Analog Computing

von

Prof. Dr. Bernd Ulmann

Oldenbourg Verlag München

Dr. Bernd Ulmann is a professor for business informatics at the FOM in Frankfurt/Main, Germany. His main area of interest is that of analog computing. He not only collects and restores old analog computers, but also works actively on future applications of analog computing for high performance computing and low power applications.

Bibliografische Information der Deutschen Nationalbibliothek

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über http://dnb.d-nb.de abrufbar.

Rosenheimer Straße 143, D-81671 München
Telefon: (089) 45051-0
www.oldenbourg-verlag.de

Lektorat: Dr. Gerhard Pappert
Herstellung: Tina Bonertz
Titelbild: Autor
Einbandgestaltung: hauser lacour
Gesamtherstellung: Beltz Bad Langensalza GmbH, Bad Langensalza

Dieses Papier ist alterungsbeständig nach DIN/ISO 9706.

ISBN 978-3-486-72897-2
eISBN 978-3-486-75518-3

To my beloved wife Rikka.

Acknowledgments

This book would not have been possible without the support and help of many people. First of all, I would like to thank my wife Rikka Mitsam who not only did a lot of proofreading and a terrific job in preparing many of the pen-and-ink drawings but also never complained about being neglected although I lived the last months more or less in seclusion writing this book.

I am particularly grateful for the support and help of Jens Breitenbach who did a magnificent job at proof reading and provided many suggestions and improvements enhancing the text significantly. He also suggested to use the Kp-fonts and spotted many LaTeX-sins of mine thus improving the overall appearance of this book considerably.

In addition to that, I would like to thank Benjamin Barnickel, Arne Charlet, Daniela Koch, Theresa Szczepanski and Dr. Reinhard Steffens for their invaluable help in proof reading.

Additionally, I would like to thank Tore Sinding Bekkedal who took the picture shown in figure 8.32, Tim Robinson who built the incredible Meccano-Differential Analyzer shown in figure 2.19, Tibor Florestan Pluto who took the photo shown on the title page, Robert Limes who took the picture shown in figure 9.3, and Bruce Baker who donated the pictures shown in figures 11.41, 11.43, and 11.44 for their permissions to use the aforementioned pictures.

Without the continuous encouragement of Dr. habil. Karl Schlagenhauf and his invaluable suggestions this book might very well not exist at all.

Last but not least, I would like to thank Prof. Dr. Wolfgang Giloi, Prof. Dr. Rudolf Lauber, Dr. Adolf Kley, and Prof. Dr. Günter Meyer-Brötz for sharing their memories of the early days of analog computing at Telefunken etc.

This book was typeset with LaTeX, using Kp-Fonts; most schematics were drawn using EAGLE and Arno Jacob's wonderful analog computing symbol library, other vector graphics were created with `xfig`.

Contents

"An analog computer is a thing of beauty and a joy forever."[1]

[1] John H. McLeod, Suzette McLeod, "The Simulation Council Newsletter", in *Instruments and Automation*, Vol. 31, March 1958, p. 488.

1 Introduction

1.1 Outline

A book on analog computing and analog computers? You might ask: Isn't that 40 years late? No, it isn't – although the beautiful analog computers of the past are long since history and only few have been preserved in museum collections, the idea of analog computing is still a marvel of elegance and may have a bright and fruitful future as the following chapters will show.

Thus the intention of this book is twofold: It will give a comprehensive description of the history and technology of analog computing but it will also show the particular strengths of the analog computation paradigm which, combined with current state of the art digital circuitry, may open new doors to high performance computing again just as traditional analog computers did many years ago in the area of high-speed simulation and many other fields.

The following chapters will first introduce the notion of analog computing before describing the early developments of analog computers starting with mechanical analog computers like the Antikythera mechanism, which was built around 100 B.C. and ending with the first analog electronic[1] analog computers developed by Helmut Hoelzer in Germany and George A. Philbrick in the United States.

Next, the basic elements of a typical analog computer will be described, followed by an introduction to programming analog computers, which is completely different from the algorithmic approach we are used to in our times. This introduction contains some practical examples ranging from the solution of very simple differential equations to the simulation of more complex dynamic systems.

Hybrid computers, i. e., analog computers coupled with stored-program digital computers, are also covered, as well as the simulation of analog computers. After describing some typical analog computers in detail, digital differential analyzers are introduced. This is followed by a large section covering a wide span of practical applications of analog computers. The end of this book covers the decline and future of analog computing that is the heritage of the long forgotten digital differential analyzers.

[1] The notion of an *analog electronic analog computer* may look like a pleonasm, which it is not as the following section will show.

1.2 The notion of analog computing

First of all it should be noted that the common misconception that the difference between *digital computers* on one side and *analog computers* on the other is the fact that the former use discrete values for computations while the latter work in the regime of continuous values is wrong![2] In fact there were and still are analog computers that are based on purely digital elements. In addition to that even analog electronic analog computers are not working on continuous values – eventually everything like the integration of a current boils down to storing (i. e., counting) quantized electrons in a capacitor.

If the type of values used in a computation – discrete versus continuous – is not the distinguishing feature, what else could be used to differentiate between *digital* and *analog* computers? It turns out that the difference is to be found in the structure of these two classes of machines: A digital computer in our modern sense of the word has a fixed structure concerning its constituent elements and solves problems by executing a sequence (or sequences) of instructions that implement an algorithm. These instructions are read from some kind of memory, thus a better term for this kind of computing machine would be *stored-program digital computer* since this describes both features of such a machine: Its ability to execute instructions fetched from a memory subsystem and working with numbers that are represented as streams of digits.[3]

An analog computer on the other hand is based on a completely different paradigm: Its internal structure is not fixed – in fact, a problem is solved on such a machine by changing its structure in a suitable way to generate a *model*, a so-called *analog* of the problem.[4] This analog is then used to *analyze* or *simulate* the problem to be solved. Thus the structure of an analog computer that has been set up to tackle a specific problem represents the problem itself while a stored-program digital computer keeps its structure and only its controlling program changes. This is summarized by Charlesworth[5] as follows:

> *"An analogue computer is a piece of equipment whose component parts can be arranged to satisfy a given set of equations, usually simultaneous ordinary differential equations."*

Similarly [Berkeley et al. 1956][p. 75] states that

> *"[a]nalog computers, as the name is intended to imply, compute by means of setups that are analog of the problems to be solved."*

Therefore it is not only a theoretical possibility to build digital analog computers, this has been done more often than just occasionally.[6] In fact this may very well be the

[2] This error has even found its way into some encyclopedias.

[3] Today these numbers are normally represented by so-called *binary digits*, *bits* for short.

[4] [Tse et al. 1964][p. 333] characterized these analogs or analogies as follows: *"The term 'analogy' is defined to mean similarity of relation without identity."*

[5] See [Charlesworth et al. 1974][p. xi].

[6] Cf. sections 10 and 11.15.7.

	Basic technology	
	Analog electronic	Digital electronic
Stored-program control	N/A	stored-program (memory programmed) digital computer
Setting up an analog	traditional analog electronic analog computer	digital differential analyzer

Table 1.1: Types of computing machines based on an analog/digital electronic implementation with control based on either a stored-program concept or the implementation of an analog.

future of the analog computing paradigm. Employing the techniques of building models, analogs of problems to be solved or analyzed, which have been developed through more than 50 years in the context of our current digital technology can and will lead to systems with exceptional computational power as well as low power consumption.

Table 1.1 shows the four possible combinations of analog/digital implementation technology and stored-program control vs. setting up an analog. Of these combinations only three are reasonable:

1. The modern stored-program digital computer,
2. the traditional analog electronic analog computer which will be called *analog computer* for simplicity in the following text and, finally,
3. the so-called *Digital Differential Analyzer*[7] that will be described in more detail in chapter 10.[8]

1.3 Direct and indirect analogies

When talking about analogies, it is necessary to distinguish between so-called *direct* and *indirect* analogies. These two terms describe two extremes of abstraction levels in building analogies.

Direct analogies are models that are based on the same physical principles as the underlying problem to be solved just with a different scaling regarding size or time of a simulation. Well-known examples for such direct analogies are the determination of minimal surfaces using soap films,[9] the evaluation of tensile structures as they are

[7] *DDA* for short.

[8] It will be shown that DDAs are more capable machines than traditional analog computers since they can deal well with the highly important class of partial differential equations which analog computers can do only with quite some effort and often only by means of discretization.

[9] See [Bild der Wissenschaft 1970] for examples.

used for roof structures like the one built for the Olympic stadium in Munich,[10] wind tunnel models for the evaluation of aerodynamic properties of airplanes and rockets and many more. Other direct analogs employed so-called *electrolytic tanks*. These are reservoirs filled with an electrolytic liquid with embedded electrodes to generate a desired potential distribution within the liquid. Using a two- or three-dimensional sensor carriage, much like an xy-recorder, the potential at any given coordinate within the tank can be determined. Such electrolytic tanks were widely used to solve problems in nuclear engineering etc. Due to their very nature such direct analogs are not too versatile and were often built and employed for a very specific purpose.[11]

Indirect analogies in contrast to that employ a much higher degree of abstraction and are thus much more versatile. The – mostly analog electronic – analog computers used for setting up indirect analogs are truly universal machines and cover a wide range of possible applications.[12] This higher level of abstraction makes the programming of this class of machines quite challenging since their setup does not bear any direct resemblance of the problem to be solved. Therefore a thorough mathematical description of the basic problem is required as a precondition for programming an indirect analog computer[13] like in the case of our modern stored-program digital computers.

Nevertheless, this kind of abstraction level necessary for the successful application of analog computers is still relatively small compared with the algorithmic approach of stored-program digital computers – last but not least, analog computers, be they direct or indirect, are models.

Due to the fact that analog computers work by acting as a model for a given problem that is represented by direct or indirect means, the amount of circuitry necessary for a simulation is determined by the complexity of the underlying problem. There is no tradeoff between time and complexity as that offered by stored-program digital computers. An analog computer with a given complement of computing elements is invariably limited by this while a digital computer can solve problems of nearly any size and complexity given enough time (and storage). Thus large problems require large analog computers regardless of the acceptable time span for yielding a solution – some systems, especially those used for aerospace applications, exceeded well over 1000 computing elements.

In addition to that a stored-program computer can always exchange compute time for precision – something an analog computer cannot do.[14]

[10] The roof structure of the Olympic stadium in Munich was modeled to a large extent using curtain net lace as well as soap bubbles for determining the structure of single roof tiles. Even Antoni Gaudí used such direct analogs in his design studies for the church of Colònia Güell and later the Sagrada Familia. Similar techniques date back to the Gothic period.

[11] More detailed information about this basic class of analogs can be found in [Jackson 1960][pp. 319 ff.], [Paschkis et al. 1968], [Master et al. 1955], [Larrowe 1955] and [Karplus 1958].

[12] One of the earliest publication about the use of electronic analogs to simulate mechanical and acoustical systems was given by [Olson 1943].

[13] Direct analogs can also be employed in cases where no complete mathematical description of the problem to be solved exists – this may be caused by a principle lack of understanding or by the sheer complexity of the underlying problem. So in some cases direct analogs may even be employed today with success.

[14] This does not hold true for digital differential analyzers, cf. section 10.

2 Mechanical analog computers

The earliest analog computers were mechanical in their very nature but were far from being simple. In fact many mechanical analog computers were successfully employed to tackle rather complex problems ranging from peaceful tide computations to wartime applications like bomb trajectories, fire control, etc. The following sections give a short overview of the era of mechanical analog computers without going too much into detail since mechanical analogs will serve just as a prelude to this book's main theme of electronic analog computers.

2.1 Astrolabes

As early as about 150 B.C. the basics of so-called *astrolabes* were developed. Such devices are basically inclinometers with some additional mechanics to model basic properties of spherical astronomy. Astrolabes are based on the apparent motion of celestial bodies, i. e., the observation that the paths described by stars in the sky are basically circles. Thus the most common type of astrolabe is the so-called *planispheric astrolabe*, developed in medieval times, which projects the firmament to the equatorial plane.

Using such an instrument it is possible to determine the position of some celestial bodies at a given time. As a navigational tool the planispheric astrolabe is far too imprecise. Nevertheless, it has been used to determine roughly the position of stars used for getting navigational fixes in advance. Detailed information about astrolabes can be found in [Dodd 1969] and [J. E. Morrison 2007].

2.2 The Antikythera mechanism

Interestingly, another early mechanical analog computer known is far from being a simple device – in fact this machine should turn out to be the most complex device made in more than 1000 years of history. This impressive machine is the so-called *Antikythera mechanism*, named for the Greek island Antikythera (Αντικύθηρα) where its remains were discovered in a Roman ship wreck carrying treasures from Greece dating back to about 100 B.C. by sponge-divers in 1900. It then took 58 years until research regarding the remains of this mechanism was started by Derek de Solla Price who summarized his astonishing discoveries as follows:[1]

> *"It is a bit frightening to know that just before the fall of their great civilization the ancient Greeks had come so close to our age, not only in their thought, but also in their scientific technology."*

[1] See [Freeth 2008][p. 7].

The Antikythera mechanism turns out to be of such a complexity that even today research is still ongoing,[2] using modern X-ray tomography techniques etc. As far as it is currently known the device modeled the movements of several celestial bodies even taking into account various anomalies which required differential gears[3] and much more complicated epicyclic gearing. This astonishing complexity of the Antikythera mechanism led Mike Edmunds[4] to the following statement:

> *"Nothing as sophisticated and complex is known for another thousand years. This machine rewrites the history of technology. It is a witness to a revolution in human thought."*

This intricate mechanism allowed the calculation of sun and moon positions at given dates and phases of the moon as well as the prediction of solar and lunar eclipses.[5] The implementation of these functions required about 30 gears, manufactured with extraordinary precision.[6]

2.3 Slide rules

One of the most common, most simple and well-known analog computers is the *slide rule*[7] which comes in basically two configurations, *linear* and *circular*. The basic idea of a slide rule is to reduce the problem of multiplication and division to that of addition and subtraction by employing logarithmically divided scales that may be displaced accordingly to each other in a lateral direction. The analog setup in this case is to mechanize the relation

$$\log(ab) = \log(a) + \log(b)$$

by using two logarithmic scales. Following the development of the logarithm by John Napier[8] and Henry Briggs,[9] who introduced the base 10 for logarithms, it was William Oughtred[10] who described the principle of the slide rule in his seminal two publications *The Circles of Proportion, and the Horizontall Instrument* and *Two rulers of proportion*.[11] Prior to his idea of using sliding scales, dividers were normally used to multiply or divide numbers by adding or subtracting their logarithms, a tedious and error-prone process.

[2] Cf. `http://www.antikythera-mechanism.gr/`.

[3] Prior to this discovery differential gears were thought to have been invented in medieval times.

[4] See [Freeth 2008][p. 9].

[5] There are still arguments whether the mechanism also featured indicators for the display of planet positions.

[6] A wealth of information about this device may be found in [de Solla Price 1974], [Freeth 2008] and [McCarthy 2009].

[7] Also known as a *slipstick*.

[8] 1550–04/03/1617

[9] February 1561–01/26/1630

[10] 03/05/1574–06/30/1660

[11] A comprehensive history of the slide rule may be found in [Cajori 1994] and [Jezierski 2000]. A great introduction to the application and use of slide rules is given in [Hume et al. 2005].

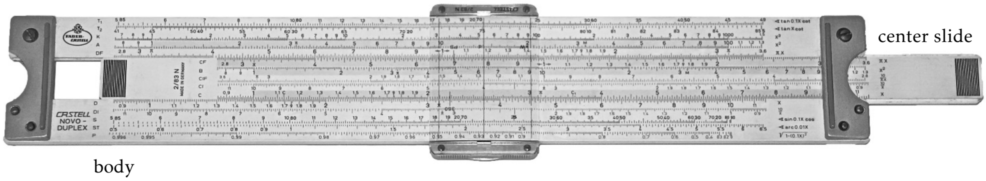


Figure 2.1: Faber Castell slide rule model 2/83N

Figure 2.1 shows one of the last, most complex and most versatile slide rules ever built, a Faber Castell 2/83N. The three main parts of traditional slide rules are clearly visible:

- The *body* which consists of the top and bottom *stator* or *stock*. The two stators are held together by two *end braces* or *end brackets*.
- The *center slide* which can be moved laterally with respect to the body.
- The *cursor* which slides in grooves of the body.

While simple slide rules only feature a couple of scales, complex ones as the 2/83N have up to 30 scales which implement functions far beyond multiplication and division.[12] Using these scales, trigonometric functions, exponentiation etc. can be evaluated. Until the 1970s when pocket calculators took over[13] slide rules were ubiquitous as early as around 1800 in some countries (like Great Britain) as the following quotation from Sedlacek[14] shows:

> *"It is said that the use of the slide rule in England is so widespread that no tailor makes a pair of trousers without including a pocket just for carrying a 'sliding rule'. During such a time, it is difficult to understand why the slide rule does not enjoy such well-deserved recognition in our own country."*

Slide rules were essential tools for the scientific and technological progress of the last three centuries ranging from civil engineering, commercial applications, electronics etc. up to applications in space flight.[15] Although they have been rendered more or less obsolete by pocket calculators 40 years ago, there are still areas of application where slide rules are employed regularly. For example, many aviators still use a flight computer like the E-6B, a special form of a circular slide rule, that allows the calculation of ground speed, i. e., the speed of an air plane corrected for wind effects, and many other crucial parameters.[16]

[12] If all scales are just on one side of the body, a slide rule is called *simplex*, if scales are found on both sides of the body, it is a *duplex* slide rule. In this case the cursor is also used to transfer partial results from one side of the body to the other.

[13] Early pocket calculators as the Hewlett Packard HP35 or some models made by Texas Instruments like the SR-10 etc. were explicitly marketed as *electronic slide rules*. The fixed number format featured by early HP calculators that was often set to display only 2 or 4 decimal places also was a reverence for the slide rule.

[14] Cf. [Jezierski 2000][p. 16].

[15] In fact the Apollo-11 mission carried a Pickett Model N600-ES *Log Log Speed Rule*.

[16] Apart from the ease of use such specialized slide rules have the advantage of not requiring any electrical power or the like for their operation.

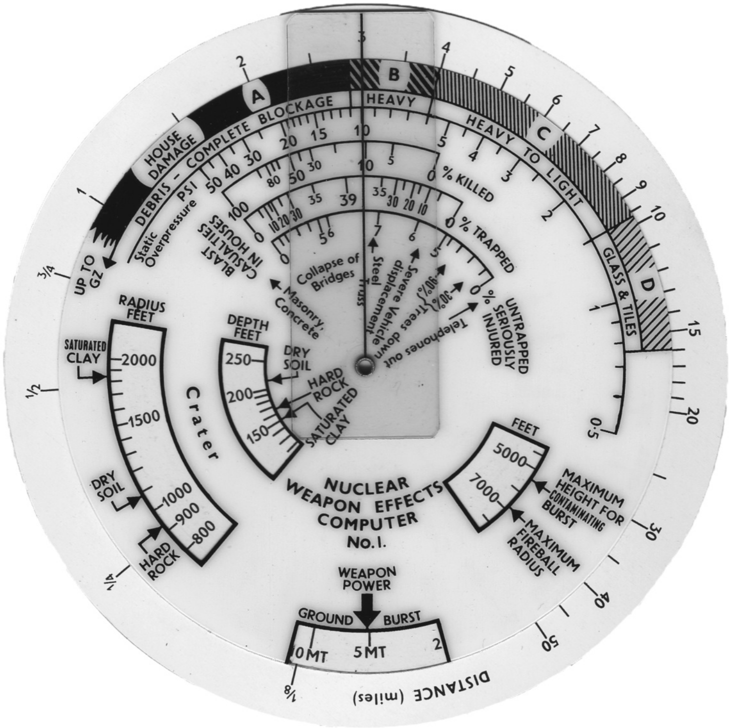


Figure 2.2: Nuclear Weapon Effects Computer

Figure 2.2 shows a creepy special purpose circular slide rule that was deployed in rather large amounts during the Cold War – a *Nuclear Weapon Effects Computer* that allowed rough estimates of fatalities and damages should a nuclear air burst occur.

2.4 Planimeters

A so-called *planimeter* is a fascinating instrument. Its purpose, most aptly described by [Henrici 1894][p. 497], is the following:

> *"The object of a planimeter is to measure an area; it has, therefore, to solve a geometrical problem by mechanical means."*

Measuring areas enclosed by some "good-natured" boundary curve is an important task in many branches of science as well as in commercial applications, registers of real estate and many more. A typical early application was to analyze pressure/volume or pressure/time indicator diagrams[17] as those written by recording steam engine in-

[17] See [Hütte 1926][pp. 380 f.].

dicators,[18] which requires the determination of the area enclosed by a curve which is either plotted in a Cartesian or more often a polar coordinate system. A simple and direct method for performing this task is to cut out the area to be determined and weigh the resulting piece of paper which yields rather good results. Although this method is sometimes still used in teaching chemistry students who regularly have to determine integrals over curves generated by spectrometers and the like, it is not really suitable for every-day usage.

As early as 1814 J. M. Hermann, a Bavarian engineer, invented a planimeter that was built, after improvement by Lämmle, in about 1817. Unfortunately, this instrument seems to have gone unnoticed by his contemporaries and had no obvious influence on the following developments.[19] In 1824 an Italian professor for mathematics, Tito Gonnella, invented a so-called *wheel-and-cone planimeter* that used a friction-wheel integrator (see section 2.5.3) to perform the necessary integration.[20] The first planimeter that was put into production was a device developed by a Swiss engineer named Johannes Oppikofer who developed two wheel-and-cone planimeters in 1827 and 1836 and a planimeter based on a friction-wheel rolling on a disk in 1849. In fact, there is a plethora of different planimeter principles and implementation variants.

The most successful type of planimeter is the so-called *polar planimeter* that was developed in 1854 by the Swiss mathematician Jacob Amsler-Laffon.[21] Figure 2.3 shows a typical polar planimeter[22] which is of a much simpler construction than most of the other instrument types.[23]

The basis of operation for planimeters in general is the Green-formula that relates a double integral over a closed region, i. e., the area to be determined, to a line integral over the boundary of this region. Thus a planimeter is a mechanization of

$$\iint\left(\frac{\partial Q}{\partial x}-\frac{\partial P}{\partial y}\right)\mathrm{d}A=\oint F\mathrm{d}r.$$

Choosing P and Q in a way that the difference under the left integral equals one yields the area sought. Interestingly, it seems as if the first explanation of the operation of planimeters using Green's formula was given in [Ascoli 1947].[24]

[18]The first of these devices was invented by James Watt's assistent John Southern around 1796 (cf. [Miller 2011]).

[19]Cf. [Henrici 1894][p. 505].

[20][Haeberlin et al.2011] gives a nice description of this instrument. Another good source is [Henrici 1894][p. 500].

[21]11/11/1823–01/03/1912

[22][Foote et al. 2007] shows how to build a polar planimeter with rather simple means.

[23]One notable exception is the *Prytz planimeter* (also known due to its physical shape as a *hatchet planimeter*) that was developed by the Danish mathematician and cavalry officer Holger Prytz (who published under the pseudonym "Z") around 1875. His instrument has no moving parts whatsoever but is of very limited precision. A good description of the principle of operation of this instrument can be found in [Foote et al. 2007][pp. 82 ff.].

[24]Explanations of the operation of planimeters with different chains of reasoning can be found in [Henrici 1894], [Meyer zur Capellen 1949][pp. 179 ff.], [Leise 2007]. Some patents describing interesting planimeter designs are [Coffin 1882], [Snow 1930] etc.

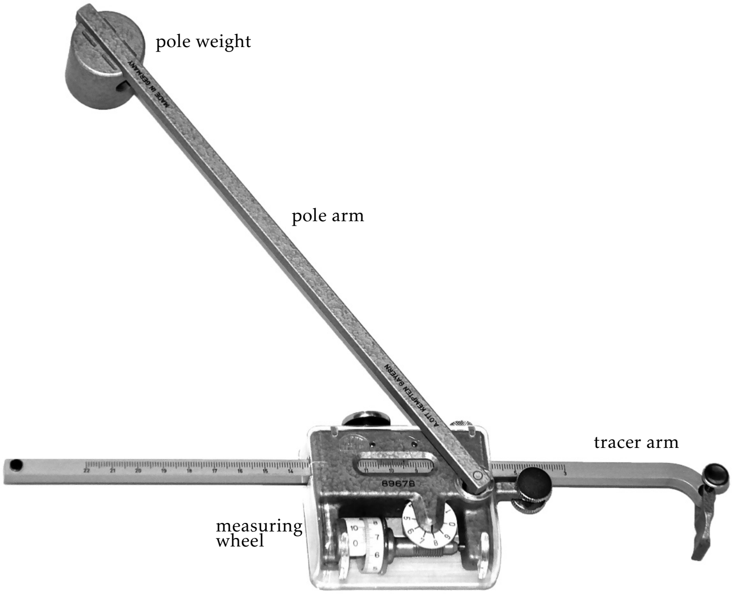


Figure 2.3: Typical polar planimeter

The main parts of a polar planimeter as the one shown in figure 2.3 are two arms connected with an elbow that is restricted to move on the circumference of a circle. The end of one arm, the *pole arm* is fixed at a point called *pole* by means of a *pole weight* or a needle, while the end of the other arm, the *tracer arm* can be moved freely by hand. This end is often formed as a needle or a magnifying lens to simplify tracing the boundary curve of the area to be measured. At the elbow is the integrating or measuring wheel.[25]

To measure the area of a circumscribed figure, the pole is fixed either outside or inside the figure, depending on its size. Then the measuring wheel with its vernier and the counting dial are reset with the needle or lens placed on the starting point of the boundary. Then the boundary line is followed manually in clockwise direction. After reaching the starting point again, the area can be read from the counting dial, the wheel and the vernier. If extended precision is necessary the same procedure can be applied a second time while following the boundary line counterclockwise[26] which, of course, yields a result with the opposite sign, the complement.

By setting the position of the elbow, which can normally be shifted along the tracer arm, various scaling values can be set, so areas can be measured in inches as well as in centimeters without the necessity to explicitly convert units.

[25]This wheel is extremely delicate – its metal wheel tread should never be touched by hand since it is engraved with microscopic rills which are vital for the overall precision of the instrument and are easily damaged or cluttered with dirt.

[26]Typical errors caused by a wheel with an axis not being perfectly horizontal can thus be compensated to a certain degree.

Determining large areas can be problematic since the counting dial can only represent one significant digit. Thus as early as 1961 planimeters coupled with electronic counters were built. [Zuse Z80 1961] describes a linear planimeter[27] that was coupled to an electronic counter capable of processing up to 250000 impulses per second. The pickup from the measuring wheel was done photoelectrically thus allowing even better precision than traditional mechanical instruments.

[Lewin 1972] describes another development that was patented in 1972: Here the position of the tracer arm is sensed by two linear potentiometers generating voltages directly proportional to the current (x,y)-position of the tracer. These voltages in turn control an oscillator and some monostable devices. The integration process is then performed basically by a chain of decade counters. This scheme proved to be too complicated and costly for the market. Another rather recent development is described by [Light 1975]. Here the position of a tracer needle or the like is determined by conductive foils which are placed under the chart containing the curve to be integrated over.[28]

As old as planimeters are, some companies still manufacture polar and linear planimeters which achieve accuracies of about $1\,{}^0\!/_{00}$. These instruments are still used regularly for surveying and mapping, for determining the area of furs and fabric etc.

2.5 Mechanical computing elements

All of the instruments shown in the preceding sections are specialized analog computers, capable only of solving just one distinct problem each. This is obviously caused by their fixed structure – a slide rule can only add lengths, so the only variation possible is that of employing different scales to extend this basic operation to multiplication and division and many more. In the same sense planimeters are specialized instruments for determining areas only. The following sections will now cover some basic mechanical computing elements that can be used to build true – in the sense of their reconfigurability – analog computers, so-called *differential analyzers*.[29]

Typically, mechanical analog computers represent values by rotations of shafts which interconnect the various computing elements or by positions of linkages. As simple as most mechanical computing elements seem, they are quite powerful tools. Precisions up to 10^{-4} are possible given precise machining of the parts involved and clever scaling of the equations to be solved. In fact, mechanical analog computers even have one advantage over analog electronic analog computers: The former can integrate over every variable while the latter can only integrate over time which often requires some ingenuity to solve partial differential equations[30] and other problems.

[27]These differ from polar planimeters in so far as the tracer arm is not free to rotate around the elbow connected to the pole arm but is guided in a strictly linear fashion which is normally implemented by a two-wheel carriage that is dragged behind by the tracer arm while tracing the curve.

[28]Although this instrument was not a financial success due to its complexity, its pickup mechanism anticipated the basic techniques used for today's touch screens and the like.

[29]In depth information about mechanical computing elements can be found in [Svoboda 1948], [Meyer zur Capellen 1949] and [Willers 1943].

[30]This particular advantage of mechanical analog computers is also exhibited by digital analog computers, see chapter 10 which make these devices quite interesting for future computer architectures.

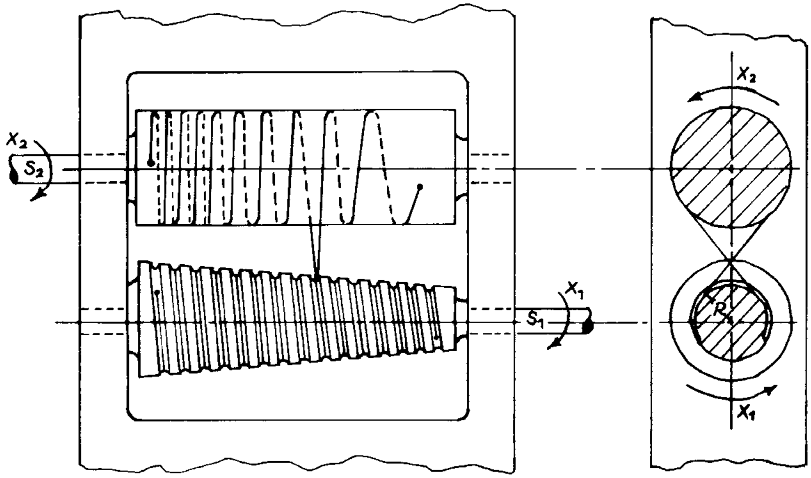


Figure 2.4: Squaring cam yielding $x_2 = x^2$ (cf. [Svoboda 1948][p. 22])

2.5.1 Function generation

A common task in simulations is the generation of functions which are either defined analytically or by measured values. Functions of a single variable are rather easily implemented using cams driven by a shaft, the angular position of which represents the input variable. A sensing pin measuring the position of the cam's surface yields the desired function value. The position of this pin can be translated into a rotational motion by a linear gear so this output value can be used as input for another element expecting an angular shaft position as its input value.

Another type of generator for a function of one variable is shown in figure 2.4. This is a so-called *squaring cam* which is used to generate a square function $f(x) = x^2$. If $\mathrm{d}x_1$ denotes the element of rotation of the input shaft, the resulting movement of the string wound around the cone is $r_1 \mathrm{d}x_1$ where r_1 denotes the average diameter of the cone at the current position of the string. With r_2 denoting the diameter of the drum, the drum rotation resulting from $\mathrm{d}x_1$ is

$$\mathrm{d}x_2 = -\frac{r_1 \mathrm{d}x_1}{r_2}.$$

r_1 is proportional to the angle of rotation x_1 thus

$$\mathrm{d}x_2 = -\frac{k x_1 \mathrm{d}x_1}{r_2}$$

with k being a constant of proportionality. Integration finally yields

$$x_2 = -\frac{k}{2r_2} x_1^2$$

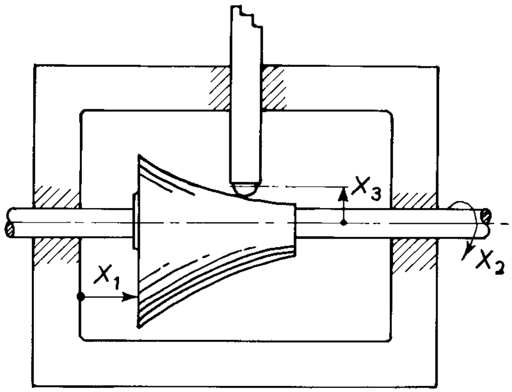


Figure 2.5: Three-dimensional cam as a function generator for a function of two variables $f(x,y)$ (cf. [Svoboda 1948][p. 23])

which is the desired square function. This mechanism does, of course, not work down to $x_1 = 0$ – if this is necessary, it may be combined with a differential gear as described in section 2.5.2.[31]

In many cases, functions of two variables are necessary – figure 2.5 shows a so-called *barrel cam*[32] used to implement a function $x_3 = f(x_1, x_2)$. The value of the input variable x_1 controls the lateral displacement of the three-dimensional barrel cam while the second input variable, x_2, controls the angular position of the cam. The output value is sensed by a pin that gauges the cam's surface. Other implementations feature a movable sensing pin carriage instead of a barrel cam that can be shifted laterally which allows for a more compact design.

Barrel cams like this were often used in mechanical fire control systems but were also used in some cases in electronic analog computers since the generation of functions of two variables is a rather complicated task for such a machine. In this case the input shafts for x_1 and x_2 are controlled by electronic servo mechanisms while the output sensing pin is connected to a potentiometer which delivers a voltage proportional to the desired function value.[33]

An important topic regarding cams and barrel cams is that their surface has to conform to several mechanical constraints to ensure that the sensing pin can follow the surface smoothly and that the pin can never block the cam in its movement due to steep slopes or grooves. This might sometimes collide with the mathematical requirements of the computer setup. In such cases functions may be generated by solving suitable differential equations or the like which normally requires much more additional computing elements than a straightforward barrel cam implementation.

[31]Cf. [Svoboda 1948][p. 22].

[32]Three-dimensional cams like this have also been called *camoids*, cf. [Svoboda 1948][p. 23].

[33]See section 4.5.7.

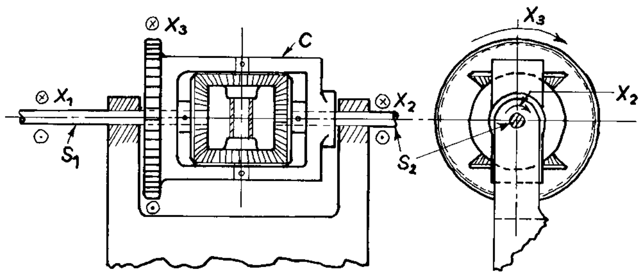


Figure 2.6: Structure of a bevel-gear differential (cf. [Svoboda 1948][p. 6])

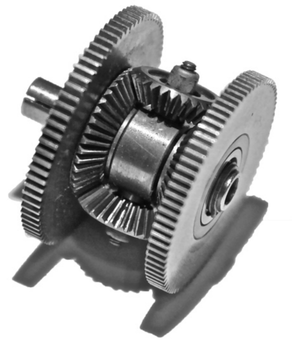


Figure 2.7: A differential gear from the dead reckoning computer PHI-4A-2 used in Starfighter jets

2.5.2 Differential gears

A barrel cam function generator like this could, in principle, be used to add or subtract two variables but a simpler, cheaper and more precise mechanism to implement this basic operation exists in form of the so-called *bevel-gear differential*[34] shown in figure 2.6. This device adds two shaft rotations x_1 and x_2 with some scaling parameters s_1 and s_2 which result from the actual design of the differential, yielding $x_3 = s_1 x_1 + s_2 x_2$.

A practical example of such a differential gear is shown in figure 2.7. This was used in the dead reckoning computer PHI-4A-2 of a Starfighter jet.[35]

2.5.3 Integrators

In contrast to most other machines, integration is a fundamental as well as natural operation for an analog computer. Given mechanical computing elements this operation is remarkably simple regarding its principles of operation.[36] The first mechanical integrators were developed in the early 19th century: In 1814 Johann Martin Hermann[37] developed an integrator consisting of a cone with a wheel rolling on its surface.[38] The position of the wheel on the envelope of the cone represents the values of the function to be integrated while the rotation of the cone corresponds to the variable of integration.[39]

[34]Such building blocks are also called *additive* or *linear* cells (cf. [Svoboda 1948][p. 6]).
[35]The central element carrying the bevel gears is called *spider block*.
[36]Far from being trivial is the implementation of integrators due to the necessary high precision in order to minimize the accumulation of errors within a simulation.
[37]1785–1841
[38]See [Petzold 1992][p. 26].
[39]A similar mechanism was developed later by Tito Gonnella and used as the basis for his planimeter.

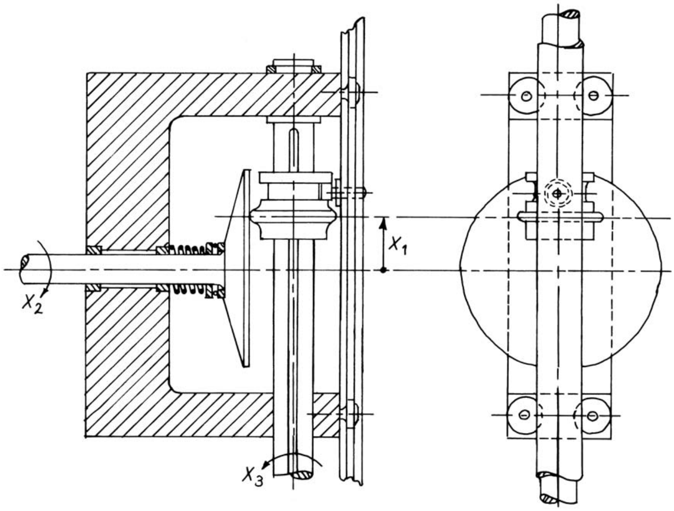


Figure 2.8: A friction-wheel integrator (cf. [SVOBODA 1948][p. 24])

Basically an integrator mechanizes the calculation of integrals like

$$x_3 = \int x_1 \, \mathrm{d}x_2$$

where x_2 is represented by the rotation of the integrator disk (or cone) while the values of x_1 control the position of the friction-wheel rolling on the disk surface. Figure 2.8 shows the structure of such an integrator.

