

## MIT Open Access Articles

*Two Pioneering Projects from the Early History  
of Computer-Aided Algorithmic Composition*

The MIT Faculty has made this article openly available. **Please share**  
how this access benefits you. Your story matters.

**Citation:** Ariza, Christopher. "Two Pioneering Projects from the Early History of Computer-Aided Algorithmic Composition." *Computer Music Journal* 35.3 (2011): 40-56. Web. 20 Jan. 2012. © 2011 Massachusetts Institute of Technology

**As Published:** [http://dx.doi.org/10.1162/COMJ\\_a\\_00068](http://dx.doi.org/10.1162/COMJ_a_00068)

**Publisher:** MIT Press

**Persistent URL:** <http://hdl.handle.net/1721.1/68626>

**Version:** Final published version: final published article, as it appeared in a journal, conference proceedings, or other formally published context

**Terms of Use:** Article is made available in accordance with the publisher's policy and may be subject to US copyright law. Please refer to the publisher's site for terms of use.



---

**Christopher Ariza**

Massachusetts Institute of Technology  
Music and Theater Arts, 4-246  
77 Massachusetts Avenue  
Cambridge, Massachusetts  
02139-4307 USA  
ariza@flexatone.net

## Two Pioneering Projects from the Early History of Computer-Aided Algorithmic Composition

Lejaren Hiller's 1970 chapter, "Music Composed with Computers: An Historical Survey" (Hiller 1970) contains numerous descriptions of projects in the computer generation of musical structures. By then, just over ten years after the publication of Hiller's and Leonard Isaacson's seminal book *Experimental Music* (Hiller and Isaacson 1959), a startling number of experiments in generative music with early computers had been completed. Hiller's early research, compositions, and publications established him as a leader in the then-emerging field of computer-aided algorithmic composition (CAAC). Some researchers, likely inspired by Hiller and Isaacson's 1956 *Illiad Suite* string quartet, even duplicated their instrumentation: in an amusing footnote, Hiller writes that "it is curious to note how many computer pieces have been written for string quartet . . . particularly since string-quartet performers seem to be among the least receptive to newer compositional ideas such as computer music" (Hiller 1970, p. 70).

Hiller's prominence in a young and decentralized field led many independent researchers to contact him directly. He became, it seems, a repository of information (often obtained only through direct correspondence). Although Hiller's chapter reports on numerous projects that are now well known and independently documented, buried in the historical survey are a few remarkable endeavors of significant historical importance that have not hitherto been fully documented, corroborated, or heard. This article brings forth two of these projects, the work of David Caplin and Dietrich Prinz, and the work of Sister (then Mother) Harriet Padberg. In his chapter, Hiller describes both of these projects primarily through letters he received from the authors.

Both projects were pioneering: they began with little or no knowledge of other applications of computers to music generation, and they explored

new techniques. Further, each was likely a historical first. The work of Caplin and Prinz in 1955 may be the first use of a computer to generate not just sound (as was done as early as 1950 or 1951; see Doornbusch 2004), but new musical structures. The work of Padberg in the early 1960s may be the first academic dissertation in CAAC, as well as the first research and software in CAAC published by a woman.

This article not only details the historical context and implementation of these projects for the first time, but offers original recordings and new realizations of these early computer-aided compositions. These audio materials are available on-line at [www.flexatone.net/docs/cmj35-3.zip](http://www.flexatone.net/docs/cmj35-3.zip), and will also be included on the *Computer Music Journal* Sound and Video Anthology DVD accompanying the next issue (35:4, Winter 2011).

This research would not have been possible without the continuing contributions of the original authors; numerous personal correspondences with both Caplin and Padberg have provided critical details on their projects. It is a great fortune that, through Hiller's desire to document the work completed in the field, and through the willingness of Caplin and Padberg to revisit and discuss work they completed a half-century ago, we can today listen to some of the earliest attempts at composing music with a computer.

### The First Ten Years of Computer-Aided Algorithmic Composition

The work of Hiller and Isaacson, leading to the completion of the *Illiad Suite* string quartet, is often credited as the first extensive use of a computer to create music with algorithms (Dodge and Jerse 1997, p. 373). As Hiller frequently notes, however, there were earlier experiments. The most well-documented of these experiments is the song *Push*

---

*Button Bertha*. This piece was the result of programming by Douglas Bolitho and Martin L. Klein, with lyrics written by Jack Owens (Anonymous 1956a, 1956b, 1956c; Darreg 1957; Klein 1957; Pierce 1961; Hiller 1970; Ames 1987). On 15 July 1956, *Push Button Bertha* aired on the television program “Adventure Tomorrow,” less than one month before 9 August 1956, the date Hiller staged a performance of an incomplete *Illiatic Suite* at the University of Illinois (Hiller and Isaacson 1959). The *Illiatic Suite* was not completed until November, three months later. The work of Caplin and Prinz in 1955 thus preceded both the *Illiatic Suite* and *Push Button Bertha*.

From 1955 to 1965 there were numerous experiments in CAAC. In addition to more well-known projects, such as Hiller and Robert Baker’s MUSIC SIMulator-Interpreter for COMpositional Procedures (MUSICOMP; Baker 1963; Hiller 1967, 1969), Gottfried Michael Koenig’s Project 1 (first tested in 1964; Koenig 1970), and Iannis Xenakis’s Stochastic Music Program (Xenakis 1965), there appear to be at least 20 other independent efforts at CAAC within this period (Ariza 2005a).

The work of Caplin and Prinz, as well as that of Padberg, is found at opposite boundaries of this ten-year period. Using computers, they implemented and extended well-known approaches to procedural or algorithmic composition that had formerly been carried out without computers. These techniques and their historical origins will be subsequently explained in depth. Caplin implemented the famous dice games attributed to W. A. Mozart; Padberg employed serial techniques along with a text-to-pitch mapping procedure strikingly similar to Guido of Arezzo’s (ca. 991–1033) vowel-to-pitch mapping routines defined in *Micrologus* (Guido of Arezzo 1978 [ca. 1026]). Both, however, quickly extended these historical approaches toward more idiomatic designs, recognizing the new and powerful opportunities available with computer implementation.

## The Work of David Caplin and Dietrich Prinz

On 21 September 1960 Hiller received a letter from Caplin describing computer programs, written by

himself and Prinz, that created music using two distinct methods. The first approach employed the combination of precomposed melodic fragments based on the dice games attributed to Mozart (Hiller 1970; Gardner 1974; Hedges 1978). The second approach built melodic lines with conditional probabilities. Hiller, in *Experimental Music* (Hiller and Isaacson 1959), mentions this research, citing as his source an earlier, though undated, letter from R. J. Lunbeck.

Caplin was well suited to design a computer music system in 1955: he had experience with music, had worked with some of the earliest computers, and had abundant access to a newly installed machine in his workplace. Caplin was born in 1927 in Leeds, England. Although lacking formal music training, he studied music independently and began composing at the age of nine. He entered the London School of Economics (LSE) in 1944 at age 17. In his last two years at the LSE he specialized in mathematical statistics, and notes as particularly influential seminars given by Karl Popper. Upon leaving the LSE he was drafted into the office of the Scientific Advisor to the Air Ministry. There, he began work in what was then called operational research, building mathematical models of real-world problems. In 1950 he was hired by Petrocarbon Ltd. (and its operating arm Petrochemicals Ltd.), and began to apply operational research in the petrochemical industry. Daunted by the time it took to solve problems, Caplin looked to the nearby Ferranti Mark I computer, delivered to Manchester University in 1951 (Tweedale 1993), for more efficient solutions. He then began studying programming. According to Caplin, it was Prinz, a member of Ferranti’s instrument department since 1947 (UK National Archive for the History of Computing 2005), who originally taught him how to program (Caplin 2008). Petrochemicals Ltd. was later acquired in 1955 by Shell Chemical, the company at which Caplin continued his work both in London and Amsterdam (Caplin 2008).

