Thomas Haigh

Defining Digitalities I: What's Digital about Digits?

2023

https://doi.org/10.25969/mediarep/20046

Veröffentlichungsversion / published version

Buch / book

MEDIEN DER KOOPERATION

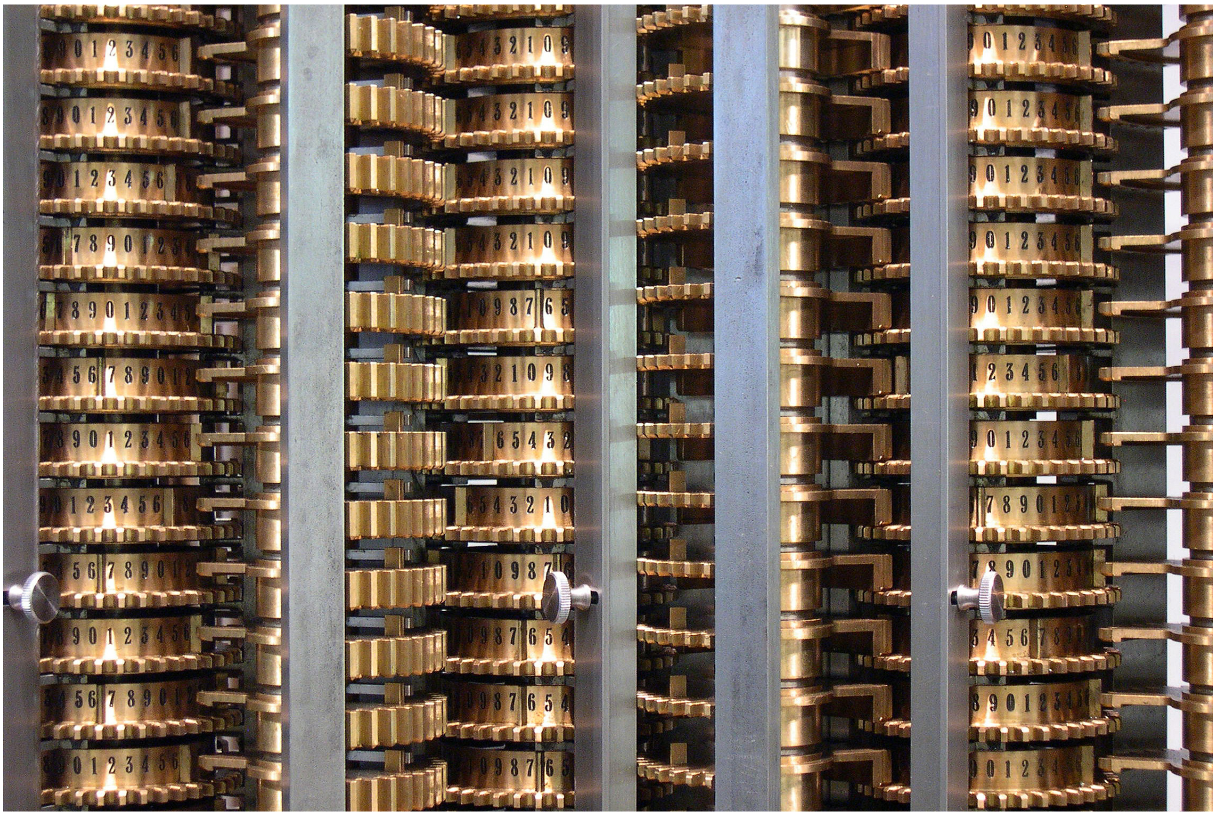
Universität Siegen

DFG

Defining Digitalities I:

What's Digital about Digits?

Thomas Haigh
University of Wisconsin, Milwaukee & Siegen University



WORKING PAPER SERIES | NO. 30 | JULY 2023
Collaborative Research Center 1187 Media of Cooperation

Working Paper Series
Collaborative Research Center 1187 Media of Cooperation

Print-ISSN 2567–2509
Online-ISSN 2567–2517
DOI doi.org/10.25819/ubsi/10259
dspace.ub.uni-siegen.de/handle/ubsi/2452
URN urn:nbn:de:hbz:467-24526

This Working Paper Series is edited by the Collaborative Research Center Media of Cooperation and serves as a platform to circulate work in progress or preprints in order to encourage the exchange of ideas. Please contact the authors if you have any questions or comments.

The Working Papers are accessible online at:
https://www.mediacoop.uni-siegen.de/de/publikationen/working-papers-media-of-cooperation/

Print copies can be ordered by sending an email to:
workingpaperseries@sfb1187.uni-siegen.de

Publication is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Project-ID 262513311 – SFB 1187.

Cover image: Image created by Wikimedia user Carsten Ullrich, used under CC BY-SA 2.5 (https://creativecommons.org/licenses/by-sa/2.5/)
Layout: Mattis Hunting

Universität Siegen
SFB 1187 Medien der Kooperation
Herrengarten 3
57072 Siegen, Germany
https://www.mediacoop.uni-siegen.de.sfb1187.uni-siegen.de
workingpaperseries@sfb1187.uni-siegen.de

Defining Digitalities I:
What's Digital about Digits?

Thomas Haigh
University of Wisconsin, Milwaukee & Siegen University

Abstract: Modern discourses emphasizes electronic immateriality as the defining feature of digital technology. The idea that digits might be digital when punched onto cards, or even written on a piece of paper, is no longer intuitive. Yet by reconstructing the context in which the categories of digital and analog were first distinguished historically in the 1940s, I argue that the concept of digitality is rooted in the mechanical representation of digits in early computers, which contemporary observers immediately recognized was shared with earlier technologies such as telephone switching systems, punched cards, and calculating devices. Digitality is not a feature of an object itself, but of the way that object is read (whether by human or by machine) as encoding symbols chosen from a finite set. In conclusion, digitality is constituted through reading practices.

Keywords: digital; analog; binary; differential analyzer; Bell Labs.

I will argue in this working paper that the historical study of digitality should begin with careful attention to digits.[1] Digits matter here in two ways. First, our current discourse of *the digital* has its historical roots in the categories of *digital* and *analog*, which were defined in the 1940s to distinguish between two approaches to automatic computation. Digital computers were digital because they carried out their mathematical operations by encoding and manipulating digits. Second, some of the crucial affordances of today's electronic digital media have their roots in the characteristics that digits exhibit whether manipulated by humans or by machines. Digits are discrete set of symbols that can be reliably transcribed from one medium to another and sequenced to represent quantities of any size or to any degree of accuracy.