Obviously it is $\mathrm{d}x_3 = kx_1 \mathrm{d}x_1$ where $k = 1/r$ denotes the radius of the friction-wheel thus yielding

$$x_3 = \int_{x_{20}}^{x_2} \frac{1}{r} x_1 \, \mathrm{d}x_2.$$

Figure 2.9 shows a friction-wheel integrator from the so-called *Oslo analyzer* – at its time one of the largest, most precise and most powerful differential analyzers in the world.[40] On the right hand side[41] the rotating disk S and the friction-wheel h with its associated shaft are visible. The wheel position on the surface of the disk with respect to the disk center corresponds to $F(x)$.

Based on this idea JAMES THOMSON, the brother of WILLIAM THOMSON – the later Lord KELVIN – developed in 1876 an integrator that replaced the friction-wheel by a steel

[40] This particular machine featured twelve integrators of this type and was used until 1954, see section 2.8.

[41] On the left side a so-called *torque amplifier* and a *frontlash unit* can be seen. These are typical devices in a mechanical differential analyzer but outside the scope of this book. Refer to [WILLERS 1943][pp. 236 ff.], [ROBINSON] or [FIFER 1961][p. 672] for more information about these units.

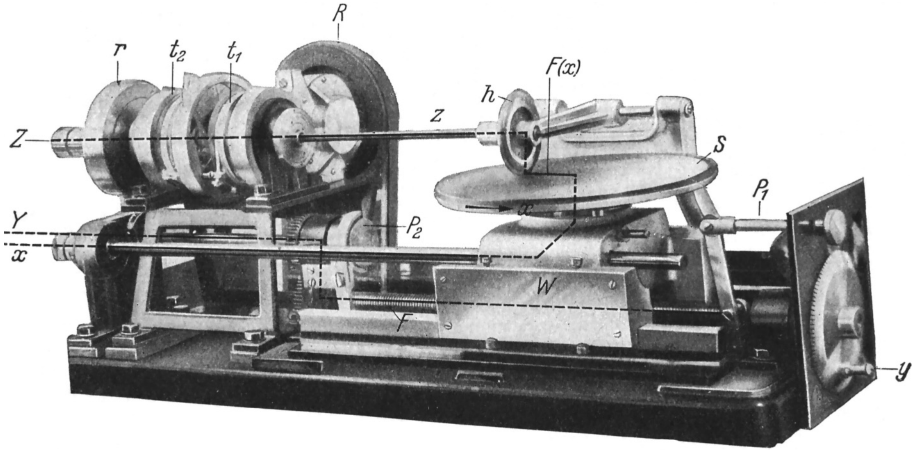


Figure 2.9: Integrator from the Oslo differential analyzer, see section 2.8 ([Willers 1943][p. 237])

sphere that runs in a movable cage (controlled by x_1). This sphere provides a frictional connection between the rotating disk and a cylinder that acts as a pickup for the integration result x_3. Figure 2.10 shows a variation of this implementation: Instead of a single sphere two stacked balls are guided in a movable cage. Such integrators are known as *double-ball integrators*.[42]

Based on this integrator developed by his brother, Lord Kelvin devised the idea of a machine suitable for solving differential equations. The basic concept was to start with the highest derivative in a differential equation to be solved and to generate all of the necessary lower derivatives by repeated integration. Using these derivatives, the highest derivative – that was the starting point for this procedure – can now be synthesized by combining the lower derivatives in an appropriate way.[43] The discovery of this method is described in [Thomson 1876] as follows:

> *"But then came a pleasing surprise. Compel agreement between the function fed into the double machine and that given out by it [...] The motion of each will thus be necessarily a solution of [the equation to be solved]. Thus I was led to the conclusion, which was unexpected; and it seems to me very remarkable that the general differential equation of the second order with variable coefficients may be rigorously, continuously, and in a single process solved by a machine."*

Interestingly, Kelvin did not build a practical machine based on this discovery which is all the more puzzling since the (double) ball integrator allows the transfer of rather high torques which allows such devices to be chained to a certain degree without the

[42]Such double-ball integrators were used well into the second half of the 20th century. This was mainly due to their high precision (a brochure published by Librascope in 1957 (see [Librascope 1957]) mentions a repeatability of 10^{-4} – quite remarkable for a mechanical device) as well as to their robustness. Accordingly such integrators were used quite often in aerospace applications like dead reckoning computers etc.

[43]This *classical differential analyzer technique* as [Korn & Korn 1964][p. 1-5] puts it, is described in more detail in section 7.2.

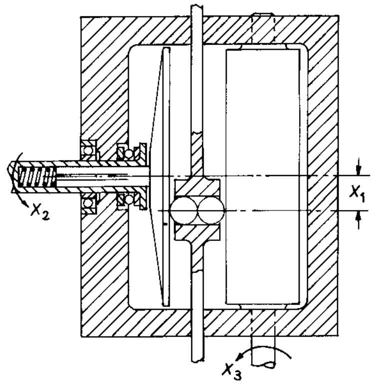


Figure 2.10: Principle of operation of a double-ball integrator (see [Svoboda 1948][p. 25])

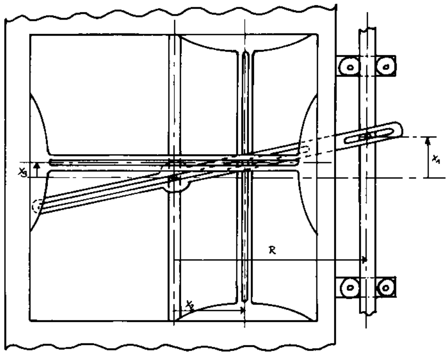


Figure 2.11: Basic structure of a slide multiplier (cf. [Svoboda 1948][p. 12])

necessity to use torque amplifiers as in the case of friction-wheel integrators which transmit only tiny amounts of force. Thus this brilliant idea of setting up an analog to represent differential equations fell into oblivion. The same happened again 45 years later when Udo Knorr[44] published a similar idea in 1921.[45] Knorr explicitly noted that coupled integrators can be used to solve a large group of differential equations of high degree.[46]

2.5.4 Multipliers

Multiplication is a rather difficult task for a mechanical analog computer. While multiplication with a constant can be easily performed with an appropriate gear mechanism or with a friction-wheel integrator where the position of the friction-wheel corresponds to the multiplier, the generalized operation where both input variables may vary is much more difficult to implement.

One common implementation is based on the product rule known from calculus. From $(uv)' = u'v + uv'$ it follows by integration that

$$uv = \int u \, \mathrm{d}v + \int v \, \mathrm{d}u \tag{2.1}$$

thus the multiplication of two variables can be implemented using two interconnected integrators.[47]

Figure 2.11 shows the structure of a so-called *slide multiplier*. It consists of two carriages sliding vertically and horizontally respectively and a lever that rotates around

[44]04/20/1887–07/10/1960

[45]Cf. [Petzold 1992][p. 33].

[46]See [Walther et al. 1949][p. 200].

[47]This requires that the integrators are not restricted with respect to the variable of integration. Only mechanical analog computers and DDAs (cf. section 10) fulfill this requirement. Analog electronic analog computers can only use time as the variable of integration, so this class of machines requires different multiplication schemes.

a center pin that is fixed to the enclosure. Both carriages and the lever are coupled with another pin that runs in grooves of these three elements. The input variables are x_1, represented by the displacement of the rotating lever's end, and x_2 which corresponds to the horizontal displacement of the second slider. The multiplication result $x_3 = kx_1x_2$ where k denotes a scaling factor that depends on the dimensions of the multiplier, is then available as the vertical displacement of the first slider.

2.6 Harmonic synthesizers and analyzers

Although Lord Kelvin did not succeed in building a true general purpose mechanical analog computer, he in fact built some quite complex special purpose analog computers[48] to predict tides by means of harmonic synthesis.[49] Even the earliest sailors had a genuine interest in tide prediction since accurate predictions lead to fewer lay days in harbors. A good description of the term *tide* is given by Kelvin himself:[50]

> *"The tides have something to do with motion of the sea. Rise and fall of the sea is sometimes called a tide; but I see, in the Admiralty Chart of the Firth of Clyde, the whole space between Ailsa Craig and the Ayrshire coast marked 'very little tide here'. Now, we find there a good ten feet rise and fall, and yet we are authoritatively told there is very little tide. The truth is, the word 'tide' as used by sailors at sea means horizontal motion of water; but when used by landsmen or sailors in port, it means vertical motion of the water."*

Tides are caused and influenced by the superposition of a number of effects that are, in fact, harmonic oscillations with different amplitudes and frequencies.[51] The most important of these effects are the earth rotation, the rotation of the earth around the sun, the rotation of the moon around the earth, the precession of the moon perigee, the precession of the plane of the moon orbit etc.

In 1872 Kelvin developed the first so-called *harmonic synthesizer*, a specialized analog computer for generating harmonics and adding these together to generate a tide prediction.[52] This machine took ten partial tides into account. Its basic construction is shown in figure 2.12.[53] Using this machine it took only four hours to compute the 1400 tides that occur during a year for a given harbor. Performing the same task manually required several months.[54]

[48]For these machines the phrase *"substitute brass for brain"* was coined (see [Zachary 1999][p. 49] and [Everyday Science and Mechanics 1932]).

[49]Following the death of his first wife, Margaret Thomson nee Crum on June 17th, 1870, his interest for seafaring increased and he bought a 126 ton schooner, the Lalla Rookh, which in turn sparked his interest in tide prediction.

[50]See [Thomson 1882][Part I].

[51]These effects are called *partial tides*.

[52]Lord Kelvin's machine is on display at the Science Museum, London.

[53]Later machines generated even more partial tides. The *United States Coast and Geodetic tide-predicting machine No. 2* that was completed in 1910 generated 37 harmonic terms. [Aude et al. 1936] describes a machine that took 62 harmonics into account (this machine was in operation in Hamburg, Germany, until 1968 and is now on display at the *Deutsches Museum* in Munich).

[54]Cf. [Sauer].

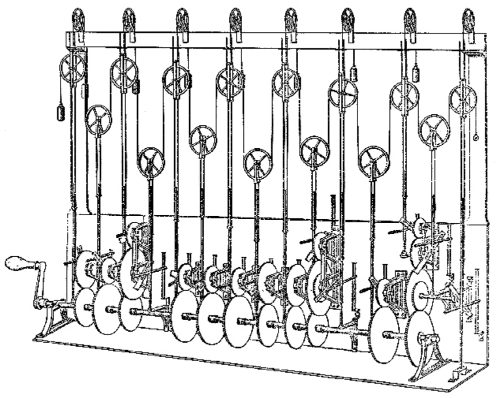


Figure 2.12: Principle of operation of KELVIN's tide predictor (see [THOMSON 1911])

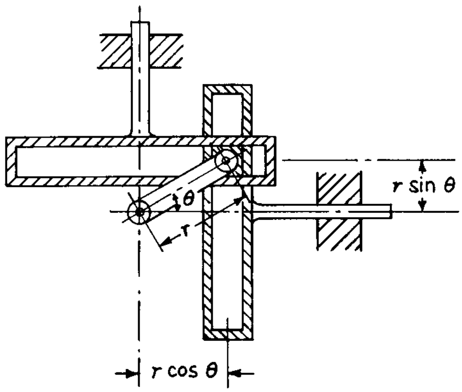


Figure 2.13: A Scotch yoke mechanism (cf. [KARPLUS et al. 1958][p. 242])

Generally speaking a harmonic synthesizer like a tide predictor generates a function $f(x)$ based on a given set of harmonics:[55]

$$f(x) = a_0 + a_1 \sin(x + b_1) + a_2 \sin(2x + b_2) + \cdots + a_n \sin(nx + b_n)$$

The basic harmonic functions are traditionally generated using a *Scotch yoke mechanism* as shown in figure 2.13. It consists of a boom that is mounted on a shaft so that it can rotate around this mounting point. The angular position of this shaft represents the input variable θ. The free end of the boom guides two carriages that are restricted to perform horizontal respectively vertical movements only. The movements of these carriages then represent the values $r\sin(\theta)$ and $r\cos(\theta)$ where r denotes the radius of the circle described by the free end of the rotating boom.

Since the values $r\cos(\theta)$ and $r\sin(\theta)$ are represented by linear displacements, harmonics generated this way cannot be added together using a differential gear. Instead a steel band of constant length is used as can be seen in figure 2.12. While KELVIN's tide predictor only generated a single output function by the superposition of the various harmonics, later harmonic synthesizers often had a more generalized structure like that shown in figure 2.14.

This fourth degree harmonic synthesizer consists of four Scotch yoke mechanisms as shown in figure 2.13, each of which generates a sine/cosine-pair and is parameterized by four crank length settings r, r^2, r^3 and r^4 (this rather unusual notation follows [KARPLUS et al. 1958][p. 242]). Two continuous tapes are threaded through the system adding all sine- and all cosine-values and driving the two coordinate inputs of a plotting mechanism.[56]

[55] Cf. [BERKELEY et al. 1956][pp. 135 ff.].

[56] A similar device, the high precision *Isograph* which contained 10 sine/cosine-units was built at the Bell Telephone Laboratories in 1937: *"In the Isograph, gears were fitted to the bearings with an accuracy of one ten-thousandth of an inch for play and concentricity."* (see [KARPLUS et al. 1958][p. 243]).

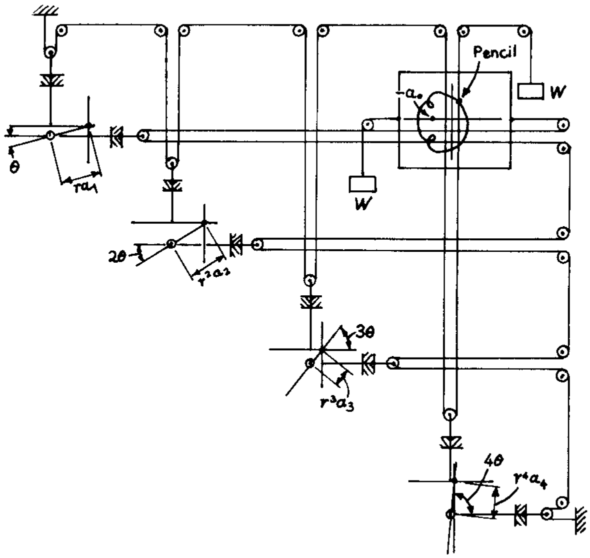


Figure 2.14: Harmonic synthesizer for a fourth-degree equation (cf. [KARPLUS et al. 1958][p. 242])

Later developments include a device described in [REDHEFFER 1953] which still used Scotch yoke mechanisms but with an electronic pickup in form of potentiometers that sensed the positions of the sine/cosine carriages. The output voltages of these potentiometers were then added by means of simple electronic networks and could directly be used to control an xy-recorder.

The inverse function of a harmonic synthesizer is performed by a *harmonic analyzer*.[57] These devices perform a FOURIER analysis, i. e., they determine the amplitude of the so-called *harmonics* that comprise a (more or less complex) signal. The first harmonic analyzer was developed in 1876 by Lord KELVIN.[58] It was capable of determining the amplitudes of eleven harmonics simultaneously. Its operation is based on the computation of

$$z_n = \int_0^{\pi} f(x)\sin(nx)\,\mathrm{d}x$$

which yields the amplitude of the n-th harmonic of the signal defined by the function $f(x)$ in the interval from 0 to π. The generation of $f(x)$ can be mechanized easily by a cam function generator or by manually tracing a curve with a tracer arm while the product nx is easily obtained using a gear with appropriate ratio. $\sin(nx)$ is then generated by a Scotch yoke mechanism and the integration is performed with an integrator like those described in section 2.5.3.[59]

[57] See [BERKELEY et al. 1956][pp. 132 ff.] and [WILLERS 1943][pp. 168 ff.].

[58] See [BERKELEY et al. 1956][p. 133].

[59] See [MCDONAL 1956] for a more detailed description of a practical harmonic analyzer.

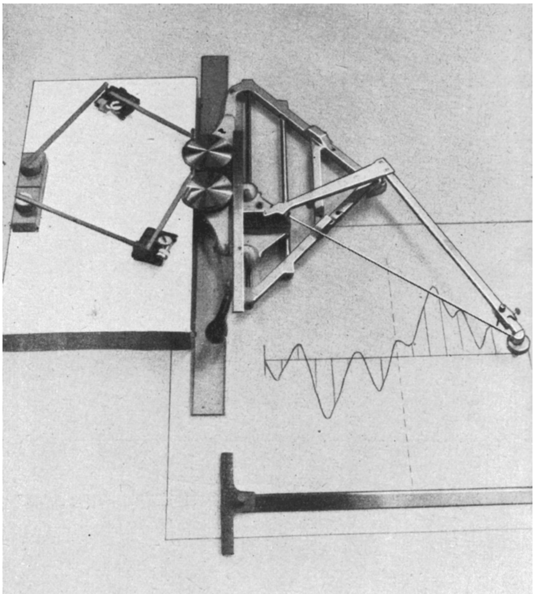


Figure 2.15: Harmonic analyzer made by Mader-Ott (cf. [Willers 1943][p. 180])

The determination of Fourier coefficients is also possible using only a planimeter. This startling technique is described in detail in [Willers 1943][pp. 171 ff.]. Since this manual process is not only tedious but also time consuming and error-prone, half-automatic harmonic analyzers based on planimeters have been developed. Figure 2.15 shows such a planimeter based harmonic analyzer that was developed and produced by Mader-Ott.[60]

Apart from obvious applications in fields like mechanical engineering where harmonic analyzers were used regularly to analyze complex vibrations and the like, they also played an important role in resource and oil exploration where seismograms must be analyzed to reveal the structure of earth's interior.

2.7 Mechanical fire control systems

Another type of very complex but still highly specialized analog computers are *fire control systems*. These military systems perform the following tasks:[61]

> *"Fire control equipment, that takes in indications of targets from optical or radar perception and using extensive calculating equipment puts out directions of bearing and elevation for aiming and time of firing for guns, according to a program that calculates motion of target, motion of the firing vehicle, properties of the air, etc."*

[60] See [Willers 1943][pp. 178 ff.].

[61] See [Berkeley et al. 1956][p. 163].

Until the late 19th century fire control was not too complicated a task given the relatively short ranges of typical gunnery. In the centuries before, firing *broadsides* was a common mode of fighting in naval warfare.[62] In the late 19th century shell ranges grew to up to six miles which made the task of aiming quite complex. Long range guns with ranges up to 17 miles were common at the end of World War II[63] – the time of flight of such a shell was in the range of about one minute – resulting in an aiming task so complex that it would have been impossible to hit a target reliably without the support of complex fire control systems which in fact were rather large mechanical analog computers. These machines were not only rugged but also extremely long-lived regarding their service life as [Bromley 1984][pp. 1 f.] puts it:

> *"Mechanical analog devices were first used for naval gunnery in World War I, were greatly developed for naval and anti-aircraft gunnery between the wars, were further developed and extended to aircraft systems during World War II, and continued in service in refined versions into the 1970s."*

Typical effects that have to be taken into account by a fire control system are (among others) these:

Wind and current effects: Flight times of up to one minute and more make shells quite susceptible for forces exerted by side winds and the like. Current effects acting on torpedoes are equally if not even more complicated to model and predict, mainly due to the long run time of such a weapon.

Heading: Both opponents are normally not at rest but moving (in the case of airplanes quite fast).

Environmental conditions: Barometric pressure, temperature, humidity etc. all influence the path of a shell or torpedo.

Aiming point vs. sensor position: Especially in the case of torpedoes there is normally a mismatch between the position of the sonar system resp. the periscope and the launching tubes on the side of the attacker and, in the case of sonar contact, the sound source and the desired aiming point on the attacked opponent.[64]

Signal run time: In case of sonar direction finding, the run time of the sound waves can often not be neglected regarding the ships' positions.

Figure 2.16 gives an impression of the immense complexity of such mechanical analog fire control systems. Depicted is the *Torpedo Data Computer Mark-3*[65] developed by the *Arma Corporation*.[66] This particular system is deservedly regarded as a masterpiece

[62] The typical range of fire during the historic encounter between the *USS Monitor* and the *CSS Virginia* on March 9th, 1862 was only about 100 meters (see [Clymer 1993][p. 21]).

[63] See [Bromley 1984][p. 2].

[64] Details of the German torpedo control system can be found in [Rössler 2005][pp. 79 ff.], including a schematic diagram of a typical analog lead-lag computer.

[65] *TDC Mark-3* for short.

[66] See [Clymer 1993][p. 28].

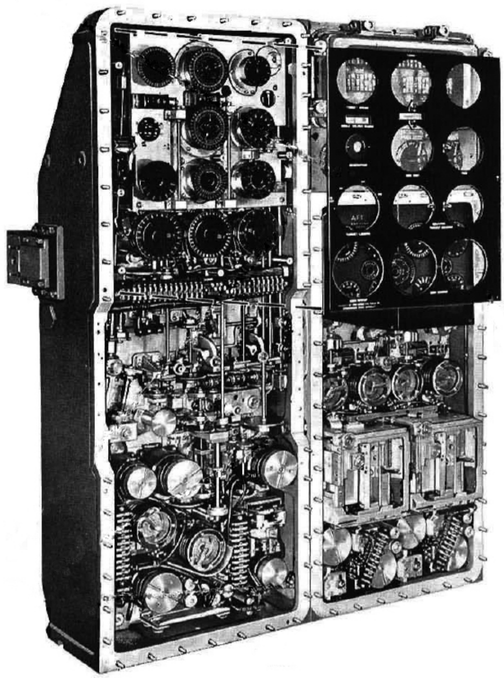


Figure 2.16: TDC Mark-3 with front panels removed (see [Bureau of Ordnance Publication 1944][p. 34])

mechanical analog computer. The two interconnected racks have a width, height and depth of 160 cm × 115 cm × 100 cm and weigh about 1.5 metric tons. The power consumption was quite impressive at 55.7 to 140 A at 115 V operating voltage.

A wealth of information about fire control systems can be found in [Clymer 1993], [Berkeley et al. 1956][pp. 286 ff.], [Friedman 2008][pp. 16 ff.], [Gray et al. 1955] and [Admiralty 1943].

2.8 Differential analyzers

A central figure in the development of the *differential analyzer* was Vannevar Bush[67] who developed a keen interest in analog computing techniques in the early 20th century.[68] He was not aware of the fundamental works of James Thomson and Lord Kelvin[69] and in later years – in fact, when he was informed about these prior de-

[67] 03/11/1890–06/30/1974

[68] As early as 1912 he submitted a patent describing a *profile tracer* that allowed to trace and plot a ground profile (see [Bush 1912]). This device contained two mechanical integrators in series that took the displacement of a pendulum mass as input thus yielding a profile trace by double integration over the acceleration of the mass.

[69] Cf. [Dodd 1969][p. 5] and [Zachary 1999][p. 49].

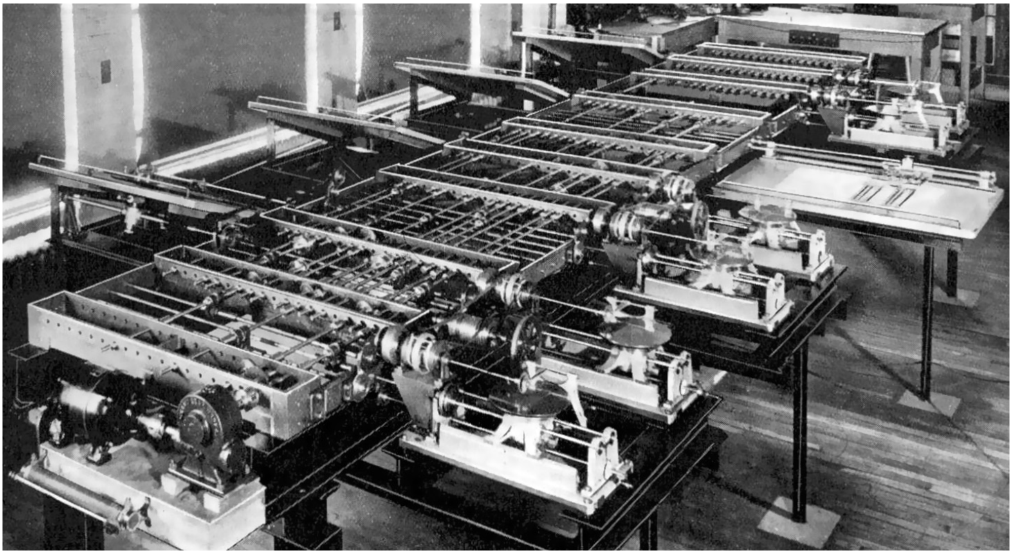


Figure 2.17: Vannevar Bush's differential analyzer (source: [Meccano 1934/2][p. 443])

velopments – he still claimed the intellectual property on his inventions for himself, stating that "*[i]nventors are supposed to produce operative results*" based on the fact that none of his predecessors had successfully built a working general purpose mechanical analog computer.[70]

In 1925, triggered by the increasing demand of computational power concerning power grid problems, Bush pursued the idea of a mechanical general purpose analog computer again. Six years later, in 1931, his machine, the first true general purpose mechanical analog computer, was completed[71] at the *Massachusetts Institute of Technology*.[72] This machine was, in fact, a collection of computing devices like those described in the preceding sections. In addition to these special units like torque amplifiers, frontlash units[73] and *helical gearboxes* had to be developed. The latter allowed the interconnection of the various computing elements on an interconnect unit that housed all input and output shafts called *bus rods* (all in all 18 such shafts). Figure 2.17 shows this differential analyzer.[74] To set up the machine to solve a particular problem, its computing elements had to be connected using these bus rods. Set up times of several days were not uncommon.[75] One eight hour day of operation was charged at US$ 400.[76]

[70]See [Zachary 1999][p. 51].

[71]Its construction cost about US$ 25,000.

[72]*MIT* for short.

[73]See [Willers 1943][pp. 236 ff.], [Robinson] and [Fifer 1961][p. 672].

[74]More information about this particular machine can be found in [Meccano 1934/2], [Zachary 1999] and [Gleiser 1980]. A rather simple but still useful mechanical differential analyzer is described in [Kasper 1955].

[75]See [Zachary 1999][p. 51].

[76]See [Macnee 1948][Sec. I].

This machine, containing six integrators that can be seen on the right in figure 2.17, proved to be highly influential and successful.[77] It inspired other researchers to implement their own differential analyzers like the following:

- A Meccano based differential analyzer built by DOUGLAS HARTREE and ARTHUR PORTER at Manchester University in 1934,[78]
- a similar Meccano based machine built by J. B. BRATT at Cambridge University in 1935,[79]
- the *Oslo Analyzer*[80] that was built from 1938 to 1942 at Oslo University's Institute of Theoretical Astrophysics under the auspices of SVEIN ROSSELAND[81] who visited the MIT and knew of the MIT differential analyzer,[82]
- a differential analyzer with six integrators that was built and completed in 1938 in Russia,[83]
- the *Integrieranlage IPM-Ott* that was built in Germany[84] (start of development in 1938),
- yet another mechanical differential analyzer with only three integrators was built as late as 1959 at the *Institut für Angewandte Mathematik und Mechanik*, Friedrich-Schiller University Jena (Germany), under the leadership of ERNST WEINEL.[85]

As an example a simple differential analyzer setup is shown in figure 2.18. The differential analyzer is used to integrate over a function $y = f(x)$ which is provided as a plot on the *input table* on the upper left.[86] The integral over $f(x)$ is to be determined between the limits x_1 and x_2. Therefore the motor drives three shafts of the differential analyzer: Two shafts control the movement of the stylus of the input and *output table* (upper right) in x-direction while the third shaft drives the integrator disk, the rotation and thus time being the variable of integration.

In the simplest case a human operator would now track the curve on the input table by turning the crank accordingly to keep the curve in the cross-hairs of the input

[77] It was soon called *thinking device, mechanical brain* or even *man-made brain* by the press (see [ZACHARY 1999][p. 51]).

[78] See [Meccano 1934] and [Meccano 1934/2].

[79] This machine is now part of the collection of the *Museum of Transport and Technology*, Auckland, New Zealand: `http://www.motat.org.nz/explore/objects/differential-analyser` (retrieved 11/30/2012).

[80] Cf. [HOLST 1982] and [HOLST 1996].

[81] 03/18/1894–01/19/1985

[82] A bit of trivia: When Germany occupied Norway during World War II, ROSSELAND removed the integrator wheels from the machine and buried them safely packaged behind the institute to make sure that his differential analyzer would not support Germany's war efforts.

[83] See [ETERMAN 1960][pp. 39 ff.].

[84] See [WALTHER et al. 1949].

[85] This machine was primarily intended to be used in education, see [KRAUSE 2006].

[86] The problem of determining the area under a curve is, in fact, so simple and could be solved by employing a planimeter (see section 2.4) that such a setup would never have been found in a practical differential analyzer installation.

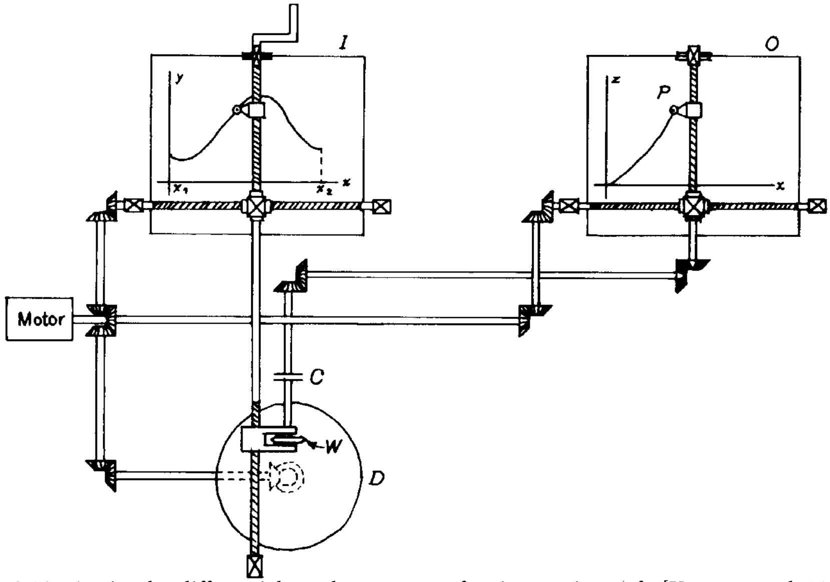


Figure 2.18: A simple differential analyzer setup for integration (cf. [KARPLUS et al. 1958][p. 190], [SOROKA 1962][p. 8-10])

table's magnifier lens. This in turn changes the displacement of the friction-wheel of the integrator in a way that the angular displacement of w represents

$$x = \int_{x_1}^{x_2} f(x)\,\mathrm{d}x.$$

The shaft coupled to the friction-wheel of the integrator finally drives the y-input of the output table which plots the integral of $f(x)$.

Even today mechanical differential analyzers are still fascinating devices. Figure 2.19 shows a modern implementation of such an analog computer using Meccano parts which is being developed by TIM ROBINSON since 2003.

The main problem of these differential analyzers were the long setup times necessary due to the direct mechanical interconnection between the various computing units. Thus so-called *electromechanical differential analyzers* were conceived which used intricate rotational sensors and servo mechanisms to make the necessary connection between the still mechanical computing elements. In 1945 the MIT announced the completion of one of the first such electromechanical differential analyzers[87] containing 18 integrators, the *MIT II*. The rotational sensors are described by [BERKELEY et al. 1956][p. 116] as follows:

[87] Another such electromechanical differential analyzer was built in Germany from 1939 to 1945. This particular machine contained four integrators, six function tables with optical sampling and two multipliers. This machine was used successfully and commercially well into the late 1950s. More information can be found in [WALTHER et al. 1949].

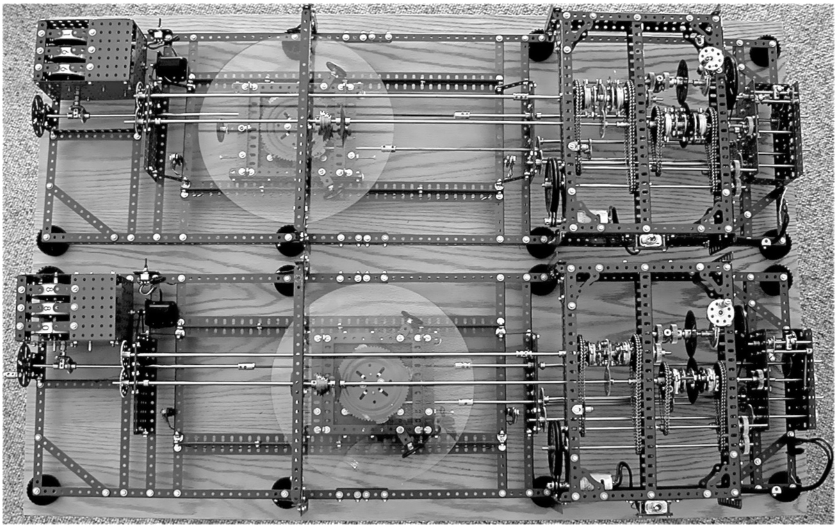


Figure 2.19: Tim Robinson's Meccano differential analyzer (reprinted with permission of Tim Robinson)

> *"[G]reat ingenuity and pains have been devoted to making it a precision mechanism. In particular, the take-off of wheel rotation is effected by an electrostatic* angle indicator. *This imposes virtually no load at all on the turning of the wheel [. . .]"*

The signals generated by these take-offs were then fed into servo circuits that drove motors which were assigned to the various input shafts of the individual units. The interconnection between these inputs and the servo outputs was accomplished by a patch panel which allowed rather short setup times and thus eliminated the main drawback of its predecessor machines. In addition to this, the summers had remotely controllable gear boxes for every single input shaft[88] thus even the change of coefficients could be performed at a central control panel. In the following years quite some of these electromechanical differential analyzers were built worldwide.

The *Minden* system is an example of this: In 1948 the German company *Schoppe & Faeser* started the development of a system that was named Minden[89] of which finally three machines were built and sold. The last system contained twelve integrators, 20 summers, and ten function tables[90] The computing elements had a precision of 0.01%. Figure 2.20 shows a double integrator unit of a Minden system while figure 2.21 gives an overview of a typical Minden installation.[91]

---

[88] See [Berkeley et al. 1956][p. 117].

[89] This naming was due to the fact that development took place in Minden, Westphalia (Germany).

[90] These function tables were based on input tables and allowed to trace a curve given an x coordinate as input yielding the y value of the curve at this particular point.

[91] This last and largest Minden system was used by the German company *Siemens-Schuckert* until 1971 (see [Petzold 1992][p. 53]). Most parts of the machine are now part of the collection of the Deutsches Museum in Munich.

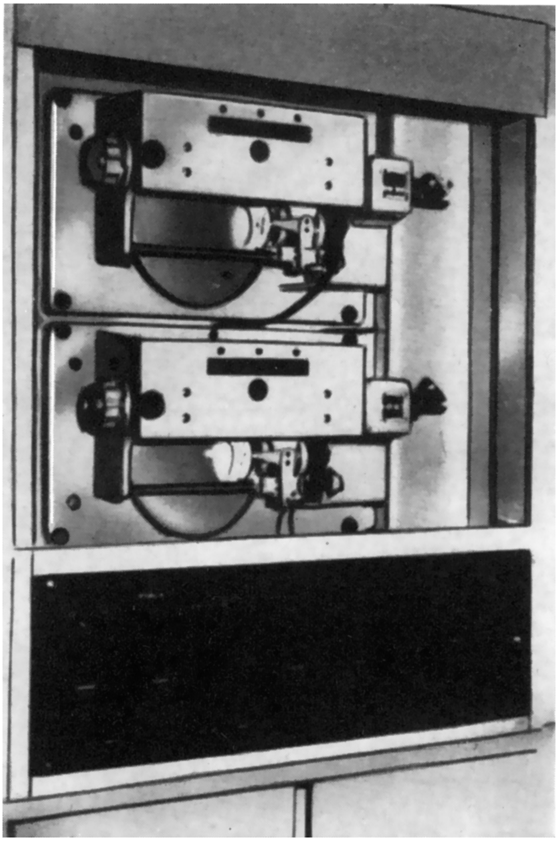


Figure 2.20: Double integrator of the *Minden* system (see [Rationalisierungskuratorium 1957][p. 51])

Despite their advantages over purely mechanical differential analyzers such electromechanical systems could not compete with the emerging analog electronic analog computers that offered much greater speed, flexibility and ease of construction and maintenance. A last attempt to build a competitive electromechanical differential analyzer was undertaken by Hans Bückner who developed the so-called *Integromat* in 1949.[92] This system was built at Schoppe & Faeser and was still based on mechanical computing elements that were interconnected electromechanically. The rationale behind this development was that traditional electromechanical differential analyzers were too complicated and thus too expensive to manufacture due to the high precision of their computing elements while this precision was not necessarily a requirement for some applications.

Accordingly, the Integromat used mechanical computing elements that were digital regarding the way in which values were represented. The integrators developed were called *Stufenintegrator*[93] and were based on sets of fine grained gears which were in-

[92]See [Bückner 1950].

[93]German for *step integrator*.

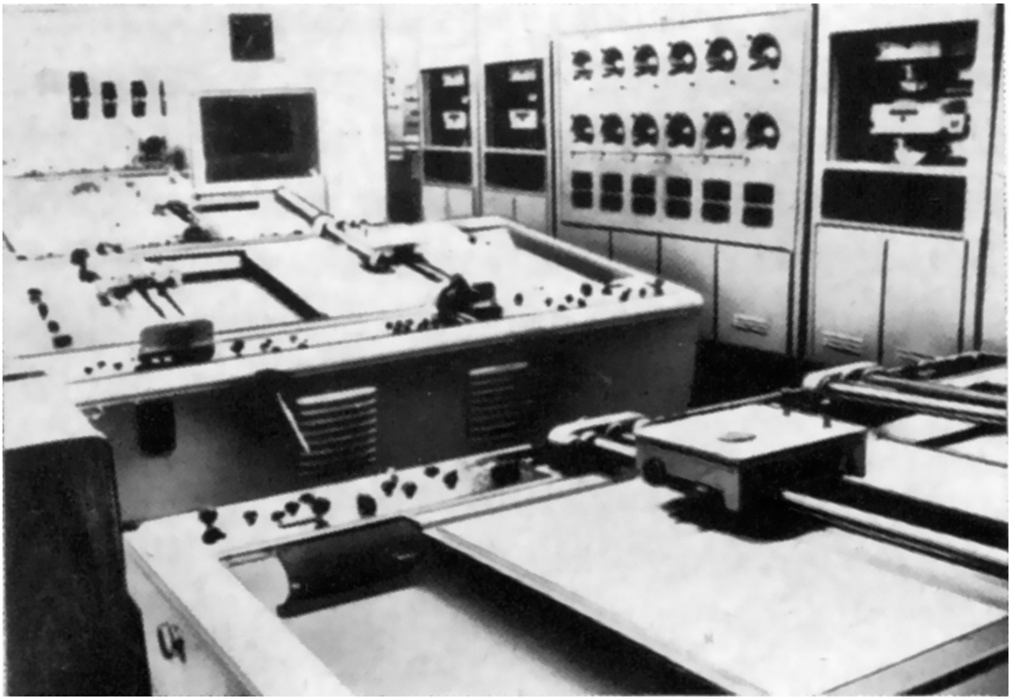


Figure 2.21: Differential analyzer *Minden* (see [Rationalisierungskuratorium 1957][p. 51])

terconnected with differential gears. The functions to be integrated controlled the gear ratios selected[94] while the variable of integration drove the input shaft of this multi-gear device.[95]

These integrators had the advantage of rather simple construction and high reliability but their precision turned out to be not satisfactory. Even a simple problem, the solution of the differential equation $\ddot{y} = -y$ which is often called *circle test* since it yields a sine/cosine signal pair that can be used to draw a circle on a two-coordinate output device, showed an error of about 0.5% after one period.[96] Research on damped oscillations showed errors in the range of 4.5% after only five periods.