Starting around September 1955 Caplin, then based in London, would commute from London to the Koninklijke/Shell-Laboratorium in Amsterdam (KSLA, now called the Shell Technology Centre Amsterdam). Caplin, placed in charge of economics and planning, developed linear programming

---

routines on the KSLA's new computer, a Ferranti Mark I\* (the star is an indication of a model with revised hardware). Yet, when time permitted, he explored applications of the computer to music. This work involved not only the generation of musical structures, but direct synthesis of elementary sounds (Lunbeck 2005a; Caplin 2008). While Hiller and Isaacson, describing these experiments, state that "the coding of this problem was carried out by D. A. Caplin" (Hiller and Isaacson 1959, p. 56), the programs were a collaboration: Prinz authored the sound-generating code and Caplin authored the musical models.

The first experiments of Caplin and Prinz predate the published work of both Hiller and Isaacson and Klein and Bolitho, and provide the earliest documented experiments in CAAC. As Hiller (1970, p. 47) notes, this research "dates back to 1955" and "ranks among the earliest attempts at computer composition." Based exclusively on the information in Hiller and Isaacson (1959) and Hiller (1970), many others have cited Caplin's early experiments (Cohen 1962; Cope 1991; Assayag 1998; Rivest 1999). Interestingly, although Caplin participated in Jasia Reichardt's 1968 Institute of Contemporary Arts exhibit "Cybernetic Serendipity" (Reichardt 1968; MacGregor 2002), there have been no other independent acknowledgments of his work in music.

As a peripheral activity of their primary research, these studies were not formally documented (Lunbeck 2005b): as noted previously, the only written information is known through a letter sent to Hiller by Lunbeck prior to 1959 (Hiller and Isaacson 1959), a letter sent to Hiller by Caplin in 1960 (Hiller 1970, p. 47), and, in the process of researching this article, correspondence with Lunbeck in 2005 and Caplin in 2008 and 2009. Caplin has recently recovered recordings, dating from 1955 and 1959, of original output from his early systems; these recordings are for the first time being made available as part of this article.

### **The Ferranti Mark I\* at the KSLA**

In 1949 Ferranti Ltd., a large and then well-known UK electronics manufacturing firm, established

a computer group and began development of the Ferranti Mark I (Tweedale 1993). The Mark I, generally considered the first digital, stored-program computer made commercially available (Moreau 1984; Cambell-Kelly and Aspray 1996), was based on a computer developed at Manchester University in the late 1940s called the Manchester Mark 1 or the Small-Scale Experimental Machine (SSEM; Napper 2005). Ferranti sold a total of nine Mark I computers. The first two were delivered to Manchester University and the University of Toronto in 1951 and 1952, respectively (Tweedale 1993). The remaining seven were "slightly modified" (Williams 1997) models, their name designated with an asterisk: the Mark I\* (Napper 2005). In 1953 Shell Chemical ordered the fourth Mark I\*, paying a reported £95,000 (Tweedale 1993). This computer was installed at the KSLA in 1953, and officially commissioned in 1955 (Koninklijke/Shell-Laboratorium 1989, p. 150; Lunbeck 2005b). The KSLA Mark I\* was given the name MIRACLE; although the letters were said to abbreviate Mokum's Industrial Research Automatic Calculator for Laboratory and Engineering (using a nickname for Amsterdam), others offered that the name stood for "May It Replace All Chaotic Laboratory Experiments" (R. 1954).

Prinz had significant programming experience: he is the author of a text titled *Introduction to Programming on the Manchester Electronic Digital Computer Made by Ferranti Ltd.* (n.d.; likely published in 1951). During this time he oversaw the installation of the MIRACLE at the KSLA (Caplin 2008), as well as the seventh Ferranti Mark I\*, installed at the Istituto Nazionale per le Applicazioni del Calcolo in Rome (Tweedale 1993; De Marco et al. 1999).

The Mark I machines were built with some 4,000 vacuum tubes, 12,000 resistors, and 2,500 capacitors; the machines had electrostatic core memory providing 256 40-bit words, drum memory providing 16,384 words, and instruction times of 1.2 msec for addition and 2.15 msec for multiplication (Williams 1997; Napper 2000). Perhaps more important to early endeavors in computer music, however, is that these machines, like the Council for Scientific and Industrial Research Automatic Computer (CSIRAC) and other early computers (Doornbusch 2004, 2005),

---

had an integrated loudspeaker. These loudspeakers were often called hooters, and the command to trigger them was a “hoot” instruction.

Alan Turing, in his *Programmers' Handbook for the Manchester Electronic Computer Mark II* (Turing 1951), provides instructions on how to generate sound with the Manchester/Ferranti lineage of computers. This lineage is not completely clear from the nomenclature Turing uses: he wrote the first edition of his handbook between the dismantling of the Manchester Mark 1/SSEM and the delivery of the Ferranti Mark I; Turing's use of “Mark II” refers to the Ferranti Mark I (Napper 2005). The second edition of the *Programmers' Handbook for the Manchester Electronic Computer Mark II*, revised by R. A. Brooker, was updated for the Ferranti Mark I\* (Turing and Brooker 1952).

Turing states that when “the hoot” instruction code (“/V”) is given, “an impulse is applied to the diaphragm of a loudspeaker” (Turing 1951, p. 26). He goes on to describe ways of using looped impulses at various rates to produce frequencies; for example, how impulses separated by 1.92 msec will produce a 521-Hz tone (Turing 1951, p. 26). Turing provides the code to generate the 521-Hz tone, as well as code for a loop to generate a tone a fifth lower, and he describes how impulses applied to the loudspeaker can also be used to create “something between a tap, a click, and a thump” (Turing 1951, p. 26). His description suggests that the sound-production method used on the Ferranti Mark I was identical to that of the CSIRAC (Doornbusch 2004, p. 16). Caplin confirms that he and Prinz knew of this method of generating sound from Turing's handbook (Caplin 2008). At the time of his research on CSIRAC, Doornbusch (2004, p. 12) knew of “no documentation about the Ferranti Mark I's means of producing music”; but Turing's handbook, which is now available on-line, offers clear documentation.

Some of the earliest computer-generated sound, and the oldest known recording of music produced by a computer, is the 1951 recording of Manchester University's Ferranti Mark I performing monophonic renditions of *God Save the Queen*, *Baa Baa Black Sheep*, and *In the Mood* (Computer Music Journal 2004; Doornbusch 2004, 2005; Fildes 2008). Although there is evidence to suggest that the

CSIRAC produced musical sound as early as 1950, these endeavors were not recorded (Doornbusch 2004). The British Library Sound Archive item 0493W C2 (and related item H3942), from 1951, is titled “Computer Programme to Play Music Written by Christopher Strachey under the Tutelage of Alan Turing,” and credits the performer as “Ferranti Mk 1 computer.” This item is described in Doornbusch (2004), was re-released on the 2004 *Computer Music Journal* DVD, and was heralded as the “oldest” computer music (Fildes 2008). Although Strachey wrote the program, the elementary method of synthesis seems to be the same as that described by Turing the same year (Turing 1951). As documented in an extended *CMJ* Editor's note, this 1951 recording is accompanied by a 1994 introduction, recorded by Frank Cooper, an engineer for the Manchester Ferranti Mark I. Cooper confirms that “Alan Turing was in charge of the programmers, and had in fact written a programmer's manual,” and that Turing himself sent a copy of his manual to Christopher Strachey with an invitation to write a program for the Ferranti Mark I (Computer Music Journal 2004, p. 126). Cooper goes on to describe how, after Strachey's initial endeavors, “everybody got interested” in writing music programs (Computer Music Journal 2004, p. 127).