Digital was, in its original context, a quite literal term confined to machines that represented numbers rather than encompassing control systems based on discrete encodings such as automatic looms or musical boxes. It was not, however, confined to electronic digits or immaterial devices. While the categories of digital and analog were created in response to the emergence of electronic computation they were immediately understood as applicable to earlier technologies going all the way back to the abacus. The initial choice of the term *digital* and its eventual resurgence as shorthand for our current technological epoch were both somewhat arbitrary. Yet taking the continuity seriously can be illuminating. The essential affordances of modern digital technologies are built on top of core affordances of digitality shared not just with earlier kinds of digital machines but with digits themselves.

The literal digitality of machines that represent digits is distinct from a broader and later sense of digitality as the encoding of sequences of symbols. Digits are a subset of the alphanumeric characters manipu-

1 By this I mean numerical digits, though others have argued for tracing the idea of digitality back through another layer of metaphor to explore correspondences between the capabilities of digital systems and human fingers. Benjamin Peters, "Digital", in Digital Keywords: A Vocabulary of Information Society & Culture, ed. Benjamin Peters (Princeton, NJ: Princeton University Press, 2016):93-108.

lated automatically by computers from the 1950s onwards, and those symbols were used in turn to represent other things such as audio, video, and pictures. When the concept of digitality was stretched to include these non-numerical capabilities it was originally a kind of metaphor, though with time the metaphor has been naturalized to the extent that the connection of *digitality* and *digit* is now easy to overlook.

Historical Origins of Digital and Analog

While digitality has recently been equated with immateriality, the antonym of *digital* is not *physical* but *analog*. As Ronald Kline has explained in his careful and exhaustively researched paper on the topic, the terms were introduced during the second world war as automatic computer projects began to proliferate.[2] Kline's earliest identified use of the words to distinguish between two classes of computer occurred in 1942, in a document by George Stibitz of AT&T's Bell Labs. During the war he worked with the National Defense Research Committee, a group chartered to bring scientific expertise to assist in the nation's struggle. Stibitz introduced the juxtaposition of analog and digital in a memo commenting on a set of proposals for the design of a computer able to direct anti-aircraft guns. That was a mathematical problem: the gun had to fire not at the plane's current position but at the place it would be when the shell's arc intersected with its own future course. This required the rapid solution of differential equations.[3]

Stibitz is best remembered as the creator, between 1937 and 1946, of a series of computers that used electromechanical relays to represent numbers. These took the approach he designated as *digital*. Their relays switched automatically between two possible positions. A cluster of relays represented a number using the binary number system. Addition, multiplication, and other mathematical operations took place automatically as electrical impulses moved through wires but the arithmetic involved followed the same basic rules that a school child might have carried out using a pencil and paper (adjusted, of course, for the differences between binary and decimal – add 1 to 1 to get 0 carry 1, rather than add 1 to 9 to get 0 carry 1). The machines were *digital* because their mechanisms encoded digits and manipulated them to reach their solutions.

The other class of machines were called *analog* because their internal structure provided a model, or analogy, of the system being investigated. Kline suggests that this term slightly predated *digital* in this context, having been used since the 1930s. Vannevar Bush at MIT had investigated the behavior of power grids by building in the laboratory what were essentially scale models – each small wire and current proportional to the much heavier wires and larger currents of the real power network. He followed this up with something more flexible and more abstract: the differential analyzer. Each of its six spinning disks represented one term in a differential equation.

The disks were mounted on shafts, which span more rapidly as quantities they represented increased. The wheels sat vertically on top of the disks. Like the stylus of a record player they could be moved closer or further from the middle of the disk. The closer they got to the outer edge of a disk they more rapidly they rotated. Motion of a wheel was mechanically amplified to control the motion of the next disk. Adjustments, including the positioning of wheels and the use of gears to add together the motion of two shafts, changed the relationships between the six terms.[4]

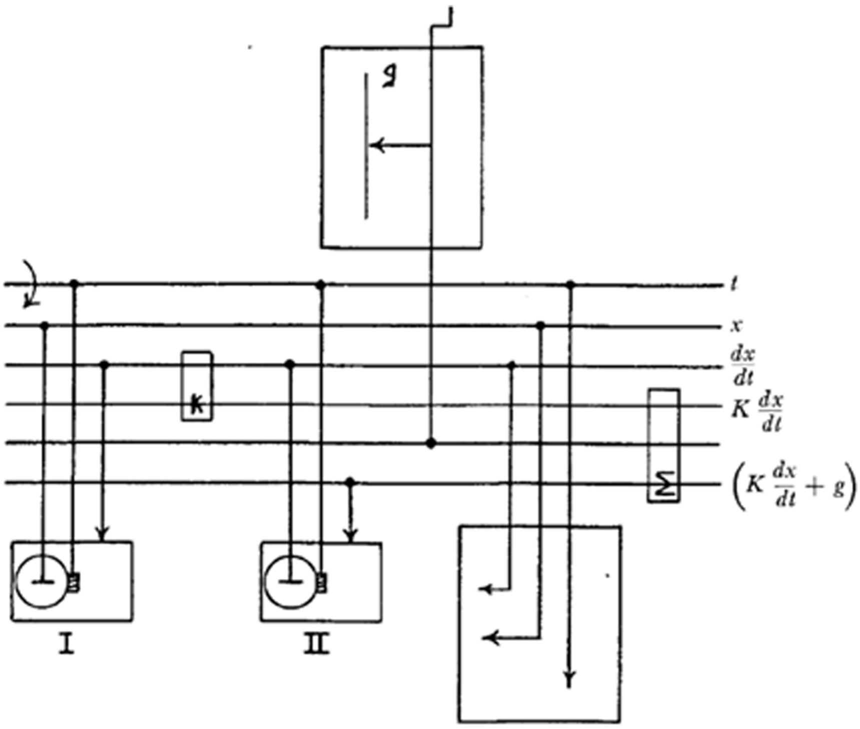
Specific parts of the integrator were thus analogous to specific parts of the system being modelled. The integrator as a whole became an embodiment of the mathematical equation, an analogy with the entire system being modelled. *Allegory* might have been a better word than *analogy* for this complex correspondence. In an allegory, such as George Orwell's *Animal Farm* or a biblical parable, each part of the story corresponds with something specific in the larger world. The relationships between the different objects in the story are the same as those between the analogous features of the world. If the rotation of one disk in the differential analyzer represents the height of a shell, another its velocity, and a third its acceleration then the relationships between those disks should, when the device is properly adjusted, be very close to the relationships of those real-world quantities. There were no digits involved –operators controlled the speed of one or more of the disks by tracing input curves using devices coupled to the motion of wheels. At the far end of the machine, a mechanical arm sketched the shape of the result. Differential analyzes were the most advanced automatic computers of the 1930s. Comparable principles were used in the gun director design chosen for the NRDC project, and had already been applied for comparable systems used for fire control on naval vessels.

