

Programm und Diagramm

Überlegungen zum digitalen Bild und zur Automatisierung
anhand der Computergrafik der 1960er Jahre von Frieder Nake

Michael Rottmann

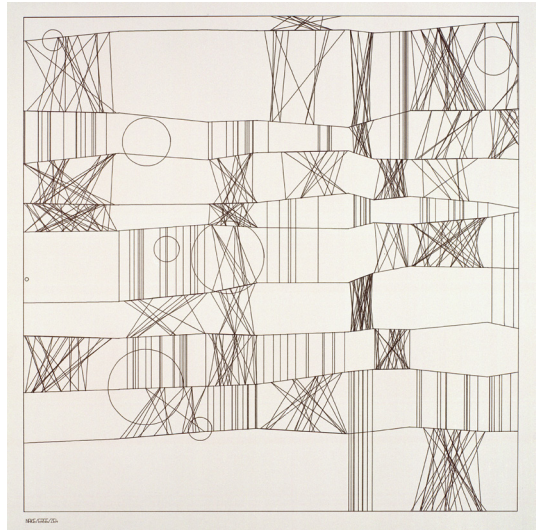
*Es gibt keine Daten ohne Datenträger.
Es gibt keine Bilder ohne Bildschirme.¹*
Claus Pias (2003)

*Ein technisches Bild entziffern heißt nicht, das von ihnen Gezeigte
entziffern, sondern ihr Programm aus ihnen herauszulesen.²*
Vilém Flusser (1985)

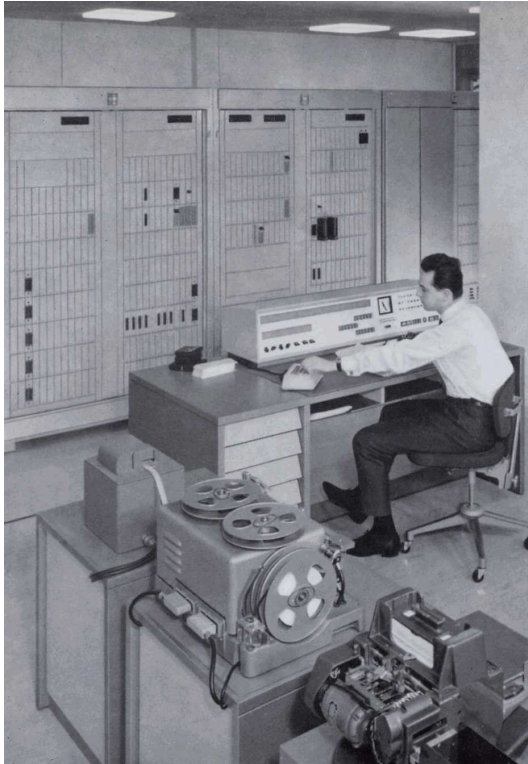
1. Was war ein digitales Bild?

Fragt man nach dem digitalen Bild, so ist wie für Bildmedien im Allgemeinen, stimuliert durch eine poststrukturalistisch und medienarchäologisch informierte Medientheorie und die Integration der Akteur-Netzwerk-Theorien in die Kulturwissenschaften, zu antworten: Bilder sind über ihre materiellen, technologischen, instrumentellen sowie sozialen, institutionellen und ökonomischen Bedingungen zu bestimmen.³ Vereinfacht gesagt, werden Bilder zu einer gewissen Zeit, unter gewissen Umständen, mit gewissen Mitteln, Techniken, Praktiken und Absichten, in gewissen sozialen Milieus gemacht, geteilt und gesehen beziehungsweise gebraucht.

Gebraucht wurde die am 13. September 1965 am Recheninstitut der Technischen Universität Stuttgart entstandene Computergrafik 13/9/65 Nr. 2, „Hommage à Paul Klee“ (1965) im engeren Sinne nicht, jedenfalls was die beruflichen Zielsetzungen ihres menschlichen Erzeugers, dem damals 26-jährigen Mathematiker Frieder Nake, angeht (Abb. 1). Sie repräsentierte keine technisch-wissenschaftlichen Sachverhalte, mit ihr waren nicht etwa Differentialgleichungen grafisch zu lösen, ohne dafür händisch Zahlenwerte auf Millimeterpapier zu zeichnen.⁴ Es sei dahingestellt, ob die Grafik zum Zeitpunkt ihrer Entstehung als ein Kunstwerk gelten kann, als ein solches war sie gedacht und wird sie heute wahrgenommen.⁵ In jedem Fall erweist sie sich als ein interessanter Gegenstand und fruchtbarer Ausgangspunkt für Überlegungen zum digitalen Bild,



1 Frieder Nake, 13/9/65 Nr. 2, „Hommage à Paul Klee“ (1965), Computergrafik, Zuse Z 64 Graphomat, schwarze Tinte auf Papier, 40 x 40 cm [1966 auch als Klee, mittlerweile kurz als *Hommage à Paul Klee* bezeichnet]



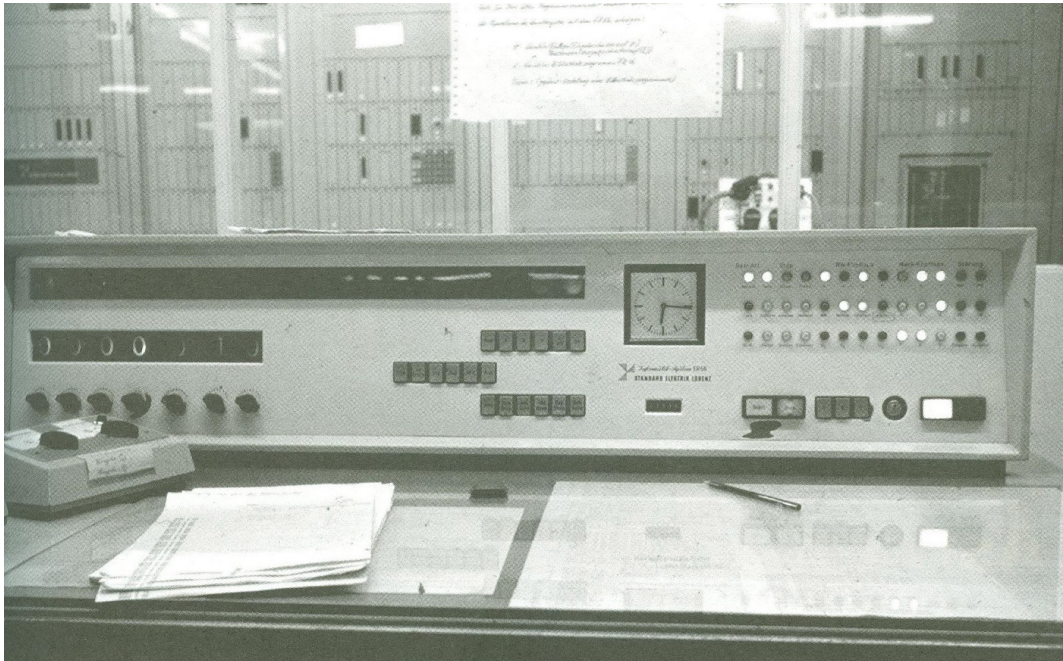
2 Anlage des SEL ER 56 (undatiert, um 1961)

weil letzteres historisch zu denken ist und die durch Paul Klees Werk *Hauptweg und Nebenwege* (1929) inspirierte Grafik durch Anwendung eines digitalen Großrechners entstanden ist – zu einer Zeit, in der die heutige digitale Kultur maßgeblich einsetzte.

Ein solches Verständnis von einem digitalen Bild, das im Zusammenhang mit einem Digitalcomputer steht, hatte sich maßgeblich seit den 1940er Jahren herausbilden können. Der traditionsreiche Begriff ‚digital‘ wurde vermutlich 1938 im Umfeld der Elektrotechnik um seine mathematisch-medientechnische Bedeutung erweitert – im Sinne von diskret, in endlichen, getrennten Einheiten, symbolisch-numerisch, ohne Umwandlung in physikalische Größen operierend – und wurde sodann zur Abgrenzung vom Analogrechner und dessen Funktionsweise auf den Digitalcomputer bezogen, womit die medientechnische Leitdifferenz ‚analog/digital‘ geboren war.⁶ Letztere machte eine beachtliche Diskurs-Karriere, die im Speziellen durch ihre Erörterung in Nelson Goodmans Symboltheorie *Languages of Art* (1968) angekurbelt wurde, und erfuhr mittlerweile,

wie die Begriffe ‚analog‘ und ‚digital‘, im Zuge des Post-digital-Diskurses ihre Dekonstruktion.⁷ Zur ‚Bild-Maschine‘ geworden war der Digitalcomputer um 1950, als erste Exemplare wie *Whirlwind* mit Bildschirmen (etwa Kathodenstrahlröhren), verschaltet wurden.⁸ Spätestens mit Ivan Sutherlands grafikfähigem Interface-System *Sketchpad* (1963) und Michael Nolls Aufsatz *The Computer as a Creative Medium* (1967) wurde der Digitalcomputer als ein Bildmedium aufgefasst und in seiner Anwendung als ein solches, auch interaktiv operabel.⁹ Wurden 1966 in einer Werbung der General Motors Research Laboratories „picture“ und „digital computer“ im Zusammenhang genannt, so war spätestens 1974 im deutschsprachigen Diskurs vom „digitalen Bild“ die Rede.¹⁰

Gut nachvollziehbar sucht und nennt die Mediengeschichte solch historische Primate. Technikgeschichtlich wäre die Bildschirmarbeit für Frieder Nake möglich gewesen. Computergrafiken wie *Hommage à Paul Klee* entstanden jedoch weder am Bildschirm, noch wurden sie auf einem solchen dargestellt.¹¹ Dafür ist weniger entscheidend, dass Nake 1963, als er mit seinen Computergrafiken begann, nichts von dem im gleichen Jahr in den USA entwickelten ‚Sketchpad‘-System wusste,¹² sondern vielmehr, dass in den 1960er Jahren Großrechner-Anlagen verbreitet bildschirmlos waren: Demgemäß verfügt der seit 1960 an der Technischen Universität Stuttgart betriebene „elektronische Rechenautomat“¹³ SEL ER 56, der seit 1956 von der Firma Standard Elektrik Lorenz AG in Stuttgart-Zuffenhausen gebaut wurde, lediglich über ein numerisches Kontrolldisplay (Abb. 2 und 3).¹⁴ Die Grafik wurde also programmiert, prozessiert und geplottet, ohne dass Nake in diesem ‚unsichtbaren Prozess‘ wissen konnte, wie sie aussehen würde, weshalb sie 1966 von der Kritik als „reine Maschinenkunst“¹⁵



3 Konsole des Großrechners SEL ER 56 an der Technischen Universität Stuttgart vor der Trennscheibe des gekühlten Rechnerraums (um 1965)

bezeichnet wurde. All dies macht Nakes Computergrafik interessant für Überlegungen zum digitalen Bild, insbesondere zu dessen Medialität und Entstehungsprozess.

2. Was ist ein digitales Bild in Kunst- und Medienwissenschaft (gewesen)?

Zunächst stellt sich die Frage: Handelt es sich bei einer gedruckten Computergrafik überhaupt um ein digitales Bild? Nake selbst sprach 1968 von der „Analog-Ausgabe bei Digitalrechnern.“¹⁶ Es ist aufschlussreich, diesbezüglich der Theoretisierung des digitalen Bildes – wegen seiner Vielfältigkeit müsste es eigentlich im Plural adressiert werden – in der Kunst- und Medienwissenschaft nachzugehen. Verstärkt erforscht wurde das digitale Bild seit den 1990er Jahren als seine Präsenz und Relevanz in Ökonomie, Wissenschaft, Alltag und nicht zuletzt in der Kunst, auch bedingt durch neue Technologien der Grafik, Virtual Reality und Vernetzung (Word Wide Web), erheblich zunahm.¹⁷ Die Kunsthistoriker Gottfried Boehm und William J. T. Mitchell, die zu dieser Zeit mit unterschiedlichen Perspektiven ihre Wendungen zum Bild diagnostizierten, behandelten das digitale Bild im Hinblick auf mediale Eigenschaften, kulturelle Implikationen und sein Verhältnis zu Neuen Medien.¹⁸ Wir können wesentliche Charakteristika erfahren: Ein digitales Bild, auch wenn es mit einer Digitalkamera erzeugt wurde, sei an den Computer und dessen Logik gebunden.¹⁹ Eine Grundlage sei der „numerische Code“ und eine andere, was einen Unterschied zu älteren mathematik-basierten Bild-Verfahren wie der (Zentral-)Perspektive ausmache, maschinelle Rechenprozesse, für die gelte: „Sie folgen einer Kette binärer Ja/Nein-Entscheidungen, einem mathematischen Programm, dessen komplexe Anleitung imstande ist, die Generierung, Speicherung und Kombination der Daten zu steuern.“²⁰ Boehm erklärte weiterhin: „Was immer wir als Bild wahrnehmen, ohne den Code der Software würde es nicht entstehen. Nur was zuvor seiner Möglichkeit nach programmiert wurde, läßt sich hernach auch als ‚Bild‘ re-

alisieren.“²¹ Darin und in der Möglichkeit der Umrechnung begründet liege seine Wandelbarkeit sowie die Eigenschaft der Imitation (anderer Bildmedien).²² Ein entscheidender Aspekt ist: Ein digitales Bild existiere als „visuelle ‚Oberfläche‘“, so Boehm, „in Pixeln und Zeilenfolgen des Bildschirms“ und in „unterlegten Codes“.²³ Diese „Doppelexistenz“ beziehungsweise „untrennbare[n] Zweiheit als Bildschirmscheinung und digitaler Datensatz“²⁴, so Gernot Grube, die oftmals dem Register von Sicht- und Unsichtbarkeit zugeordnet wird, stellt einen Topos in der Forschungsliteratur dar und findet sich mit unterschiedlichen Begrifflichkeiten und Stoßrichtungen auch bei Lev Manovich und nicht zuletzt bei Frieder Nake, der von „sichtbarer Oberfläche“²⁵ und einer bearbeitbaren „unsichtbaren Unterfläche“ spricht. Es existiert jedoch keine einheitliche Auffassung darüber, ob als digitales Bild allein das visuelle Bild oder dieses nur gemeinsam mit seinem nicht-bildhaften Code aufzufassen ist.²⁶ Claus Pias postulierte mit seiner Zuspitzung „[d]as digitale Bild gibt es nicht“, dass es nicht in einem substantiellen, sondern, auf der Basis von Code, in einem informatischen, variablen und kontingenten Sinne existiert, es notwendigerweise mittels Infrastruktur wie Bildschirm oder Beamer als analoges Bild visualisiert werden muss und derart digitale Daten darstellt.²⁷ Kurzum: Numerischer Code und elektronische Signale sind per se kein Bild.²⁸ Was den Status der gedruckten Computergrafik angeht, ist festzustellen: Während manche Autoren für das digitale Bild ein Flüchtigkeit evozierendes, stets aktualisierendes Prozessieren („refresh“) innerhalb eines elektronisch-digitalen Systems erfüllt wissen wollen – im ‚Digital Materialism‘ wird in Anlehnung an den ‚New Materialism‘ von einem vibratorischen Status gesprochen²⁹ – beziehen andere auch dauerhaftere, sozusagen ‚still gestellte‘ Materialisierungen wie beispielsweise ein (3D-)Druck mit ein, weil auch solche Bilder in einer gewissen, wenn auch nicht mehr aktuellen und aktualisierbaren Verbindung mit ihrem Code stehen, ihre Spezifität in ihrer digital-technischen Provenienz liegt und sie ansonsten als Forschungsgegenstand digitaler Kultur ausgeschlossen würden.³⁰

Zusammengefasst ist für unsere Zwecke festzuhalten: Das digitale Bild wird als ein computer-technisches Bild mit ontologischem Doppelstatus aufgefasst, dessen Voraussetzungen Code und Programmierung sind. Davon gibt es aktualisierbare, interne, wie ich es beschreiben möchte, etwa auf dem Bildschirm, und nicht-aktualisierbare, externe digitale Bilder wie Nakes Computergrafik.

