Chapter 10

Chip music: low-tech data music sharing

Anders Carlsson[1]

Ever since I was very young, I have been using Commodore computers to make music. As a teenager I fiercely defended low-tech computing in the face of more user-friendly systems. This purism was put to the test as I started collaborating with people from different backgrounds: when the Spanish designer duo Entter produced my first video clip, they showed an IBM PC instead of what should have been the Commodore 64. This was hard for me to accept, since the old computers were the basis of my music and had important symbolic value to me as an alternative to contemporary technology. Today I am not such a purist, but to me the question remains: how important is it to use 'authentic' technology when making music?

Nowadays, it is common to see influences from old video games within popular culture. In media publications this re-use is often explained by the fact that during the past 25 years, children in the Western world have grown up with rapidly evolving video games. Consequently, these games play important roles in their childhood memories. But there is another reason for the recent nostalgia: the growth of file sharing. Digital subcultures have been spreading free software on an international level since there were home computers with modems in the mid 1980s. This spreading started out with the distribution of home-made applications as well as commercial software and evolved into the spreading of audio-visual presentations, and more recently stand-alone music. This musical style has become known as 'chip music', music composed by using, emulating or sampling old digital sound chips. At present there are thousands of chip music songs available on the Internet for free. In many cases this music is presented in an open-source format, including the source of the music, which can be changed or adapted by the user. This practice of open sharing distinguishes parts of the chip music scene from most other music genres and scenes.[2]

To the uninitiated, chip music is often perceived as homogenized, mostly due to its distinctive sound, and therefore is often categorized as one specific musical genre. The artistic freedom, based on the technological constraints of old chips, has, however, produced 'bleepy' music that sounds very different from the songs of the 1980s' video games, as today's composers are influenced by more contemporary

[1] I would like to thank Rosa Menkman for assistance, and also 4-mat, Autoboy, Aleksi Eeben, Linus Walleij, Johan Kotlinski, Rambones and Julian Van Aalderen.

[2] Here I define 'scene' as an 'active creation of infrastructure to support … bands and other forms of creative activity'. Alan O'Connor, 'Local Scenes and Dangerous Crossroads: Punk and theories of Cultural Hybridity' *Popular Music* 21/2 (2002), p. 226.

musical genres and are not obliged to make the songs fit into a game. In this sense, it could be argued that the technological minimalism of chip music is the most significant defining element of the music: no matter what is done with a sound chip in terms of harmony, structure or rhythm, it is still chip music. There is, however, a difference between regarding chip music as medium and chip music as form. I will discuss this difference below.

The purpose of this chapter is to discuss chip music with a historical perspective based in the 'demoscene', a subculture of creative computer users submitting their work to competitions with audio-visual productions programmed to be generated in real time for the viewer. Since today's chip music scene is essentially non-commercial, the commercial industry of video games is out of the scope of this text.[3] There is not much literature on either the demoscene or chip music, and so much of this text is based on my own experiences from the northern European music and demoscenes of the Commodore 64 and Amiga.

The demoscene

The demoscene grew out of the movement of removing copy protection from games, commonly known as 'cracking'. This view of making software freely available for the public has roots in both hacker ethics and free software movements. Parts of the cracker community have ideals similar to those of hackers and phreakers[4], but it seems likely that a lot of early crackers were kids cracking games for fun, for intellectual challenge and for 'street credibility'.[5] The European cracking scene (from out of which the demoscene formed) stems from the 1979 American Apple II and the Atari 800 cracker scene, which was linked to Europe through modems and the BBS (Bulletin Board Service) culture.[6] Hackers and phreakers were, however, generally more 'underground' than the game-crackers, in the sense that the nicknames of crackers would appear on hundreds of thousands of screens while kids were loading the cracked games. Although most people did not know or care who 'Mr.Z' or 'Dynamic Duo' were, these names would, for some, represent mysterious heroes who provided them with countless hours of free entertainment. For the crackers (usually boys in their early teens), this was an exciting way of obtaining notoriety. Crackers would compete to make the best cracks and to be first to have a game released on an American BBS for international use. This activity required both programming skills and the ability to make free international calls.