Analog computers were sold and developed into the 1970s. They used a range of media to represent changes in the quantities being computer. In some fluid dripped between tanks, in others variations in electrical cur-

2 Ronald R Kline, "Inventing an Analog Past and a Digital Future", in Exploring the Early Digital, ed. Thomas Haigh (Cham, Switzerland: Springer, 2019):19-39.

3 David A Mindell, Between Human and Machine: Feedback, Control, and Computing Before Cybernetics (Baltimore: Johns Hopkins University Press, 2002), ch. 9 & 11 provides an account of the NRDC's work in this area that foregrounds the role of Stibbitz and Shannon and follows the legacy of this project into Stibbiz's general purpose relay computing projects.

4 Larry Owens, "Vannevar Bush and the Differential Analyzer: The Text and Context of an Early Computer," Technology and Culture 27, no. 1 (January 1986):63-95. Vannevar Bush, "The Differential Analyzer. A New Machine for Solving Differential Equations," Journal of the Franklin Institute 212, no. 4 (October 1931):447-488.



$$\frac{d^2x}{dt^2} + k\frac{dx}{dt} + g = 0.$$

Figure 1: Bush's paper describing the differential analyzer used this schematic notation to describe the relationship between each of the six shafts and the corresponding term in a mathematic equation describing the motion of a falling body. The diagram also specifies the relationships between shafts, implemented mechanically with devices such as integrators and gear boxes.[5]

rent replaced the changes of rotational speed used in the differential analyzer. But the many kinds of analog computer shared two crucial features. Firstly, as with the differential analyzer each quantity used in the computation was represented by a different part of the machine, and the relationships between these components were proportional (i.e. analogous) to those between the things being computed. Second, variations were continuous. In practice there were limits to precision. An operator might not trace a curve perfectly, for example. But in theory any variation, however slight, in the input should lead to a corresponding variation in the output.

As Kline showed, while the need to distinguish between these two fundamentally different approaches to computing was widely accepted during the mid-1940s the specific pairing of analog vs. digital was only one of many used to accomplish this – even among scientists connected to the NRDC. One pairing was between computers that measured and those that counted. Digital systems were sometimes called pulse or impulse computers, because many of them encoded numbers as electrical pulses. Others, drawing on mathematical categories, described them as *continuous-variable* (or simply *continuous*) and *discrete-variable* (or simply *discrete*) machines. Stibitz himself used these alternative forms when giving a lecture at as part of the University of Pennsylvania's 1946 summer school for people interested in building electronic computers.[6]

By the end of the 1940s, however, the language of *digital* vs. *analog* was generally accepted by those discussing automatic computers. Consistent use of *digital* by John von Neumann in his 1945 *First Draft of a Report on the* EDVAC, the first description of the architecture of modern computers, must have helped.[7] The concept of digitality was also applied, retroactively, to older computing devices. A 1949 article in *Scient ific American* on "Mathematical Machines" surveyed the latest digital computers like ENIAC and IBM's SSEC,

5 Bush, "The Differential Analyzer. A New Machine for Solving Differential Equations", p. 457.

6 George Stibitz, "Introduction to the Course on Electronic Digital Computers", in The Moore School Lectures: Theory and Techniques for Design of Electronic Digital Computers, ed. Martin Campbell-Kelly and Michael R Williams (Cambridge, MA: MIT Press, 1985):3-16. As Ron Kline has observed, the word "digital" occurs only in the title of the lecture, but not in its actual text. Kline, "Inventing an Analog Past and a Digital Future". Given that the lecture was titled after the course as a whole, to serve as an introduction, it seems likely that it reflected the preferred terminology of organizers of the course (primarily Carl C. Chambers of the Moore School) rather than of Stibitz himself who was a last-minute substitute for the speaker originally scheduled to give the lecture.

7 John von Neumann, "First Draft of a Report on the EDVAC," IEEE Annals of the History of Computing 15, no. 4 (October 1993):27-75.

