound“-Musik möglich wurde (zit. nach Büsser 1996: 11).

Auch bei Micromusic ist eine Prägung von Computerspielmusik teilweise erkennbar, die sich neben Remixen bekannter Themen vor allem im Aufgreifen archetypischer formaler Figuren äußert. Ein Beispiel ist das Stück „Breakout Button“ der finnischen Gruppe „Desert Planet“ (♫100), in dem neben einer Fülle klassischer Soundeffekte auch die für Soundchip-Musik typischen, oktavierten Bässe verwendet werden und ein Großteil der verwendeten Klangfarben denen der frühen Soundchips (einfache Rechteckwellen ohne nennenswerte Klangmanipulation) entspricht. Im genannten Beispiel erklingen gesampelte Drumloops zusammen mit u.a. vom COMMODORE VC20 gesampelten Soundeffekten.

Micromusic kann aufgegliedert werden in diese 'New School der Chiptunes' und die eher Performance-orientierte Toy- oder Gadget- Micromusic. Erstere lässt sich von Chiptunes vor allem durch die Verwendung der Soundchip-Klänge in Form von Samples oder PlugIns abgrenzen. Sie bricht die aus den Produktionskulturen der Computerspiele und Demos stammenden Paradigmen der Chiptunes auf und mischt deren Klangfarben und Verfahrensweisen mit ihr fremden (Desert Planet). Die Technologie von Soundchips wird in Micromusic also aufgrund des eigenen „Sounds“ und der mit diesem verbundenen „Sound Culture“ (bzw. seiner „Sound Signatures“ wegen) verwendet. Die Technikkultur der Soundchips ist zwar interessant und geschätzt aber anders als für Computerspielmusik und Chiptunes nicht Grundlage und Rahmen des eigenen Schaffens. Für Micromusic ist somit die „Sound Culture“ und nicht so sehr die Technikkultur von Soundchips von Interesse.

Die sog. Toy- oder Gadget-Micromusic baut klingendes Plastik-Spielzeug und Kinder-Keyboards, die meist auf einfachen Sample-Schaltkreisen basieren und ggf. ein Ergebnis des eigenen *Circuit-Bendings* darstellen, in die Musik mit ein. Auch GAME BOYS und andere Soundchip-Klangerzeuger werden gerne verwendet. Eine spektakuläre Live-Performance, häufig mit Verkleidungen, unterstreicht den sehr auf Unterhaltung, Performance und einer gewissen DIY-Attitüde ausgelegten künstlerischen Impetus dieser Musiker. Ein Beispiel stellt die französische Gruppierung „Teamtendo“ (♫101) dar, die in Hamster-Kostümen auftreten und in ihrer Musik viel elektronisches Spielzeug verwenden, außer dem Namen und den teilweise verwendeten GAME BOY-Klängen jedoch musikalisch wenig Anknüpfungspunkte an Chiptunes erkennen lassen.

Auf der einen Seite ist Micromusic also von den stark durch Technologien determinierten Produktionskulturen der Soundchip-Musik der 1980er Jahre befreit, auf der anderen Seite stellt sie deren Fortführung dar und besitzt ebenso ihre eigene technologische Grundlage. Erst das Phänomen der Micromusic hat zu einer größeren Verbreitung der ihr zugrunde liegenden „Sound Culture“ beigetragen und Soundchip-Musik in die öffentliche Wahrnehmung gelangen lassen. Die Micromusic enthaltenen „Sound Signatures“ können gleichzeitig dafür sorgen, dass ihre Rezipienten wiederum ein gesteigertes Interesse gegenüber der Klangerzeugung und den historischen Vorläufern in Form von Demo-Chiptunes und Computerspielmusik entwickeln. Neben der „Sound Culture“ sind also vor allem die gestiegenen Möglichkeiten der digitalen Musikproduktion und die zeitlich parallel verlaufende, zunehmende Vernetzung der Computernutzer für das Phänomen der Micromusic verantwortlich zu sehen, welches im Rahmen der grundsätzliche Diversifizierung im Bereich der populären elektronischen Musik seit Ende der 1990er Jahre seinen Platz in der elektronischen Musikkultur gefunden hat.

5 Fazit und Ausblick

Im Rahmen der Arbeit sollten medienmusikalische Prozesse unter Berücksichtigung der eigenen „Medienrealität“ von Soundchip-Musik beschrieben werden. Die vorliegende Arbeit stellt somit eine erste Reflektion des Phänomens der Soundchip-Musik dar, die den von Großmann postulierten Forderungen für Medienmusik zu entsprechen versucht. So wendet sich dieser gegen die Grundannahmen einer veralteten musikwissenschaftlichen Perspektive auf medienmusikalische Prozesse:

„Solange medienmusikalische Prozesse primär unter dem Blickwinkel des 'Vermitteln' bzw. 'Abbildens', bzw. der Manipulation, der Täuschung oder Simulation betrachtet werden [...], bleibt Musik in den Medien lediglich als Reproduktion oder Substitut akzidentiell zu einer unabhängig von ihren Mitteln existenten 'wahren' Musik. Die zu beschreibende Differenz liegt m.E. gerade nicht auf dichotomischen Ebenen wie 'reales Objekt' vs. 'mediales Abbild' oder 'Wahrheit' vs. 'Schein', sondern vermutlich in einer spezifischen Materialität der Medien und der Herausbildung spezifischer Handlungszusammenhänge. In diesem Sinne wird [...] versucht, Musik in den Medien als Medienmusik im Sinne eines technikkulturellen Begriffs zu beschreiben, als eine Musik, der die elektronischen Medien nicht nur äußerlich bleiben“
(Großmann 1997b: 63, Hervorh. i. O.).

Als von ihren technischen Voraussetzungen determinierte Musik stellt Soundchip-Musik ein Paradebeispiel für die Verschränkung von Technologie und klingenden musikalischem Ergebnis dar. In Abgrenzung zu anderen Genres populärer Musik ist in ihr ein Genre zu sehen, das gar mit einer bestimmten Technologie der Klangerzeugung synonym gesehen werden kann. Ein derart offensichtlicher Zusammenhang von Musik und Technik in der populären Musik ist als einzigartig zu werten²³⁶. Die Medienrealität von Soundchip-Musik ist entlang der erfolgten Analyse immer wieder durch die ihr zugrunde liegenden diversen Limitierungen und Beschränkungen charakterisiert worden.

Historisch lassen sich für Soundchip-Musik aus der Analyse drei geschichtliche Entstehungsphasen mit jeweils eigenen Produktionskulturen nachzeichnen:

- Die erste ist geprägt von der Anwendung in Computerspielen und reicht von den ersten Musiken für Computerspiele (ab Anfang der 1980er Jahre) bis zu den späten Titeln für z.B. NES oder C64 (Anfang der 1990er Jahre) sowie GAME BOY (bis Ende der 1990er Jahre); neue Computerspiele für die eigentlich obsolet gewordenen Plattformen werden indes auch heutzutage noch von einer kleinen Heimentwickler-Szene programmiert.
- Die zweite Phase der Soundchip-Musik ist von ihrer Anwendung im Rahmen der Demo-Szene geprägt. Sie entsteht vor allem auf den Heimcomputern C64, AMIGA und ATARI ST bis Ende der 1990er Jahre und wird ebenfalls bis in die heutige Zeit weitergeführt.

²³⁶ Für House oder Techno-Stile können einzelne elektronische Instrumente wie z.B. die ROLAND TB-303 stilbildend gewesen sein, trotzdem werden z.B. in Acid-House zwingend auch andere Klangerzeuger verwendet.

- Die dritte Phase kennzeichnet die aus der „Sound Culture“ von Soundchip-Musik entstehende populäre Soundchip-Musik, die unter den eingeführten Begriffen Chiptunes und Micromusic gefasst wird.

Aus den genannten Punkten wurde eine mögliche Visualisierung der verschiedenen Entwicklungsstadien von Soundchip-Musik im Spiegel der parallel fortschreitenden Entwicklung der von Computerspielen genutzten Klangerzeugungen entwickelt, wie in Abb. 20 zu sehen²³⁷. Sie bestärkt die am Ende des letzten Kapitel formulierte These, nach der ein Wegfallen der Soundchip-„Sound Culture“ im Bereich der Computerspielmusik erst zu Chiptunes und verbunden mit dem Fortschritt der digitalen Musikproduktion zur populären Soundchip-Musik geführt hat.