The cracking scene has been compared to graffiti culture mainly because the two both involve illegally invading corporate and public space in a search for identity or

[3] Chip music has been used in more commercial purposes such as with Malcolm McLaren's launch of chip music as the new 8–bit punk, and will be discussed more later on in the chapter.

[4] Linus Walleij, 'Copyright Finns Inte' (1998). Online English Version, Chapter 5 http://home.c2i.net/nirgendwo/cdne/mainindex.htm (accessed 12 February 2007).

[5] Lassi Tasajärvi, *Demoscene: The Art of Real-Time* (Finland, 2004), p. 12.

[6] Walleij, 'Copyright Finns Inte', Chapter 5.

fame.[7] Initially the crackers' name tags (nicknames) started as basic text messages but gradually included more complex programming and graphics, and became known simply as 'crack-intros' or 'intros'. Around 1985, crack-intros were released separately from the games and were made famous within the scene by groups such as 1001 Crew and The Judges, both from the Netherlands.[8] By this time, the demoscene was starting to form in Western Europe, although it was connected to the one in Australia, and to lesser extent those in North America and Eastern European countries.[9] The key demoscene computers were those of Commodore, Atari and Sinclair – very similar to those now part of the chip music scene, although today the USA and Japan have a lot more prominence, and Nintendo products have become popular.

A 'demo' (abbreviated from 'demonstration') demonstrated what a user could do with a specific technology. Everything seen and heard would be generated by the computer in real time and running at full frame rate. Demos can be understood in opposition to animation that consists of a big file of pre-generated data that is exactly the same every time you play it. Originally, animations were not used in demos because they used too much memory. Later on in the demoscene, as programmers had memory to spare for animations, real-time generated audio and visuals were still used and it became an important convention in the demoscene.

Tasajärvi uses a metaphor to explain how a theatre director instructs the actor who then performs the script live, as opposed to a film that has a final fixed product.[10] A demo, a theatre play and a pure chip music song are normally generated as the listener is experiencing it, rather than being pre-recorded. Just as a theatre director can play with the rules of what theatre is supposed to be, the demosceners and chip music composers found ways to work beyond technological boundaries thought to be unbreakable. Demosceners would, for example, launch new graphic modes to expand resolutions and colours, work around bugs in the sound chips, experiment with samples and expand the computers' possibilities in general. These tricks were not rational mathematical programming, but rather trial and error based on in-depth knowledge of the computers used. Demosceners managed to accomplish things not intended by the designers of the computers.

During the 1980s the programming was more important than the graphics and music in demos, but by the 1990s so-called 'design demos' started appearing, a term which implies how much the programming dominated the previous demos. These more recent demos were based on graphical and musical concepts, rather than maximizing the hardware through programming. The Danish/French group Melon Dezign has been credited with starting this movement in the Amiga demoscene in the early 1990s. As for the music, there would be a growing gap between bleepy chip

[7] Tasajärvi, *Demoscene: The Art of Real-Time*, p. 31. Sky Frostenson, 'Bombing the System', Winter ACE Seminar: Virtual Identities (2004). Walleij, 'Copyright Finns Inte', Chapter 5.

[8] Támás Polgár, *Freax: The Brief History of the Computer Demoscene* (Winnenden, Germany, 2006), p. 57.

[9] Polgár, *Freax: The Brief History of the Computer Demoscene*, pp. 58–68.

[10] Tasajärvi, *Demoscene: The Art of Real-Time*, p. 17.

music and songs that sounded more like studio music. The studio-oriented music would typically sound like jazz funk or techno, a unique style of the demoscene that became known as 'doskpop'. Swedish artist Lizardking was the father of doskpop, which can be described as soft digital disco music, somewhere in between 1980s electronic disco music such as Laserdance and Vangelis.[11]

In the 1980s most home computer set-ups were identical, but in the 1990s the market for hardware expansion grew bigger. This meant that one Amiga set-up could be so different from another set-up that you could not run the same demos; a problem most notable in the IBM PC demoscene. This meant that demos had to be more system-friendly to work on different set-ups of CPUs, memory and graphic and sound cards. Traditional demoscene norms of making the computer do 'impossible' things would weaken on the PC as possibilities expanded. Some demo groups would start using Windows' 3D software or MP3 files in their demos, stepping away from the older norms of the demoscene towards music videos. However, certain people would stick with tradition to only use default hardware and programming everything themselves – which became known as 'dogma-demos'.[12] With demos increasingly taking the form of music videos, the gap between the demoscene and the major music industry decreased. Demoscene composers began to appear on large record labels, such as Brothom States on Warp and Bogdan Raczynski on the Rephlex label.