Another problem was the slow speed of these step integrators which was worsened by the necessity to use very small step widths in order to keep errors small. Times of up to 45 minutes for the simulation of a single damped oscillator's oscillation were not uncommon.[97] Accordingly this machine turned out to be a dead-end although it was used well into the 1950s.

[94] Thus only discrete values could be used.

[95] All input shafts were driven by forerunners of today's stepper motors.

[96] Here and in the following a dot over a variable denotes its derivative with respect to time. Accordingly, two dots represent the second derivative and so on.

[97] See [Eggers 1954].

3 The first electronic analog computers

It is difficult to write about the "first" electronic analog computers because there is nothing like a single first machine. In fact, the 1930s and 1940s seemed to be ripe for the idea of building a differential analyzer based on analog electronic components. Thus the following sections concentrate on four largely independent early developments, namely the works of Helmut Hoelzer, the birth of electronic fire control systems, the early developments by George A. Philbrick and finally MIT's first electronic differential analyzer. These four lines of development form the foundation for nearly all of the developments of analog computers to come in the following decades.

3.1 Helmut Hoelzer

In 1935 a student of the University Darmstadt, Germany, Helmut Hoelzer[1] realized that using electronic circuits, mathematical operations like integration or differentiation could be implemented rather easily. He was led to these thoughts by the observation that there was no instrument that directly displayed the ground speed of an airplane. He noted[2] that he realized the following:

> *„[Es gab] in der ganzen Fliegerei nicht ein einziges Gerät [...], welches die absolute Geschwindigkeit eines Flugzeuges [...] gegenüber der Erde messen kann. Aha, dachte ich, das ist ja ganz einfach, man nimmt die Beschleunigung, die man ja messen kann, integriert sie und* voilà*! hier ist die Geschwindigkeit.“*[3]

Unfortunately, his ideas were neglected for a rather long time – especially mathematicians were reluctant about the idea to "compute" with arcane electronic devices instead of solving complex problems by pure thought. It was not until the development of the *A4 rocket*[4] that he could eventually explore and implement his ideas of a true general purpose analog computer.[5] In addition to that his developments led to the so-called *Mischgerät*,[6] world's first electronic on-board computer for guiding a rocket.

[1] 02/27/1912–08/19/1996

[2] See [Hoelzer 1992][p. 4].

[3] *"There was no device available in the whole area of aviation to indicate the ground speed of an airplane. So I thought, that is easy, just take the acceleration which can be measured, integrate it and there is the velocity."*

[4] See [Lange 2006] and [Dungan 2005] for information about the A4.

[5] The A4 rocket was later known as the "V2", short for "Vergeltungswaffe 2", German for "Vengeance weapon 2".

[6] This term can be roughly translated as *mixing unit*.

3.1.1 The "Mischgerät"

When World War II broke out, Hoelzer worked as an engineer for the renowned German company *Telefunken*,[7] developing radio transmission systems. One evening he was approached by Wernher von Braun and two of his friends, Hermann Steuding and Ernst Steinhoff, who led the department of *Orientierung, Steuerung und Bordinstrumentierung*[8] in *Peenemünde* where the A4 rocket was under development. They asked him to come to Peenemünde and help designing a radio guidance system for this new rocket system.

After following this call he started working on the proposed radio guidance system.[9] One particular problem that aroused his interest was the necessity to compute derivatives and integrals in flight – something completely unheard of before. Based on his thoughts about this topic about five years earlier he decided to use a capacitor as the central element for a circuit capable of differentiation and integration.

The current $i(t)$ flowing into a capacitor is described by

$$i(t) = C\dot{v}(t)$$

where C denotes the capacity and $v(t)$ is the voltage across the capacitor plates at time t. This, in turn, yields

$$v(t) = \frac{1}{C}\int i(t)\,\mathrm{d}t,$$

so both operations, differentiation and integration, can be implemented using a capacitor being charged. A simple, yet in many cases sufficient approximation is implemented by a simple RC circuit.[10] In the first case the current flowing through the capacitor can be measured as voltage drop over a sufficiently small resistor, in the second case the voltage across the capacitor's plates could be used directly as an approximation for the time integral of some varying input voltage.

In the 1940s this was easier said than done due to the fact that it requires DC coupled amplifiers since the voltages to be amplified and processed further are mainly DC voltages. An ideal DC amplifier would yield an output voltage of 0 Volts if its input voltage is 0 Volts, regardless of its gain which should be as high as possible. Due to imbalances, aging of components, temperature effects etc. this is not true for a basic DC coupled amplifier in which the various amplifier stages are coupled directly. Thus a small error voltage at an early stage will be propagated through the following stages, finally yielding a non-zero output signal even for a zero input signal. Such drift effects would preclude the application of such an amplifier as the active element of an electronic integrator since the integrating capacitor would integrate not only over the

[7] After the war Telefunken played a major role in the development of analog computers in the European market.

[8] German for *guidance, steering and on-board instrumentation*.

[9] See [Tomayko 1985][p. 230]. One of the finally developed radio guidance systems was named *Hawaii I* and was built by Telefunken (see [Lange 2006][pp. 163 ff.]).

[10] A circuit consisting of a resistor and a capacitor.

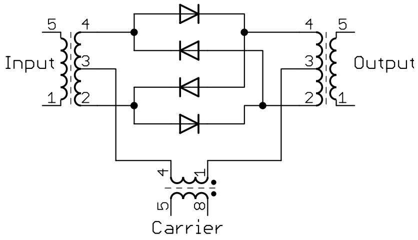


Figure 3.1: Basic ring modulator circuit (cf. [Aschoff 1938][p. 379])

Figure 3.2: Sirutor (scale in centimeters)

input function but also over a more or less random drift voltage rendering the result more or less useless.

Hoelzer decided to use a simpler AC coupled amplifier instead since AC coupled amplifiers do not exhibit excessive drift due to their individual stages not being galvanically coupled. Unfortunately this makes this kind of amplifier unsuitable for the use with DC voltages. So Helmut Hoelzer was faced with the problem of converting an input DC voltage to an AC voltage suitable for amplification by an AC coupled amplifier. The resulting output voltage had to be rectified again to get the desired DC signal.

The transformation of a DC signal into an AC voltage was finally done using a ring modulator[11] the basic structure of which is shown in figure 3.1. This circuit was described first in the early 1930s and used widely for radio applications.[12] Remarkably, most of even the earliest ring modulators already used some kind of solid state diodes, which were called *Sirutor* in Germany.[13] These devices were developed in 1934 and used a small stack of copper(I) oxide[14] pills. Figure 3.2 shows a typical Sirutor like those that were employed by Helmut Hoelzer for his ring modulators.

In 1940 the first usable radio guidance system for the A4 rocket was developed[15] and Hoelzer turned his interest to the development of a gyro based stabilization and guidance system. Like earlier developments of Robert Goddard[16] the A4 used movable vanes in the rockets engine exhaust, so-called *exhaust rudders*, as well as *air rudders* to control the flight path of the rocket. The exhaust rudders were the most effective means of control but could only work during the powered ascent phase. After reentry into earth's atmosphere the air rudder became effective.[17] This system turned out to

[11] The name stems from the fact that the central four diodes are often arranged in a ring-like structure.

[12] See [Aschoff 1938] and [Chance et al. 1949][p. 379].

[13] Short for *Siemens-Rundfunk-Detektor*.

[14] Cu_2O

[15] See [Tomayko 1985][p. 230].

[16] 10/05/1881–08/10/1945

[17] [Bate et al. 1971] give a thorough description of guidance and control of ballistic missiles.

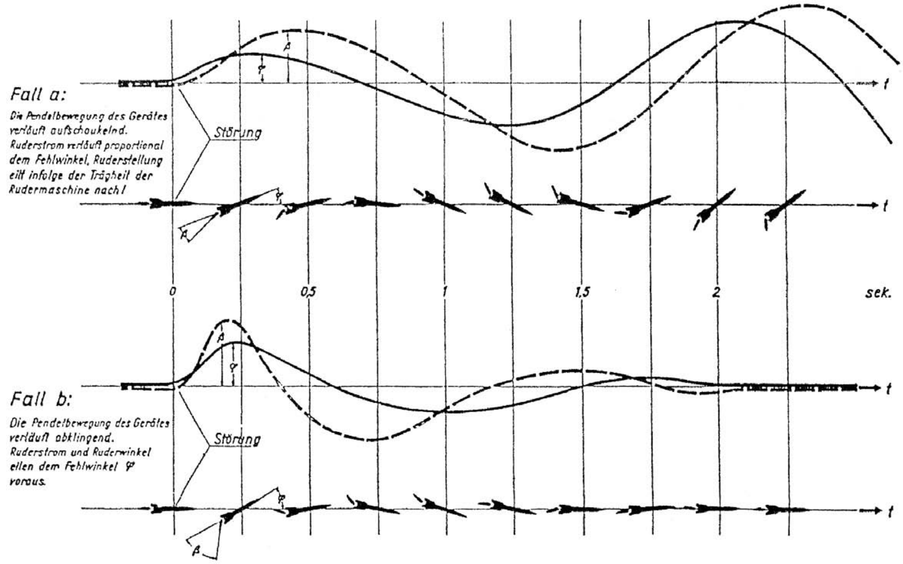


Figure 3.3: Control problem of the A4-rocket ([N. N. 1945][fig. 82])

be rather difficult to control due to the sheer size of the A4 and its associated inertia. The control system had to take these parameters into account to avoid a fatal build up of regulating oscillations. Figure 3.3 illustrates this problem: In the upper half the oscillation of the A4 with its increasing amplitude is shown that would result from a simple linear control unit. The lower half shows the desired behavior of the controller – the oscillation is quickly damped out which requires more than just a linear controller.

At first the company *Kreiselgeräte GmbH* attempted to develop a gyro based controller which turned out to be unstable as shown in the top half of figure 3.3. This first implementation did not take any derivatives of the signals generated by the gyro platform into account. Another approach using bank-and-turn indicators in addition to the main gyro turned out to be too costly. Von Braun approached Helmut Hoelzer after several unsuccessful attempts to build a working control system for the A4 since he was aware of the fact that Hoelzer's radio guidance system also needed derivatives and time integrals of sensor values which were generated using an RC combination instead of costly additional instruments. This turned out to be a viable approach – while the bank-and-turn indicators initially proposed cost about US$ 7000 the simple circuit developed by Hoelzer cost about US$ 2.50[18] and worked very well.

The idea of employing a differentiating circuit to derive rate of change signals from sources such as a gyro system was completely novel at that time and industry was sceptical to say the least. Hoelzer was approached with substantial hostility by the companies involved in the development work for the A4. Only the obvious success of his control scheme convinced these naysayers in the end. The resulting control sys-

[18]See [Tomayko 1985][p. 232].

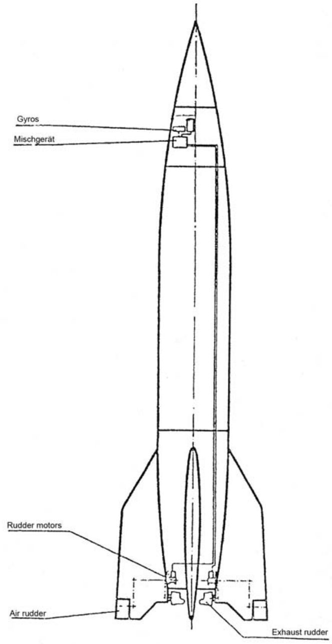


Figure 3.4: Main elements of the A4 control system ([N. N. 1945][fig. 83])

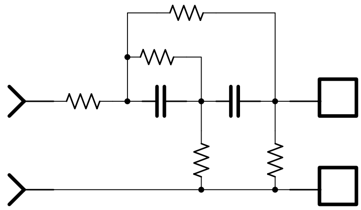


Figure 3.5: Structure of the basic RC combination used in the Mischgerät

tem was dubbed *Mischgerät* which can be translated as *mixing unit*, a camouflage term invented to disguise the real purpose of this first completely electronic on-board computer.

Figure 3.4 shows the main parts of the A4 guidance and control system. At the top of the rocket, directly below the war head, is a compartment that houses the gyro system, an optional radio receiver/transmitter, and the Mischgerät which controls the servo motors that operate the exhaust and air rudders at the lower end of the rocket.

A rather detailed schematic of the control system is shown in figure 3.6. On the top of the picture is the main bus of the rocket supplying all subsystems with 27 Volts DC. Using two motor-generator units, denoted by *Umf. I* and *Umf. II*,[19] this supply is converted into the necessary voltages for the gyro motors and the Mischgerät. The two gyros, *Richtgeber D* and *Richtgeber EA* are connected to the Mischgerät which is shown in the middle of the figure. Using RC combinations denoted by *RC Glied* the input signals are differentiated which effectively yields a PD controller.[20] The resulting output signals are used to drive the servo motors attached to the rudders. Figure 3.5 shows the structure of the basic RC combination used in the Mischgerät.

[19] Short for *Umformer*, German for a motor-generator unit.

[20] Using simple RC circuits directly to generate derivatives and integrations yields quite imprecise results. Since the gyro based control scheme for the A4 rocket only required linear terms and first and second derivatives, this was not a problem as long as the controller was operating stably. More precise computing circuits as they were used in general purpose analog computers would have been too costly, too large and too complicated for the intended use of the rocket.

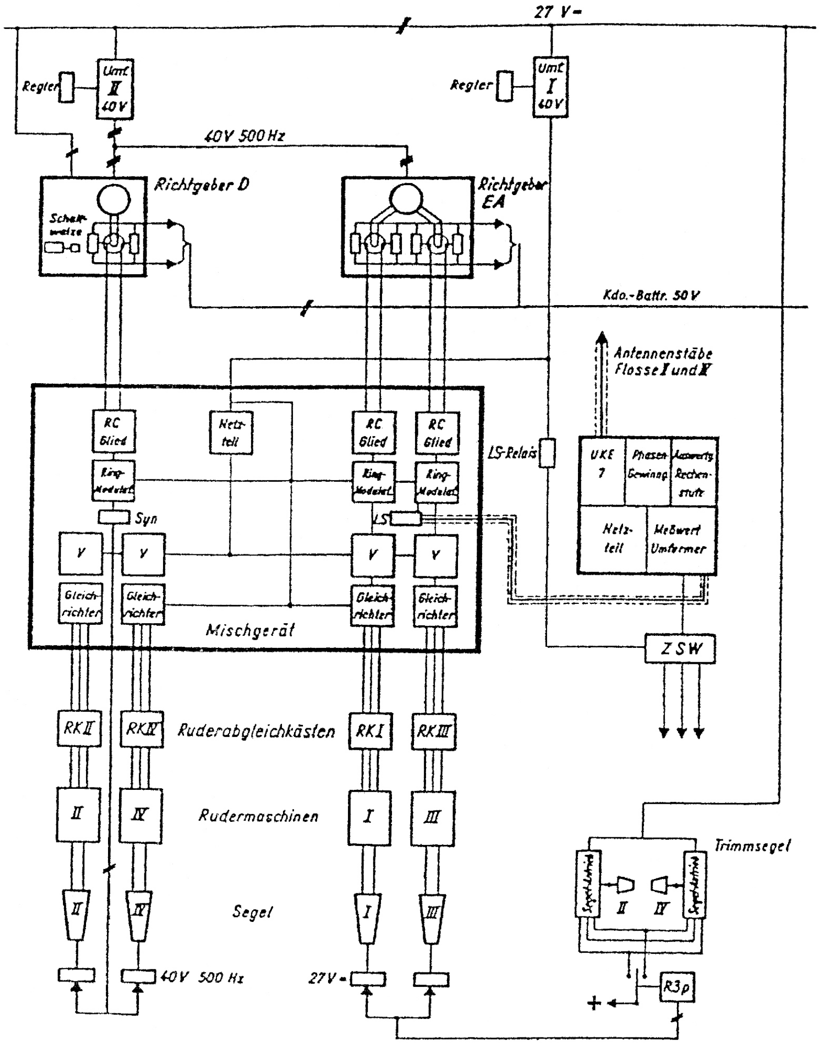


Figure 3.6: Simplified schematic of the A4 control system (see [N. N. 1945][fig. 80])

Figure 3.7 is a photograph of the control system's compartment in an A4 rocket. The Mischgerät itself is denoted by *MG*, *BB* and *ZB* are the main battery and a supplemental battery, while *U II* is the second motor-generator unit that powers the gyros which are installed in one of the other quadrants of this compartment.

The Mischgerät itself is shown in figure 3.8 – its small size and rugged design are obvious. It is built around a base plate which holds five plug-in modules, two at the bottom and three at the top. The empty space on the bottom right contains the connections to the rocket's subsystems.

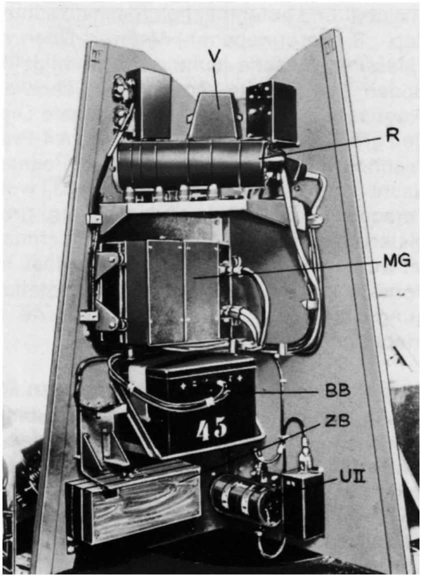


Figure 3.7: The installed Mischgerät ([Trenkle 1982][p. 134])

Figure 3.8: The Mischgerät (photo: Adri de Keijzer, reprinted with permission)

This device was described by [Tomayko 2000][p. 15] as the *"first fully electronic active control system"* and it had great influence on further developments as this quotation from [Bilstein 2003][p. 243] shows:

> *"Further work by other Peenemünde veterans and an analog guidance computer devised with American researchers at the Redstone Arsenal culminated in the ST-80, the stabilized platform, inertial guidance system installed in the Army's 1954 Redstone missile [...] The ST-80 of the Redstone evolved into Jupiter's ST-90 (1957) [...]"*

3.1.2 Hoelzer's analog computer

Parallel to his work on the Mischgerät and initially unnoticed by his supervisor, Helmut Hoelzer started development of a fully electronic general purpose analog computer in 1941. This was to be based on his idea of using RC combinations for integration and differentiation. Here he was confronted again with many objections and prejudice of other people. Hermann Steuding once commented about Hoelzer's ideas:[21]

> *"Young man, when I compute something, the results will be correct and I do not need a machine to verify it. By the way, machines cannot do this."*

He was even forced by his supervisor to stop fiddling around with this electronic contraption and concentrate on his real work as he remembers vividly:[22]

> *„Mein Chef kam ins Labor, sah [den] elektronische[n] Drahtverhau und sagte nur: ‚Hoelzer, hören Sie doch endlich auf mit dieser elektrischen Spielerei und kümmern Sie sich um Ihre Aufgabe. Ab morgen ist das alles weg, verstanden?' Ich sagte das einzige, was man in solcher Situation sagen konnte: ‚Jawohl, Herr Doktor.' Am nächsten Morgen war alles weg und zwar war es jetzt in einem kleinen Raum ohne Fenster [...] Aber mir schien es auch wichtig, ein Gerät zu schaffen, welches, wie man damals dachte, in der Hauptsache für die Entwicklung von Raketensteuerungen von nicht zu überbietender Wichtigkeit war. [...] Als alles funktionierte, wurde mir dann vergeben."*[23]

Further development on the analog computer continued in secrecy. After its completion it did not only prove to be a valuable tool, moreover it played a central role in the A4 development process. Without this analog computer, many problems, especially in the area of guidance and control of a ballistic missile like the A4, would not have

[21] See [Tomayko 1985][p. 234].

[22] See [Hoelzer 1992][pp. 13 f.].

[23] My boss came to the laboratory where he saw the electronic contraption. He said: 'Hoelzer, stop playing with this toy and do your work! I gave the only answer possible under these circumstances: 'Yes, sir!' Next morning, nothing was left in the lab – the computer had been moved to a small, windowless room. [...] Nevertheless, it seemed important to me to develop a system that could aid the development of rocket control systems significantly. [...] When it finally worked, everything was forgiven.

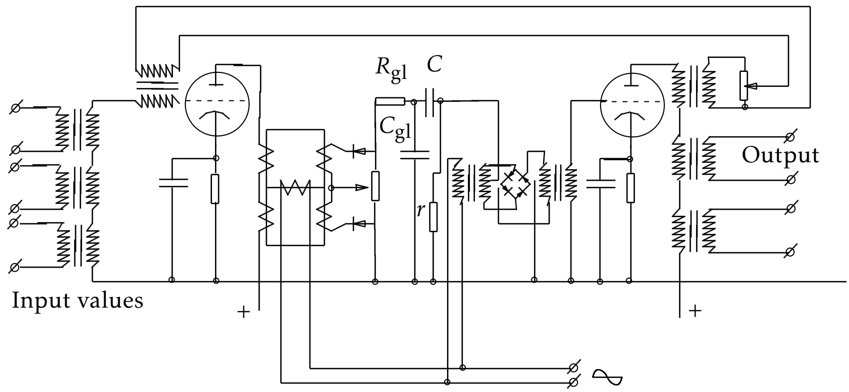


Figure 3.9: Schematic of Hoelzer's differentiator circuit (cf. [Hoelzer 1946][fig. I, 13])

been solved, at least not in time. In a short span of time many departments in Peenemünde used this machine actively to solve a variety of problems – even the word *analog* used to denote the setup of an analogon became part of engineer's lingo.[24] The role of this computer cannot be overestimated as [Neufeld 2007][p. 133] makes clear. He describes this computer as *"a fundamental innovation that really made a mass-produced guidance system possible"*.

Since this analog computer was intended to be a research instrument it had to exhibit a better precision than the simple RC combinations used in the Mischgerät for integration and differentiation. Therefore Hoelzer employed a controlled current source to charge the capacitor in the case of integration while a controlled voltage source was necessary for the differentiator. Both sources were effectively AC amplifiers due to the drift problems of DC amplifiers that had not yet been overcome.[25]

Figure 3.9 shows the schematic of the differentiator developed by Hoelzer for his analog computer.[26] The input values for this computing element are AC voltages that are coupled inductively into the computing element and are amplified by a triode based input stage. A synchronous demodulator in the anode circuit of this tube rectifies the amplified input AC signal. The resulting DC voltage is then used to charge the capacitor C while the charging current, which represents the derivative that is to be determined, results in a small voltage across resistor r. This voltage is then converted back into an AC voltage by a ring modulator. The output of this modulator drives the grid of the output amplifier stage that in turn feeds several output transformers connected in series.

Apart from this demodulation and modulation scheme there are three noteworthy aspects in this circuit: One of the outputs is used to generate a feedback signal, the amplitude of which can be set with the potentiometer shown in the upper right corner of figure 3.9. The modulator and demodulator are both driven by a common carrier

[24]See [Petzold 1992][p. 57].

[25]In 1949 Edwin A. Goldberg applied for a patent that described a practical scheme of eliminating the drift effects in DC amplifiers. It was this invention that made the following rapid development of analog computers possible (see section 4.1.2).

[26]The symbols in the schematic differ from those used today and are those used in Hoelzer's dissertation.

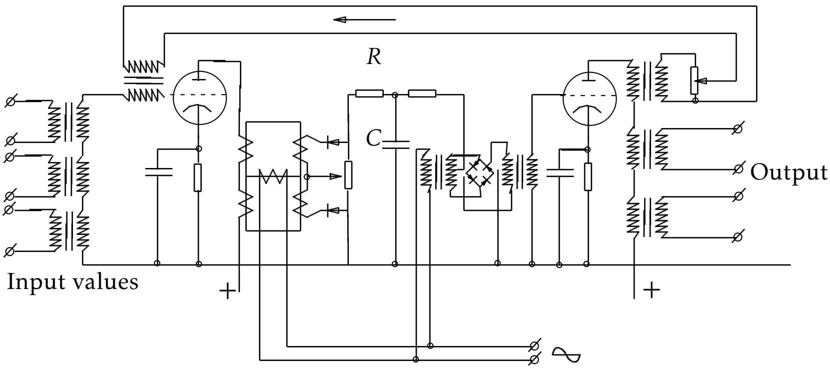


Figure 3.10: Schematic of Hoelzer's integrator circuit (cf. [Hoelzer 1946][fig. I, 11])

signal. Finally the differentiator employs a simple low pass filter consisting of R_{gl} and C_{gl}[27] in order to suppress excessive noise.[28]

The integrator circuit differs from this circuit only in a few respects: The low pass filter consisting of R_{gl} and C_{gl} is no longer necessary since the operation of integration itself acts as a low pass filter. In addition to that the output stage is fed directly with the modulated voltage across the capacitor plates, so the resistor r is also unnecessary. Figure 3.10 shows the schematic of the integrator used in Hoelzer's analog computer.

The analog computer also contained additional computing elements such as multipliers, dividers and square root function generators.[29] Especially interesting is a servo circuit suggested by Hoelzer to implement division and other operations since this device is a direct forerunner of the later servo function generators and multipliers described in sections 4.5.1 and 4.6.1. Figure 3.11 shows this computing device which essentially forms a bridge circuit: Central element is a servo motor driving two potentiometers R_1 and R_{out}. R_1, the fixed resistor R_2, and the two input transformers form the actual bridge circuit. Every imbalance of this bridge will yield an error voltage that is amplified by the Amplifier A which in turn drives the servo motor which will then cancel out the imbalance by readjusting R_1 accordingly.

Since R_1 and R_{out} are driven in tandem this self-adjusting bridge yields

$$y = \frac{e_1(t)}{e_2(t)}$$

at the output of the voltage divider R_{out}.[30]

[27]The abbreviation *gl* denotes *glätten*, German for smoothing.

[28]Differentiators are normally avoided in electronic analog computers since they obviously tend to increase the noise of a signal excessively. Hoelzer alleviated this problem by the inclusion of the low pass filter.

[29]Summing devices were not necessary since the modulated signals used to transmit values from one unit to another could simply be fed into the multiple inputs of the computing elements. The transformers in series connection (see figs. 3.9 and 3.10) then performed the summing operation.

[30]This voltage divider needs an auxiliary supply e_{aux}.

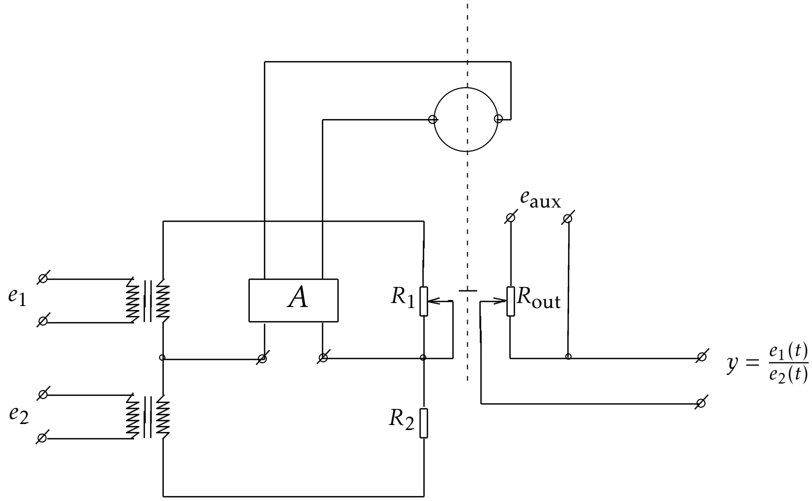


Figure 3.11: Simplified schematic of HOELZER's self-adjusting bridge (cf. [HOELZER 1946][fig. II, 11])

However, this divider circuit was not built during war time since the necessary servo motor could not be obtained. Instead HOELZER implemented a divider based on the solution of the differential equation

$$\frac{1}{A}\dot{y} + ax = b$$

where A denotes the gain of the amplifier used which should be as high as possible to minimize errors. The multiplication ax was implemented by two ring modulators connected in series.

Figure 3.12 shows an improved integrator circuit that was proposed by HOELZER. Unfortunately this circuit, too, was not built during the war years since integrators based on the design shown in figure 3.10 were already built and used successfully. The novel idea of this proposed circuit was to simplify the integrator by using only a single AC coupled amplifier instead of two such stages. Since the integration capacitor needs a DC current to be charged, HOELZER's idea was to place the capacitor at one end of a ring modulator while the other side of the ring modulator was used to close a feedback loop over the amplifier. This idea anticipated the basic integrator design that would dominate analog electronic analog computers for the following decades.

The analog computer as it was built in 1941 and the following years is shown in figure 3.13. The three top frames contain the computing elements such as integrators, differentiators and the like. Below is an electromechanical function generator based on multiple cams mounted on a central shaft that is driven by a motor. The bottom compartment contains the power supplies.

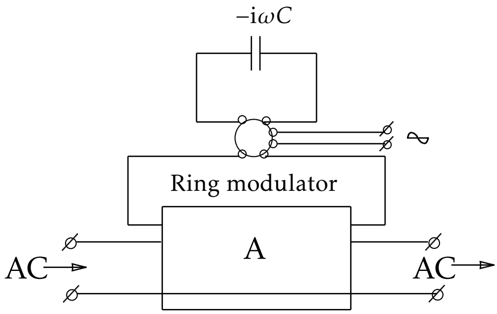


Figure 3.12: An improved integrator circuit (cf. [Hoelzer 1946][fig. II, 11])

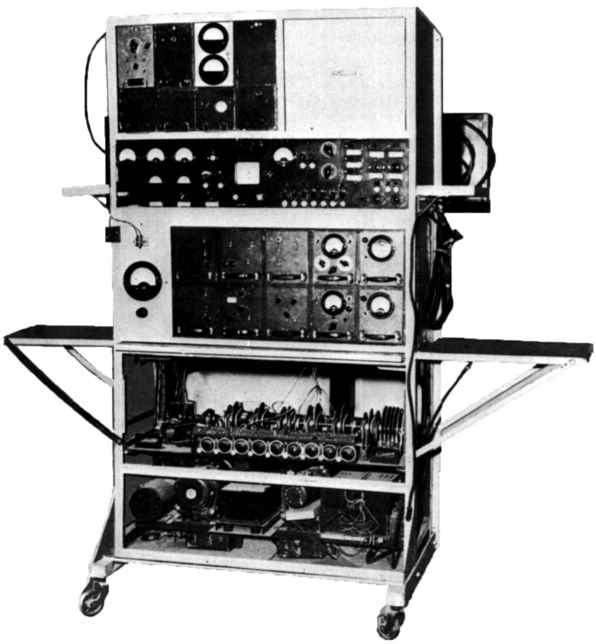


Figure 3.13: Hoelzer's analog computer as it was found after World War II (source: NASA, Marshall Space Flight Center)

Two of these analog computers were finally built,[31] one of which was brought to the United States where it was used until about 1955.[32] It was applied e. g. during the development of the *Hermes* rocket. Based on this machine an improved version was developed under von Braun in 1950 that was used for about ten years and proved to be a valuable tool in the development of the *Redstone* and *Jupiter* rockets as well as for the design of the first satellite of the United States, *Explorer I*.[33]

[31] In 1993 Helmut Hoelzer started building a replica of this first analog computer which was completed in 1995. This machine is now part of the collection of the *Deutsches Technikmuseum Berlin*.

[32] This machine is the one shown in figure 3.13.

[33] See [Tomayko 2000][p. 236].

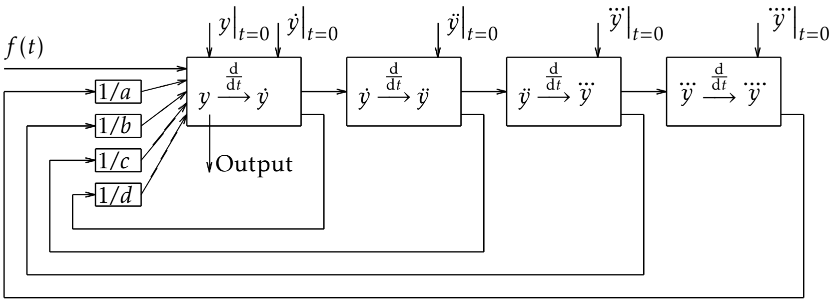


Figure 3.14: Solution of a differential equation of fourth degree $\ddddot{y} + a\dddot{y} + b\ddot{y} + c\dot{y} + dy = f(t)$ with a feedback circuit employing four differentiators (cf. [Hoelzer 1946][fig. II, 2])

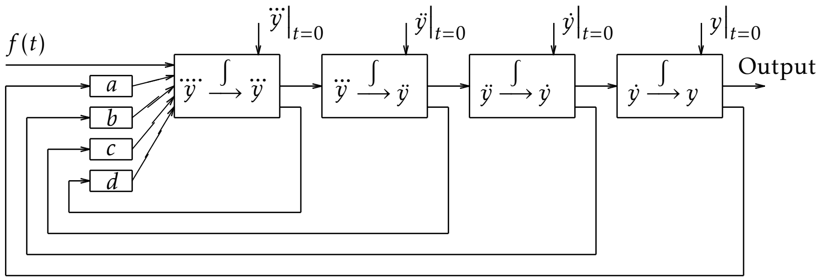


Figure 3.15: Solution of a differential equation of fourth degree $\ddddot{y} + a\dddot{y} + b\ddot{y} + c\dot{y} + dy = f(t)$ with a feedback circuit employing four integrators (cf. [Hoelzer 1946][fig. II, 1])

Hoelzer gives a good example how this first analog computer was programmed in his 1946 dissertation.[34] Three different approaches for solving the following differential equation of fourth degree are outlined in detail:

$$\ddddot{y} + a\dddot{y} + b\ddot{y} + c\dot{y} + dy = f(t) \tag{3.1}$$

Figure 3.14 shows a straightforward approach using four differentiating elements in series to derive $\dot{y}$, $\ddot{y}$, $\dddot{y}$ and $\ddddot{y}$ from y. The required y is generated by adding these derivatives (multiplied by proper coefficients) and a function input $f(t)$.

As straightforward as this solution is, it has a severe drawback: The differentiator chain will increase noise in the variables used excessively, even though there are low pass filters in each differentiator. Since Hoelzer was aware of that fact, he proposed a second approach using four integrators which is shown in figure 3.15. This setup is the same as any later analog computer programmer would have proposed it. In fact, it is based on Kelvin's feedback idea which is described in more detail in section 7.2.

Unfortunately, in the early 1940s this solution was not without problems. It did solve the noise problem but introduced yet another problem owed to the non-ideal characteristics of the computing elements, namely the capacitors: Every integrator introduced some amount of drift due to leakage in its integration capacitor, thus this circuit was also suboptimal to solve the proposed fourth degree differential equation.

[34] See [Hoelzer 1946].

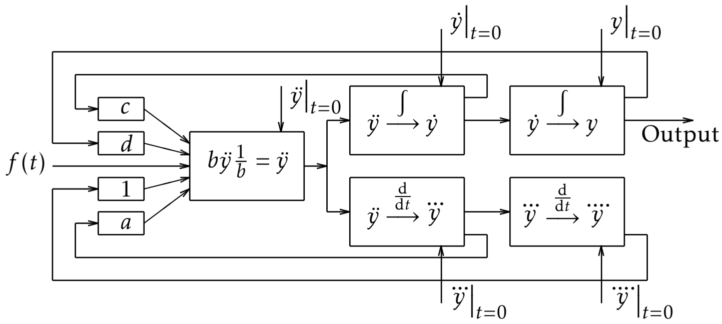


Figure 3.16: Solution of a differential equation of fourth degree $\ddddot{y} + a\dddot{y} + b\ddot{y} + c\dot{y} + dy = f(t)$ with a feedback circuit employing two differentiators and two integrators (cf. [Hoelzer 1946][fig. II, 3])

Therefore Hoelzer suggested a third setup shown in figure 3.16. This approach uses two differentiators as well as two integrators and thus minimized the negative effects of each type of computing element.[35]

3.2 George A. Philbrick's Polyphemus

Another development that would finally result in a special purpose all electronic analog computer was begun in 1938 by George A. Philbrick who worked for Foxboro and wrote a proposal describing a novel *simulator*[36] for process control systems. The main goals of the development were stated as follows:[37]

> *"We attempt to describe a method for the rapid and easy solution of problems which arise in connection with the technical study of process control. Also included is an electrically operated unit capable of disclosing the behavior of controlled systems as influenced by their various physical characteristics."*

Philbrick started with simple analogies like a capacitor modeling a tank, a resistor representing a valve restricting flow of a medium etc. Figure 3.17 shows an example used by Philbrick as early as 1938. Depicted are three tanks T_1, T_2 and T_3 connected by pipes P_1 and P_2 which contain valves V_1 and V_2. These series connected tanks can be filled via P_{in} and drained by P_{out}. The latter pipe also contains a valve V_{out}.

He then proposed to model this three tank system by means of passive electronic components like resistors representing the valves and capacitors representing the tanks. Figure 3.18 shows a circuit equivalent to the setup in figure 3.17.

[35] Later analog computers avoided differentiators wherever possible but this required some substantial technological advances, most notably very high gain drift stabilized DC amplifiers and high precision computing capacitors.