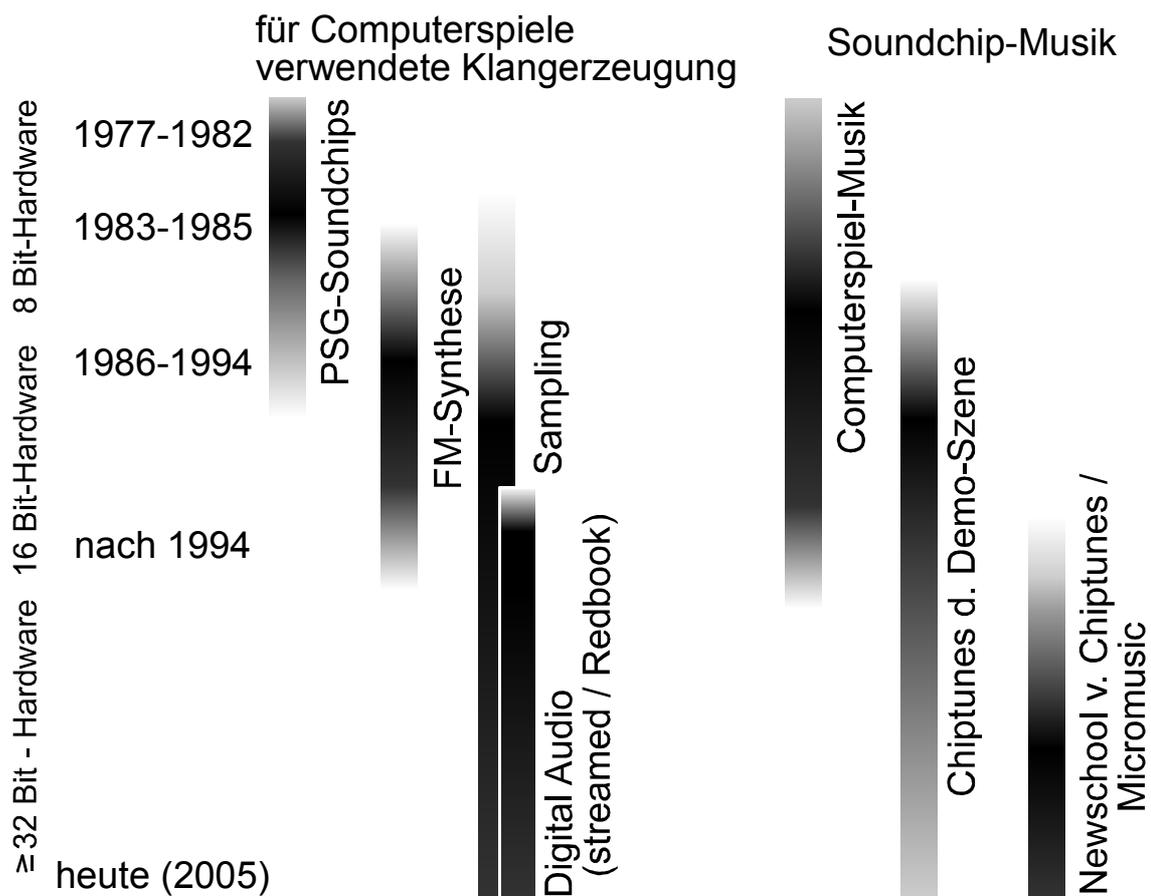


Abbildung 20 Gegenüberstellung der technischen Verfahren zur Klangerzeugung in Computerspielen und der Entwicklung der Genres von Soundchip-Musik im zeitlichen Rahmen.

Es ist auffällig, dass Soundchip-Musik für Computerspiele nur in einem äußerst begrenzten Zeitraum (~1981-~1994) programmiert wurde. Die Demo-Szene stellt selbst in heutiger Zeit noch ein vitales Umfeld für Soundchip-Musik dar und die auf Basis der „Sound Culture“ entstandenen Stilrichtungen von modernen Chiptunes und Micromusic kennzeichnen die Anwendung der Klangerzeugung im Rahmen der popularmusikalischen Verwendung, welche ohne die stattgefundenene Entwicklung von der elektronischen zur digitalen Musikproduktion kaum möglich gewesen wäre.

²³⁷ Die verschiedenen Graustufen markieren das Aufkommen der verschiedenen Kategorien in den entsprechenden Zeiträumen.

Eine erste Betrachtung der Komposition und Programmierung von Computerspielmusik wurde in Kapitel 3 geleistet, aus welcher auch das Hauptfazit der Arbeit zu ziehen ist, nämlich die Abhängigkeit ihrer Medienrealität von der Einwirkung der limitierten Technologie und funktionalen Notwendigkeiten. Diese nehmen auf den kompositorischen Prozess und damit das musikalische Ergebnis gehörigen Einfluss.

Michael Pummell bemerkt zu dieser mittlerweile obsoleten Technologie im Vergleich zur Entwicklung für die Musik in Computerspielen nach 1994²³⁸:

„The thing about that older technology: it had limitations. They [die Soundchips, N.D.] gave you limitations and spelled it right out. Here is what you have to work with, make the best of it. And that's what we don't have now. We don't have any restrictions now. [...] In a way it's good, we don't have boundaries any more, we can do anything we want²³⁹, but on a downside, it has had a negative impact on compositional techniques. Because you need to define boundaries on a composition. That technology that you are talking about, that generation, gave us those boundaries. You could be a good composer but if you didn't know what you're doing musically, there's no way you could maintain anyone's interest beyond three minutes“ (Pummell 2005)

Ein kompositorischer Wert von Soundchip-Musik, der gleichzeitig auch ihre Qualität darstellt, lässt sich ihm zufolge also aus dem Umgang mit der limitierten Technologie ableiten. Auch wenn diese Perspektive nicht grundsätzlich romantisierbar ist. Chris Hülsbeck z.B. bewertet die technologische Entwicklung hin zu Sampling und Redbook-Audio grundsätzlich positiv, wenn sich auch bei ihm ein Gefühl der Nostalgie einstellt:

„Das finde ich natürlich grundsätzlich positiv, weil man sich nicht mehr unbedingt mit den Bits und Bytes rumschlagen muss. Es hat alles seinen Reiz aber irgendwann wird das auch langweilig und mühsam und das ist ein Grund, warum ich mir die alten Zeiten nicht unbedingt zurückwünsche. Auf der anderen Seite hat aber eben auch diese Chipmusik und diese ganze Ära eine bestimmte [...] Ästhetik, die verloren gegangen ist. Eine Verspieltheit oder so, die einem teilweise heutzutage fehlt. Es gibt zum Beispiel nix, was klingt wie ein C64. Da gibt es nichts vergleichbares“ (Hülsbeck 2005a).

Die Einzigartigkeit der in PSG-Soundchips enthaltenen Klangerzeugung, welche in Form der „Sound Culture“ aufgegriffen und fortgesetzt wird, markiert einen weiteren nicht zu unterschätzenden Aspekt für das medienmusikalische Phänomen von Soundchip-Musik, der wiederum stark mit den technischen Limitierungen und somit ihrer Medienrealität zusammenhängt.

²³⁸ Diese ist von der Implementierung von *streamed* bzw. Redbook-Audio und der damit verbundenen Befreiung der Komponisten / Programmierer von den vorher rigiden Limitierungen geprägt.

²³⁹ Auch für heutige Audio-Engines bestehen freilich weiterhin Begrenzungen in Hinsicht auf Rechenleistung etc., die von Pummell erwähnten Freiheiten betreffen in erster Linie die weggefallenen Beschränkungen der Klangerzeugung.

Die Verschränkung von Technologie und musikalischer Form markiert somit die wichtigste Eigenart von Soundchip-Musik, die sie gegenüber anderen Musikgenres auszeichnet und abgrenzt und auf die sich alle ihrer angesprochenen Eigenheiten immer wieder zurückführen lassen. Diese treten am offensichtlichsten in der klanglichen und kompositorischen 'Armut' v.a. der frühen Computerspielmusik hervor, lassen sich jedoch ebenso deutlich in den Strategien der Gestaltung finden.

In den klanglichen und formalen Eigenheiten der Soundchip-Musik und damit ihrer „Sound Culture“ wird die Wechselwirkung zwischen Limitierungen und musikalischem Ergebnis immer wieder thematisiert. Auch im zeitlichen Rückblick ist die Entwicklung der musikalischen und klangästhetischen Form von Computerspielmusik vor allem im Spiegel der nun *wegfallenden* technischen Limitierungen zu sehen, wodurch diese als Hauptmerkmale von Soundchip-Musik abermals bekräftigt werden.

Bibliographie

Computerspielmusik

Belinkie, Matthew (1999): „Video game music: not just kid stuff“

URL: <http://www.vgmusic.com/vgpaper.shtml> (25.10.2004)

Boer, James (2003): „Game Audio Programming“, Hingham

Brandon, Alexander (2003): „Adaptive Audio: A Report“ in: Sanger, George Alistair: „The Fat Man on Game Audio: Tasty Morsels of Sonic Goodness“, S. 194–209. Indianapolis

Brandon, Alexander (1998): „Interactive Music: Merging Quality with Effectiveness“

URL: http://www.gamasutra.com/features/19980327/brandon_01.htm (28.10.2004)

Brandon, Alexander / **Fuller**, Brad (1999): „A Look Back“ in: Brandon, Alexander (Hg.): „The Interactive Audio Journal“, Vol. 1 No.3, November 1999.

