philosophies

Article

Turing and Von Neumann: From Logic to the Computer

B. Jack Copeland [1,*] and Zhao Fan [2]

1 Department of Philosophy, University of Canterbury, Christchurch 8041, New Zealand
2 Graduate School of System Informatics, Kobe University, Kobe 657-8501, Japan; fanzhao@people.kobe-u.ac.jp
* Correspondence: jack.copeland@canterbury.ac.nz

Abstract: This article provides a detailed analysis of the transfer of a key cluster of ideas from mathematical logic to computing. We demonstrate the impact of certain of Turing's logico-philosophical concepts from the mid-1930s on the emergence of the modern electronic computer—and so, in consequence, Turing's impact on the direction of modern philosophy, via the computational turn. We explain why both Turing and von Neumann saw the problem of developing the electronic computer as a problem in logic, and we describe their joint journey from logic to electronic computation. While much has been written about Turing's and von Neumann's individual contributions to the development of the computer, this article investigates less well-known terrain: their interactions and mutual influences. Along the way we argue against 'logic skeptics' and 'Turing skeptics', who claim that neither logic nor Turing played any significant role in the creation of the modern computer.

Keywords: history of philosophy; history of logic; Turing; von Neumann; philosophy and history of computing; Entscheidungsproblem; universal Turing machine; logical control; stored-program concept; logical adder

Citation: Copeland, B.J.; Fan, Z. Turing and Von Neumann: From Logic to the Computer. *Philosophies* **2023**, *8*, 22. https://doi.org/10.3390/philosophies8020022

Academic Editors: Joseph E. Brenner and Marcin J. Schroeder

Received: 31 October 2022
Revised: 4 December 2022
Accepted: 5 December 2022
Published: 9 March 2023

> 'The problem of developing a computing machine can be considered as a problem in logic.'
>
> John von Neumann [1] (p. 12)
>
> '[A] computing machine is really a logic machine. Its circuits embody the distilled insights of a remarkable collection of logicians, developed over centuries.'
>
> Martin Davis [2] (p. xii)

1. Introduction

This article concerns the history of philosophy and of logic, in particular the penetration of certain logico-philosophical ideas into computing. We demonstrate the profound impact that those migrating ideas had on the development of the modern computer—and so, ultimately, upon philosophy itself.

It is a truism that discourse based on the computer dominates much of contemporary philosophy and logic. Philosophical areas where discussion pivots around computational ideas range from the ethics of artificial intelligence and the philosophy of the mind and brain to debates about representation, semantics, and indeterminacy, as well as deep questions about human nature and the nature of human intelligence and consciousness. Questions like 'What is a computer?', 'What is an algorithm?', and 'What is a solvable problem?' are philosophically challenging and remain open. In modern logic too the computational turn has changed the landscape. In a sense, then, the logico-philosophical ideas that, in the mid twentieth century, migrated into the field of computer development have come back home again, in enriched form, to shape and inform much of modern philosophy.

Those fundamental logico-philosophical ideas were Turing's. Our analysis will demonstrate his iconoclastic impact on computing, and thereby on philosophy itself. Our findings additionally support a new perspective on Turing: the distinction—customarily drawn or implied—between Turing the computer scientist and Turing the philosopher is an artificial one. His contributions to computing, on the one hand, and to logic, the philosophy of mathematics, and the philosophy of mind on the other, form an organic and interconnected whole.

We revisit an established debate about the extent to which, if at all, ideas from mathematical logic impacted the development of the electronic computer, and we discuss the implications of Turing's work in the philosophy and foundations of mathematics for this debate. One question we address in considerable detail is the crucial one of whether Turing's logical ideas influenced von Neumann, as he pioneered the electronic computer in the US. We present a new analysis and fresh conclusions.

In the course of our analysis of the transfer of this key cluster of ideas to computing, we will explain why both Turing and von Neumann saw the problem of developing the electronic computer as a problem in *logic*. Both made essential use of ideas from this cluster in resolving the question of how to harness the speed of electronics for the purpose of computation. Along the way, we answer the claims of 'logic skeptics', who hold that the origins of the modern computer did not in part lie in mathematical logic, and also the claims of 'Turing skeptics', who hold that the concepts Turing set out in his paper 'On Computable Numbers' [3] were irrelevant to those designing the first electronic computers—and in particular had no influence on von Neumann.

In order to understand fully the impact of Turing's logical ideas upon computing, an awareness of the relevant history is important, and here we present the history in a new way. Thus, this is a paper for both philosophers and historians. To demonstrate our central claims regarding the transmission of Turing's logico-philosophical ideas, we take a deep dive into the history of modern computing. Much of our discussion is based around detailed analyses of archival documents (some not previously discussed in the literature), as we trace out the warp and weft of Turing's and von Neumann's interactions. After briefly backgrounding Turing's relevant work in Section 3, and von Neumann's in Sections 4 and 5, we analyze their joint journey from logic to electronic computation in Sections 6–8. Much has been written about Turing, von Neumann, and their respective contributions to the development of the computer, but our focus here is on a story less explored: the interactions between the two men, and the migration, via these interactions, of ideas from the realm of logic into the world of practical computing. A new picture of Turing's and von Neumann's influence upon one another will emerge. Sections 7 and 8 also present our case against the 'Turing skeptics' and 'logic skeptics', while Section 9 raises some open questions concerning Turing's and von Neumann's interactions.

2. A Philosophers' Stone

Once upon a time, logicians were searching for what Hilbert's collaborator Behmann called a philosophers' stone ('Stein der Weisen'). When found, this would enable its users 'to decide on the provability or unprovability of any mathematical proposition, at least in principle', wrote Hilbert and Ackermann [4] (p. 86).

By means of this philosophers' stone, Behmann said, mathematicians and philosophers would be able to 'solve every problem' in logic and mathematics, and would also be able, if they wished, to 'assign the work of proving propositions to a mathematical labourer'—or even to 'a machine' [5] (pp. 3–4, 6–7).[1] Schönfinkel, another member of the Hilbert group, went even further in describing the powers of the stone. It would not only solve all mathematical problems but could be used for 'solving all problems', he said, because once there is a method for solving all mathematical problems, 'everything else is lightweight, once this "Gordian knot" is undone (since "the world is written in mathematical characters")' [6] (pp. 1–2). It was Behmann it seems who dubbed the search for the stone the *Entscheidungsproblem* or decision problem [7] (p. 164). The 'stone' was an *Entscheidungsverfahren* or decision method—a computational process. The Entscheidungsproblem dominated logic and the philosophy of mathematics in the first part of the 20th century. Ramsey called it 'one of the leading problems of mathematical logic' [8] (p. 264).

Peirce, a strong influence on Hilbert and his group, was the first to consider the decision problem in roughly the form in which Turing tackled it. Peirce devised a decision method for propositional logic and searched in vain for a method for first-order logic [9]. Wittgenstein also tackled the problem, explaining in correspondence with Russell that he had found a method for propositional logic, and conjecturing that his method also applied to first-order logic [10], [11] (p. 515). Gödel, Langford, and others solved special cases of the Entscheidungsproblem, including Ramsey who solved it for the particularly interesting case of the universally quantified formulae he described as representing 'general laws' [8] (p. 272), [12–14]. As Quine summarized matters, 'tests adequate to various important subclasses of formulae' had been discovered [15] (p. 84). The search for the elusive Entscheidungsverfahren for the whole of first- (and second-) order logic ran hot. Hilbert for one was eager to use the Entscheidungsverfahren in connection with various 'challenging epistemological problems', and foremost among these was the question of how to 're-establish mathematics' old reputation for incontestable truth' in light of the recently discovered paradoxes: the emergence of Russell's paradox and other paradoxes of set theory had been, Hilbert said, 'nothing short of catastrophic in the world of mathematics' [16] (pp. 412–413), [17] (p. 160), [18] (p. 169). He believed that arithmetic and analysis would be proved consistent by means of the Entscheidungsverfahren, thereby re-establishing mathematics as the 'paradigm of reliability and truth' [18] (p. 170). His dream was never realised, but discoveries of even greater moment flowed from the search for the stone.

Doubt began to set in regarding the possibility of a general Entscheidungsverfahren. Carnap, for example, said the search for the philosophers' stone 'seems hopeless' [19] (pp. 163–164). But, at that stage, nobody had actually proved the Entscheidungsproblem to be unsolvable. This is where Turing entered the story. The connection Behmann had drawn between the Entscheidungsproblem and a machine designed to carry out a 'computational exercise' [5] (pp. 3, 6–7) occurred also to the young Turing. He took on the Entscheidungsproblem and proved that the philosophers' stone cannot exist [3]. It was one of Turing's major intellectual achievements; but, more importantly, it was his work on the Entscheidungsproblem that gave birth to the fundamental logical blueprint for the modern computer, his 'universal computing machine'.

Looking back on that work, Turing said in 1947:

> Some years ago I was researching on what might now be described as an investigation of the theoretical possibilities and limitations of digital computing machines. I considered a type of machine which had a central mechanism, and an infinite memory which was contained on an infinite tape … [D]igital computing machines … are in fact practical versions of the universal machine. There is a certain central pool of electronic equipment, and a large memory, [and] the appropriate instructions for the computing process involved are stored in the memory.
>
> [20] (pp. 378, 383)

The computer in the form in which we know it today grew out of logical and philosophico-logical ideas. We continue Turing's story in Section 3. Von Neumann, like Turing, also progressed from work on the formal logical analysis of computation to his contributions—fundamental pioneering contributions—to the development of electronic computers. He too was engrossed by the Entscheidungsproblem, publishing important papers on it in the 1920s and early 1930s. As he later struggled with the myriad problems involved in designing an electronic computer, he brought his experience in mathematical logic to bear on practical questions, such as how to design what he called the 'logical control' for a computer and also the 'central arithmetic part', the calculating heart of the machine.

We present a detailed analysis of von Neumann's ideas on computer design in Sections 6 and 7, and we explain there how Turing's ideas impacted them. But first, some curated background.

3. Turing: Logician to Computer Designer

3.1. Attacking the Main Problem of Mathematical Logic

Turing established his philosophical credentials early, giving a talk to the Cambridge philosophers' Moral Sciences Club in 1933 on the inadequacy of logicism as a philosophy of mathematics.[2] His interest in the philosophical foundations of mathematics led him to attend some 1935 lectures by the Cambridge logician Newman, where he had his first significant brush with what Hilbert and Ackermann called 'das Hauptproblem der mathematischen Logik' [4] (p. 77). Newman said later 'I believe it all started because he attended a lecture of mine on foundations of mathematics and logic':

> I think I said in the course of this lecture that what is meant by saying that [a] process is constructive is that it's a purely mechanical machine—and I may even have said, a machine can do it.
>
> And this of course led [Turing] to the next challenge, what sort of machine, and this inspired him to try and say what one would mean by a perfectly general computing machine.
>
> [21]

Sadly, there seems to be no record of what else Newman said at that crucial juncture in his lecture. However, his 1923 discussion 'The Foundations of Mathematics from the Standpoint of Physics' does record some of his related thinking. There he introduced the term 'process' (as used in the above quotation) saying 'All logic and mathematics consist of certain *processes*' [22] (p. 12).[3] He emphasized the requirement that a process should *terminate* with the required result (such as a theorem or number); and he gave a formal treatment of processes:

> The properties of processes are formally developed from a set of axioms, and a general method reached for attacking the problem of whether a given process terminates or not.
>
> [22] (p. 12)

Newman did not mention the Entscheidungsproblem in his 1923 paper, let alone moot its unsolvability; there is no evidence that, pre-Turing, he thought the problem unsolvable. Yet, with hindsight, he certainly laid some suggestive groundwork for an assault on the problem. He wrote:

> The information of the first importance to be obtained about a process or segment of a process is whether it is *possible* to perform it ... The statement that [process] $\Phi \mid \mid \alpha\rho$ is possible means that this process *terminates*: comes to a halt ...
>
> [22] (p. 39)

Newman even proposed an 'apparatus', a 'symbolic machine', for producing numbers by means of carrying out processes of the sort he analysed, and gave a profound discussion of real numbers from the standpoint of this proposal [22] (p. 130ff). The question of the extent to which Newman's detailed ideas provided the context, and perhaps also some of the resources, for Turing's logical breakthroughs demands further investigation.

Newman's lecture on the Entscheidungsproblem was in February or March 1935, and by the summer, Turing's attack on the problem was well under way. He dreamed up what he called the 'universal computing machine', the functional parts of which are: (1) the memory where instructions and data are stored, and (2) the instruction-reading-and-obeying control mechanism. The abstract universal Turing machine is a bare-bones logical model of almost every modern electronic digital computer; his idea of storing instructions as well as data in the memory would lead to a solution of the important practical problem of how to make programmed instructions available to a computer at electronic speed.

Turing presented his results in his famous 1936 paper 'On Computable Numbers, with an Application to the Entscheidungsproblem' [3]. A gem of conceptual analysis, this mixes logical and outright philosophical arguments. The core philosophical arguments in the paper, which Turing referred to as Arguments I, II, III, continue to inspire sophisticated philosophical work (see, e.g., Kripke's recent analysis of argument II [23]). Gandy called the paper 'a paradigm of philosophical analysis' [24] (p. 86). So it was that the two most fundamental ideas in computer science, the twin concepts of the stored program and the universal machine, emerged from a paper concerning logic and the philosophy of mathematics.

At the end of 1936, Turing left Cambridge to study for a PhD with Church at Princeton.[4] Church had also established the unsolvability of the Entscheidungsproblem but by a different method [25]. Under Church's influence, Turing's PhD thesis was notationally dense (Gandy described it as 'a stinker to read' [26]). Nevertheless, its basic orientation was philosophical: Turing's project was to analyse the nature and scope of what he called the faculty of intuition [27]. In 1938 he returned to Cambridge, and 1939 saw him attending Wittgenstein's lectures on the foundations of mathematics—and simultaneously travelling to London to work in secret on Enigma, the German cipher machine [28] (p. 217).

3.2. Turing Meets Electronic Computation

With the declaration of war, Turing took up residence at Bletchley Park to engage full-time with German military codes. In his spare moments, he found the energy to work on logic with Newman (who was still in Cambridge), and they completed a paper on Church's theory of types in 1941 [29]. By late 1943, Turing's celebrated Bombes—logic in hardware—were cracking Enigma messages at the rate of roughly two every minute, an average of 84,000 broken messages each month [30] (p. 29). In terms of technology, the Bombes were based on relays; a staple of the pre-electronic era, the relay was a relatively slow electromechanical switch. Less well-known than Turing's work on Enigma is his attack on a German teleprinter cipher machine manufactured by the Lorenz company [31] (pp. 39–40). This was used by Hitler and the army chiefs in Berlin to communicate directly with their front-line commanders. Breaks by Turing and Tutte led to the construction of Colossus, the world's first large-scale electronic computer. This blazingly fast room-sized digital computer was designed and built by Flowers, with Newman overseeing the design work; the first Colossus went into action in early 1944.[5] During 1944 and 1945, Newman presided over a growing installation of Colossi, nine by the end of the war. It was in effect the world's first electronic computing centre.