[36] The term simulator is defined by [Holst 1982][p. 144] as follows: *"A simulator is a fixed (to a large degree) structure embodying one unique model. [...] A simulator's purpose is specific: to provide the accurate realization of its model for various parameters, stimuli, and operator interactions of interest to its users."*

[37] Cf. [Holst 1982][p. 143].

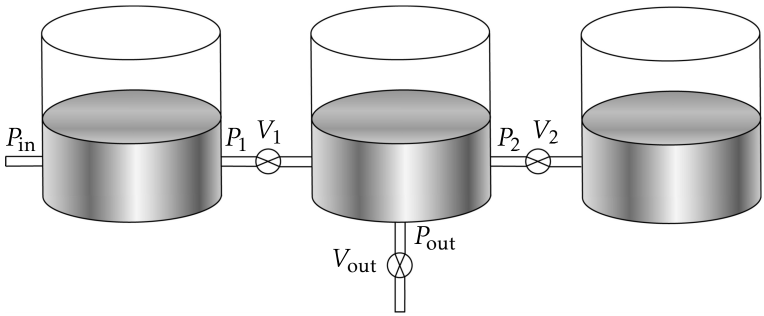


Figure 3.17: Three interconnected tanks (cf. [Holst 1982][p. 146])

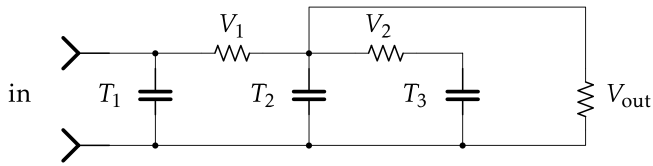


Figure 3.18: Simulation circuit for the three interconnected tanks of figure 3.17 (cf. [Holst 1982][p. 146])

Of course more complex simulations would require some kind of amplifier so Philbrick was faced with pretty much the same problem as Hoelzer in Germany: No such devices existed in the late 1930s and the current state of the art DC amplifiers showed excessive drift making precise simulations impossible. Thus Philbrick decided to use AC amplifiers in some setups like Hoelzer did[38] but omitted the modulator and synchronous demodulator circuits since his simulator was intended to be used in a mode of operation that became known as *repetitive operation*. In this mode a simulation is run automatically over and over again in quick succession. This makes it possible to display a steady figure on an oscilloscope that will reflect the behavior of the system being simulated as its parameters are modified manually on a control panel.

Based on these ideas Philbrick set out to develop a process simulator that was eventually called *Polyphemus* due to its single oscilloscope that was mounted in the top section of a 19-inch rack holding the simulator's components. With some imagination this system looked like Polyphemus, the Cyclops of the Odyssey, thus the name chosen.

Although the basic structure of this simulator was fixed it could be adapted to different simulation tasks within limits by means of a removable cardboard front plate. This plate contained a schematic diagram showing the structure of the process to be simulated and annotations for the various control elements of the simulator like potentiometers etc. Figure 3.19 shows Polyphemus with a faceplate representing a system consisting of two liquid baths with stirrers and steam and cold water inflows.[39] Other face plates depicted pneumatic controllers etc.

[38] Cf. [Holst 1982][p. 149].
[39] Cf. [Holst 1982][p. 152].

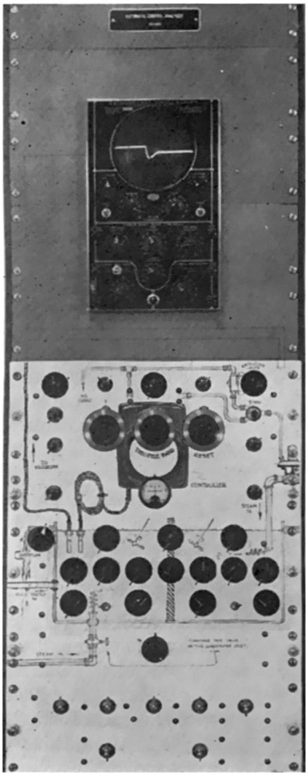


Figure 3.19: George A. Philbrick's Polyphemus (see [Philbrick 1948][p. 108])

Interestingly, and rather unexpectedly Polyphemus was not only used as a research tool as originally envisioned. After some time, it was mainly employed as a training simulator and as an aid for sales people that could easily demonstrate the behavior of a proposed process controller to prospective customers.[40] In retrospective Polyphemus has been described as follows:[41]

> *"In those days, an electronic analog machine was a pioneering venture. The Kelvin-Bush differential analyzers were mechanical, expensive, and very large. Appropriate electronic techniques, if they existed, were not available. Nevertheless, it was evident that only this medium offered the required flexibility and speed, not to mention economy."*

This world's first all-electronic process simulator is now part of the collection of the *Smithsonian* – its successor, featuring two oscilloscopes and thus no longer looking like a Cyclops, was in active use at Foxboro well into the 1980s.[42]

[40] See [Holst 1982][pp. 153 f.].
[41] See [GAP/R Evolution].
[42] Cf. [Holst 1982][pp. 155 f.].

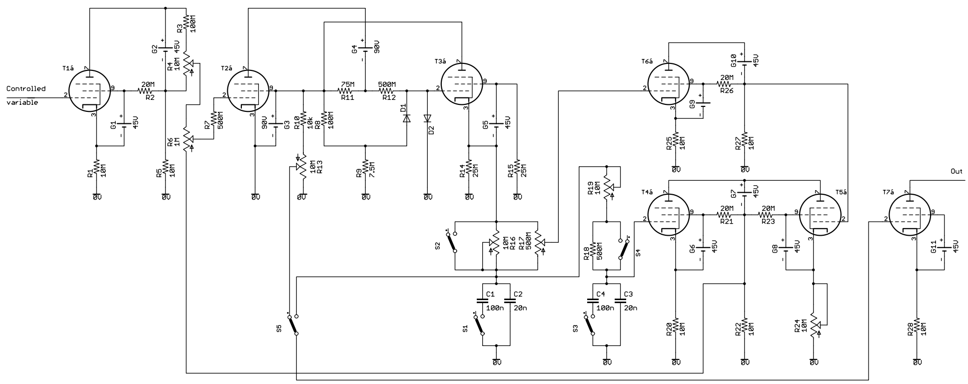


Figure 3.20: Seven-stage amplifier from George A. Philbrick's laboratory notebook, March 11, 1940 (see [Holst 1982][p. 154])

To get an impression of the complexity of the circuits developed by George A. Philbrick for his numerous process simulators, figure 3.20 exemplarily shows the schematic diagram of a seven stage circuit that models a process controller – noteworthy are the two diodes for implementing non-linear responses as well as the DC coupling between the various stages.

In 1946 George A. Philbrick founded a company named *GAP/R*, short for *George A. Philbrick Researches* that developed and sold products ranging from operational amplifiers[43] to complete analog computers and simulators which owed much to his early experiences with Polyphemus and its successors. One of the most famous products was the first commercially available operational amplifier, the *K2-W*[44] which shaped a whole industry. In 1966 GAP/R merged with *Teledyne* to form *Teledyne-Philbrick*.

3.3 Electronic fire control systems

The decline of mechanical fire control systems as those described in section 2.7 already begun before World War II broke out.[45] This was not due to insufficient accuracy or precision or even slow response times – in fact, the military feared that in the case of a war the sheer complexity of these machines would prevent their production in sufficient quantities. The main obstacles were that the necessary parts and resources would be short in supply in such a situation and the necessary highly skilled laborers would be unavailable due to their military service.

To make things even worse the rapid development of radar systems rendered the classic fire control systems quite obsolete since they required a direct electronic approach for coupling the radar system with the fire control computer.

[43] See section 4.1.
[44] See section 4.1.
[45] See [Clymer 1993][p. 30].

In 1940 David B. Parkinson who worked for the *Bell Telephone Laboratories*[46] and his coworkers were developing an *automatic level recorder*, a kind of a strip-chart recorder employing a logarithmic scale. To drive the recorder's pen a servo mechanism based on a feedback scheme was necessary which inspired Parkinson to think about using this technique for the automatic control of azimuth and elevation of a gun.[47] His supervisor, Clarence A. Lovell was quite enthusiastic about this idea and both started a thorough investigation of the necessary technology and mathematics.[48]

It was decided to use DC voltages to represent the various variables in a fire control calculation. Coordinates of the target and the gun were represented in a three-dimensional Cartesian coordinate system requiring computer circuits for implementing trigonometric functions etc. Target speed components would be generated by electronic differentiation of the position signals. Parkinson summarized the main features of the proposed fire control computer as follows:[49]

> *"It required (1) a means of solving equations electrically (potentiometers), (2) a means of deriving rate for prediction (an electrical differentiator), and (3) a means of moving the guns in response to firing solutions."*

Obviously such a fire control system needs computing elements to implement basic trigonometric functions and operations as well as differentiators, summers etc. The patent application [Lovell et al. 1946] states this explicitly:

> *"An important form of the invention is a device which, when supplied with electrical voltages proportional to two sides of a triangle, will set itself to indicate an angle of the triangle and to produce a voltage proportional to the other side of the triangle.*
> *Another form of the invention is a device which, when supplied with voltages proportional to the rectangular coordinates of a point, will set itself to indicate the polar coordinates of the point."*

The basic problem of aiming a gun is shown in figure 3.21. The Cartesian coordinates of a target T with respect to the position D of the gun are given as x, y and z. Based on these the azimuth and elevation angles α and ε have to be computed.

Lovell, Parkinson and Weber developed a number of basic computing elements to implement functions like this. Figure 3.22 shows the structure of a trigonometric function generator used in their electronic fire control computer. The motor on the far left is running with constant speed and drives two bevel gears 3 and 4 which will run in clockwise and counterclockwise direction respectively. Two electromagnetic clutches, 5 and 6, can be activated by a control signal thus controlling which gear will drive the vertical shaft that in turn drives the two potentiometers shown at the top left of the figure.

[46] *BTL* for short.

[47] In [Fagen (ed.) 1978][p. 135] Parkinson describes a lucid dream that led him to this development.

[48] It is remarkable that *"[n]either* Parkinson *nor* Lovell *had any experience in fire control; they did not even know of the existence of mechanical gun directors"* as [Fagen (ed.) 1978][p. 137] states.

[49] See [Mindell 1995][p. 73].

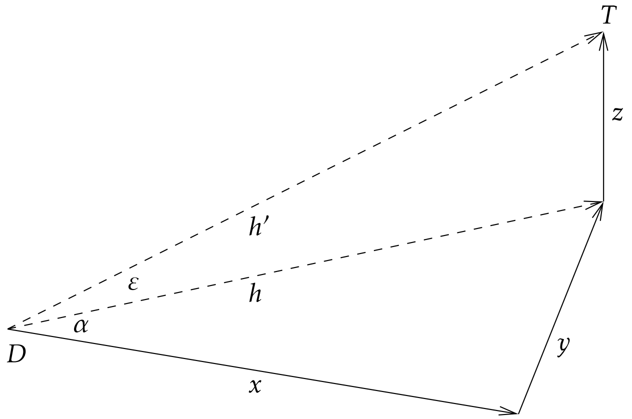


Figure 3.21: Basic fire control task

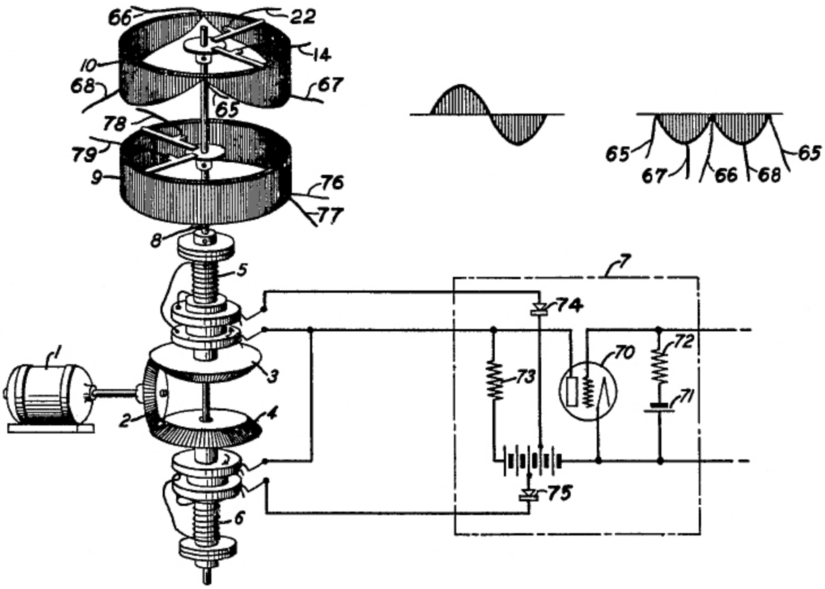


Figure 3.22: Sine/cosine generator used in the T-10 fire control computer ([Lovell et al. 1946][fig. 1/2])

The control signals for these two clutches are generated by a comparator circuit shown in the box denoted by 7. This circuit has three output conditions: Neither clutch or only the upper/lower clutch will be engaged. These states are controlled by the input signal connected to the appropriately biased grid of the triode 70.

It is to be noted that the lower of the two potentiometers shown is a linear potentiometer yielding a voltage varying linearly with the angular position of its wiper when connected as a voltage divider, while the upper potentiometer is specially wound to deliver a voltage that corresponds to the desired trigonometric function.

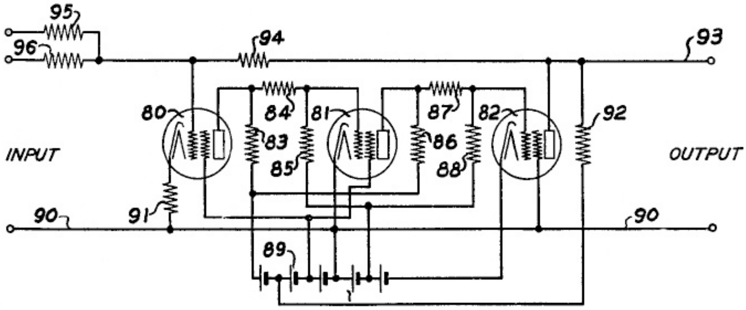


Figure 3.23: Operational amplifier used in the T-10 fire control computer ([Lovell et al. 1946][fig. 5])

The combination of this circuit with an amplifier finally yields a servo system that can be used to automatically compute $\sin(\alpha)$ or the like given an angle α. Therefore Lovell, Parkinson and Weber developed an amplifier as shown in figure 3.23 which is already reminiscent of later so-called *operational amplifiers*.[50]

This amplifier acts as an inverting summer[51] with two inputs connected to the resistors 95 and 96 and a feedback resistor 94. Using this amplifier and the device shown in figure 3.22, a servo system can now be implemented by connecting the output of this summer with the input of the comparator controlling the two clutches. The inputs of the summer are connected to the output of the linear potentiometer and a circuit yielding α. Whenever α deviates from the angular position of the potentiometers' wipers the summer will generate a positive or negative output signal thus forcing the comparator to engage either the upper or the lower clutch to minimize the error between the wipers' position and the value α.

Lovell, Parkinson and Weber realized that the computing elements they proposed were not limited to fire control but could be used to solve a variety of problems as the following quote from [Mindell 1995][p. 74] shows:

> *"A digression from the principal subject is made to comment that the use of servo mechanisms to solve simultaneous systems of equations is feasible and, in a large number of cases, practicable. This fact may lead to the application of this type of mechanism to the solution of many types of problems dissociated from the one in question."*

The actual circuit developed for the implementation of a coordinate transformation relies on the observation that

$$x = h\cos(\alpha) \tag{3.2}$$

$$y = h\sin(\alpha) \tag{3.3}$$

given the triangle x, y, h in figure 3.21. (3.2) and (3.3) can now be expanded to

$$x\sin(\alpha) = h\cos(\alpha)\sin(\alpha)$$
$$y\cos(\alpha) = h\sin(\alpha)\cos(\alpha)$$

[50] See section 4.1.
[51] See section 4.2.

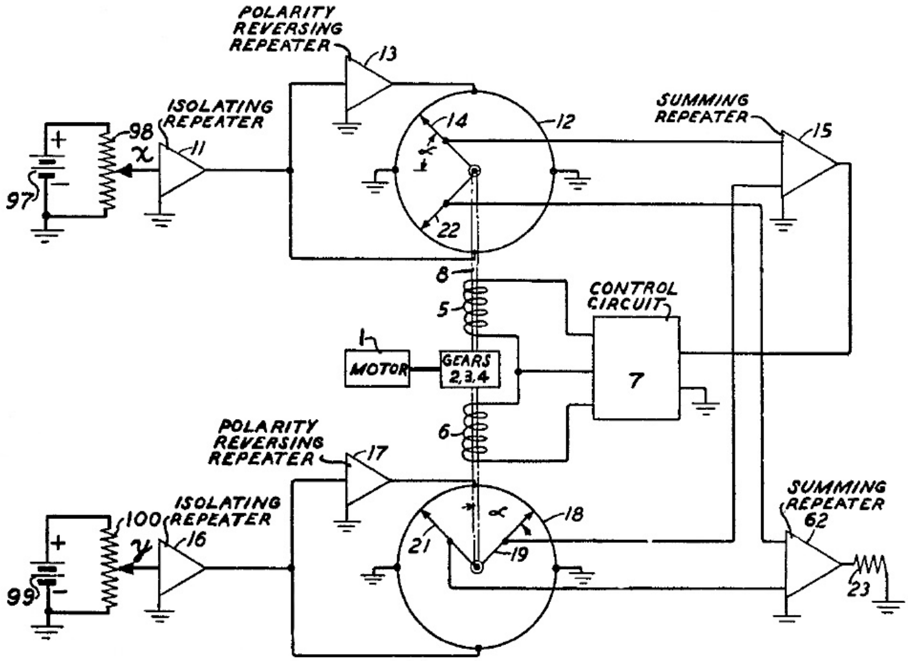


Figure 3.24: Coordinate resolver used in the T-10 fire control computer ([Lovell et al. 1946][fig. 4])

yielding $x\sin(\alpha) - y\cos(\alpha) = 0$ which is used to control the basic servo circuit of the coordinate transformer shown in figure 3.24.

This circuit has two inputs x and y which supply two specially wound potentiometers 12 and 18 with four connections each. The two *polarity reversing repeater* devices are in fact summers with only one input each and gain 1. Both potentiometers are connected to $-x$, ground (representing the value 0), x, 0 and $-y$, 0, y, 0 respectively. Each potentiometer has two wipers mounted at an angle of $\pi/2$ driven by a common shaft that can be connected to one of the two bevel gears driven by a motor.

The potentiometers are wound in a way that their resistance represents a sine/cosine function (due to the two wipers each potentiometer features sine as well as cosine functions are generated simultaneously). Thus the wipers 14 and 22 yield $x\sin(\alpha)$ and $x\cos(\alpha)$ while the lower potentiometer generates $-y\cos(\alpha)$ and $y\sin(\alpha)$ at its wipers 19 and 21 respectively. The summing *repeater* 15 now computes the value

$$c = x\sin(\alpha) - y\cos(\alpha) \tag{3.4}$$

which is used to control the comparator 7 driving the two clutches 5 and 6 thus closing the servo loop. As long as c equals zero, both clutches are disengaged. As soon as c deviates from zero, one of the clutches will be activated, thus connecting the central shaft driving the two potentiometers to the motor which will in turn minimize the error term c until it reaches zero again. Thus the angular position of the wipers is α which can be used to control the gun's azimuth.

The second summing repeater, 62, computes $h = x\cos(\alpha) + y\sin(\alpha)$ which is the horizontal range to the target. Using this value and z as inputs for a second, identical circuit, eventually yields h' and the elevation angle ε represented by its potentiometer shaft position. It was suggested to make the motors driving these two sub-circuits strong enough so that they could actually drive the gun's mounting to point it to the target T.

The resulting fire control computer *T-10* consisted of four such servo circuits as well as 30 summing amplifiers and five power supplies. Since the voltages representing the Cartesian coordinates of the target with respect to the gun's location often changed rather slowly the amplifiers had to be DC coupled amplifiers thus introducing drift effects into the calculations. Assuming that amplifier 15 of figure 3.24 exhibits a drift error voltage e this will cause the servo circuit to try making

$$c = x\sin(\alpha) - y\cos(\alpha) + e$$

zero instead of the term (3.4). This, in turn, results in a non-negligible error regarding α, ε, h and h'. Accordingly the T-10 computer featured additional inputs to its summing amplifiers which were connected to potentiometers which in turn had to be adjusted manually to minimize the drift voltage e of each amplifier. In addition to that nuisance, the T-10 relied on the computation of derivatives thus introducing noise into the control signals used to drive the servo circuits. This required the implementation of filter networks approximating a least squares behavior, further complicating its circuitry.

Despite of these shortcomings, the T-10 which was eventually produced under the designation *M-9* and coupled directly to a *SCR-584* radar system, proved to be a practical replacement for the earlier mechanical fire control computers. Due to its easier manufacturing it was decided on February 12th, 1942 to stop all developments of mechanical fire control systems and to concentrate on all-electronic devices in the future.[52] The hopes were high regarding the M-9:[53]

> *"The M-9 Director, electrically operated, is, we feel in Ordnance, one of the greatest advances in the art of fire control made during this war, and we anticipate from the M-9 Director very great things as the war goes on."*

These hopes were not in vain as it turned out – the M-9 was deployed in large numbers to Great Britain to shoot down *V1* German flying bombs. Initially 10 to 34 percent of the incoming flying bombs could be downed automatically and sometimes even higher rates were achieved:[54]

> *"In a single week in August, the Germans launched 91 V1's from the Antwerp area, and heavy guns controlled by M-9's destroyed 89 of them."*

[52] See [Higgins et al. 1982][p. 225].
[53] See [Fagen (ed.) 1978][pp. X f.].
[54] Cf. [Fagen (ed.) 1978][p. 148].

Later developments such as the experimental fire control computer *T-15* took much more parameters and effects into account like drift of the shell during flight, windage correction, muzzle velocity, air density etc.[55] The T-15 performed all computations in polar coordinates thus eliminating the coordinate system transformations otherwise necessary.[56]

The technologies developed for these fire control computers formed the basis for following developments like a trainer for aerial gunners[57] that already used continuously operating servo systems in contrast to the controlled clutches of the T-10. Even more complex was the *Flight Training Apparatus* developed in 1948 by Dehmel[58] that even implemented the basic operation of integration by means of a servo system controlling a potentiometer, the angular wiper position of which corresponds to the integral of the function applied to the servo circuit etc.

3.4 MIT

This last section describes a development started in the fall of 1945 by A. B. Macnee who set out to develop an all-electronic analog computer at the MIT since[59]

> *"[...] it was felt that there was considerable need for a differential analyzer of somewhat different characteristics from any then in existence or under development. There appeared to be the need for a machine having the following characteristics: (a) moderate accuracy, of perhaps 1 to 10 per cent, (b) much lower cost than existing differential analyzers, (c) the ability to handle every type of ordinary differential equation, (d) high speed of operation, (e) above all, extreme flexibility in order to permit the rapid investigation of wide ranges of equation parameters and initial conditions."*

These goals summarize quite well the main advantages of an all-electronic approach regarding analog computers. Especially the high-speed operation[60] mentioned is an important feature since it allows to explore the behavior of systems modeled on the analog computer in a highly interactive way and, in many cases, even faster than real-time. The operator can change the parameters of a simulation and immediately see the effects these changes have on the system.

This particular system developed by Macnee consisted of integrators, summers, a high-speed function generator[61] and a high-speed multiplier.[62] One of the novel ideas employed in this computer was that it ran repetitively with a repetition rate of 60 Hz

[55] [Boghosian et al. 1950] gives an impression of the complexity of these fire control computers.
[56] Another system that performed all calculations in Cartesian coordinates is described in [Bedford et al. 1952].
[57] See [Dehmel 1949].
[58] See [Dehmel 1954].
[59] See [Macnee 1948][p. 1].
[60] Even high compared to our modern day memory programmed digital computers in some cases.
[61] This function generator was based on optical feedback as described in section 4.5.3.
[62] See section 4.6.2.

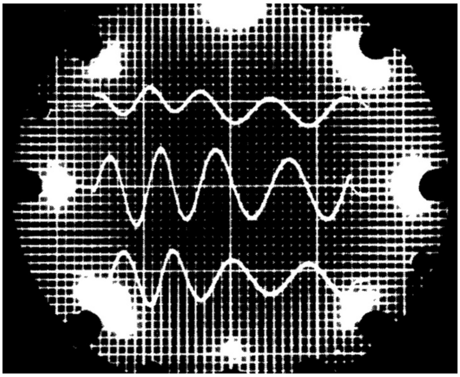


Figure 3.25: A particular solution of the Mathieu equation (3.5) obtained with Macnee's all-electronic analog computer, 1948

thus allowing a standing picture to be displayed on an oscilloscope's tube. Apart from the fact that this greatly facilitated the above mentioned interactive operation where an operator can change various parameters of a simulation, this has the additional beneficial effect that AC coupled amplifiers can be used in the computer thus avoiding drift problems that would be caused by DC amplifiers. So this computer, too, uses AC amplifiers like the machines of Helmut Hoelzer, but without the complicated modulation and demodulation circuitry.

As an example of the capabilities of this early machine, figure 3.25 shows a particular solution for the Mathieu *equation*

$$\ddot{I} + \omega_0^2 (1 + \varepsilon \cos(\omega_m t)) = 0 \tag{3.5}$$

obtained by this electronic analog computer. The three curves shown on the oscilloscope's display are I, $-\dot{I}$ and $\ddot{I}$.

No other technology apart from an all-electronic differential analyzer would have allowed the solution of such an equation in only 1/60 of a second. Machines like this made it ultimately clear that the days of mechanical and electromechanical analog computers were gone and the future belonged to high-speed electronic devices. These machines turned out to be crucial elements for the technological progress of the time frame 1950 to the mid 1970s. The following chapter will now describe the basic computing elements employed in such analog electronic analog computers.

4 Basic computing elements

The following chapters give an overview over the most important computing elements of a typical analog electronic analog computer. Some functions, like multiplication, can be implemented quite differently. Although often only one implementation variant prevailed, other technologies will be also presented to show the wide range of possible implementations.

Such an analog computer relies on voltages for transmitting values between the various computing elements that comprise a computer setup suitable for solving a particular problem. These voltages can usually take on values between ±100 V, which is the standard case for most large vacuum tube based analog computers, or ±10 V which is standard for later transistorized and integrated circuit based machines.[1] To simplify things it is common to talk about normalized values for variables regardless of their actual value for a given computer system. The bounds for these normalized values are called *machine units* and correspond to the actual minimum and maximum voltages for any given machine. Thus a variable which is represented by +50 V on a ±100 V machine is said to have the value 0.5. The same value would describe a variable represented by 5 V on a ±10 V transistorized system. All calculations are thus with respect to these machine units of ±1 and never with respect to the actual range of values determined by a particular analog computer.

So all computing elements described in the following sections will work on normalized values in the range of ±1 machine unit.

4.1 Operational amplifiers

The central element of every analog electronic analog computer is the so-called *operational amplifier* which forms the heart of most computing circuits such as summers, integrators etc. This central role of the operational amplifier led to the following remark by [Truitt et al. 1964][p. 2-58]: "*The Operational Amplifier is the King of Analog Computing Components*". The necessity of having high gain amplifiers capable of working on DC voltages with negligible error voltages caused by drift effects and the like was obvious from the very first developments of analog electronic analog computers on. These first machines were plagued by the deficiencies of early vacuum tube based amplifiers as can be seen in the developments of Hoelzer, Philbrick and Macnee. While Hoelzer successfully employed complex modulator and demodulator circuits to make AC amplifiers useful in DC applications, others, like Macnee, abandoned the

[1] These two ranges of values are not the only ones used – some tube based circuits work with values in the range of ±50 V and some transistorized analog computers even allow values between +100 and −100 V.

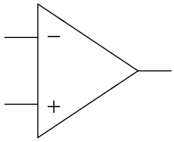


Figure 4.1: Symbol of an operational amplifier

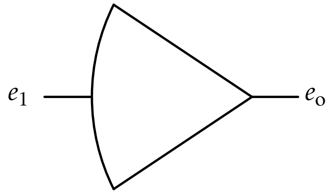


Figure 4.2: Symbol of an *open* or *uncommitted* amplifier

idea of DC amplification all together and more or less forbid DC voltages by enforcing a repetitive very high-speed operation of their analog computers.

It was not until 1949 that some feasible drift correction technique was developed[2] by Edwin A. Goldberg that made the implementation of high precision operational amplifiers with extremely high gain and very low drift possible. This development led to a revolution in the development of analog electronic analog computers.[3]

The term *operational amplifier* was coined by John Ragazzini in 1947 and defined as follows:[4]

> *"As an amplifier so connected can perform the mathematical operations of arithmetic and calculus on the voltages applied to its input, it is hereafter termed an 'Operational Amplifier'."*

Figure 4.1 shows the symbol used today to denote an operational amplifier in a schematic. Typically such an amplifier has two inputs, one inverting and one non-inverting, and one output that represents the amplified sum of the values applied to the inputs. In typical analog computers, the non-inverting input is not available for any connection in a computer setup and is either connected to ground – representing the value zero – or used for drift compensation. Thus, differing from the standard symbol used in today's electronics – an *open* operational amplifier, that is an amplifier that has no elements connecting its output to its input, is normally denoted by the symbol shown in figure 4.2 in an analog computer setup diagram. The input is always inverting.

In the early 20th century amplifiers were normally operated in a straightforward mode of operation. These amplifiers were quite unstable and difficult to use since even the slightest changes regarding their components resulted in grossly differing behavior of the amplifier as a whole. Then, in 1927, Harold Stephen Black[5] invented the so-called *negative-feedback amplifier* that should turn out to become one of the most influential basic circuits at all. A detailed account of Black's work is given in

[2] See section 4.1.2.

[3] A comprehensive account of the history of operational amplifiers can be found in [Jung 2006].

[4] See [Ragazzini et al. 1948]. It should be noted that the developments on which this paper is based were mostly done by Loebe Julie who was working under the auspices of George A. Philbrick at Bell Labs (cf. [Jung 2006][p. 779]).

[5] 04/14/1898–12/11/1983

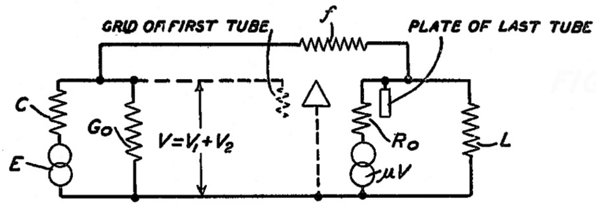


Figure 4.3: The basic principle of negative-feedback according to [Black 1937]

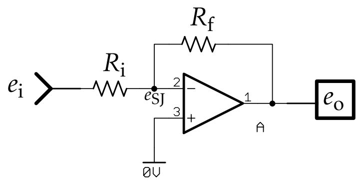


Figure 4.4: Operational amplifier with negative feedback

[Kline 1993] and [Mindell 2000]. Black was elected as a member of the "Electrical Engineering Hall of Fame" for his seminal work on amplifiers ([Britain 2011]). The idea of negative-feedback was so novel and unusual that it took nine years from Black's patent application to its issuance (see [Jung 2006][p. 767]). It should be noted that Black was not the only developer working on negative-feedback schemes – others included Paul Voigt (in memoriam [Klipsch 1981]), Alan Blumleini and B. D. H. Tellegen (cf. [Jung 2006][p. 767]).

Back then, Black was working for the Bell telephone laboratories and his negative-feedback scheme that is taken as granted today allowed the Bell telephone system to successfully employ carrier telephony which was until then hampered by poor stability and high distortion of existing amplifiers. Amplifiers without negative-feedback were prone to errors caused by varying supply voltages, aging of circuit elements (especially the tubes) etc.

Figure 4.3 shows the basic concept of Black's invention: An amplifier consisting of an odd number of inverting stages (thus yielding an overall inverting amplifier) is fed with two signals that are applied to its inverting input: The desired signal E itself that is connected via a series resistor C and a feedback signal that originates at the output of the amplifier and is connected to the input through a feedback resistor f. Figure 4.4 shows this principle using today's standard symbols.

The idea of feeding back some amount of the output signal of an inverting amplifier to its input allows to trade gain for stability as the following consideration shows: With A denoting the gain of the amplifier and e_SJ representing the voltage at the summing junction at its inverting input, as in figure 4.4, its output voltage is

$$e_\text{o} = -Ae_\text{SJ}$$

yielding

$$e_\text{SJ} = -\frac{e_\text{o}}{A}. \tag{4.1}$$

The currents at the summing junction are the following:

$$i_\mathrm{i} = \frac{e_\mathrm{i}}{R_\mathrm{i}} \qquad \text{(input current through } R_\mathrm{i}\text{)}$$

$$i_\mathrm{f} = \frac{e_\mathrm{o}}{R_\mathrm{f}} \qquad \text{(feedback current through } R_\mathrm{f}\text{)}$$

$$i_- \approx 10^{-9}\ \mathrm{A} \quad \text{(current flowing in the inverting input)}$$

Applying Kirchhoff's law we get

$$i_- = i_\mathrm{i} + i_\mathrm{f} = \frac{e_\mathrm{i} i - e_\mathrm{SJ}}{R_\mathrm{i}} + i\frac{e_\mathrm{o} - e_\mathrm{SJ}}{R_\mathrm{f}}. \tag{4.2}$$

Since i_- is negligible small, a few nA at most, equation (4.2) can be rearranged to

$$\frac{e_\mathrm{i} - e_\mathrm{SJ}}{R_\mathrm{i}} = -\frac{e_\mathrm{o} - e_\mathrm{SJ}}{R_\mathrm{f}}.$$

Replacing e_SJ according to (4.1) yields

$$\frac{e_\mathrm{i} + \frac{e_\mathrm{o}}{A}}{R_\mathrm{i}} = -\frac{e_\mathrm{o} + \frac{e_\mathrm{o}}{A}}{R_\mathrm{f}}.$$

Rearranging this equation results in

$$e_\mathrm{o}\left(\frac{1}{AR_\mathrm{i}} + \frac{1}{R_\mathrm{f}} + \frac{1}{AR_\mathrm{f}}\right) = -\frac{e_\mathrm{i}}{R_\mathrm{i}}$$

and thus

$$e_\mathrm{o} = \frac{-\frac{e_\mathrm{i}}{R_\mathrm{i}}}{\frac{R_\mathrm{f} + AR_\mathrm{i} + R_\mathrm{i}}{AR_\mathrm{i}R_\mathrm{f}}} = \frac{-\frac{e_\mathrm{i}}{R_\mathrm{i}}AR_\mathrm{i}R_\mathrm{f}}{R_\mathrm{f} + AR_\mathrm{i} + R_\mathrm{i}}.$$

Dividing by R_i and A and rearranging finally yields

$$e_\mathrm{o} = \frac{-\frac{R_\mathrm{f}}{R_\mathrm{i}}e_\mathrm{i}}{1 + \frac{1}{A}\left(\frac{R_\mathrm{f}}{R_\mathrm{i}} + 1\right)}. \tag{4.3}$$

Since the gain A is usually very large – some amplifiers used in analog computers reached values up to $A = 10^8$ and even $A = 10^9$ – and since the ratio $R_\mathrm{f}/R_\mathrm{i}$ is normally quite small, rarely exceeding 10, equation (4.3) can be further simplified to

$$e_\mathrm{o} = \frac{R_\mathrm{f}}{R_\mathrm{i}}e_\mathrm{i}, \tag{4.4}$$

so the behavior of a high gain amplifier with negligible input current at its inverting input is completely determined by the circuit elements in the feedback loop and at its input.

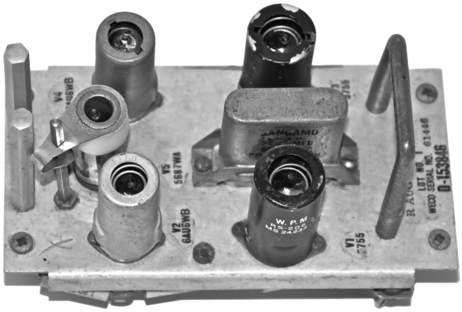


Figure 4.5: Top view of a typical double operational amplifier used in the Nike computer

Figure 4.6: Bottom view of a typical double operational amplifier used in the Nike computer

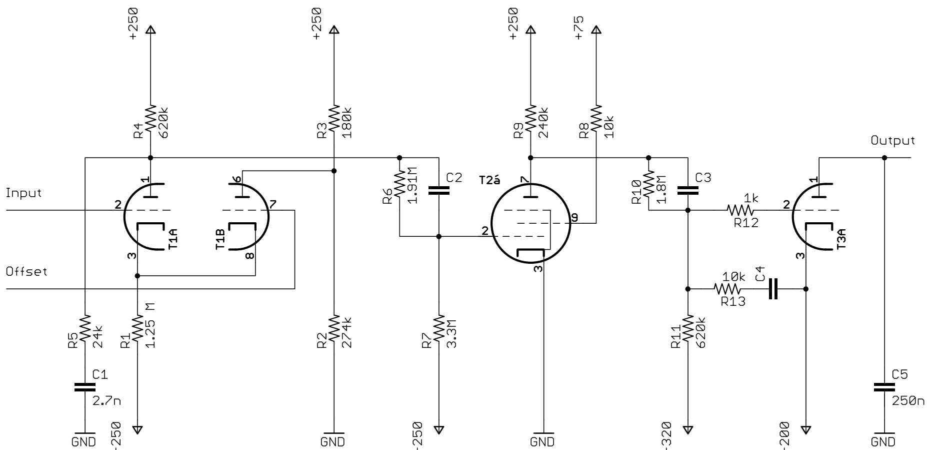


Figure 4.7: Operational amplifier used in the Nike computer

4.1.1 Early operational amplifiers

The following section shows some early operational amplifiers designs, historic references that might be of interest in that context are [Gray 1948] and [Chance et al. 1947].

An early dual channel operational amplifier is shown in figures 4.5 and 4.6. This amplifier was used in the ground based analog computer of a *Nike* surface to air missile system. The first version of this missile system was deployed from 1954 on into the early 1960s. It bears quite some resemblance to the amplifiers used in the M-9 fire control computer, in fact, the amplifiers used for Nike were identical to those used in the *M-33* fire control computer which is a successor of the M-9.[6]

The schematic diagram for a single channel of this operational amplifier is shown in figure 4.7. It features a now classic differential input stage consisting of the systems of a double triode. The interstage-coupling is done via the common cathode resistor R1.