At the KSLA, the hoot instruction on the MIRACLE was refined by Prinz into a reusable playing routine. This routine was then used to perform the output of Caplin's music-generating programs. Hiller (1970, p. 48) quotes Caplin:

The Mozart dice game was originally programmed for the Ferranti Mark [I\*] computer which had a ‘hoot’ instruction on it. This meant that we could write loops corresponding to each note of the chromatic scale and then play tunes which were ‘composed.’ We were then able to make a tape recording of the computer composing and then playing a tune.

These are the recordings Caplin has recovered; see the present article's sound examples.

Prinz's playing routine is described in his *Introduction to Programming on the Manchester Electronic Digital Computer Made by Ferranti Ltd* (n.d., p. 20). Under “Electronic Instructions”



Figure 1. A photograph, taken around 1953, of the operator's desk of the MIRACLE, a Ferranti Mark I\* computer used by David Caplin and Dietrich Prinz to generate music in

1955. At the left of the desk is the input device, which read five-hole teleprinter tape. The panel of three rows of switches, at the center of the desk, could be used to specify

functions or feed data to various storage or output destinations. Above this panel, on either side, are two cathode ray tubes. At the right of the desk is the teleprinter tape output

(Caplin 2011). Photo courtesy of Onno Zweers and Lidy Zweers-De Ronde.



Prinz describes how, through the use of the hoot command, “a series of periodically repeated pulses is obtained so that an audible ‘hoot’ is emitted”; further, “the pitch of the hoot can be varied by the programmer, and this facility may be used to give an immediate indication as to which of various possible events has occurred” (Prinz n.d., p. 20). Prinz suggests using variations in hoot pitch as a practical means of giving feedback to the user.

In addition to performing newly generated compositions, the MIRACLE was used to perform fixed compositions. Numerous sources confirm that *Het Wilhelmus*, the Dutch national anthem, was encoded and played back by the MIRACLE (Rooijendijk 2007, p. 96). This synthesis of the Dutch national anthem with the Mark I\* in Amsterdam was parallel to, but perhaps uncoordinated with, the earlier synthesis of the British national anthem with the Mark I in Manchester. In an interview, Lidy Zweers-De Ronde, one of the early programmers of the MIRACLE, describes how the machine was used to play short melodies, including *Het Wilhelmus* (Eskens 2002). These and related programs were known at the KSLA as “visitors’ programs”: software demonstrations for visitors to the KSLA computer room. Other visitors’ programs included random number

generators, calendar programs, and prime number analyzers (Lunbeck 2005a). Photographs of the MIRACLE, taken in the 1950s, are provided in Figures 1 and 2.

### **An Implementation of the Mozart *Musikalisches Würfelspiel***

Sometime in 1955 Prinz obtained a copy of the Mozart *Musikalisches Würfelspiel*, or dice game. While he had already created programs to synthesize sound on the Ferranti Mark I\*, he lacked the musical knowledge to implement the dice game, so he turned to Caplin for assistance.

The Mozart *Musikalisches Würfelspiel* is one of at least twenty musical dice games common in the late 18th and early 19th centuries (Hedges 1978). The version attributed to Mozart was first published in 1793 (two years after his death) by Johan Julius Hummel (Mozart 1793), and it includes two similar games: one for minuets and another for contredances. Although numerous additional printings and translations followed (Gardner 1974, p. 133), it appears that all subsequent editions derived from the Hummel publication (Hedges

Figure 2. A photograph, taken around 1953, of the internal components of the MIRACLE. Photo courtesy of Onno Zweers and Lidy Zweers-De Ronde.



1978). One of the most widely distributed modern editions is published by B. Schott (Mozart 1956); the date of that publication, however, precludes use by Caplin and Prinz.

When composing with this tool, two eight-bar homophonic phrases are created. The music is given as two-part keyboard music, with harmony in the left hand and the melody in the right hand. Dice are used to select from 176 pre-composed measures. For each bar, two dice are thrown, summing to 11 unique values (integers 2 through 12). This number is used to select one of 11 pre-composed measures assigned to each bar position; for 14 of the 16 bar positions, there are 11 possible measures. The diversity of options is reduced, and proper cadences are assured, by having bars 8 and 16 offer only one pre-composed option for each respective position (Gardner 1974).

Caplin reports that Prinz obtained a copy of the *Musikalisches Würfelspiel* from the Hirsch Collection, then part of the British Museum in London. An item in the British Library, attributed to W. A. Mozart, titled *Anleitung Englische Contretänze mit zwei Würfeln zu componiren so viele man will, ohne etwas von der Musik oder der Composition zu verstehen*, and published by Nikolaus Simrock in Bonn in 1798 (Mozart 1798;

Hedges 1978), is found in the British Library with shelf mark Hirsch IV.237, indicating its inclusion in the Hirsch Collection. It is likely that this item, specifying the composition of English contredances (*Englische Contretänze*), is the work implemented by Caplin and Prinz. Although other editions of the *Musikalisches Würfelspiel* are also found in the British Library, including those specifying the composition of minuets, Caplin's recordings are clearly in duple meter, confirming the use of the English contredance games.

Caplin's composition system, ignoring the left-hand harmonies, employed only the melodic lines from the measure-length segments in the published edition of the *Musikalisches Würfelspiel*. As the computer was not fast enough to generate high frequencies with the hoot command, the melodic lines were transposed down an octave. Each note was assigned two parameters: one for pitch and the other for duration. As the magnitude of the impulse supplied to the loudspeaker could not be varied, dynamics were not possible with the synthesis technique employed (Caplin 2008).

These one-measure segments of note data, after being stored, were recombined into new compositions, employing the guidelines of the *Musikalisches*

---

*Würfelspiel* and the Ferranti Mark I\*'s random number generator. Output was provided by hoot-based synthesis and, optionally, printing or displaying the resulting compositions as symbolic alphanumeric data (Caplin 2008).

It is fitting that one of the best known early examples of algorithmic composition, the *Musikalisches Würfelspiel*, was implemented as what may be the first CAAC system. The availability of both a basic synthesis technique and a random number generator in the computer, as well as the positive benefits of impressing guests with "vistor programs," led to this project's success. While Caplin and Prinz could have generated endless random melodies, the implementation of Mozart's *Musikalisches Würfelspiel* offered a connection to historical practice and a suggestion of musical legitimacy. This convincing demonstration of new sounds and techniques with old music bears some similarity to Wendy Carlos's 1968 recording *Switched-On Bach*, in which the music of J. S. Bach was, for the first time, widely heard on a Moog synthesizer.

Recognizing further opportunities for using a computer and constrained randomness to generate musical structures, Caplin and Prinz went on to explore more sophisticated and creative compositional routines.