As Flowers designed Colossus in 1943, Newman had told him to read 'On Computable Numbers' [32]. It is significant that the relevance of the universal Turing machine to electronic computing was clear in Newman's mind even at that early stage. However, the generality of the universal machine was not required for the cryptanalytical task at hand, and Flowers built a much more limited computer; he did not include Turing's idea of storing coded instructions in memory, employing instead a form of programming he was familiar with, involving setting switches and moving around plugs in a socket-board. This type of programming was satisfactory for machines designed to carry out only a narrow range of tasks, but of no use at all for an all-purpose electronic computer—where Turing's stored-program idea held the key to the problem of how to make the instructions available for every task required of the machine. (Colossus was in fact so far from being an all-purpose computer that even long multiplication lay outside its mathematical repertoire [33].) But Colossus opened Turing's eyes to the potential of electronic computing. Flowers said that, from then on, it was just a matter of Turing's waiting to see what opportunities might come along for putting his concept of the universal computing machine into practice [32].

3.3. Turing's ACE

Turing did not have to wait very long. In June 1945, he was approached by Womersley, who wanted to recruit him to the National Physical Laboratory (NPL) [34] (pp. 39–40). Womersley (recently appointed as Superintendent of the NPL's newly created Mathematics Division) intended that his new organization would 'explore the application of switching methods (mechanical, electrical and electronic) to computations of all kinds'.[6] He had read

'On Computable Numbers' not long after its publication in 1936, and when, during the war, he saw Aiken's experimental relay-based computer at Harvard, he described it as 'Turing in hardware' [35]. Womersley hired Turing to design a computer that would be called the Automatic Computing Engine (ACE); by the end of 1945, Turing's design document 'Proposed Electronic Calculator' [36] was ready [37,38].[7]

Some modern historians find it hard to believe that Turing's abstract universal computing machine had any relevance for the development of high-speed all-purpose computers. Haigh and Priestley, for example, expressed the opinion that Turing machines were 'irrelevant to people trying to design the first real electronic computers' [39] (p. 32). However, that was certainly not how those involved in these developments viewed matters at the time. Sir Charles Darwin, Director of the NPL, wrote as follows about the proposed ACE, which he described as 'a computing machine of very much greater potentialities than anything done hitherto':

> The possibility of the new machine started from a paper by Dr. A. M. Turing some years ago ['On Computable Numbers'] ... The principles he enunciated have now become practicable since it is possible to use electronics in the machine.
>
> [40]

At Bletchley, Newman as well as others in his computing unit (known as the 'Newmanry') certainly understood the relevance of Turing's universal machine. Newman's assistant Michie stated: 'In the Newmanry we were fully aware of the prospects for implementing physical embodiments of the UTM [universal Turing machine] using vacuum-tube technology' [41]. Nevertheless, some have found room to doubt even whether Turing himself, as he was designing the ACE in 1945, considered there to be any connection between his universal machine and the ACE (e.g., [42] (p. 51)). After all, his remarks from 1947, quoted in Section 2, might have been informed by hindsight. Skeptics make the point that he did not so much as mention 'On Computable Numbers' or his universal machine in 'Proposed Electronic Calculator'.[8]

That point, however, can be refuted, thanks to an archival discovery (by Copeland). A fragment of a draft of 'Proposed Electronic Calculator' survived, and there Turing described himself as modifying 'the arrangement in "Computable numbers"' so as 'to give a practical form of machine' [43] (p. 456). The fragment is only three or four typewritten pages in extent, and what further observations Turing might have made, in the surrounding pages, about his universal machine of 1936 is unknown. In the fragment, he explained that the form of memory he described in 1936 'could not be taken over as it stood to give a practical form of machine'. Quite possibly, he would also have made a similar point about the control mechanism of the universal machine. The control arrangements involved a large ontology of configurations, or states, of the mechanism, and for this reason did not make for a practical form of machine.[9] Nevertheless, the essential feature of these arrangements, namely the use of a control mechanism that reads and obeys instructions stored in a memory, was fundamental to the practical development of electronic computers.

A prototype of Turing's ACE (called the Pilot Model ACE) was running by 1950 [34] (p. 71). So successful was it that the English Electric Company decided to copy and market it. Under the name DEUCE, Turing's computer was adopted across the UK, as well as in Australia and Norway.[10] The DEUCE was an able workhorse during the first decades of electronic computing, and some DEUCEs remained in service until about 1970. (Other computers based on the ACE included the highly secret defence computer MOSAIC, the E.M.I. Business Machine, the Packard-Bell PB250, and the Bendix G15, arguably the first personal computer [34] (pp. 368–371).) The large ACE that Turing had specified in 'Proposed Electronic Calculator' was not up and running until after Turing's death [34] (pp. 76–80); an early supercomputer, it was used for artificial intelligence research.

3.4. The Universal Machine: Spreading the Word

Turing himself, then, was one conduit by which his logico-philosophical ideas of 1936 migrated into the sphere of practical computing. We will argue that von Neumann was another conduit: the universal machine concept led him to his scheme for logical control. Moreover, Turing was soon transferring his universal machine concept and other logical ideas directly into the philosophy of mind. In his contributions to a 1949 Manchester Philosophy Department 'Discussion on the Mind and the Computing Machine', chaired by Emmet, he emphasized the importance of the universal machine, pointing out that it is 'capable of turning itself into any other machine' [45]. He also tossed in a key concept from his 1948 manifesto 'Intelligent Machinery' [46], the idea of 'a machine containing neuron-models'. Mays' record of the seminar ended: 'much discussion with TURING ensued'. The following year Turing brought the universal machine concept to a wide philosophical audience, writing in *Mind*:

> This special property of digital computers, that they can mimic any discrete state machine, is described by saying that they are universal machines. The existence of machines with this property has the important consequence that, considerations of speed apart, it is unnecessary to design various new machines to do various computing processes.
>
> [47] (p. 441)

Of Turing's many logical contributions, the universal machine concept is perhaps the most important of all. Von Neumann called it Turing's 'great positive contribution', in a letter to Wiener in 1946. He also said in his letter that 'Turing-cum-Pitts-and-McCulloch' had demonstrated

> that anything and everything Brouwerian can be done by an appropriate mechanism, and specifically by a neural mechanism—even one, definite mechanism can be 'universal'.
>
> [48]

We will return in Section 7 to the significance of von Neumann's remark that the universal machine can take the form a neural mechanism.

4. Von Neumann: Logician to Computer Designer

4.1. Von Neumann's Perspective on the Entscheidungsproblem

Von Neumann's magisterial exposition and critique of Hilbert's ideas, 'Zur Hilbertschen Beweistheorie' (On Hilbert's Proof Theory) was published in 1927 and quickly became prominent [49]. Hardy (Sadleirian Professor of Mathematics at Cambridge) studied it, declaring that he found von Neumann's exposition 'sharper and more sympathetic than Hilbert's own' [50] (pp. 13–14). Von Neumann had a gut feeling that Hilbert's Entscheidungsverfahren could not possibly exist, saying:

> So it seems that there is no way to find the general decision criterion for whether a given normal formula [i.e., a well-formed formula with no free variables] is provable. At present, of course, we cannot demonstrate this. Moreover, no clue whatsoever exists how such an undecidability proof would have to be conducted.
>
> [49] (p. 11)

He continued dramatically:

> The day that undecidability lets up, mathematics in its current sense would cease to exist; into its place would step a perfectly mechanical rule, by means of which anyone could decide, of any given proposition, whether this can be proved or not.
>
> [49] (p. 12)

Presumably von Neumann tried hard to prove his hunch that the Entscheidungsproblem is unsolvable, but, unlike Turing, had no success. With a hint of resignation, he wrote in 1931 that the Entscheidungsproblem was 'far too difficult' [51] (p. 121).

4.2. First Encounter with Automatic Computation

Joining the Princeton Institute for Advanced Study (IAS) in 1933 [52] (pp. 8–13), von Neumann was very ready to make his mathematical genius available to the military of his newly adopted country, and in 1937, the year he was granted US citizenship[11], he became a consultant to the US Army Ordnance Department at the Ballistic Research Laboratory (BRL), a position he retained until the end of the war and beyond [53]. His passion for computation was first aroused by his work on high explosives. A few weeks before Pearl Harbor, in September 1941, he joined Division 8 of the National Defense Research Committee, working on shaped charges, and then in September 1942 moved to the Section for Mine Warfare at the US Navy Bureau of Ordnance; in 1943, the Bureau of Ordnance sent him to England [53]. He arrived in London on 16 February. 'London is certainly very interesting and exciting, and like nothing else on earth', he said the next day in a letter to his wife Klara [54]. Three weeks later, he told her: 'The work here is very satisfactory . . . Things are actually much more interesting than I ever dreamed they would be.' [55]. One new interest was computing.

Thinking about how to carry out complex calculations for gas dynamical problems, von Neumann accompanied Todd of the Admiralty Computing Service in London on a visit to the Nautical Almanac Office (NAO) in Bath [56]. The NAO carried out the calculations for the *Nautical Almanac* and other navigational and astronomical tables. Thanks to the legendary pioneer of mechanical computing Comrie (a New Zealander with a Cambridge PhD in astronomy), the NAO was a cutting-edge centre for scientific computation. Prior to Comrie's arrival there in 1925, 'the Nautical Almanac was computed by retired Cornish clergymen with long white beards, using dog-eared 7-figure logarithm tables' [57]. Comrie progressively mechanized the NAO's computations, installing desk calculating machines and devising increasingly sophisticated numerical methods [58]. By 1933, Comrie had installed a Class 3000 National Cash Register Accounting Machine, which he applied to complex scientific computations [58] (pp. 33–35). Known as the National Accounting Machine, this was an electrically-powered desk calculator with an attached typewriter. Its flexible arrangement of six storage registers offered the potential for a degree of automatic control, and Comrie exploited this to the hilt. With Todd, von Neumann examined the National Accounting Machine and quickly recognised the potential of automatically-controlled computation. Travelling back to London in a blacked-out railway carriage, he and Todd wrote a program for the National Accounting Machine (for interpolating to halves, a special case of Bessel interpolation) [56]. The program was a quaint mixture of instructions to the machine's human operator and specifications of sequences of automatically controlled operations.

Von Neumann's encounter with the National Accounting Machine and his associated work with Todd was a turning point, 'a decisive impulse which determined my interest in computing machines', he said [59]. Not long afterwards, he wrote from London to Veblen in the US, saying he was developing 'an obscene interest in computational techniques' [60]. Von Neumann's metamorphosis into a computer pioneer had begun.

4.3. Getting Up to Speed

He departed from the UK for Washington on 27 June[12] and there continued his consultancies with the Army and Navy ordnance bureaus, working on problems in high explosives, and also aerodynamics; and in August began setting up a project under the aegis of the National Defense Research Committee, to develop new computing methods for military aerodynamical calculations [53,61]. At about the same time, September 1943, he accepted a consultancy for the Manhattan Project at Los Alamos [53].[13] By March 1944, he had developed a new computational approach to problems in gas dynamics [63]. There was strong official interest in his new method [64,65], and he began making dedicated efforts to learn about the state of the art in mechanized computation. With help from Weaver (Chair of the National Defense Research Committee's Applied Mathematics Panel), he got in touch in March with Chaffee at Harvard, whose recent doctoral student Aiken had built the relay-based computer mentioned in Section 3 [66]. Von Neumann found the idea of running his gas dynamical calculations on Aiken's machine 'exceedingly tempting', and later reported 'a very extensive exchange of views with Aiken' [64,67]. Eager also to learn about the Bell Labs relay-based computer and the computing methods under development there, he wrote to Stibitz, again in March [68]. When von Neumann visited, Stibitz (he said) explained 'in detail the principles and the working of his relay counting mechanisms', and they discussed 'the possibilities of these machines compared with the I.B.M. type, and their adaptability to the problems in which I am interested' [69].

In April, von Neumann contacted Schilt, in charge of the Thomas J. Watson Astronomical Computing Bureau at Columbia University [65]. The Columbia Bureau was already using its IBM punched-card machines for carrying out bombing calculations, and the idea of utilizing IBM machines for von Neumann's new method seemed promising. Later in April, he had 'to go west'—to Los Alamos—for three or four weeks [69], where he showed great interest in the IBM punched-card equipment:

> [H]e spent two weeks working in the punched-card machine operation, pushing cards through the various machines, learning how to wire plugboards and design card layouts, and becoming thoroughly familiar with the machine operations.
>
> [62] (p. 351)

By the time of this visit to Los Alamos, von Neumann had built up a detailed picture of current developments in automatically-controlled computing. He passed his knowledge on to the Los Alamos team, suggesting a trial to compare the performance of Aiken's computer to the Los Alamos IBM machines, by running the same problem on both [62] (pp. 351–352). A problem was 'cast in unclassified form' and von Neumann took it to Aiken at Harvard.[14]

In August 1944, von Neumann reported to Oppenheimer, the director of the Manhattan Project, on his ongoing investigations into computer developments [70]. He recommended the acquisition of a Stibitz relay computer, saying this would be '5 times or more faster than any I.B.M. aggregate'. This important report demonstrates the depth of understanding of computing machinery that von Neumann had acquired by this time. His description of the Stibitz computer's instruction format, 'Take the contents of register a, also the contents of register b, add (or subtract, or multiply, etc.), and put the result into register c', was a pre-echo of the electronic era of computing that was about to ensue.

5. ENIAC, EDVAC, and the Princeton Computer

5.1. Von Neumann Meets Electronic Computation

The all-electronic ENIAC, under construction at the Moore School of Electrical Engineering in Philadelphia, was designed to compute at speeds far exceeding those of the relay-based equipment von Neumann had been investigating. He first heard about the use of electronics for computing in the summer of 1944, when Goldstine, a member of the ENIAC group, mentioned he was involved in 'the development of an electronic computer capable of 333 multiplications per second' [71] (p. 182). Von Neumann rapidly got himself invited to the Moore School, where he learned that the principals of the ENIAC team, Eckert and Mauchly, were planning a second, vastly improved electronic computer, EDVAC. ENIAC itself was of the era preceding stored-programming, and to set up the machine for a new job, its operators had to rewire it, by rerouting cables and throwing switches; moreover, ENIAC was optimized for gunnery calculations. EDVAC, on the other hand, would be an all-purpose computer. Von Neumann understood the potential.