URL: <http://www.iasig.org/pubs/journals/jour0103.shtml#Features> (06.05.2005)

Clark, Andrew (2001): „Adaptive Audio“

URL: http://www.gamasutra.com/resource_guide/20010515/clark_pfv.htm (01.11.2004)

Collins, Karen (2004a): „'One Bit Wonders' – Technology's Effect on early Video Game's Music“

URL: <http://www.dullien-inc.com/collins/texts/onebit.pdf> (25.10.2004)

Collins, Karen (2004b): „Fine Tuning the Terrible Twos: The Musical Aesthetics of the Atari VCS“

URL: <http://www.dullien-inc.com/collins/texts/flat2.pdf> (25.10.2004)

Collins, Karen (2003a): „From Bits to Hits: Video Game Music changes its Tune“

URL: <http://www.dullien-inc.com/collins/texts/bits2hits.pdf> (25.10.2004)

Collins, Karen (2003b): „Video Game Audio – work in progress“

URL: <http://www.dullien-inc.com/collins/texts/vgaudio.pdf> (26.10.2004)

Dittbrenner, Nils (2002): „Mobiles 8Bit-Gewitter. Be A Little Sound DJ With Your Game Boy.“ in: De:Bug. Zeitung für Elektronische Lebensaspekte. Ausgabe 60, Jahrgang 6, Juni 2002, S.34

Edge (2003): „Dance Dance Evolution“ in: Edge. The Future of Electronic Entertainment. Ausgabe 124, Jahrgang 10, Juni 2003, S. 54-61

Fukai, Atsushi (o.J.): „An Introduction to Game Music“

URL: <http://mushi.net/hanashi/gmintro.html> (25.10.2004)

Glashüttner, Robert (2000): „SID. Der Soundchip des Commodore 64“ (unveröffentlichte Diplomarbeit, AED 3/2000 SAE Wien)

Leonard, Sean (2001): „Scores of Glory, Fantasy, and Plumbing: The Concise History and Principal Genres of Video Game Music“ URL: <http://www.seanspace.com/iSphere/scores.htm> (27.10.2004)

Marks, Aaron (2001): „The Complete Guide to Game Audio“, Lawrence

McDonald, Glenn (2004): „Gamespot Features: A History of Video Game Music“

URL: <http://www.gamespot.com/features/6092391/index.html> (14.03.2005)

N.N. (2003): „Game Audio Gallery: Michael Pummell“

URL: http://www.gamasutra.com/galleries/audio/michael_pummell/index.htm (01.11.2004)

N.N. (o.J.): „Classics Arcade Database – Audio CPU“ URL: http://iep.ath.cx/cad/index_audio.htm (27.10.2004)

N.N. (o.J.): „Explore: Video Game Music – Dictionary of Computer and Video Games“

URL: http://www.explore-games.com/computer_and_video_games/v/video_game_music.html (08.04.2005)

- N.N.** (o.J.): „MAME emulated Soundchips“ URL: <http://www.mameworld.net/hardware/sound.html> (27.10.2004)
- N.N.** (o.J.): „Retro Game Music Streaming Radio“ URL: <http://gyusyabu.ddo.jp/MP3/MP3.html> (15.06.2005)
- Pidkameny, Eric** (2002): „Levels of Sound“ URL: <http://www.vgmusic.com/information/vgpaper2.html> (25.10.2004)
- Ross, Rob** (2001): „Interactive Music...er, Audio“
URL: http://www.gamasutra.com/resource_guide/20010515/ross_01.htm (28.10.2004)
- Sanger, George Alistair** (2003): „The Fat Man on Game Audio: Tasty Morsels of Sonic Goodness“, Indianapolis
- Slocum, Paul** (2003): „Atari 2600 Music And Sound Programming Guide“ URL: http://qotile.net/files/2600_music_guide.txt (17.02.2005)
- Stockburger, Axel** (2003): „The Game Environment from an auditive Perspective“
URL: <http://www2.hku.nl/~audiogam/ag/articles/gameenvironment%20axel%20stockburger.pdf> (20.07.2005)
- Stolberg, Eckhard** (2000): „Atari 2600 VCS Sound Frequency and Waveform Guide“
URL: <http://home.arcor.de/estolberg/texts/freqform.txt> (15.04.2005)
- Tasajärvi, Lassi** (2004c): „Utilising a handheld toy from the past.“ in: ders. (Hg.): „Demoscene: the art of real-time“, S.37-44. Helsinki
- Tozuka, Yoshikazu** (1999): „Video Game Music“ (jp., ISBN 4-89369-707-2), Tokyo
- Trixter** (o.J.): „IBM PC Ramblings“ URL: <http://www.oldschool.org/sound/pc> (28.10.2004)
- Turcan, Peter / Wasson, Mike** (2004): „Fundamentals of Audio and Video Programming for Games“, Redmond
- Vosburgh, Matthew** (o.J.): „The Complete Works of Rob Hubbard: Interview by Electronic Music Maker“
URL: http://www.freenetpages.co.uk/hp/tworh/int_elec.htm (10.05.2005)
- Wetzke, Jürgen** (2000): „Digital Sound and Music in Computer Games“
URL: <http://www.tu-chemnitz.de/phil/hypertexte/gamesound/> (26.10.2004)
- Whalen, Zach** (2004): „Play Along – An Approach to Videogame Music“
URL: <http://www.gamestudies.org/0401/whalen/> (25.01.2004)
- Wing, Eric** (o.J.): „The History of PC Game Midi“
URL: <http://www.queststudios.com/quest/midi.html> (28.10.2004)
- Wireman, Casey** (2003): „Sound Formats and Their Uses in Games“
URL: <http://www.gamedev.net/reference/articles/article1902.asp> (28.10.2004)
- Zaremba, Jon** (2002): „Interview with Nullsleep“ URL: <http://www.savetheuniverse.us/null.htm> (12.03.2004)

Computerspielhardware und -software

- Barrett, Terry / Colwill, Steve** (1984): „Das große Commodore 64 Spiele-Buch“, Landsberg am Lech
- Bessler, Harald** (1987): „Computerzeit. Wissenswertes rund um den Computer“, Haar bei München
- Burnham, Van** (2001): „Supercade. A visual history of the videogame age 1971-1984“, Cambridge / Massachusetts
- Dachsel, Thomas** (1984): „Das Musikbuch zum Commodore 64“, Düsseldorf

- Eccles**, Allen (o.J.): „Gamespy Hall of Fame: The Ad Lib Sound Card“
URL: http://archive.gamespy.com/legacy/halloffame/adlib_a.shtm (27.10.2004)
- Forster**, Winnie (2005): „Gameplan 1.5 - Spielkonsolen und Heim-Computer 1972-2005“, Utting
- Forster**, Winnie (2002): „Gameplan - Spielkonsolen und Heim-Computer 1972-2002“, Utting
- Göhler**, Stefan (2004): „Phonomenal ... ein Rückblick auf die Soundkartengeschichte“
URL: <http://crossfire-designs.de/link/soundkarten/> (28.10.2004)
- Graetz**, J.M. (2001): „The Origin of Spacewar!“ in: Burnham, Van (2001): „Supercade. A visual history of the videogame age 1971-1984“, S.42-48. Cambridge / Massachusetts
- Grote**, Florian (2002): „Die Entwicklung von Sampling auf Heimcomputersystemen (C64 bis Echtzeit-Granularsynthese). Unveröffentlichte Hausarbeit.
URL: <http://www.florian-grote.de/pdf/softsampling.pdf> (27.10.2004)
- Herman**, Leonhard (2001): „Atari VCS“ in: Burnham, Van (2001): „Supercade. A visual history of the videogame age 1971-1984“, S.148-151. Cambridge / Massachusetts
- Höh**, Martin / **Pütz**, Klaus (1986): „Das MSX - Buch“, Düsseldorf
- Juul**, Jesper (2001): „A clash between Game and Narrative. A thesis on computer games and interactive fiction“ (OA: 1999) URL: <http://www.jesperjuul.dk/thesis/AClashBetweenGameAndNarrative.pdf> (25.06.2003)
- King**, Geoff / **Krzywinska**, Tanya (Hg.)(2002): „ScreenPlay“, London
- Kratzberg**, Reiner (2002): „Pseudo Stereo-SID. Ein Stereo-Klang der anderen Art“, in: „GO 64! Das Commodore – Fanzine“ Jahrgang 6, Ausgabe 10/2002, S. 8f
- Lischka**, Konrad (2002): „Spielplatz Computer. Kultur, Geschichte und Ästhetik des Computerspiels“, Hannover
- Mädicke**, Bernd (o.J.): „Die Geschichte von Atari“
URL: <http://www.maedicke.de/atari/history.html> (24.02.2005)
- Munnik**, Josha / **Oostendorp**, Eric (1992): „Das Sound Blaster Buch“, Düsseldorf
- N.N.** (2000): „Exotica! Formats / Replays Page“ URL: <http://exotica.fix.no/frames.html> (21.03.2004)
- N.N.** (2001): „Super Mario Sales Data: Historical Units Sold Numbers for Mario Bros. on NES, SNES, N64...“
URL: http://www.gamecubicle.com/features-mario-units_sold_sales.htm (15.06.2005)
- Otten**, Hans (2004): „Yamaha Sound generators of the 1980s home computers“
URL: <http://www.members.chello.nl/h.otten/vortexion.htm> (02.04.2005)
- Pias**, Claus (2002): „Computer Spiel Welten“, München
- Sellers**, John (2001): „Arcade Fever – The Fan's Guide to the Golden Age of Video Games“, Philadelphia
- Sheff**, David (1993): „Nintendo - 'Game Boy'. Wie ein japanisches Unternehmen die Welt erobert“, München
- Stolz**, Axel (1992): „Das grosse Soundblaster-Buch“, Düsseldorf
- Tasajärvi**, Lassi (2004a): „A Brief History of the Demoscene_“ in: ders. (Hg.): „Demoscene: the art of real-time“, S.11-32. Helsinki
- Tasajärvi**, Lassi (2004b): „Let it beep (Case: Tracker Music)“ in: ders. (Hg.): „Demoscene: the art of real-time“, S.33-35. Helsinki
- Vogel**, James / **Scrimshaw**, Nevin B. (1984): „Commodore 64 Musik-Buch“, Basel