The many improvements in technology were one of the reasons some demoscene artists returned to the older machines, like the Commodore 64 and ZX Spectrum. In addition, the Nintendo Entertainment System (NES), Super Nintendo (SNES) and Game Boy also gained popularity as new developments in hardware and software for hobbyists appeared. Although the ever-present demoscenes of the 8-bit machines remain underground today, the PC demoscene has gained popular recognition, as has chip music in general.

A brief history of chip music

Computer-based music took its first steps in the 1950s, although analogue electronic music already had a 50 year head start.[13] In 1951 the Australian mathematician Geoff Hill was the first to generate music in real time.[14] In 1957 Max Mathews began programming music software for the IBM 704 in the USA. During the next decades digital music was experimented with by pioneering multimedia artists, *musique concrète* composers and scientists, and began to appear in video games such as Atari's *Pong* (1972).[15] By 1975 a computer was playing 'Daisy Bell', realized by having an Altair 8800 manipulating a nearby AM radio, which was made by the

[11] Polgár, *Freax: The Brief History of the Computer Demoscene*, p. 134.

[12] Tasajärvi, *Demoscene: The Art of Real-Time*, p. 68.

[13] See Peter Manning, *Electronic and Computer Music* (Oxford, 2004), pp. 19–100.

[14] Paul Doornbusch, *The Music of CSIRAC: Australia's First Computer Music (Altona, Australia, 2005), Online version* http://www.csse.unimelb.edu.au/dept/about/csirac/music/introduction.html (accessed 19 December 2006).

[15] Karen Collins, 'From Bits to Hits: Video Games Music Changes its Tune', *Film International* 13 (2004), pp. 4–19.

hobbyists in the Homebrew Computer Club[16] and has been described as the world's first demo.[17] Although the generated sounds were not digital and it is therefore problematic to describe the song as a demo, this example is of interest here since it is an early example of hobby users playing music with a computer.

In the 1970s video games and arcade games music was not a primary concern for manufacturers. During the 1980s, however, this sentiment changed, making sounds and music more important for interactive game-play.[18] During the 1980s all of the music in video games, computer games and arcade games used basic sounding chips, meaning all computer music was 'chip music'.[19] From quite early on there was music software available on home PCs, such as *Commodore Music Maker* (1982) for the Commodore 64, using traditional notation techniques. However, these programs were not efficient enough to be used for games and demos, and composers basically had to be programmers as well because of the need to use assembler or even machine code (basically only a list of numbers) to make instruments and sequence notes.[20] The first popular music program for games and demo music on the Commodore 64 was *Soundmonitor*, made by Chris Hülsbeck in 1986. Instead of using a programming language to make music, it became possible to use this interface to make music. Although some artists still preferred the low-level composing techniques which offered more flexibility and efficiency, such as the game composer Rob Hubbard and the demoscener Laxity, *Soundmonitor* proved to be an important step towards the most popular tool for making chip music: the tracker format.

In *Soundmonitor*, a composer did not have an overview of a song, since she or he had to switch between the song arrangement screen and the pattern editor where the notes were sequenced. The composer would write small sequences of notes with his or her instruments of choice, and could add portamento, arpeggio or transpositions. This note and effect data was written straight into a specific place in the memory. For example, the computer could store a 32-byte sequence (including more than just note data) between B200 and B21F in the memory, and then point to it in the song arrangement screen with B200. All of the numbers in *Soundmonitor* were written in hexadecimal form, which remains common among trackers even today.