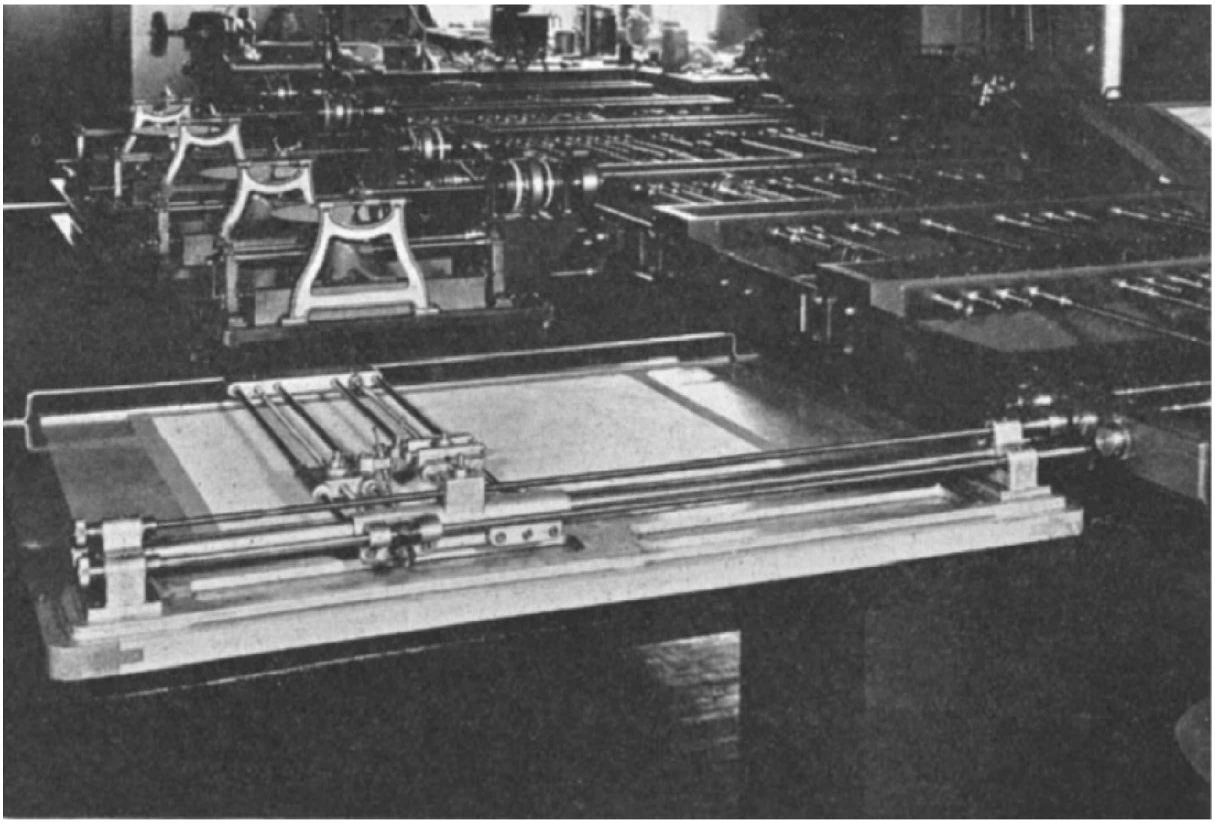


Figure 2: The output table, on which a pencil moved by the differential analyzer would draw a curve representing the solution to the problem traced on the input table.[8]

but went much further back in history. "The first artificial digital computing device," claimed science writer Harry M. Davis, "was the abacus, a manually operated mechanical memory of great antiquity." IBM punched cards, mechanical adding machines, teletypes, tape-controlled relay computers, and Charles Babbage's unfinished difference engine were also invoked as examples of digital technology.[9] While the presence of teletypes on the list suggests that Davis was already inching towards a concept of digitality that included encodings of text as well as numbers, the other examples were all literally digital in the sense that they encoded and manipulated numerical digits.

Davis understood that these numbers could be represented in many different media, some easily readable by humans (indeed, joined to the human body) and others invisible to our unaided senses. He explained "The Digital Idea" as follows:

> The digital computer is distinguished by the fact that it does not measure: it counts. It never responds to a greater or lesser degree; at every stage of its action, it is an 'all or nothing' device operating with discrete signals that either exist or do not exist. The simplest digital computer is the human hand, from which, of course, we have our decimal system. Corresponding to such primitive indicators of a numerical unit as a finger, a pebble, or a stylus scratch, the new automatic computers represent digits by such methods as: A round hole in a strip of tape. A square hole in a piece of cardboard. A current in an electromagnet. An armature attached to the magnet. A closed pair of electrical contacts. A pulse of current in an electrical transmission line. An electronic tube in which current is permitted to flow from filament to plate. A magnetized area on a steel or alloyed wire. A magnetized area on a coated tape. A darkened area on a strip of photographic film. A charged area on the face of a cathode-ray tube. A moving ripple in a tank of mercury.

Although *digital* and *binary* are today often conflated, Davis was well aware of that decimal is no less digital than binary. Both are number systems that use digits, and both can be encoded in many different media, including digital electronics. Witness his advocacy for the abacus as the original digital computer, and his extensive comparison of the use of decimal, binary coded decimal, and pure decimal number systems for electronic computers.

8 Ibid., p. 454.

9 Harry M Davis, "Mathematical Machines," Scientific American 180, no. 4 (April 1949):28-39.

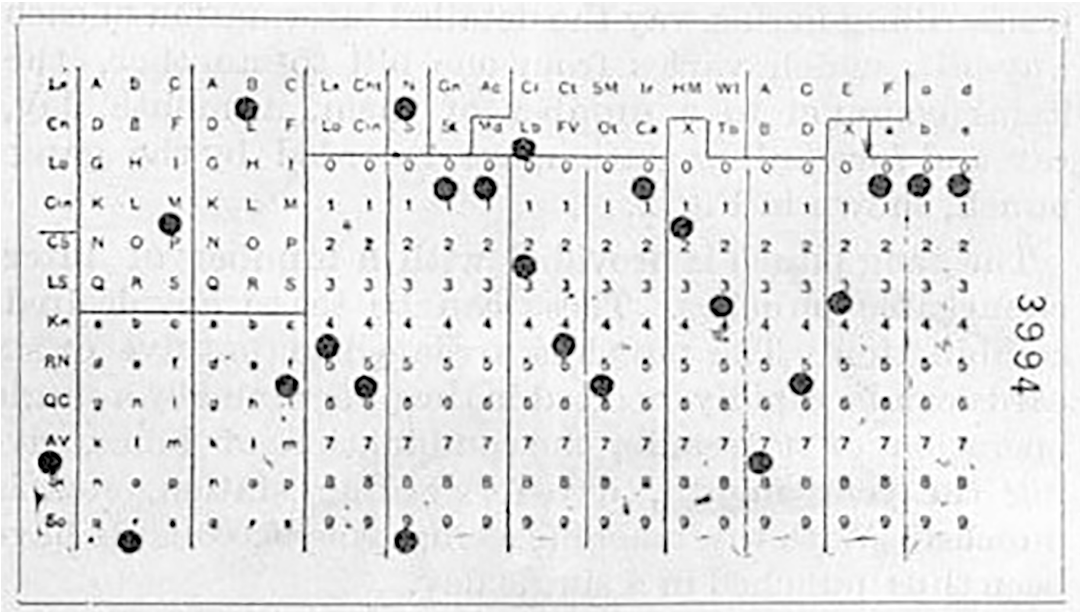


Figure 3: An example punched card in the early 24 column format, taken from an 1895 issue of the Railway Gazette. Most of the fields represent two- and three-digit decimal numbers, in a format that could be tallied automatically by tabulating machines. The non-numeric fields were used to sort and filter the cards. Printed labels on the card aided human legibility. Source: Wikimedia.

Digitality as a Reading Practice

In the beginning, then, what made digital computers digital was their representation of quantities as digits and their manipulation of these digits by mechanizing the ordinary processes of arithmetic. They did their mathematics in the same ways learned by school children. Within each digital computer were mechanisms to encode digits. I term this sense of digital *numerical digitality* in the sense of numerical mathematics, which relies on the manipulation of digits to provide approximate solutions to equations.