[6] The schematic of an M-9 operational amplifier can be found in [Jung 2006][p. 781].

Figure 4.8: K2-W and K2-X operational amplifiers (GAP/R)

The non-inverting input is explicitly used here as an offset input for (automatic) drift compensation. The following stages are no longer differential and consist of a pentode section and a simple triode output stage.

Since the output stage is not of the push-pull type, an external pull-up resistor is necessary. This resistor would be connected between the output triode's anode and the +320 V supply. All in all this amplifier needs five different supply voltages of −320 V, ±250 V, −200 V and +75 V and has an output voltage swing of ±100 V although the full output swing can only be maintained up to about 100 Hz[7] which is enough to steer a surface to air missile.

A far simpler yet commercially very successful operational amplifier is shown on the left hand side of figure 4.8: The rather famous *K2-W* amplifier developed by GAP/R, the company George A. Philbrick founded after his early process simulator work at Foxboro.[8] The K2-W amplifier only requires two supply voltages of ±300 V and has an open-loop gain of about 15,000 with an output voltage range of ±50 V at a maximum output load of 1 mA. Rise time is about 2 μs, the input impedance is in the order of 100 MΩ corresponding to an input current of about 10 nA.

Figure 4.9 shows the schematic diagram of this pluggable operational amplifier that is based on two 12AX7 double triodes and appeared on the market in 1953. The first 12AX7 works as differential input stage with common cathode resistor[9]. This input stage is followed by a single triode inverter and a cathode follower output stage driving the output which is connected by three paralleled pull-down resistors to the negative

[7] See [Pease 2003].

[8] As [Pease 2003] notes, this is not the first "modern" operational amplifiers – there were earlier developments, but none could match the K2-W in terms of practicability and success.

[9] This configuration is called a *long-tailed pair*.

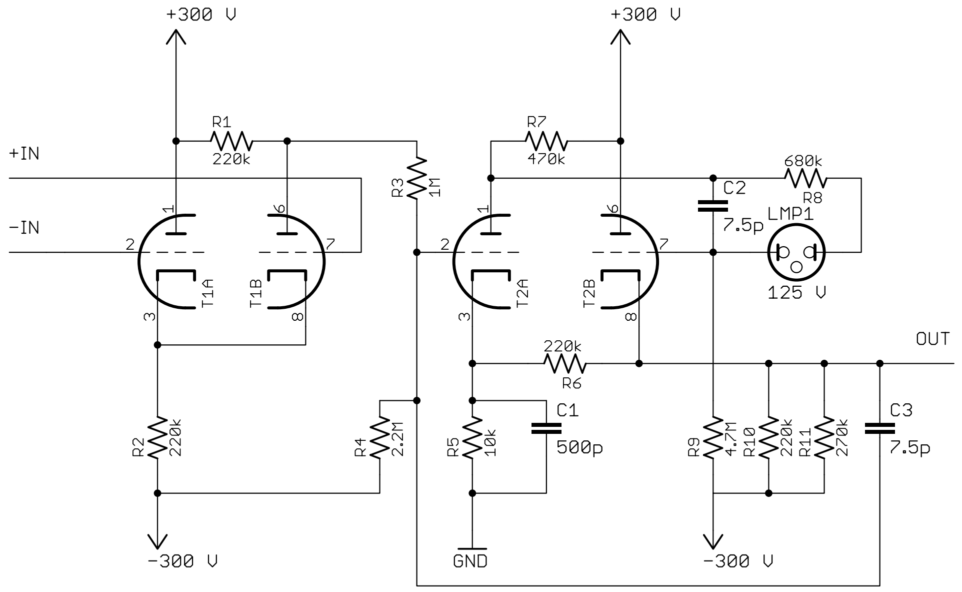


Figure 4.9: Schematic of the K2-W operational amplifier (cf. [GAP/R K2W])

supply voltage. An interesting detail is the Neon light bulb in series with a 680 kΩ resistor. This circuit acts as a level shifter.[10]

The amplifier shown in the right half of figure 4.8 is an improved version of the K2-W, called *K2-X*. This operational amplifier has a shorter output signal rise time than the K2-W and features an output voltage swing of ±100 V at 3 mA.[11]

4.1.2 Drift stabilization

All of these DC coupled operational amplifiers were mainly plagued by drift, i. e., small error signals that propagate through the various amplifier stages and show up with a reasonable amplitude at the output causing significant errors in computations. Recalling figure 4.4 and taking into account that the input current even of early vacuum tube based operational amplifiers is negligible small, it is clear that the voltage e_{SJ} at the summing junction is zero for a drift-free amplifier since the input current i_i flowing through R_i and the feedback current i_f through R_f cancel each other.

Any drift caused by effects internal to the amplifier will result in an output signal different from 0 V even if $e_i = 0$. Thus all drift effects in an amplifier can be modeled by a drift voltage e_d acting on the summing junction of the amplifier. If one could inject

[10] A Neon bulb is the 1950s equivalent to a Zener diode today. In early exemplars of the K2-W the level shifter consisted of a so-called *thyrite device* Today this would be called a *varistor*. These are made from silicon carbide and were not only used as level shifters in the early days of electronic analog computers but also as function generators in applications where precision was not of prime importance. Today such devices are used as current arrestors.

[11] The schematic of the K2-X operational amplifier can be found in [Russell 1962][p. 6-20].

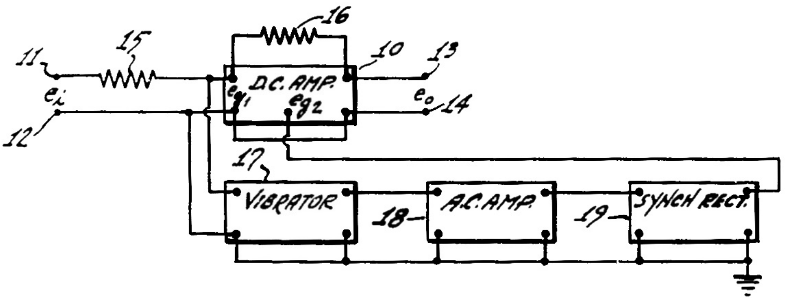


Figure 4.10: Drift stabilization scheme according to Edwin A. Goldberg and Jules Lehmann (cf. [Goldberg et al. 1954])

a commensurate voltage with reverse sign at the non-inverting input of the amplifier, it would cancel out the drift error.

This is the basic idea Edwin A. Goldberg describes in his 1949 patent.[12] If one could measure the error voltage due to drift, e_d at the summing junction and amplify it without introducing further errors with a drift-free amplifier, the resulting output signal could be used to practically cancel out the drift of the main amplifier. Figure 4.10 shows the setup proposed by Goldberg.

The DC amplifier in the upper half of the picture is the main amplifier whose drift is to be eliminated. For this purpose the error voltage at the input of this amplifier, the summing junction, is converted to an AC voltage by means of an electromechanical chopper relay called a *vibrator* in figure 4.10.[13] The output signal of this vibrator[14] is of rectangular shape swinging between 0 V and the error voltage e_d sampled at the summing junction of the main amplifier. This AC voltage can then be amplified with next to no additional drift error by a separate AC amplifier. The output signal of this amplifier is then rectified yielding a DC signal that can then be used to compensate the drift of the main DC amplifier.[15]

This ingenious scheme was quickly put into use in nearly all operational amplifiers used for analog computers. Apart from the automatic drift compensation this so-called *chopper stabilization* had a second advantage: For low frequency input signals the gains of the main DC amplifier and the stabilizing AC amplifiers essentially add up yielding a very high gain overall amplifier. As equation (4.3) shows, a high gain is essential for keeping the errors of a feedback circuit low. Accordingly the amplifier developed by Goldberg had an overall DC gain of 150,000,000 corresponding to 163 dB.[16] Later developments achieved even higher gains for DC signals.

[12] See [Goldberg et al. 1954].

[13] Such vibrators were in common use until the 1960s in plate supplies driven by batteries etc.

[14] These vibrators are mostly driven by a 60 Hz or a 400 Hz signal resulting in a very typical audible hum of such amplifiers.

[15] The output of the drift compensation stage is often used to detect amplifier overload. If an amplifier is overloaded it cannot maintain the necessary feedback signal to keep the summing junction at ground potential which is reflected by an excessively high output value of the AC amplifier that can easily be detected. Such overloads may result from improper setup and/or scaling of a problem's equations. An overloaded amplifier will yield an erroneous result, thus normally a computation is stopped in case of an overload.

[16] See [Jung 2006][p. 780].

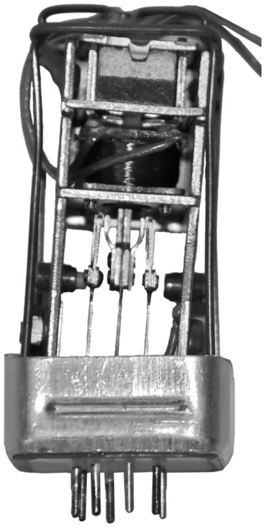


Figure 4.11: Typical electromechanical chopper

Electromechanical choppers were used well into the 1970s to stabilize precision operational amplifiers. This was due to their low resistance when the relay contacts are closed and the very high resistance for the opposite state. Nevertheless, there were some problems to be overcome in such electromechanical choppers: First of all the commercially available devices had a rather short life-span. Second: The contact pairs often acted as thermocouples thus introducing another error in the drift-compensation circuit.

Figure 4.11 shows a typical 400 Hz chopper made by the German manufacturer *KACO* for use in Telefunken analog computers. The 400 Hz excitation voltage is supplied through a top connector (not visible here) to minimize injection of noise into the amplifier circuit. The top half of the chopper contains the coil which causes the middle contact to oscillate between its two counterparts on the left and right side. The chopper system itself is suspended from a mounting frame with an elastic band to minimize vibration of the overall enclosure.

In some cases a common AC amplifier was used to stabilize more than one operational amplifier in a round robin fashion thus reducing cost and complexity but sacrificing the very high overall gain of the combined amplifier. An example for this technique can be found in the analog computer used for steering the Nike missiles. Here a rotating switch connected one AC amplifier with its input chopper and output rectifier sequentially to a number of amplifiers. This required an additional capacitor at the main amplifier inputs used for drift compensation to store the signal generated by the AC amplifier.

Figure 4.12 shows a typical vacuum tube operational amplifier with chopper stabilization. The chopper is the gray rectangle in the upper right part of the picture. This particular amplifier is used in a Solartron *Minispace* analog computer.

Figure 4.12: Chopper stabilized operational amplifier (Solartron)

Figure 4.13 shows one of the earliest commercially available transistorized precision operational amplifiers that was part of the first transistorized tabletop analog computer made by Telefunken.[17] The circuit card on the left side contains the DC coupled main amplifier with its two large germanium output transistors visible on the top. The card on the left is the AC amplifier that is fed with the chopped error signal and provides the correction signal for the main amplifier. The resulting combined amplifier has a DC gain of up to 10^9.

Figure 4.14 shows an example for a late precision amplifier with an all-electronic chopper. This amplifier is used in a Telefunken *RA 770* precision analog computer and has a DC gain of $3 \cdot 10^8$. The chopper is shielded by the box shown in the right upper part of the picture. The output stage transistors with their heat sinks are in the lower right part.

4.2 Summers

One of the most basic computing elements of an analog computer is the so-called *summer* or *summing amplifier* which is basically a direct consequence of equations (4.3) and (4.4). Figure 4.15 shows the simplified schematic of one of the earliest such summing amplifiers developed by Karl D. Swartzel in 1941 at Bell Labs for application in a fire control computer.[18]

[17] This remarkable amplifier was developed by Günter Meyer-Brötz in the late 1950s.

[18] Cf. [Jung 2006][pp. 777 f.].

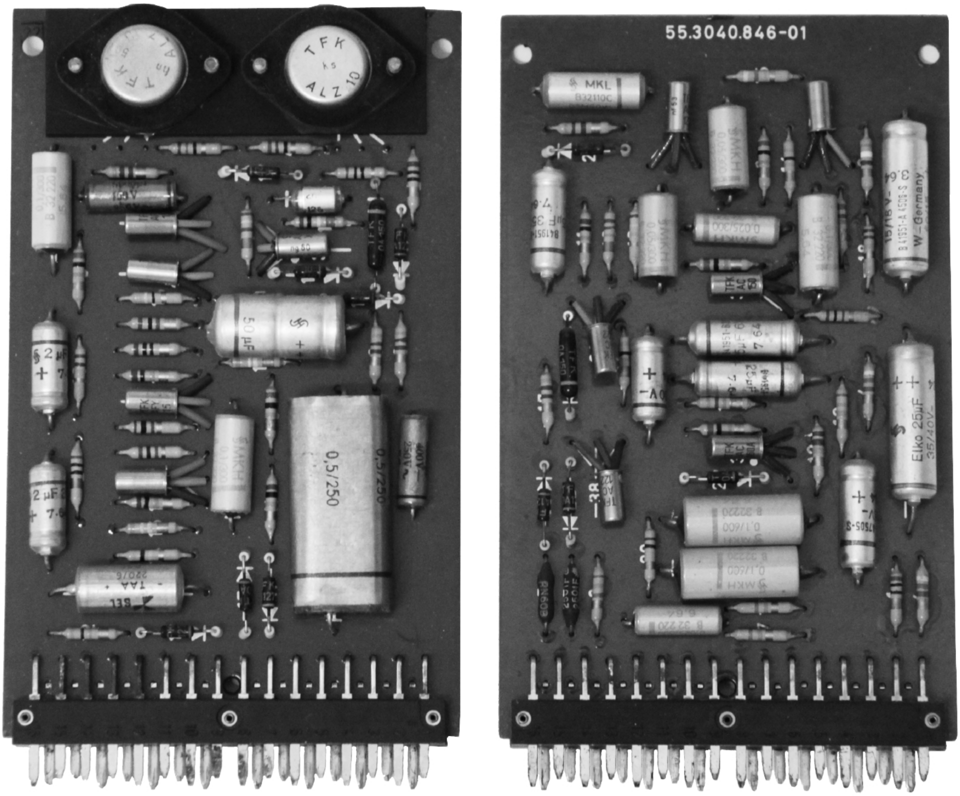


Figure 4.13: Early fully transistorized chopper stabilized amplifier

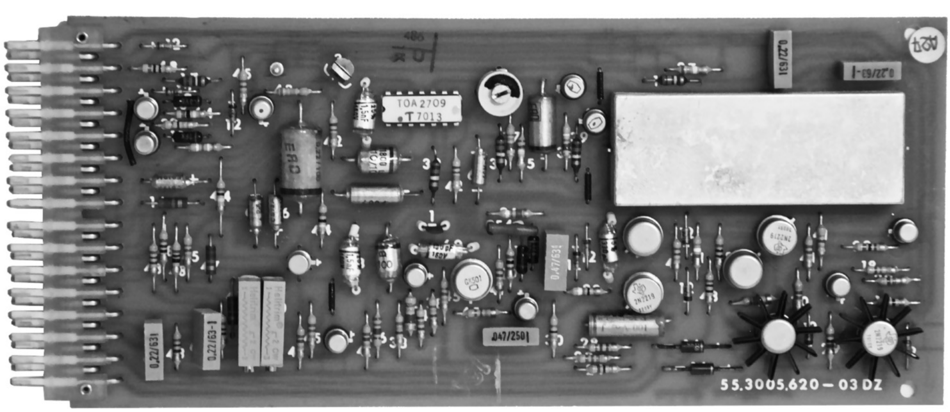


Figure 4.14: Late chopper stabilized amplifier with all-electronic chopper

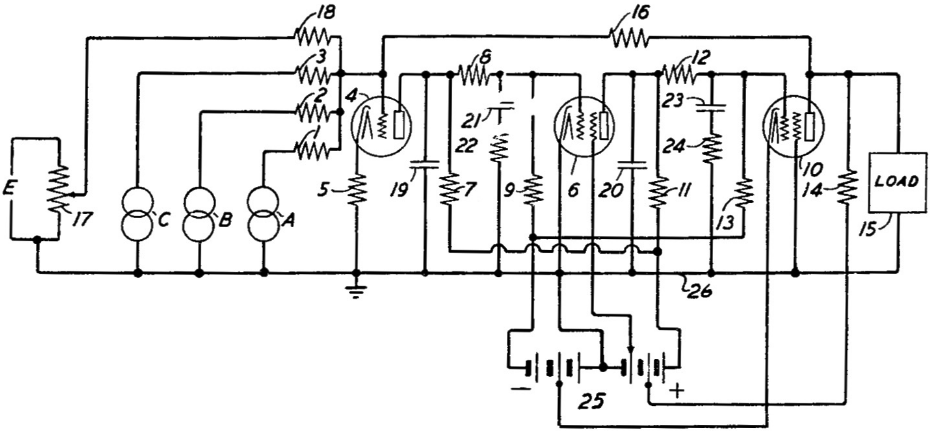


Figure 4.15: Schematic of the "Summing Amplifier" developed by Karl D. Swartzel (cf. [Swartzel 1946])

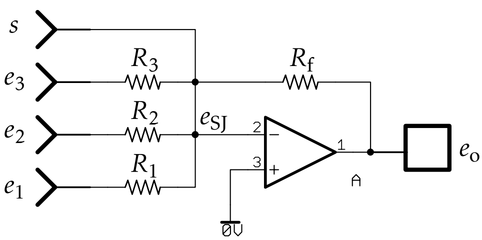


Figure 4.16: Schematic of a three-input summer

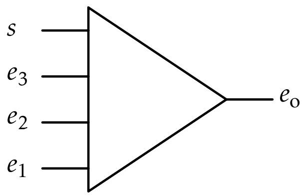


Figure 4.17: Symbol of a summer

This particular amplifier has three inverting stages employing one triode in the input stage and two tetrodes for the remaining two stages, so the overall amplifier is of the inverting type which allows the application of a negative feedback scheme as developed by Black. The summing junction at the grid of the triode input stage is connected to four input resistors 1, 2, 3 and 18 as well as to the feedback resistor 16. Of these four inputs only three are used for signals to be processed while the fourth input is reserved for a manual drift compensation circuit consisting of a voltage source E and a potentiometer 17. This amplifier requires supply voltages of ±350 V, −135 V, +250 V and +75 V and has an overall gain of about 60,000. The various RC networks between the three amplifier stages help stabilizing the circuit.

Figure 4.16 shows a corresponding setup using an operational amplifier. In this case three inputs e_1, e_2 and e_3 and a connection to the summing junction[19] as well as an output connection are available. The simplified symbol for such a summer is shown in figure 4.17.

Given sufficient gain A of the operational amplifier used, the operation performed by this circuit is determined by the passive elements at its input and in the feedback loop.

[19]Many analog computers allow direct connections to the summing junction of the operational amplifiers used in summers and integrators.

Extending equation (4.4) yields

$$\sum_{i=1}^{n} \frac{e_i}{R_i} = -\frac{e_\mathrm{o}}{R_\mathrm{f}} \tag{4.5}$$

for n inputs connected to input resistors $R_1, \ldots, R_n$ and a feedback resistor R_f. Introducing weighting coefficients $a_i = R_\mathrm{f}/R_i$ for all inputs, the output voltage of this circuit is just

$$e_\mathrm{o} = -\sum_{i=1}^{n} a_i e_i.$$

Thus a typical summing amplifier will always yield the negative of the sum of its input voltages.[20] Typical values for a_i are 1, 10 and in some cases 4 or 5.[21] If other weights are required in a computation, inputs can be coupled or external resistors or potentiometers can be connected to the summing junction input.

Figure 4.18 shows a typical early summing amplifier – a model K3-A *adding component* made by GAP/R. This device computes the sum of four input voltages e_1, e_2, e_3 and e_4 plus an adjustable constant value e_0 that can be set manually using the potentiometer in the top half of the device. It delivers two output signals with inverse signs satisfying $\pm k e_0 + e_1 + e_2 + e_3 + e_4$ with $-1 \le k \le 1$ and $e \in 0, 5, 50$ V.

4.3 Integrators

Replacing the feedback resistor R_f of a summer with a feedback capacitor C yields a circuit capable of integration as shown in figure 4.19. Since the feedback current i_f flowing through the capacitor C is determined by

$$i_\mathrm{f} = C\dot{e}_\mathrm{o}$$

equation (4.5) can be changed to

$$\sum_{i=1}^{n} \frac{e_i}{R_i} = -C\dot{e}_\mathrm{o}. \tag{4.6}$$

Redefining the weighting coefficients as $a_i = 1/R_i C$ and solving (4.6) for e_o yields

$$e_\mathrm{o} = \int_0^t \sum_{i=1}^{n} a_i e_i \,\mathrm{d}t + e(0)$$

where $e(0)$ denotes the initial value of the integrator.

[20] This holds true for most components of an analog electronic analog computer and is often a source of confusion.

[21] If an input's weight differs from one it is written next to the input in the triangle of the simplified summer symbol shown in figure 4.17. If no weight is noted, the corresponding input is weighted with 1.

Figure 4.18: Model K3-A adding component (GAP/R), front and interior view

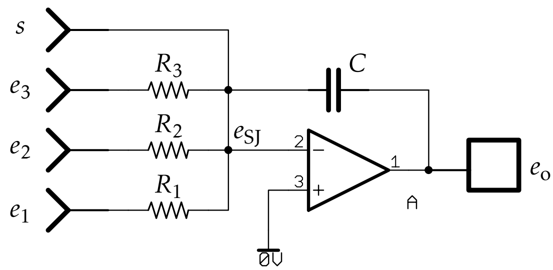


Figure 4.19: Basic structure of an integrator

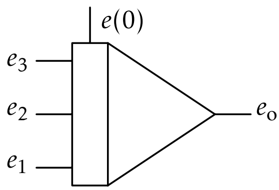


Figure 4.20: Symbol of an integrator with three inputs

Figure 4.20 shows the symbol normally used to denote an integrator in an analog computer setup. Since such an integrator is obviously working with time being the free variable it needs some means to reset or even better preset the integrating capacitor C to an initial value. In addition to this, additional circuitry is necessary to start and stop the operation of the integrator. Figure 4.21 shows the (still simplified) schematic of a more realistic integrator as it is used in a typical analog computer.

The central control elements are the two switches labeled *IC*, short for *Initial Condition* and *OP*, short for *Operate/Hold*. The integrator shown in this figure is currently in its IC mode: The inputs e_1, e_2 and e_3 are grounded through the operate/hold switch while

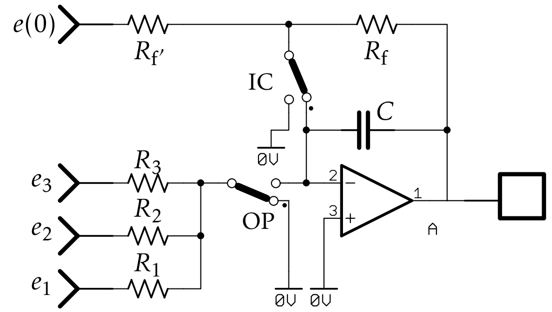


Figure 4.21: More detailed schematic of an integrator showing the control switches *h* and *c*

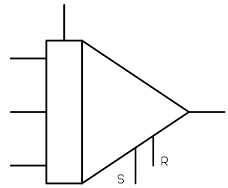


Figure 4.22: Integrator symbol showing control inputs for IC and OP switches

the initial condition switch effectively turns the circuit into a simple summer with only one input $e(0)$. In addition to that, the integration capacitor C is switched parallel to R_f. Since the summing junction at the inverting input of the operational amplifier is always maintained at 0 V, the capacitor will be charged to the potential at the output of the amplifier. This, in turn, is determined by the input signal at $e(0)$ and the ratio $R_f/R_{f'}$ which is normally 1. In this mode of operation the capacitor is pre-charged to a given initial value $-e(0)$.[22]

To switch the integrator into the *Operate Mode* both switches change their state. This grounds the connection between $R_{f'}$ and R_f thus disabling the input $e(0)$ and connects the input resistors R_1, R_2 and R_3 to the summing junction of the amplifier. In this state the integrator performs an integration with respect to time.

If the switch OP is reset the integrator enters *Hold Mode* – in this mode the last value of the integration will be stored.[23] This mode is normally used to pause a computation to read out values of interest from the various elements of the current setup.

In most cases the control switches IC and OP of all integrators in an analog computer setup are controlled by a common set of signals. In some cases it is necessary to control integrators individually, so one group of integrators can solve some equations based on initial values determined by another group etc. In addition to this, an integrator can also be used as an analog memory cell by using only the initial condition and hold modes. If an integrator is to be operated under individual control the symbol shown in figure 4.22 is used which features two additional inputs controlling the switches IC and OP.

A realistic integrator circuit is still more complex than the schematic shown in figure 4.21: High-speed integrators use electronic switches in place of the relay contacts shown here. Additionally most integrators feature more than one integration capacitor

[22] In the early 1950s there were even Neon tubes used to reset (short circuit) the integration capacitors. Small Neon tubes were enclosed in a coil that could be energized with a RF signal which in turn caused the tube to conduct. Since this does not provide any means to preset an integrator to $e(0)$ this value had to be introduced later in the computer setup by a summer or the like. A schematic of a Philbrick K2-J integrator with a reset Neon tube triggered by an RF generator can be found in [Russell 1962][p. 6-24].

[23] Due to small leakage currents in the capacitor as well as still acting small drift effects the output of the integrator will slowly change during hold mode.

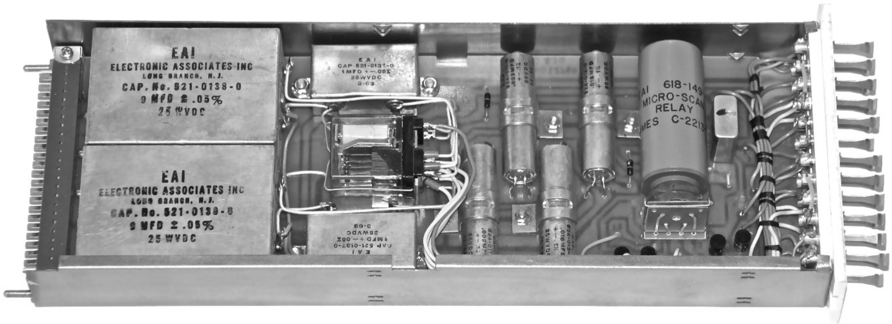


Figure 4.23: EAI double integrator used in an EAI 580 analog computer

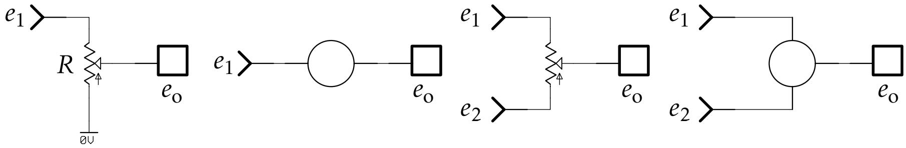


Figure 4.24: Coefficient potentiometers

C thus allowing to select different ratios a_i for their operation. Some machines offer as many as four or five different values for C (usually powers of ten) that can be selected either individually using shorting plugs or from a central control panel.

Figure 4.23 shows a double integrator made by *Electronic Associates Inc.*, *EAI* for short. The chassis shown contains the passive elements for two integrators – the associated operating amplifiers are contained in a second drawer and are not shown here. Each integrator channel contains four capacitors yielding selectable time constants of 1 second, 100 ms, 2 ms and 200 μs for high-speed operation.

4.4 Coefficient potentiometers

The fixed input weights a_i of summers and integrators are not sufficient for setting coefficients in a computer setup. Therefore so-called *coefficient potentiometers* are required. In the simplest case these are just potentiometers connected as voltage dividers as shown in figure 4.24. Such a voltage divider, shown on the left, is normally represented by a circle with one input and one output connection. In some cases it is necessary to have a potentiometer with two inputs, which is shown in the right half of the figure.

Let α denote the position of the potentiometer's wiper. Effectively this partitions R into an upper and a lower resistor R_u and R_l connected in series:[24]

$$R_u = (1-\alpha)R$$
$$R_l = \alpha R$$

[24]Here and in the following it is always assumed the R is a linear potentiometer.

Without load the operation of a coefficient potentiometer can be described by

$$\frac{e_o}{e_1} = \frac{R_l}{R}.$$

In a traditional analog computer this is a hypothetical case since the following computing elements normally act as a load to this voltage divider that cannot be neglected.[25] A computing element connected to the wiper of a potentiometer acts as a resistive load R_{load} connected in parallel to R_l. Thus

$$R_l \parallel R_{\text{load}} = \frac{R_l R_{\text{load}}}{R_l + R_{\text{load}}}.$$

So the total resistance of the coefficient potentiometer with a connected load is

$$R_{\text{total}} = R_u + R_l \parallel R_{\text{load}} = R_u + \frac{R_l R_{\text{load}}}{R_l + R_{\text{load}}}.$$

For the loaded potentiometer it is

$$\begin{aligned}
\frac{e_o}{e_1} &= \frac{R_l \parallel R_{\text{load}}}{R_{\text{total}}} = \frac{\dfrac{R_l R_{\text{load}}}{R_l + R_{\text{load}}}}{R_u + \dfrac{R_l R_{\text{load}}}{R_l + R_{\text{load}}}} = \frac{R_l R_{\text{load}}}{R_u(R_l + R_{\text{load}}) + R_l R_{\text{load}}} \\
&= \frac{R_l}{\dfrac{R_u R_l}{R_{\text{load}}} + R_u + R_l} = \frac{\alpha R}{(1-\alpha)R\dfrac{\alpha R}{R_{\text{load}}} + R} \\
&= \frac{\alpha}{(1-\alpha)\alpha\dfrac{R}{R_{\text{load}}} + 1}.
\end{aligned} \tag{4.7}$$

Figure 4.25 shows the effect of potentiometer loading with respect to the value set by α. It is obvious that setting a potentiometer manually by just looking at the value of its dial will not yield a correct setup since the load the potentiometer is connected to has to be taken into account. In the simplest case load-correction graphs can be used which plot the necessary offset for α for some typical loads.

Despite being simple, this method does not yield very good results, so most analog computers have a special mode of operation, called *Potentiometer Set* or *Pot Set* for short. In this mode that can be selected from the central control panel of the computer, all integrators and summers have their input resistor networks grounded, so no actual computation takes place while every computing element exhibits its real load to its signal sources. To set potentiometers in this mode, small and simple systems offer a push button next to each potentiometer that will connect its input to +1 machine unit and connect a precision voltmeter – either a compensation voltmeter or a digital

[25] Some late implementations used buffer amplifiers for the coefficient potentiometers as will be shown in the following.

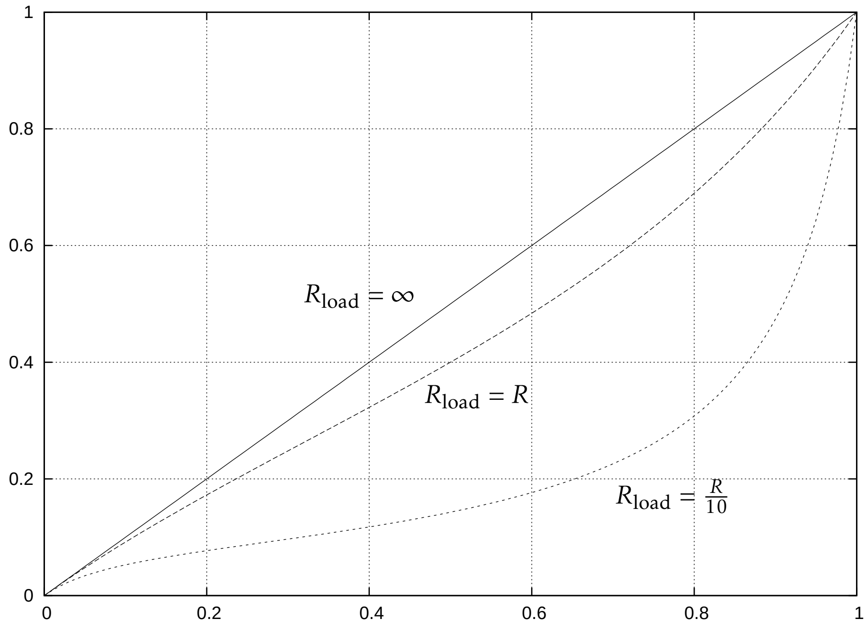


Figure 4.25: Coefficient potentiometer with $R_{\text{load}} = \infty$, $R_{\text{load}} = R$ and $R_{\text{load}} = R/10$

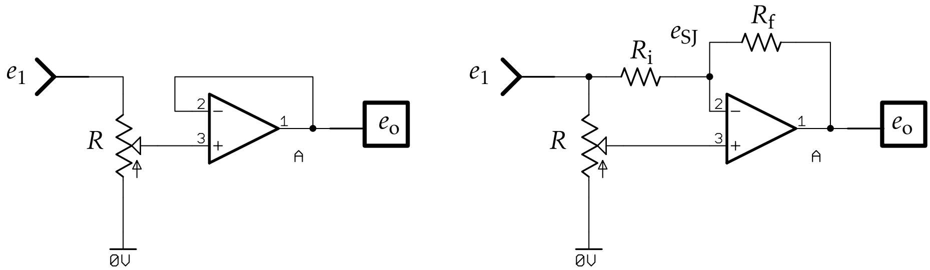


Figure 4.26: Buffered coefficient potentiometers

voltmeter – to its wiper. Turning the potentiometer's dial the desired coefficient value can be set very exactly by means of the voltmeter.

When integrated operational amplifiers became cheap some late analog computers employed simple buffer stages or even more elaborate schemes to decouple the coefficient potentiometers from the load of the following computing element. Figure 4.26 shows two typical buffering schemes: The circuit on the left uses an operational amplifier as a non-inverting output buffer for the potentiometer R.

The circuit shown on the right in figure 4.26 is more versatile and is used for example in the EAI 2000 hybrid computer. While a simple buffer coupled with a voltage divider can only be used as an attenuator according to

$$\frac{e_o}{e_1} = \alpha$$

this circuit allows coefficient values to be set in the range $-1\dots 1$ as the following consideration shows: If α is set to 0, the potentiometer's wiper is at its grounded end, thus the non-inverting input of the operational amplifier is grounded and the amplifier acts as a simple inverting amplifier with gain $R_\mathrm{f}/R_\mathrm{i}$. Typically $R_\mathrm{f} = R_\mathrm{i}$ is chosen so that the amplifier gain is 1. The potentiometer setting $\alpha = 1$ places the wiper at the upper end of the potentiometer thus connecting the non-inverting input of the amplifier with the input e_1. Since the amplifier will maintain the voltage e_SJ at its inverting summing junction at 0 V, it follows that $e_\mathrm{o} = e_1$. So this circuit can be described by[26]

$$\frac{e_\mathrm{o}}{e_1} = 2\alpha - 1.$$

Large installations often had servo driven potentiometers that could be set from a central control panel or from an attached paper tape reader or even a stored-program computer to allow rapid setup of a problem. Figure 4.27 shows a typical servo potentiometer from an EAI 580 analog computer. The 10-turn wire wound potentiometer is visible on the left while the motor driving a reduction gear can be seen in the upper right half. The small circuit board below the motor holds the relay used to select this potentiometer for readout. Later hybrid computers[27] often employed digitally controlled attenuators.

Using only summers, integrators and coefficient potentiometers, an analog computer would be restricted to the solution of only very simple differential equations of the form

$$\sum_{i=0}^{n} a_i y^{(i)} = 0.$$

Since this would be too much of a restriction, the following sections describe some additional basic computing elements which are necessary to give an analog electronic analog computer its outstanding computational power.

4.5 Function generators

One very basic problem is the generation of a function $f(x)$. In the easiest case such a function can be described by the solution of a differential equation. Many typical problems require functions which are defined by experimental values or the like. In cases like this, special computing elements called *function generators* are required. Generating arbitrary functions with purely analog electronic means is often complicated and/or requires an immense amount of circuitry which was one of the reasons to develop hybrid computers[28] in which a stored-program digital computer is connected to an analog computer and can be used to generate functions in an algorithmic way. Especially difficult, at least in the common case, is the generation of functions of more

[26] Cf. [Bader 1985][pp. 76 f.].
[27] See section 9.
[28] See section 9.

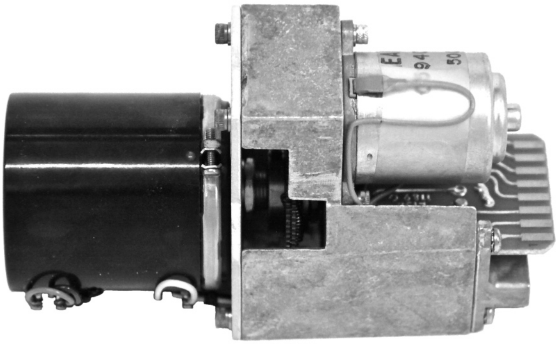


Figure 4.27: EAI servo potentiometer

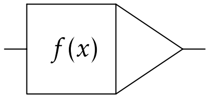


Figure 4.28: Symbol of a function generator

than one variable by analog electronic means. Figure 4.28 shows the general symbol of a function generator with one input.

A plethora of different approaches has been developed to generate more or less arbitrary functions. The following sections describe some of the more widely used or technologically more interesting implementations.

4.5.1 Servo function generators

The earliest function generators as used by Lovell et al. in their fire control computer were based on a servo circuit driving at least two potentiometers mounted on a common shaft by means of a motor. One of these potentiometers is used as feedback for the servo circuit while the other(s) generate the function(s) desired. Figure 3.22 shows such a setup used to generate trigonometric functions.

Variants of these servo based function generators employ cams driven by the servo controlled motor with a feeler gauge that in turn drives an output potentiometer. Another implementation of a cam potentiometer is described in [Loveman 1962].

Both of these techniques have the disadvantage that specially wound potentiometers are very delicate instruments and difficult to manufacture. The same is true for machined precision cams. Thus quite early so-called *tapped potentiometers* were employed for function generation. These are based on typical 10-turn wire wound potentiometers that have been modified to feature multiple taps welded equidistantly to the resistive element. Using these taps, the potentiometer can be used as a linear interpolator: Each tap is connected to an adjustable voltage source – the tap's position corresponds to the breakpoint while the voltage applied to the tap represents the desired function value at this particular breakpoint. Using a servo system driving the potentiometer's shaft according to an input value x such a setup will yield a function $f(x)$ which is the result of linear interpolation between the values set at the breakpoints.[29]

[29] See [Loveman 1962][pp. 3-36 ff.] for more information about tapped potentiometers.