In 1987 Karsten Obarski released *Soundtracker* which turned the Commodore Amiga into a very price-worthy tool for composers, using 8-bit samples up to 29 kHz. It was disassembled and improved upon to be re-released by numerous programmers in the demoscene, the most popular releases being *Noisetracker* (1989) and *Protracker* (1990). With these releases, it was possible to use 32 samples and sequence them in a more user-friendly tracker style, adding effects in an effect

[16] Sheet Music/BASIC listing of 'Fool on the Hill' played on the Altair 8800 computer, Digibarn Computer Museum, http://www.digibarn.com/collections/weirdstuff/altair–sheetmusic/index.html (accessed 19 December 2006).

[17] Polgár, *Freax: The Brief History of the Computer Demoscene*, p. 41.

[18] Collins 'From Bits to Hits: Video Games Music Changes its Tune' p. 6.

[19] 'Chiptune', *VORC Internet Chiptune Encyclopedia*, http://www.vorc.org/en/info=Chiptune (accessed 15 May 2007).

[20] Thomas Egeskov Petersen, 'Music Recollection', *Music Recollection Issue 2*, http://www.atlantis-prophecy.org/recollection/?load=online&issue=1&sub=article&id=11 (accessed 15 May 2007).

column right next to the notes. The songs created with these trackers were called 'modules' or 'mod-files' for short. These trackers were the most popular form for composition in the demoscene, and were also used in the game industry to create music for games. Two disks of sampled sounds were provided with *Soundtracker* – known as ST-01 and ST-02 – and were used so frequently that they created a distinct aesthetic in the sound of Amiga.

In 1989 the British composer 4-mat, along with contemporaries such as Turtle and Duz, started using sample-based trackers to make chip music. Previously, the bleepy chip music style had been made with waveforms straight off the sound chip (as with C64 or Spectrum), or with software synthesized waveforms (as with *Future Composer* on the Amiga), but the new technique meant taking samples and cutting them up into tiny segments of 200 bytes or less, and then looping them. This would generate a tone, which would sound different depending on the waveform of the sample, but basically sounding like different kinds of completely static 'bleeps'. On computers such as the Commodore 64, however, the tone was generated in real time by the chip itself, and the sounds could be modulated to sound different every time they were played. In an Amiga *Protracker* chip song, these dynamics would instead be programmed by having several similar bleep sounds that the composer would rapidly switch between, or would add effects to in the patterns. Effects such as pitchbend, portamento, arpeggio, vibrato and volume envelopes were entered straight into the pattern data, which made it easier than the effect handling on the Commodore 64 trackers. The sample-based chip music, then, offered a different way of creating sounds for chip music.

When releasing songs in games or demos with the tracker format, it was possible to look into the memory to see how the song worked in terms of the routine that defines how to access the sound chip, and the key was to tweak the chip to its fullest. It was therefore useful to look into the code to see how artists would produce sounds. One programmer known as 'Predator', for instance, made a tracker based on the code of games composer Rob Hubbard. When releasing a song, it was possible for users to access the player routines, along with instruments, note data, effects and samples. Even though these elements could be hidden with various techniques, there would usually be a way to hack into the code, which is why it is possible to find most Commodore 64 games songs released online today (for instance, in the High Voltage SID Collection). As long as a user has the tracker that was used for making the song, it is possible to load the song into an editor and gain total control of the song. This practice has been more heavily used on the Amiga since the mod format was more standardized, and it was possible to access the actual sample data. Today, however, chip music is not often released in its original format, but rather as an inadaptable MP3.

In the 1990s the most popular trackers in the demoscene were *Fasttracker* on the PC and *Protracker* on the Amiga, although *Little Sound DJ* (*LSDJ*) is probably the tracker that has gained most attention outside of the demoscene. *LSDJ* is a music program for the Game Boy in which patterns are individual for each track, much like *Soundmonitor*, but which has a new way of creating chains of patterns that could be then arranged in a song view. This is why the creator Johan Kotlinski 'doesn't really see it as a tracker'.[21] *Little Sound DJ* brought trackers into the mainstream and the Game Boy came to be a popular symbol for the art of chip music composing and performance. *LSDJ* offered more possibilities than any tracker before, both in arranging the song and in editing notes, effects and instruments in real time.