3. Unser Programm

Vor diesem Hintergrund und die historische Situation aufgreifend, sollen im Folgenden ausgehend von der Computergrafik *Hommage à Paul Klee* medientheoretische Überlegungen zum digitalen Bild angestellt werden, die im Prinzip auch auf interne digitale Bilder übertragen werden könnten. Um den Blick vom Sujet auf die technisch-medialen Bedingtheiten, auf Prozesse und Handlungen und derart die historische Verfasstheit des Bildlichen zu lenken, für die William J. T. Mitchell insbesondere die „soziale Praxis“³¹ berücksichtigt wissen wollte, ist diese Studie praxeologisch und medienarchäologisch grundiert.³² Zur Bestimmung des Digitalen genügt es eben nicht, um der Forderung von Wolfgang Ernst nach einer materialistischen Medienarchäologie zu folgen, wie Goodman eine Symboltheorie des Ästhetischen vorzulegen, in der das Digitale zwar gedacht, nicht aber die „Arbeit der tatsächlich in Hardware realisierten Rechenmaschine“ berücksichtigt wird.³³ Eine Argumentation über tiefgründige Betrachtungen technischer Funktionalitäten allein wird der historischen Situation jedoch nicht gerecht. Nimmt man den Entstehungsvorgang von Nakes Computergrafik in einem produktionsästhetischen Sinne in den Blick, so ist zu diagnostizieren, dass sie auch am Computer, tatsächlich aber in einem Mensch-Maschinen-Medien-Netzwerk im Sinne der Akteur-Netzwerk-Theorie (ANT) entstand, insbesondere im Zusammenspiel von Mensch, Maschinen, Programm, Diagramm

und ihren Praktiken.³⁴ Nakes Diagramme, hier verstanden „im engeren Sinne einer Kombination aus Zeichnung, Linie und Schrift“³⁵, wurden vereinzelt veröffentlicht und ihre Funktion angesprochen, aber meiner Kenntnis nach nicht in weitergehende Betrachtungen einbezogen, insbesondere was seine Praktiken angeht.³⁶ Würde man nach dem Dispositiv der Computergrafik der 1960er Jahre im Sinne Michel Foucaults fragen, so müssten und könnten die Kreise leicht größer gezogen werden: es wäre über notwendige disziplinäre Expertise wie Programmierkenntnisse, institutionelle Aspekte wie den Zugang zu und Zugriff auf sperrige, hochpreisige und nicht leicht verfügbare Großrechner und über deren Anwendungskontexte und -ziele zu sprechen.³⁷ Das würde auch deshalb Sinn machen, da Nake ein politischer Mensch war und ist. Insbesondere kritisierte er um 1970 die Situation der Computerkunst, weil sie vom Markt vereinnahmt worden sei und beendete seine Produktion zeitweilig.³⁸ Auch wenn das digitale Bild mittlerweile zurecht hinsichtlich seiner soziopolitischen Dimensionen erforscht wird,³⁹ soll es hier, über seine digitaltechnische Basis gekennzeichnet, im obigen Sinne untersucht werden. Thematisiert wird damit auch der historische Wandel ästhetisch-kultureller Produktion, der sich der Kunstkritik der 1960er Jahre zufolge an die „Kuriosa automatischer Gestaltungen“⁴⁰ der Kulturgeschichte seit Leonardo anschließen lasse.

Auf die (technische) Programmierung, gemeint ist der Vorgang der Programmerstellung im Zusammenhang mit Computern,⁴¹ die in den 1960er Jahren nicht bloß in der Informatik, sondern auch in Kunst und Philosophie ein Thema war und mittlerweile zum Gegenstand der Medienwissenschaft wurde, wird zuerst fokussiert. Gut nachvollziehbar kann der Programmbegriff, wie Hartmut Winkler zu bedenken gab, dafür dienen, die wechselseitigen Überführungen von „Prozessen/Praxen“ in „materiellen Niederlegungen“ sowie ihre Übergangsstellen zu erforschen.⁴² Angesprochen ist im vorliegenden Fall weniger eine ‚direkte‘ Arbeit am Material eines Werkes, wie in althergebrachter manueller Grafik oder Bildhauerei, die immer auch an Techniken gebunden waren, sondern vielmehr, dass digitale Objekte wie der Code und Praktiken wie die Programmierung im Anschluss an Ansätze des Digitalen Materialismus, ausgehend von der Prämisse, dass es keine reine Information gibt, als materialbasiert und in einer Kette manuell und maschinell organisierter medialer Transformationen stehend, aufgefasst werden.⁴³

Mit der angesprochenen Vorgehensweise erlaubt es diese Studie bezüglich der Grundlagen technischer Bilder, ihrer Produktion sowie der dafür notwendigen Verarbeitung, Übertragung und Speicherung von Daten,⁴⁴ an konkretem Material zu argumentieren. Die historisch-praxeologische Perspektive, die hier für die Kunstgeschichte stark gemacht werden soll, wird bestätigen, dass Programmierung ein an Hard- und Software-Konfigurationen sowie Handlungen gebundener Vorgang ist und derart, was oftmals in der Geschichte der digitalen Kunst unberücksichtigt bleibt, erhebliche historische Varianz zeigt. Florian Cramer konstatierte diesbezüglich gar eine „Geschichte der Ignoranz gegenüber Programmierung und Programmierern“ in Verbindung mit einem Denken in „black boxes“.⁴⁵ So sollen die technisch bedingten Abläufe rekonstruiert werden. Dies schließt die Behandlung von Datenträgern und ihrer Produktion sowie Funktion ein. Dem grundlegenden Problem der Rekonstruktion vergangener „Prozesse und Praxen“⁴⁶, das Winkler für das Programmieren explizit ansprach, wird durch den Einbezug von historischen Quellen sowie Auskünften von Frieder Nake entgegengewirkt. Über die Datenträger führt diese Studie, obwohl Dieter Mersch das Diagramm als ‚das‘ Medium des Programms ausmachte,⁴⁷ auf einen blinden Fleck: Der diagrammatische Entwurf – von Programmen und letztlich Bildern, den es darzulegen und wofür es Erkenntnisse der Diagrammatik und Medienwissenschaft zum Verhältnis von Programm und Diagramm heranzuziehen gilt. Davon berührt wird auch ein etabliertes Text-Bild-Verhältnis: Man sagt heute noch „ein Computerprogramm

schreiben“ und behandelt den Programm-Code als „Text“ beziehungsweise „Schrift“;⁴⁸ im Folgenden wird entgegen dem etablierten binären Bild-Code- beziehungsweise Bild-Text-Verhältnis dafür plädiert, dass Bild, Code und Diagramm zusammenzudenken sind. Programmieren wird hier, im Einklang mit der Post-digital-Debatte, als verschränkt mit den ‚analogen‘ Kulturtechniken Schreiben und Zeichnen auf dem Papier vorgestellt werden.⁴⁹ Es ist ein Anliegen dieser Studie, binäre Denkweisen zugunsten nicht oder wenig berücksichtigter Dritter, insbesondere des Diagramms, zur Disposition zu stellen. Derart besitzt sie multiple Stoßrichtungen: Sie will die (Kunst-)Geschichte der frühen (künstlerischen) Computergrafik erweitern und damit einen Beitrag zur Theorie der Zeichnung beziehungsweise Grafik, digitaler Kultur sowie künstlerischer Schaffensprozesse leisten.

4. Im Zentrum des Rechnens: Programmieren um 1965

Wie lief der Entstehungsprozess der Grafik *Hommage à Paul Klee* ab? In den 1960er Jahren war die Vorstellung verbreitet, dass der Computer von selbst arbeitet – diese mehr als fragwürdige Vorstellung von der Autonomie der Maschine wirkt bis in die aktuellste Debatte um Künstliche Intelligenz (KI) hinein. Als der Kunstkritiker Günther Vogt in der Frankfurter Allgemeinen Zeitung vom 9. Februar 1966 bezüglich der künstlerischen Computergrafik die wiederkehrende Rede vom Knopfdruck bediente, so relativierte er sie immerhin, indem er notwendige Vorarbeiten eines (selbstredend männlichen) Programmierers beschrieb.⁵⁰ Diese Rede mag sich durch die vorherrschende, auch existentiell aufgeladene, besonders auf den Bereich der Ökonomie bezogene Automatisierungsdebatte sowie die Geschichte von Apparate-Werbung erklären, die (etwa im Haushalt) Simplifizierung versprach und in welcher der Kodak-Slogan für Fotoapparate ‚You press the button, we do the rest.‘ vermutlich am nachhaltigsten Eingang in das kollektive Gedächtnis gefunden hat. Der Realität der Computerpraxis in den 1960er Jahren entsprach dem keineswegs, im Gegenteil: Die Herstellung von Computergrafik war ein aufwendiger, mehrstufiger, auch nicht linearer, teils rekursiver Prozess, der sich, je nach technischen Gegebenheiten, mehr oder weniger komplex gestaltete. Man ist geneigt zu sagen, Nake drückte Knöpfe und tat den Rest. Im Fall des Settings an der TU Stuttgart handelte es sich wenigstens um ein Zusammenwirken von Programmierer, Computer, Programmen, Datenträgern sowie Peripheriegeräten, insbesondere einem Plotter sowie Papier und Farbstiften. Im Kern musste ein Programm erstellt und auf dem Computer ausgeführt, dessen Ergebnisse über einen Datenträger auf den Plotter übertragen und von diesem gedruckt werden. Demgemäß beschrieb Nake 1966 seine Vorgehensweise, typisch für die Zeit die Automatisierung betonend, mit folgenden „drei Arbeitsgängen“: 1. „Aufstellen eines Programms für die Rechenanlage“, 2. „Automatisches Durchführen des Programms auf der Rechenanlage“, 3. „Automatisches Umsetzen der ausgegebenen Informationen, die in einem Lochstreifen enthalten sind, in die Linien der Zeichnung durch die Zeichenmaschine“.⁵¹

Konkret war es die Aufgabe des Programmierers, den Rechenautomat in solch einer Weise in Maschinencode beziehungsweise Assembler mit Anweisungen beziehungsweise Zahlen zu ‚füttern‘, so dass ein Programm, weil es zu dieser Zeit zumeist nicht kommerziell erworben werden konnte, zur Verfügung stand, das bei seiner Ausführung die gewünschten Berechnungen durchführte und Ergebnisse lieferte. Dafür besaß der SEL ER 56 ein Bedienungspult mit Kontrolllampen, einer Uhr und vor allem Knöpfen und Drehreglern als Eingabeeinheit, das um eine Tastatur ergänzt werden konnte (Abb. 3). Die sieben Drehschalter (links unten) dienten der Eingabe der Werte 0 bis 9 in die Bereiche ‚Adresse‘ (vier Stellen), ‚Register‘ (eine Stelle eines besonders schnellen Speichers) und ‚Code‘ (zwei Stellen), zum Beispiel bedeutete „1000/1/15“ Addieren, was in der siebenstelligen numerischen Anzeige darüber angezeigt wurde (hier „0000010“). In der 14-stelligen Anzeige ganz oben gaben die ers-

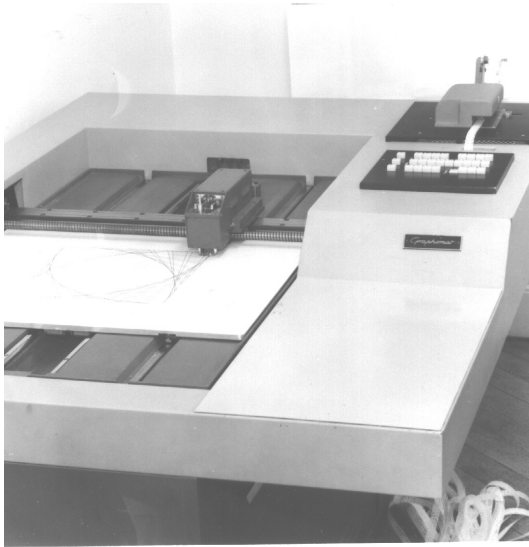


4 ZUSE Z 64 Graphomat, Anlage an der Technischen Universität Stuttgart (vermutlich 1965, nicht später als 1966)

ten sieben Stellen den Inhalt des Akkumulators, die nächsten sieben Stellen den Inhalt des Multiplikators in Dezimalziffern von 0 bis 9 an.