### Melodic Lines from Transitional Probabilities

Hiller, based on his correspondence with Caplin, reports that additional experiments were conducted by Caplin and Prinz until at least 1960, using the Ferranti Mark I\* in early 1956, and then the much more powerful Ferranti Mercury computer in 1959. The Mercury was a significant improvement over the Mark I\*: a Ferranti brochure states that the machine offered instruction times of 180 microseconds for addition and 300 microseconds for multiplication, as well as an expanded core memory of 1,024 words (Ferranti Ltd. 1956). Caplin states that the Mozart *Musikalisches Würfelspiel* program, after being rewritten for the Ferranti Mercury, was 20 times faster than the same program on the Ferranti Mark I\* (Hiller 1970).

The source material for this new composition system was the song *The Foggy, Foggy Dew*. The

chord progressions from this song were used to derive a harmonic progression and harmonic rhythm. For beats 1 and 3 of each 4/4 bar, pitches were randomly selected from the appropriate chord. Pitches at other metric positions were selected using weighted random selection of transitional probabilities, where probabilities for a pitch at position  $n$  was based on the pitches at positions  $n - 1$  and  $n + 1$ . Caplin manually entered the probabilities for these transitions. Hiller describes how pitches "were selected from tables using transitional probabilities" (Hiller 1970, p. 47). This approach is similar to a first-order Markov chain, yet employs both past and future events to select intervening states. Rhythms were algorithmically generated "according to some fairly simple rules" (Hiller 1970, p. 47): random selection from a fixed collection of bar-length rhythms. Caplin notes that their "first set of experiments bore some resemblance" (Hiller 1970, p. 47) to Hiller's earliest work on the *Illiac Suite*.

Caplin was familiar with Markov chains: he had employed them in his statistical models and was aware of creative applications such as the use of Markov chains to generate text, as presented in *A Mathematical Theory of Communication* (Shannon and Weaver 1949). Caplin describes how his probabilistic approach came more from "a statistical approach to time series analysis" (Caplin 2008). These motivations were similar to those of Hiller, who as early as 1956 states that he had been investigating "the application of certain techniques of probability theory for selecting successive melodic intervals" (Hiller 1956, p. 248). In 1958 Hiller and Isaacson further describe their approach to "Markoff chain music" as used in the *Illiac Suite*: transition probabilities "can be made to apply simply to successive melodic intervals, or more complexly in relation to a 'tonal center'" (Hiller and Isaacson 1958, p. 156). By selecting notes between chord tones derived from a fixed harmonic background, Caplin arrived at a similar approach.

Output from this later system was again provided in both symbolic alphanumeric form and in direct sound synthesis. Hiller and Caplin report of only one performance by humans of the symbolic music output. While describing the results as "rather dull tunes of the sort which one used to hear in Victorian



---

hotel lounges," an output, titled *Berolina*, was taken by Prinz to a computer conference in Darmstadt, where it was "...hotted up by the dance band in the hotel where he was staying and was given a warm reception at its first performance" (Hiller 1970, p. 48). Caplin estimates that this conference was in 1956 or 1957 (Caplin 2008).

### Recordings of the Ferranti Mark I\* and Mercury

In 1955 Caplin, with the assistance of Lunbeck, made a recording of the Ferranti Mark I\* performing his Mozart Dice Game system. This recording was made by holding a microphone near the computer's integrated loudspeaker and recording to a reel-to-reel analog recorder (Caplin 2009d). The occasional high-frequency noises in the recording are a result of the random signal generation; the occasional stutters in the sound, Caplin suggests, are due to the time necessary to transfer data from the magnetic drum storage to the fast access storage (Caplin 2009a). This recording is Audio Example 1 of the present article.

In 1959 Caplin commissioned Elizabeth Innes, a programmer in the Shell Computer Development Division, to rewrite the 1955 Mozart Dice Game System and playing routine for the Ferranti Mercury; the output of this computer was recorded using a similar method (Caplin 2009d). This recording contains frequent clicks due to the starting and stopping of the recorder (Caplin 2009a). This is Audio Example 2.

These two recordings were stored on analog reels until the late 1990s, when Caplin transferred them to analog cassette (Caplin 2009b). In the spring of 2009, Caplin digitized the analog cassette recordings to a personal computer (Caplin 2009c). The digital files provided by Caplin have been trimmed and normalized by this author; no noise reduction or other audio processes have been applied.

### The Work of Harriet Padberg

Sometime in the early 1960s, less than ten years after Caplin and Prinz's first experiments, Hiller

began a correspondence with Harriet Padberg (Hiller 1970), a pioneering researcher whose work in computer music employed techniques and approaches later to become widespread. Padberg was completing her dissertation in mathematics and music: she was then developing, and by 1964 had completed, a computer-based system for algorithmically generating canons and what she called free fugues (Padberg 1964).

Padberg's 1964 dissertation, *Computer-Composed Canon and Free Fugue*, is an original and thoroughly researched approach to CAAC. Her dissertation provides significant background to her research, implementation details and complete Fortran code, and score tables of complete compositions.

Padberg's extensive studies in music (including the organ) and mathematics, along with a spiritual dedication to the principles of education and a contemplative life, provided a foundation for her brief work in computer music. Padberg was born in 1922 in Saint Louis, Missouri. She earned a Bachelor of Arts degree from Maryville College in 1943. Shortly after earning this degree, she entered the novitiate of the Society of the Sacred Heart in Albany, New York. For much of the rest of the 1940s, she taught music and mathematics at numerous Academies of the Sacred Heart in locations around the United States. While continuing with her organ studies, in the late 1940s she began and completed a Masters of Music degree in musicology from the Cincinnati Conservatory of Music. By 1956 she had received a Masters of Arts degree in Mathematics from Saint Louis University, focusing on studies in statistics. After the accumulation of three degrees and extensive teaching experience, she began her PhD research in mathematics and music at Saint Louis University in 1959, the same year Hiller and Isaacson published *Experimental Music*.

Padberg set out to create a computer system to generate microtonal music. Remarkably, she completed all of the research and programming for this system without any access to sonic realizations. She describes how, with the assistance of students, she would fill glass bottles with water to try to audition the 24-tone scale employed in the compositions (Padberg 2008). Hiller reports that

---

Max Mathews “converted” these compositions into sound (Hiller 1970); Padberg adds that this took place in the summer of 1968 (Padberg 2008), and Mathews suspects that Music V was used to synthesize the output (Mathews 2009). Padberg gave these recordings of Mathews’s realization to the archive of the United States Province of the Society of the Sacred Heart. As the complete score tables are included in the dissertation, new transcriptions and realizations of her compositions have been completed by the present author for this article.

### Musical Context and Composition Methods

Padberg was familiar with *Experimental Music* (Hiller and Isaacson 1959), as well as the procedural models and analytical methods of Pinkerton (1956), Brooks et al. (1957), Moles (1960), Fucks (1962a, 1962b), and others. Padberg’s research preceded the publication of both Xenakis’s Stochastic Music Program (Xenakis 1965) and Koenig’s research with Project 1 and Project 2 (Koenig 1970).