- Wirsig**, Christian (2003): „Das große Lexikon der Computerspiele“, Berlin
- Wolf**, Mark J.P. (2001b): „Space in the Video Game“ in: ders. (Hg.) „The Medium of the Video Game“, S. 51-75. Austin
- Wolf**, Mark J.P. (2001a): „The Video Game as a Medium“ in: ders. (Hg.) „The Medium of the Video Game“, S. 13-33. Austin
- Wolf**, Mark J.P. (2001c): „Time in the Video Game“ in: ders. (Hg.) „The Medium of the Video Game“, S. 77-91. Austin
- Wurster**, Christian (2002): „Der Computer. Eine illustrierte Geschichte“, Köln
- Yamauchi**, Hiroshi (2003): ‚Family Computer‘ – On Its 20th Anniversary“ in: Inuoe, Kentaro (Hg.): „Family Computer 1983-1994. Tokyo Metropolitan Museum of Photography“, S.6-7. Tokyo
- Young**, Sean (o.J.): „MSXnet: Parodius“ URL: <http://www.msxnet.org/konami/parodius/> (25.06.2005)

Scans von originalen Datenblättern / Application Manuals

- Commodore Semiconductor Group** (o.J.): „HMOS 6581 Sound Interface Device, preliminary information“
URL: <http://www.funet.fi/pub/cbm/documents/chipdata/6581.zip> (27.10.2004)
- N.N.** (o.J.): „AY-3-8910/8912 Programmable Sound Generator Data Manual“
URL: <http://bulba.at.kz/AY-3-8910.rar> (27.10.2004)
- N.N.**: (o.J.) „AY-3-8910/8912/8913 Programmable Sound Generator Datasheet“
URL: http://bulba.at.kz/AY-3-8910_2.rar (27.10.2004)
- N.N.** (o.J.): „The Engineering Staff of TEXAS INSTRUMENTS Semiconductor Group: SN 76489 AN“
URL: <http://www.funet.fi/pub/msx/mirrors/msx2.com/vortexion/sn76489.zip> (14.03.2005)
- The Atari Historical Society** (o.J.): „Atari CO 12294 Pokey“
URL: <http://www.funet.fi/pub/msx/mirrors/msx2.com/vortexion/pokey.pdf> (02.04.2005)
- Yamaha Corporation** (o.J.): „YM2151 Application Manual [ohne Titelblatt gescannt, N.D.]“
URL: <http://www.funet.fi/pub/msx/mirrors/msx2.com/vortexion/ym2151docs.pdf> (02.04.2005)
- Yamaha Corporation** (o.J.): „Yamaha LSI YM 2610 アプリケーションマニュアル [Application Manual, N.D.]“ (Japanische Version) URL:
<http://www.ylw.mmtr.or.jp/~hosino/ma-net/sndgen/japanese/manual/ym2610/ym2610am.htm> (02.04.2005)
- Yamaha Corporation** (o.J.): „Yamaha LSI YM3526 Data Manual“
URL: <http://www.funet.fi/pub/cbm/documents/chipdata/ym3526/index.html> (27.10.2004)
- Yamaha Corporation** (o.J.): „Yamaha LSI YM3812 FM Operator Type-L II (OPL II)“
URL: <http://www.funet.fi/pub/msx/mirrors/msx2.com/vortexion/ym3812.pdf> (02.04.2005)
- Yamaha Corporation** (1987a): „Yamaha LSI YM2149 Software-Controlled Sound Generator (SSG)“
URL: <http://bulba.at.kz/ym2149.rar> (27.10.2004)
- Yamaha Corporation** (1987b): „Yamaha LSI YM2151 FM-Operator Typ-M (OPM)“
URL: <http://www.funet.fi/pub/msx/mirrors/msx2.com/vortexion/YM2151.pdf> (02.04.2005)
- Yamaha Corporation** (1987c): „Yamaha LSI YM2203 FM Operator Type-N (OPN)“ URL:
<http://www.funet.fi/pub/msx/mirrors/msx2.com/vortexion/ym2203.pdf> (27.10.2004)
- Yamaha Corporation** (1987d): „Yamaha LSI YM2413 OPLL (FM Operator Type-LL)“
URL: <http://www.fee.vutbr.cz/~arnost/opl/lost+found/index.html> (14.03.2005)

Yamaha Corporation (1988): „Yamaha LSI YMF262 FM Operator Type L3 (OPL3)“
URL: <http://www.fee.vutbr.cz/~arnost/opl/lost+found/index.html> (14.03.2005)

FAQs / Anleitungen zu Spielkonsolen und Soundchips

Arnost, Vladimir (2000): „Programmer's Guide to Yamaha YMF 262/OPL3 FM Music Synthesizer V.1.12“ URL:
<http://www.fee.vutbr.cz/~arnost/opl/opl3.html> (13.03.2005)

Clifford, Paul (o.J.): „PC Engine Programmable Sound Generator“ URL:
<http://www.plasma.demon.co.uk/pcengine/psg.txt> (18.03.2005)

Disch (2004): „FDS Sound“ URL: <http://nesdev.parodius.com/FDS.txt> (18.03.2005)

Ewing, Lance (o.J.): „The 6561 VIC Chip“ URL: <http://www.geocities.com/rmelick/6561.txt> (27.10.2004)

Feldman, Mark (1999): „Programming the PC Speaker“
URL: <http://www.gamedev.net/reference/articles/article442.asp> (28.10.2004)

Frohwein, Jeff (2000): „Game Boy Sound Table“
URL: <http://www.devs.com/gb/files/sndtab.html> (20.04.2005)

Frohwein, Jeff (1998): „Everything you ever wanted to know about GAMEBOY but were afraid to ask“
URL: <http://www.devs.com/gb/files/gbspec.txt> (20.04.2005)

Gevaryahu, Jonathan (2004): „Nintendo(TM) Gameboy sound system (PAPU) guide, v0.99.20“
URL: <http://www.devs.com/gb/files/hosted/GBSOUND.txt> (20.04.2005)

Hall, Gareth (o.J.): „The Bubble Bobble FAQ Version 2.0“
URL: http://www.bubandbob.com/faqs/bubblebobble_hall.txt (25.06.2005)

Horton, Kevin (1999a): „VRCVI Chip Info“ URL: <http://www.tripoint.org/kevtris/nes/vrcvi.txt> (25.10.2004)

Horton, Kevin (1999b): „VRCVII Chip Info“
URL: <http://www.tripoint.org/kevtris/nes/vrcvii.txt> (25.10.2004)

Horton, Kevin (2000): „NES Music Format Spec“
URL: <http://www.tripoint.org/kevtris/nes/nsfspec.txt> (04.08.2004)

Israel, Kirk (1993): „Atari Tech-Specs“ URL: <http://www.alienbill.com/vgames/atari.tech.html> (28.10.2004)

Johnson, Jeremiah (o.J.): „MCK/MML Beginners Guide V. 1.0“
URL: http://www.nullsleep.com/treasure/mck_guide/mck_guide_v1.0.txt (02.08.2004)

Kortink, John (o.J.): „SN76489 sound chip details“
URL: <http://web.inter.nl.net/users/J.Kortink/home/articles/sn76489/> (21.02.2005)

Korth, Martin (2002): „Gameboy Advance Technical Info – Extracted from no\$gba version 1.4 – Hardware Programming – Sound Controller“ URL: <http://www.work.de/nocash/gbatek.htm> (28.10.2004)

Lee, Jeffrey S. (1992): „Programming the AdLib/Sound Blaster FM Music Chips Version 2.0“
URL: http://thorkildsen.no/faqsys/docs/adlib_sb.txt (04.08.2004)

Lorenzen, Erik / Weiner, Keith (1997): „Programming Digitized Sound On the Sound Blaster“
URL: http://www.gamasutra.com/features/19950415/weiner_01.htm (28.10.2004)

Mamiya (o.J.): „NEZplug Version 0.9.4.8“ URL: http://nezplug.sourceforge.net/in_nez.txt (28.10.2004)

Maxim (2003): „SN 76489 notes“ URL: <http://www.smspower.org/maxim/docs/SN76489.txt> (21.02.2005)

N.N. (o.J.): „Intellivions Classic Video Game System“
URL: <http://www.webcom.com/inty/site.shtml> (28.10.2004)

- N.N.** (o.J.): „NES Specifications“ URL: <http://nocash.emubase.de/everynes.htm> (15.06.2005)
- N.N.** (o.J.): „The Qbasic Station – Advanced Tutorial – PLAY“
URL: http://www.geocities.com/qbasicstation/tutor/advanced/a_play.html (02.08.2004)
- Of the Veldt, Gau** (o.J.): „SPC 700 Documentation“
URL: <http://www.classicgaming.com/epr/super/spc.htm> (14.03.2005)
- SEGA** (o.J.): „Genesis Sound Software Manual“ in: „Sega Technical Overview 1.00“, S.87-120
URL: <http://www.classicgaming.com/epr/genesis/segadoc.zip> (14.03.2005)
- Talbot-Watkins, Richard** (1998): „Sega Master System Technical Information“
URL: <http://www.geocities.com/vjkemp/sega/smsinfo.txt> (18.03.2005)
- Taylor, Brad** (2000a): „Delta modulation channel tutorial 1.0“
URL: <http://web.textfiles.com/games/dmc.txt> (27.10.2004)
- Taylor, Brad** (2000b): „The NES sound channel guide 1.8“
URL: <http://web.textfiles.com/games/nessound.txt> (27.10.2004)
- Vawter, Noah** (o.J.): „Home Video Game Console Sound Chip Round-Up compiled by shifty@gweep.net“
URL: <http://www.gweep.net/~shifty/txt/videogamemusic.txt> (27.10.2004)
2004)