5.2. EDVAC

He was soon closely associated with the Moore School group, reporting to Weaver that he had been 'asked to act as their adviser, mainly on the matters connected with logical control, memory, etc.' [67]. Very quickly, he became immersed in electronic engineering. Around the end of August 1944, he was in discussions about electronic memory with Zworykin at the RCA Laboratories in Princeton [72]. In his 'iconoscope', Zworykin had exploited cathode ray tube technology in order to store the images generated by a television camera for a brief period prior to their transmission, and von Neumann wanted to adapt the iconoscope for storing binary digits (bits) and making them available at electronic speeds [73,74].

Von Neumann visited the Moore School regularly for discussions on EDVAC [71] (p. 186), [72]. Warren, the supervisor of the EDVAC project, recalled that the meetings occurred 'perhaps every couple of weeks' [75] (p. 13). (Warren said of von Neumann, 'I never thought of him as an engineer—he was a mathematician and philosopher' [75] (p. 14).) At the Moore School, von Neumann would meet around a blackboard with Eckert, Mauchly, and others, including Goldstine and Burks. Burks, a philosopher, had studied logic under Langford and completed a philosophy PhD in 1941, writing on Peirce [76]. He explained '[M]y interest in logic was a reason for wanting to work on ENIAC' [76]. 'Modern electronic digital computers are logic machines', he said [44] (p. 185). Not that ENIAC was the most coherent of logic machines. Burks remembered that, as von Neumann first became acquainted with ENIAC, he was sometimes puzzled: '[I]t was the way we had designed the ENIAC logically ... this was not a natural way for somebody who had come to machines from pure mathematical logic and a knowledge of Turing machines, as von Neumann had' [76].

One central issue in which von Neumann took a great interest was the problem of logical control—the problem of how, in concept, to arrange for an electronic machine to perform the desired operations in the desired sequence. Turing's 1936 paper was highly relevant to this problem. By February 1945, von Neumann had a detailed solution in mind, writing to Goldstine from Los Alamos: 'I am continuing working on the control scheme for the EDVAC, and will definitely have a complete writeup when I return' [77]. At the end of March, a Moore School report summarizing progress on the EDVAC said:

> Dr. von Neumann plans to submit within the next few weeks a summary of these analyses of the logical control of the EDVAC together with examples showing how certain problems can be set up.
>
> [78] (p. 2)

When completed, this 'summary' ran to over 100 pages. By the end of April, von Neumann had sent Goldstine an advanced draft [79]; he had worked out the logical design of the EDVAC, specifying an overall structure in which memory, the control unit, and the arithmetic unit were separate elements. Goldstine soon reported to von Neumann in a letter: 'All of us have been carefully reading your report with greatest interest and I feel that it is of the greatest possible value since it gives a complete logical framework for the machine' [80]. The report became known as von Neumann's 'First Draft of a Report on the EDVAC' and it set out what is today called the von Neumann architecture. Goldstine said in his 1972 history of the computer:

> In a sense, the report is the most important document ever written on computing and computers ... It is obvious that von Neumann, by writing his report, crystallized thinking in the field of computers as no other person ever did.
>
> [71] (pp. 191, 197–198)

5.3. Taking the Logical Framework to Princeton

Von Neumann's report was also the undoing of the EDVAC group, however. Goldstine rushed to distribute it, without placing Eckert's and Mauchly's names on it alongside von Neumann's, and this precipitated the break-up of the group. In the summer of 1945, von Neumann began planning his own Electronic Computer Project at the Institute for Advanced Study [81,82]. He reported in November: 'The purpose of this project is to develop and construct a fully automatic, digital, all-purpose electronic computing machine' [83]. Von Neumann's eventual machine, known simply as the 'Princeton computer', was dedicated in 1952 [84]. It was not the first of the new stored-program electronic computers (the University of Manchester's 'Baby machine' was running as early as the summer of 1948), but von Neumann's computer was the most influential. Soon a number of machines modelled on the Princeton computer were being built, including the IBM 701, the company's first mass-produced stored-program electronic computer [85]. It was the start of a new era.

Von Neumann's interest in the Hilbert programme seems to have waned after his 1931 paper. But his involvement with computing brought logical matters, and in particular the work of Turing, again to the fore, especially in connection with the problem of logical control. A report he wrote at Princeton in 1946, together with Burks and Goldstine, stated:

> 3.0. First Remarks on the Control and Code: It is easy to see by formal-logical methods, that there exist codes that are in abstracto adequate to control and cause the execution of any sequence of operations which are individually available in the machine and which are, in their entirety, conceivable by the problem planner. The really decisive considerations from the present point of view, in selecting a code, are more of a practical nature: Simplicity of the equipment demanded by the code, and the clarity of its application to the actually important problems together with the speed of its handling of those problems.
>
> [86] (p. 37)

Burks later explained:

> The first sentence of this quotation refers to the development of mathematical logic in the period 1920 through the work of Kurt Gödel and Turing. For the 'codes that are in abstracto adequate ...' are: (1) Gödel's system of bounded quantifiers, and (2) Turing's concept of a state-transition table of a single-purpose Turing Machine when that table is placed on the tape of a universal Turing Machine.
>
> [87] (p. 890)

Turing's work, co-author Goldstine also later emphasised, 'made it very clear that from the point of view of formal logics there was no problem to devise codes [that were] in abstracto adequate' [71] (p. 258).

In the following sections, we present the results of our detailed investigation into Turing's influences on von Neumann and von Neumann's on Turing. Our overall conclusion is that each learned significantly from the other, and that in particular von Neumann found in Turing's universal machine of 1936 the materials needed to solve the problem of logical control.

6. Backdrop to the Transfer of Ideas—Turing's and von Neumann's Intersecting Physical and Intellectual Paths

In this section, we piece together the matrix of intellectual and physical interactions which formed the backdrop to the transmission of ideas between Turing and von Neumann. The timeframe of our analysis extends from the 1930s through to 1949, the date of their last known communication. Of particular interest is what passed between the two—whether through physical presence, or simply familiarity with each other's work—as each became increasingly committed to the development of practical computing machinery.

In 1959, the Earl of Halsbury (a British government official closely involved with the development of electronic computers in the UK) stated that there had been a 'meeting of the late Doctors Turing and Von Neumann during the war', at which computers were discussed [88] (p. 154). This was, Halsbury said, 'a meeting of two minds which cross-fertilized one another at a critical epoch'. Unfortunately, Halsbury did not cite any supporting evidence for the occurrence of this meeting, nor did he even say where or when it was supposed to have occurred. Our new detailed analysis of Turing's and von Neumann's wartime movements establishes that there were certainly opportunities for physical encounters. Whether or not a meeting of the form described by Halsbury occurred, however, the most important route for the transfer of Turing's ideas into von Neumann's deliberations on electronic computer design was the latter's thorough knowledge of Turing's 1936 paper, as we demonstrate in Section 7.

Von Neumann was an important figure for Turing from early on. He requested von Neumann's 1932 book *Mathematische Grundlagen der Quantenmechanik* (Mathematical Foundations of Quantum Mechanics) as his sixth form Mathematics Prize at Sherborne School [89] (p. 39), and would later refer to this work in his 1935 King's College fellowship dissertation [90]. It was in 1935 that the two met for the first time.

6.1. Cambridge 1935

Turing's earliest publication concerned von Neumann's work. When, in November 1934, he starting borrowing from the Cambridge Philosophical Society Library, the first item he took out was the July issue of the 1934 *Transactions of the American Mathematical Society*.[15] This contained von Neumann's paper 'Almost Periodic Functions in a Group. I' [92]. Turing found a way of improving von Neumann's results and soon, in April 1935, his first publication was ready to send off to the *Journal of the London Mathematical Society* [93].

Not long after Newman's 1935 lecture on the Entscheidungsproblem, von Neumann himself arrived in Cambridge [94]. Visiting from Princeton until June, he began delivering his lectures 'Almost Periodic Functions in Groups'[16] on 30 April [95–97]. If Turing had initially borrowed 'Almost Periodic Functions in a Group' with the idea of preparing the ground for the illustrious visitor's arrival, he could hardly have done a better job. But whether Turing planned it or not, the two had topics to talk about. They continued to discuss group theory after von Neumann's return to Princeton, exchanging several letters during the remainder of 1935. By December, von Neumann was calling Turing 'My dear Mr. Turing'. His letter of December 5, many pages long, amiably pointed out 'an essential error' in an argument Turing had given in an earlier letter [98].[17]

Within a few months of von Neumann's Cambridge visit, Turing would devise his groundbreaking proof that—just as von Neumann had hypothesized—'It is generally undecidable whether a given . . . formula is provable or not' [49] (p. 12). Did von Neumann and Turing discuss the Entscheidungsproblem (which was, after all, the main problem of mathematical logic) in the spring or early summer of 1935? Did von Neumann perhaps play a role in Turing's decision to take on this fundamental problem and prove it unsolvable? These are tantalizing questions, and we may never know for sure. But the chronology fits: Gandy said Turing told him 'that the "main idea" of the paper came to him when he was lying in Grantchester meadows in the summer of 1935' [24] (p. 82).

6.2. Princeton 1936–1938

Shortly after von Neumann's Cambridge lectures ended, he wrote to Veblen in Princeton saying that Turing (misspelling it 'Touring') 'will come next year', adding 'he is quite promising' [97]. Turing arrived in Princeton in September 1936, where he continued his work on group theory, writing a few weeks after his arrival: 'I have got one or two things on hand at present not connected with my work in logic, but in theory of groups' [99]. His official supervisor was Church, with whom he worked on logic, but von Neumann's work on group theory was the mainstay of a result Turing submitted for publication in the spring of 1937, in which he cited a result by von Neumann in the crucial first step of his main proof [100] (p. 110).

Turing's association with von Neumann continued. In June 1937, von Neumann wrote 'I know Mr. Turing very well from previous years: during the last term of 1935, when I was a visiting professor in Cambridge, and during 1936–1937, which year Mr. Turing has spent in Princeton' [101]. Perhaps Turing sat in on von Neumann's Princeton lectures on mathematical quantum theory. It is likely they also discussed periodic functions and group theory. Whether they spoke in Princeton of Turing's work on the Entscheidungsproblem is unknown. Von Neumann wrote warmly of Turing's two papers involving his own ideas: 'He has done good work in branches of mathematics in which I am interested, namely: theory of almost periodic functions, and theory of continuous groups' [101]. This endorsement probably reflects the fact that von Neumann was by this time no longer interested in the Entscheidungsproblem—and his interest in computation and the problem of logical control lay several years in the future.

The following year, von Neumann offered Turing a job as his assistant [102]. The salary, at 1500 dollars per annum, was about the same as Turing's fellowship at King's. But, with war in the offing, Turing turned him down and went home to Cambridge.

6.3. Wartime Proximity

We do not know whether Turing and von Neumann met during the war years, but there were certainly opportunities. In November 1942, Turing arrived back in the United States to liaise with cryptologists [103]; during November and December, he spent time in Washington DC, and then from January 1943 worked at Bell Labs in Manhattan, on speech encipherment [104].[18] In Washington, he was based at the US Navy's codebreaking organisation, OP-20-G, which at that time was located in the Navy Department building on Constitution Avenue.[19] (Known also as 'Main Navy', the Navy Department building was a stone's throw from the White House.) At the time of Turing's visit, von Neumann was also a temporary Washington resident, working as a consultant to the US Navy.[20] He was attached to the Bureau of Ordnance, which was located in the selfsame building as OP-20-G.[21] His contract also mentions the Naval Ordnance Laboratory, situated a short distance away in the Navy Yard [105] (pp. 18–19). Surviving Navy memoranda from 1943 and 1944 addressed to von Neumann indicate that, by then at least, he had an office and telephone extension in the Navy Department building.[22]

Von Neumann's visit to the UK from February until the end of June 1943 provided further opportunities for meeting. He was based in London but travelled 'to all parts of England', including Oxford and Cambridge, as he pursued his enquiries into British work on high explosives [106]. In March, Turing returned from the US to Bletchley Park, midway between Cambridge and Oxford and a short train ride from each [89] (p. 72). Von Neumann was in Cambridge and Oxford a lot. On February 28, less than two weeks after his arrival in England, he wrote to Klara saying he had 'spent $2\frac{1}{2}$ days in Cambridge this week' [107]. Another letter to Klara begins 'I had two rather busy weeks, mainly in Oxford and Cambridge' [108]. Von Neumann's rounds of meetings with explosives experts, together with the experiments he was involved in at Oxford and elsewhere [60], evidently left him with enough time for academic socializing in Cambridge. He saw Dirac and Hardy, as well as spending 'a good deal' of time with Taylor, an expert on shock waves at Trinity College, and Fowler, also involved in ordnance research and also at Trinity [106]. He was in touch with Newman too, who would return regularly from his codebreaking activities at Bletchley Park to his house just outside Cambridge. (Newman and Turing worked closely together at Bletchley Park from 1942.) Von Neumann had got to know Newman during his 1935 visit to Cambridge, saying 'he is very attractive both from the topological and from the human side' [97]. At the end of March 1943, he wrote to Veblen that in a few days he was going to Cambridge 'for a weekend with Newman' [106]. The odds of Turing and von Neumann having encountered one another in Cambridge seem reasonably high.

6.4. Postwar: Two Computer Architects

The war over, Turing and von Neumann, on opposite sides of the Atlantic, both tackled the problem of how to design an electronic computer. Womersley, who had visited the Moore School and been shown around the ENIAC, was on the distribution list for von Neumann's First Draft [35,109]. He showed his copy to Turing, who not unnaturally found it of great interest [110], and Turing referred to it while producing his own design for the Automatic Computing Engine. With back-handed generosity, Turing told a newspaper reporter in 1946 that he gave 'credit for the donkey work on the A.C.E. to Americans', no doubt a reference to the First Draft.[23] However, he and von Neumann produced very different designs, von Neumann a machine with the now familiar centralised architecture and central processing unit (CPU), and Turing a decentralised machine without a CPU (its logical and arithmetical functions were distributed among its memory units).

The design proposals Turing set out in his 'Proposed Electronic Calculator' were in fact more concrete than those in the First Draft, which described the EDVAC at a high level of abstraction and hardly mentioned electronics.[24] (Huskey, the engineer whose job it was to draw up the first detailed hardware designs for the EDVAC, said he found the First Draft to be of 'no help' [112].) Turing, on the other hand, gave detailed specifications of the hardware, and also included sample programs in machine code. Turing and von Neumann kept in touch during this period, as did Newman and von Neumann. In 1946, Newman wrote to von Neumann: 'with the development of fast machine techniques mathematical analysis itself may take a new slant, apart from the developments that may be stimulated in symbolic logic', adding 'I am of course in close touch with Turing' [113].