Matter itself is neither digital nor analog. What is digital or analog is not an object itself, but the way in which an object is read. Digitality is active: the practice of examining a part, usually a very small part, of the world and classifying it as falling into one of a finite number of valid states. Digitality is enacted by reading practices. As we shall see in the next paper in this series, not all of the processes now viewed as digital involve actual digits. But the process of digital reading takes place literally if we attempt to read a telephone number written on a piece of paper, or peer myopically at a credit card trying to type the number it contains into a web browser. These number are inarguably and literally digital: strings of digits. Most of the numbers we deal with are written using Arabic numerals, which means that they are written out using the digits 0 to 9.

These are the digits that gave rise to the broader category of digitality. Reading them is made easier because there are only ten possible values for each digit and the symbols were chosen to be easily distinguished from each other. They can be misread – for example a badly formed 9 might be mistaken for a 0. But we cannot change their value by making them bigger or smaller as we might do in an analog system of representation. (This observation may strike you as trite, but consider the humble bar chart, in which larger quantities are represented by drawing proportionally longer bars. Or infographics, in which dollar signs of different sizes might be used to represent amounts of money. Analog representations of that kind are good for visualizing quantitative data, but bad for recording it). Neither can we represent a number part way between 1 and 2 by writing down a symbol that looks a bit like a 1 and a bit like a 2. If presented with a squiggle that doesn't clearly map to a valid representation of any of the ten digits we would either guess which it was meant to represent based on context or reject it as unreadable. These characteristics underly the discreteness of digital representations: each digit is constrained to one of ten possible values with no valid intermediate states.

Machines read digitally with sensor mechanism that controls part of the action of the machine. On this level there is no distinction between reading programs and data. That is true whether the mechanism in question directs a loom, increments an accumulator, or transmits an encoding of the information just read, thus transcribing it from one digital format to another. Some part of the machine must change from one state to an-

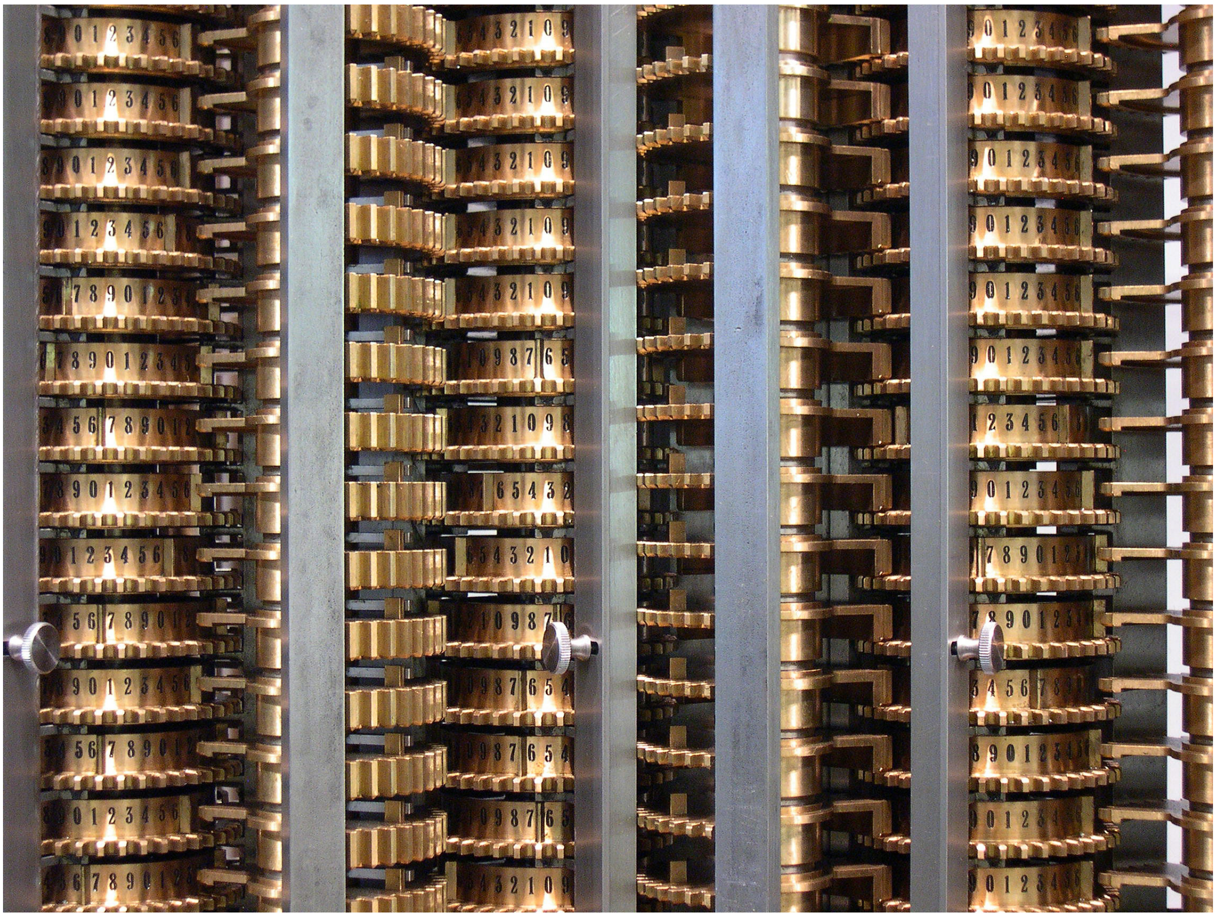


Figure 4: A detail from the mechanical difference engine constructed by the London Science Museum according to the design of Charles Babbage. The position of each wheel encoded a single decimal digit; each column encoded a full number. A full rotation of a wheel caused the wheel above it to advance by one place, carrying 1 to the next digit as it reset to zero. Linkages from between wheels allowed the machine to add together the contents of adjacent columns. Humans read the numbers by looking at the markings on the wheels. The leftmost column was connected to a printing mechanism, able to read and transcribe its value once the computation was complete. Image created by Wikimedia user Carsten Ullrich, used under CC BY-SA 2.5 (https://creativecommons.org/licenses/by-sa/2.5/).

other according to the value being read. That part might be a circuit that fills with current if a hole is punched in a certain position on a card, a hammer that strikes a string in a player piano when a hole is sensed on a roll of paper, or a sensor that changes its resistance in response to the momentary light fluctuations on a fiber optic cable.