Padberg frames her musical aesthetic within the context of 20th-century European and American serialism, citing the procedures of Arnold Schoenberg, Anton Webern, Karlheinz Stockhausen, and Milton Babbitt (Padberg 1964). She specifically cites the use of transformed or mapped text as a source of melodic content, noting the well-known musical “signature” motives of J. S. Bach, Anton Webern, and Robert Schumann (Padberg 1964, pp. 19–20). This approach to mapping natural language to musical parameters is critical to her programs. Padberg is, however, careful to note that “it is evident that no exclusive use of mechanical devices, no matter how logical these may be, is sufficient to produce a masterpiece in any medium” (Padberg 1964, p. 20).

Padberg does not, however, mention Guido of Arezzo’s vowel-to-pitch mapping system defined in *Micrologus* (Guido of Arezzo 1974 [ca. 1026]). That approach, instead of mapping each character, maps each vowel of each syllable to a pitch; this technique, described many centuries before the *Musikalisches Würfelspiel*, is frequently cited as the first documented method of algorithmic composition (Roads 1985).

Padberg programmed two systems, both in Fortran, and both printed in whole in her dissertation. The first, for the IBM 1620, permitted the creation of canons in two or four voices based on a single row. Due to the limitations of the IBM 1620, only one row per canon was employed (Padberg 1964, p. 24). The second system, for the IBM 7072, expands upon the first system and permits canons in two or four voices using from one to three rows, and adds the ability or create a “free-fugue” in two voices using from one to three rows. All tone rows were limited in length to between 5 and 24 pitches (Padberg 1964). Padberg did not use random number generation in either system; instead, she used a natural-language source text as a variable source of parameter mappings for pitch and rhythm generation.

The differences between the two computers Padberg used are significant. The computer she used first, the IBM 1620, was introduced in 1959. This machine was intended as a “small, scientific computer” for industrial process control and related applications (Bashe et al. 1986). As a cost- and power-saving design, the system had no arithmetic unit and no general data registers, instead using stored arithmetic tables for all math (Bashe et al. 1986). The system had “an extremely limited instruction set that nevertheless could be programmed to do a lot of useful work” (Ceruzzi 2003, p. 218). In terms of performance, this machine offered instruction times of 56 microseconds for addition and 496 microseconds for multiplication (Moreau 1984), making it faster than the Ferranti Mercury in some aspects and slower in others.

The computer Padberg used second, the IBM 7072, was introduced in 1962 as an improved version of the IBM 7070. The 7070, introduced in 1960, was the first transistorized computer from IBM and was designed to be a “large business computer” (Ceruzzi 2003). Offering significantly more flexibility and power with an expanded instruction set, performance was in some aspects comparable to the IBM 1620, offering instruction times of 72 microseconds for addition and 924 microseconds for multiplication (Moreau 1984). In this case, however, the machine did arithmetic with logic circuits rather than stored tables.

Padberg's system started by generating pitch sequences from a source text input. The characters of the source text are translated into a tone row and then, in some cases, transformed with the serial operations of retrograde, inversion, and retrograde inversion. Padberg used a direct mapping from letter to 24-tone pitch scale. The mapping is one to one, except for two cases. The first exception handles the letter Y. Conceptually, the letter Y, when used as a vowel, is mapped as the letter I; when used as a consonant, the letter Y is mapped as the letter Z. Padberg generalizes this concept by mapping Y to I on its first appearance and to Z on subsequent appearances. The second exception maps V and W to the same pitch (Padberg 1964).

Repeated letters are handled in a special manner, where each pitch mapped from a repeat is shifted one or more octaves up or down, depending on the number of repeats (Padberg 1964). For example, the first repeat transposes up an octave; the second down an octave; the third up two octaves. Noting that repeated letters have a specific function within words in the English language, Padberg states that in the resulting musical lines the "contrast of registers and the extreme angularity of a melody . . . indicates the continual repetition of certain letters" (1964, p. 38).

The 24-tone scale is not an equal division of the octave but is based on partials 24 through 47 of the harmonic series. Formed within one octave, Padberg repeats these tones over nine octaves (1964, p. 42). She cites many reasons for the use of a 24-tone microtonal scale, including near correspondence with the number of letters in the alphabet and an increasing compositional interest in microtones "with the advent of electronic music and the possibility of producing every shade in the spectrum of sound" (Padberg 1964, p. 29). She also notes that her scale shares intervals with Just intonation and Pythagorean tuning (Padberg 1964). Table 1 illustrates frequencies and approximate pitches of Padberg's 24-tone scale.

### Program 1

The first system completed by Padberg, Program 1, was written for the IBM 1620. The input text,

**Table 1. Harriet Padberg's Non-Equal-Tempered Microtonal Scale, with 24 Tones per Octave**

<i>Partial Number</i>	<i>Source Text Letters</i>	<i>Frequency</i>	<i>Nearest equal-tempered quarter-tone (cent deviation)</i>
24	A	440	A4
25	B	458.3333	A#4 (+21)
26	C	476.6666	A#4 (-11)
27	D	495	B4 (+4)
28	E	513.3333	B#4 (+17)
29	F	531.6666	C#5 (-22)
30	G	550	C#5 (-14)
31	H	568.3333	C#5 (-7)
32	I	586.6666	D5(-2)
33	J	605	D#5 (+1)
34	K	623.3333	D#5 (+3)
35	L	641.6666	D#5 (+3)
36	M	660	E5 (+2)
37	N	678.3333	E5 (-1)
38	O	696.6666	F5 (-4)
39	P	715	F#5 (-9)
40	Q	733.3333	F#5 (-16)
41	R	751.6666	F#5 (-23)
42	S	770	F#5 (+19)
43	T	788.3333	G5 (+10)
44	U	806.6666	G#5 (-1)
45	V,W	825	G#5 (-12)
46	X	843.3333	G#5 (-24)
47	Z	861.6666	G#5 (+14)

consisting of up to 160 characters, is first entered into the computer. This text is then divided into blocks, where words consisting of five or more letters form a single block, and words of less than five letters are joined with adjacent words to form a block (1964, p. 40). The vowel and consonant distributions of each block are stored. The first blocks that sum to less than 24 characters are joined and are used as the source of the "tone-row of the canon." In mapping to pitch, repeated characters are shifted to different octaves as described earlier. As Padberg states, "the resulting tone-row derived from the input is the melody-line of the canon, no two notes of which are identical" (Padberg 1964, p. 43).

Numerous features of the source text and resulting tone row are used to determine rhythmic

---

structures. The number of vowels, the number of consonants, the number of total characters, the number of pitches in the center register, and the number of pitches in outer registers are all employed (Padberg 1964). The least common multiple (LCM) of these numbers is found, and then up to five “relatively prime” factors of this LCM are found. Two integers are relatively prime if they share no common positive divisors except 1. Two or three relatively prime numbers are then extracted from this collection. These numbers are employed with subroutines that implement a form of Joseph Schillinger’s method of rhythmic resultants (Padberg 1964).

*The Schillinger System of Musical Composition* (Schillinger 1941) describes the “generation of resultant rhythmic groups as produced by the interference of two synchronized monomial periodicities” (p. 4), where monomial periodicities are whole-number multiples of a fixed duration. Schillinger offers as an example the interference of a monomial periodicity of four and a monomial periodicity of three. Although he encouraged visually drawing such generators on graph paper (Schillinger 1941), here, numerical sequences of duration values can be used. The monomial periodicity of four will produce the following time points (assuming 1 as a unit of duration): [0, 4, 8, 12]. The monomial periodicity of three will produce the following time points: [0, 3, 6, 9, 12]. The union, or “interference,” of these two lists results in the time points [0, 3, 4, 6, 8, 9, 12]. A duration list can be produced by taking the difference of adjacent values, producing the interference group [3, 1, 2, 2, 1, 3].