Wie die Programmierer seiner Zeit, musste Nake, anders als in der Theorie des Programmierens, die sich an dem idealen, abstrakt-mathematischen, mit sequenziellen Abläufen operierenden Modell der Turing Maschine orientierte, mit einer realen, konkreten, physikalischen Maschine und ihren Eigenheiten zurechtkommen.⁵² Diese Diskrepanz zwischen Theorie und Praxis war seit den 1940er Jahren erkannt worden: Programmierung galt nicht mehr als eine rationale und simple ‚straight-forward‘-Angelegenheit, sondern eine spezielle intellektuelle Aktivität, die Kreativität, ein individuelles Vermögen und einen idiosynkratischen Stil erforderte, weshalb – wohl auch in Anlehnung an die frühe Druckkunst – im Feld der ‚Computer Science‘ von ‚black art‘⁵³ und mit Blick auf die notwendigen, aber merkwürdigen und trickreichen Vorgehensweisen von ‚unhygienic‘-creativity⁵⁴ gesprochen wurde.⁵⁵ Das ‚Problem des Programmierens‘⁵⁶ hatte sich in den 1960er Jahren verschärft, da immer mehr und leistungsfähigere Computer nicht mehr nur für spezifische Nutzungsanforderungen von Kunden gefertigt und eingerichtet wurden, sondern von diesen selbst für alle erdenklichen Aufgaben – dem Versprechen des Universalrechners gemäß – genutzt und programmiert werden wollten. Die Kluft zwischen der Leistungsfähigkeit von Hardware und Programmen wurde als ‚software crisis‘⁵⁷ bezeichnet.

An der TU Stuttgart musste Frieder Nake 1963, als der Zeichenautomat Z 64 Graphomat der Konrad Zuse KG Bad Hersfeld installiert wurde, ein ‚Übersetzungsprogramm‘⁵⁸ erstellen, weil der Plotter mit dem SEL ER 56, also einem Großrechner eines anderen Herstellers betrieben werden sollte und für den Datenaustausch zwischen beiden Maschinen kein Druckertreiber, wie wir heute sagen würden, existierte. Im Zuge der Erarbeitung dieser Software, die für wissenschaftliche Anwendungen gedacht war, überlegte Nake eigene Grafiken zu schaffen. Einerseits, um den Graphomat aus-



5 ZUSE Z 64 Graphomat in einer kompakteren Version, ein Teil des Lochstreifens liegt auf dem Boden (rechts unten)

zuprobieren und funktional zu testen, andererseits um ihn zu erkunden und seinen Möglichkeitsraum auszuloten. Es ist ein interessantes Detail, dass Nake seinen Gerätetest optimierte, indem er die Ansteuerung des Plotters nicht systematisch in alle Richtungen beziehungsweise für alle Positionen durchführte, sondern den selbst programmierten Zufall, auf Basis des informatischen Pseudo-Zufalls, nutzte, um nur die wenigen, sich daraus ergebenden Positionen, aber zugleich den Automat mit hinreichender Wahrscheinlichkeit zu prüfen.⁵⁹ Dies sei auch deshalb erwähnt, weil diese zufallsbasierte Vorgehensweise in gewisser Hinsicht der zur gleichen Zeit verbreiteten Strategie der methodischen Systematisierung von Entwurfsprozessen in Kunst und Design entgegenstand und der Zufall als Methode wegen seiner Nähe zur zeitgenössischen Physik in der Kunst, etwa in-

nerhalb der Minimal Art, auch abgelehnt wurde.⁶⁰ War schon Nakes Programmierpraxis für die wissenschaftliche Arbeit weit entfernt vom „Konsumverhalten des Knopfdrucks“, wie Friedrich Kittler es einmal nannte, so musste er für seine zweckfreien Experimente mit der Computergrafik umso mehr „mit allen Knöpfen spielen“, wie Kittler sein Plädoyer für „Synergien zwischen Mensch und Maschine“ im Sinne einer kritisch-explorativen Praxis im Umgang mit Medientechnik umschrieb.⁶¹ Künstliche Intelligenz, die Nake während seines Studiums kennengelernt hatte, wurde in den 1960er Jahren in seinem Arbeitsumfeld diskutiert, sie spielte jedoch für seine Arbeit keine Rolle.⁶²

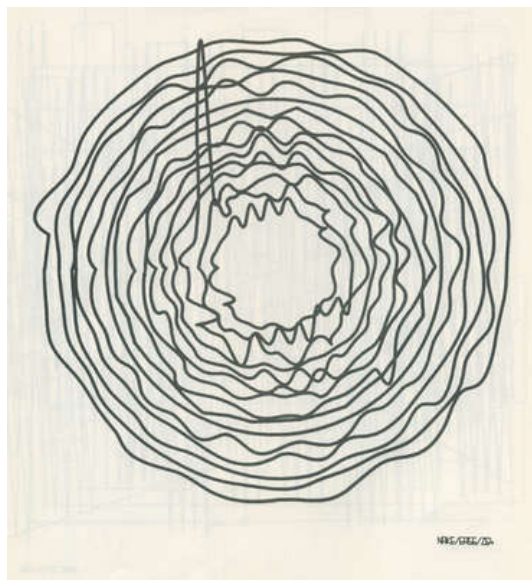
5. Computergrafik oder das Programm als Lochstreifen

Gedruckt wurde *Hommage à Paul Klee* mit einem Plotter, die hauptsächliche ‚output-device‘ der 1960er Jahre. Der eingesetzte ZUSE Z 64 Graphomat (Abb. 4 und 5) war als ein lochkarten- oder lochstreifengesteuerter Zeichentisch und für Anwendungen des „automatischen Zeichnens“⁶³ etwa im Hoch- und Tiefbau, in der Geodäsie oder wissenschaftlichen Visualisierung entwickelt worden. Er bestand im Wesentlichen aus einer Bedienkonsole zur direkten Ansteuerung, einem Lochstreifenleser, dem flachen Zeichentisch für Großformate (Modell G1: 550 × 600 Millimeter, Modell G4: maximal 1200 × 1400 Millimeter) sowie dem Zeichenkopf, der mit vier Tuschestiften (Rapidograph), Punktstech- oder Graviernadeln bestückt und von zwei Dualstufengetrieben, mit einer Feinheit von einem sechzehntel Millimeter über Längs- und Querbewegungen in 225 Richtungen, exakt navigiert werden konnte, sodass gerade Strecken, Kurven, Buchstaben, Ziffern und weitere Schriftzeichen ausgegeben werden konnten, was dem Hersteller zufolge auf Papier, Karton, Folien, beschichtetem Glas und Metall geschehen konnte.⁶⁴

Damit unterschied der Plotter sich beispielsweise von dem SEL Zeilen-Schnelldrucker SP 160, der keine Stifte aufnehmen konnte, und dem SEL Mosaik-Schnelldrucker, dessen Druckzeichen aus 5 × 5 Punkten zusammengesetzt wurden; beide Geräte konnten nur alphanumerische Zeichen drucken.⁶⁵ Nakes Grafik *Kreis-Variationen* (1966), in welcher die Radien konzentrischer Kreise zufällig

variiert werden, belegt eindrucksvoll, dass der Graphomat nicht bloß – wie oftmals in der Literatur beschrieben – gerade, sondern auch gekrümmte Linien zeichnen konnte (Abb. 6).

Es ist ein entscheidender Aspekt, dass der ZUSE Z 64 Graphomat in einer späteren Version, Compugraph genannt, „on-line“ [sic!], das heißt über ein Datenkabel, mit dem Rechner Zuse Z 25 betrieben werden konnte,⁶⁶ dass aber an der TU Stuttgart Graphomat und SEL ER 56 nicht direkt aneinander angeschlossen waren. Die Ergebnisse des Großrechners mussten zuerst auf einem Lochstreifen ausgegeben werden (auch eine Lochkarte war möglich), wofür SEL den Lochstreifenstanzer SL 614 und Lochstreifen-Schnellstanzer CR 3000 anbot. Der papierne Datenträger – heute würden wir einen USB-Stick für den Datenaustausch in relativer räumlicher Nähe benutzen – wurde vom Graphomat eingelesen. Auf Basis



6 Frieder Nake, *Kreis-Variationen, Nr. 3* (04.05.1965), Computergrafik mit Zuse Z 64 Graphomat, schwarze Tinte auf Papier, 25 x 25 cm

dieser Anweisungen und indirekten Ansteuerung wurde *Hommage à Paul Klee* gedruckt. Man könnte formulieren: Das Programm als Lochstreifen. Tatsächlich erläuterte Nake 1967: „In unserem Fall ist das ‚Programm‘ ein Lochstreifen, ein Magnetband oder ein Lochkartenstapel – je nachdem wie die Zeichenanlage gesteuert wird. Sie verarbeitet das vom Rechner gelieferte Material so, daß die Zeichnung entsteht.“⁶⁷ Allerdings war dies nicht immer so. Nachdem Nake eine Zeichnung programmiert und einen Lochstreifen erstellt hatte und damit nach Bad Hersfeld, dem Standort der Konrad Zuse AG, mit dem Zug fuhr, um den dortigen Graphomat zu nutzen, blieb aber stecken.⁶⁸ Die Anekdote lässt uns an dieser Stelle drei Aspekte festhalten: Digitale Bilder können durch die Prozessierung mit einem Programm oder durch die Überführung von Daten in Bilder erzeugt werden, letzteres kann mit beliebigen Daten (Visualisierung) und formatierten, konventionellen Bilddaten (Dateiformate für Grafikprogramm oder einen Drucker) geschehen. Die Datenübertragung mittels Lochstreifen über große räumliche Distanzen lässt erkennen, dass gewisse Charakteristika des digitalen Bildes wie Zirkulation und Distribution nicht intrinsisch, sondern an externe Infrastrukturen wie Computer-Netzwerke – wie das Internet – gebunden und historisch unterschiedlich ausgeprägt sind. Vor der Folie der Kybernetik sowie der Maschinisierung, Computerisierung und Mathematisierung der 1960er Jahre lassen Minimal, Serial und Conceptual Art eine Auseinandersetzung mit und ohne Maschinen daran sowie eine Kritik an den Versprechen von Automatisierung, Rationalität und Kontrolle erkennen.⁶⁹

Kehren wir zur Herstellung von *Hommage à Paul Klee* zurück. Die Grafik entstand in mehreren Druckphasen, beispielsweise wurden die Kreise als Bildelemente in einem zweiten Durchlauf aufgebracht.⁷⁰ Dieser Aspekt lässt sich auf zwei grundlegende Prinzipien der Gestaltung beim Zusammenreffen unterschiedlicher Bildelemente am gleichen Ort beziehen, wie es auch in der althergebrachten Grafik und Malerei der Fall ist: die Überlagerung und die Aussparung. Konrad Zuse erläuterte dies 1965 für die Computergrafik und mit Blick auf Kompositionen mit variierten Kreisen: Es könnte ent-

weder die gesamte Grafik in einem Zug auf der Grundlage der vollständig vorbereiteten und dafür berechneten Bilddaten oder die Grafik sukzessive auf der Grundlage jeweils aktuell berechneter Anteile gedruckt werden; im ersten Fall könnten Überlagerungen durch Vorkehrungen im Programm umgangen, im zweiten Fall nicht ausgeschlossen werden.⁷¹ Nakes Arbeit zu den *Kreis-Variationen* (1966) ist in diesem Zusammenhang zu sehen und dies gilt umso mehr, wenn man berücksichtigt, dass sich Zuses-Manuskript im Besitz von Nake befindet.

Vor Ort an der TU Stuttgart musste der Druckvorgang von *Hommage à Paul Klee* kontrolliert und gegebenenfalls in denselben eingegriffen werden: Die Tusche in den Stiften konnte eintrocknen, wenn sie längere Zeit nicht gebraucht wurden, was sich in der Farbgebung niederschlug. Das stellenweise feucht werdende Papier – Nake legte Wert auf qualitativvolles und verwendete nicht etwa Standard-Druckerpapier – konnte sich wellen und musste korrigierend mit der Hand flach- und dafür auf den Untergrund gedrückt werden.⁷² Er selbst beschrieb solcherlei Widrigkeiten: „Doch ist es in praxi eben so, daß Sie die Maschine selten unbewacht lassen können: die Tuschefüller trocknen ein oder schmierern, das Papier liegt nicht glatt auf der Glasplatte, der Lochstreifen reißt aus oder ist nicht einwandfrei gelocht.“⁷³ Nake, der seine Grafiken durchaus außerhalb der gewöhnlichen Arbeitszeiten schuf, wenn die Computersysteme nicht anderweitig im Einsatz waren, musste, um zeitökonomisch zu arbeiten, zwischen dem gekühlten Drucker- und Rechnerraum und seinem ungekühlten Arbeitsplatz „hin- und her rennen“.⁷⁴ Manchmal entstanden dennoch Druckfehler beim Plotten, Artefakte auf dem Papier durch fehlerhafte Algorithmen oder Plotterfunktionen.⁷⁵ In solch unvorhergesehenen Phänomenen geht der Eigensinn der involvierten Materialien, ihre ‚Agency‘ ein, zeigt sich im Bild die Technizität der Medialität. Dabei handelt es sich nicht um eine Störung im Sinne einer künstlerischen Strategie, sondern um Nebeneffekte der experimentellen Untersuchung eines Medien-Werdens. Es wird verständlich, weshalb Nake, wenn er 1967 feststellte, dass es in der Kunst Bereiche gebe, die „heute eben einfach programmierbar sind“, zugleich klarstellte: „Mit Ausmerzung der Seele, Abschaffung des Schöpfungsprozesses, kaltem Automatismus hat das nichts, aber auch gar nichts zu tun.“⁷⁶ Die Unwägbarkeiten bei der Materialisierung der digitalen Bilder machen umso mehr darauf aufmerksam, dass keine eindeutige Abbildung zwischen Code und digitalem Bild in einem mathematischen Sinne existiert, wie Gernot Grube es einmal vorschlug.⁷⁷ Das digitale Bild unterhält, um mit Claus Pias zu sprechen, ein „paradoxes Verhältnis von Information und Präsentation“.⁷⁸ Letztlich betrifft dies auch die ästhetische Erfahrung eines Bildes, die nicht ohne eine konkrete Materialisierung denkbar ist, welche allerdings mit den Daten des Bildcodes nicht erschöpfend bestimmt werden kann – und dies gilt auch für „Bildschirmerscheinungen“⁷⁹, auf die sich Grube bezog (Stichworte sind Hardware-Spezifika wie Auflösung, Farbeinstellung, Kalibrierung). „Variabilität“⁸⁰, wie Lev Manovich es benannte, ist aufgrund vielfältiger rechnerischer Modifikationsmöglichkeiten, etwa Farbtransformation, Skalierung oder Verzerrung, geradezu ein Kennzeichen digitaler Bilder, besonders solcher, die in ihrem digital-technischen System und also aktualisierbar bleiben. Im Speziellen kann ein Programm so angelegt sein, indem es etwa mit Pseudo-Zufall arbeitet, dass es viele unterschiedliche Bilder erzeugen kann.⁸¹ Um dieses Verhältnis von Potenz und Akt zu fassen, bestimmte Frieder Nake, als Mathematiker, Platonist und Marxist, das Verhältnis von Programm und Bild über das Konzept der „Klasse“⁸². Ein Programm, das nur eine einzige Grafik erzeugen kann, ist denkbar und würde eine Klasse mit genau einem Element erzeugen – so etwas sei nach Nake der Programmierung nicht würdig und verkenne die Möglichkeiten des Computers.⁸³ Das Konzept der Klasse erinnert uns daran, dass die Maschinen dieser Zeit nicht kreierten, vielmehr sind die realisierten Zeichnungen, die uns als Kreativleistungen der Maschine vorkommen mögen, alle in dem durch das Programm definierten Möglichkeitsraum enthal-