Here (in figure 3) is an example of a digital representation intended to be readable by both humans and machines. Punched cards holding numbers were introduced for the 1890 census. The original cards had 45 columns and twelve rows. The card could be conceptualized as containing a single 45-digit decimal number, though in practice cards usually encoded several distinct data fields of a few digits each. Space on the card could be partitioned to code different data fields. Within each field, only one hole was punched – akin to a person representing a digit by folding one of ten digits of their hands. Tabulating machines were configured accordingly, to total the values stored in specific fields from cards that met certain criteria. IBM machines produced from 1928 onwards standardized on a larger, 80 column card.

Numbers on punched cards could be read by humans and by machines, though with different practices. Humans used complicated neural mechanisms to interpret light reflected from the cards. Tabulating machines probed the card with electrical connectors, using the values encoded in particular columns to sort cards into one pile or another or to increment counters.

Although Harry Davis insisted that digital reading involve signals with just two states, *existing* and *not existing,* this is not always the case. Many digital systems, for example, distinguish between ten different states representing the ten decimal digits. Sometimes, as with the punched cards (or with counting on one's fingers) ten different values of a decimal digit are represented with ten different objects, each of which has one of two possible states, such as punched or not punched. In other cases, there may be one object with ten valid states or positions – as with digits written on a piece of paper.

Mechanical adding machines and calculators all had to use some mechanism to represent digits. Most did it with cog wheels of one kind or another, rotating through ten different positions, as seen in figure 4. When a wheel advanced from 9 back to 0 it would push the wheel next to it to advance by one position, performing a carry to the next digit place. Rather than being an "all or nothing" signal, the wheel had ten stable positions.

Analog to Digital Conversion

Digits have such a good fit with processes of tallying that the difference between analog and digital was sometimes expressed as the difference between measuring and counting.[10] If we are attempting to count the number of times the word "digital" appears in this text, or the number of marbles in a jar, the result will be a whole number (technically, a positive integer) which can be expressed in digital form with no loss of precision. In mathematical terms, the thing being counted is itself discrete. The digital representation can capture the quantity perfectly.

Sometimes we must assign digits to approximately represent the value of something continuous. Imagine we are using a ruler to measure the length of an object, or a traditional thermometer to measure a temperature. The thermometer itself is analog: as its temperature rises and falls the fluid within expands and contracts proportionally. To read a thermometer we turn that continuous variation into a number. In both cases we visually compare the length of something continuous to a measuring scale marked out with gradations. We pick the closest marking and record a length as 87mm, an angle as 27 degrees, or a temperature as 39.5 degrees. In doing this we map the analog reality of continuous variance onto whichever number seemed closest. If we have access to a better instrument, or a magnifying glass, we might be able to specify the result to a higher degree of precision, adding digits after the decimal point. But any analog system has inherent physical limits to its precision.

The processes described above are known as analog to digital conversions, a task usually undertaken by electronic systems that translate continuous variation on an input circuit to output pulses that encode digits. Analog to digital conversion is the process of turning a measurement into a number.

Digital systems approximated the continuous variability of the natural world by encoding a finite sequence of digits, each of which was restricted to a predetermined set of possible values (0-9 for decimal, 1 and 0 for binary, and so on). It might seem odd to focus here on the representation of numbers using a finite set of encoded symbols as the original hallmark of digitality. As children learn in school the set of integers is infinite because any number, however large, can be incremented. We should distinguish here between numbers and digits. Each digit has only ten possible values. But two decimal digits together have one hundred possible values, three have a thousand, and so on to infinity. We can always add more digits to the sequence. By introducing a decimal point, sequences of digits can be made to approximate fractions to a finite but arbitrary level of accuracy.

Figure 5: This digital thermometer automates the process of digital reading traditionally carried out by a human peering at the gradations marked along the side of a tube containing mercury or alcohol. Image created by Wikimedia user Hedwig Storch, used under license CC BY-SA 3.0 (https://creativecommons.org/licenses/by-sa/3.0/).

Reading vs. Writing

Why do I call digitality a way of reading, rather than way of writing? Surely the digits extracted from a mechanism, or read from a scrap of paper, had to be written before they were read. Thus, you might suggest, it is the act of writing a message encoded in a finite set of possible symbols, or perhaps the conjunction of writing and reading, that define digitality.

Yet the digits produced by the process of digital reading were not always encoded by a sender. In fact, many digital reading practices capture information from nature. Consider, for example, a digital thermometer – a widely used modern device that is literally digital in the sense proposed by Stibitz back in 1943. It automates the measuring process required to use a conventional thermometer. The thermometer measures the ambient temperature, i.e. the thermal energy of molecules in the environment, and outputs a set of digits. This is a form of digital reading. The digital en-

10 In the mid-1940s Norbert Wiener, the founder of cybernetics, preferred the terminology of measurement device/counting device to analog/digital.

coding used is determined by the machinery, but the content of message comes from nature.

The same is true, on a vastly greater scale, of images produced by digital cameras. Each of the many millions of pixels in the sensor array is measuring the intensity and color of the light falling on it and converting this to a numerical value.

One might attempt to distinguish between digital sensors of this kind, each measuring a single value, and the act of reading which involves looking at a sequence of coded symbols. But sampling values from a sensor at regular intervals, as the analog to digital converters used in digital audio recorders do, will produce a time sequence of values.

Nature includes at least one example of a more complex digital code with no human author. Historians of science have written about the use of information theory by researchers investigating DNA in the 1950s. The very phrase "genetic code" makes assumptions tied to information theory and digital communication. Attempts were made, without much success, to use analysis of this kind to make predictions about how genetic information was stored and, once the role of DNA was clear, how base sequences coded for particular amino acids.

Suddenly chemical sequences without a human author were being treated as a medium, holding a digital message. The title of Lilly Kay's book *Who Wrote the Book of Life* captures her objection to this: researchers viewed themselves as reading a text but were in fact constructing one, bring ideas from information theory that hindered more than they helped.[11] Her point is an interesting one, but subsequent developments in gene sequencing and manipulation suggest that the digital information perspective on the genome eventually became a source of leverage. The six billion nucleotides contained in a genome can be read and transcribed into a data file that fits comfortably inside a modern smartphone. While the connection between that data and human life is not fully understood, it can nevertheless be searched for informational markers signaling traits and disease tendencies.