Schillinger employed various combinations of rhythmic resultants, and speculated that “all rhythmic patterns in music are either complete or incomplete resultants” (Schillinger 1941, p. 10). Schillinger’s rhythmic resultants are closely related to Xenakis’s sieves (Xenakis 1990; Ariza 2005b), where monomial periodicities can be seen as residual classes and “interference” can be achieved by the combination of residual classes by union. This example, employing the Xenakis sieve notation presented in Ariza (2005b), is represented as the sieve 4@0|3@0.

Padberg provides two theorems, and the necessary subroutines, to produce rhythmic resultants with

either two or three numbers. Whereas she explicitly uses relatively prime numbers, Schillinger makes no reference to prime numbers, demonstrating generators with values of four, six, eight, and nine (Schillinger 1941). Padberg’s procedure (Padberg 1964) takes two or three numbers and calculates all multiples of these numbers up to the LCM of all of the numbers. She then forms the union of these sets of multiples, and sorts the values. To obtain durations, the difference of successive values is then calculated. As shown with the earlier example, this procedure produces symmetrical duration sequences.

After analysis of the source text, preparation of the tone row, and preparation of the rhythmic materials, the primary voice of the canon is created. This voice combines the tone row with the defined rhythm pattern; both are repeated until their endings coincide. The secondary voice is temporally shifted a number of beats based on the number of consonants in the source text, and then restates the pitches and rhythms of the primary voice. The primary voice sustains its last pitch until the second voice arrives at the same pitch (Padberg 1964). If four voices are employed, they begin and end in a similar manner (Padberg 1964).

## Program 2

Program 2 extended Program 1, taking advantage of the greater resources of the IBM 7072. Program 2 included all of Program 1’s functionality, permitting the creation of canons in two or four voices, but it now also permitted the creation of free fugues in two voices. Compositions could now include up to three tone-rows (Padberg 1964).

The canon-generation procedures of Program 2 follow those of Program 1. In addition to permitting the use of three tone rows, Program 2 additionally derives rhythm patterns from each component block of the source text.

The free fugue compositions, however, use numerous new procedures. Here, retrograde, inversion, and retrograde inversion statements of the tone row are created and stored. It is notable that Padberg does not transpose the various presentations of the

tone row. Rhythm patterns derived from blocks are used independently of the complete tone-row-derived rhythm described previously (Padberg 1964). The free fugue treats the original tone row as the fugal subject, and the block-derived melodies and rhythms as motives.

The form of the free fugue intermingles various presentations of the tone rows in prime, retrograde, inversion, and retrograde inversion forms. Padberg describes this plan as exemplifying “imitation and transformation, of contrast and symmetry” (1964, p. 67). She additionally notes that, considered as sets, the four presentations of the theme constitute an Abelian group, where “each element is its own inverse” (Padberg 1964, p. 68). In this context, her form carefully deploys theme presentations to maximize balance and symmetry.

At times, each voice presents a different form of the theme; at other times, forms of the theme are presented in strict imitation. Rhythms are derived from the same text blocks as pitches, or are derived from the complete tone row (Padberg 1964). Voices rest at the conclusion of various sections, the duration of the rest again determined by the number of consonants in the source text. The fugue ends with retrograde forms of the tone row in strict imitation, assuring that the last tone is the unison of the initial pitch. Padberg provides a table to summarize the deployment of melodic material transformations (Padberg 1964).

Although she does not provide details on producing a four-voice fugue, she offers that “the principles illustrated . . . for the two-voice fugue can readily be applied in a separate program” for four voices (Padberg 1964, p. 70).

Undeterred by the aforementioned difficulty of sonic realization, Padberg explored the creation of novel musical structures, employing well-established canonic techniques in the context of a unique microtonal scale. For Padberg, the computer offered a tool for rapid and diverse transformation of a single input into an extended and complex composition. Instead of randomness as a source of variation, complex and creative procedural manipulations are explored and applied to all relevant musical parameters. Going beyond the basic serial compositional techniques, Padberg extended the

idea of achieving compositional unity by using a single source for organizing parameters.

## Transcriptions and Sonic Realizations

Representative output of Padberg’s programs is provided in her dissertation as score tables (Padberg 1964). Padberg’s score tables provide only pitch and duration parameters, and use an equal time unit for each line of text such that the text-based output provides a proportional visual representation of duration (Padberg 1964).

Padberg describes the result of her system as “lacking in aesthetic appeal” but retaining “the qualities traditionally associated with both absolute and program music: it is a logically conceived, unified composition, integrally bound to its ‘title’ or ‘tone-rows’ in melody and rhythm . . . , [and] not dependent on outside connotations for its explanation or development” (Padberg 1964). Although this statement might appeal to a strict formalism, Padberg elsewhere notes that endless computer transformation of melodic material has limited value; that “unless a composer uses . . . creative vision in addition to symmetrical technique, the result will be no more than mathematical formulas in sound” (Padberg 1964, p. 70).

She includes five untitled compositions in her dissertation. These compositions will be referred to here as Composition 1 through Composition 5. For this article new realizations and transcriptions of all of Padberg’s compositions have been made. These realizations were made by transcribing the printed score tables by hand into a tab-delimited data table. This table was then processed by a Python script to provide output as a MusicXML score and as a Csound CSD file. For a clear presentation of the pitch and rhythm structures with a harp-like tone, a simple Csound instrument using the *wgpluck* opcode is used with a quickly attacked envelope. Small amounts of reverb (via the *freeverb* opcode) and stereo panning are employed to support voice separation.

Compositions 1 through 4 all use the same source text as input (“college canon”) and thus begin with the same primary voice. Composition 4 is unique,



Figure 3. Excerpt of Composition 4, a four-part microtonal canon by Harriet Padberg. Pitch and quarter-tone spellings are approximate.



and perhaps more musically compelling, in that it is the only four-voice example provided by Padberg. Figure 3 is a transcribed excerpt of Composition 4; the complete Csound realization is available as Audio Example 3. Composition 5 is the only example of a free fugue provided by Padberg. The source input text is “university canon and twentieth century fugue.” Figure 4 is a transcribed excerpt of Composition 5, and the complete Csound realization is Audio Example 4.

## Conclusion

Caplin left Shell in 1964 and, after various other projects in industrial operations planning, began work for the United Nations (UN) Development Programme. Through the UN, Caplin worked for the Ministry of Development of the Israeli government, building mathematical models of the Israeli chemical industry, and establishing Israel Chemicals Ltd. He went on to do strategic and policy planning at the World Institute, Jerusalem; for the Ministries of Finance, Trade and Industry, Defense, and Transportation; and for the Water Planning and Ports Authorities. In 1975 Caplin joined the World Bank in Washington, DC, and worked as a

senior industrial economist in the Chemical Projects Department. Caplin retired from the World Bank in 1989, though until 1997 he continued to work on World Bank projects in Romania, Hungary, China, and elsewhere. Caplin is still an active composer, writing vocal, choral, and chamber music for friends and family.

Padberg, although not continuing her investigations in computer music, kept music as an important part of her career. After completing her PhD, she was appointed professor of mathematics and music at Maryville University in 1964 (Padberg 2011). With Ruth Sheehan she established a music therapy program at Maryville in the early 1970s. She began work as a music therapist in 1980 (Padberg 2008) and, since retirement in 1992, has dedicated two days a week to music therapy at the Emmaus Homes (Anonymous 2006). Her work has been recognized with recent honors, including the Epsilon Sigma Alpha’s statewide Distinguished International Academy of Noble Achievement Award and Maryville’s Dean’s Award in the School of Health Professions (Anonymous 2006).