Sonstige Sekundärliteratur

- Ackermann, Philipp** (1991): „Computer und Musik“, Wien
- Altmann, Günter** (1968): „Musikalische Formenlehre“, Leipzig
- Berrisch, Sigmar** (1996): „Klänge aus einer fremden Welt. Die Anfänge elektronischer Klangerzeugung“ in: „test-card – Beiträge zur Popgeschichte #3“: Sound, S.150-157. Oppenheim
- Bogdanov, Vladimir et al. (Hg.)** (2001): „All Guide Music to Electronica“, San Fransisco
- Bonz, Jochen** (2001): „Vorwort“ in: ders. (Hg.): „Sound Signatures. Pop-Splitter“, S. 9-16. Frankfurt
- Bristow, David / Chowning, John** (1986): „FM Theory & Applications“, Tokyo
- Büsser, Martin** (1996): „The Art of Noise, The Noise of Art – Eine kleine Geschichte der Sound Culture“ In: „test-card, Beiträge zur Popgeschichte #3: Sound“, S.6-19. Oppenheim
- Großmann, Rolf** (1997a): „Abbild, Simulation, Aktion – Paradigmen der Medienmusik“, Sonderdruck aus Flessner, Bernd (Hg.) „Die Welt im Bild. Wirklichkeit im Zeitalter der Virtualität“, S.239-257. Freiburg
- Großmann, Rolf** (1997b): „Konstruktiv(istisch)e Gedanken zur 'Medienmusik'“, in: Hemker, Thomas / Müllensiefen, Daniel: „Medien, Musik, Mensch“, S.61-77. Hamburg
- Harenberg, Michael** (2003): „Virtuelle Instrumente zwischen Simulation und (De)Konstruktion“ in: Kleiner, Marcus S. / Szepanski, Achim (Hg.): „Soundcultures. Über elektronische und digitale Musik.“ S. 69-93. Frankfurt a. M.
- Harenberg, Michael** (1989): „Neue Musik durch neue Technik ? - Musikcomputer als qualitative Herausforderung für ein neues Denken in der Musik“. Kassel
- Klages, Thorsten** (2000): „Musik und Form – Musik in den (Re-)Produktionsmedien“. Magisterarbeit, Universität Lüneburg, Lüneburg
- Kleiner, Marcus S.** (2003): „Soundcheck“ in: ders. / Szepanski, Achim (Hg.): „Soundcultures. Über elektronische und digitale Musik.“ S. 7-17. Frankfurt a. M.

Levy, Steven (1994): „Hackers. Heroes of the Computer Revolution“, London (OA: 1984)

Ruschkowski, André (1998): „Elektronische Klänge und musikalische Entdeckungen“, Stuttgart

Schlicke, Cornelius (o.J.): „Segmentierung als Grundlage kultureller Praxis. Eine Untersuchung der Musikkultur Techno“ URL: <http://www2.rz.hu-berlin.de/fpm/texte/schlicke.htm> (10.06.2005)

Toynbee, Jason (2000): „Making Popular Music“, New York

Vom Autor geführte Interviews

Hülsbeck, Chris (2005a): telefonisches Interview vom 27.05.2005, Transkription beim Autor

Hülsbeck, Chris (2005b): telefonisches Interview vom 06.06.2005, Transkription beim Autor

Pummell, Michael (2005): telefonisches Interview vom 06.05.2005, Transkription beim Autor

Warhol, David (2005): telefonisches Interview vom 12.05.2005, Transkription beim Autor

Abbildungsverzeichnis

Titelseite:

„Dimensions“ des YM 3526-Soundchips (bearbeitet). URL:

<http://www.funet.fi/pub/cbm/documents/chipdata/ym3526/ym3526-09.png>

S. 10: Abbildung 1

Auszug des Instrumenten-Sets von Slocum (2003).. Quelle: Textdatei „Atari 2600 Music And Sound Programming Guide“ URL: http://qotile.net/files/2600_music_guide.txt (17.02.2005)

S. 15: Abbildung 2

Blockdiagramm des AY-3-8910. Quelle: N.N. (o.J.): „AY-3-8910/8912 Programmable Sound Generator Data Manual“ URL: <http://bulba.at.kz/AY-3-8910.rar> (27.10.2004)

S. 16: Abbildung 3

Das Register-Array des AY-3-8910. Quelle: ebd.

S. 17: Abbildung 4

Differenz zwischen wohltemperierter Stimmung und Stimmung des AY-Soundchips bei einer Taktfrequenz von 2 MHz (Auszug). Quelle: ebd.

S. 17: Abbildung 5 Die verschiedenen Hüllkurven-Typen des AY-Chips in Abhängigkeit zu den in R15 aktivierten Bits. Quelle: ebd.

S. 22: Abbildung 6

Blockdiagramm des SID-Chips. Quelle: **Commodore Semiconductor Group**: „HMOS 6581 Sound Interface Device, preliminary information“, URL: <http://www.funet.fi/pub/cbm/documents/chipdata/6581.zip> (27.10.2004)

S. 23: Abbildung 7

Registertabelle des SID. Quelle: ebd.

S. 47: Abbildung 8

Die verschiedenen Wellenformen des YM 3812. Quelle: **Yamaha Corporation** (o.J.): „Yamaha LSI YM3812 FM Operator Type-L II (OPL II)“ URL: <http://www.funet.fi/pub/msx/mirrors/msx2.com/vortexion/ym3812.pdf> (02.04.2005)

S. 48: Abbildung 9

Die verschiedenen Wellenformen des YMF 262 / OPL3. Quelle: **Yamaha Corporation** (1988): „Yamaha LSI YMF262 FM Operator Type L3 (OPL3)“ URL: <http://www.fee.vutbr.cz/~arnost/opl/lost+found/index.html> (14.03.2005)

S. 59: Abbildung 10

Die einleitende Melodie von Pac-Man. Eigene Darstellung.

S. 70: Abbildung 11

Hintergrundmusik von Bubble Bobble mit Vorspiel (nur Melodie- und Bassstimme). Eigene Darstellung.

S. 71: Abbildung 12

Bonusrundenmusik von Bubble Bobble (nur Melodie- und Bassstimme). Eigene Darstellung.

S. 72: Abbildung 13

Highscore-Musik von Bubble Bobble (nur Melodie- und Bassstimme). Eigene Darstellung.

S. 72: Abbildung 14

Schlusskadenz von Bubble Bobble (nur Melodiestimme). Eigene Darstellung.

S. 73: Abbildung 15

Hintergrundmusik d. versteckten Levels in Bubble Bobble (nur Melodie- und Bassstimme). Eigene Darstellung.

S. 87: Abbildung 16

Die Produktionskette von Soundchip-Musik für Computerspiele. Eigene Darstellung.

S. 89: Abbildung 17

Screenshot von Parodius (Arcade-Version).

Quelle: <http://www.classicgaming.com/reviews/parodius/compare-arcade.png> (27.07.2005)

S. 90: Abbildung 18

Screenshot von Bubble Bobble (Arcade-Version)

Quelle: <http://www.vgmuseum.com/pics4/bubble%20bobble%2007a.gif> (27.07.2005)

S. 91: Abbildung 19

Screenshot von The Secret of Monkey Island (PC-Version, VGA)

Quelle: <http://www.mobygames.com/game/dos/secret-of-monkey-island/screenshots/gameShotId.3162/> (27.07.2005)

S. 106: Abbildung 20

Gegenüberstellung der technischen Verfahren zur Klangerzeugung in Computerspielen und der Entwicklung der Genres von Soundchip-Musik im zeitlichen Rahmen. Eigene Darstellung.

Tabellenverzeichnis

S. 24: Tabelle 1 Die verschiedenen Dauern der Hüllkurven-Phasen in Relation zu den Registerwerten

(vgl. Commodore o.J.:5)

S. 27: Tabelle 2 Die für die Klangerzeugung zuständigen Register des 2A03 (nach Taylor 2000b)

S. 28: Tabelle 3 Die verschiedenen Tondauern, die der jew. Zähler bei einer Frequenz von 60 Hz

aus den Bits 7-3 generiert (nach Taylor 2000b)

S. 32: Tabelle 4 Die verschiedenen PSG-Soundchips im Vergleich

S. 37: Tabelle 5 Die verschiedenen YM-Chips (ausgenommen die OPL-Reihe)

S. 43: Tabelle 6 Die auf Sampling basierenden Soundchips der 16 Bit-Ära im Vergleich

S. 48: Tabelle 7 Die verschiedenen OPL-Chips (mit Ausnahme des YM 2413 und späterer OPL-Versionen)

S. 68: Tabelle 8 Klassifizierung der verschiedenen Musiktypen in Computerspielen

Erwähnte Computerspiele

Sofern nicht anders gekennzeichnet handelt es sich bei den Datumsangaben um das Erscheinungsjahr für die jeweils erste Version. Umsetzungen der Spiele für andere Hardware-Plattformen erscheinen in der Regel mit einigen Jahren Verspätung.