Although composing with old sound chips is usually accomplished with trackers today, there are other options. In addition to traditional notation software, there are alternatives for trackers in text-based format like *Prophet64* (C64), *Nanoloop* (Game Boy) and *Midines* (NES). *MCK* is one program to make NES music, and is arguably the most popular text-based composing tool today, especially in Japan. Today, non-purist chip music is often made with modern software like *Cubase* and *Reason* with VST instruments emulating the old sound chips. There are also modern trackers like *Renoise* that combine the *Soundtracker* tradition with more modern approaches such as MIDI and VST. Some chip music purists, preferring trackers running on the original hardware, condemn this practice and prefer not to label these compositions chip music, leading to a difficulty in defining the genre.

Defining chip music

Although I have chosen to use the term 'chip music' throughout this text, alternatives to this term exist, such as 'chiptune' and '8-bit' with several lesser-known subgenres such as 'bitcore' and 'chiphop'. 'Chiptune' was probably first used in the Amiga demoscene in the early 1990s, and is used today as widely as 'chip music'. 8-bit is more specific to music being made with 8-bit technology, which I will discuss further below. Sometimes this music is referred to as 'micromusic' and 'bitpop'. The term 'micromusic' stems from the community micromusic.net and usually refers to music made with home computers or video game consoles that is more or less inspired by 'old school' computer sounds. Bitpop is similar, but includes the use of analogue synthesizers, and is even more oriented towards poppy dance music than other styles. Chip music, on the other hand, is a broader term which can refer both to music made with old sound chips and bleepy sounding music made with other hardware.

Chip music is often defined by technology, which is to say that chip music is often defined as all music made with old sound chips. The problem with this distinction is that, generally speaking, it can be hard to tell the difference between simple waveforms being played by a 1980s sound chip or a modern synthesizer or emulator. A strict technological definition also brings the problem of defining which sound chips are acceptable to the community. There are, for instance, sound chips in toys, alarm clocks, mobile phones, doorbells and Game Boy Advance, and then there is also General MIDI and FM synthesis. Which of these would be included in a technological definition of chip music? One option would be to only accept sound chips up to those of the 16-bit era, but most sound chips are not easily defined as 4-, 8- or 16-bit, since different parts of the chip works with different resolutions. For

[21] Tasajärvi, *Demoscene: The Art of Real-Time*, p. 43.

example, the Commodore 64's SID chip has 16-bit envelope control and 16-bit pitch register, although the chip is typically referred to as 8-bit.

As chip music has gained popularity, it has influenced and been influenced by other kinds of music, making it hard to maintain a purist technological definition. Even as the term was popularized on the Amiga, it referred to sample-based music, as mentioned above: It was not about the medium, but rather the form the composition took. However, *Protracker* on the Amiga was also used to make music very far from bleepy C64 nostalgia. In the Dutch gabber scene there were, for example, *Protracker EP* (1993) and *The Three Amiga's EP* (1993) by Neophyte. In Australia, Nasenbluten and Xylocaine released hardcore *Protracker* songs on records, and Digital Hardcore Recordings in Germany had artists such as Patric Catani (under various pseudonyms), eq8or and Christoph de Babalon who used similar trackers. Although these artists used the tracker format, their music is not generally considered chip music. It is chip music as a medium, but not chip music as form. In the 1990s sample-based chip music became more a conscious choice of style rather than a direct consequence of the hardware and software used to create it. Although chip music songs were originally used in the demoscene due to their small file sizes, there would soon grow a small scene of dedicated chip music creators and listeners.

Some composers have more recently turned away from the technologically imposed sounds to impress the listener. In the C64 demoscene programmers and composers would evolve the sounds with new tricks in order to beat the competition, and on the Amiga the capability of sampling led to some music resembling studio recorded music with, for example, the works of Audiomonster or Moby. However, the chip music composers on the Amiga were often embracing the (arguably) simplistic sounds of the C64 and its games, giving it a value and aesthetic in and of itself. For the most part, more recent chip songs have been short and happy-sounding loops in 4/4, often flirting with C64 music from the 1980s,[22] but composers have used chip sounds to make songs sounding like jazz, noise, death metal or even hip-hop.