6.5. Princeton 1947

In January 1947, Turing visited von Neumann in Princeton for a week or more [71] (p. 218), [114] (p. 288), [115]. Computational mathematics was discussed, in particular a key matter about which Turing and von Neumann both proved results, the problem of errors accumulating due to 'rounding off' numbers as a calculation progresses [114] (p. 288).[25] Turing was supposed to be giving a series of lectures in London at this time but evidently considered talking to von Neumann more profitable, and cut his own lectures (leaving his assistant Wilkinson to give them). In his official report, Turing summed up: 'My visit to the U.S.A. has not brought any very important new technical information to light, largely, I think, because the Americans have kept us so well informed during the last year' [115]. So far as is known, that was the last time Turing and von Neumann met. Turing ended his report of his visit by saying he would 'try to keep in touch' with the Princeton group, but if he did so, little evidence of ongoing communication has survived. In what appears to be von Neumann's last letter to Turing, in 1949, he said:

> Our machine project is moving along quite satisfactorily, but we aren't yet at the point where you are.
>
> [116]

Did Turing's universal machine impact von Neumann's thinking on how to design an electronic computer? As well as being an important question in the history of logic, this is perhaps the paramount question in the early history of electronic computing. Clearly there were opportunities for von Neumann and Turing to have discussed 'On Computable Numbers' in person; whether or not they did so, it is indisputable that von Neumann was well aware of Turing's paper at the time of his engagement with electronic computing. Is there evidence that his knowledge of the universal machine concept contributed significantly to his thinking about computer design? We will argue that the answer is clearly yes, and that there were therefore two conduits for the migration of Turing's logico-philosophical ideas of 1936 into computing—Turing himself in the UK, and von Neumann in the US.

7. Impact on von Neumann of Turing's Universal Machine Concept: The Evidence

7.1. Testimony and Skepticism

Several of those who worked with von Neumann during the crucial period—which is to say, from 1943, with the onset of his interest in computational techniques, through to the development of the Princeton Electronic Computer Project—have put on record statements about the place he accorded Turing's 'On Computable Numbers'. For example, Bigelow, chief engineer at the Princeton computer project, said that von Neumann's understanding of the stored-program idea was due in part 'to the work of A.M. Turing' [117] (p. 23). (Just as Newman had told Flowers to read 'On Computable Numbers', Bigelow stated that von Neumann told him to read Turing's paper at an early stage of the Princeton project [52] (p. 178), [118].) Turing's universal machine 'was the germinal idea', Bigelow said [117] (p. 24). Frankel, von Neumann's collaborator at Los Alamos on the machine computation side of things, said: 'I know that in or about 1943 or '44 von Neumann was well aware of the fundamental importance of Turing's paper of 1936' [119]. Frankel went on: 'Many people have acclaimed von Neumann as the "father of the computer" . . . but he firmly emphasized to me, and to others I am sure, that the fundamental conception is owing to Turing—insofar as not anticipated by Babbage, Lovelace, and others.' In Frankel's view, 'von Neumann's essential role was in making the world aware of these fundamental concepts introduced by Turing'.

Nevertheless, some modern writers doubt the influence of 'On Computable Numbers' on von Neumann. As recently as 2021, Denning and Tedre described the view that 'the stored program idea was the implementation of Turing's universal machine' as 'folklore' [120] (p. 99), and De Mol deemed this and similar views a 'side effect' of computer science's 'need for heroes' [121] (p. 6). We have already examined Haigh and Priestley's statement in 2020 that the universal Turing machine was 'irrelevant to people trying to design the first real electronic computers'—including, they emphasized, von Neumann [39] (p. 32). These Turing skeptics give short shrift to statements from von Neumann's contemporaries about Turing's impact on him, including the above quotations from Frankel and Bigelow. Daylight argued:

> Such recollections should be handled with care because they come from people who participated in a success story that was already several decades old and were in no better position than most of us today to identify what only very few men like Turing and von Neumann knew.
>
> [122] (p. 38)

7.2. New Evidence

A crucial missing link, namely the lack of any mention of Turing's 'On Computable Numbers' in von Neumann's known works on computer design, certainly helped to fuel persistent doubts about Turing's influence. Recently, however, some fresh evidence has turned up, thanks to a lucky archival find (ironically by a Turing skeptic, Priestley). This is a typescript [1] headed 'High Speed Computing by John von Neumann' (HSC). (Extracts from the typescript are shown in Figures 1–3, below.) HSC consists of text and diagrams for three consecutively numbered lectures (all untitled). It seems that von Neumann himself did not prepare the typescript. A number of errors and misspellings are unlikely to be his—e.g., 'scalon' is consistently typed for 'scalar', 'imputs' for 'inputs', 'finary' is typed for 'binary', and 'incints' for 'circuits'; and mathematical errors include a nonsensical confusion of the variables i and j. Perhaps 'High Speed Computing by John von Neumann' is a record of his lectures written up by somebody else; or perhaps it is the typed version of a handwritten resumé or lecture guide prepared by von Neumann. Either way, HSC forms key evidence and is as close to utterances directly from von Neumann's own lips in 1944–1945 as we are likely to get.

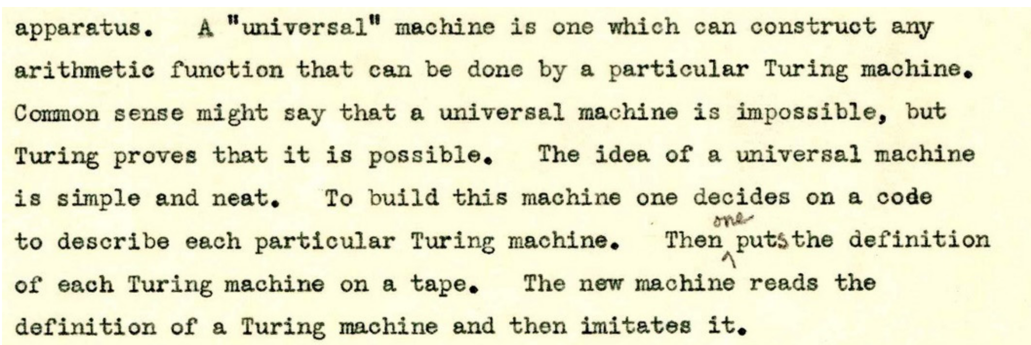
apparatus. A "universal" machine is one which can construct any arithmetic function that can be done by a particular Turing machine. Common sense might say that a universal machine is impossible, but Turing proves that it is possible. The idea of a universal machine is simple and neat. To build this machine one decides on a code to describe each particular Turing machine. Then one put~~s~~ the definition of each Turing machine on a tape. The new machine reads the definition of a Turing machine and then imitates it.

Figure 1. Von Neumann described Turing's universal machine in his third lecture. Courtesy of the American Philosophical Society.

The lectures are undated (a matter to which we return in Section 8) but appear to belong to those exciting months immediately following von Neumann's report to Weaver, on 1 November 1944, that he had been asked to act as the EDVAC group's 'adviser, mainly on the matters connected with logical control, memory, etc.' [67]. Lectures 1 and 2 ranged over the various technologies he had encountered during the explorations set off by his sudden interest in computational techniques (human computation, desk multipliers, punched card machines, Aiken's computer at Harvard, the analog differential analyser at MIT, and the Cambridge Mallock machine, which he probably saw during his 1943 visit to the UK). He compared the performance of electro-mechanical relays with vacuum tubes; the former, he pointed out, were cheap but slow. Von Neumann clearly intended to leave his listeners in no doubt that vacuum tubes were the best means to achieve high-speed computing. He did not mention the EDVAC by name but hinted at its abilities, saying that 'a time of .001 seconds for the multiplication of two ten-integer numbers can be expected with existing objects of computing' [1] (p. 1). (This is the figure he gave in a number of memos for the EDVAC's proposed multiplication time [123,124].) A significant theme in his first two lectures was the problem that he and Turing later both addressed, the accumulation of error due to the 'necessary rounding of numbers' [1] (p. 7).

In Lecture 1, explaining the requirements for high-speed computing, he emphasised the necessity for a large fast memory, and also a solution to the problem of logical control in a fast machine. 'Instructions written on a sheet of paper furnish the logical control for the use of a desk multiplier', he said; 'the need for the development of logical control has held back the advance in "fast" computing' [1] (p. 3). Turing and his universal machine figured in Lecture 3, whose principal topic (von Neumann explained in the first paragraph) was precisely logical control. He gave Turing's 'Logical Machine' star billing, along with Pitts's 'attempts to describe the human nervous system in terms of logic' [1] (pp. 12, 14). Here, von Neumann was referring to Pitts's contributions to his foundational paper with McCulloch, showing how to assemble logic circuits from simplified neurons [125]. The McCulloch–Pitts paper (published in December 1943) painted a novel picture of Turing machines, presenting these as mechanisms assembled out of McCulloch–Pitts neurons. (In fact, Turing machines formed the underlying topic of the McCulloch–Pitts paper, as McCulloch later emphasized (during von Neumann's 1948 Hixon Symposium lecture): 'What we thought we were doing (and I think we succeeded fairly well) was treating the brain as a Turing machine' [126] (p. 319).)

7.3. Von Neumann's Lecture 3

In an introductory passage to Lecture 3, von Neumann set the scene: 'We shall consider two systems of logic which could be used in building a computing machine' [1], (p. 12). The two systems are Turing's and Pitts's, and the lecture sketched how these might be used together to solve the problem of logical control. (Notice by the way the construction-oriented language: *used in building* a computing machine.) Von Neumann briefly outlined the Turing machine, and the lecture's first illustration depicted the tape and the scanner with its control mechanism; and then he described the universal machine (Figures 1 and 2):

> Common sense might say that a universal machine is impossible, but Turing proves that it is possible. The idea of a universal machine is simple and neat. To build this machine one decides on a code to describe each particular Turing machine. Then one puts the definition of each Turing machine on a tape. The new machine reads the definition of a Turing machine and then imitates it.
>
> [1] (p. 13)

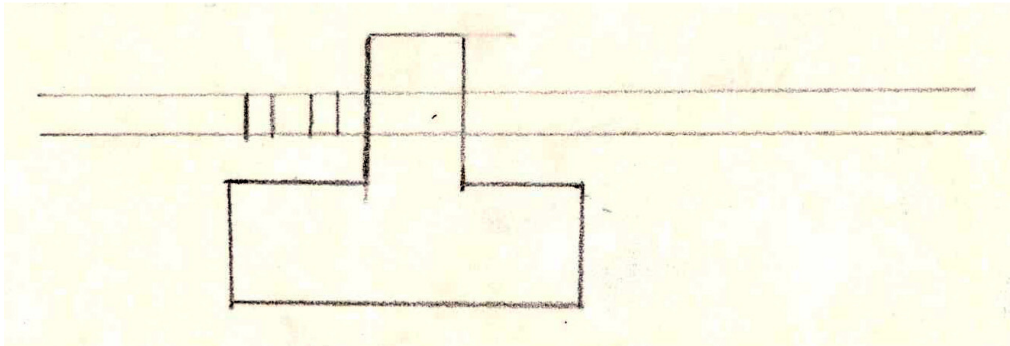

Figure 2. Von Neumann introduced what he called Turing's 'logic machine' at the start of Lecture 3, using this diagram (Figure 1 of the lecture) depicting the tape and the scanner with its control mechanism. Courtesy of the American Philosophical Society.

Von Neumann moved on after this to a discussion of Pitts's work, saying 'His [Pitts's] idealized neuron could be imitated by a circuit of vacuum tubes' and 'Hence, we are interested in his developments' [1] (p. 14). The connection von Neumann was drawing between the universal machine on the one hand, and Pitts's idealized neurons and vacuum tubes on the other, is not stated explicitly in the lecture, but his line of development is easy enough to follow. The relevance of the universal machine to the problem of logical control seems clearly to be that (as Bigelow later expressed it) 'Turing['s] . . . universal computation machine, would obey any set of orders' [117] (p. 24). And in the wake of the 1943 McCulloch–Pitts paper, the universal Turing machine's control mechanism could be taken to be a network of idealized neurons. Gandy put the point clearly:

> McCulloch and Pitts asserted, justifiably, that the control mechanism of the universal Turing machine could be simulated by a finite assembly of idealized neurons.[26]
>
> [24] (p. 87)

Lecture 3, then, appears to be presenting this solution to the logical control problem: von Neumann is arguing that the problem is to be solved by constructing some fixed mechanism (a 'permanent apparatus') akin to the instruction-reading-and-obeying control mechanism of Turing's universal machine, and that the way to build such a device is to use Pitts's neurons implemented as vacuum tube circuits.

The Entscheidungsproblem led to the universal machine, and the universal machine led to this powerful idea for achieving logical control, which was crucial in the development of the modern computer. As we shall explain, von Neumann soon made use of his Turing–Pitts recipe for dealing with logical control.

7.4. Logical Control in the First Draft

Discussing logical control in the First Draft, von Neumann explained that a 'system of orders [is] used in controlling the device' (i.e., the computer), and that the function of the control mechanism (which he termed CC for 'Central Control') 'is to receive these orders' and 'to carry them out, or to stimulate properly those organs which will carry them out' [74] (pp. 84–85). In Turing's universal machine, the orders, or instructions in his terminology, were stored in the memory, together with data the instructions operated upon. The instructions were read by the Turing machine's control apparatus and obeyed. Von Neumann employed much the same arrangement: the orders received by CC 'come from M [the memory], i.e., from the same place where the numerical material is stored' [74] (p. 85).[27]

The fundamental idea of Turing's universal machine was that the specific instructions for carrying out a particular task were to be distinct from a control apparatus governed by its own table of instructions, and the instructions in that table enabled the control apparatus to read and obey any sequence of instructions placed in the computer's memory. Von Neumann introduced CC in exactly that way:

> If the device [the computer] is to be elastic, that is as nearly as possible all purpose, then a distinction must be made between the specific instructions given for and defining a particular problem, and the general control organs [CC] which see to it that these instructions—no matter what they are—are carried out.
>
> [74] (pp. 3–4)

7.5. The Turing Skeptics

Returning to the skeptics, the following view of Turing, expressed here by McCarthy, is a motif in skeptical narratives: '[I]t does not seem that the work of . . . Turing . . . played any direct role in the labors of the men who made the computer a reality' [127] (p. 68). Of course, McCarthy's statement is refuted by Turing's own labours to make the computer a reality. Another version of the motif was expressed by Burks: 'Turing's work had absolutely no influence on the design of the Von Neumann Machine' [44] (p. 187); 'Turing's concept of a computing machine, described in his 1936–1937 paper' lay, Burks said, 'outside that causal chain' leading to the 'twentieth century revolution in electronic computers' [44] (p. 193). We think Burks was simply wrong about that. Daylight, developing the skeptical narrative further, said the universal Turing machine and the computer form 'a notable example' of ideas that were not 'understood as connected' at the time but were 'retrospectively integrated' by historians and computer scientists [128] (p. 220). Mounier-Kuhn said the idea that the universal Turing machine and the computer were linked is 'a founding myth of computer science, an a posteriori reconstruction' [129] (p. 25). Daylight and Mounier-Kuhn are far from alone in this view; see, e.g., [42] (p. 51), [121] (p. 6), [130] (pp. 19–20), and [131] (p. 243) (and for further assertions of Turing's lack of influence on practical developments see, e.g., [39] (p. 32), [120] (p. 99), [132] (p. 41), and [133] (p. 4)). Our argument is that a careful reading of HSC and the First Draft shows that skepticism concerning Turing's influence on von Neumann's thinking is ungrounded.