Although these two early projects, separated by ten years, offer considerably different approaches, both employ methods closely related to pre-computer approaches to algorithmic composition.

Figure 4. Excerpt of Composition 5, a microtonal free fugue by Harriet Padberg. Pitch and quarter-tone spellings are approximate.



Caplin and Prinz built a system related to Mozart's *Musikalisches Würfelspiel*, whereas Padberg built a system related to Guido of Arezzo's pitch-to-vowel mapping technique. Both, in turn, took these approaches further to arrive at techniques more idiomatic to the computer, such as employing transitional probabilities and organizing microtonal pitch scales. These techniques continue to be used and extended in contemporary computer music systems.

The performance of music with hoot-based synthesis techniques, as explored on the CSIRAC in the early 1950s (Doornbusch 2004), provides context for evaluating the significance of other early computer music experiments. In regard to work done with the CSIRAC, Doornbusch writes that "the idea of using a computer . . . to create music was certainly a leap of imagination at the time" (p. 23); the experiments described herein demonstrate similar leaps of imagination, but not just in producing musical sound, but also in using a computer to generate new musical structures. Doornbusch comments on the unrealized potential of the CSIRAC, speculating that, under the guidance of interested composers, "perhaps we would have had very early examples of computer-based microtonal music and algorithmic composition" (p. 23): surprisingly, such early examples are exactly what we find with the music of Padberg, Caplin, and Prinz.

## Acknowledgments

Research for this article began in 2005, while I was living in the Netherlands and conducting research at the Institute of Sonology, The Hague, under a grant sponsored by the United States Fulbright program, the Institute for International Education (IIE), and the Netherlands-America Foundation (NAF). Finding contact information for Caplin and Padberg took years, and was the result of the generosity of numerous individuals.

Foremost, I am deeply grateful to Lunbeck, Caplin, and Padberg for discussing with me at length their recollections. Thanks also to Caplin for finding, transferring, and releasing his recordings, and to Padberg for permission to create new transcriptions and realizations of her works.

Correspondence with Lunbeck was made possible in part by Nicole Falque (Shell Coordination Centre S.A., Monnet Centre International Laboratory, Louvain-La-Neuve, Belgium) and Lida van der Horst and Anneke Schelvis (Shell Global Solutions International B.V., Shell Research and Technology Centre, Amsterdam, the Netherlands). Correspondence with Caplin was made possible in part by Joan Bartlett of the World Bank 1818 Society. Correspondence with Padberg was made possible in part by Matthew Guerrieri.

Thanks to Onno Zweers and Cordula Rooijendijk for sharing their research on the MIRACLE. Thanks also to Paul Berg, Paul Doornbusch, Max Mathews, Jasia Reichardt, and Steve Cork and Christopher Scobie of the British Library for research assistance.

## References

- Ames, C. 1987. "Automated Composition in Retrospect: 1956–1986." *Leonardo* 20(2):169–185.
- Anonymous. 1956a. "'Brain' Computes New Tune for TV." *New York Times* July 3: 51.
- Anonymous. 1956b. "The First Ballad to be Composed by an Electronic Computer." *International Musician* 55:21.
- Anonymous. 1956c. "Syncopation by Automation." In *Data from ElectroData*, vol 2. Pasadena, California: Burroughs Corporation (ElectroData Division).
- Anonymous. 2006. "Speech of Angels." *Heart: A Journal of the Society of the Sacred Heart, U.S. Province* 4(3):8–10.
- Ariza, C. 2005a. "An Open Design for Computer-Aided Algorithmic Music Composition: athenaCL." PhD Dissertation, New York University, Department of Music, Graduate School of Arts and Sciences.
- Ariza, C. 2005b. "The Xenakis Sieve as Object: A New Model and a Complete Implementation." *Computer Music Journal* 29(2):40–60.
- Assayag, G. 1998. "Computer Assisted Composition Today." Available on-line at [recherche.ircam.fr/equipes/repmus/RMPapers/Corfou98](http://recherche.ircam.fr/equipes/repmus/RMPapers/Corfou98). Accessed April 2011.
- Baker, R. A. 1963. "MUSICOMP: Music-Simulator for Compositional Procedures for the IBM 7090 Electronic Digital Computer." Technical Report No. 9, University of Illinois, Experimental Music Studio.
- Bashe, C. J., et al. 1986. *IBM's Early Computers*. Cambridge, Massachusetts: MIT Press.
- Brooks, F. P., et al. 1957. "An Experiment in Musical Composition." *IRE Transcripts on Electronic Computers* 6(3):175–182.
- Cambell-Kelly, M., and W. Aspray. 1996. *Computer: A History of the Information Machine*. New York: Basic Books.
- Caplin, D. 2008. Written correspondence with the author. 8 December 2008.
- Caplin, D. 2009a. Written correspondence with the author. 1 July 2009.
- Caplin, D. 2009b. Written correspondence with the author. 12 May 2009.
- Caplin, D. 2009c. Written correspondence with the author. 13 May 2009.
- Caplin, D. 2009d. Written correspondence with the author. 17 March 2009.
- Caplin, D. 2011. Written correspondence with the author. 23 April 2011.
- Ceruzzi, P. E. 2003. *A History of Modern Computing*. Cambridge, Massachusetts: MIT Press.
- Cohen, J. E. 1962. "Information Theory and Music." *Behavioral Science* 7(2):137–163.
- Computer Music Journal. 2004. "DVD Program Notes." *Computer Music Journal* 28(4):116–129.
- Cope, D. 1991. *Computers and Musical Style*. Oxford: Oxford University Press.
- Darreg, I. 1957. "The Electronic Computer Takes Up Music Lessons." Available on-line at [sonic-arts.org/darreg/dar6.htm](http://sonic-arts.org/darreg/dar6.htm). Accessed April 2011.
- De Marco, G., et al. 1999. "The Early Computers of Italy." *IEEE Annals of the History of Computing* 21(4):28–36.
- Dodge, C., and T. A. Jerse. 1997. *Computer Music: Synthesis, Composition, and Performance*. New York: Schirmer.
- Doornbusch, P. 2004. "Computer Sound Synthesis in 1951: The Music of CSIRAC." *Computer Music Journal* 28(1):10–25.
- Doornbusch, P. 2005. *The Music of CSIRAC: Australia's First Computer Music*. Altona, Victoria, Australia: Common Ground.
- Eskens, E. 2002. "Interview met Lidy Zweers-De Ronde, rekenaarster (programmeur) van de MIRACLE (Ferranti Mark I\*), de eerste commerciële computer in Nederland, bij Shell labs in Amsterdam noord." Available on-line at [www.xs4all.nl/~onnoz/miracle/extra/texts/transcriptie%20interview%20Lidy.txt](http://www.xs4all.nl/~onnoz/miracle/extra/texts/transcriptie%20interview%20Lidy.txt). Accessed April 2011.
- Ferranti Ltd. 1956. *An Introduction to the Ferranti Mercury Computer*. Hollinwood, Lancashire, UK: Ferranti Ltd.
- Fildes, J. 2008. "'Oldest' computer music unveiled." BBC News. Available on-line at [news.bbc.co.uk/2/hi/technology/7458479.stm](http://news.bbc.co.uk/2/hi/technology/7458479.stm). Accessed April 2011.
- Fucks, W. 1962a. "Mathematical Analyses of Formal Structure of Music." *IRE Transactions on Information Theory* 8(5):225–228.
- Fucks, W. 1962b. "Musical Analysis by Mathematics, Random Sequences, Music and Accident." *Gravesano Review* 6(23):146–155.
- Gardner, M. 1974. "Mathematical Games: The Arts as Combinatorial Mathematics, or, How to Compose Like Mozart with Dice." *Scientific American* 231(6): 132–136.