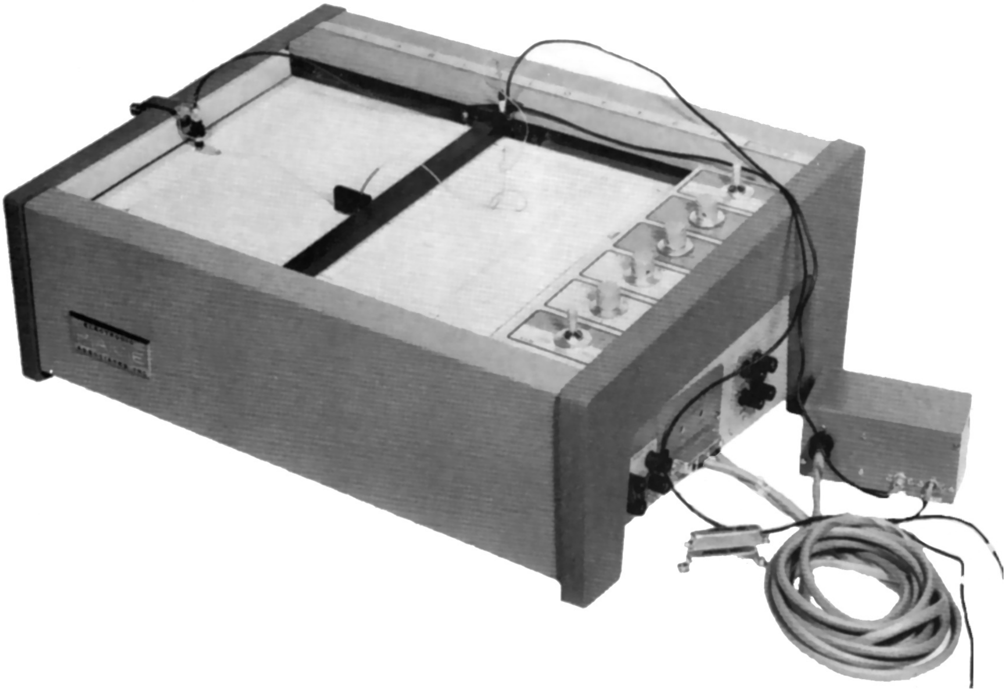


Figure 4.29: EAI curve follower

4.5.2 Curve followers

Another servo based function generator is the so-called *curve follower* which is basically a servo circuit connected to a xy-recorder as shown in figure 4.29.

The basic idea is as simple as it is elegant: The function to be generated is painted or plotted on a sheet of paper with conductive ink. This graph is then placed onto the table surface of the curve follower which is a modified xy-recorder that has a pickup coil mounted in place of the pen normally used. An external sine generator is connected to the conductive trace of the function graph on the paper. A simple servo circuit will then control the y-input of the recorder in such a way that the pickup coil will always be placed directly over the conductive ink line, while the x-input to the curve follower is fed with the function argument. The y value used to drive the y-input of the curve follower then represents the value $f(x)$.

Other curve followers use an optical pickup to follow the function graph instead of conductive ink and an inductive pickup. The basic principle of operation remains the same.

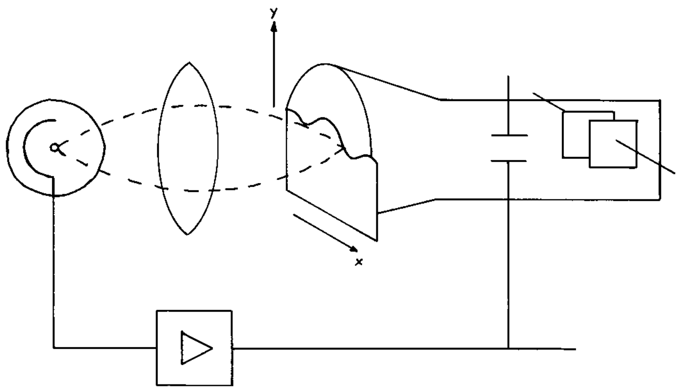


Figure 4.30: Optical function generator (cf. [Haug 1960][p. A3])

4.5.3 Photoformers

As elegant and demonstrative these electromechanical function generation techniques are, their main disadvantage is that they fail for rapidly varying function arguments x due to the masses to be moved in order to follow the function's graph or moving a cam etc. But the idea of the curve follower can readily be implemented using the all-electronic approach shown in figure 4.30.

The function to be generated is modeled by an opaque mask placed in front of an oscilloscope tube. The input to the x-deflection plates of the oscilloscope tube is fed by the function argument. Using a photomultiplier and a simple servo circuit a signal for driving the y-deflection plates is now generated in a way that guarantees that the light spot will always be placed at the boundary of the opaque mask, so this system effectively generates $y = f(x)$.[30]

4.5.4 Varistor function generators

A completely different approach to function generation is the application of *varistors*.[31] A varistor is a current sensitive non-linear resistor implementing a basic function of the form[32]

$$e_o = ae_1^n$$

with n being determined at production and normally set to $n = 2$ or $n = 3$. The precision of these varistors is quite remarkable, [Korn 1962][p. 3-75] quotes an accuracy of 0.4 V for the *Quadratron*,[33] a varistor made by *Douglas Aircraft Company*.

[30] Cf. [Macnee 1948][Sec. IIIE], [Haug 1960][p. A3] and [Korn 1962][pp. 3-68 ff.] for more detailed information. Even a Williams-tube like construction was once employed. This proved useful in repetitive operation of high-speed analog computers when the result of one run had to be used as the input for the next run, see [Haug 1960][p. A3].

[31] See [Korn 1962][pp. 3-73 ff.].

[32] Cf. [Korn 1962][p. 3-75].

[33] See [GAP/R 1959].

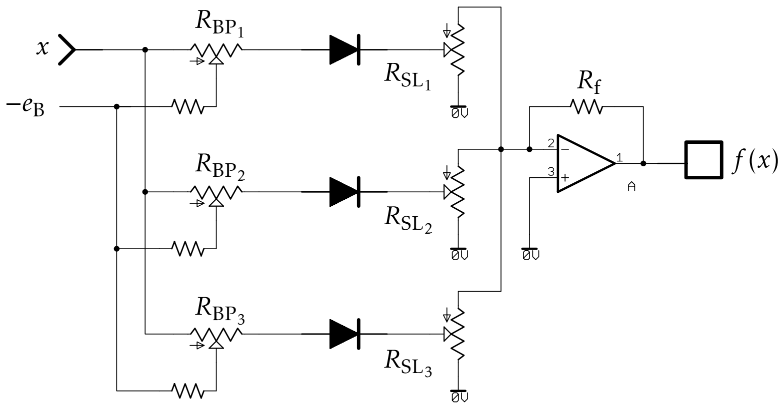


Figure 4.31: Principle of operation of a diode function generator

Of course having square and cube functions does not give the kind of flexibility a photoformer offers but compared to this the Quadratrons were extremely cheap. In fact, these devices were cheap enough to be used in rather high numbers to generate functions by truncated power series.[34]

4.5.5 Diode function generators

The availability of reliable and cheap germanium and later silicon rectifiers gave rise to yet another function generator principle to be implemented successfully, the so-called *diode function generator* or *polygon function generator*. The basic idea is shown in figure 4.31.

A (perfect) diode acts like a switch that will conduct current only in one direction but not in the reverse. So a simple idealized diode might be used to implement a function like

$$f(x) = \begin{cases} ax & \text{if } x > 0 \\ 0 & \text{otherwise} \end{cases}$$

where the slope a can be set by a series resistor. If one could shift the point where the diode begins to conduct arbitrarily, a function could be generated by a piecewise linear approximation. The circuit shown in figure 4.31 is based on this principle: Each of the three diodes will yield one linear function with variable breakpoint. The breakpoint position is determined by the negative bias voltage $-e_b$ and the setting of the corresponding bias potentiometer R_{BP_i}. The value a of the respective linear function each diode yields, is determined by the slope potentiometer settings of R_{SL_i} and the feedback resistor R_f of the operational amplifier acting as a three-input inverting summer.

Of course, a practical diode function generator has much more than three segments.[35] Figure 4.32 shows a typical implementation of a diode function generator implement-

[34] Cf. [Korn & Korn 1964][pp. 6-14 f.] for more details.
[35] More information can be found in [Korn 1962][pp. 3-66 ff.] and [Haug 1960][p. A4] for example.

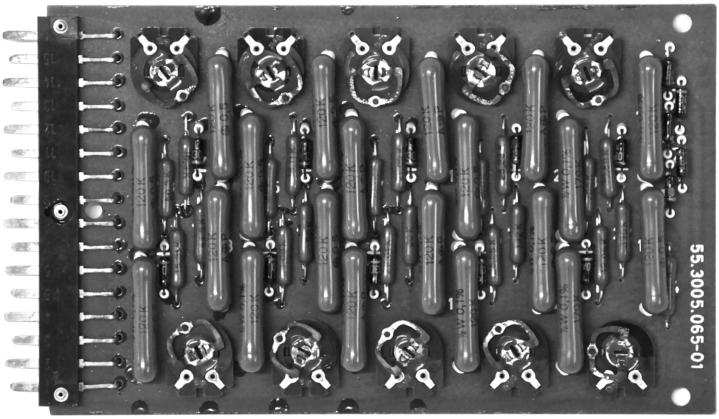


Figure 4.32: $f(x) = x^2$ for $x \geq 0$

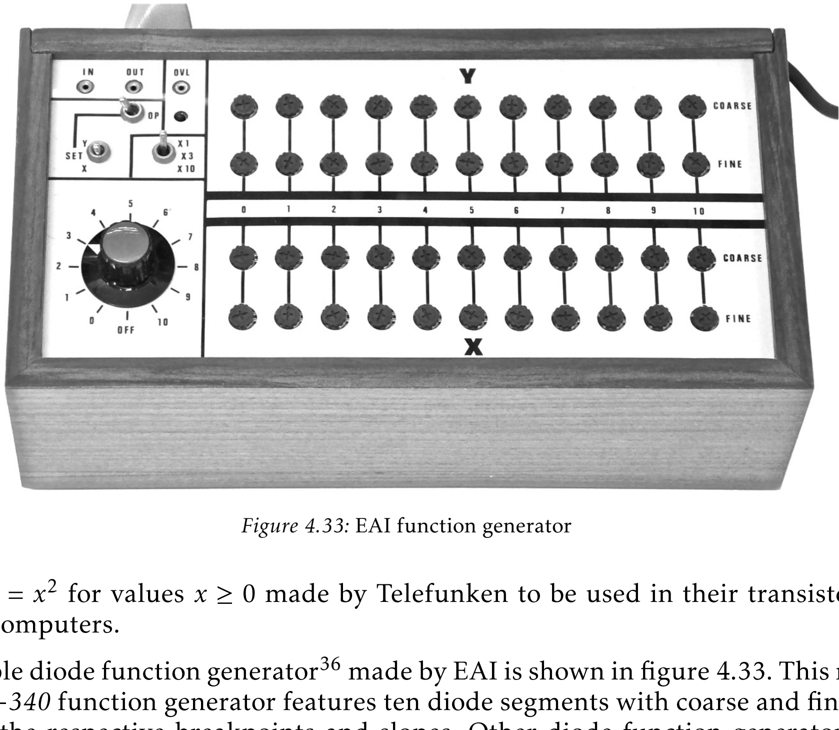


Figure 4.33: EAI function generator

ing $f(x) = x^2$ for values $x \geq 0$ made by Telefunken to be used in their transistorized analog computers.

A variable diode function generator[36] made by EAI is shown in figure 4.33. This model *EAI 180-340* function generator features ten diode segments with coarse and fine control for the respective breakpoints and slopes. Other diode function generators dispensed with the variable breakpoints and implemented 21 equally spaced breakpoints with only the slopes to be set manually.[37]

[36] *VDFG* for short.

[37] Both implementations have their specific advantages: Being able to set slope and breakpoint for each segment often allows a better function approximation while diode function generators with fixed breakpoints are easier to setup.

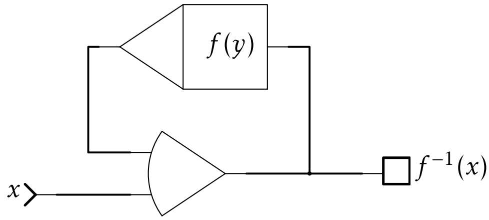


Figure 4.34: Generating an inverse function using an open amplifier

4.5.6 Inverse functions

Many functions can be readily implemented by generating the inverse of another function. Using a function generator for the negative feedback of an operational amplifier yields the inverse function (as long as there is no sign change and the feedback function is reasonably smooth – in most cases it is advisable to add a small capacitor, about 680 pF, between the output of the amplifier and its summing junction to stabilize the circuit). Figure 4.34 shows the basic setup to generate an inverse function $f^{-1}(x)$ using a function generator $f(y)$ and an open amplifier as described.

4.5.7 Functions of two variables

Generating functions of more than one variable, the case with two variables being the most common, is rather difficult using analog electronics only.[38] Some approaches to solve this problem involved the use of three-dimensional cams[39] with appropriate servo motor circuits. Another technique is based vaguely on the curve follower idea: An xy-recorder with its pen replaced by a pickup stylus is controlled via its x- and y-inputs moving the stylus over a specially prepared conductive sheet. On this sheet equipotential contour lines can be drawn with conductive ink – these lines are then connected to adjustable voltage sources. The conductive sheet then performs a two-dimensional linear interpolation between these equipotential lines.

A variation of that scheme is the *electrolytic trough*. In the simplest case this is just a two-dimensional bath filled with a conductive fluid and a pickup electrode that is again moved by a xy-recorder. Using wires embedded in the bath, equipotential lines can be implemented as above. Using a non-flat bottom plate even more complicated functions can be represented. In some cases even a three-dimensional movement of the pickup-electrode has been employed to yield $f(x,y,z)$.

Yet another possibility is to use a couple of function generators of one variable and feed a tapped potentiometer function generator with their respective outputs. In cases where the response times that can be achieved with these servo-based techniques is not sufficient, the use of an oscilloscope tube, a translucent mask, and a photomultiplier has been proposed. The main problem here is the manufacturing of the translucent mask that represents the function $f(x,y)$ by its transparency.

[38] Some basic information about this topic can be found in [Korn 1962][pp. 3-78 ff.] and [Haug 1960][pp. A9 f.].

[39] Cf. figure 2.5.

4.6 Multiplication

Despite being simple to use analog techniques to implement an integrator, it turns out to be difficult to model multiplication. Of course, the principles described above concerning the generation of functions $f(x,y)$ could be employed to implement a multiplier which has been done occasionally as the following sections show.

4.6.1 Servo multipliers

An obvious way to implement a multiplier is based on a servo function generator driving a number of linearly wound potentiometers mounted on a common shaft. The input variable to control the servo circuit would be x while the ends of the free potentiometers, i. e., those that are not needed for the servo-feedback, are connected to $+y_i$ and $-y_i$.[40] If x is allowed to take on values between $+1$ and -1 this setup allows for so-called *four-quadrant operation*.

Figure 4.35 shows a five channel servo multiplier made by Solartron: The left half of the device houses the transformers for the power supply.[41] The power transistors driving the servo motor are mounted on the two large black heat sinks visible in the upper middle of the picture. The heart of the servo multiplier can be seen in the lower right half: six potentiometers mounted on a common shaft driven by the rather small servo motor on the left. One of these six potentiometers is used to generate the feedback signal for the servo circuit while the remaining five potentiometers are available for multiplication. Thus, with one multiplicator input x, this servo multiplier can generate five products $xy_1, xy_2, \dots, xy_5$ given multiplicands $+y_i$ and $-y_i$, $1 \leq i \leq 5$.

Of course servo multipliers have the restriction that the multiplicator value x is limited regarding its rate of change. If it changes too rapidly, the servo will be unable to follow precisely introducing errors in the products. Nevertheless, in many applications this restriction turns out to be not a severe one – the advantage of generating a number of products based on a common multiplicator often outweighs it.

4.6.2 Crossed-fields electron-beam multiplier

One of the first all-electronic multiplication schemes is the *crossed-fields electron-beam multiplier* that has been described by [Macnee 1948][Sec. IIIA]. The central element of this type of multiplier is an oscilloscope tube with x- and y-deflection plates as well as an additional single deflection coil.[42]

The x-input to the multiplier drives the x-deflection plates acting on the electron beam resulting in a horizontal velocity of $|v_x| \sim x$. The y input signal drives the deflection coil wound around the neck of the tube by means of a suitable power amplifier. This coil

[40] Normally, these potentiometers feature a middle tap that is connected to ground to enforce $0 \cdot y$ is as near to zero as possible. Otherwise slight errors in the potentiometer's linearity would interfere with that requirement.

[41] Servo multipliers normally have a dedicated power supply for the servo circuit to minimize noise coupling into the precision supplies of the analog computer.

[42] Cf. [Haug 1960][pp. A6 f.].

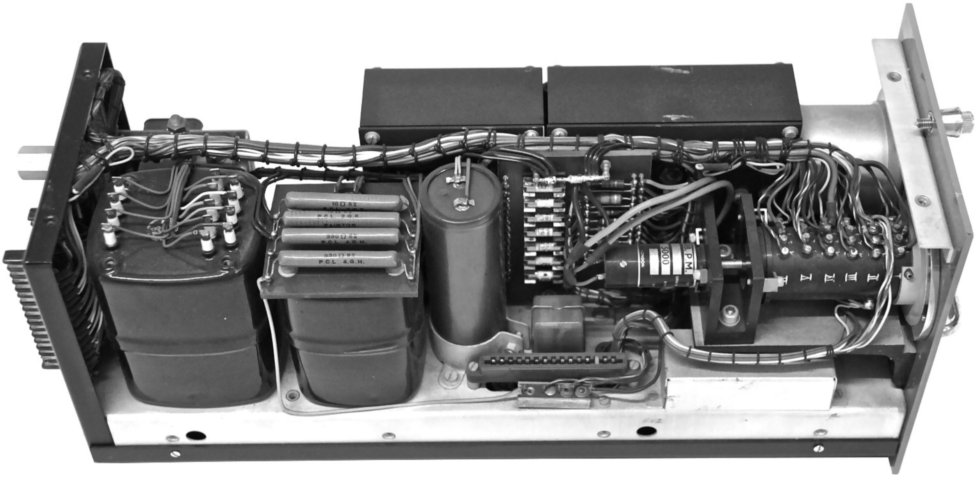


Figure 4.35: Solartron servo multiplier

will in turn generate a magnetic field $B \sim y$ in axial direction that will deflect the electron beam in vertical direction by an amount xy.

Using an opaque mask dividing the screen into two equal parts and a photodetector,[43] a servo circuit like the one outlined in the section covering the photoformer can now be employed to generate a signal to drive the y-deflection plates of the oscilloscope tube in such a way that the spot of light will always stay on the edge of the opaque mask. Thus, y will be in proportion to xy.

4.6.3 Hyperbolic field multiplier

An even more arcane multiplication scheme based on a very special vacuum tube is the *hyperbolic field multiplier* which was invented by Gundlach in the early 1950s.[44]

Figure 4.36 shows the structure of this hyperbolic field tube. The central elements are two pairs of vertical and horizontal deflection plates and, in addition to this, two pairs of deflection plates that are shaped accordingly to generate a hyperbolic field which the electron beam must cross. The vertical deflection plates are driven by the multiplicator x while the multiplicand is fed crosswise to the two hyperbolic deflection plate pairs. These hyperbolic plates create an electric field proportional to $y\eta\xi$ where η and ξ are the coordinates of a coordinate system with origin in the middle of these four deflection plates.[45]

An electron beam that has not been deflected by the vertical deflection plates will cross the hyperbolic field undisturbed since $\eta = \xi = 0$. A beam deflected horizontally will,

[43] Alternatively, a shield that divides the tube's screen into two equal halves together with two photodetectors has been used successfully.

[44] See [Gundlach 1955].

[45] See [Haug 1960][p. A7].

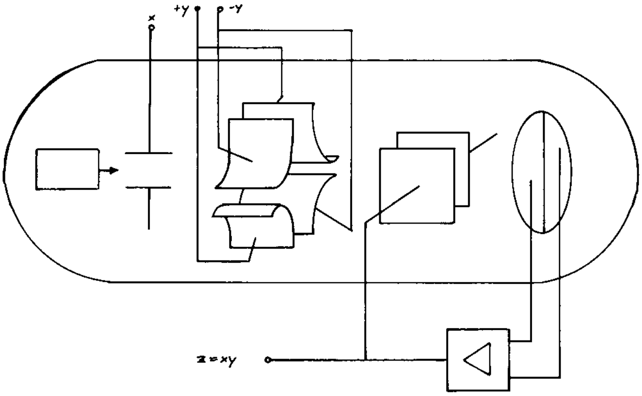


Figure 4.36: Structure of a hyperbolic field tube (cf. [Haug 1960][p. A7])

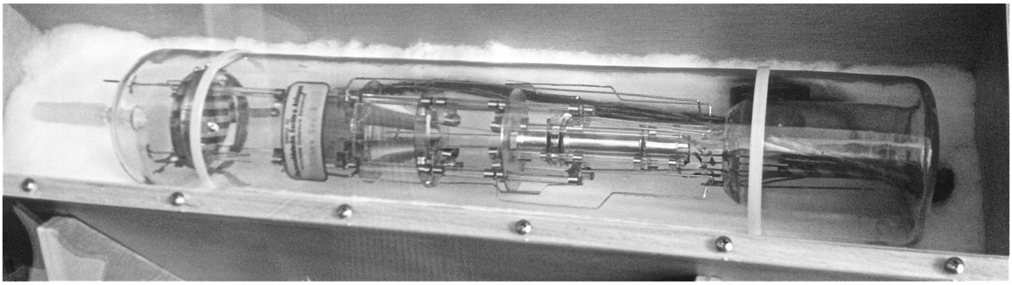


Figure 4.37: Hyperbola tube

in fact, experience an additional deflection by this hyperbolic field which results in a overall horizontal deflection proportional to xy, so this tube works quite similar to the crossed-fields electron-beam multiplier but without the external deflection coil that has been replaced by the hyperbolic deflection plates.

The electron beam then hits a target consisting of two closely spaced pickup plates that are connected to an amplifier that is driving the horizontal pair of deflection plates. This, in fact, forms a servo circuit that will force the electron beam to stay right in the middle of the two pickup plates, so the signal driving the horizontal deflection plates corresponds to the product xy.

This type of multiplier has the advantage of a rather high upper operating frequency compared with the crossed-fields electron-beam multiplier due to the purely electrostatic deflection system instead of a magnetic one. [Haug 1960][p. A7] mentions a dynamic multiplication error of only about 1% at a signal frequency of 80 kHz. A complete schematic of a multiplier based on this special tube can be found in [Dhen 1960][p. 31]. Figure 4.37 shows a hyperbolic field tube.[46]

[46]This specimen is part of the *Informatiksammlung* of the *Darmstadt University of Applied Sciences*.

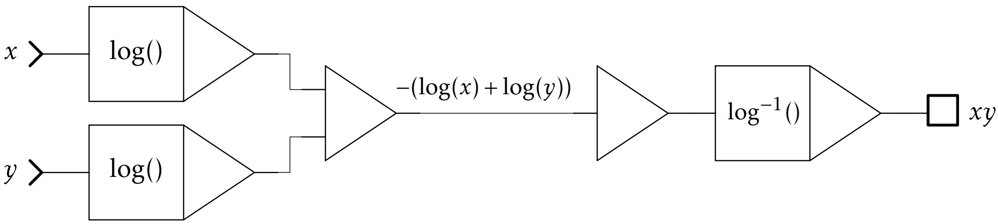


Figure 4.38: Logarithmic multiplier

4.6.4 Time division multipliers

Another multiplier that is related to the servo multiplier described in section 4.6.1 is the so-called *time division multiplier*[47] which is just one particular implementation of a larger family of *modulation multipliers*.

A servo multiplier is essentially just a controllable voltage divider using potentiometers to build the products. Such voltage division can also be implemented readily by modulating a square wave with respect to the mark- and space-times. If this modulator is driven with the variable x a 1:1 ratio of mark to space would correspond to $x = 0$ while $x = +1$ and $x = -1$ would yield a signal being constantly high or low respectively. Generating the average of this square wave signal would now yield an output proportional to the modulation input x.

If the amplitude of this square wave signal is now controlled by the multiplicand y this average signal will yield xy. As in the case of the servo multiplier, this scheme readily extends to the computation of multiple products with a common multiplicator x, so most time division multipliers contain one mark-space modulator driving several amplitude modulators which are controlled by multiple inputs y_i.[48]

4.6.5 Logarithmic multipliers

Another quite obvious implementation of multiplication as a computer circuit is based on the old idea of adding logarithms to achieve a multiplication – the very same principle slide rules depend on. Figure 4.38 shows the basic setup of such a *logarithmic multiplier* which requires three function generators, two yielding the logarithm of an argument and one yielding the inverse logarithm.[49]

The main problem of this type of multiplier is the fact that the two variables x and y are restricted to positive values only unless special precautions are taken which involves generating absolute values and restoring the sign of the multiplication result.

[47] See [Korn & Korn 1964][pp. 7-10 ff.]. Other terms for this device are *mark-space multiplier* or *pulsed-attenuator multiplier*.

[48] A very detailed and rather early circuit description of such a time division multiplier can be found in [Lilamand 1956]. A modern implementation might use a switched capacitor building block in a circuit as described in [Linear Technology][p. 9].

[49] Cf. [Morrill 1962][pp. 3-41 f.].

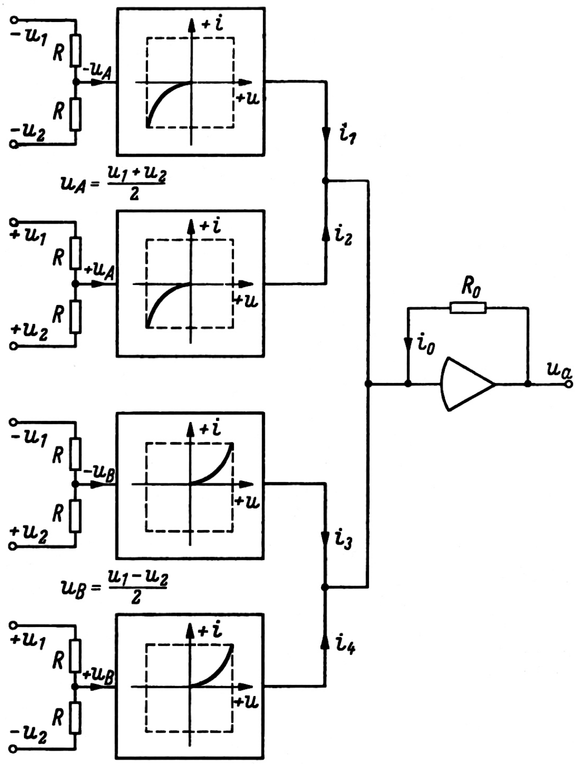


Figure 4.39: Principle of operation of a quarter square multiplier (cf. [Giloi et al. 1963][p. 92])

4.6.6 Quarter square multipliers

So-called *quarter square multipliers* belong to the most widely used classic multiplication schemes. The basic idea is to compute

$$xy = \frac{1}{4}\left((x+y)^2 - (x-y)^2\right)$$

which only requires the computation of two square functions[50] and some summations. The division by 4 is easily done by a voltage divider or by a proper feedback resistor of the output amplifier. Figure 4.39 shows a typical implementation of such a multiplier.

A particular requirement of this type of multiplier is that both variables, the multiplicator as well as the multiplicand, have to be supplied with both signs which normally is not a problem in a practical analog computer setup due to the sign changing behavior of most computing elements. The main advantages of this multiplier are its high bandwidth and rather low price. Unfortunately there is no way to generate multiple

[50]The function generator card shown in figure 4.32 is actually part of such a quarter square multiplier.

products of a common multiplicator which belongs to the domain of servo and time division multipliers.

An interesting variation of this multiplication technique has been used by Hitachi. The Hitachi 505E analog computer featured multipliers based on the generation of $(x-1)^2$ instead of x^2 yielding[51]

$$xy = \frac{1}{2}\left(\frac{(x+y-1)^2}{2} - \frac{(x-y-1)^2}{2}\right) + y.$$

4.6.7 Other multiplication schemes

There are plenty other multiplication schemes that have been developed and implemented during the years. There are *dynamometer multipliers* that balance the forces of multiple electromagnets against each other[52] as well as *strain-gauge multipliers* that are based on balancing a strain-gauge with voice coils,[53] *heat-transfer multipliers,*[54] *Hall effect multipliers*[55] etc.

Today's multipliers are most often based on so-called Gilbert *cells* which were developed by Barrie Gilbert in the late 1960s.[56]

4.7 Division and square root

Division and square roots are normally implemented in a straightforward fashion using the idea of generating inverse functions and a multiplier. Figure 4.40 shows how a division can be implemented using this technique by employing a multiplier in the feedback loop of an amplifier. In the very same way a square root function may be implemented by connecting the y input to the output of the circuit shown here.

4.8 Comparators

A number of problems need some means to implement step functions or to select from various value sources etc. Operations like these require some kind of a decision element which is called *comparator* in an analog computer context. Basically, a comparator is just an operational amplifier with biased diodes in its feedback loop thus limiting its output to typically only two possible values like +1 and −1.

Figure 4.42 shows the general principle of a classic comparator.[57] At its heart is an amplifier with typically two inputs weighted by resistors R_1 and R_2 (usually $R_1 = R_2$

[51] See [Hitachi 505E][p. 16].
[52] See [Morrill 1962][p. 3-40].
[53] Cf. [Morrill 1962][p. 3-41].
[54] See [Savet 1962].
[55] Cf. [Haug 1960][p. A7].
[56] See [Analog Devices 2008].
[57] Cf. [Giloi et al. 1963][p. 80]. The S denotes the summing junction input of the amplifier.

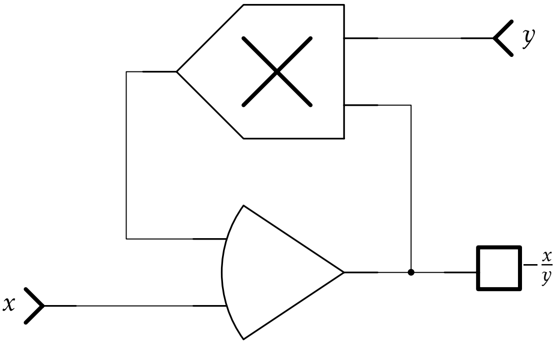


Figure 4.40: Division as inverse function of multiplication

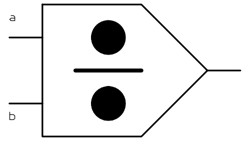


Figure 4.41: Symbol for a divider

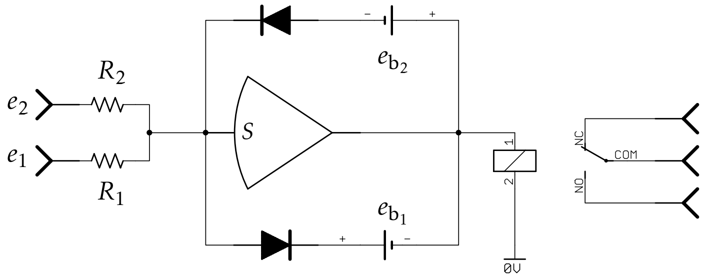


Figure 4.42: Principle of operation of a classic comparator

holds). The feedback loop of this amplifier consists of two diodes biased by e_{b_1} and e_{b_2}.[58] These diodes limit the output signal to only two values which can be used to directly control a relay. If the sum of the two input signals e_1 and e_2 is greater than zero the output will be driven to its upper limit and while the lower limit will be reached for a sum less than zero. Thus a comparator can be readily used to compare two or more signals in size.

Small early tabletop computers often did not provide comparators as such – the operational amplifiers were too precious to be committed to a specific function like this. Instead their patch panel often features a couple of diodes that can be used to setup a comparator from existing computing elements according to the circuit shown if necessary. Such comparators directly driving a relay are called *relay comparators*.

Later and bigger machines always have a number of comparators available but break with the habit of controlling a relay directly. Instead the comparators yield a digital output signal that can be used for a variety of purposes in an analog computer setup. Such comparators are called *electronic comparators*. Figure 4.43 shows the symbol used to denote a relay comparator in a diagram while the symbol for an electronic comparator is shown in figure 4.44.

[58]Of course these bias sources are not actual batteries in a real circuit – they can be readily implemented using a coefficient potentiometer with its wiper connected to the diode and both inputs connected to the amplifier's output and ±1 respectively.

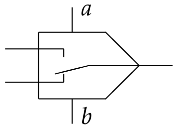


Figure 4.43: Symbol of a relay comparator

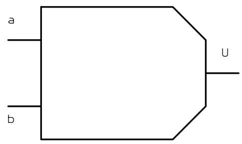


Figure 4.44: Symbol of a comparator

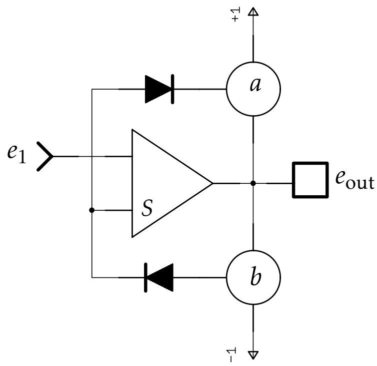


Figure 4.45: Simple limiter circuit

4.9 Limiters

Limiters are used to limit the value of a variable which is often necessary in simulations of mechanical systems where moving parts can hit stops etc. The basic circuit of a limiter is similar to that of a comparator. By using appropriately biased diodes connecting the output and the summing junction of an amplifier, its output signal can be limited to an upper and a lower bound. Figure 4.45 shows the structure of a basic limiter.[59]

4.10 Resolvers

So-called *resolvers* are among the most complex computing elements in an analog computer and normally only large systems offer resolvers as stand-alone units. The purpose of a resolver is to generate the sine and cosine of a variable and convert polar coordinates into rectangular ones and vice versa. Alternatively resolvers also allow the rotation of coordinate systems. One of the earliest resolvers described in literature is the device developed by LOVELL et al. as described in chapter 3.3. This device is a typical *servo resolver* – this type of resolver was used widely in aerospace applications where coordinate transformations are abundant and the rates of change of the variables to be transformed are normally reasonably slow to be suitable for an elec-

[59]This simple circuit is often sufficient although the slope of the limited portion of the signal is not zero. See [GILOI et al. 1963][pp. 207 ff.] and [KORN 1962][pp. 3-62 ff.] for implementation variants yielding more precise results.

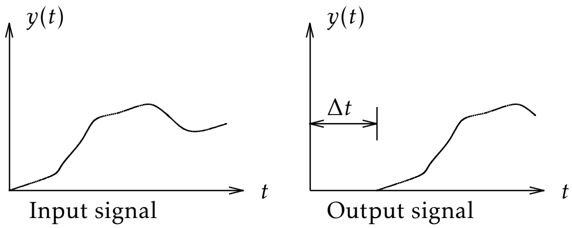


Figure 4.46: Behavior of an ideal time delay

tromechanical approach. Late analog computers also feature all-electronic resolvers allowing high-speed simulation of complex mechanical systems etc.

Resolvers are rather complicated and there are too many implementation variants to be described in detail here. [Carlson et al. 1967][pp. 58 ff.], [Giloi et al. 1963][pp. 95 ff.], [Loveman 1962][pp. 3-2 ff.], [Morrill 1962][pp. 3-56 ff.], [Kley et al. 1966] and finally [Vogel 1977] contain details and implementation examples.

4.11 Time delay

Many complex simulations require some form of time delay to model transport, diffusion effects etc. An ideal time delay unit would act as shown in figure 4.46: An input signal is delayed by a user selectable value Δt, so given an input signal $y(t)$ the output $e_o(t)$ of the time delay unit is given by $e_o(t) = y(t - \Delta t)$.[60]

There have been many different approaches to achieve a near ideal time delay for use in an analog computer setup. The most basic idea from today's point of view is to digitize the input signal, store it in a digital memory system and read it out with the selected delay. Based on these readout values, an analog signal is generated by a *digital-analog converter*.[61] Although this yields very good results and is quite independent of the input signal form[62] it became feasible only in the late 1970s when *RAMs*[63] became cheap enough to be used in such special purpose applications. Figure 4.47 shows a typical digital time delay unit made by EAI.

Accordingly, earlier implementations had to rely on other technologies such as tape recording systems[64] or capacitor storage systems. The latter are based on a wheel or drum which houses a set of capacitors oriented radially and sharing a common connection at the axis while the other terminal of the capacitors are connected to contacts on the surface of the drum or wheel. These connections are then touched by two brushes: One brush is used to store a value into a capacitor while the other is connected to a readout amplifier. Δt is determined by the angular velocity of this capacitor wheel.

[60] During start-up time, $t < \Delta t$, the output is usually zero.
[61] *DAC* for short.
[62] Apart from questions regarding the sampling period and the resolution of the converters.
[63] Short for *Random Access Memory*.
[64] Cf. [Kennedy 1962][pp. 6-10 ff.].