7.6. Sharing a Vision and Learning from Each Other's Work

The correct view of matters seems to be that Turing and von Neumann had a shared vision of an electronic universal machine and each learned significantly from the other. Just as the First Draft was influenced by Turing's 'On Computable Numbers', Turing's 'Proposed Electronic Calculator' was influenced by the First Draft. (Turing in fact recommended that it be read 'in conjunction with' the First Draft.) Turing followed von Neumann in employing the Pitts–McCulloch neuron notation (although extended it considerably), and also followed von Neumann in using Pitts neurons as the conceptual building blocks of the computer. Both conceived of an electronic computer as a control mechanism acting in tandem with a memory in which instructions were stored. Turing wrote: 'The logical control (LC) . . . is the very heart of the machine' [36] (p. 373).

Perhaps it was from Turing's reading of the First Draft that he realised circuit elements corresponding to Pitts neurons offered an effective route to a 'practical form' of his 1936 idea of a control mechanism accessing task-specific instructions stored in memory. But, if so, 'Proposed Electronic Calculator' nevertheless went far ahead of the First Draft in its discussion of logical control. The First Draft ended without giving details of how the logical control was to be implemented.[28] Burks said that his, Goldstine's and von Neumann's Princeton report 'Preliminary Design of an Electronic Computing Instrument' (dated 28 June 1946) contained 'the first description of a Program Control for a universal electronic computer' [44] (p. 181), but this is incorrect. It was Turing's 'Proposed Electronic Calculator' that gave the first thoroughgoing description of logical control [36] (pp. 382–383, 393–397, and 436–437).[29]

Did Burks never read 'Proposed Electronic Calculator'? If he did, he had certainly forgotten it when, at age 87—and still an influential figure—he endorsed Turing-skepticism (above) and relegated Turing to the role of mere theoretician: 'Turing's theory of imaginary computing machines was the first theory of digital computing machines, while the EDVAC and Von Neumann Computer [at Princeton] were the first practical universal computing machines' [87] (p. 891).[30] He meant, of course, the first practical universal computing machines on paper (since the physically constructed versions of the EDVAC and the Princeton computer were both preceded by a number of other such machines, starting with the Manchester computer in 1948 and including the Pilot ACE in 1950). In fact, the first practical universal computing machines on paper were, in order of appearance, the EDVAC and Turing's ACE. Interactions between von Neumann's thinking and Turing's were, we have argued, of paramount importance to these practical developments. Each of them played his role as a conduit from Turing's 1936 paper into the practical world.

8. Using Logic to Automate Arithmetic

In this section, we outline the origins of the actual calculating parts of the EDVAC and the ACE, which both von Neumann and Turing called the 'central arithmetic part', and both abbreviated to CA. Von Neumann opened Lecture 3 with the statement quoted at the beginning of this article: 'The problem of developing a computing machine can be considered as a problem in logic.' Logical control was one part of the problem. Another was how to build CA. In von Neumann's and Turing's solutions, we again see concepts moving from mathematical logic into the practical arena. Also in this section, we position the content of von Neumann's Lecture 3 on the trajectory (outlined in Sections 4 and 5) of his developing ideas about computation.

8.1. The Central Arithmetic Part

CA is the core of the computer, and a principal function of the logical control is to pass instructions on to CA. Turing said 'There is no very distinct line between LC and CA'; and (as we saw previously) von Neumann described the function of CC as being either to carry out the orders itself or, if appropriate, 'to stimulate properly' the part of the computer that 'will carry them out', typically CA [36] (p. 373), [74] (p. 85). It is therefore no surprise that von Neumann's lecture on logical control also discussed arithmetical circuits.

Following an introduction to what Pitts and McCulloch had called their 'logical calculus'—the second of the 'two systems of logic' Lecture 3 focussed on—von Neumann turned to the problem of how to build a binary adder. This would become the fundamental element of the arithmetic part of the computer. Figure 7 of Lecture 3 displayed his solution to the problem of designing an adder (see our Figure 3).

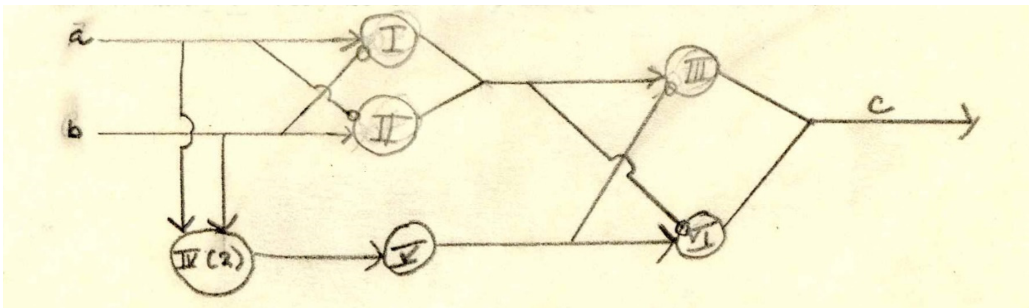

Figure 3. This logic diagram from von Neumann's Lecture 3 (Figure 7 of the lecture) shows a binary adder. The numbered circles are neurons; small circles represent inhibitory inputs into the neurons; and arrowheads represent stimulating (i.e., energizing) inputs into the neurons. 'a' and 'b' represent the two bits that are to be added. With the exception of neuron IV, the threshold of each neuron is 1; neuron IV's threshold is shown as 2. In fact the diagram is not quite accurate: von Neumann explained clearly in the lecture that neuron IV is supposed to obtain 'one input from itself'; but whoever drew the diagram omitted the essential arc from IV back to IV. Courtesy of the American Philosophical Society.

Von Neumann applied existing logical knowledge very directly to the adder problem: his adder consisted essentially of two neuron-circuits, each of which implemented the truth-table for exclusive disjunction. These two either–or units worked together, the first (neurons I and II in Figure 3) executing the main part of the binary addition, and the second (neurons III and VI in Figure 3) taking care of any 'carry'.

8.2. Dating the Lecture Content

The adder is important in dating the content of von Neumann's lecture, since we can connect the adder with events in our previous chronology of the development of von Neumann's ideas. While working with Goldstine's nachlass, we discovered a page of pencilled notes, in von Neumann's hand, containing the diagram in Figure 4 [134]. This depicts the same adder that was presented in Lecture 3. As the diagram shows, von Neumann referred to the adder as '5t'. Probably 't' abbreviated 'tube': the diagram depicts 5 neurons, and von Neumann appears to have been assuming that each neuron could be realised by a single tube (in fact an impractical assumption, as will soon be explained). The pencilled notes also include diagrams of two other adders, labelled '4t' and '14t', depicting 4 and 14 neurons respectively. A previous page of these pencilled notes included the legend 'L A December 2, 1944'. 'L A' presumably abbreviated 'Los Alamos'. But whether 2 December was also the precise date of the adder diagram is uncertain.[31]

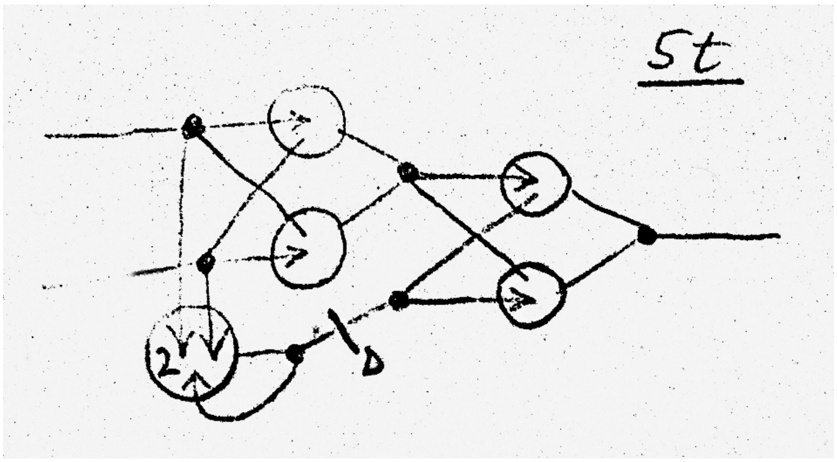

Figure 4. This logic diagram from von Neumann's 'LA Notes' shows the Lecture 3 adder. Again, stimulating inputs into the neurons are indicated by an arrowhead, and in this diagram inhibitory inputs into the neurons are indicated simply by the absence of an arrowhead. The device labelled 'D' (for 'Delay') has been replaced in the Lecture 3 diagram (Figure 3) by Neuron V. D's function is simply to delay the signal, so the 'carry' digit arrives at the right moment; and neuron V in the Lecture 3 diagram performs the same function. Notice, by the way, the returning arc on the bottom left neuron, missing from the incorrectly drawn Lecture 3 diagram. Courtesy of the American Philosophical Society.

The 'LA Notes' (to coin a name for these five sides of pencilled notes), while compressed, and clearly intended only for von Neumann's own eyes, are of great interest. The early logical designs in these notes for the all-important electronic binary adder are among computing's (already vast) debts to von Neumann, but it is not usually recognized in the literature that he played a leading early role in adder design. These pages of notes, moreover, convey the breadth, and also the depth of detail—and perhaps even something of the excitement—of von Neumann's thinking during the early phase of his research on how to build an electronic computer.[32]

A different document enables us to make a further connection between the 5t adder of the LA Notes and other events in our chronology of von Neumann's involvement in computing. Miraculously, the written minutes of four of his 1945 meetings at the Moore School still survive. Taken by Burks, these cover meetings in March and April. On 23 March, Burks reported that 'Adder units were discussed' [136]. Von Neumann, he said, 'suggested a scheme in which the tubes would be used as inhibitors only'. This description fits the adder labelled 14t, where the notation made clear that all the inputs into the neurons were inhibitory. The minutes further relate that von Neumann also 'suggested a couple of schemes in which tubes were both inhibited and energized and involving five to eleven and four to seven tubes'. Both 5t and 4t contain a mix of inhibitory and energizing (stimulating) connections. Von Neumann seems, therefore, to have presented all three of 14t, 5t and 4t at this meeting. Burks has scaled up von Neumann's tube counts; in an interview 30 years later, he explained why. At the meeting, von Neumann had

> announced cheerily . . . that he could build a serial adder with 5 tubes, and Pres [Eckert] said, 'No, it would take at least 10 tubes'. Von Neumann said, 'I'll prove it to you', so he went to the board and drew his 5 tube adder. Now, this 5 tube adder was logically correct, but our immediate response was, 'Well, that won't work, because this particular tube that you have, while it does the work logically, does not have enough power to drive the next tube in the time that's allowed. You must therefore put in another tube. That will change the polarity and call for still another tube'. And so the discussion went, and after a few minutes he was convinced . . . [H]e said 'You are right, it does take 10 tubes to add—5 tubes for logic and 5 tubes for electronics'.
>
> [76]

In summary: whether or not the date on the first page of von Neumann's LA Notes (2 December 1944) was also the date on which he devised 4t, 5t and 14t, we know from Burks' account that von Neumann presented these three adders at the Moore School in March 1945. (4t in fact went on to appear in a more august venue: it was the adder von Neumann selected to be included in his First Draft (Ref. [74], his Figure 3), no doubt because it required fewer tubes.) It appears, therefore, that Lecture 3's content belongs to that exciting period, during late 1944 and the first months of 1945, when von Neumann was beginning to formulate his ideas on how to design an electronic computer. It was during this period that he took the first steps towards producing his design for CA, and applied his knowledge of the universal Turing machine to the problem of logical control.

8.3. Critiquing Some Bad Arguments on the Date Issue

Next, we briefly consider some arguments, due to Haigh and Priestley, that in effect challenge our positioning of Lecture 3's content in time. They argue that the lecture belonged to a later period in the development of von Neumann's thought, and say they are 'confident' that HSC dates 'from the summer or fall of 1945' [39] (p. 31). Their arguments are an attempt to shore up Turing-skepticism, since by that time von Neumann had already set out his solution to the logical control problem in the First Draft, and his thoughts had by then turned to the design of what would become the Princeton computer.

Haigh and Priestley present three arguments, which we will label *A*, *B* and *C*. Arguments *A* and *B* both concern a letter from Goldstine to von Neumann dated 15 May 1945 [80]. Haigh and Priestley argue that the content of Lecture 3 must postdate this letter. They say they

> examined a letter in which Herman Goldstine sent von Neumann comments on the text of the First Draft, including a sketch of the design of 'an adder that Pres [Eckert] and John M[auchly] are patenting.' Von Neumann's adding circuit in the lectures (see Figure [3 above]), closely resembles this sketch. Goldstine also suggested some changes to the notation, such as adding arrowheads to the lines linking the neuron symbols. These appear in the lecture but not in the First Draft, the text of which was not altered before is was [sic] circulated in June. These facts suggest that the lecture was written after von Neumann received Goldstine's letter in mid-May 1945.
>
> [39] (pp. 30–31)

Goldstine's sketch (Figure 5) shows his attempt to redraw Eckert and Mauchly's vacuum-tube circuit in McCulloch–Pitts neuron notation.[33] Argument *A* is that von Neumann's diagram 'closely resembles' Goldstine's [39] (p. 30), which suggests (Haigh and Priestley say) that von Neumann's lecture diagram originated later. Clearly, this argument is invalid: a resemblance, no matter how close, cannot tell us which was first. In any case, the argument's premiss that von Neumann's adder closely resembles Goldstine's cannot be sustained. For one thing, Goldstine's adder is incapable of adding, because of five fatal errors he introduced when trying to deal with what Eckert called 'corrective delays' [137] (p. 92). (Haigh and Priestley appear not to have noticed this.) Furthermore, superficial examination shows the adders to have different numbers of and-gates (i.e., neurons with threshold 2), and feedback loops in different places; and probing more deeply, there are significant differences in the way the 'carry' logic is handled.[34] These diagrams in fact represent two rather different approaches—by von Neumann on the one hand and Eckert and Mauchly on the other—to solving the problem of designing a tube-based binary adder.