ten – daran ändern auch die oben angesprochenen Unvorhersehbarkeiten bei ihrer Materialisierung nichts. Was den Zufall angeht, so nutzte Nake nicht die Gleichverteilung, sondern arbeitete als promovierter Stochastiker mit eigenen Zufallsmethoden und insbesondere mit bedingter Wahrscheinlichkeit.⁸⁴ Der Gebrauch des Zufalls adressiert, so möchte ich vorschlagen, das ohnehin mit der Computerarbeit aufgerufene kunsttheoretische Verhältnis von Original und Kopie. So werden die konkreten grafischen Blätter durch die mathematische Bestimmungsgröße Zufall in ihrem So-oder-So-Sein nicht nur auf einer ästhetischen, sondern auch auf einer konzeptuellen Ebene differenziert. Nicht alle möglichen Computergrafiken eines Programms – die Elemente einer Bild-Klasse – sind gleichwertig, weil manche grafischen Ausprägungen wahrscheinlicher, andere unwahrscheinlicher sind. Wird beispielsweise für jedes Feld eines gerasterten Blattes mit einer gegebenen Zufallsfunktion über die Füllung mit schwarzer Farbe entschieden, so ist ein vollständig bedrucktes oder unbedrucktes Blatt äußerst unwahrscheinlich.

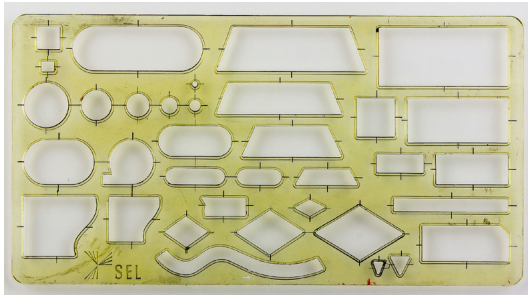
Doch kehren wir zurück zum Stuttgarter Produktionsprozess. Neben den oben angesprochenen, korrigierenden Eingriffen während des Druckvorgangs brach Nake – dessen Arbeitsweisen nicht immer gleich waren – manchmal aus ästhetischen Gründen den Druckvorgang vorzeitig manuell ab, auch um die (künstlerische) Kontrolle zu behalten, wie er rückblickend erklärte; so ist auf seiner Grafik 16.3.65 (Tinte auf Papier, 22 × 22 Zentimeter, 1965) neben der Gesamtdruckzeit 17 Minuten auch die Abbruchzeit 7:30 Minuten notiert.⁸⁵ Das Ineinandergreifen von maschinellen Prozessen und menschlichen Handlungen ist ein Beleg dafür, dass die in der Forschungsliteratur vorhandene Theoretisierung des Schaffens generativer Kunstformen in der Façon „ein Künstler schafft ein sprachliches, logisches oder maschinelles System, das wiederum Kunstwerke generiert“, wie es oft für Sol LeWitt vorgebracht wurde, zwar keine prinzipiell falsche, aber in Anbetracht der vielen möglichen maschinenbezogenen Produktionsweisen eine unterkomplexe Beschreibung darstellt.⁸⁶ Es handelt sich bei Nake eben nicht um „reine Maschinenkunst“⁸⁷, wie es die Kunstkritik beschrieb, um die Rolle des Computers zu betonen.

Es ist ein zweiter dünner Lochstreifen im Produktionsprozess, an dem die weitere Argumentation hängt: Die oben beschriebene Programmierung war direkt am Rechner möglich, aber praktisch zu mühsam. So erstellte Nake zunächst einen ersten Lochstreifen, der das Programm für den SEL ER 56 enthielt und über dessen Lochstreifen-Eingabewerk in den Arbeitsspeicher eingespeist und dort zum elektronischen Programm wurde.⁸⁸ Dieses Programm als Lochstreifen wurde, wie es üblich war, von Hand gegebenenfalls mit einem Handlocher gestanzt und führt uns auf weitere „Programmierung-

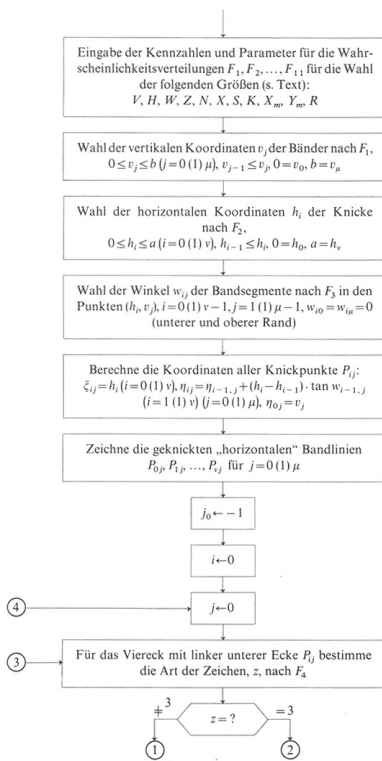
Blatt

RECHENINSTITUT TH-STUTTGART		ZEI 56		ER 56	
Pos.	Befehle	Befehl symbolisch	Bemerkungen	Zeit	
1	4000 0 0 7		Startwert		
2	0000 0 0 7		Startwert		
3	2000 0 0 7		100 * 10		
4	4000 0 0 7				
5	6000 0 0 7				
6	8000 0 0 7				
7	0000 0 0 7				
8	2000 0 0 7				
9	4000 0 0 7				
10	6000 0 0 7				
11	8000 0 0 7				
12	0000 0 0 7				
13	2000 0 0 7				
14	4000 0 0 7				
15	6000 0 0 7				
16	8000 0 0 7				
17	0000 0 0 7				
18	2000 0 0 7				
19	4000 0 0 7				
20	6000 0 0 7				
21	8000 0 0 7				
22	0000 0 0 7				
23	2000 0 0 7				
24	4000 0 0 7				
25	6000 0 0 7				
26	8000 0 0 7				
27	0000 0 0 7				
28	2000 0 0 7				
29	4000 0 0 7				
30	6000 0 0 7				
31	8000 0 0 7				
32	0000 0 0 7				
33	2000 0 0 7				
34	4000 0 0 7				
35	6000 0 0 7				
36	8000 0 0 7				
37	0000 0 0 7				
38	2000 0 0 7				
39	4000 0 0 7				
40	6000 0 0 7				
41	8000 0 0 7				
42	0000 0 0 7				
43	2000 0 0 7				
44	4000 0 0 7				
45	6000 0 0 7				
46	8000 0 0 7				
47	0000 0 0 7				
48	2000 0 0 7				
49	4000 0 0 7				
50	6000 0 0 7				
51	8000 0 0 7				
52	0000 0 0 7				
53	2000 0 0 7				
54	4000 0 0 7				
55	6000 0 0 7				
56	8000 0 0 7				
57	0000 0 0 7				
58	2000 0 0 7				
59	4000 0 0 7				
60	6000 0 0 7				
61	8000 0 0 7				
62	0000 0 0 7				
63	2000 0 0 7				
64	4000 0 0 7				
65	6000 0 0 7				
66	8000 0 0 7				
67	0000 0 0 7				
68	2000 0 0 7				
69	4000 0 0 7				
70	6000 0 0 7				
71	8000 0 0 7				
72	0000 0 0 7				
73	2000 0 0 7				
74	4000 0 0 7				
75	6000 0 0 7				
76	8000 0 0 7				
77	0000 0 0 7				
78	2000 0 0 7				
79	4000 0 0 7				
80	6000 0 0 7				
81	8000 0 0 7				
82	0000 0 0 7				
83	2000 0 0 7				
84	4000 0 0 7				
85	6000 0 0 7				
86	8000 0 0 7				
87	0000 0 0 7				
88	2000 0 0 7				
89	4000 0 0 7				
90	6000 0 0 7				
91	8000 0 0 7				
92	0000 0 0 7				
93	2000 0 0 7				
94	4000 0 0 7				
95	6000 0 0 7				
96	8000 0 0 7				
97	0000 0 0 7				
98	2000 0 0 7				
99	4000 0 0 7				
100	6000 0 0 7				

7 Frieder Nake, Code des Teilprogramms ZeiPro (Zeichenprogramm) des Programmpakets COMPART ER 56 (begonnen 1964)



8 Schablone aus Kunststoff für Flussdiagramme, Zubehör zum SEL 9829



9 Frieder Nake, Flussdiagramm zu *Hommage à Paul Klee*, 1974 (1965)

gezeichnet und ihre Texte sind gesetzt, was die ‚Lesbarkeit‘ befördert (Abb. 9). Solch editierte Flussdiagramme wurden für Kommunikations-, Dokumentations- und Publikationszwecke, auch im Nachhinein, auf der Basis von bereits vorliegendem Programmcode angefertigt. In unserem Fall verweisen sie auf vorgängige Hand-Diagramme.

Gut nachvollziehbar können Flussdiagramme im Speziellen als ein Kommunikationsmedium in arbeitsteiligen und teambasierten Programmierprozessen dienen, wie es auch dem Handbuch zum Universal Automatic Computer *Introduction to Programming* (1949) zu entnehmen ist: Der Program-

gen von Hand‘: Nake schrieb seine Programme zuerst mit dem Stift auf Papier – beide Hilfsmittel liegen auf dem Arbeitstisch vor der Konsole des SEL ER 56 (Abb. 3). Für diese Arbeit standen vorgedruckte Codierungs-Blätter zur Verfügung (Abb. 7). Insbesondere, wenn es zu kompliziert wurde, etwa für komplexere Programme, nutzte Nake Flussdiagramme.⁸⁹

6. Das Programm als Diagramm

Ein Flussdiagramm (engl. ‚flow chart‘) wird im Kontext der Programmierung verwendet, um die Funktionslogik eines Programms festzuhalten, zu kommunizieren und nicht zuletzt auszuarbeiten.⁹⁰ In ihm können die Programmschritte je nach ihrer Funktion durch spezielle grafische Elemente – zum Beispiel ein Rechteck für eine einzelne Aktion, eine Raute für eine bedingte Verzweigung auf Basis einer Entscheidung oder ein Parallelogramm für eine Aus- oder Eingabe (etwa nach DIN 66001) – mit darin befindlichen Anweisungen wiedergegeben und die lineare Abfolge (Sequenz) entsprechend der Verbindungen durch Linien oder Pfeile organisiert werden. Im Diagramm festgehalten ist im Speziellen eine ‚reine‘, von Programmiersprachen losgelöste Funktionalität, deren Basis das Repertoire logischer Funktionen ist – ein Algorithmus. Zeichenschablonen aus Kunststoff waren als Zubehör für Großrechner wie den SEL ER 9829 erhältlich (Abb. 8). Noch in den 1970er Jahren wurden Lehrbücher der Datenverarbeitung mit Schablonen aus Papier ausgeliefert und deren Anwendung besprochen.⁹¹ Mit einer Schablone sind auch die Formen der Flussdiagramme zur Computergrafik *Hommage à Paul Klee* ge-

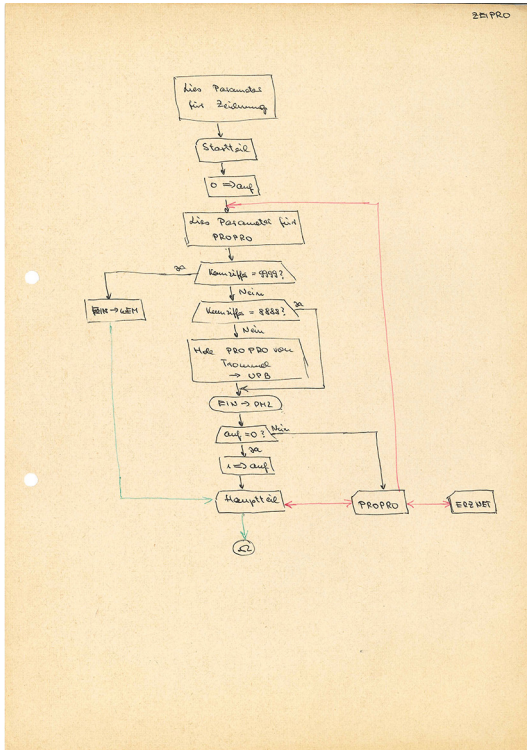
mierer („programmer“) erstellt das Diagramm („flow chart“), nachdem er das zu lösende Problem studiert und eine angemessene Methode bestimmt hat. Der Codierer („coder“) setzt es in spezifische Instruktionen, etwa als Programmcode oder Lochkarte, in Hinsicht auf die Maschine um.⁹² Interessanterweise wurde hier in Anlehnung an das Programmiermodell von Herman Goldstine und John von Neumann zwischen einem „managerial programmer“, dem die eigentliche Aufgabe der Analyse zukam, und dem „technical coder“, dessen Tätigkeit als „trivial and mechanical“ angesehen wurde, differenziert.⁹³