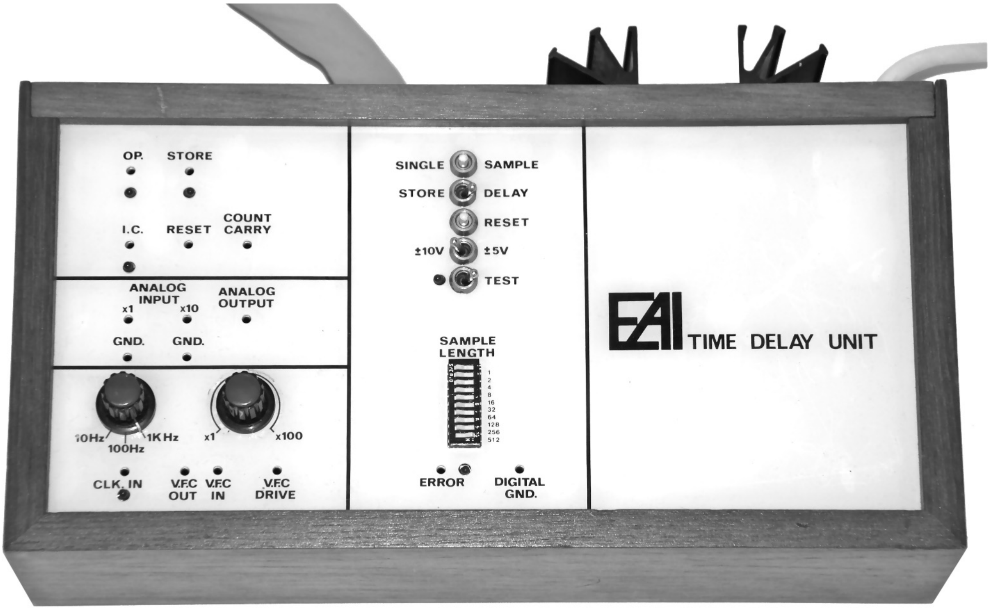


Figure 4.47: EAI time delay unit

The main disadvantage of systems like this is the very limited number of storage elements which effectively limit the number of breakpoints where the function to be delayed is sampled.[65] Yet another approach is based on the curve follower.[66] The function to be delayed is plotted by a strip-chart recorder. The paper containing the plot is then continuously fed into a one-dimensional curve follower which in turn delivers the delayed function.[67]

In cases where the substantial expenses of a dedicated time delay system could not be justified, delay functions were set up using so-called Padé *approximations* or a Stubbs-Single *approximation*. In Laplace-transform notation delays of these types are characterized by the following general linear transfer function:

$$\frac{Y(s)}{X(s)} = \mathrm{e}^{-Ts}$$

To give an impression of the complexity of these approaches, figure 4.48 shows the implementation of a 4th-order Stubbs-Single approximation which is based on the transfer function

$$S_4(s) = \frac{1 - \frac{1}{2}Ts + \frac{15}{134}T^2s^2 - \frac{13.55}{1072}T^3s^3 + \frac{1}{1072}T^4s^4}{1 + \frac{1}{2}Ts + \frac{15}{134}T^2s^2 + \frac{13.55}{1072}T^3s^3 + \frac{1}{1072}T^4s^4}.$$

[65] [Kennedy 1962][pp. 6-8 ff.].

[66] Cf. section 4.5.2.

[67] See [Carlson et al. 1967][p. 225].

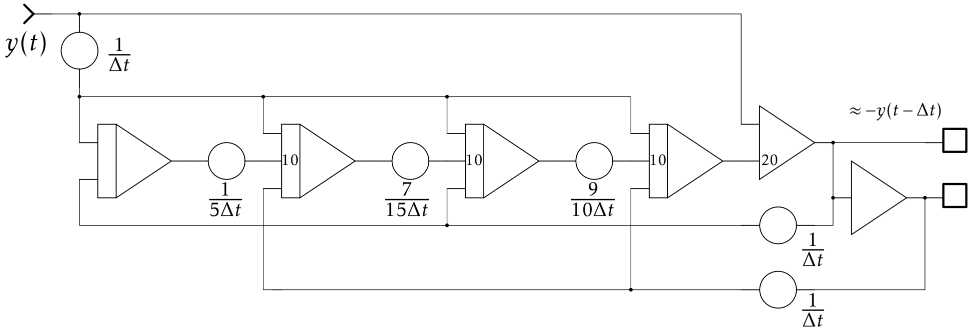


Figure 4.48: 4th-order Stubbs-Single approximation for time delay

The amount of circuitry necessary to implement time delays like this is significant as can be seen and the setup of these circuits is quite time consuming due to the many coefficient potentiometers depending on Δt. More details and examples regarding Padé and Stubbs-Single approximations can be found in [Kennedy 1962][pp. 6-5 ff.], [Carlson et al. 1967][pp. 225 ff.] and [Stubbs et al. 1954].

4.12 Random noise generators

Problems which are normally described by random processes can also be subject of an analog computer simulation if a suitable random signal source is available.[68] Typical applications are the analysis of correlation functions and delay errors in signal networks, mechanical system analysis, the measurement of spectral densities and even optimization processes by sequential random perturbations and the like.

Typical random sources employed in analog computers are based on the decay of radioactive sources and – more often – on resistor noise or PN-junction noise in semiconductors. Often complex demodulation schemes and special function generators are employed to yield a desired probability distribution of the random signal.[69]

4.13 Output devices

Since an analog electronic analog computer works on values normally represented by voltages, typical output devices are strip-chart recorders as shown in figure 4.49, *xy*-recorders (see figure 4.29) and oscilloscopes to display rapidly changing values.

In cases where a graphical output is not necessary, *digital voltmeters*[70] are used to read out various values of a simulation run. Large systems often feature a built-in digital

[68] See [Rideout 1962], [Korn & Korn 1964][pp. 12-5 f.] and [Giloi et al. 1963][pp. 357 ff.] for application examples and techniques.

[69] See [Korn 1962][pp. 81 ff.].

[70] *DVM* for short.

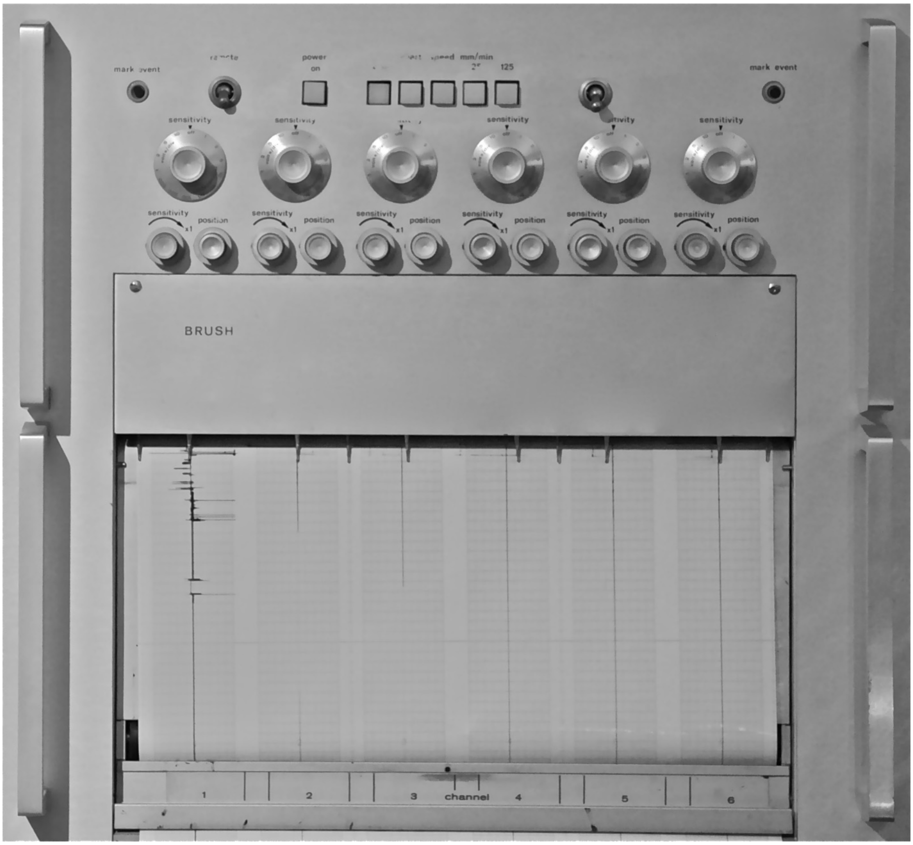


Figure 4.49: Brush six channel recorder

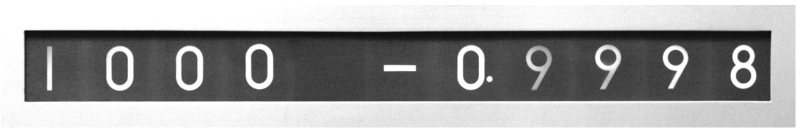


Figure 4.50: Readout of integrator I000 on a Telefunken RA 770 analog computer

voltmeter with an automatic selector.[71] To get a snapshot of the current state of a computer run, all integrators are placed into hold mode first. Then the output value of every computing element (or a subset of these) is read out by the digital voltmeter and displayed or printed in decimal form. Alternatively, these values can be transmitted to an attached stored-program digital computer or can be stored on some digital medium.

[71] This selector was based on stepping relays in early installations which were replaced later by readout relays located on every computing element. These readout relays can be used to connect the output of every element to a central readout bus that is in turn connected to the input of the digital voltmeter.

Figure 4.50 shows the display of a typical precision digital voltmeter which is part of a Telefunken RA 770 analog computer. The left half of the display[72] shows the coordinates of the computing element selected for readout: I000 denotes integrator number 000. The output value of this particular element is currently +0.9998. It is to be noted that most of the digital voltmeters used in analog computers display their values with respect to the machine unit of the system, so the above value corresponds to +9.998 V since the RA 770 is a ±10 V system.

The following chapter will now show the anatomy of a typical medium sized analog electronic analog computer to give an impression of the interaction of the various computing and control elements described in the preceding sections.

[72]This particular display is of the projective type, i. e., every display position can be illuminated by one out of a number of incandescent bulbs which are positioned behind a mask containing the various symbols to be displayed.

5 Analog computer anatomy

The system chosen exemplarily for the following sections is the *EAI-580*, a typical medium sized analog electronic analog computer which is shown in figure 5.1.

5.1 Analog patch panel

The most prominent feature of any analog computer is its so-called *(analog) patch panel* which contains up to several thousands of connectors into which patch cables can be plugged to interconnect the various computing elements. Depending on the precision of an analog computer, these connectors and patch cables may be non-shielded or shielded cables. Since the process of patching an analog computer is quite time consuming which would make it impossible to change programs rapidly, the patch panels of all but the smallest analog computers are exchangeable.

Figure 5.2 shows the analog patch panel of the EAI-580 system. This panel is of the modular type, it contains four rows of computing elements which consist of 15 com-

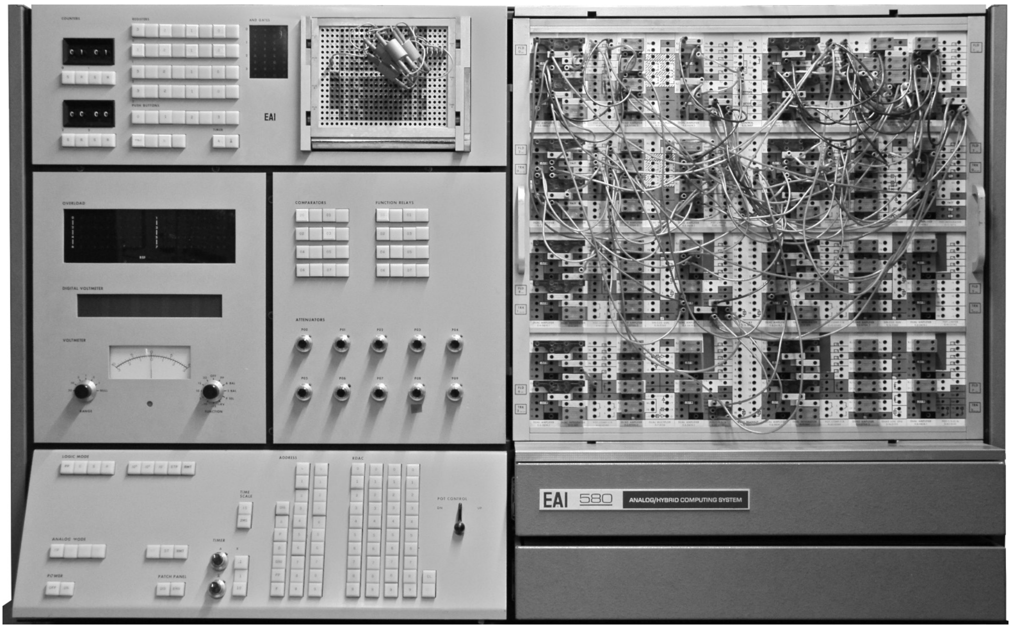


Figure 5.1: EAI 580 analog computer

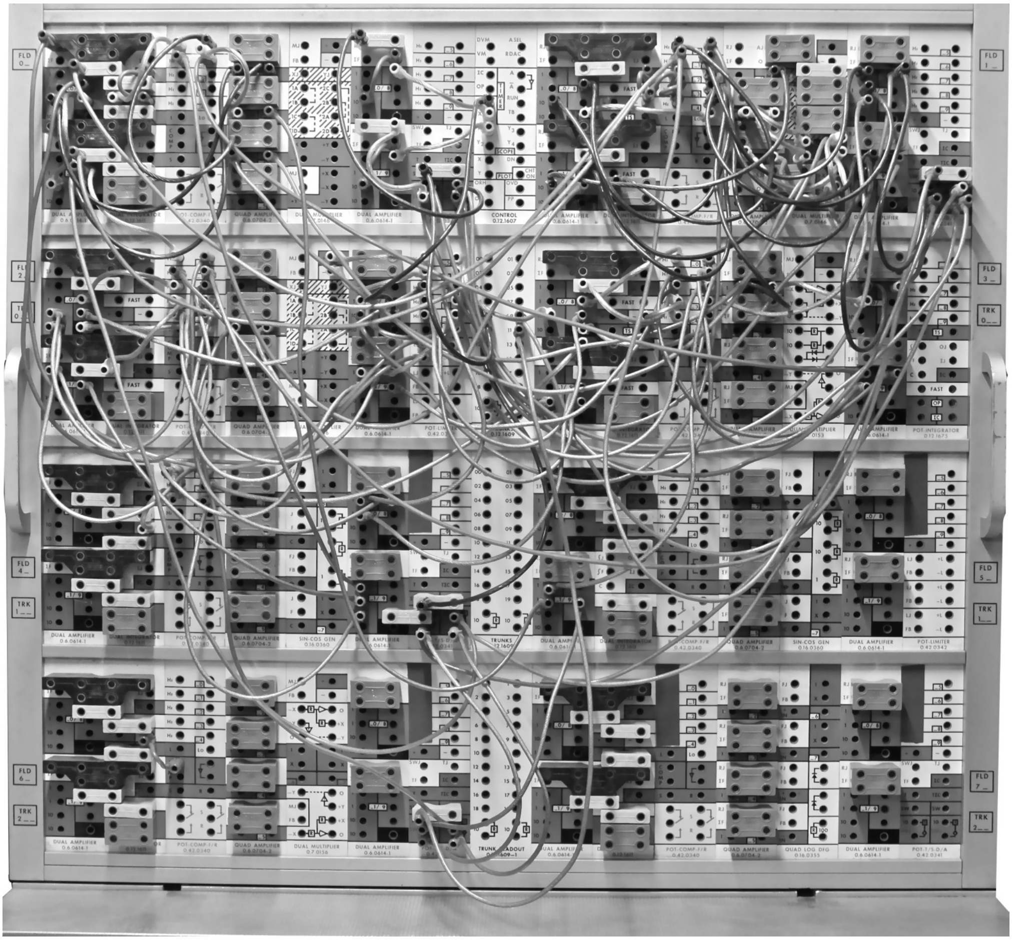


Figure 5.2: EAI 580 analog patch panel

puting elements each. Every computing element is represented by a particular inset featuring 2 columns and 12 rows of connectors. Typical computing elements are double operational amplifiers which are usually paired with double integrator circuits,[1] quadruple amplifiers which are often used in conjunction with square function generators to form multipliers or act as simple inverters, comparators, function generators, *trunk lines*[2] etc.

[1] In contrast to other systems, the EAI-580 requires the programmer to setup an integrator by explicitly interconnecting an operational amplifier with a dedicated integrator circuit. This approach has the distinct advantage of giving maximum freedom to the programmer. On the downside, this setup requires to see the machine not as a strictly mathematical instrument but more as a fascinating construction kit for engineers and scientists.

[2] A trunk line is a direct connection between a patch panel connector and a connector on the back of the machine where input/output-devices may be connected in a convenient way without further cluttering the analog patch panel.

Figure 5.3: EAI 580 function generators

5.2 Function generators

Located below the analog patch panel are two drawers containing four diode function generators each. Each of these function generators has ten diode segments with adjustable breakpoint and slope.[3] Figure 5.3 shows the extended lower drawer.

In general, there are two basic variants of diode function generators: Those with adjustable breakpoints and those with fixed, equidistant breakpoints. The former implementation has the advantage that the breakpoints for a function approximation can be clustered in regions where the function oscillates heavily or is otherwise “interesting”. This approach has the disadvantage of a rather long and complicated setup procedure since both, breakpoints and slopes, must be set manually. The second implementation variant, which is quite prevalent in machines built by the German manufacturer Telefunken, simplifies the setup by omitting the variable breakpoints but sacrifices precision of the function approximation in some cases. Eventually, it is mainly a matter of personal preference which implementation is preferred.

[3]Cf. figure 4.31.

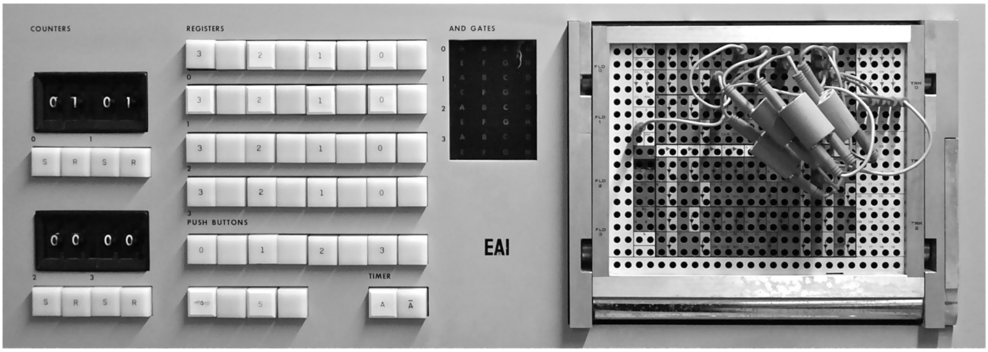


Figure 5.4: EAI 580 digital controls and patch panel

5.3 Digital patch panel and controls

The topmost element of the left half of the EAI-580 contains a digital patch panel. In addition to that, this unit contains two thumb wheel switches to preset digital counters/dividers and a number of switches which display the status of the digital registers and can be used to set or reset individual bits. The remaining switches are associated with a number of flip-flops and a timer.

These digital elements, which are basically simple logic gates, inverters, counters, and dividers, can be interconnected by means of patch cables plugged into the removable digital patch panel. In many cases a computation performed with an analog computer does not require any digital elements at all, but more complex tasks, especially those involving decision circuits or the optimization of parameters of complex systems, can benefit substantially from digital elements. A typical example might be the simulation of an automatic transmission for a car: The behavior of the gears, the torque converter, etc. can be simulated readily by the analog part of the computer. The control of these transmission parts, the decision of switching gears under varying loads, etc. is then easily implemented with the digital elements of a digital unit like this.

The inputs to these digital elements are either clock signals generated by a central timing unit or the outputs of comparators used in the analog setup, manual switches, etc. Normally, the digital outputs are then used to control electronic switches on the analog side[4] or to control the operation of integrators or groups of integrators. Some tasks like optimization often require at least two groups of integrators which are controlled in an alternating fashion: One integrator group is halted and delivers initial conditions for a second group which will step through the modes *initial condition* and *operate*. After some time this second group might be set to hold to read out the results. The next step would then put the integrators of the first group into compute mode for some predetermined time or until some condition is met. Then this group is placed into hold mode again and the second group will start with new initial conditions etc.[5]

[4]Using such switches differing gear ratios can be selected in the automatic transmission example.

[5]Some machines like the Telefunken RA 770 feature an intricate control timing system that allows operating modes like this to be implemented without having to resort to individually patched digital control elements.

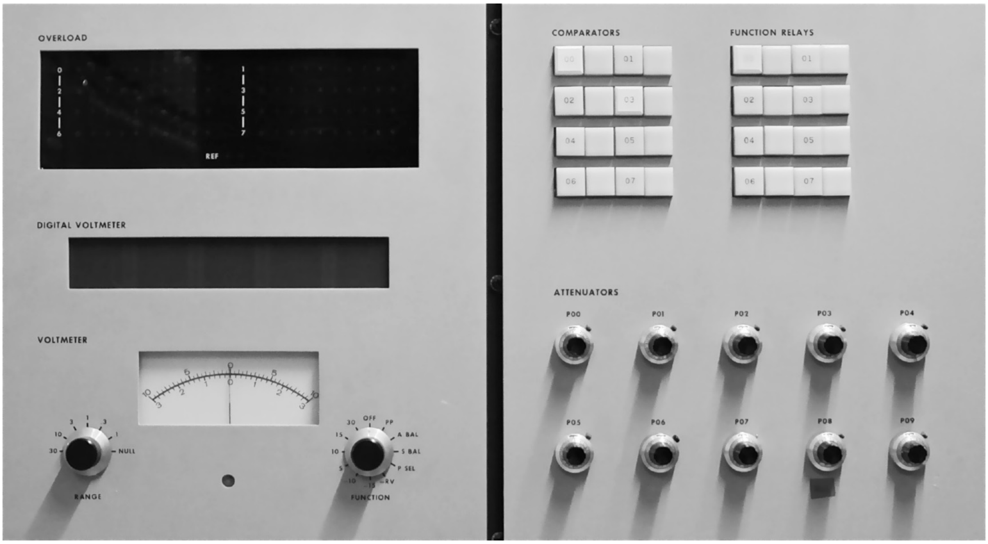


Figure 5.5: EAI 580 readout, overload display, manual potentiometers and manual switches

5.4 Readout

Below the digital control unit is the overload display and readout system on the left and a number of manual coefficient potentiometers and switches associated with a number of comparators and function relays on the right hand side. Figure 5.5 shows these subsystems of the EAI-580 analog computer in detail.

The overload display on the upper left has one incandescent light bulb for each amplifier in the system to display overload conditions. There are two basic types of overload that may occur in an operational amplifier: In both cases it will be unable to deliver the necessary output voltage required to satisfy the condition that the inverting summing point will be maintained at (near) ground potential. Taking a summer with two inputs e_1, e_2 and associated weights $a_1 = a_2 = 1$ as an example, this error condition could result from the attempt to add two input voltages of +1 machine unit each. This would require the output to yield a voltage representing +2 machine units which is obviously not possible.[6] The second type of error that would yield an overload condition is to connect the output of an operational amplifier to a load with a resistance requiring an output current of the output amplifier in excess of its capabilities.[7]

When chopper stabilized amplifiers are used, the detection of both overload conditions is equally simple: In each case the inverting summing junction can no longer be maintained at (near) ground potential. This will in turn yield an excessive error correction signal at the output of the stabilizing amplifier which is normally used to trigger the corresponding overload light for this amplifier.

[6] See chapter 4.

[7] Connecting the output of an amplifier to ground or to ±1 machine unit would result in such an overload condition as would feeding too many following computing elements from a single operational amplifier's output.

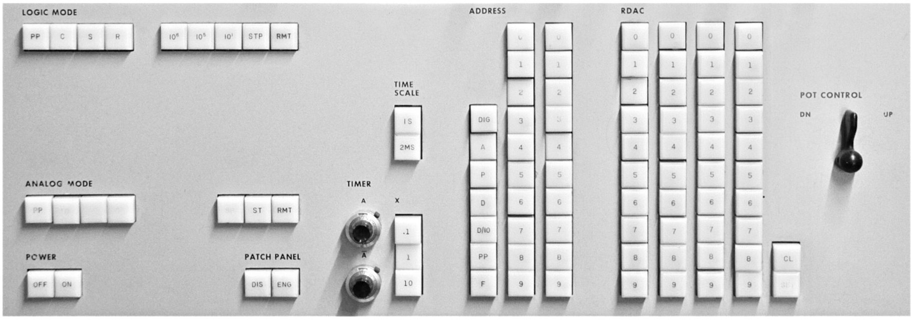


Figure 5.6: EAI 580 control panel

Since an overload condition normally renders a computation useless, it will in most cases place the analog computer into hold mode. Then the programmer can check the various program variables to determine the reason for the overload.

Below the overload indicator panel is the display of a four digit digital voltmeter which can be used to read out the values of program variables.[8] In most cases digital voltmeters like this can be connected to a printer and a selector unit to facilitate automatic readout and logging of a number of variables.

On the bottom left is an analog voltmeter which can also be used to determine the value of program variables as well as for a quick checkout of the computer. By means of the rotary switch on the right all power supply voltages can be checked easily. In addition to this, the voltmeter is used for manual balancing of the operational amplifiers.[9]

The right half of this panel contains a couple of switches connected to the comparators and function relays of the analog computer. Below these elements are ten manual ten-turn coefficient potentiometers, called *attenuators*.

5.5 Control

The unit on the lower left of the computer is the central control panel which is used to control the overall operation of the computer and for setting the servo coefficient potentiometers. Figure 5.6 shows this control panel in detail.

On the far left are, from bottom to top, the power switches, four control switches to select the mode of operation of the analog part and similar switches for the digital part. The possible analog modes are the following (from left to right[10]):

PP: Short for *Program Panel*, this mode transfers the overall control of all integrators to the digital unit on the top left.

[8] Normally, the computer is placed into hold mode for readout.

[9] In some cases the drift of an amplifier is so excessive that the chopper stabilization stage cannot compensate it completely. Therefore each amplifier provides a potentiometer for manual coarse zeroing.

[10] The key labels have faded away due to the age and usage of the particular machine depicted.

IC: Place all integrators into initial condition mode.

HD: All integrators are placed into hold mode.

OP: Switch all integrators into operate mode.

The switches for the logic mode are used to transfer the control of the digital elements to the digital patch panel (PP) and to clear (C), stop (S) and run (R) the digital circuit setup on the digital patch panel. The switches labeled 10^6, 10^5, 10^1 and STP select the basic digital clock rate, one pulse per microsecond being the default cycle time, or place the digital subsystem into single step mode. The RMT switch places the digital unit into remote mode which allows several EAI-580 systems to be coupled together or to couple an EAI-580 analog computer with a memory programmed computer, basically forming a hybrid computer.[11]

The two switches labeled DIS and ENG are used to disengage and engage the large analog patch panel. Due to its many connections, its locking mechanism is driven by a motor which is controlled by these two switches.

Above these two switches a row of three buttons labeled SP, ST and RMT is located, that place the machine into the potentiometer-set mode (SP), a check mode called *static test*[12] or into remote mode which transfers the control of the analog portion to an external system like a memory programmed computer.

The column next to these controls contains two switches to select the basic time scale of the analog portion of the computer and some timer controls. By default all integrators are set to a time scale of 1 second, so that an input voltage corresponding to one machine unit will yield a minus one machine unit output signal after one second. Pressing the button labeled 2MS will exchange all capacitors in the feedback loops of the integrators by capacitors with 1/500 of the basic capacity thus speeding up the computation by a factor of 500. This mode is very useful for the study of very slow processes.

The two ten-turn potentiometers control the timers for *repetitive operation*. In this mode a computation will be performed over and over again, cycling the integrators through their initial condition, operate and hold modes automatically. This mode allows to generate a stable display on an oscilloscope showing the solution of a problem. The keys labeled .1, 1 and 10 set a global timer multiplication factor that affects the repetitive operation.

The next column of switches is labeled ADDRESS and allows to select any computing element of the analog section of the EAI-580 for readout or setup in the case of servo potentiometers. Each element has a two digit decimal address and a type representing its function like A for an amplifier, P for a potentiometer etc. The value at the output of an element addressed in this way is displayed on the digital voltmeter.

The next block of switches is labeled RDAC and forms a simple, yet precise resistor based analog-digital converter. Its main purpose is to set selected servo coefficient

[11] See section 9.

[12] This mode is quite similar to the initial condition, but instead of initial conditions check voltages can be applied to the integrators allowing a static check of a program without actually running it.

potentiometers: Using the ADDRESS switches, a potentiometer to be set is selected. In the next step the desired value is keyed into the RDAC keys and the SET key is pressed. This causes the servo motor of the selected potentiometer to run until the value is reached. In some cases it is desirable to control a servo potentiometer completely manually which can be done with the big lever on the far right. Turning it to the left will cause the servo motor of the selected potentiometer to decrease the coefficient's value; turning it to the right will increase the value.

5.6 Performing a computation

Performing a basic computation on the EAI-580 analog computer is now performed as follows:

1. First the computer is switched on by pressing the ON button.
2. Then the programs prepared on the analog and digital panels have to be inserted into the machine. While the digital patch panel can be engaged and disengaged manually by means of a lever, the analog patch panel is moved by a motor controlled by the DIS and ENG switches on the control panel.
3. According to the program description the (servo) coefficient potentiometers have to be set.
4. If necessary, the diode function generators must be setup manually.
5. If necessary, initial conditions for the digital computing elements have to be set by setting the switches associated with the counters, flip-flops etc. accordingly.
6. The next step involves setting the desired time scale if repetitive operation is necessary.
7. Selecting the modes *initial condition* and *operate* through the switches controlling the analog mode will then start the computation. If the program needs digital control elements, the digital subsystem must be placed into the run mode, too.

6 Typical systems

The following sections describe some typical analog computers without even attempting to give a comprehensive overview since there were too many companies and computer models in the market to be covered in detail here.

6.1 Telefunken RA 1

In the early 1950s Dr. Ernst Kettel started the development of an analog computer at the German company Telefunken which was known before for its high-quality radio transmitters, receivers, and the like but not for computing machines at all. It seems reasonable to assume that Dr. Kettel met Helmut Hoelzer during his work in Peenemünde where he developed radio control systems for rocketry. There he has probably seen Hoelzer's general purpose analog computer which would explain his later interest in this topic.

However, Dr. Kettel started the development of a vacuum tube based analog computer in about 1953 with Dr. Adolf Kley. The resulting machine became known as the *RA 1*[1] and is shown in figure 6.1.

[Feilmeier 1974][p. 18] describes the RA 1 as a typical system of the so-called *heroic age* of analog computing that spans from about 1945 until 1955. The machine is of a completely modular design. Three racks hold a variety of computing and support elements. The right most rack contains the power supplies delivering the highly precise machine voltages of ±100 V as well as the main supply voltages of ±200 V etc. The middle and left racks hold the various computing elements.

The rack on the left contains the following subsystems from top to bottom: A vacuum diode function generator with 21 fixed breakpoints and and 21 precision potentiometers and switches to select the slope, four quarter square multipliers[2], eight summers, an oscilloscope of rather interesting design with two display tubes, a collection of limiters and the like and some *free diodes*[3], relays to setup comparator circuits, five coefficient potentiometers, a selectable integrator/summer/amplifier and a summer.

[1] *RA 1* is the abbreviation for *Rechner, Analog* number one. ("Rechner" is the German term for computer.) Telefunken followed a simple scheme for naming its analog computers which were all denoted by *RA* followed by a number. Later tabletop machines were labeled with an additional *T* to distinguish these from the larger systems.

[2] Of these eight units, seven employ double diode vacuum tubes EAA 91 to implement the square functions while one module is of a clearly experimental nature: This left most multiplier uses germanium diodes instead of vacuum tubes.

[3] Many analog computers contain so-called *free* or *uncommitted* diodes which are very useful to limit the output of some computing element, to implement absolute value functions, etc.

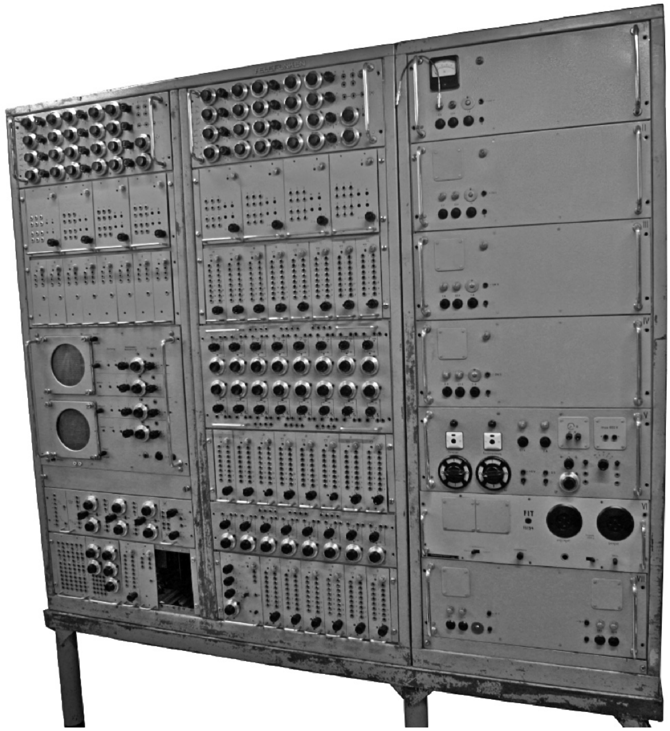


Figure 6.1: The Telefunken RA 1 as it is currently preserved in a private collection

The rack in the middle contains (top to bottom) a second diode function generator, four quarter square multipliers, eight selectable integrators/summers/amplifiers, 16 coefficient potentiometers, another eight switchable integrators/summers/amplifiers, eight coefficient potentiometers and a drawer containing the central timing and control module as well as six selectable integrators/summers/amplifiers.

It is quite obvious that the RA 1 was built as a piece of laboratory equipment and not as the prototype of a product to be sold to customers. All computing elements are interconnected with shielded patch cables but there is no central patch panel, there is no means to setup coefficient potentiometers, there is no built-in voltmeter etc. It must have come quite as a surprise for Telefunken that customers who saw this machine in the laboratory in Ulm were eager to buy such a machine. Forced by such customer demand, Telefunken agreed to build a batch of ten systems which were slightly improved over the prototype RA 1. A very large installation of this production model, named *RA 463/2* is shown in figure 6.2.[4] Since the first batch was sold even before the machines were actually built, additional systems were manufactured and Telefunken entered the analog computer business.

[4]See [Telefunken 1958]. A typical RA 463/2 system, containing only three racks and no operator console table, was priced at 158,000 DM in 1958 (see [Telefunken 1958]). This corresponds to about 410,000 US$ today.

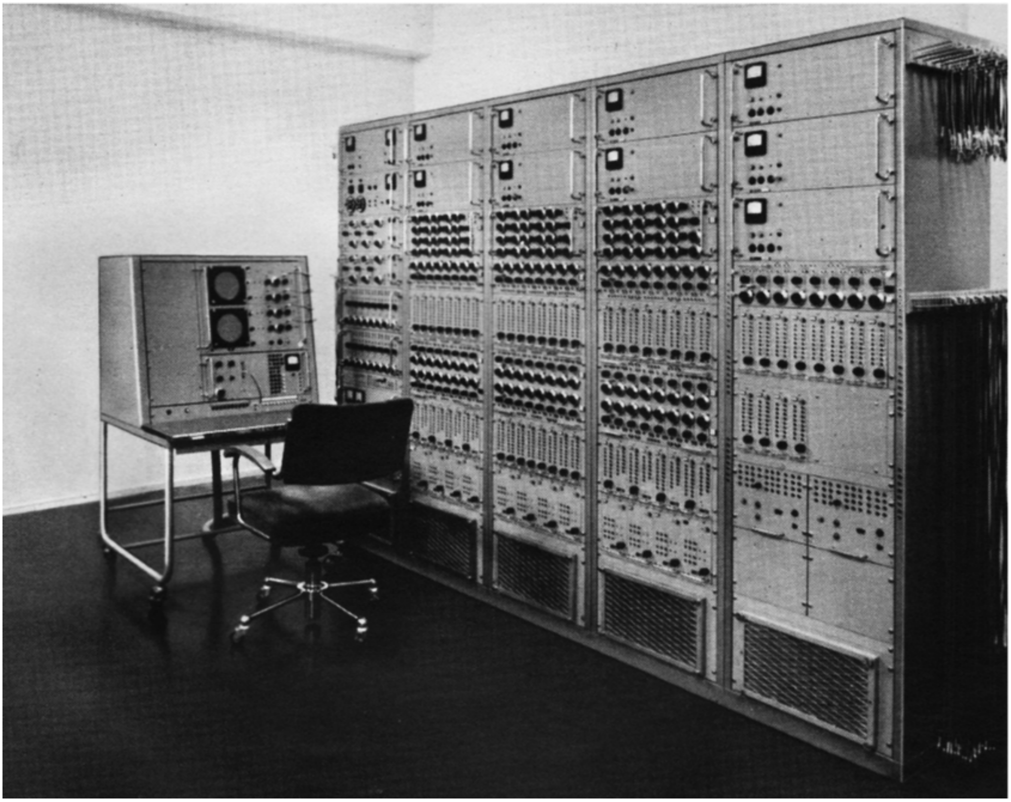


Figure 6.2: A large Telefunken RA 463/2 analog computer installation (see [Ammon][p. 1])

6.2 EAI 231R

The Electronic Associates *EAI 231R* is one of the machines that became archetypal for analog computers as such. It was introduced in 1959 and already had a long ancestry which made this system far superior compared to the Telefunken RA 1 and its production model RA 463/2. The 231R and it predecessors were part of the *PACE*[5] series which included the earlier models 16-24A, 16-24D, 16-31R and 16-131R.

A typical EAI 231R analog computer is shown in figure 6.3. This system became one of the most widespread analog computers and systems of this type were used well into the late 1970s. The system is dominated by the large patch panel on the right side containing 3450 plug sockets. These sockets as well as the patch cables are completely shielded minimizing cross talk and noise injection. Below the patch panel is the operator control panel which contains 20 manually operated coefficient potentiometers on the right and the timing control, analog voltmeter, and computing element selection circuitry on the left. The two large bays on the top of the system can hold either

[5]Short for *Precision Analog Computing Equipment*.