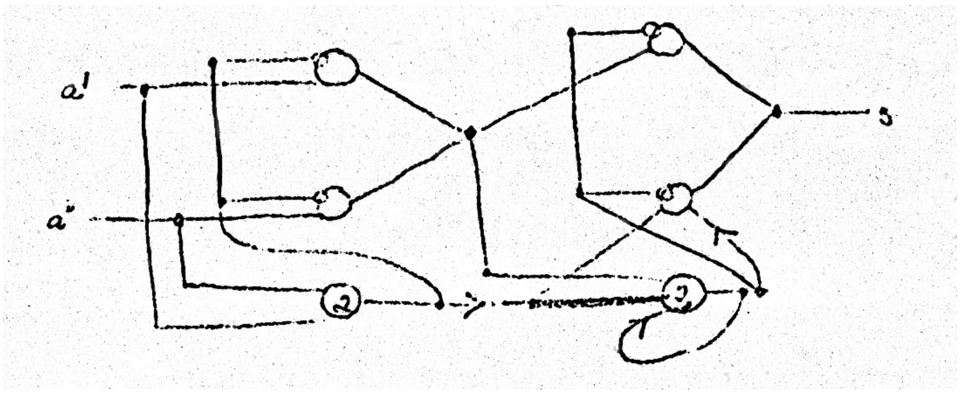

Figure 5. Goldstine drew this adder in what appears to be a draft version of a letter. Whether the letter was actually sent to von Neumann is uncertain. Courtesy of the American Philosophical Society.

Argument *B* is that (as claimed in the above quotation) modifications to von Neumann's notation suggested in Goldstine's letter were taken up in Lecture 3—specifically, Goldstine's suggestion to add arrowheads to lines between neurons to indicate direction. However, von Neumann did *not* adopt this convention in Lecture 3, where there was no specific notation to indicate direction. In fact, he said clearly in Lecture 3 that he was using the arrowhead to indicate something else, namely that the input was 'stimulating' as opposed to 'inhibiting' (see also Figure 4)—a notation very different from the one Goldstine proposed in the letter. In summary: nothing about Goldstine's letter suggests it must have predated the content of Lecture 3.

Argument *C* concerns a diary-like notebook kept by Mooers, an engineer at the Naval Ordnance Laboratory [139]. (Mooers was part of a project at the laboratory to build an EDVAC-like computer for Navy use, and he described von Neumann as 'the guy that convinced the Navy to give us the computer project' [140] (p. 60).) His notebook recorded that, on 26 October 1945, he and von Neumann travelled by train from Philadelphia to Princeton, where 'v. Neumann lectured on his tentative coding scheme for the computer'. This was clearly a different lecture from Lecture 3, which contained no mention of a coding scheme. Nevertheless, Haigh and Priestley said that Mooers' notebook entry 'builds our confidence' concerning their dating of Lecture 3. The reason they offer for this increased confidence is that Mooers described (in their words) 'von Neumann spontaneously launching into similar material in October' [39] (p. 32). Here, they are referring to a brief statement in the notebook entry for 26 October that von Neumann 'reviewed some work of Turing and Pitts' (at some point during the time Mooers spent in his company that day). However, this tells us little about Lecture 3—merely that von Neumann's discussing the work of Turing and Pitts was certainly not a one-off occurrence. Argument *C* is even feebler than arguments *A* and *B*.

8.4. Summing Up

Von Neumann turned to Turing's universal machine early in his deliberations on how to build an electronic computer, in the key period prior to the First Draft when his ideas were forming and developing. Lecture 3 outlines his path to what would in the First Draft become CC and CA—and it is striking that the only names von Neumann considered worth mentioning in that outline were Turing and Pitts. This lecture, happily preserved, affords a unique view of Turing's abstract logical concept of 1936 moving into the arena of hardware design.

9. Open Questions

The transfer of ideas between Turing and von Neumann, both outstanding figures in the genesis of the electronic computer, is a fundamental topic in both the history of logic and the history of computing—and there are many open questions demanding further research. For example, are there additional undiscovered wartime documents in which von Neumann discusses Turing's work? There are questions too about the similarities and differences between Turing's and von Neumann's computer designs. To what extent was Turing's design in 'Proposed Electronic Calculator' influenced by von Neumann's in the First Draft? What led Turing to reject the centralised architecture he found in the First Draft, in favour of the distributed or decentralised architecture presented in 'Proposed Electronic Calculator'?

Further questions concern their roughly contemporaneous turn to biological issues. Did they discuss these issues when they met in January 1947? Turing was at that time planning to use the ACE to study the brain, saying in a letter to Ross Ashby: 'In working on the ACE I am more interested in the possibility of producing models of the action of the brain than in the practical applications to computing' [141]. A few weeks prior to meeting Turing, on 29 November 1946, von Neumann had written to Wiener, proposing as a prelude to the study of the human nervous system, with its 'exceptional complexity', a study of 'simpler systems', such as viruses and bacteriophages [48]; in the same letter, he advocated the study of 'self-reproductive mechanisms', saying that his formulation of the problem was 'in about the style in which Turing did it for his mechanisms'. It would not be at all surprising if Turing and von Neumann discussed these matters in person. A powerful indication of the importance von Neumann accorded to Turing and his ideas is the fact that the only people mentioned by name in his subsequent book *The Computer and the Brain* were Turing and Shannon.

Our narrative has itself raised some important open questions. Did Turing and von Neumann discuss the Entscheidungsproblem in Cambridge in the spring of 1935, at that critical juncture when Turing's interest in the problem was building? Moving on to their next period of time together, what passed between them during Turing's two academic years in Princeton in 1936–38? Aspray's account of his 1979 interview with Church's student Rosser is intriguing:

> Von Neumann, who, as a member of the Institute for Advanced Study, had an office in the same building at Princeton, was attracted to Turing ... Turing's view on the computer and the brain was disputed by von Neumann, and the two discussed the issue on many occasions while Turing was completing his dissertation.
>
> [142] (pp. 147–148)

We have not been able to find any independent confirmation of such discussions, but Rosser's reflections clearly demonstrate the need for further research on this formative period of Turing's—and von Neumann's—thinking.

Intriguingly, there is a comment in von Neumann's Lecture 3 concerning the number of states N that the universal machine requires. He said, 'It can be shown that N = 700 will be sufficient', adding 'This is less than Turing estimated'. But Turing gave no estimate of the number of states in 'On Computable Numbers'. Whether he voiced an estimate to von Neumann is another open question. Did the two discuss the matter while Turing was studying at Princeton, or during the war years?

A further open question, of course, is whether the two met during the war, on one side of the Atlantic or the other. We have established that there were undoubtedly opportunities for them to meet; so far, however, we have found no firm evidence that they did so. But whether or not Turing and von Neumann met between 1938 and 1947, the important finding is that von Neumann was well aware of 'On Computable Numbers' in the war years, and was applying the concepts spelled out in Turing's paper to the issue of logical control soon after he first became involved in electronic computing.

10. Conclusions: Transmission of Turing's Logico-Philosophical Ideas to Computing—And Consequences for Philosophy

We have argued that logico-philosophical ideas introduced by Turing and published in his 1936 paper 'On Computable Numbers' had a tremendous impact beyond the field of mathematical logic. In less than a decade, concepts pioneered in that paper moved into the arena of computer design and engineering, where they played a fundamental role. This transfer of ideas was due on the one hand to Turing himself, as he designed the Automatic Computing Engine, and on the other to von Neumann, as he worked on the designs of EDVAC and the Princeton computer. Evidence furnished by von Neumann's lectures on high-speed computing answers skepticism over whether he did act as a conduit for Turing's ideas in this way. His lecture linked Turing's universal machine, with its instruction-reading-and-obeying control mechanism, to the problem of logical control in electronic computers, and this solution was developed in detail in von Neumann's First Draft. Turing and von Neumann, on their respective sides of the Atlantic, were both responsible for transferring Turing's logico-philosophical ideas of 1936 into the hands of the engineers who built the first modern computers. Nor was the impact of logic on the emerging field of electronic computing limited to the problem of logical control. We also described a second example of the transfer of concepts from the one field to the other: the use in CA, the calculating heart of the computer, of McCulloch–Pitts type propositional logic, with the reduction of binary addition to exclusive disjunction plus timing.

Almost immediately, concepts from computing started making the return journey to logic and philosophy. As Turing so correctly said in 1947: 'I expect that digital computing machines will eventually stimulate a considerable interest in symbolic logic and mathematical philosophy' [20] (p. 392). Even Turing, though, could hardly have predicted the extent of this computational turn, or foreseen that his work of 1936 would ultimately be responsible for transformations in approaches and emphases across the full sweep of philosophy, from aesthetics and the afterlife through to Zeno.

To conclude by returning to our title, and to the logic skeptics and our opening quotation from von Neumann, our narrative has demonstrated that the 'problem of developing a computing machine' was indeed 'a problem in logic' as well as a problem in engineering. Yet skepticism concerning the role of logic in the early history of electronic computing is currently fashionable (see, e.g., [39,42,122,132,143]). It was claimed in the introduction to a recent special issue of *IEEE Annals of the History of Computing* on the topic of Logic and Computation that 'the image that computing is based on logical insights . . . misrepresented the more complex historical reality', and therefore this image needs to be 'corrected' by today's historians [121] (p. 6). We have demonstrated, however, that the 'image' is far from a misrepresentation. As we have explained, 'logical insights' played a most fundamental role in von Neumann's and Turing's development of the electronic computer.

Author Contributions: Conceptualization, B.J.C. and Z.F.; Investigation, B.J.C. and Z.F.; Writing—original draft, B.J.C. and Z.F.; Writing—review and editing, B.J.C. and Z.F. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: We are grateful to Diane Proudfoot for her comments on an earlier version of this article. Zhao Fan is a JSPS International Research Fellow (ID No.: P22767), and he is grateful to receive a postdoctoral fellowship from the Japan Society for the Promotion of Science (JSPS). The lead illustration is a detail from 'Turing Diptych' by Jack Copeland, Peter Fitzpatrick, and Vicki Hyde. http://www.AlanTuring.net/turing_diptych.jpg.

Conflicts of Interest: The authors declare no conflict of interest.

Notes

1 Material originally in German has been translated by Copeland.

2 Minutes of the Cambridge Moral Sciences Club, Min. IX. 43, p. 144. Cambridge University Archives, Cambridge, UK.

3 Quoted by permission of the Master and Fellows of St John's College, Cambridge, UK.

4 Turing was in the Department of Mathematics, which at that time was co-located with the Institute for Advanced Study (IAS) where von Neumann worked.

5 Flowers noted in his diary entry for 5 February: 'Colossus did its first job. Car broke down on way home' [31] (p. 75).

6 'Research Programme for the Year 1945–46'. National Physical Laboratory, London, UK. October 1944. Items 6502, 6502.1, 6502.2. Available online: The Turing Archive for the History of Computing, http://www.AlanTuring.net/research_programme_1945-46 (accessed on 1 October 2022).

7 Woodger [37] records the existence of a National Physical Laboratory file giving the date of Turing's completed report as 1945; the file was destroyed in 1952.

8 George Davis, in comments at a meeting of the BCS Computer Conservation Society at the Science Museum, London, UK, on 28 October 2004.

9 Burks [44] gives a discussion of the impracticality of the control mechanism detailed in 'On Computable Numbers'.

10 *DEUCE News*, English Electric Company, circa 1963 (thanks to Robin Vowels for supplying a copy).

11 Von Neumann's citizenship papers. U.S. Citizenship and Immigration Services.

12 Memorandum from Navy Department Bureau of Supplies and Accounts to Claims Division, 24 July 1943. Library of Congress, Washington, DC, USA. John von Neumann and Klara Dan von Neumann Papers, Box 15.

13 At Los Alamos the perception was that von Neumann 'consulted for several government projects at such a pace that he seemed to be in many places at the same time' [62] (p. 352).

14 The ensuing race was easily won by the IBM machines, in part because Aiken's computer delivered results at much higher precision [62].

15 In Copeland and Fan [91] we discuss our discovery of, and the historical and philosophical significance of, the records of Turing's borrowings from the Cambridge Philosophical Society Library.

16 *Cambridge University Reporter*, 18 April 1935, p. 826.

17 It is not known precisely how many letters von Neumann and Turing exchanged at this time, since the rest of the correspondence seems not to have survived (in his 5 December letter, von Neumann mentioned a proof 'about which I wrote to you in my preceding letter', but the letter he is referring to is absent).

18 Loveday [104] reports diary entries by Alexander Fowler, Turing's supervisor at Bell Labs.

19 'Naval Security Station Moved to Nebraska Avenue, February 7, 1943', Station HYPO, https://stationhypo.com/2020/02/07/naval-security-station-moved-to-nebraska-avenue-february-7-1943/ (accessed on 1 October 2022) (thanks to Frode Weierud for information).

20 US Navy Department Bureau of Ordnance, communications to von Neumann concerning travel orders, 27 October 1942, 30 November 1942, 3 December 1942, 30 December 1942. Library of Congress, Washington, DC, USA. John von Neumann and Klara Dan von Neumann Papers, Box 15.

21 US Navy contract 171.60998, 9 July 1942. Library of Congress, Washington, DC, USA. John von Neumann and Klara Dan von Neumann Papers, Box 15.

22 US Navy Department Memoranda, August 1943, January 1944. Library of Congress, Washington, DC, USA. John von Neumann and Klara Dan von Neumann Papers, Box 15.

23 *Evening News*, 23 December 1946. (The cutting is among a number kept by Sara Turing.)

24 Numerico [111] gives an interesting comparative discussion of 'Proposed Electronic Calculator' and the First Draft.

25 Goldstine, present at some of the meetings, reported that topics of discussion also included mercury delay-line memory and signal-to-noise ratios [71] (p. 191).

[26] 'Showed' is perhaps more accurate than 'asserted'.

[27] Turing used various symbols to 'mark' the items in memory [3] (Sections 6 and 7), and von Neumann also noted the need for a 'distinguishing mark, which indicates whether it [a series of digits in memory] is a standard number or an order' [74] (p. 85).