Die Bezeichnung des Entwicklers des Flussdiagramms als Programmierer überrascht insofern, weil sie einer verbreiteten Unterscheidung von Algorithmus und Programm zuwiderläuft, die letztlich über das Kriterium des Maschinenbezugs vorgenommen wird: Während ein Algorithmus als eine Abfolge von Operationen zur Lösung eines Problems auch abstrakterer Natur und etwa in einem Flussdiagramm oder auch natürlich-sprachlich festgehalten sein kann, ist das korrespondierende Programm auf eine konkrete Rechnerkonfiguration bezogen und in einer Programmiersprache wie Assembler, Algol oder Fortran formuliert.⁹⁴ Dementgegen hieß Programmieren im historischen Sprachgebrauch der USA auch ein Flussdiagramm zu schaffen, das den Algorithmus repräsentiert. Diese Auffassung lässt sich ebenso im Deutschland der 1960er Jahre ausmachen, wie das Beispiel eines Berichts im ZUSE Forum 1964 über den Neubau des Rechenzentrums der Firma Rheinstahl-Wagner in Dortmund, einem Kunden der ZUSE KG, belegt. Darin war zu lesen, dass „neben einem 85 m² großen Maschinensaal [...] ein Vorbereitungsraum zur Herstellung der Lochstreifen [und] drei Programmierräume“⁹⁵ eingerichtet wurden. Für letztere wurde festgehalten: „Die Arbeitsräume für die Programmierer sind modern eingerichtet und bieten sechs Personen reichlich Platz für ihre Tätigkeit.“⁹⁶ In Übereinstimmung mit der begleitenden Abbildung spricht daraus – und daran ändert die im Magazin naheliegender Weise anklingende Fortschritts- und Technikgläubigkeit nichts – diese Arbeitssituation: Es wurde mit Stift und Papier in ‚Reinräumen‘ an computerfreien Schreibtischen programmiert (Abb. 10).



10 Arbeitsraum für die Programmierer (1964)

Gleichsam war für Frieder Nake nach eigenen Aussagen das Flussdiagramm ein zentrales, wenn nicht gar ‚das‘ zentrale Mittel der Programmierung.⁹⁷ Er erklärte: „Ich habe nie ernsthaft programmiert, ohne mit einem Flussdiagramm begonnen zu haben.“⁹⁸ Auch für ihn stellte die Entwicklung des Algorithmus, weil sie mit der Problemlösung und Kreativleistung verbunden ist, die eigentliche Programmierarbeit dar; die Codierung könne im Anschluss von einem Assistenten übernommen werden oder wie Nake, der mit beiden Tätigkeiten zugange und also vertraut war, es zuspitzte: „Codierung kann jeder.“⁹⁹ Das Flussdiagramm war für ihn weniger Mittel der Kommunikation als vielmehr Hauptumschlagplatz des Entwurfs – neben Gedankenskizzen, die er auch verwendete. Nake nutzte dabei kein Lineal, sondern zeichnete seine Flussdiagramme von Hand.¹⁰⁰ So zeigt es das Beispiel des Teilprogramms ZEIPRO, das Bestandteil des seit 1964 begonnenen Programmpakets COMPART



11 Frieder Nake, Flussdiagramm zum Teilprogramm ZeiPro (Zeichenprogramm) des Programmpakets COMPART ER 56 (begonnen 1964)

Markierungen angehängt und auf diese verwiesen, Korrekturen vorgenommen oder gewisse Teile ungültig gemacht werden, wofür Nake insbesondere spezielle Symbole, verschiedene Farben und Anmerkungen einsetzte (Abb. 12). Diesbezüglich schätzte Nake die rasche und leichte Modifikation sowie – ganz im Sinne der Theorie der „Kulturtechnik der Fläche“¹⁰⁴ – die Möglichkeiten des flächigen Papierblattes: „Wo man auch nach links und rechts abzweigen und wieder an obige Stellen zurückkehren konnte.“¹⁰⁵ Dass unterschiedliche Diagrammtypen, etwa Baum- versus Kreis-Diagramm, jeweils spezifische Repräsentationsmöglichkeiten bieten, wurde in der Forschungsliteratur dargelegt.¹⁰⁶ Auf dem Papier konnte Nake auch, im Wegenetz der Anweisungen des Diagramms – eine Art gerichteter Graph(ismus) – im Zusammenspiel von Hand und Hirn, den Ablauf nachvollziehen und die an die Maschine gerichteten Operationen mental wieder ‚verflüssigen‘. Demgemäß erschöpfen sich Diagramme nach Steffen Bogen und Felix Thürlemann nicht in einem Sein von „statischen graphischen Formen“¹⁰⁷, sondern sind bezüglich ihrer Produktion und Rezeption als prozessuale „Medien des Denkens“ zu verstehen; bei ihrer Produktion können Sachverhalte verdichtet, bei ihrer Rezeption wieder entfaltet werden.¹⁰⁸ Auf dem Papier realisierte Nake Testläufe, erst danach programmiert er die Maschine.¹⁰⁹ Im Zuge dessen führte Nake das Programm aus und gelangte im günstigen Fall zu den identischen numerischen Zwischenergebnissen wie die Maschine. Zur Überprüfung des ordnungsgemäßen Ablaufs des Programms in der Maschine plante er besondere Zwischenschritte ein. So konnte er seine im Voraus selbst berechneten Werte – dabei wurde er selbst gemäß der einstigen Berufszeichnung zum „Computer“¹¹⁰ – und die Ergebnisse der Maschine auf Übereinstimmung hin ver-

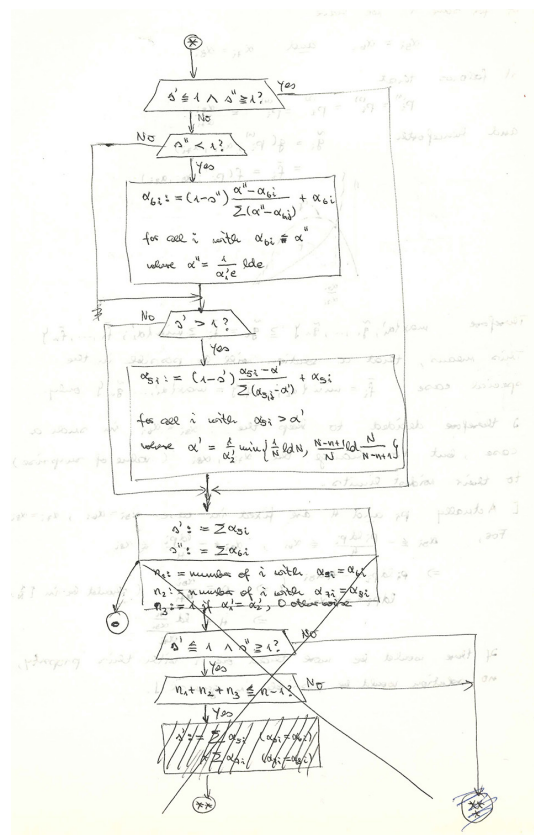
ER 56 zur Herstellung von Computergrafiken war und das alle Zeichenanweisungen für den Graphomat verarbeitete (Abb. 11).¹⁰¹ Bei der Arbeit auf dem Papier konnte mit den grafischen Elementen der Ablauf eines Programms entwickelt, ausprobiert und geprüft werden. Der Entwicklung eines Algorithmus förderlich, konnte im Speziellen der Gesamtprozess in endliche viele Einzelschritte zerlegt werden. Max Bense benannte diesen Vorgang bezüglich des „künstlerzeugenden Prozesses“¹⁰² in *Ästhetik und Programmierung* (1966) als ein Ziel seiner intendiert rationalen, auf mathematische Prinzipien basierenden, generativen Ästhetik, bei dessen Erfüllung „eine ‚definitive‘ Ästhetik“ auszumachen sei, für die gelte: „Sie führt im Idealfall zur Aufstellung von ‚Programmen‘, die der Herstellung von ‚ästhetischen Zuständen‘ mit Hilfe programmgesteuerter Rechenanlagen dienen.“¹⁰³

In der Diagrammarbeit konnten im Verbund mit den papierbasierten Medienpraktiken Schreiben und Zeichnen auf einfache Weise Ergänzungen, zum Beispiel Verbindungen über Linien hergestellt oder Teilprogramme über

gleichen.¹¹¹ Diese Vorarbeit wurde wesentlich auf dem Papier und auf der Grundlage eines Flussdiagramms durchgeführt. Wenn man annimmt, dass das Flussdiagramm für Nake wie für spätere Rezipient*innen in der Art einer „symbolischen Maschine“¹¹² fungierte, mit der er operierte und deren Regeln ihn führten, dann müsste konstatiert werden, dass es eine papierne und eine digital-technische Maschine gab, wobei die Arbeitsprozesse mit beiden verschränkt gewesen sind, weil etwa Fehler bei der Ausführung des Computers Revisionen des in der papiernen Maschine festgehaltenen Algorithmus evozierten.

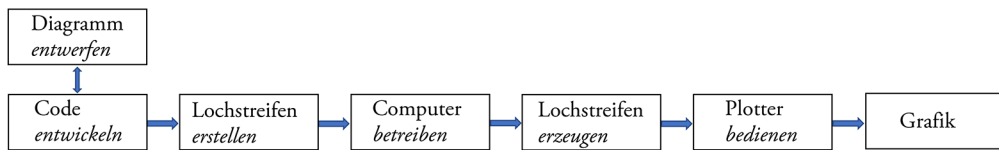
An dieser Stelle stellt sich die Frage nach dem Sein eines Computerprogramms. Wie bis hierher deutlich wurde, konnte ein und dasselbe Programm – und dies erinnert uns an die bereits erörterten Modi des digitalen Bildes – in unterschiedlichen Medialitäten existieren, etwa als Quellcode auf Papier oder als Lochstreifen aus Papier, also einem maschinenlesbaren externen Datenträger, die jeweils ineinander und insbesondere in die ‚elektronische Seinsweise‘ im Computer überführt werden können, wofür der Quellcode eingegeben oder der Lochstreifen eingelesen werden muss. Nake sprach von „Übersetzung“¹¹³, die insbesondere der Programmierer zu leisten habe. In seiner Zuführung zum internen Arbeitsspeicher wird das Programm maschinell operabel und aktualisierbar, also ausführbar. „Das Programm ist ästhetikabhängig, d. h. es gibt die ‚maschinenorientierten‘, also in die Programmiersprache bzw. in den Maschinencode übertragbaren ästhetischen Kriterien“, wie Max Bense es beschrieb und ergänzte: „Die vorhergehende Notation der maschinenorientierten ‚ästhetischen Programme‘ in einer abstrakt bzw. numerisch orientierten Ästhetik ist also evident.“¹¹⁴ An dieser Stelle wird wiederum deutlich, Diagramm und Text fungieren bei der Arbeit am Computer als Umschlagplätze, die althergebrachten Medien dienen (auf dem Papier) der Verarbeitung und Steuerung – weshalb sie als Interfaces aufgefasst werden können.¹¹⁵

Kann hinsichtlich des Flussdiagramms ebenso von einem Programm gesprochen werden? Der gängigen, oben eingeführten, auf das Kriterium der Maschinenbezogenheit fußenden Unterscheidung von Algorithmus und Programm zufolge nicht; letzteres ist demzufolge nur gemeinsam mit einer spezifischen Maschine zu denken. Nicht nur die bereits dargelegten historischen Bedingungen und Praktiken des Programmierens lassen an diesem Kriterium und also einer strikten Trennung zweifeln. Noch in den 1960er Jahren fand Programmierung verbreitet auf dem Papier und nicht an der Maschine statt, weil eine solche nicht zur Verfügung stand oder bedient werden konnte (offline statt online). Dies erinnert an die Geschichte des Computerprogramms: So wird Ada Lovelace deshalb als erste Programmiererin und ihre Vorschrift für die Berechnung von Bernoulli-Zahlen (1842/1843) als erstes Com-