- 1942 (CAPCOM, 1984)
688 ATTACK SUB (ELECTRONIC ARTS, 1989)
ACID DROP (SALU LTD, 1992)
AFTER BURNER II (SEGA, 1990)
ALEX KIDD: THE LOST STARS (SEGA, 1989)
ALTERED BEAST (SEGA, 1988)
AMPLITUDE (SONY, 2003)
ASTEROIDS (ATARI, 1979)
AZTEC CHALLENGE (COSMI, 1984)
BATTLEZONE (ATARI, 1980)
BOMB JACK (TEHKAN, 1984)
BUBBLE BOBBLE (TAITO, 1986)
BUZZ BOMBER (MATTEL, 1983)
CALIFORNIA GAMES (EPYX, 1987)
CARNIVAL (SEGA, 1980)
CASTLEVANIA 3, AKUMAJOU DENSETSU (KONAMI, 1989)
CENTIPEDE (ATARI, 1980)
COMPUTER SPACE (NUTTING ASSOCIATES, 1971)
DARWINIA (INTROSPECTIVE SOFTWARE, 2005)
DIG DUG (NAMCO, 1982)
DONKEY KONGA (NINTENDO, 2004)
DRAGON QUEST (ENIX, 1986)
ELEVATOR ACTION (TAITO, 1983)
FINAL FANTASY (SQUARE, 1987)
FINAL FANTASY III (SQUARE, 1994)
FREQUENCY (SONY, 2001)
FROGGER (KONAMI, 1981)
GAUNTLET II (MINDSCAPE, 1990)
GHOSTBUSTERS (ACTIVISION, 1984)
GHOULS 'N' GHOSTS (CAPCOM, 1989)
GIMMICK (SUNSOFT, 1992)
GOLDEN AXE (SEGA, 1989)
GRADIUS (NEMESIS) (KONAMI, 1985)
GYRUSS (KONAMI, 1983)
JUST BREED (ENIX, 1992)
KING OF KINGS (NAMCO, 1988)
KING'S QUEST (SIERRA, 1984)
KING'S QUEST IV (SIERRA, 1988)
LAGRANGE POINT (KONAMI, 1991)
LEISURE SUIT LARRY (SIERRA, 1987)
LEMMINGS (PSYGNOSIS, 1990, C64: 1993)
LINKS (ACCES, 1990)
LOOM (LUCASFILM GAMES, 1990)
M.A.S.H (FOX VIDEO GAMES, 1982)
MARBLE MADNESS (ATARI, 1984)
MAX PAYNE 2: THE FALL OF MAX PAYNE (ROCKSTAR GAMES, 2003)
METAL GEAR SOLID 2: SONS OF LIBERTY (KONAMI, 2002)
MISSION: IMPOSSIBLE (ACCLAIM, 1998)
MOONWALKER (SEGA, 1990)
MOURYOU SENKI MADARA (KONAMI, 1990)
MR. DO! (UNIVERSAL, 1982)
MR. DO'S CASTLE (UNIVERSAL, 1983)
PAC-MAN (NAMCO, 1980)
PARAPPA THE RAPPER 1 (SONY, 1996)
PARAPPA THE RAPPER 2 (SONY, 2001)
PARASOL STARS (TAITO, 1990)
PARODIUS (KONAMI, 1988)
PARODIUS DA! (KONAMI, 1990)
PINBALL DREAMS (DIGITAL ILLUSIONS, 1992, PC: 1993)
PINBALL FANTASIES (DIGITAL ILLUSIONS, 1992, PC: 1994)
POKÉMON (NINTENDO, 1995)
POLE POSITION (NAMCO, 1982)
PONG (ATARI, 1972)
R-TYPE (IREM, 1987; AMIGA: 1989)
RAINBOW ISLANDS (TAITO, 1987)
RESIDENT EVIL (CAPCOM, AB 1996)
REZ (SEGA, 1999)
SAMBA DE AMIGO (SEGA, 2000)
SHARK! SHARK! (MATTEL ELECTRONICS, 1982)
SHIN 4-JIN UCHI MAHJONG: YAKUMAN TENGOKU (NINTENDO, 1991)

SHOOTING GALLERY (SEGA, 1987)	TEMPEST (ATARI, 1981)
SILENT HILL (KONAMI, AB 1999)	TETRIS (NINTENDO, 1989)
SILPHEED (SIERRA, 1988)	THE GREAT GIANA SISTERS (RAINBOW ARTS, 1987)
SIM CITY (MAXIS, 1989)	THE LEGEND OF ZELDA (NINTENDO, 1986)
SPACE INVADERS (TAITO, 1978)	THE SECRET OF MONKEY ISLAND (LUCASFILM GAMES, 1990)
SPACE QUEST (SIERRA, 1986)	TURRICAN II (RAINBOW ARTS, 1991)
SPLINTER CELL (UBI SOFT, 2002)	TOEJAM & EARL (SEGA, 1991)
STAR WARS (ATARI, 1983)	UNREAL (EPIC MEGAGAMES, 1998)
STREET FIGHTER II (CAPCOM, 1991)	VANGUARD (SNK, 1981)
SUPER HANG-ON (SEGA, 1989)	VIB RIBBON (SONY, 2000)
SUPER MARIO ALLSTARS (NINTENDO, 1993)	WORLD CLASS LEADER BOARD (ACCES, 1988)
SUPER MARIO BROS. (NINTENDO, 1985)	XEVIOUS (NAMCO, 1982)
SUPER MARIO SUNSHINE (NINTENDO, 2002)	ZAK McKRACKEN (LUCASFILM GAMES, 1988)

Hörbeispiele

(Auf die jeweilige Quelle wird verwiesen. Viele der Hörbeispiele wurden auf Basis der entsprechenden ROMs mit Hilfe des Emulators MAME, dem Player WINAMP mit entsprechendem Input-PlugIn und dem Programm SIDPLAY angefertigt)

Nr.	Titel / Spiel	Klangerzeugung
001:	Der 'Pong-Sound' aus PONG (ATARI, 1972)	diskrete Schaltkreise ²⁴⁰
002:	Musik aus ACID DROP (SALU LTD, 1992)	ATARI VCS / Stella ²⁴¹
003:	BGM ²⁴² aus STAR WARS (ATARI, 1983)	4x ATARI Pokey
004:	Spielgeschehen aus STAR WARS (ATARI, 1983)	4x ATARI Pokey / TI 5220
005:	Game Over - Musik aus SHARK! SHARK! (MATTEL ELECTR., 1982)	AY-38914 ²⁴³
006:	Spielgeschehen aus GYRUSS (KONAMI, 1983)	5x AY-38912
007:	Spielgeschehen Level 1 aus BOMB JACK (TEKHAN, 1984)	3x AY-38910
008:	Titelmelodie aus M.A.S.H (FOX VIDEO GAMES, 1982)	SN 76489 AN ²⁴⁴
009:	Chris Hülsbeck: „Shades“ original	MOS SID ²⁴⁵
010:	Chris Hülsbeck: „Shades“ mit korrigierten Filtereinstellungen	MOS SID ²⁴⁵
011:	Skaven: „Catch that Goblin!!“	8 Bit-Samples (SM3) ²⁴⁶
012:	Hintergrundmusiken aus MOONWALKER (SEGA, 1990)	YM 2612 und DAC ²⁴⁷

²⁴⁰ Quelle: <http://www.atariage.com/forums/index.php?act=Attach&type=post&id=19291> (26.07.2005).

²⁴¹ Quelle: http://www.tu-chemnitz.de/phil/hypertexte/gamesound/files/acid_drop.mp3 (26.07.2005).

²⁴² Aus Platzgründen wird „Hintergrundmusik“ bei den Klangbeispielen durch „BGM“ (Background Music) abgekürzt.

²⁴³ Quelle: <http://www.tu-chemnitz.de/phil/hypertexte/gamesound/files/sharkshark.mp3> (26.07.2005).

²⁴⁴ Quelle: <http://www.tu-chemnitz.de/phil/hypertexte/gamesound/files/mash.mp3> (26.07.2005).

²⁴⁵ Die Filter-Probleme lassen sich gut ab 3:39 hören. Quelle: <http://www.hvsc.c64.org/> (26.07.2005).

²⁴⁶ Das Stück wurde 1995 auf dem SCREAM TRACKER 3 (PC) programmiert. Quelle: <http://www.modarchive.com/cgi-bin/download.cgi?C/ctgoblin.s3m> (26.07.2005).

²⁴⁷ Die in dem Klangbeispiel verwendeten Stücke sind allesamt Adaptionen von Michael Jackson - Popsongs: „Bad“, „Beat It“, „Billy Jean“ und „Smooth Criminal“ Quelle: <http://www.zophar.net/gym/moonwalk.rar> (07.08.2004).