- Guido of Arezzo. 1978. "Micrologus." In C. V. Palisca, ed. *Hucbald, Guido, and John on Music: Three Medieval Treatises*. Translated by W. Babb. New Haven, Connecticut: Yale University Press, pp. 57–83.
- Hedges, S. A. 1978. "Dice Music in the Eighteenth Century." *Music and Letters* 59(2):180–187.
- Hiller, L. 1956. "Abstracts: Some Structural Principles of Computer Music." *Journal of the American Musicological Society* 9(3):247–248.
- Hiller, L. 1967. "Programming a Computer for Music Composition." In G. Lefkoff, ed. *Computer Applications in Music*. Morgantown: West Virginia University Library, pp. 65–88.
- Hiller, L. 1969. "Some Compositional Techniques Involving the Use of Computers." In H. von Foerster and J. W. Beauchamp, eds. *Music by Computers*. New York: John Wiley and Sons, pp. 71–83.
- Hiller, L. 1970. "Music Composed with Computers: An Historical Survey." In H. B. Lincoln, ed. *The Computer and Music*. Ithaca, New York: Cornell University Press, pp. 42–96.
- Hiller, L., and L. Isaacson. 1958. "Musical Composition with a High-Speed Digital Computer." *Journal of the Audio Engineering Society* 6(3):154–160.
- Hiller, L., and L. Isaacson. 1959. *Experimental Music*. New York: McGraw-Hill.
- Klein, M. L. 1957. "Syncopation in Automation." *Radio-Electronics* 36.
- Koenig, G. M. 1970. "Project One." In *Electronic Music Report*, vol 2. Utrecht: Institute of Sonology, pp. 32–46.
- Koninklijke/Shell-Laboratorium. 1989. *Research aan het Ij: LBPMA 1914—KSLA 1989*. Baarn: Market Books.
- Lunbeck, R. J. 2005a. Written correspondence with the author. 25 April 2005.
- Lunbeck, R. J. 2005b. Written correspondence with the author. 30 April 2005.
- MacGregor, B. 2002. "Cybernetic Serendipity Revisited." *Proceedings of the 4th Conference on Creativity and Cognition*, pp. 11–13.
- Mathews, M. 2009. Personal correspondence with the author. 9 August 2009.
- Moles, A. A. 1960. *Les Musiques Expérimentales*. Paris: Fondation Internationale.
- Moreau, R. 1984. *The Computer Comes of Age: The People, the Hardware, and the Software*. Cambridge, Massachusetts: MIT Press.
- Mozart, W. A. 1793. *Anleitung zum Componiren von Walzern so viele man will vermittelst zweier Würfel ohne etwas von der Musik oder Composition zu verstehen*. Berlin: Johan Julius Hummel.
- Mozart, W. A. 1798. *Anleitung Englische Contretänze mit zwei Würfeln zu componiren so viele man will, ohne etwas von der Musik oder der Composition zu verstehen*. Bonn: Chéz Nikolaus Simrock.
- Mozart, W. A. 1956. *Musikalisches Würfelspiel*. Mainz: Schott.
- Napper, R. B. E. 2000. "The Manchester Mark 1 Computers." In R. Rojas and U. Hashagen, eds. *The First Computers: History and Architectures*. Cambridge, Massachusetts: MIT Press, pp. 365–377.
- Napper, R. B. E. 2005. "Computer 50: The University of Manchester Celebrates the Birth of the Modern Computer." Available on-line at [www.computer50.org](http://www.computer50.org). Accessed April 2011.
- Padberg, H. A. 1964. "Computer-Composed Canon and Free Fugue." PhD Dissertation, St. Louis University.
- Padberg, H. A. 2008. Telephone conversation with the author. 3 July 2008.
- Padberg, H. A. 2011. Telephone conversation with the author. 11 February 2011.
- Pierce, J. R. 1961. *Symbols, Signals and Noise: The Nature and Process of Communication*. New York: Harper.
- Pinkerton, R. C. 1956. "Information Theory and Melody." *Scientific American* 194(2):77–86.
- Prinz, D. G. n.d. *Introduction to Programming on the Manchester Electronic Digital Computer Made by Ferranti Ltd*. Manchester, UK: Ferranti Ltd.
- R. 1954. "Miracle Reken Voor U." *Lab Spiegel* 6:4–5.
- Reichardt, J. 1968. *Cybernetic Serendipity: The Computer and the Arts*. London: Studio International, W. and J. Mackay.
- Rivest, J. 1999. "In Advance of the Avant Garde: John Cage at the University of Illinois, 1952–1969." Lecture presented at the University of Illinois, 22 April 1999. Available on-line at [emfinstitute.emf.org/articles/rivest99/rivestoncage992.html](http://emfinstitute.emf.org/articles/rivest99/rivestoncage992.html). Accessed April 2011.
- Roads, C. 1985. "Research in Music and Artificial Intelligence." *ACM Computing Surveys* 17(2):163–190.
- Rooijendijk, C. 2007. *Alles moest nog worden uitgevonden*. Amsterdam: Uitgeverij Atlas.
- Schillinger, J. 1941. *The Schillinger System of Musical Composition*. New York: Carl Fischer.
- Shannon, C. E., and W. Weaver. 1949. *A Mathematical Theory of Communication*. Urbana: University of Illinois Press.
- Turing, A. M. 1951. *Programmers' Handbook for the Manchester Electronic Computer Mark II*. 1st ed. Turing Digital Archive. Available on-line at [www.turingarchive.org/browse.php/B/32](http://www.turingarchive.org/browse.php/B/32). Accessed April 2011.

- 
- Turing, A. M., and R. Brooker. 1952. *Programmers Handbook for the Manchester Electronic Computer Mark II*. 2nd ed. Turing Digital Archive. Available on-line at [www.computer50.org/kgill/mark1/progman.html](http://www.computer50.org/kgill/mark1/progman.html). Accessed April 2011.
- Tweedale, G. 1993. "A Manchester Computer Pioneer: Ferranti in Retrospect." *IEEE Annals of the History of Computing* 15(3):37–43.
- UK National Archive for the History of Computing. 2005. *UK National Archive for the History of Computing: Draft Catalogue*. Manchester: Centre for the History of Science, Technology and Medicine. Available on-line at [www.chstm.manchester.ac.uk/downloads/media,38917,en.pdf](http://www.chstm.manchester.ac.uk/downloads/media,38917,en.pdf). Accessed April 2011.
- Williams, M. R. 1997. *A History of Computing Technology*. Los Alamitos, California: IEEE Computer Society Press.
- Xenakis, I. 1965. "Free Stochastic Music from the Computer. Programme of Stochastic Music in Fortran." *Gravesaner Blätter* 26:54–92.
- Xenakis, I. 1990. "Sieves." *Perspectives of New Music* 28(1):58–78.