Thus digitality always involves a practice of reading that maps a continuous range of possible states in the physical world, such as the almost infinite range of actual temperatures, onto one of a finite number of possible states. In some cases, such as reading numbers from a punched card, the effect of this process is intended to be the recovery of information deliberately written to the medium. But in other cases, such as a digital thermometer or digital audio recording, the information captured by the digital reading practice was not deliberately encoded by an author.

11 Lily E Kay, Who Wrote the Book of Life: A History of the Genetic Code (Stanford: Stanford University Press, 2000).

Representing Numbers with Switches

As Harry Davis recognized in his 1949 article popularizing the concept of digitality, the new idea described many earlier technologies but had been introduced to held categorize a proliferation of new ways of encoding numbers using electronic and electromechanical methods. The engineering techniques used to build electronic digital computers have several historical origin points. One is in electronic circuits used to tally, a technique pioneered in the 1920s and 1930s by physicist Charles E. Wynn-Williams for use in nuclear physics instrumentation.

Another origin point, and the one I shall focus on here, is in switching. Automatic telephone exchanges, introduced for local calls in the early twentieth century, received decimal digits as sequences of pulses generated as telephone dials rotated themselves back to their resting positions. The exchange equipment read these pulses digitally, tallying them by advancing its switching equipment to its next position each time a pulse was received. The next digit dialed on the handset, represented as another sequence of pulses, controlled the next switch.

US telephone numbers used three digits to code which exchange within a city the call should be directed, and thus told the local exchange of the caller which cable to switch the connection onto. Once this connection was made, the last four digits set the switches in the destination exchange to complete the electric connection from the caller's telephone line to the telephone line whose number had been dialed.[12] Automatic dialing of calls between cities, which added an additional three optional digits for long distance connections, automatically took a few decades more to become widely established because of the complexity of the task. Switching equipment was bulky. AT&T spread local exchanges throughout the neighborhoods served, and built central exchanges for major cities in large, windowless buildings.

The relay, a switch that turned on and off under electrical control, was invented for telegraphy. Hence the name: relays were first used to boost and repeat signals on long distance lines. But they could also be used to switch telephone calls. In 1937, Claude Shannon was part way through a master's degree in engineering at MIT when he was hired for a summer internship by Bell Labs. His exposure to its network of switching circuits, the most complex in the world, provided him with the subject for his thesis. Shannon had already experienced analog computing, as an

12 The same system had been used with human operators, with the destination exchange specified by name and only the last four digits given numerically. To help in switching between the two methods, which coexisted for decades, letters were printed on the dial and exchange numbers were chosen to correspond with the names of the exchanges to make them easier to remember.

operator of a differential analyzer, but he conceptualized the switching circuits he encountered at Bell Labs in terms of logic rather than numbers. In switching circuits, wires either carried electrical pulses or they didn't. Relays opened or closed. It would be five years until Stibitz, also of Bell Labs, would introduce the terminology of digital and analog. Shannon drew not on numerical mathematics but on mathematical logic, specifically Boolean algebra. He equated switches that were turned on with logical statements that were true, and switches that were turned off with logical statements that were false. The circuits used to interconnect those switches corresponded to the basic logical operators: AND, NOT, and OR. Shannon argued that switching circuits could be converted into logical expressions. Once expressed algebraically the circuits could be manipulated to transform them into the simplest possible representations, which could in turn be mapped back onto circuit diagrams, ensuring that the simplest and most efficient designs would be used. The vocabulary later used to talk about digital electronics: digital logic, logic gates, truth tables, and so on is rooted in this equivalence of digital circuits and logical propositions. Shannon also equated true with 1 and zero with false, providing numerical interpretations of the switches which he showed, in one of his examples, could be used to create a binary adder.[13]

Shannon's thesis has been called the most consequential master's degree thesis in history, though historians have argued against the assumption that this one document can explain a revolution in engineering practice. For one thing, Shannon was not the first or only person attempting to combine logic and circuit design. For another, his method took considerable refinement over many years before it was used for practical purposes by ordinary engineers.[14]

Relay switches of the kind used in some 1930s telephone exchanges and many early digital computers rely on a metal strip to move physically from one position to another, and thus could switch at most a few hundred times a second. That was more than enough to keep up with the speed of a telephone dial, but it put a severe cap on the maximum speed of a digital computer. The Harvard Mark 1 computer, built by IBM and installed in 1944, took three seconds to carry out a multiplication.

Electronic circuits could switch much faster than relays. One of the crucial building blocks of digital electronics is the flip-flop circuit, also known as the latch. This is the electronic equivalent to a relay switch. The circuit has two stable states, meaning that it stores a single bit of information. Its output line carries either a high or low voltage, to allow other circuits to read its content. The information stored in it will persist until a pulse is received on its reset line, which primes it to store the value provided at that instant on its input line. Early digital electronics used two vacuum tubes to produce a flip-flop; later systems used two transistors. Each flip-flop was the equivalent of a single relay switch.

In a sense, analog to digital conversion occurs all the time inside digital computers. Within computers and other digital electronic devices, most digital reading maps sensor data onto a set of just two valid states, typically corresponding to the binary digits 1 and 0 or to true and false. Consider, for example, computer electronics. When data is being moved around insider a computer, voltage levels on a given data or address line rise and fall millions of times every second. We talk of computers being stuffed with 1s and 0s, but those states are actually represented by high and low voltages. Traditionally a 5-volt power supply is used. Ideally the power supply would give a constant output of exactly 5 volts, and logic gates would switch instantly from 5 volts to 0 volts. In practice though, power supplies fluctuate and give only approximate voltages and components do not switch instantly or conduct perfectly. So the manufacturer of a chip might guarantee that it will treat inputs between 5 volts and 2 volts as high, and all inputs of between 0.8 volts and 0 volts. This is called thresholding. The continuous variation of the actual voltage compressed into just two valid states.

Because electronic systems so often rely on reading methods with only two valid values it is common to conflate digital and binary. This is not true, even for electronics. One could, for example, use voltages from 0V to 9V to encode the digits 0 to 9, rounding off to the nearest volt. A value of 2.2V would be rounded to 2, of 4.9V to 5, and so on. But the circuitry required to do this would be far more complex, and far more likely to be read incorrectly. In practice, digital electronic computers have relied almost entirely on two-value encodings, whether or not they use binary arithmetic. Even computers built using ternary (base 3) rather than binary logic and arithmetic still relied on two-value hardware in their memory units and logic circuits. This meant that each trit (ternary digit) was encoded inefficiently as two bits.[15]

Each flip-flop stored a single bit, but the circuits were joined together to store larger numbers. For example, eight flip flops could store an 8-bit binary number, which since the 1960s has been known as a byte. This simplifies the design of computer logic – binary adding and multiplying circuits are trivial in comparison to their decimal equivalents, though using binary

13 Jimmy Soni and Rob Goodman, A Mind at Play: How Claude Shannon Invented the Information Age (New York, NY: Simon & Schuster, 2017)@ch. 4.