013: Titelmusik aus PINBALL FANTASIES (DIGITAL ILLUSIONS, 1994)	PC-Speaker (tweak) ²⁴⁸
014: Titelmusik aus THE SECRET OF MONKEY ISLAND (LUCASFILM GAMES, 1990)	PC-Speaker ²⁴⁹
015: Titelmusik aus STAR WARS: X-WING (LUCAS ARTS, 1993)	ROLAND LAPC-1 ²⁵⁰
016: Belohnungsmusik aus DIG DUG (NAMCO, 1982)	Namco ²⁵¹
017: High-Score-Entry-Musik aus DIG DUG (NAMCO, 1982)	Namco
018: Einleitende Melodie aus PAC-MAN (NAMCO, 1980)	Namco
019: BGM 'Oberwelt' aus SUPER MARIO BROS. (NINTENDO, 1985)	2A03 ²⁵²
020: Einleitende Melodie aus DIG DUG (NAMCO, 1982)	Namco
021: Hintergrundmusik aus CARNIVAL (SEGA, 1980)	AY-38910
022: Spielgeschehen aus FROGGER (KONAMI, 1981)	AY-38910
023: BGM aus KNUCKLE BUSTERS (MELBOURNE HOUSE, 1986)	MOS SID ²⁵³
024: BGM 'Unterwelt' aus SUPER MARIO BROS. (NINTENDO, 1985)	2A03
025: BGM 'Unterwasser' aus SUPER MARIO BROS. (NINTENDO, 1985)	2A03
026: BGM 'Burg' aus SUPER MARIO BROS. (NINTENDO, 1985)	2A03
027: BGM aus DIG DUG (NAMCO, 1982)	Namco
028: BGM aus DIG DUG (NAMCO, 1982) mit adaptivem Element	Namco
029: Sirene und Bonusrunde aus BOMB JACK (TEKHAN, 1984)	3x AY-38910
030: Soundeffekt 'Tod' aus SUPER MARIO BROS. (NINTENDO, 1985)	2A03
031: Soundeffekt 'Game Over' aus SUPER MARIO BROS. (NINTENDO, 1985)	2A03
032: Soundeffekt 'Tod' aus PARODIUS (KONAMI, 1990)	HuC6280
033: Soundeffekt 'Level Complete' aus DIG DUG (NAMCO, 1982)	Namco
034: Soundeffekt 'Level Complete' aus BOMB JACK (TEKHAN, 1984)	3x AY-38910
035: Soundeffekt 'Level Complete' aus SUPER MARIO BROS. (NINTENDO, 1985)	2A03
036: BGM aus AZTEC CHALLENGE (COSMI, 1984)	MOS SID ²⁵⁴
037: Soundeffekt 'hurry up', schnellere BGM und Level Complete-Musik aus RAINBOW ISLANDS (TAITO, 1987)	YM 2151
038: Soundeffekt 'hurry up' und schnellere BGM aus SUPER MARIO BROS. (NINTENDO, 1985)	2A03
039: BGM 'Samba' aus SUPER MARIO BROS. (NINTENDO, 1985)	2A03
040: Soundeffekt 'Automatenstart' aus BUBBLE BOBBLE (TAITO, 1986)	YM 3526 / YM 2203
041: Einleitung aus BUBBLE BOBBLE (TAITO, 1986)	YM 3526 / YM 2203
042: BGM aus BUBBLE BOBBLE (TAITO, 1986)	YM 3526 / YM 2203
043: Soundeffekt 'hurry up' und schnellere BGM aus BUBBLE BOBBLE (TAITO, 1986)	YM 3526 / YM 2203

²⁴⁸ Deutlich hörbar ist die Samplingfrequenz (~1 kHz). Aufgenommen und zur Verfügung gestellt von Stefan Göhler.

²⁴⁹ Komponist: Michael Land, das Beispiel wurde aufgenommen und zur Verfügung gestellt von Stefan Göhler.

²⁵⁰ Die ebenso in den Beispielen 003 und 004 adaptierte STAR WARS-Filmmusik wurde von John Williams komponiert. Aufgenommen und zur Verfügung gestellt von Stefan Göhler.

²⁵¹ Die Klangerzeugung früher NAMCO-Automaten wird in MAME allein als „Namco“ gekennzeichnet, weitere Details über ihre Funktionsweise sind nicht verfügbar. Komponistin: Yuriko Hirono (vgl. Tozuka 1999:17).

²⁵² Die Hörbeispiele aus SUPER MARIO BROS. entstammen der NSF-Datei aus <http://www.zophar.net/nsf/smb1.zip> (07.08.2004).

²⁵³ Komponist: Rob Hubbard. Quelle: <http://www.hvsc.c64.org/> (26.07.2005).

²⁵⁴ Komponist: Paul Norman, Quelle: <http://www.hvsc.c64.org/> (26.07.2005).

044: Soundeffekt 'hurry up' und schnellere BGM im Kontext aus BUBBLE BOBBLE (TAITO, 1986)	YM 3526 / YM 2203
045: Bonusrundenmusik aus BUBBLE BOBBLE (TAITO, 1986)	YM 3526 / YM 2203
046: Soundeffekt 'Extend' und Bonusrundenmusik aus BUBBLE BOBBLE (TAITO, 1986)	YM 3526 / YM 2203
047: High-Score-Musik aus BUBBLE BOBBLE (TAITO, 1986)	YM 3526 / YM 2203
048: Game-Over-Musik aus BUBBLE BOBBLE (TAITO, 1986)	YM 3526 / YM 2203
049: BGM aus THE GREAT GIANA SISTERS (RAINBOW ARTS, 1987)	MOS SID ²⁵⁵
050: BGM aus RAINBOW ISLANDS (TAITO, 1987)	YM 2151
051: BGM 'Oberwelt' von SMB1 aus SUPER MARIO ALLSTARS (NINTENDO, 1993)	SPC 700 ²⁵⁶
052: BGM 'Plattform-Level' aus SUPER MARIO SUNSHINE (NINTENDO, 2002)	NINTENDO GAMECUBE ²⁵⁷
053: BGM aus GHOSTBUSTERS (ACTIVISION, 1984)	MOS SID ²⁵⁸
053b: BGM aus GHOSTBUSTERS (ACTIVISION, 1984)	ATARI Pokey ²⁵⁹
054: Intromusik aus GHOULS 'N' GHOSTS (CAPCOM, 1989)	MOS SID ²⁶⁰
055: Intromusik aus CALIFORNIA GAMES (EPYX, 1987)	MOS SID ²⁶¹
056: Boss-Musik 'Fandarole' aus PARODIUS DA! (ARCADE) (KONAMI, 1990)	YM 2151 / 053260
057: Boss-Musik 'Fandarole' aus PARODIUS (PC-ENGINE) (KONAMI, 1992)	HuC6280 ²⁶²
058: Boss-Musik 'Fandarole' aus PARODIUS (SNES) (KONAMI, 1992)	SPC 700 ²⁶³
059: Boss-Musik 'Fandarole' aus PARODIUS (NES) (KONAMI, 1990)	2A03 ²⁶⁴
060: Boss-Musik 'Fandarole' aus PARODIUS (GAME BOY) (KONAMI, 1991)	DMG-SYSTEM ²⁶⁵
061: Boss-Musik 'Hummelflug' aus PARODIUS DA! (ARCADE) (KONAMI, 1990)	YM 2151 / 053260
062: Boss-Musik 'Hummelflug' aus PARODIUS (PC-ENGINE) (KONAMI, 1992)	HuC6280
063: Boss-Musik 'Hummelflug' aus PARODIUS (SNES) (KONAMI, 1992)	SPC 700
064: Boss-Musik 'Hummelflug' aus PARODIUS (NES) (KONAMI, 1990)	2A03
065: Boss-Musik 'Hummelflug' aus PARODIUS (GAME BOY) (KONAMI, 1991)	DMG-SYSTEM
066: BGM 'Klavierkonzert' aus PARODIUS DA! (ARCADE) (KONAMI, 1990)	YM 2151 / 053260
067: BGM 'Klavierkonzert' aus PARODIUS (PC-ENGINE) (KONAMI, 1992)	HuC6280
068: BGM 'Klavierkonzert' aus PARODIUS (SNES) (KONAMI, 1992)	SPC 700
069: BGM 'Klavierkonzert' aus PARODIUS (NES) (KONAMI, 1990)	2A03
070: BGM 'Klavierkonzert' aus PARODIUS (GAME BOY) (KONAMI, 1991)	DMG-SYSTEM

²⁵⁵ Komponist: Chris Hülsbeck, Quelle: <http://www.hvsc.c64.org/> (26.07.2005).

²⁵⁶ Quelle: <http://www.zophar.net/zsnes/spc/allstars.rar> (26.07.2005).

²⁵⁷ Quelle: [ed2k://file/Super%20Mario%20\(Sunshine\)%20Accapella.mp3|1570657|44E1AA4F30110CE727BFDA452502229B/](http://ed2k://file/Super%20Mario%20(Sunshine)%20Accapella.mp3|1570657|44E1AA4F30110CE727BFDA452502229B/) (26.07.2005).

²⁵⁸ Musik: Adam Bellin, Quelle: <http://www.hvsc.c64.org/> (26.07.2005).

²⁵⁹ Quelle: http://www.tu-chemnitz.de/phil/hypertexte/gamesound/files/ghostbusters_atari.mp3 (29.07.2005)

²⁶⁰ Musik: Tim Follin, Quelle: <http://www.hvsc.c64.org/> (26.07.2005).

²⁶¹ Musik: Chris Grigg, Quelle: <http://www.hvsc.c64.org/> (26.07.2005).

²⁶² Die Hörbeispiele von PARODIUS für PC-ENGINE entstammen der HES-Datei aus <http://www.zophar.net/hes/parodius.zip> (07.08.2004).

²⁶³ Die Hörbeispiele von PARODIUS für SNES entstammen der SPC-Datei aus <http://www.zophar.net/zsnes/spc/parod1.rar> (07.08.2004).

²⁶⁴ Die Hörbeispiele von PARODIUS für NES entstammen der NSF-Datei aus <http://www.zophar.net/nsf/parodius.zip> (07.08.2004).

²⁶⁵ Die Hörbeispiele von PARODIUS für GAME BOY entstammen der GBS-Datei aus <http://www.zophar.net/gbs/parodius.zip> (07.08.2004).