Chip music after 2000

Chip music today has grown in popularity and in style. One important part of the evolution of chip music has been the website micromusic.net, formed in 1999 and still very much alive. It offers a selection of user-uploaded songs merging old and new computer sounds, usually dance-oriented, but also some quite experimental songs. The website has become an important name in the blossoming media coverage of 'chipoid' music, but with founders from the subversive Internet art scene, it has effectively stayed clear of labels and generalizations. Through the website, chip music has started appearing on stages mainly around Europe. Some of the artists that would make pure chip music popular outside of the demoscene in Europe are Rolemodel, Gwem, Lo-Bat, Firestarter, Teamtendo, Bodenständig 2000, Puss (the latter being nominated for a Swedish Grammy in 2003) and Goto80. In North America, artists such as 8-bit Construction Set, Bitshifter and Nullsleep, along with the 8bitpeoples

[22] Personal e-mail correspondence with 4-mat, December 2006.

label have become important figures. Japan's chip music scene has also gained notoriety, with artists such as Cow'P and Hex125.[23] Despite the popularity, chip music also remains tied to its origins, with some cracked software for PC having spread composers like Maktone once again into the homes of computer users.

Fuelled by a growing popularity of 1980s culture, chip music has enjoyed success far beyond video games and demos. In 2003 punk pioneer Malcolm McLaren wrote an article in *Wired* magazine that would bring chip music even more into the media spotlight. After having worked with fashion and music projects like the Sex Pistols, he then launched a view of chip music as the new 8-bit punk. Chip music was described as a subversive and cheap way of making Game Boy music for vinyl releases, deliberately staying underground.[24] His statements upset a lot of chip music people, who feared he would commercialize the music, as described by micromusic.net.[25] Although McLaren's hype of chip music led to positive attention for a lot of artists, his outsider perspective also cultivated some unfortunate misunderstandings, such as the claim that subversive chip music composers hacked into video games to make music.[26] This is a rare part of the chip music scene, as most chip music composers simply copy a program that someone else has made and use it to make music. Chip music composers can, however, be called subversive in the sense that they do perhaps 'abuse' corporate technology by not using it for its intended purpose. The software is usually developed by hobbyists in the demoscene with the intent of maximizing and reaching beyond the hardware's limits, and then disseminating it free of charge. Some of the machines used by chip music artists were not even meant to be musical devices. Although some chip music artists do have ideological or artistic motives, these are not usually outwardly expressed in the music. One of the problems for McLaren would have been to generalize chip music. Judging from media coverage and online chip music archives, the dance and pop approaches are in the majority, but especially after 2000, chip music has been merged with a lot of genres and instruments. Some of the more prominent approaches of adding traditional instruments and vocals to chip music come from David Sugar, Mark DeNardo, Bodenständig 2000, Bud Melvin, Anamanaguchi, Gwem and Superdöner. The noisier chip music, often released on low bit-rate Internet labels such as North American Hardcore or 20kbps Rec, has some common ground with the *Protracker* hardcore music in the 1990s. Some artists are either by choice or 'accident' not really a part of the traditional chip music forums, as for example Patric Catani or the frequently touring DJ Scotch Egg.

There are also major commercial artists incorporating the sounds of chip music into their songs, such as Beck's 'Hell Yes' (2005) and also Nelly Furtado's song 'Do

[23] 'VGM or Chiptune of the Year 2002', *VORC*, http://www.vorc.org/en/columns/hally/2002best10e.html (accessed 5 June 2007).

[24] '8-Bit Punk', *Wired Magazine* 11/11 (November 2003), http://www.wired.com/wired/archive/11.11/mclaren.html (accessed 15 May 2007).

[25] 'Open Letter to Malcolm McLaren', *Micromusic* (2004), http://micromusic.net/public_letter_gwEm.html (accessed 15 May 2007).