[28] Burks explained that 'for various reasons [von Neumann] stopped writing this report after specifying the machine code (program language) of the EDVAC . . . and before designing a Control that could execute that program language' [44] (p. 181).

[29] Turing and Womersley presented 'Proposed Electronic Calculator' to a meeting of the Executive Committee of the National Physical Laboratory on 19 March 1946. The meeting minutes summarise Turing's presentation. Available online: The Turing Archive for the History of Computing, http://www.AlanTuring.net/npl_minutes_mar1946 (accessed on 1 October 2022).

[30] We thank an anonymous reviewer for pointing out that use of the term 'universal computing machine' in accounts of the aims and achievements of early computer projects highlights the relevance and importance of Turing's 1936 paper.

[31] Comparisons with other documents in the same folder (in particular, another sheet of notes dated December 4 and a letter from von Neumann to Goldstine dated 10 December) show that the first three sides of the LA Notes [134] date from this period in early December. While there is no reason to think that the final two sides, which contain the adder and other neuron diagrams, are any later, this has not been verified by reference to other documents. As we point out in the text, the adders certainly pre-date the 23 March 1945 meeting at the Moore School.

[32] Apart from the adders, the LA Notes contain logic diagrams for a multiplier, a gating circuit, and a circuit involving an adder and a discriminator acting together to process digits flowing from memory, as well as a diagram showing actual vacuum tubes, rather than the more abstract Pitts–McCulloch neurons. The notes also concern von Neumann's search for a viable memory technology. Options mentioned in the notes include variants of the iconoscope, and various possibilities for a 'delay' (or 'cyclical') memory, in which pulses (bits) would be stored by recirculating them through a device that delayed them for a fraction of a second—the pulses would go round and round until required. Candidates in the notes for a delay device included standard off-the-shelf items known as electrical delay lines and, more esoterically, liquid-filled devices under development at Bell Labs and at the recently founded MIT Radiation Laboratory. Von Neumann additionally mentioned research on oscilloscope-type memory at the Radiation Lab. In the First Draft, he discussed both delay memory and the iconoscope type, which he described as 'prima facie . . . more natural' than the delay type, its advantage being the feature now termed 'random access' (as in RAM), the ability to access digits without the variable delays inherent in delay memory (dependent on the digit's position in the cyclical memory at the time it is needed) [74] (sect. 12.6). However, the iconoscope never worked successfully as a computer memory, and von Neumann's description in the First Draft of the iconoscope idea was out of date almost as soon as he wrote it. Eckert perfected a form of the liquid-filled delay device, known as a mercury delay line, and used it in his and Mauchly's UNIVAC, as did Turing in the ACE. Von Neumann retained his preference for cathode-ray tube (CRT) memory, and his Princeton computer employed 'Williams tubes', a CRT random-access memory perfected by Williams at Manchester University, and inspired by the Radiation Lab work on oscilloscope memory mentioned in the LA Notes [73,135].

[33] What Goldstine was trying to draw in Pitts' notation appears to have been the tube circuit that Eckert and Mauchly subsequently included in a September 1945 report concerning EDVAC [137] (p. 92 & Figure PY-2-101), and later made the subject of a patent filed in October 1950 [138]; although by the time of the patent, the 1945 circuit's tetrodes, which had never operated satisfactorily, had been replaced by pentagrid tubes.

[34] Von Neumann's adder simply generates and then delays the carry digit, so that it is fed into the second half-adder at the right time; whereas Goldstine's adder, having generated and delayed a carry digit, requires an additional feedback loop and then a further additional delay in order to handle carries at the second half-adder.

References

1. von Neumann, J. High Speed Computing. Undated, 194?. American Philosophical Society Library, Philadelphia, PA, USA. Herman Heine Goldstine Papers, Box 49.
2. Davis, M. *The Universal Computer: The Road from Leibniz to Turing*; W.W. Norton & Company: New York, NY, USA, 2000.
3. Turing, A.M. On Computable Numbers, with an Application to the Entscheidungsproblem. *Proc. Lond. Math. Soc.* **1936**, *42*, 230–265.
4. Hilbert, D.; Ackermann, W. *Grundzüge der Theoretischen Logik*; Springer: Berlin, Germany, 1928.
5. Behmann, H. Entscheidungsproblem und Algebra der Logik. 1921. In Mancosu, P.; Zach, R. Heinrich Behmann's 1921 Lecture on the Decision Problem and the Algebra of Logic. *Bull. Symb. Log.* **2015**, *21*, 177–187.
6. Schönfinkel, M. Zum Entscheidungsproblem der mathematischen Logik. Heft I. Undated, 192?. ETH Zurich, Zurich, Switzerland. Bernays Papers, Hs 974.282.
7. Mancosu, P.; Zach, R. Heinrich Behmann's 1921 Lecture on the Decision Problem and the Algebra of Logic. *Bull. Symb. Log.* **2015**, *21*, 164–187. [CrossRef]
8. Ramsey, F.P. On a Problem of Formal Logic. *Proc. Lond. Math. Soc.* **1930**, *30*, 264–286. [CrossRef]
9. Peirce, C. Drafts of the Second and Third Lowell Lectures (R S32, R 462, and R 464). 1903. In *Charles S. Peirce: Logic of the Future (Volume 2, Part 2), Writings on Existential Graphs: The 1903 Lowell Lectures*; Pietarinen, A.-V., Ed.; de Gruyter: Berlin, Germany, 2021.

10. Wittgenstein, L. Letter to Bertrand Russell. November 1913. In *Wittgenstein in Cambridge: Letters and Documents 1911–1951*; McGuinness, B., Ed.; Blackwell: Oxford, UK, 2008; pp. 53–54.
11. Lampert, T. Wittgenstein's Conjecture. In *Philosophy of Logic and Mathematics: Proceedings of the 41st International Wittgenstein Symposium*; Mras, G., Ritter, B., Weingartner, P., Eds.; DeGruyter: Berlin, Germany, 2019; pp. 515–534.
12. Gödel, K. Zum Entscheidungsproblem des logischen Funktionenkalküls. *Monatsh. Math.* **1933**, *40*, 433–443. [CrossRef]
13. Langford, C.H. Some Theorems on Deducibility. *Ann. Math.* **1926**, *28*, 16–40. [CrossRef]
14. Langford, C.H. Theorems on Deducibility (Second Paper). *Ann. Math.* **1927**, *28*, 459–471. [CrossRef]
15. Quine, W. Review of Grundzüge der theoretischen Logik by D. Hilbert, W. Ackermann. *J. Symb. Log.* **1938**, *3*, 83–84. [CrossRef]
16. Hilbert, D. Axiomatisches Denken. *Math. Ann.* **1917**, *78*, 405–415. [CrossRef]
17. Hilbert, D. Neubegründung der Mathematik: Erste Mitteilung. *Abh. Math. Semin. Univ. Hambg.* **1922**, *1*, 157–177. [CrossRef]
18. Hilbert, D. Über das Unendliche. *Math. Ann.* **1926**, *95*, 161–190. [CrossRef]
19. Carnap, R. Ein Gültigkeitskriterium für die Sätze der klassischen Mathematik. *Monatsh. Math.* **1935**, *42*, 163–190. [CrossRef]
20. Turing, A.M. Lecture on the Automatic Computing Engine. 1947. In *The Essential Turing*; Copeland, B.J., Ed.; Oxford University Press: Oxford, UK, 2004; pp. 378–394.
21. Newman, M.H.A. Interviewed by Christopher Evans. Undated, circa 1977. Science Museum, London, UK. Transcription by Copeland.
22. Newman, M.H.A. The Foundations of Mathematics from the Standpoint of Physics. 1923. Fellowship Dissertation. St John's College, Cambridge, UK. Records of St John's College, SJCR/SJAC/2/1/5/1.
23. Kripke, S. The Church-Turing 'Thesis' as a Special Corollary of Gödel's Completeness Theorem. In *Computability: Turing, Gödel, Church, and Beyond*; Copeland, B.J., Posy, C., Shagrir, O., Eds.; MIT Press: Cambridge, MA, USA, 2013; pp. 77–104.
24. Gandy, R. The Confluence of Ideas in 1936. In *The Universal Turing Machine: A Half-Century Survey*; Herken, R., Ed.; Kammerer & Unverzagt: Berlin, Germany, 1988; pp. 55–111.
25. Church, A. A Note on the Entscheidungsproblem. *J. Symb. Log.* **1936**, *1*, 40–41. [CrossRef]
26. Gandy, R. Letter to Max Newman. Undated, 195?. King's College, Cambridge, UK. King's College Archives, AMT/A/8.
27. Turing, A.M. Systems of Logic Based on Ordinals. *Proc. Lond. Math. Soc.* **1939**, *45*, 161–228. [CrossRef]
28. Copeland, B.J. (Ed.) *The Essential Turing*; Oxford University Press: Oxford, UK, 2004.
29. Newman, M.; Turing, A.M. A Formal Theorem in Church's Theory of Types. *J. Symb. Log.* **1942**, *7*, 28–33. [CrossRef]
30. Hinsley, H. *British Intelligence in the Second World War*; Her Majesty's Stationery Office: London, UK, 1981; Volume 2.
31. Copeland, B.J. *Colossus: The Secrets of Bletchley Park's Codebreaking Computers*; Oxford University Press: Oxford, UK, 2010.
32. Flowers, T. Interviewed by Jack Copeland. July 1996.
33. Good, I.J. Interviewed by Jack Copeland. February 2004.
34. Copeland, B.J. *Alan Turing's Automatic Computing Engine*; Oxford University Press: Oxford, UK, 2005.
35. Womersley, J.R. A.C.E. Project—Origin and Early History. 26 November 1946. National Physical Laboratory, London, UK. Available online: http://www.alanturing.net/ace_early_history (accessed on 1 October 2022).
36. Turing, A.M. Proposed Electronic Calculator. 1945. In *Alan Turing's Automatic Computing Engine*; Copeland, B.J. Oxford University Press: Oxford, UK, 2005; pp. 369–457.
37. Woodger, M. Untitled handwritten note. Undated. Science Museum, London, UK. Woodger Papers, M15/78.
38. Woodger, M. Letter to Copeland. 27 November 1999.
39. Haigh, T.; Priestley, M. Von Neumann Thought Turing's Universal Machine was 'Simple and Neat': But That Didn't Tell Him How to Design a Computer. *Commun. ACM* **2020**, *63*, 26–32. [CrossRef]
40. Darwin, C.G. Automatic Computing Engine (ACE). 17 April 1946. In *Alan Turing's Automatic Computing Engine*; Copeland, B.J. Oxford University Press: Oxford, UK, 2005; pp. 53–57.
41. Michie, D. Letter to Jack Copeland. 14 July 1995.
42. Corry, L. Turing's Pre-War Analog Computers: The Fatherhood of the Modern Computer Revisited. *Commun. ACM* **2017**, *60*, 50–58. [CrossRef]
43. Turing, A.M. Notes on Memory. 1945. In *Alan Turing's Automatic Computing Engine*; Copeland, B.J. Oxford University Press: Oxford, UK, 2005; pp. 455–457.
44. Burks, A. Turing's Theory of Infinite Computing Machines (1936–1937) and Its Relation to the Invention of Finite Electronic Computers (1939–1949). In *Theory and Practical Issues on Cellular Automata*; Bandini, S., Worsch, T., Eds.; Springer: London, UK, 2001; pp. 179–197.
45. Mays, W. Rough Draft of the Discussion on the Mind and the Computing Machine, Held on Thursday, 27 October 1949, in the Philosophy Seminar. Manchester University, UK. Available online: http://www.rutherfordjournal.org/article010111.html (accessed on 1 October 2022).
46. Turing, A.M. Intelligent Machinery. 1948. In *The Essential Turing*; Copeland, B.J., Ed.; Oxford University Press: Oxford, UK, 2004; pp. 410–432.
47. Turing, A.M. Computing Machinery and Intelligence. *Mind* **1950**, *59*, 433–460. [CrossRef]
48. von Neumann, J. Letter to Norbert Wiener, 29 November 1946. Library of Congress, Washington, DC, USA. John von Neumann and Klara Dan von Neumann Papers, Box 7.
49. von Neumann, J. Zur Hilbertschen Beweistheorie. *Math. Z.* **1927**, *26*, 1–46. [CrossRef]
50. Hardy, G. Mathematical Proof. *Mind* **1929**, *38*, 1–25. [CrossRef]