12 Frieder Nake, Flussdiagramm (o. J.)

puterprogramm aufgefasst, weil sie auf dem Papier nicht bloß einen Algorithmus, sondern auch konkrete Zustände der Rechenmaschine mit Blick auf die nicht gebaute Analytical Engine von Charles Babbage festhielt (weshalb ihr Programm nur manuell ausführbar war) und derart universelle und „spezielle Maschine“ voneinander unterscheiden wusste.¹¹⁶ In den 1960er Jahren konnte ein hinreichend komplexes Programm eigentlich direkt an der Maschine erstellt werden, in der Praxis war dies jedoch nur schwer oder gar nicht möglich, weil beispielsweise sein Quellcode, etwa in Maschinencode, nicht instantan und kontinuierlich beim Eingeben interpretiert und – wie an der TU Stuttgart – auch nicht auf einem Bildschirm in einer höheren Programmiersprache dargestellt werden konnte. Ein Programm musste darum vorab, etwa mit einem Flussdiagramm, konzipiert und vorbereitet und gegebenenfalls in eine Lochkarte oder einen Lochstreifen transformiert werden. Das Programmieren im Sinne der Verfassung des Quellcodes an einer Maschine war somit nicht von der konzeptuellen Vorarbeit der Ausarbeitung des Algorithmus mithilfe eines Flussdiagramms zu trennen. Sind auch Programm und Diagramm zweifelsohne verschieden, so ist doch eine Verschränkung festzustellen: Erstes geht aus Letzterem hervor, Programmänderungen an der Maschine evozierten oftmals eine Angleichung des zugehörigen Diagramms. Wenn heutzutage die Erzeugung eines Flussdiagramms zu einem Programm automatisiert durch geeignete Software (neudeutsch „App“) geschehen kann, so mag dies die Enge des Verhältnisses unterstreichen. Die Entwicklung von Programmiersprachen, wie Algol, die maschinenunabhängig sind, konnte wiederum das Programmieren an der Maschine ein Stück weit von deren konkreter Beschaffenheit unabhängig machen. Ein weiterer und wichtiger Aspekt ist, dass sich auch die Entwicklung von Flussdiagrammen in den 1960er Jahren als letztlich maschinenbezogen darstellt, denn beim Offline-Entwurf mussten sich die Programmierer auf die Funktionalität der involvierten Maschinen einlassen. So musste etwa der anvisierte Aufbau eines Bildes, hier der Computergrafik, gemäß der spezifischen digitalen Logik des Rechners und Funktionalität des Druckers, der für das Linienzeichnen konzipiert worden war, in Bewegungsschritten und diskreten Elementen gedacht werden. Programmierung war so gesehen ein technik-empathisches Unterfangen, das mit Erfahrungswissen zusammen zu denken ist. Die Kunst des Programmierens, wie es Nake mir gegenüber eingängig formulierte, sei es: „Sich zu zwingen, so zu denken, wie die Maschine dächte, wenn sie es könnte.“¹¹⁷ Eine Denkweise, die einem ausgebildeten Mathematiker nicht fremd ist. Weil die Maschine vor allem rechnen kann, basiert auch die Computergrafik in einem erheblichen Maße auf Mathematik. So war es beispielsweise zur Wiedergabe von geraden und gekrümmte Linien auf einem Plotter notwendig, den 1962 von Jack Bresenham bei IBM entwickelten und nach ihm benannten Algorithmus zu implementieren, der die Überführung von gerundeten reellen Koordinaten in diskrete Geometrien der Raster erlaubte.¹¹⁸ Linien werden dabei zu einer Menge von Punkten. Dass auch die Arbeit mit dem Computer nicht bloß eine mathematikbasierte Technik ist, sondern ein spezifisches Denken evoziert, welches das übliche menschliche transzendiert, legte Luciana Parisi dar.¹¹⁹ Shintaro Miyazaki gab zu bedenken, dass Computerprozesse in technischer Hinsicht nicht allein durch Berechnungen (Addition), sondern auch durch nicht-rechnende Vorgänge (Verschiebung, Löschung, Übertragung et cetera.), die einem Algorithmus folgen, zu erklären sind.¹²⁰ Solche Detailbetrachtungen erinnern daran, dass die Rede vom Text-Bild-Verhältnis bezüglich des Codes, als alphanumerische Textfolge oder als ein am Bildschirm angezeigtes Programm, und der Computergrafik zu kurz greift, weil in der Maschine eine Vielzahl von Verarbeitungs- und Transformationsprozessen ablaufen, von denen Text als eine sichtbare Entität eines auf die menschliche Wahrnehmung optimierten Interfaces beziehungsweise von Hochsprachen der Programmierung fungiert. Zumal der Code als elektronisches Programm üblicherweise im Kontext einer Programmierumgebung – eines Programmier-Programms –



13 Schema eines mehrstufigen Herstellungsprozesses der Computergrafik und mit Transformationen im optimalen, nicht-rekursiven Fall

und damit eines Programms zweiter Ordnung dem User auf dem Bildschirm entgegentritt, womit er in Anlehnung an Überlegungen zur Beschaffenheit der Erscheinung und Operativität von Software-Anwendungen als diagrammatisch ausgemacht werden kann.¹²¹ Mit ähnlicher Stoßrichtung, aber theoretischer und verwickelter, will Sybille Krämer in ihren Überlegungen zur „Familienähnlichkeit“ von Programm und Diagramm, ersteres im Sinne einer „diagrammatischen Inskription“, verstehen.¹²² Ausgehend von der diskussionswürdigen Prämisse „die Medialität von Programm [...] und Programmieren deutet unweigerlich auf deren Schriftcharakter“¹²³, dass also Programme dem Bereich der Schrift angehören, wobei Schrift in einem weiten, nicht-phonographischen und notationalen Sinne verstanden wird, werden Programme mit dem Feld der „Diagrammatik“ im Sinne Krämers kurzgeschlossen, die sie als eine Praxis des operativen Graphismus versteht, die wiederum Schriften, Diagramme, Karten, Grafen et cetera umfasst. Demgemäß will sie von einem Programm gesprochen wissen, „wenn eine Handlungsfolge vor ihrer Ausführung in eine Abfolge von symbolischen Sequenzen transkribiert wird.“¹²⁴ „Eine Abfolge in der Zeit wird“, und das ist nach Krämer entscheidend für ein Programm, „in Raumrelationen übersetzt und als das simultane Strukturbild einer Handlung einer Oberfläche inskribiert.“¹²⁵ Oder wie Krämer es auch formulierte: „ein Zeitbild entsteht“¹²⁶. Nakes Diagramme zeugen jedoch davon, dass die vorliegende Komplexität Verweise auf andere Blätter (Oberflächen) erforderte und so nur beschränkt von Simultaneität und Synopsis die Rede sein kann.

Wenn bei Nake Flussdiagramm und Programm oder, wie man sagen könnte, papierne und digitale Maschine gekoppelt sind, so geht es bei ihm nicht um theoretische Familienähnlichkeiten von Programm und Diagramm, sondern um ihr reales Zusammenspiel und ihre Überführung ineinander – und dies berührt nicht zuletzt das Verhältnis von Computer und Computergrafik. Gut nachvollziehbar spricht Nake von einem Wandel vom Zeichnen mit der Hand zum „Zeichnen mit dem Kopf“, wenn der Akt des Zeichnens an den Plotter übergeben wird und sich auf das Erstellen des Programms verlagert;¹²⁷ dies setzte jedoch in einem mehrstufigen Prozess (Abb. 13) – und das ist eine weitere Pointe – eine andere grafische Praxis mit der Hand, nämlich diejenige mit Diagrammen voraus.

7. Fazit

Wie trägt dieser Aufsatz mit seiner historisch-praxeologischen Perspektive zum Verständnis des digitalen Bildes bei? Die Rekonstruktion und Analyse des historischen Schaffensprozesses von (künstlerischer) Computergrafik am Beispiel von Frieder Nake zeigte, dass Diagramm, Programm und Bild nur gemeinsam, in ihren wechselseitigen Relationen und Transformationen zu denken sind, die wiederum als Teil eines Akteur-Netzwerks zu bestimmen und als Glieder einer reversiblen Referenzkette im Sinne von Bruno Latour zu verstehen sind.¹²⁸ Das (Fluss-)Diagramm konnte als ein zentrales Mittel des Entwurfs für Programme, ja der Programmierung herausgestellt werden. So gesehen wurde auf dem Papier das zukünftige Bild entwickelt. Technischen Bildern wie der Computergrafik sind also

nicht bloß, wie Vilém Flusser für Fotografie und Computer argumentierte, ihre Maschinen eingeschrieben, weshalb diese Bilder gemäß der Logik ihrer Hervorbringung erscheinen und nicht Wirklichkeit abbilden würden oder wie Flusser einmal formulierte, ihnen „eine neue Art von Magie, nämlich die programmierte“ zu eigen ist.¹²⁹ Die diagrammatische Perspektive konnte die Auffassung von einem digitalen Bild und dem Code-Bild-Dualismus revidieren. Wenn die Computergrafik maßgeblich aus der Papierarbeit hervorgeht, so ist das digitale Bild nicht bloß wie eingangs ausgeführt hinsichtlich seiner Rezeption, etwa bei seiner Aufführung auf einem analogen Gerät, als analoge Entität zu verstehen, sondern auch, gemäß dieser Denkweise, was seine Produktion angeht. Will man die Existenz des digitalen Bildes verneinen, so müsste dies weniger in einem ontologischen Sinne, bezogen auf eine gedacht separierbare Sphäre des Digitalen, als vielmehr, vor der Folie des Post-digital-Diskurses, im Sinne einer unauflösbaren Beziehung von ‚analogen‘ und ‚digitalen‘ Techniken und Praktiken geschehen. Damit fügt sich diese Studie in solche ein, welche digitale und analoge Arbeitsweisen als verschränkt darstellen. Die praxeologische Perspektive bestätigte des Weiteren, dass etablierte Charakteristika des digitalen Bildes wie Distribution und Zirkulation und damit das digitale Bild selbst als historisch variant anzusehen sind. Gleichzeitig führte die Betrachtung der historischen Produktionssituation vor Augen, dass sich die Praxis der Programmierung seit den 1960er Jahren bis heute erheblich gewandelt hat – zuletzt hin zum ‚teach-in‘-Verfahren bei Industrierobotern, bei dem zukünftige Bewegungsabläufe des Roboterarms vorab durchgeführt und so Start-, Zwischen- und Endpositionen festgelegt wurden.¹³⁰ Der Fokus auf Handlungen ließ die vielfältigen Verschränkungen zwischen manuellen und maschinellen Prozessen erkennen, womit auch diese Studie die gängige Rede von der (maschinellen) Automatisierung im Kontext digitaler Systeme relativiert. Dies betrifft den Entwurf und die Herstellung einer Computergrafik – davon ausgehend äußerte Frieder Nake einmal: „Wenn ich so darüber nachdenke, glaube ich, mein Leben ist Entwurf.“¹³¹

Anmerkungen

Dieser Aufsatz entstand im Rahmen des SNF-Projektes ‚Automatisierte Innovationen‘, das sich Maschinenkünsten im 20. und 21. Jahrhundert und insbesondere Fragen der Automation widmet. Dem Schweizerischen Nationalfonds sei für die Förderung gedankt. Otto Ihns überließ mir das Buch *Datenverarbeitung* von Othmar Bischoff. Lysan Stemmler führte das Korrekturat durch. Ganz besonders danken möchte ich Frieder Nake für die informativen, aufschlussreichen und tiefgründigen, aber auch unterhaltsamen, kurzweiligen und nicht zuletzt humorigen Gespräche.

- 1 Claus Pias, „Das digitale Bild gibt es nicht - Über das (Nicht-)Wissen der Bilder und die informatische Illusion“, in: *Zeitenblicke* 2/1 (2003), abrufbar unter: <http://www.zeitenblicke.historicum.net/2003/01/pias/index.html> (08.05.2003).
- 2 Vilém Flusser, *Ins Universum der technischen Bilder*, 5. Aufl., Göttingen 1996 [1985], S. 53.
- 3 Vgl. etwa William J. T. Mitchell, *Das Leben der Bilder. Eine Theorie der visuellen Kultur*, 2. Aufl., München 2012 [2005], S. 164ff.
- 4 Vgl. Frieder Nake, „Informations-Ästhetik und Computergraphik“, in: *Umschau in Wissenschaft und Technik*, 6 (1968), Frankfurt am Main, S. 177f., hier S. 177.
- 5 Produziert seit 1963 im Wissenschaftskontext, wurden Nakes Grafiken erstmals im November 1965 in der Stuttgarter Galerie Wendelin Niedlich im Kunstkontext ausgestellt. Vgl. Frieder Nake, „Computer Art. A Personal Recollection“, in:

- C&C '05: Proceedings of the 5th Conference on Creativity & Cognition*, April 2005, S. 54–62, abrufbar unter: <https://doi.org/10.1145/1056224.1056234> (12.04.2005).
- 6 Zu einer umfangreichen Aufarbeitung vgl. Jens Schröter, „Analog/Digital – Opposition oder Kontinuum?“, in: *Analog/Digital – Opposition oder Kontinuum? Zur Theorie und Geschichte einer Unterscheidung*, hrsg. von Alexander Böhnke und dems., Bielefeld 2004, S. 7–30, hier insbesondere S. 10ff. Vgl. auch Michael Rottmann, *Analoge und digitale Bilder* (Theoretische Staatsexamensarbeit, Staatliche Akademie der Bildenden Künste Stuttgart), Stuttgart 2002, S. 16ff.; sowie Michael Rottmann, „Das digitale Bild als Visualisierungsstrategie der Mathematik“, in: *Verwandte Bilder: Die Fragen der Bildwissenschaft*, hrsg. von Ingeborg Reichle, Steffen Siegel und Achim Spelten, Berlin 2007, S. 281–296. Als Debattenort seien exemplarisch die mir vertrauten Lüneburger Hyperkult-Veranstaltungen genannt.
 - 7 Fragwürdig wurde insbesondere die strikte Opposition der Begriffe und ihr überwiegender Bezug auf Medientechnik. Zur Debatte vgl. etwa Florian Cramer, „What is ‚Post-Digital?‘“, in: *APRJA*, 3/1 (2014), abrufbar unter: <http://lab404.com/142/cramer.pdf> (20.10.2021).
 - 8 Vgl. etwa Shintaro Miyazaki, „Unanschauliche Datenschwärme. Eine Medienhistorische Analyse interaktiver Visualisierungen mehrdimensionaler Daten am Computer“, in: *Geschichte und Informatik, 18/19: Visualisierung von Daten in der Geschichtswissenschaft* (2015), S. 211–226, hier S. 217. Vgl. auch Rottmann 2007 (wie Anm. 6), S. 267f. bzw. Frieder Nake, *Ästhetik als Informationsverarbeitung. Grundlagen und Anwendungen der Informatik im Bereich ästhetischer Produktion und Kritik*, Wien/New York 1974, S. 106ff.
 - 9 Vgl. Schröter 2004 (wie Anm. 6), S. 14ff. bzw. Ivan Sutherland, *Sketchpad. A Man-Machine Graphical Communication System* (Dissertation Massachusetts Institute of Technology), Cambridge (MA) 1963 und Michael Noll, „The Digital Computer as a Creative Medium“, in: *IEEE Spectrum*, 4/10 (1967), S. 89–95. Vgl. auch Michael Rottmann, „Checking Creativity. Machines, Media and Mathematics in Early Computer, Serial and Conceptual Art“, in: *Proceedings Conference EVA Copenhagen 2018: Politics of the Machine – Art and After*, Aalborg Universität Kopenhagen (DK), abrufbar unter: <https://www.scienceopen.com/document?vid=ac0eb96b-4fde-42bb-b819-56704bc0f33d> (01.05.2018); Michael Rottmann, „Before Ink Starts to Blink. Scripts and Diagrams on Paper as Interfaces for Machines and Humans (in Creative Processes)“, in: *Proceedings ICLI 2018, 4th International Conference On Live Interfaces. Inspiration, Performance, Emancipation*, hrsg. von José Alberto Gomes, Miguel Carvalhais und Rui Penha, Universität Porto (PT), abrufbar unter: <http://liveinterfaces.org/2018/> (01.12.2018).
 - 10 O. A., „General Motors Research Laboratories, Pictures Spoken Here“, in: *Scientific American*, 215/3 (1966), S. 60.
 - 11 Heutzutage ist das schwer nachvollziehbar, da Bildschirme in Smartphones, Digicams, Notebooks oder Tablets als zentrales Interface integriert sind, über die digitale Bilder angefertigt und bearbeitet werden können.
 - 12 Nake in einem Gespräch mit mir am 28.07.2021 in Bremen-Borgfeld.
 - 13 Vgl. Informationsbroschüre *Elektronischer Rechenautomat ER 56*, hrsg. von Standard Elektrik Lorenz AG Stuttgart, Stuttgart-Zuffenhausen o. J., bzw. *Elektronischer Rechenautomat mit Transistoren ER 56*, hrsg. von Presse- und Werbeabteilung der Standard Elektrik Lorenz AG, 3. Aufl., Stuttgart-Zuffenhausen 1960, S. 1.
 - 14 Das Rechenzentrum der TU Stuttgart betrieb zudem seit 1958 eine Zuse Z 22 und seit 1959 eine Ferranti Pegasus sowie seit 1963 eine Telefunken TR 4. Vgl. Christoph Hoffmann, „Eine Maschine und ihr Betrieb: Zur Gründung des Recheninstituts der Technischen Hochschule Stuttgart (1956–1964)“, in: *Ästhetik als Programm: Max Bense/Daten und Streuungen, Band 5: Kaleidoskopien*, hrsg. von Barbara Büscher, dems. und Christian von Hermann, Berlin 2004, S. 118–129, hier S. 120f., 126. Bzw. Frieder Nake, „Computer-Graphik an der Universität Stuttgart“, in: *Technische Mitteilungen AEG-Telefunken, 2. Begleitheft Datenverarbeitung*, 1968, S. 1–3, hier S. 2.
 - 15 Günther Vogt, „Computer vor den Galerien. Graphik und Gedicht des Elektronengehirns in Darmstadt“, in: *Frankfurter Allgemeine Zeitung*, 09.02.1966, S. 20.
 - 16 Nake 1968a (wie Anm. 4), hier S. 177.
 - 17 In den 1980er Jahren gewann die Theoretisierung von digitaler Kultur, wie sie insbesondere Vilém Flusser vornahm, durch die wachsende Verbreitung von bildfähigen Home- bzw. Personal Computern an Dringlichkeit. Die Verdrängung des ‚Analogen‘ durch das ‚Digitale‘, die mit ‚re-mediation‘ einher geht und bis heute nicht in einer Ersetzung mündete, was technische Medien angeht, begünstigt die Reflexion, weil vorgängige Medien im Spiegel der neuen erscheinen. Vgl. auch Wolfgang Ernst, „Den A/D-Umbruch aktiv denken – medienarchäologisch, kulturtechnisch“, in: *Analog/Digital – Opposition oder Kontinuum? Zur Theorie und Geschichte einer Unterscheidung*, hrsg. von Alexander Böhnke und Jens Schröter, Bielefeld 2004, S. 49–65, hier S. 51.
 - 18 Vgl. *Was ist ein Bild?*, hrsg. von Gottfried Boehm, München 1994. Bzw. William J. T. Mitchell, *Picture Theory. Essays on Verbal and Visual Representations*, Chicago u.a. 1994.
 - 19 Vgl. Gottfried Boehm, „Vom Medium zum Bild“, in: *Bild – Medium – Kunst*, hrsg. von Yvonne Spielmann und Gundolf Winter, München 1999, S. 165–177, hier S. 175.
 - 20 Vgl. ebd.
 - 21 Ebd.
 - 22 Vgl. ebd.
 - 23 Vgl. ebd.

- 24 Vgl. Gernot Grube, „Digitale Abbildungen - ihr prekärer Zeichenstatus“, in: *Konstruierte Sichtbarkeiten. Wissenschafts- und Technikbilder seit der Frühen Neuzeit*, hrsg. von Martina Heßler, München 2006, S. 179–196, hier S. 187.
- 25 Vgl. Lev Manovich, *The Language of New Media*, Cambridge (MA)/London 2001. Bzw. Frieder Nake, „Das doppelte Bild“, in: *Bildwelten des Wissens – digitale Form*, hrsg. von Horst Bredekamp, Matthias Bruhn und Gabriele Werner, Band 3,2, Berlin 2005, S. 40–50, hier S. 47.
- 26 Vgl. Jens Schröter, „Digitales Bild“, in: *IMAGE 25* (2017), S. 89–106, hier S. 89f.
- 27 Vgl. Pias 2003 (wie Anm. 1), o. S.
- 28 Heute wird auch von Bildern gesprochen, die nicht mehr für den Menschen, sondern von Maschinen für Maschinen und Mustererkennung gemacht sind.
- 29 Vgl. Jussi Parikka, „New Materialism as Media Theory: Medianatures and Dirty Matter“, in: *Communication and Critical/Cultural Studies* 9/1 (2012), S. 95–100. Bzw. Jane Bennett, *Vibrant Matter: A Political Ecology of Things*, Durham (NC) 2009.
- 30 Gernot Grube will nicht bloß „errechnete oder mathematisch-binär codierte Bilder“ einbeziehen, sondern auch „Bilder und Grafiken, die von einem Computer prozessiert und mittels eines Ausgabemediums sichtbar gemacht werden“. Grube 2006 (wie Anm. 24), hier S. 187. Für Andreas Broeckmann ist das digitale Bild verbunden mit instabilen Prozessen auf Basis fortlaufender und permanenter Berechnungen, wohingegen das ausgedruckte Computerbild zu „analogen, angehaltenen Ergebnissen“ gehöre. Vgl. Andreas Broeckmann, „Image, Process, Performance, Machine: Aspects of an Aesthetics of the Machinic“, in: *Media Art Histories*, hrsg. von Oliver Grau, Cambridge (MA) 2010 [2007], S. 193–205, hier S. 196. Katja Kwastek zufolge ist ein Begriff ‚media art‘ problematisch, der Prozessualität als konstitutiv setzt, weil er gedruckte Computerbilder bzw. -grafiken ausschließt. Vgl. Katja Kwastek, *Aesthetics of Interaction in Digital Art*, first paperback edition, Cambridge (MA) 2015 [2013], S. 1.
- 31 Vgl. Mitchell 2012 (wie Anm. 3), S. 167.
- 32 Wie von Gernot Grube für digitale Medien angeregt. Vgl. Grube 2006 (wie Anm. 24), hier S. 18.
- 33 Ernst 2004 (wie Anm. 17), hier S. 51f.
- 34 Eine praxeologische Perspektive im Sinne Latours. Vgl. etwa Bruno Latour, *Pandora's Hope. Essays on the Realities of Science Studies*, 2. Aufl., Cambridge (MA)/London 2000.
- 35 Vgl. Sybille Krämer, „Schrift, Diagramm, Programm – Kulturtechniken der Inskription“, in: *Programm(e)*, hrsg. von Dieter Mersch und Joachim Paech, Berlin/Zürich 2014, S. 159–174, hier S. 164.
- 36 Vgl. etwa Nake 1974 (wie Anm. 8).
- 37 Was waren beispielsweise die Ziele der Computerforschung an der TU Stuttgart mit Blick auf Forschungslandschaft und Forschungspolitik? Vorgebracht wurden etwa diese Aspekte: Anwendung für andere Disziplinen vs. eigenständige Forschung, indirekte Förderung von Industrie und rechnergestützten Wissenschaften, etc. Vgl. Christian von Hermann und Christoph Hoffmann, „Der geistige Mensch und die Technik‘. Max Bense im ‚Laboratorium für Hochfrequenzphysik‘“, in: *Ästhetik als Programm: Max Bense/Daten und Streuungen, Band 5: Kaleidoskopien*, hrsg. von Barbara Büscher und dies., Berlin 2004, S. 18–31. Bzw. Christoph Hoffmann, „Eine Maschine und ihr Betrieb. Zur Gründung des Recheninstituts der TU Stuttgart“, in: *Ästhetik als Programm: Max Bense/Daten und Streuungen, Band 5: Kaleidoskopien*, hrsg. von Barbara Büscher, ders. und Christian von Hermann, Berlin 2004, S. 118–129.
- 38 Vgl. Frieder Nake, „Statement for Page“, in: *Page. Bulletin of the Computer Arts Society* 8 (1970), S. 2.
- 39 Etwa anlässlich der online-Konferenz ‚The Digital Image – Social Dimensions, Political Perspectives and Economic Constraints‘ an der Ludwig-Maximilians-Universität München (28.-30.04.2021).
- 40 Vogt 1966 (wie Anm. 15), S. 20.
- 41 Vgl. Stichwort „Programmierung“, in: *Duden Informatik. Ein Sachlexikon für Studium und Praxis*, 2. vollständig überarb. und erw. Aufl., Mannheim u.a. 1993.
- 42 Vgl. Hartmut Winkler, „2. Einführung: Sektion: ‚Was ist Programmieren?‘“, in: *Programm(e)*, hrsg. von Dieter Mersch und Joachim Paech, Berlin/Zürich 2014, S. 121–123, hier S. 122.
- 43 Vgl. Nathalie Casemajor, „Digital Materialisms: Frameworks for Digital Media Studies“, in: *Westminster Papers in Culture and Communication* 10/1 (2015), S. 4–17, hier S. 6, abrufbar unter: <http://dx.doi.org/10.16997/wpcc.209> (20.10.2021).
- 44 Vgl. Schröter 2004 (wie Anm. 6), hier S. 22.
- 45 Vgl. Florian Cramer, „Concepts, Notations, Software, Art“, 2002, abrufbar unter: http://cramer.pleintekst.nl/all/concept_notations_software_art/concepts_notations_software_art.html (01.09.2021) [Übersetzung des Autors].
- 46 Vgl. Winkler 2014 (wie Anm. 42), hier S. 121.
- 47 Neben der Tabelle und der Matrix. Vgl. Dieter Mersch, „Pro-Grammata. Einige Überlegungen zu einer Theorie der Programme“, in: *Programm(e)*, hrsg. von ders. und Joachim Paech, Berlin/Zürich 2014, S. 461–486, hier S. 474.
- 48 Vgl. Krämer 2014 (wie Anm. 35), S. 159–174.
- 49 Vgl. dazu auch Boris Ewenstein und Jennifer Whyte, „Wissenspraktiken im Design“, in: *entwerfen – wissen – produzieren. Designforschung im Anwendungskontext*, hrsg. von Claudia Mareis, Gesche Joost und Kora Kimpel, Bielefeld 2010, S. 47–80.
- 50 Vgl. Vogt 1966 (wie Anm. 15), S. 20.