[26] Peter Culshaw, 'So I pitched my Oscar Wilde film to Spielberg', *Guardian Unlimited*, 21 March 2004, http://arts.guardian.co.uk/features/story/0,11710,1175408,00.html (cited 15 May 2007).

It' (2006), produced by Timbaland, which ran into trouble after it sampled a song by the Scandinavian demosceners GRG and Tempest without credit. It has also been my experience that various TV channels and radio channels have used my own music without credit. The question I asked myself after these incidents was: is it more wrong when a big business company does it? To me this is an interesting conflict with chip music. These examples and other uses of chip music without credit to the original authors have raised discussions about how chip music composers could protect their work from illegal use. The question of free distribution is undoubtedly something to keep in mind when speaking about the future distribution of chip music. Chip music has its roots in the demoscene where music was traditionally shared in open-source formats and extracted in parts, but as the music is now available for any Internet user, the question is how to handle 'unlawful' use of chip music especially by mainstream artists and companies. In the discussions on the forums at micromusic.net, it seems that most people defend the traditional copyright laws and not the demoscene style of sharing: especially not when there is money involved. Although chip music artists do not necessarily want to make more money from their work, with the growing possibility of more commercial activities, there might arise a conflict between the ideologically driven file-sharers and culture-jammers on the one side, and more traditional record-releasing artists on the other. Compromises have been reached, such as with the American label 8bitpeoples, which sells music but also offers it free for download.

Another interesting thought is what will happen to chip music when kids are no longer raised with bleepy video games. It leads to the question of whether the digitally primitive interfaces and sounds have something unique to offer, or if it is just a nostalgic fad. This brings me back to my initial question of how much importance one should put on using 'authentic' hardware when making music and what consequences it has on the creative processes. To me, starting as a purist chip music composer, I have gradually learned to appreciate chip music as something other than a result of the medium. Ultimately the music is more interesting than the technology, and there have been very exciting recent developments in chip music. However, I also strongly believe that composing on old computers and consoles gets the composer closer to the machine, which has an effect on the music. There is something beautiful about learning how to use the so-called constraints of old technology to expand creativity. It would therefore be interesting to see some more research on how chip music tools are used and how the use of these tools affects the realization of musical ideas.

Chapter 11

Left in the dark: playing computer games with the sound turned off

Kristine Jørgensen

In audio-visual contexts we often get the impression that the use of audio is merely ornamental and present only for the purpose of supporting a specific atmosphere. Although this is an important and interesting feature of audio in such contexts, it is only one of several roles that audio may have. Audio may also work to provide specific information about a setting and situation, and may have direct influence on actions and events in the environment. In the context of computer games, audio has clear usability functions in addition to supporting a specific mood and the sense of presence in the game environment. This chapter will take the role of game audio into account by demonstrating how playing without any sound affects the game experience, both on the level of the game environment and on the level of the game system.

Recently, game audio has become a hot commodity in the game industry. This is demonstrated by an increased awareness during the last couple of years of the need to implement audio in games in innovative ways. One example is the *Grand Theft Auto 3* games in which the player may change in-game radio stations when driving around in the city, and another is how the Xbox 360 game console allows the player to add music from his or her computer hard drive into the game she/he is currently playing. This may be seen as a symptom of the golden rule in game audio design, which is to never let the player become annoyed or bored by the repetitiveness of the sound. By allowing the player to be in charge of the music, the developer may avoid players turning off the music intended for the game. In the context of this chapter, however, the question is whether (and if so, how) removing the implemented game audio influences the game experience. I will argue that taking away the sound has consequences for the player's orientation and awareness in the game world. In addition, certain kinds of information are harder to grasp when sound is removed. The argument is based on qualitative studies of empirical game players' understanding of game audio in context, and how they experience the game without any sound present. Since theories specifically aimed towards game audio at the time of writing are scarce, the empirical data will be further supported by auditory display studies and film theory. Auditory display studies will emphasize game audio as a usability feature, while film theory underlines game audio as a support for the game environment.

In audio-visual contexts it is often believed that sound and image work as two complementary information systems that compete in the meaning making process.