51. von Neumann, J. Die formalistische Grundlegung der Mathematik. *Erkenntnis* **1931**, *2*, 116–121. [CrossRef]
52. Aspray, W. *John von Neumann and the Origins of Modern Computing*; MIT Press: Cambridge, MA, USA, 1990.
53. von Neumann, J. Letter to Frank Aydelotte. 11 October 1945. Institute for Advanced Study, Princeton, NJ, USA. Shelby White and Leon Levy Archives Center, Box 34.
54. von Neumann, J. Letter to Klara Dan von Neumann. 17 February 1943. Library of Congress, Washington, DC, USA. John von Neumann and Klara Dan von Neumann Papers, Box 1.
55. von Neumann, J. Letter to Klara Dan von Neumann. 13 March 1943. Library of Congress, Washington, DC, USA. John von Neumann and Klara Dan von Neumann Papers, Box 1.
56. Todd, J. John von Neumann and the National Accounting Machine. *SIAM Rev.* **1974**, *16*, 526–530. [CrossRef]
57. Tee, G.J. New Zealand and the Pioneers of Early Computing. *N.Z. J. Comput.* **1989**, *1*, 5–10.
58. Croarken, M. *Early Scientific Computing in Britain*; Clarendon Press: Oxford, UK, 1990.
59. von Neumann, J. Letter to John Todd. 17 November 1947. In Todd, J. John von Neumann and the National Accounting Machine. *SIAM Rev.* **1974**, *16*, 526.
60. von Neumann, J. Letter to Oswald Veblen. 21 May 1943. Library of Congress, Washington, DC, USA. Oswald Veblen Papers, Box 15.
61. von Neumann, J. Letter to Warren Weaver. 23 August 1943. Library of Congress, Washington, DC, USA. John von Neumann and Klara Dan von Neumann Papers, Box 7.
62. Metropolis, N.; Nelson, E.C. Early Computing at Los Alamos. *Ann. Hist. Comput.* **1982**, *4*, 348–357. [CrossRef]
63. Weaver, W. Letter to John von Neumann. 22 March 1944. Library of Congress, Washington, DC, USA. John von Neumann and Klara Dan von Neumann Papers, Box 7.
64. von Neumann, J. Letter to E. L. Chaffee. 29 March 1944. In *Howard Aiken: Portrait of a Computer Pioneer*; Cohen, I.B., Ed.; MIT Press: Cambridge, MA, USA, 1999.
65. Weaver, W. Letter to John von Neumann. 28 March 1944. Library of Congress, Washington, DC, USA. John von Neumann and Klara Dan von Neumann Papers, Box 7.
66. Weaver, W. Letter to Emory Chaffee. 22 March 1944. Library of Congress, Washington, DC, USA. John von Neumann and Klara Dan von Neumann Papers, Box 7.
67. von Neumann, J. Letter to Warren Weaver. 1 November 1944. Library of Congress, Washington, DC, USA. John von Neumann and Klara Dan von Neumann Papers, Box 7.
68. von Neumann, J. Letter to Warren Weaver. 27 March 1944. Library of Congress, Washington, DC, USA. John von Neumann and Klara Dan von Neumann Papers, Box 7.
69. von Neumann, J. Letter to Warren Weaver. 10 April 1944. Library of Congress, Washington, DC, USA. John von Neumann and Klara Dan von Neumann Papers, Box 7.
70. von Neumann, J. Letter to Robert Oppenheimer. 1 August 1944. Los Alamos National Laboratory, Los Alamos, NM, USA. Los Alamos National Laboratory Archive.
71. Goldstine, H. *The Computer from Pascal to von Neumann*; Princeton University Press: Princeton, NJ, USA, 1972.
72. Brainerd, J. Letter to Paul Gillon. 13 September 1944. In *The Computer from Pascal to von Neumann*; Goldstine, H. Princeton University Press: Princeton, NJ, USA, 1972; pp. 186–187.
73. Copeland, B.J.; Haeff, A.; Gough, P.; Wright, C. Screen History: The Haeff Memory and Graphics Tube. *IEEE Ann. Hist. Comput.* **2017**, *39*, 9–28.
74. von Neumann, J. First Draft of a Report on the EDVAC. 30 June 1945. Moore School of Electrical Engineering, University of Pennsylvania, Philadelphia, PA, USA. Available online: https://library.si.edu/digital-library/book/firstdraftofrepo00vonn (accessed on 1 October 2022).
75. Warren, S.R. Interviewed by Nancy Stern. 5 October 1977. Charles Babbage Institute, Minneapolis, MN, USA. Oral History Interviews, OH 38.
76. Burks, A. Interviewed by Christopher Evans. Undated, circa 1977. Science Museum, London, UK. Transcription by Copeland.
77. von Neumann, J. Letter to Herman Goldstine. 12 February 194[5]. American Philosophical Society Library, Philadelphia, PA, USA. Herman Heine Goldstine Papers, Box 9.
78. Eckert, J.P.; Mauchly, J.; Warren, S.R. PY Summary Report No. 1. 31 March 1945. Dartmouth College Library, Hanover, NH, USA. George Stibitz Papers, Box 30.
79. von Neumann, J. Letter to Herman Goldstine. 8 May 1945. American Philosophical Society Library, Philadelphia, PA, USA. Herman Heine Goldstine Papers, Box 9.
80. Goldstine, H. Letter to John von Neumann. 15 May 1945. American Philosophical Society Library, Philadelphia, PA, USA. Herman Heine Goldstine Papers, Box 21.
81. Aydelotte, F. Letter to John von Neumann. 22 August 1945. Library of Congress, Washington, DC, USA. John von Neumann and Klara Dan von Neumann Papers, Box 2.
82. von Neumann, J. Letter to Frank Aydelotte. 25 August 1945. Library of Congress, Washington, DC, USA. John von Neumann and Klara Dan von Neumann Papers, Box 2.
83. von Neumann, J. Memorandum on the Program of the High-Speed Computer Project. 8 November 1945. American Philosophical Society Library, Philadelphia, PA, USA. Herman Heine Goldstine Papers, Box 9.

84. Bigelow, J. Computer Development at the Institute for Advanced Study. In *A History of Computing in the Twentieth Century*; Metropolis, N., Howlett, J., Rota, G., Eds.; Academic Press: New York, NY, USA, 1980; pp. 291–310.
85. Hurd, C. Computer Development at IBM. In *A History of Computing in the Twentieth Century*; Metropolis, N., Howlett, J., Rota, G., Eds.; Academic Press: New York, NY, USA, 1980; pp. 389–418.
86. Burks, A.; Goldstine, H.; von Neumann, J. Preliminary Discussion of the Logical Design of an Electronic Computing Instrument. 28 June 1946. In *John von Neumann: Collected Works*; Taub, A.H., Ed.; Pergamon Press: London, UK, 1961; Volume 5, pp. 34–79.
87. Burks, A. The Invention of the Universal Electronic Computer—How the Electronic Computer Revolution Began. *Future Gener. Comput. Syst.* **2002**, *18*, 871–892. [CrossRef]
88. Halsbury, A. Ten Years of Computer Development. *Comput. J.* **1959**, *1*, 153–159. [CrossRef]
89. Turing, S. *Alan M. Turing*; Heffer: Cambridge, UK, 1959.
90. Turing, A.M. On the Gaussian Error Function. 1935. King's College, Cambridge, UK. King's College Archives, AMT/C/28.
91. Copeland, B.J.; Fan, Z. Did Turing Stand on Gödel's Shoulders? *Math. Intell.* **2022**, *44*, 308–319. [CrossRef]
92. von Neumann, J. Almost Periodic Functions in a Group. I. *Trans. Am. Math. Soc.* **1934**, *36*, 445–492. [CrossRef]
93. Turing, A.M. Equivalence of Left and Right Almost Periodicity. *J. Lond. Math. Soc.* **1935**, *10*, 284–285. [CrossRef]
94. von Neumann, J. Letter to Marshall Stone. 31 March 1935. Library of Congress, Washington, DC, USA. John von Neumann and Klara Dan von Neumann Papers, Box 6.
95. Smithies, F. Notes on John von Neumann's 1935 Cambridge Lectures. 1935. Part 1. St John's College College, Cambridge, UK. St John's College Personal Collections, GB 275 Smithies/H/H67.
96. Smithies, F. Notes on John von Neumann's 1935 Cambridge Lectures. 1935. Part 2. St John's College College, Cambridge, UK. St John's College Personal Collections, GB 275 Smithies/H/H68.
97. von Neumann, J. Letter to Oswald Veblen. 6 July 1935. Library of Congress, Washington, DC, USA. Oswald Veblen Papers, Box 15.
98. von Neumann, J. Letter to Alan Turing. 5 December [1935]. King's College, Cambridge, UK. King's College Archives, AMT/D/5.
99. Turing, A.M. Letter to Sara Turing. 3 November 1936. King's College, Cambridge, UK. King's College Archives, AMT/K/1/44.
100. Turing, A.M. Finite Approximations to Lie Groups. *Ann. Math.* **1938**, *39*, 105–111. [CrossRef]
101. von Neumann, J. Letter to H.R. Dean. 1 June 1937. Princeton University, Princeton, NJ, USA. Available online: https://www.turing.org.uk/sources/vonneumann.html (accessed on 1 October 2022).
102. Turing, A.M. Letter to Sara Turing. 17 May 1938. King's College Archives, Cambridge, AMT/K/1/68.
103. Turing, A.M. Report on Cryptographic Machinery Available at Navy Department Washington. 1942. G.C. & C.S., National Archives, Kew, HW57/10.
104. Loveday, E.V. Letter to Sara Turing. 20 June 1960. King's College, Cambridge, UK. King's College Archives, AMT/A/20/01.
105. Rowland, B.; Boyd, W.B. *U.S. Navy Bureau of Ordnance in World War II*; Bureau of Ordnance: Washington, DC, USA, 1953.
106. von Neumann, J. Letter to Oswald Veblen. 30 March 1943. Library of Congress, Washington, DC, USA. Oswald Veblen Papers, Box 15.
107. von Neumann, J. Letter to Klara Dan von Neumann. 28 February 1943. Library of Congress, Washington, DC, USA. John von Neumann and Klara Dan von Neumann Papers, Box 1.
108. von Neumann, J. Letter to Klara Dan von Neumann. 2 June 1943. Library of Congress, Washington, DC, USA. John von Neumann and Klara Dan von Neumann Papers, Box 1.
109. Warren, S.R. Notes on the Preparation of 'First Draft of a Report on the EDVAC' by John von Neumann. 2 April 1947. Hagley Museum & Library, Wilmington, DE, USA. Sperry Rand Corporation, Univac Division Records (1825-I), Box 24.
110. Gandy, R. Interviewed by Jack Copeland. October 1995.
111. Numerico, T. From Turing Machine to 'Electronic Brain'. 2005. In *Alan Turing's Automatic Computing Engine*; Copeland, B.J. Oxford University Press: Oxford, UK, 2005; pp. 173–192.
112. Huskey, H. Letter to Jack Copeland. 4 February 2002.
113. Newman, M.H.A. Letter to John von Neumann. 8 February 1946. Manchester University, Manchester, UK. Available online: http://www.alanturing.net/newman_vonneumann_8feb46 (accessed on 1 October 2022).
114. Turing, A.M. Rounding-off Errors in Matrix Processes. *Q. J. Mech. Appl. Math.* **1948**, *1*, 287–308. [CrossRef]
115. Turing, A. Report on Visit to U.S.A. 1–20 January 1947. Bletchley Park, Bletchley, UK. Available online: http://www.alanturing.net/turing_usa_visit (accessed on 1 October 2022).
116. von Neumann, J. Letter to Alan Turing. 13 September 1949. King's College, Cambridge, UK. King's College Archives, AMT/D/5.
117. Bigelow, J. Interviewed by Richard Mertz. 20 January 1971. National Museum of American History, Washington, DC, USA. Archives Center, Computer Oral History Collection, 1969–1973, 1977.
118. Bigelow, J. Letter to Jack Copeland. 12 April 2002.
119. Frankel, S. Letter to Brian Randell (no date supplied). In *On Alan Turing and the Origins of Digital Computers*; Randell, B. Computing Laboratory Technical Report Series, No. 33; University of Newcastle upon Tyne: Newcastle, UK, 1972; p. 15.
120. Denning, P.; Tedre, M. Mathematics, Logic, and Engineering in Computing. *IEEE Ann. Hist. Comput.* **2021**, *43*, 97–100. [CrossRef]
121. De Mol, L. Logic, Programming, and Computer Science: Local Perspectives. *IEEE Ann. Hist. Comput.* **2021**, *43*, 5–9. [CrossRef]
122. Daylight, E. A Turing Tale. *Commun. ACM* **2014**, *57*, 36–38. [CrossRef]
123. von Neumann, J. Memorandum to Re2. 27 August 1945. American Philosophical Society Library, Philadelphia, PA, USA. Herman Heine Goldstine Papers, Box 9.

124. von Neumann, J. Memorandum to Frank Aydelotte. 5 September 1945. Library of Congress, Washington, DC, USA. John von Neumann and Klara Dan von Neumann Papers, Box 12.
125. McCulloch, W.; Pitts, W. A Logical Calculus of the Ideas Immanent in Nervous Activity. *Bull. Math. Biophys.* **1943**, *5*, 115–133. [CrossRef]
126. McCulloch, W. Discussion. 1948. In *John von Neumann: Collected Works*; Taub, A.H., Ed.; Pergamon Press: London, UK, 1961; Volume 5, pp. 319–328.
127. McCarthy, J. Information. *Sci. Am.* **1966**, *215*, 64–73. [CrossRef]
128. Daylight, E. Towards a Historical Notion of 'Turing—The Father of Computer Science'. *Hist. Philos. Log.* **2015**, *36*, 205–228. [CrossRef]
129. Mounier-Kuhn, P. Logic, Formal Linguistics and Computing in France: From Non-Reception to Progressive Convergence. In *History and Philosophy of Computing*; Gadducci, F., Tavosanis, M., Eds.; Springer: Cham, Switzerland, 2016; pp. 24–41.
130. De Mol, L.; Bullynck, M.; Daylight, E. Less Is More in the Fifties: Encounters Between Logical Minimalism and Computer Design During the 1950s. *IEEE Ann. Hist. Comput.* **2018**, *40*, 19–45. [CrossRef]
131. Haigh, T.; Priestley, M.; Rope, C. *ENIAC in Action*; MIT Press: Cambridge, MA, USA, 2016.
132. Haigh, T. Actually, Turing Did Not Invent the Computer. *Commun. ACM* **2014**, *57*, 36–41. [CrossRef]
133. Wilkes, M. Interviewed by David Anderson. *Commun. ACM* **2009**, *52*, 1–5.
134. von Neumann, J. Untitled Notes. 1944?. American Philosophical Society Library, Philadelphia, PA, USA. Herman Heine Goldstine Papers, Box 9.
135. Copeland, B.J. The Manchester Computer: A Revised History, Part I: The Memory, Part II: The Baby Computer. *IEEE Ann. Hist. Comput.* **2011**, *33*, 4–37. [CrossRef]
136. Burks, A. Notes on Meeting with Dr. von Neumann. 23 March 1945. Hagley Museum & Library, Wilmington, DE, USA. Sperry Rand Corporation, Univac Division Records (1825-I), Box 24.
137. Eckert, J.P.; Mauchly, J. Automatic High-Speed Computing: A Progress Report on the EDVAC. 30 September 1945. University of Pennsylvania, Philadelphia, PA, USA. Kislak Center for Special Collections, Rare Books and Manuscripts, John W. Mauchly Papers, Box 7.
138. Eckert, J.P.; Mauchly, J. Signal Responsive Apparatus. U.S. Patent 2,600,744, filed October 1950, issued June 1952.
139. Mooers, C. Record Book, Naval Ordnance Laboratory (Book 3). 1945-6. Charles Babbage Institute, Minneapolis, MN, USA. Calvin N. Mooers Papers, Box 28.
140. Mooers, C. The Computer Project at the Naval Ordnance Laboratory. *IEEE Ann. Hist. Comput.* **2001**, *23*, 51–67.
141. Turing, A.M. Letter to Ross Ashby. Undated, circa 1946. National Physical Laboratory, London, UK. Available online: http://www.alanturing.net/turing_ashby (accessed on 1 October 2022).
142. Aspray, W. From Mathematical Constructivity to Computer Science: Alan Turing, John von Neumann, and the Origins of Computer Science in Mathematical Logic. Ph.D. Thesis, University of Wisconsin-Madison, Madison, WI, USA, 1980.
143. Haigh, T.; Priestley, M. Where Codes Come From: Architectures of Automatic Control from Babbage to Algol. *Commun. ACM* **2016**, *59*, 39–44. [CrossRef]