- 51 Vgl. Frieder Nake, „Bemerkungen zur Herstellung von Computer-Grafiken“, in: *Herstellung von zeichnerischen Darstellungen, Tonfolgen mit elektronischen Rechenanlagen*, Programm-Information, hrsg. von Deutsches Rechenzentrum Darmstadt, Darmstadt 1966, S. 3–34, hier S. 3.
- 52 Vgl. Nathan Enslinger, *The Computer Boys Take Over. Computers, Programmers, and the Politics of Technical Expertise*, Cambridge (MA)/London 2010, S. 30.
- 53 Vgl. ebd., S. 29.
- 54 Vgl. Edsger Dijkstra, „Some Meditations on Advanced Programming“, in: *Proceedings of IFIP Congress*, hrsg. von C. Poplewell, Amsterdam 1963, S. 535–538, hier S. 535.
- 55 In den 1960er Jahren wurde der Begriff ‚Hacking‘ am MIT auf das Programmieren bezogen. Vgl. Steven Levy, *Hackers: Heroes of the Computer Revolution*, Garden City (NY) 1984.
- 56 Vgl. Enslinger 2010 (wie Anm. 52), S. 29 [Übersetzung des Autors].
- 57 Vgl. Edsger Dijkstra, „The Humble Programmer“, in: *Communications of the ACM*, 5/10 (1972), S. 859–866, hier S. 861.
- 58 Vgl. Nake 1968b (wie Anm. 14), hier S. 1.
- 59 Vgl. Frieder Nake, *Three Drawings and One Story*, März/Oktober 2020, abrufbar unter: https://dam.org/museum/essays_ui/essays/three-drawings-and-one-story/ (25.01.2021).
- 60 Vgl. Claudia Mareis und Michael Rottmann, *Entwerfen mit System* (Studienhefte Problemorientiertes Design 10), hrsg. von Jesko Fezer, Oliver Gemballa und Matthias Görlich, Hamburg 2020. Bzw. Michael Rottmann, *Gestaltete Mathematik. Geometrien, Zahlen und Diagramme in der Kunst in New York um 1960. Mel Bochner – Donald Judd – Sol LeWitt – Ruth Vollmer*, München 2020, S. 183ff.
- 61 Vgl. Friedrich Kitzler, „Synergie von Mensch und Maschine“, in: *Kunst machen. Gespräche und Essays*, hrsg. von Florian Rötzer und Sara Rogenhofer, Leipzig 1993 [1989], S. 83–102, hier S. 94.
- 62 Nake glaubte sich daran zu erinnern, dass die Frage ‚Was ist Intelligenz?‘ im Zusammenhang mit Wahrscheinlichkeitstheorie diskutiert wurde und gab zu bedenken, dass es noch keine Neurowissenschaften gab. Einschlägige Forschungsbestrebungen an der TU Stuttgart waren ihm nicht bekannt. Nake im fernmündlichen Gespräch mit mir am 29.10.2020.
- 63 Vgl. Werbebroschüre: *ZUSE Z64 Graphomat*, hrsg. von Konrad Zuse KG Bad Hersfeld, o. J., o. S.
- 64 Vgl. ebd. bzw. Nake 2020 (wie Anm. 59), S. 4f.
- 65 Vgl. SEL 1960 (wie Anm. 13), S. 22, 27.
- 66 Vgl. Zuse KG o. J. (wie Anm. 63), o. S.
- 67 Frieder Nake, „Teamwork zwischen Künstler und Computer“, in: *Format. Zeitschrift für visuelle Kommunikation* 3/11 (1967), S. 38f. Hier zit. nach: *Ästhetik als Programm: Max Bense/Daten und Streuungen, Band 5: Kaleidoskopien*, hrsg. von Barbara Büscher, Christoph Hoffmann und Christian von Hermann, Berlin 2004, S. 220–225, hier S. 221.
- 68 Nake im fernmündlichen Gespräch mit mir am 10.10.2020.
- 69 Vgl. Mareis/Rottmann 2020 (wie Anm. 60) und Rottmann 2020 (wie Anm. 60).
- 70 Nake im persönlichen Gespräch am 28.07.2021 in Bremen-Borgfeld.
- 71 Vgl. Konrad Zuse, *Ein Verfahren zur Konstruktion von grafischen Kompositionen mit Hilfe von Rechen- und Zeichenautomaten*, Manuskript, 1965, S. 2.
- 72 Nake im fernmündlichen Gespräch mit mir am 29.10.2020.
- 73 Frieder Nake, „Computer-Grafik“, in: *Exakte Ästhetik. Kunst aus dem Computer* 5 (1967), S. 21–33, hier S. 29.
- 74 Nake im fernmündlichen Gespräch mit mir am 29.10.2020.
- 75 Vgl. Nake 2020 (wie Anm. 59), S. 6.
- 76 Nake 1967b (wie Anm. 73), hier S. 32.
- 77 Vgl. Grube 2006 (wie Anm. 24), hier S. 186.
- 78 Vgl. Pias 2003 (wie Anm. 1), o. S.
- 79 Vgl. Grube 2006 (wie Anm. 24), hier S. 186.
- 80 Vgl. Manovich 2001 (wie Anm. 25), S. 36ff. [Übersetzung des Autors]
- 81 Zeichnungen können demselben Programm entstammen, sich jedoch durch variierende Bestimmungsgrößen wie Parameter oder Pseudo-Zufall unterscheiden. Vgl. Nake 2020 (wie Anm. 59), S. 5f.
- 82 Vgl. Nake 2005 (wie Anm. 25), hier S. 50.
- 83 Nake im persönlichen Gespräch am 28.07.2021 in Bremen-Borgfeld.
- 84 Nake im persönlichen Gespräch am 28.07.2021 in Bremen-Borgfeld.
- 85 Vgl. Nake 2020 (wie Anm. 59), S. 5f.
- 86 Vgl. Michael Rottmann, „Kalkulierte Innovationen. Zur Kritik der Systematisierung von Entwurfs- und Innovationsprozessen in der Kunst um 1960“, in: Mareis/Rottmann 2020 (wie Anm. 60), S. 123–221.
- 87 Vogt 1966 (wie Anm. 15), S. 20.
- 88 Nutzbar wäre auch das Lochkarten-Eingabewerk 400 gewesen.
- 89 Nake im fernmündlichen Gespräch mit mir am 10.10.2020.

- 90 Es ist auch von Programmablaufplan (PAP) die Rede. Vgl. Hans Westermayer, *Programmierlogik, Programmablaufpläne*, München 1971.
- 91 Vgl. Othmar Bischoff, *Datenverarbeitung. Informationen, Daten und ihre Verarbeitung im kaufmännischen Büro*, Bad Homburg/Berlin/Zürich 1974.
- 92 Vgl. Ensmenger 2010 (wie Anm. 52), S. 36ff.
- 93 Vgl. ebd., S. 39.
- 94 Vgl. Stichwort „Algorithmus“, in: *Duden Informatik. Ein Sachlexikon für Studium und Praxis*, 2. vollständig überarb. und erw. Aufl., Mannheim u.a. 1993.
- 95 Dietrich Stahnke, „Das technische Rechenzentrum Rheinstahl-Wahner“, in: *ZUSE Forum* 8 (1964), S. 4–8, hier S. 5.
- 96 Ebd.
- 97 Naked in einem fernmündlichen Gespräch mit mir am 29.10.2020.
- 98 Naked im persönlichen Gespräch am 28.07.2021 in Bremen-Borgfeld.
- 99 Naked im persönlichen Gespräch am 28.07.2021 in Bremen-Borgfeld.
- 100 Naked in einem fernmündlichen Gespräch mit mir am 29.10.2020. Am 28.07.2021 führte er es mir eindrucksvoll vor. Die Diagramme in *Ästhetik als Informationsverarbeitung* (1974), darunter dasjenige für die Grafik *Hommage à Paul Klee*, sind von John McIntosh mit Lineal gezeichnet. Vgl. Naked 1974 (wie Anm. 8), S. VII.
- 101 Vgl. ebd., S. 191ff.
- 102 Vgl. Max Bense, „Ästhetik und Programmierung“, in: *IBM Nachrichten*, 16/180 (1966), S. 294–296. Hier zitiert nach: *Ästhetik als Programm: Max Bense/Daten und Streuungen, Band 5: Kaleidoskopien*, hrsg. von Barbara Büscher, Christoph Hoffmann und Christian von Hermann, Berlin 2004, S. 209–213, hier S. 211.
- 103 Vgl. ebd.
- 104 Vgl. Sybille Krämer, *Figuration, Anschauung, Erkenntnis: Grundlinien einer Diagrammatologie*, Berlin 2016, S. 11ff.
- 105 Naked in einem fernmündlichen Gespräch mit mir vom 29.10.2020.
- 106 Weiterführend vgl. Rottmann 2018b (wie Anm. 9), hier S. 103.
- 107 Vgl. Steffen Bogen und Felix Thürlemann, „Jenseits der Opposition von Text und Bild. Überlegungen zu einer Theorie des Diagramms und des Diagrammatischen“, in: *Die Bildwelt der Diagramme. Joachims von Fiore: Zur Medialität religiös-politischer Programme im Mittelalter*, hrsg. von Alexander Patschovsky, Ostfildern 2003, S. 1–22, hier S. 10.
- 108 Vgl. ebd., hier S. 8.
- 109 Naked im fernmündlichen Gespräch mit mir am 10.10.2020.
- 110 In den Anfängen arbeiteten zuvorderst Frauen als „Computer“. Ensmenger 2010 (wie Anm. 52), S. 35f.
- 111 Naked in einem fernmündlichen Gespräch mit mir am 10.10.2020.
- 112 Vgl. Sybille Krämer, *Symbolische Maschinen. Die Idee der Formalisierung in geschichtlichem Abriss*, Darmstadt 1988, S. 3.
- 113 Naked 1967a (wie Anm. 67), hier S. 221f.
- 114 Vgl. Bense 1966 (wie Anm. 102), hier S. 212.
- 115 Zu Schrift und Diagramm auf Papier als Interface vgl. Rottmann 2018b (wie Anm. 9).
- 116 Vgl. Sybille Krämer, „Wieso gilt Ada Lovelace als ‚erste Programmiererin‘ und was bedeutet überhaupt ‚programmieren‘?“, in: *Ada Lovelace. Die Pionierin der Computertechnik und ihre Nachfolgerinnen*, hrsg. von ders., Paderborn 2015, S. 75–90, insbesondere S. 88.
- 117 Naked in einem Gespräch mit mir am 28.07.2021 in Bremen-Borgfeld.
- 118 Jack E. Bresenham, „Algorithm for Computer Control of a Digital Plotter“, in: *IBM Systems Journal* 4/1 (1965), S. 25–30.
- 119 Vgl. Luciana Parisi, *Contagious Architecture. Computation, Aesthetics, and Space*, Cambridge (MA)/London 2013.
- 120 Vgl. Shintaro Miyazaki, unveröffentlichtes Skript zur Vorlesung „Kritik der reinen Komputation“, Wintersemester 2020/21, Humboldt-Universität zu Berlin. Bzw. Shintaro Miyazaki, *Algorhythmisiert. Eine Medienarchäologie digitaler Signale und (un)erhörter Zeiteffekte*, Berlin 2013, S. 125.
- 121 Vgl. Bogen/Thürlemann 2003 (wie Anm. 107), hier S. 4f.
- 122 Vgl. Krämer 2014 (wie Anm. 35), hier S. 159.
- 123 Vgl. ebd., hier S. 163f. Obwohl Krämer die Etymologie und damit das „Programm“ und „Diagramm“ gemeinsame „gramma“ („Buchstabe“) als ein Indiz für den Schriftcharakter heranzieht (S. 168) wären für ihre Argumentation weitere historische Differenzierungen von Programm und Programmierung wünschenswert gewesen. Zwar besitzen Computerprogramme zweifelsohne einen Schriftbezug, insbesondere mit Blick auf Krämers weitgefassten Schriftbegriff; sie können in Schrift überführt werden, doch programmiert wurde auch, wie etwa der ENIAC der 1940er Jahre, mit Knöpfen und Stöpseln. Vgl. Thomas Haigh, Mark Priestley, Crispin Rope, *ENIAC in Action. Making and Remaking the Modern Computer*, London/Cambridge (MA) 2016.
- 124 Vgl. ebd., hier S. 171.
- 125 Ebd.
- 126 Ebd.

- 127 Es wurde auch ein Zeichnen bzw. Zeigen mit dem Lichtgriffel möglich. Vgl. Frieder Nake, „Zeigen, Zeichnen und Zeichnen. Der verschwundene Lichtgriffel“, in: *Mensch-Computer-Interface. Zur Geschichte und Zukunft der Computerbedienung*, hrsg. von Hans Dieter Hellige, Bielefeld 2008, S. 121–154, hier insbesondere S. 147.
- 128 Vgl. Latour 2000 (wie Anm. 34).
- 129 Vgl. Vilém Flusser, *Für eine Philosophie der Fotografie*, 9. Aufl., Göttingen 1999 [1983], S. 18 bzw. S. 14, 24. Bzw. Flusser 1996 (wie Anm. 2), S. 53.
- 130 Vgl. Stefan Hesse, *Industrieroboterpraxis: Automatisierte Handhabung in der Fertigung*, Braunschweig/Wiesbaden 1998.
- 131 Nake im persönlichen Gespräch am 28.07.2021 in Bremen-Borgfeld.

Bildnachweise

- Abb. 1: Frieder Nake, *13/9/65 Nr. 2*, „Hommage à Paul Klee“ (1965), Computergrafik, Zuse Z 64 Graphomat, schwarze Tinte auf Papier, 40 × 40 cm [1966 auch als *Klee*, mittlerweile kurz als *Hommage à Paul Klee* bezeichnet]: mit freundlicher Genehmigung von Frieder Nake.
- Abb. 2: Anlage des SEL ER 56 (undatiert, um 1961), in: Informationsbroschüre *Elektronischer Rechenautomat ER 56*, hrsg. von Standard Elektrik Lorenz AG Stuttgart, Stuttgart-Zuffenhausen o. J., S. 2.
- Abb. 3: Konsole des Großrechners SEL ER 56 an der Technischen Universität Stuttgart vor der Trennscheibe des gekühlten Rechnerraums (um 1965), in: *Ästhetik als Programm: Max Bense/Daten und Streuungen, Band 5: Kaleidoskopien*, hrsg. von Barbara Büscher, Christoph Hoffmann und Christian von Hermann, Berlin 2004, S. 183.
- Abb. 4: ZUSE Z 64 Graphomat, Anlage an der Technischen Universität Stuttgart (vermutlich 1965, nicht später als 1966): mit freundlicher Genehmigung von Frieder Nake.
- Abb. 5: ZUSE Z 64 Graphomat in einer kompakteren Version, ein Teil des Lochstreifens liegt auf dem Boden (rechts unten): Public domain, https://de.wikipedia.org/wiki/Datei:Graphomat_Zuse_Z64.jpg (01.10.2021).
- Abb. 6: Frieder Nake, *Kreis-Variationen, Nr. 3* (04.05.1965), Computergrafik mit Zuse Z 64 Graphomat, schwarze Tinte auf Papier, 25 × 25 cm: mit freundlicher Genehmigung von Frieder Nake.
- Abb. 7: Frieder Nake, Code des Teilprogramms ZeiPro (Zeichenprogramm) des Programmpakets COMPART ER 56 (begonnen 1964): mit freundlicher Genehmigung von Frieder Nake.
- Abb. 8: Schablone aus Kunststoff für Flussdiagramme, Zubehör zum SEL 9829: © Raimond Spekking / CC BY-SA 4.0 (via Wikimedia Commons), Flow Chart Stencil SEL-9829, https://commons.wikimedia.org/wiki/File:Flow_Chart_Stencil_SEL-9829.jpg (01.10.2021).
- Abb. 9: Frieder Nake, Flussdiagramm zu *Hommage à Paul Klee*, 1974 (1965), in: Frieder Nake, *Ästhetik als Informationsverarbeitung. Grundlagen und Anwendungen der Informatik im Bereich ästhetischer Produktion und Kritik*, Wien/New York 1974, S. 216, mit freundlicher Genehmigung von Frieder Nake.
- Abb. 10: Arbeitsraum für die Programmierer (1964), in: Dietrich Stahnke, „Das technische Rechenzentrum Rheinstahl-Wahner“, in: *ZUSE Forum* 8 (1964), S. 4–8, hier S. 4.
- Abb. 11: Frieder Nake, Flussdiagramm zum Teilprogramms ZeiPro (Zeichenprogramm) des Programmpakets COMPART ER 56 (begonnen 1964): mit freundlicher Genehmigung von Frieder Nake.
- Abb. 12: Frieder Nake, Flussdiagramm (o. J.): Mit freundlicher Genehmigung von Frieder Nake.
- Abb. 13: Schema eines mehrstufigen Herstellungsprozesses der Computergrafik und mit Transformationen im optimalen, nicht-rekursiven Fall: Grafik von Michael Rottmann.

Dieser Beitrag ist auch unter folgender Internetadresse abrufbar:
<https://www.kunstgeschichte-ejournal.net/589/>