101: Teamtendo: „Got the Tendo (feat Eve)“ von der 12“-EP „We“ (Auszug) (Institubes 2004) ²⁷⁷	
102: Titelmusik aus R-TYPE (IREM, 1987; AMIGA: 1989)	8 Bit-Samples (TFMX)
103: BGM Welt 1 aus TURRICAN II (RAINBOW ARTS, 1991)	8 Bit-Samples (TFMX)
104: „Welcome Great Warrior“ aus GAUNTLET II (MINDSCAPE, 1990)	2A03 (DMC) ²⁷⁸

Internet-Linkverzeichnis

Das folgende Verzeichnis erhebt keinen Anspruch auf Vollständigkeit und ist als Anregung für den Leser gedacht. Alle Quellen, die unmittelbar zum Fliesstext der Arbeit beigetragen haben, sind in diesem genannt bzw. im Literaturverzeichnis enthalten. Daher wird auf eine Kommentierung der folgenden Quellen verzichtet. Alle angegebenen URLs waren am 28.07.2005 verfügbar.

WWW-Archive für Computerspielmusik

Portal-Seite für verschiedene Plattformen
<http://www.mirsoft.info>

AY-Format (ZX SPECTRUM UND AMSTRAD CPC 646):
<http://www.worldofspectrum.org/projectay>

MOD-Formate (COMMODORE AMIGA):
<http://exotica.fix.no/tunes/unexotica/titles/a.html>
<http://www.modarchive.com/>

GBS-Format (GAME BOY):
<http://www.zophar.net/gbs/>

NSF-Format (NES):
<http://www.zophar.net/nsf/>
<http://www.2a03.org>

GYM-Format (SEGA GENESIS / MEGA DRIVE):
<http://www.zophar.net/gym/>

SPC-Format (SNES):

<http://www.zophar.net/zsnes/spc/>
<http://www.snesmusic.org/spcsets/>

MDX-Format (NEC x68000):
<http://xperiment.rainemu.com/x68music.html>

VGM-Format (SEGA MASTER SYSTEM):
<http://www.smspower.org/music/vgm/>

ADLIB/OPL-Formate (PC):
<http://www.chiptune.de>

YM-Format (ATARI ST):
<http://www.chiptune.de>

MID-Format (General Midi):
<http://www.vgmusic.com>

WWW-Archive und -Portale (Demo-Szene, Chiptunes, Micromusic)

Micromusic.net – Community rund um Micromusic und Chiptunes <http://www.micromusic.net>

Chiptune.de (Sammlung von AD LIB- und ATARI ST-Chiptunes) <http://www.chiptune.de>

Pouët.net (Demo-Szene-Community für verschiedene Plattformen) <http://www.pouet.net/>

Amiga Demo Archive (diverse Demos für AMIGA)
 (diverse Cracktros für AMIGA) <http://ada.planet-d.net/>
<http://cracktros.planet-d.net/news.html>

Intros.c64.org (Sammlung von C64-Intros) <http://intros.c64.org/>

Vorc.org (jp./en. Portal für Chiptunes) <http://www.vorc.org>

Scene Music – Info (Portal für Scene-Music) <http://noerror.scene.org/>

.MOD in memoriam – MOD-Dateien der Tracker-Szene <http://www.mono211.com/st-00/>

²⁷⁷ Quelle: <http://institubes.com/images/team-1.mp3> (26.07.2005).

²⁷⁸ Quelle: <http://www.tu-chemnitz.de/phil/hypertexte/gamesound/files/gauntlet2.mp3> (01.08.2005).

Amiga Music Preservation – Informationen zu über 8000 MOD-Komponisten etc.

<http://amp.dascene.net/home.php>

tracking.atari.org – Portal rund um Atari St – Musik

<http://tracking.atari.org/>

Emulation, Player etc.

Audio Overload – Player für über 20 verschiedene Formate (Mac, PC & Linux)

<http://bannister.org/software/ao.htm>

WinAmp – populärer MP3-Player für Windows, über Input-PlugIns für viele Formate nutzbar

<http://winamp.com>

Dazu: Diverse Input-PlugIns für WinAmp

<http://www.winamp.com/plugins/browse.php?sortby=&shownum=20&ctype=P&category=6>

Zophar's Domain – Emulation und VGM (siehe Archive oben)

<http://www.zophar.net/>

KbMedia Player – Japanischer Player für über 50 verschiedene Soundchip-Musik-Formate

(Englisch-Patch verfügbar)

<http://home7.highway.ne.jp/Kobarin/>

Deliplayer – Player für über 200 Chiptunes- und MOD-Formate

<http://www.deliplayer.com>

Renoise – Umfangreicher Tracker für Windows XP mit VST, Asio etc. Unterstützung:

<http://www.renoise.com>

Emulation Number 9 – Portal mit Neuigkeiten und umfangreicher Sammlung div. Emulatoren

<http://www.emulation9.com/>

Künstler-Homepages (Micromusic und moderne Chiptunes)

(in Klammern: Klangerzeuger und Herkunftsland)

6955 (GAME BOY, Kanada)

<http://come.to/6955>

8bitweapon (C64, GAME BOY, INTELLIVISION u.a., USA)

<http://www.8bitweapon.com>

Alexi Eben (VIC 20, GAME BOY, C64, AMIGA)

<http://www.cncd.fi/aebeben/>

AY Riders (ZX SPECTRUM u.a., Polen)

<http://ayriders.zxdemo.org/>

Bit Shifter (GAME BOY, USA)

<http://www.bitshifter.cc/>

Bodenständig 2000 (ATARI ST, C64 u.a., Deutschland)

<http://bodenstaendig.de/2000/>

Casionova (div. Casios, C64, GAME BOY, ATARI VCS,

Australien) <http://www.casionova.com>

Chrim and Fenris (C64 u.a., Australien)

<http://www.sciencegirlrecords.com/cf/>

Covox (GAME BOY, Schweden)

<http://www.covox.net/>

Cow'p (aka Com.b) (GAME BOY, Japan)

<http://www.19-t.com/comb/>

Desert Planet (diverse, Finnland)

<http://www.desertplanet.com>

Drop da Bomb (ATARI ST, Schweiz)

<http://www.dropdabomb.org/>

FirestARTer (C64, GAME BOY u.a., Deutschland)

<http://www.firestarter-music.de/>

Gameboyzz Orchestra (GAME BOYS, Polen)

<http://gameboyzz.com/>

GNG (diverse, Frankreich)

<http://www.gngmusic.com/>

Goto80 (C64, AMIGA, GAME BOY, Schweden)

<http://www.goto80.com>

Ht (diverse, Schweden)

<http://www.goto80.com/ht>

Midr2 (NES, GAME BOY, Japan)

<http://midr2.under.jp/>

Lektrogirl (Game Boy, div., Großbritannien)

<http://www.lektrogirl.com>

Lithis (diverse, Schweden)

<http://lithis.x-dump.com/>

Lo-Bat (GAME BOY, AMIGA, Belgien)

<http://www.lo-bat.be/>

Nullsleep (Game Boy, Nes, USA)

<http://www.8bitpeoples.com/nullsleep>

Paul Slocum (ATARI VCS, diverse, USA)

<http://qotile.net/>

Paza (ATARI ST, div. modif. Synths, Schweden)

<http://paza.x-dump.com/>

Plemo (diverse, Deutschland)

<http://plemo.com/>

Psilodump (diverse, Schweden)

<http://www.psilodump.com/>

Rolemodel (AMIGA, GAME BOY, Schweden)

<http://www.rebelpetset.com>

Rugar and I (GAME BOY, NES u.a., Schweden)

<http://www.rugarandi.com/>

Teamtendo (GAME BOY, Toys, Frankreich)

<http://www.teamtendo.com>

Tobiah (GAME BOY, C64, Schweden)

<http://www.tobiah.tk/>

Trash80 (GAME BOY, diverse, USA)

<http://www.trash80.net/>

Virt (Tracker, USA)

<http://virt.vgmix.com/index.php>

Welle:Erdball (C64 u.a., Deutschland)

<http://www.welle-erdball.de/>

YMCK (GAME BOY, NES, Japan)

<http://www.ymck.net/>

YM Rockerz (ATARI ST, diverse, Deutschland)

<http://www.creamhq.de/ymrockerz/>

Webradios

Nectarine – Webradio für Scene-Music

<http://www.scenemusic.net/>

Retro PC GAME MUSIC Streaming Radio – Spielmusik der japanischen Personal Computer

<http://gyusyabu.ddo.jp/MP3/MP3.html>

Kohina – Webradio für Soundchip-Musik

<http://www.kohina.com/>

Erklärung nach § 25 (6) der Magisterprüfungsordnung

Der Unterzeichnende versichert, dass er die vorliegende Magisterarbeit komplett selbstständig verfasst und keine anderen als die von ihm angegebenen Hilfsmittel benutzt hat. Die Stellen der Arbeit, die anderen Werken dem Wortlaut oder dem Sinne nach entnommen sind, wurden in jedem Fall unter Angabe der Quellen (einschließlich des World Wide Web) kenntlich gemacht. Dies gilt auch für beigegebene Zeichnungen, bildliche Darstellungen, Skizzen und dergleichen.

Lüneburg, den 04.08.2005

Danke:
Meinen Eltern, Nina, Jan.

Den Korrektoren für ihre konstruktive Kritik:
Winnie Forster, Heiko Gogolin, Christian Kühnel, Andreas Otto,
Tobias Ruderer und Mike Witschi.

Für Anregungen und Material:
Anders Carlsson, Pero Fotar, Stefan Göhler, Johan Kotlinski, Jukka Tarkiainen

Den Interviewpartnern Chris Hülsbeck, Michael Pummell und David Warhol,
ohne deren Mithilfe diese Arbeit nicht hätte entstehen können.

Außerdem: dem Rechen- und Medienzentrum der Universität Lüneburg,
insbesondere dem Projekt Ästhetische Strategien ((audio))