14 Maarten Bullynck, "Switching the Engineer's Mindset to Boolean: Applying Shannon's Algebra to Control Circuits and Digital Computing (1938-1958)", in Exploring the Early Digital, ed. Thomas Haigh (Cham, Switzerland: Springer, 2019):87-99.

15 Francis Hunger, SETUN: An Inquiry into the Soviet Tenary Computer (Leipzig, Germany: Institut für Buchkunst Leipzig).

does create extra work to convert output into decimal form for the benefit of humans.

But the same bimodal circuits and switches could also be used to represent decimal numbers. ENIAC, the first programmable electronic computer, was entirely decimal.[16] It grouped together ten flip-flops to represent a single decimal digit, in an assembly known as a "ring counter." Only one of the ten flip-flops was active at a time. Each input pulse to the counter advanced its position by one, for example from 3 to 4. This design was conceptually straight forward – the electronic equivalent of a cog with ten possible positions or a card punched in one of ten holes. But using hardware capable of storing ten bits to store just one decimal digit was inefficient. Other early computers that used decimal, rather than binary, arithmetic packed their digits more effectively, storing each decimal digit in just four bits by coding digits with combinations of active flip-flops. As Davis noted in his 1949 article, this method was far more efficient, allowing IBM's SSEC to represent each decimal number using less than half the number of tubes requirement by ENIAC. IBM continued to use decimal number representations in its computers intended for business use well into the 1960s, and its competitors Univac and Burroughs also released decimal machines.

16 Or at least ENIAC used only the decimal number system and made no use of the binary number system. One can distinguish here between two senses of the word binary. The most general is to describe a choice with only two valid values. For example, the traditional but now disparaged idea of gender. The most common is to describe the base 2 numbering system. Almost all digital electronic logic is binary in the former sense because it is based around components that signal to each other using two valid states. Those signals may or may not represent numbers coded in binary. In talking about digital computers, however, the conventional way of classifying them is according to the numbers coded by these dyadic pulses. Some computers performed their arithmetic on decimal numbers, some on octal numbers, some on binary numbers, and some on hexadecimal numbers.

Conclusion

The modern discourse of digitality has departed quite dramatically from a direct connection with the literal representation of digits. Some so-called *digital formats*, such as those for audio and video, do involve the conversion of analog inputs to encoded numbers but this is rarely what people have in mind when they talk about *the digital* or about *digital cultures*.

In fact, the concept of digitization, while literally extremely appropriate, has rarely been invoked by people discussing processes of quantification as used, for example, by governments to describe their populations. Neither would a *digital historian* be liable to risk confusion with a *quantitative historian* (particularly as the latter are virtually extinct, while the former have recently proliferated).

Yet it is important to emphasize the early and enduring connection of digitality with digits. Digits are digital, whether counted on figures, written on paper, encoded on a punch card or represented by minute electrical fluctuations. By the 1950s, however, the concept of digitality was broadening to include systems of representation based on sequences of symbols of any kind, not just on encoded digits. As I will explore in two further working papers, coauthored with Sebastian Gießmann, this reflected both the evolution of computer technology toward non-numerical applications and the conceptual influence of Claude Shannon's mathematical treatment of communication.

References

Bullynck, Maarten. "Switching the Engineer's Mindset to Boolean: Applying Shannon's Algebra to Control Circuits and Digital Computing (1938-1958)." In *Exploring the Early Digital*, edited by Thomas Haigh, 87-99. Cham, Switzerland: Springer, 2019.

Bush, Vannevar. "The Differential Analyzer. A New Machine for Solving Differential Equations." *Journal of the Franklin Institute* 212, no. 4 (October 1931): 447-488.

Davis, Harry M. "Mathematical Machines." *Scientific American* 180, no. 4 (April 1949): 28-39.

Hunger, Francis. *SETUN: An Inquiry into the Soviet Tenary Computer*. Leipzig, Germany: Institut für Buchkunst Leipzig.

Kay, Lily E. *Who Wrote the Book of Life: A History of the Genetic Code*. Stanford: Stanford University Press, 2000.

Kline, Ronald R. "Inventing an Analog Past and a Digital Future." In *Exploring the Early Digital*, edited by Thomas Haigh, 19-39. Cham, Switzerland: Springer, 2019.

Mindell, David A. *Between Human and Machine: Feedback, Control, and Computing Before Cybernetics*. Baltimore: Johns Hopkins University Press, 2002.

Owens, Larry. "Vannevar Bush and the Differential Analyzer: The Text and Context of an Early Computer." *Technology and Culture* 27, no. 1 (January 1986): 63-95.

Peters, Benjamin. "Digital." In *Digital Keywords: A Vocabulary of Information Society & Culture*, edited by Benjamin Peters, 93-108. Princeton, NJ: Princeton University Press, 2016.

Soni, Jimmy, and Rob Goodman. *A Mind at Play: How Claude Shannon Invented the Information Age.* New York, NY: Simon & Schuster, 2017.

Stibitz, George. "Introduction to the Course on Electronic Digital Computers." In *The Moore School Lectures: Theory and Techniques for Design of Electronic Digital Computers*, edited by Martin Campbell-Kelly and Michael R Williams, 3-16. Cambridge, MA: MIT Press, 1985.

von Neumann, John. "First Draft of a Report on the EDVAC." IEEE Annals of the History of Computing 15, no. 4 (October 1993): 27-75.

Author

Thomas Haigh is a Professor of History and Computer Science at the University of Wisconsin—Milwaukee and visiting Comenius Professor at Siegen University. He is the primary author of *A New History of Modern Computing* (MIT, 2021) and *ENIAC in Action* (MIT, 2016) and the editor of *Histories of Computing* (Harvard 2011) and *Exploring the Early Digital* (Springer, 2019). Learn more at www.tomandmaria.com/tom.

The author would like to thank Ron Kline for the detailed feedback he made on a draft of this paper, and to acknowledge a debt to Erhard Schüttpelz, Sebastian Giessmann, and the participants in our workshop series on digitality for the many conversations that have shaped the ideas within.