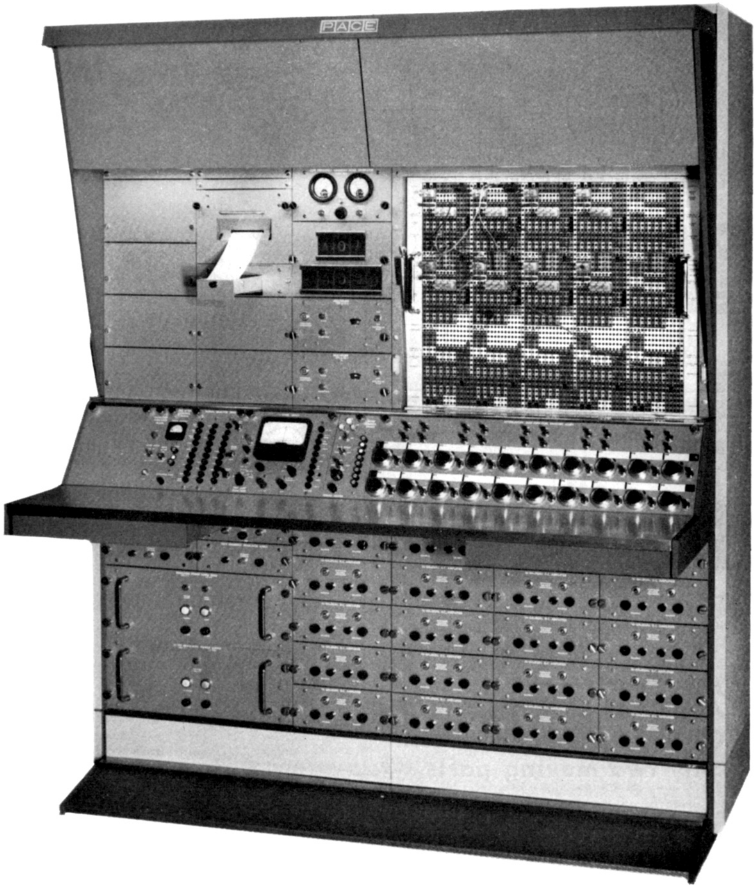


Figure 6.3: EAI 231R analog computer

additional 80 manual coefficient potentiometers or servo potentiometers.[6] Above the operator panel are a random noise generator (recognizable by its two circular analog instruments), a digital voltmeter, two servo multipliers and a printer connected to the DVM.[7] The bays below the operator console contain 24 drawers with four operational amplifiers each and two power supplies. All input and feedback resistors and capacitors are contained in a temperature controlled oven to minimize drift effects due to temperature variations. Figure 6.4 show the removable patch field. The system depicted has manual coefficient potentiometers installed in the top enclosures and an extension rack mounted to its right.

A rather large EAI 231R installation is shown in figure 6.5. It consists of two maximum extended EAI 231R analog computers, each with one extension rack to its left and one to its right. A large horizontal plotter can be seen in the left foreground, in the mid-

[6]The system shown here is equipped with servo potentiometers.
[7]Short for *Digital Voltmeter*.

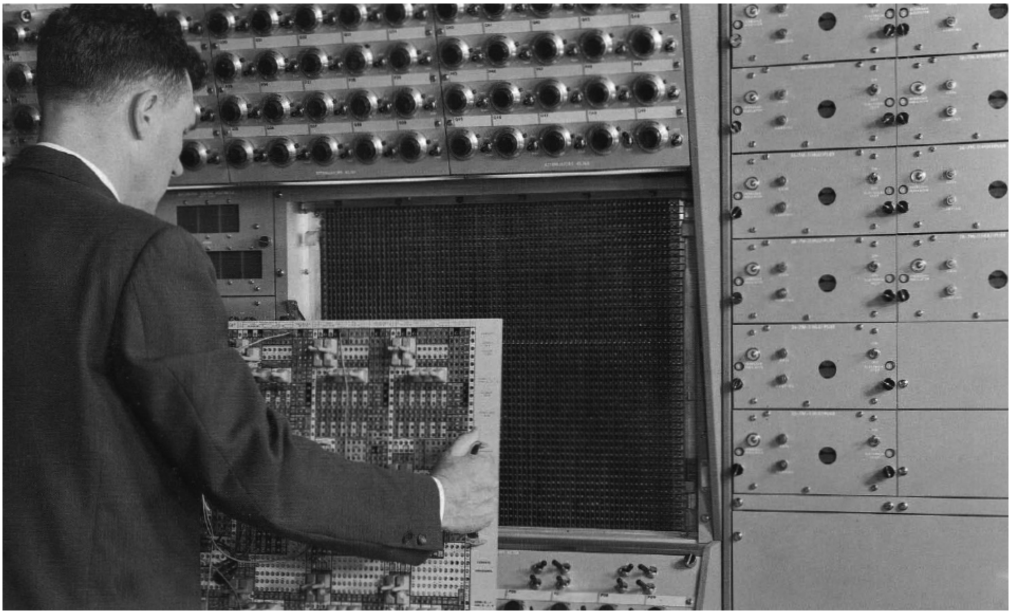


Figure 6.4: Removing the patch field from an EAI 231R computer (see [EAI PACE 231R][p. 6])

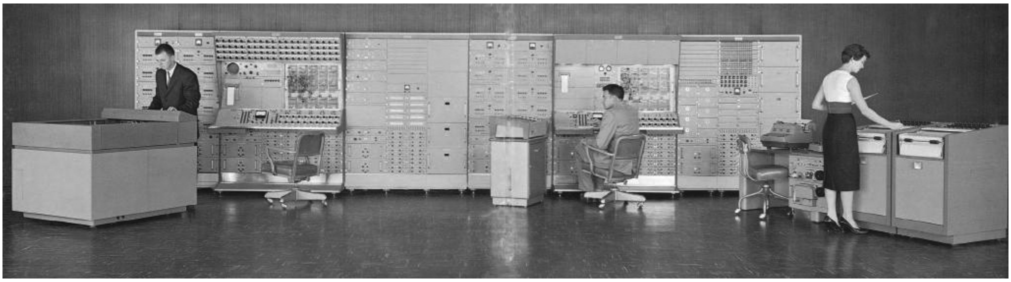


Figure 6.5: Installation of two EAI 231R with various plotters and an ADIOS console

dle of the picture is a smaller EAI *Variplotter* and on the right are two multi-channel recorders and a so-called *ADIOS*[8] console which can be used to automatically setup servo potentiometers and the like under paper tape control.

The 231R was such a successful system that EAI developed an enhanced version, the EAI 231RV, that was put on the market in 1964. In contrast to the 231R, it contains a rather sophisticated digital control system that allows individual integrator control and easy implementation of complex control sequences. In addition to that, the integrators were expanded with additional time constants. These systems found widespread use in nearly all scientific and engineering branches but most prominently in aerospace applications.

[8]Short for *Automatic Digital Input Output System*. Some background information about ADIOS can be found in [Van Wauve 1962].

Figure 6.6: EAI HYDAC-2000 system (see [N. N. 1964/1])

Another impression of a large EAI installation gives figure 6.6. The right half is occupied by an EAI 231RV with expansion racks to its right and an eight channel recorder. On the left a DOS 350 digital console can be seen. This system consists of a plethora of basic digital circuit elements that can be connected by means of a central patch panel. Using these elements, complex control circuits can be implemented. This combination of an EAI 231RV and an associated EAI DOS 350 was marketed as the *HYDAC 2000*[9] by EAI.

6.3 Early transistorized systems

Telefunken turned out to be the first company that successfully developed a fully transistorized analog computer. After the success of the vacuum tube based RA 463/2 it was decided to quit the vacuum tube approach and start over from scratch, developing a transistorized precision operational amplifier with chopper stabilization.[10] Hired for this development task was Günter Meyer-Brötz who just finished university.[11]

As a proof of concept, a small tabletop analog computer was built that is shown in figure 6.7. Its connection to the RA 1 and RA 463/2 can be seen – there is no central

[9]Such a HYDAC system, worth over one million Dollars, was acquired by General Dynamics in 1964 for the development of the F-111 jet fighter (cf. [N. N. 1964/2]).

[10]The amplifiers in the RA 1 and its production model RA 463/2 had no drift compensation and thus these machines were capable of repetitive operation only with a maximum integration time of 110 seconds. Longer runs would compromise the results too much due to drift effects.

[11]Prof. Dr. Meyer-Brötz told the author that he was hired without having any real experience in the field of amplifier development because he did not know that it was considered impossible to build a fully transistorized precision operational amplifier. Nearly everybody in the 1950s considered transistors to be sufficient for switching operation but not for precision analog applications.

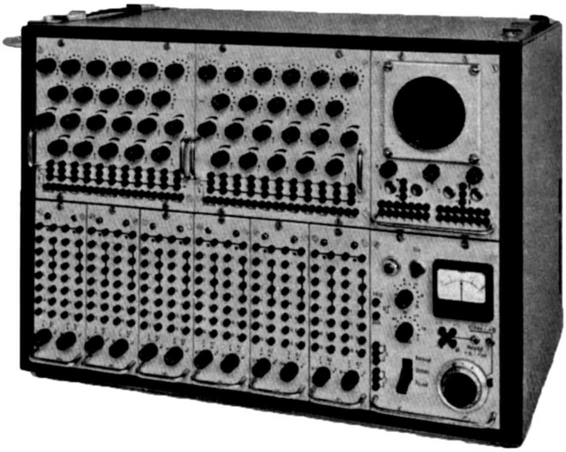


Figure 6.7: First fully transistorized tabletop analog computer made by Telefunken (see [Ernst 1960][p. 255])

patch panel, all computing elements are housed in separate drawers and the machine features a built-in oscilloscope.

Based on this prototype it was decided to develop a production model which would become the *RAT 700*.[12] The operational amplifier used in this first fully transistorized analog computer has been shown in chapter 4, figure 4.13. The RAT 700 was sold for the first time in 1959 and became the progenitor of a rather long series of analog tabletop computers built by Telefunken. Figure 6.8 shows one of the machines sold in 1959.[13]

In contrast to the RA 1 and RA 463/2 there is a central patch panel[14] located in the lower of four drawers which contains the input and feedback networks of the summers and integrators and additional computing elements like free diodes, relays etc. On the right hand side of this drawer the operator control panel can be seen. The middle of the machine holds two half-height drawers containing 20 precision coefficient potentiometers[15] and two diode function generators with fixed breakpoints. The top drawer holds the precision power supplies on the right and 15 chopper stabilized precision operational amplifiers on the left.

[12]Short for *Rechner Analog Tisch – analog tabletop computer*.

[13]Later models had a different color scheme and a slightly different enclosure.

[14]Even an option to add a removable patch panel was offered due to customer requests.

[15]To protect these rather expensive potentiometers against damage caused by patching errors, a small incandescent bulb has been placed in series with the wiper. Under normal circumstances only a tiny current flows through the wiper and the small resistance of the bulb in its cold state does not introduce any noticeable error into a computation. In case of a erroneous connection when excessive current flows through the potentiometer's wiper, the bulb lights up and limits the current to a value that will not damage the potentiometer.

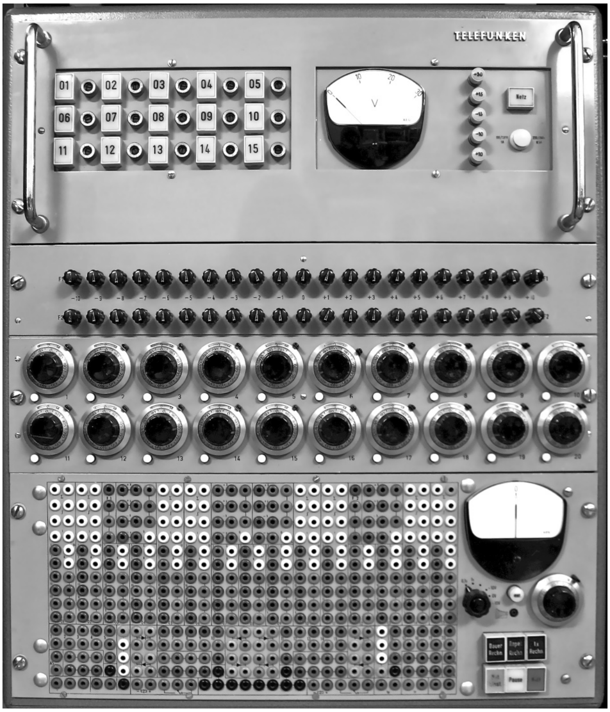


Figure 6.8: The RAT 700 tabletop analog computer

A fully expanded RAT 700 contains 19 operational amplifiers,[16] four quarter square multipliers, two diode function generators, 20 coefficient potentiometers, eight free diodes and two relays that can be used to setup comparators.

The RAT 700, which sold remarkably well, was in fact just a by-product of the development of a large precision analog computer that was completed in 1960 and introduced to the public at the Hanover trade show the same year. This machine, the *RA 800*, which is shown in its basic configuration in figure 6.9, was living proof that a precision analog computer with a machine unit of ± 10 V could be built with transistors instead of the ubiquitous vacuum tubes. Around this time, other companies started their own developments of transistorized analog computers that could compete with the RA 800.

The left rack contains, from top to bottom, a power supply, the central removable patch panel, 50 precision coefficient potentiometers, two servo resolvers and two drawers with four time division multipliers each. The right rack contains, top to bottom, a power supply, two drawers with four diode function generators each, ten manual switches, a digital voltmeter, the central control panel with various timers, compensation voltmeter and computing element selection logic, 50 additional potentiometers, two drawers each holding 30 operational amplifiers and ten electronic comparators.

[16]Eight of these operational amplifiers are committed to switchable integrators/summers, seven are part of summers while the remaining four are needed for the multipliers and function generators.

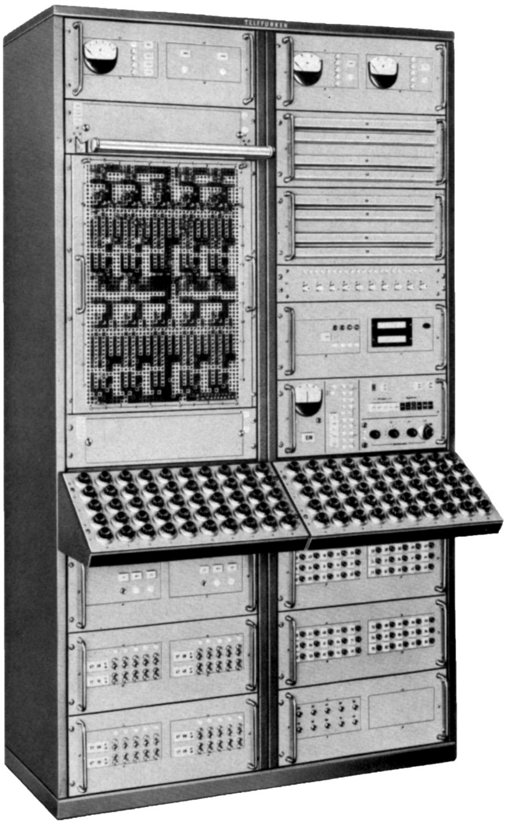


Figure 6.9: A basic RA 800 precision analog computer (see [Meyer-Brötz 1960][p. 176])

EAI was also developing a transistorized operational amplifier that would form the heart of its first tabletop analog computer, the *TR-10*, which was introduced to the market in 1960. Figure 6.10 shows the front view of a TR-10. Instead of a central patch panel, it exposes its various components to the programmer.[17] Interestingly, this machine – like several other later systems made by EAI – does not offer integrators and summers as pre-wired computing components but requires to build these elements by interconnecting operational amplifiers with additional elements during programming.[18]

From top to bottom, this particular TR-10 contains 20 coefficient potentiometers, twelve computing networks to build summers, integrators etc., ten dual operational amplifiers that have to be connected to the computing networks above, and the control panel at the bottom.

[17] A removable patch panel was later offered as an option.

[18] This is in sharp contrast to other systems like all Telefunken machines that hide the operational amplifiers from the outside and offer mathematical operations as the basic elements for a program.

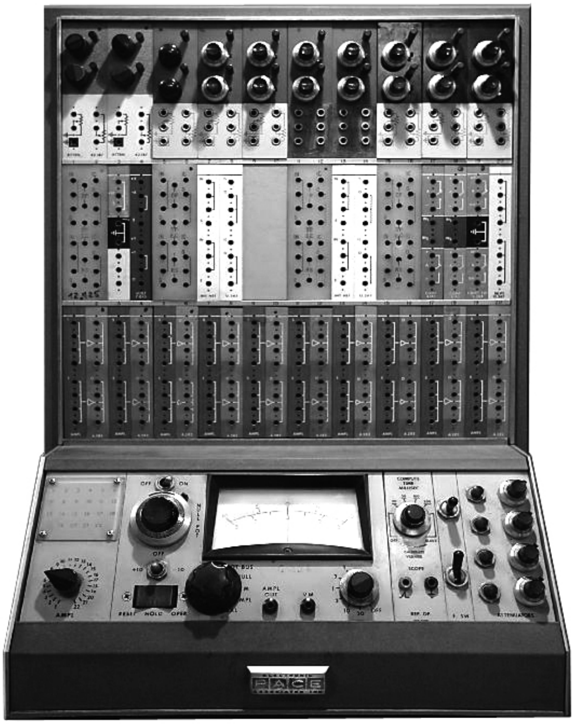


Figure 6.10: The EAI TR-10 tabletop analog computer

6.4 Late analog computers

In the mid 1960s it became clear that analog computers had to be connected to stored-program digital computers to form so-called *hybrid computers*[19] and most manufacturers started development of such systems by extending existing models with the necessary hardware like analog-digital- and digital-analog-converters etc. A typical example for this strategy is the EAI HYDAC-2000 shown in figure 6.6. Telefunken used the same approach by extending the RA 800 into the *RA 800H*, the *H* denoting its hybrid capability. In fact this machine was basically still a RA 800 system but had a digital control system quite like the DOS 350 of the HYDAC-2000.

Figure 6.11 shows a RA 800H system. The rack on the left holds the digital extension, the so-called *DEX 802*[20] which is just a collection of basic digital elements like logic gates, counters and timers. In addition to this, there are additional drawers containing function generators, an electronic resolver etc. Nevertheless, this rather simple extension made the RA 800H a much more versatile system than the RA 800. Using the DEX 802 complex computations like parameter optimizations became possible.

[19]See section 9.

[20]Short for "Digital Experimentierzusatz" which translates roughly to "digital extension for experiments".

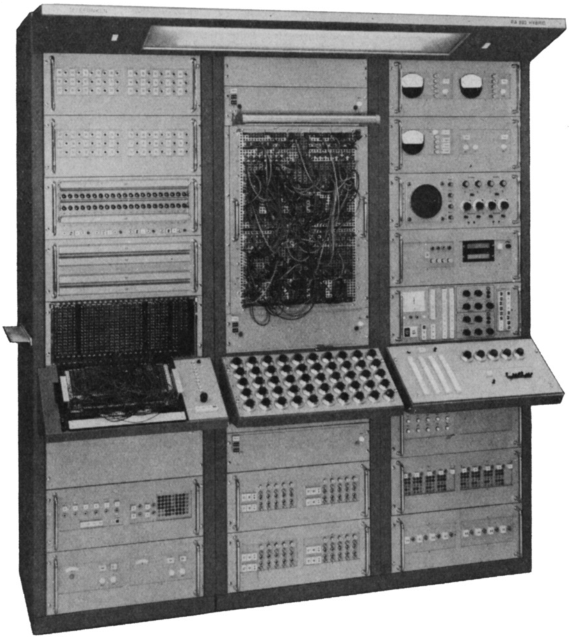


Figure 6.11: A RA 800H system (see [Adler 1968][p. 273])

The last Telefunken analog computer is the RA 770 which is a significant improvement in comparison to the RA 800H. Shown in figure 6.12 this system achieves an overall precision of 10^{-4} and uses improved operational amplifiers employing silicon transistors and fully electronic choppers.[21] Using integrated circuits in the control system and other improvements over the RA 800H the RA 770 is physically smaller and yet offers a wider complement of computing elements.

Both, the RA 770 and the RA 800H, can be controlled from an external stored-program digital computer by means of a so-called *hybrides Koppelwerk*[22] *HKW 860* forming a real hybrid computer. Unfortunately, Telefunken stopped the development of analog computers in the early 1970s with no successor to the RA 770 which was announced in 1966. Its analog computer branch went out of business in the 1970s while other manufacturers, like EAI, were still bringing new and greatly enhanced systems to the market.

As an example of such a late EAI system, a *PACER 500* hybrid computer is shown in figure 6.13. From left to right, the console typewriter for the stored-program digital processor, the digital processor itself, the dominating analog patch field, the operator control panel, including a small digital patch field, and a small cathode ray tube display can be seen. Systems like these were in widespread use in nearly all scientific and engineering branches well into the 1980s.

---

[21] These amplifiers, described in [Meyer-Brötz et al. 1966], and shown in figure 4.14, chapter 4, have an open-loop gain of $3 \cdot 10^8$.

[22] German for *hybrid coupler*.

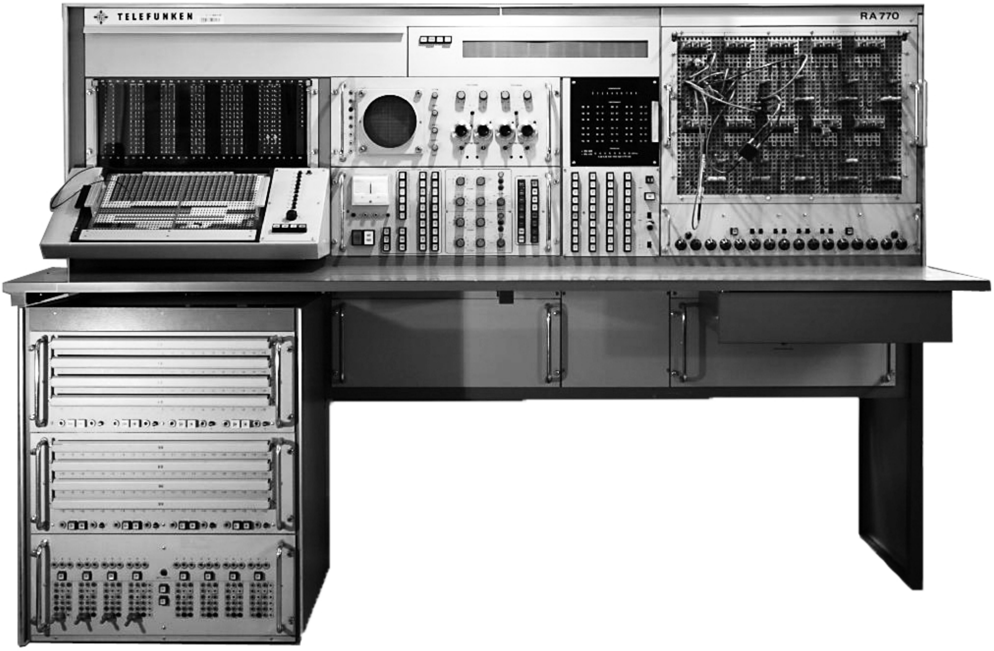


Figure 6.12: Telefunken RA 770 precision analog computer

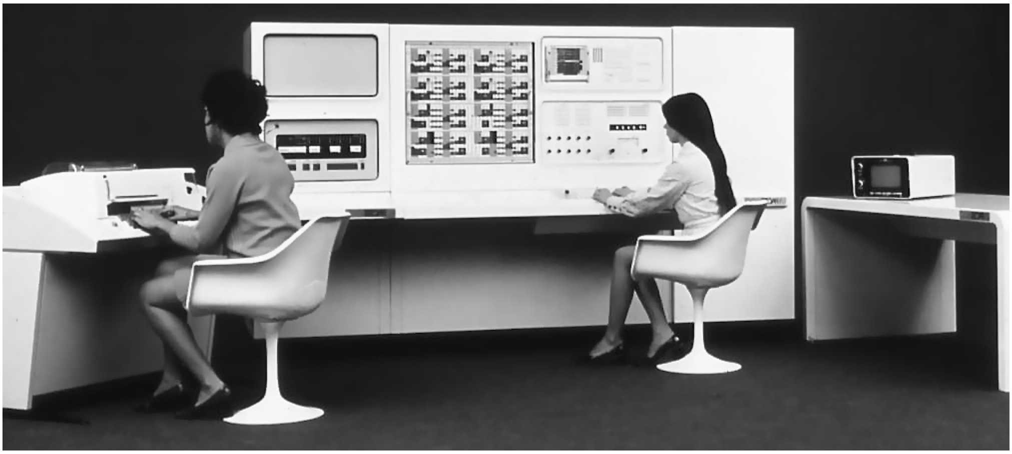


Figure 6.13: An EAI PACER 500 hybrid computer system

Since the purpose of this chapter is to give only an impression of various analog computers, many influential systems, like the EAI-580, the EAI-680, the large systems EAI-7800 and EAI-8800, and many other vendors like Beckman, COMCOR, Dornier, Simulators Inc., Goodyear, GAP/R, Solartron etc. are missing. This should not imply that their machines are inferior to those described here.

7 Programming

Programming an analog computer is completely different from programming a stored-program digital computer. In the early days of computing as such, programming analog computers was considered being easier and even more "natural" than developing and implementing an algorithm for a digital computer. Especially users from the engineering faculties took the analog computer as a natural extension of their way of thinking about systems and their simulation. This was due to the fact that an analog computer readily implements a working model of a system to be analyzed.

The situation has changed radically since the 1970s. The abstract algorithmic approach which is the basis for programming nearly all of our current digital computers has completely eliminated the analog approach from the typical mental programming toolkit. Thus, from today's perspective, programming an analog computer is a curiosity while programming a stored-program computer is considered being something natural.

The following sections show the basics of analog computer programming with some practical examples of increasing complexity. Since programming, be it analog or algorithmic, is an art to a certain degree, a lot of hands-on experience is required to become a good analog computer programmer. First of all it is necessary to abandon the algorithmic approach, as natural as it may seem, while reading the following sections.

7.1 Basic approach

The basic approach of programming an analog computer is depicted in figure 7.1: First of all a thorough analysis of the system to be simulated is made which normally yields systems of coupled differential equations.[1] Based on these differential equations a *program* for the analog computer is devised. This program is a computer setup that describes the necessary connections between computing elements to implement those equations. After defining this setup that will be implemented on the patch panel of the analog computer, a *normalization step* is necessary to ensure that no variable of a setup will exceed the range of ± 1 machine unit.[2] This step yields the required coefficient potentiometer settings for the problem.

Given sufficient initial conditions for the integrators of such a computer setup, the system of which an *analog* has been implemented can now be simulated.

[1] This first step is the same for developing an algorithmic solution for a given problem.
[2] This is often the hardest part in programming an analog computer.

Figure 7.1: Typical workflow in analog simulation (cf. [Truitt et al. 1964][p. 1-108])

To derive an analog computer program from a set of equations, normally either the so-called *feedback technique*[3] developed by Kelvin in 1875/1876 or the *substitution method*[4] is employed. Both methods start with a set of differential equations of the general form

$$y^{(n)}(x) + a_{n-1}y^{(n-1)}(x) + \cdots + a_1 y'(x) + a_0 y(x) = f(x) \tag{7.1}$$

with initial conditions

$$y\Big|_{x=0} = y_0, \ldots, y^{(n-1)}\Big|_{x=0} = y_0^{(n-1)}$$

[3]Cf. [Schwarz 1971][pp. 25/165 ff.].
[4]See [Schwarz 1971][pp. 168 ff.].

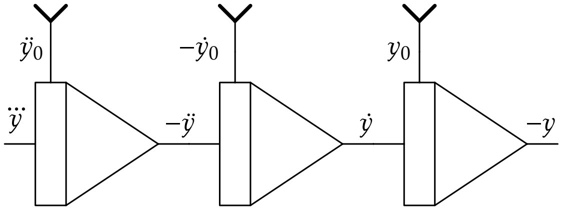


Figure 7.2: Kelvin's feedback technique, 1st step

based on the system to be simulated. Derivatives with respect to time t or the scaled *machine time* τ, like $\frac{\mathrm{d}y}{\mathrm{d}t}$, which are directly accessible for an analog electronic analog computer that only has time as the free variable of integration,[5] will be written as $\dot{y}$ in the following.

Normally no differentiations are used in an analog computer setup due to the inevitable increase in noise that is caused by this operation.[6] Instead integration is used wherever possible to reduce the order of the derivatives involved.

7.2 Kelvin's feedback technique

Kelvin's feedback technique, which also has been called the *classical differential analyzer technique*,[7] is based on the idea of a series of integrations starting with the highest derivative of a differential equation and successively yielding all lower derivatives that are necessary to represent the equation in question. As an example the following equation with initial conditions $\ddot{y}_0$, $\dot{y}_0$ and y_0 is to be solved with an analog computer:

$$\dddot{y} + a_2\ddot{y} + a_1\dot{y} + a_0 y = 0 \tag{7.2}$$

First, the equation is solved for its highest derivative, $\dddot{y}$:

$$\dddot{y} = -a_2\ddot{y} - a_1\dot{y} - a_0 y. \tag{7.3}$$

Using three integrators connected in series as shown in figure 7.2 the lower derivatives $-\ddot{y}$, $\dot{y}$ and $-y$ can be computed.[8]

Combining these derivatives, the right hand side of equation (7.3) can be obtained as shown in figure 7.3. The coefficients a_0, a_1 and a_2 are implemented using three coefficient potentiometers while an additional summer is necessary to yield $-\dot{y}$. Since every integrator (and summer) performs a sign reversal, the three initial conditions have to be supplied with alternating signs in this setup, too.

[5]This is a noteworthy restriction that only applies to analog electronic computers. Classic mechanical or electromechanical differential analyzers as well as their digital counterparts (see chapter 10) do not have this restriction.

[6]See section 3.1.2.

[7]See [Korn & Korn 1964][p. 1-5].

[8]The alternating signs are caused by the sign changing nature of typical integrators.

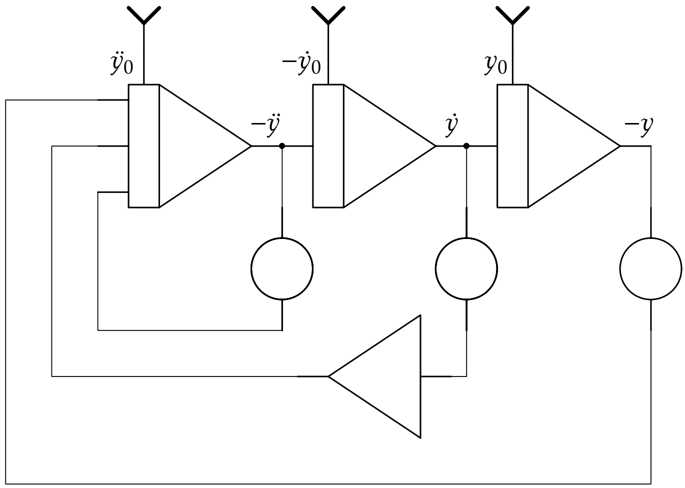


Figure 7.3: Kelvin's feedback technique, 2nd step

The computer setup derived by this simple method is now an analog of the system described by the differential equation (7.2). Depending on its scaling, controlled by the coefficients a_0, a_1 and a_2 and the time constants of the integrators, it can react in the same span of time as the original system or faster or slower. While many problems, especially those with hardware- or humans-in-the-loop, require no time scaling, other problems benefit considerably from stretching or compressing time.[9]

Using these steps of the Kelvin method, an analog computer setup can be derived quite automatically given differential equations to start with:

1. Rearrange the equations so that the highest derivative is separated.

2. Generate all lower derivatives that are part of the equations by successive integration steps.

3. Using these lower derivatives the right hand sides of the equations can be generated.

4. Since these right hand sides are equal to the highest derivatives used in the first step, the feedback loop can be closed by feeding these values into the inputs of the first integrator stage.

[9]Typical examples are the simulation of processes in a nuclear reactor or problems in population dynamics.

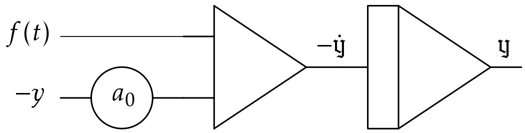


Figure 7.4: First sub-circuit derived by the method of substitution

7.3 Substitution method

The approach of the *substitution method*[10] is best shown by deriving a computer setup for solving the differential equation[11]

$$\ddot{y} + a_1\dot{y} + a_0 y = f(t). \tag{7.4}$$

To simplify matters, the initial conditions $\dot{y}_0$ and y_0 are assumed to be zero.

The basic idea is now to separate differential equations of first degree from equation (7.4) by repeated substitution. Each substitution lowers the degree of the remaining differential equation by one. The substitution

$$\mathfrak{y} = \dot{y} + a_1 y \tag{7.5}$$

yields

$$\dot{\mathfrak{y}} = \ddot{y} + a_1\dot{y} \tag{7.6}$$

as its first derivative with respect to time. Substituting (7.6) into (7.4) yields the non-homogenous differential equation

$$\dot{\mathfrak{y}} + a_0 y = f(t). \tag{7.7}$$

Solving (7.7) for its highest derivative yields

$$\dot{\mathfrak{y}} = f(t) - a_0 y$$

which corresponds to the circuit shown in figure 7.4.

Now a second circuit representing equation (7.5) has to be set up. Solving this equation for its highest derivative yields

$$\dot{y} = \mathfrak{y} - a_1 y$$

which corresponds to the circuit shown in figure 7.5. $\mathfrak{y}$ is readily available from the circuit shown in figure 7.4.

Combining these two sub-circuits in a straightforward fashion yields the setup shown in figure 7.6. Realizing that every summer and integrator performs a sign inversion, this circuit can be simplified considerably.[12] The simplified setup is shown in figure 7.7. Compared with 7.6 it saves two summers.

[10] Also known as *partial feedback method*.

[11] Cf. [Schwarz 1971][pp. 168 ff.].

[12] Since every operation causes some error, setups for analog computers should always be minimized with respect to the number of computer elements involved.

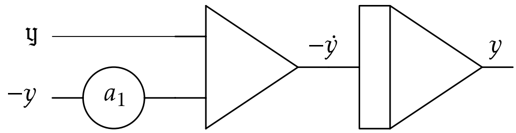


Figure 7.5: Second sub-circuit derived by the method of substitution

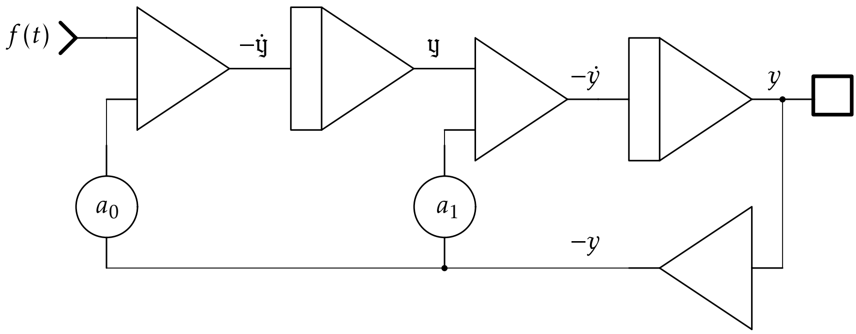


Figure 7.6: Non-optimized circuit derived by the method of substitution

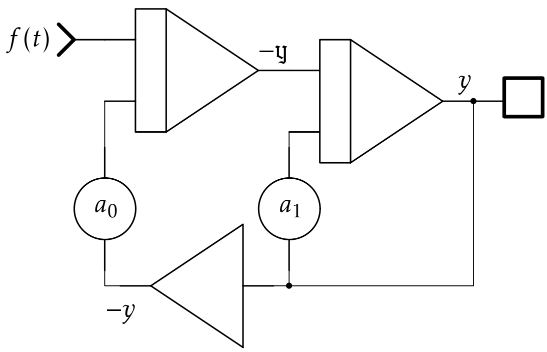


Figure 7.7: Optimized circuit derived by the method of substitution

A major drawback of the substitution method is the complexity involved with non-zero initial conditions. While the initial condition y_0 can be used directly to setup the last integrator in a program derived by this method, all other initial values have to be transformed according to the substitutions performed before. So the initial value $\dot{y}_0$ has to be transformed into an equivalent initial value

$$\dot{\mathfrak{y}}_0 = \dot{y}_0 + a_1 y_0.$$

Only $\mathfrak{y}$ can be used to initialize the corresponding integrator. This initial value transformation makes the application of the substitution method rather cumbersome in real applications. Thus the following sections will rely on the simpler Kelvin method.

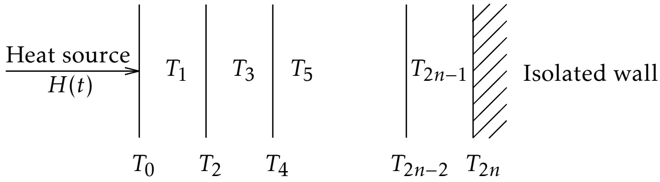


Figure 7.8: One-dimensional heat-transfer

In general, both techniques, Kelvin's feedback method as well as the substitution method can be applied to linear and non-linear and linear differential equations.[13]

7.4 Partial differential equations

Many problems require the solution of partial differential equations which contain derivatives with respect to more than variable.[14] Since an analog electronic analog computer can only use time as the free variable of integration, such partial differential equations, *PDE* for short, are normally solved by either employing a quotient of differences or by separation of variables.[15]

7.4.1 Quotient of differences

The *quotient of differences*-method is based on the idea to handle all derivatives not depending on time t as quotients thus discretizing the underlying problem suitably. Thus the idea is to approximate a derivative of the form

$$\frac{\mathrm{d}f(x)}{\mathrm{d}x} = \lim_{\Delta x \to 0} \frac{f(x+\Delta x) - f(x)}{\Delta x}.$$

by a quotient of differences like this:

$$\frac{\Delta f(x)}{\Delta x} \approx \frac{f(x+\Delta x) - f(x)}{\Delta x}$$

As simple as this method is, it usually requires an enormous amount of computing elements, which sometimes outweighs the advantage of simplicity. Figure 7.8 shows a simple one-dimensional heat-transfer problem that is to be solved with an analog computer and the quotient of differences method.

[13]In the case of non-linear differential equations the handling of initial conditions can be quite tricky. Sometimes special computer setups are used. Cf. [Soudack 1968] and [Brown 1969] for more information. Some special classes of non-linear differential equations can be tackled with a method different from those described here, see [White 1966].

[14]Most often derivatives with respect to time t and Cartesian coordinates (x, y, z) or spherical coordinates (t, φ, θ) are encountered.

[15]See [Howe et al. 1953], [Howe 1962], or [Forbes 1972].

A heat conducting rod is connected to a time dependent heat source $H(t)$ on its left side while the right side is placed against an insulating wall.[16] In general such a problem is described by the PDE

$$\frac{\partial^2 T}{\partial x^2} = k\dot{T}.$$

The rod is divided into slices of equal width as shown in figure 7.8. Each of these slices corresponds to one difference quotient. Given the heat source on the left and the insulator on the right side yields the following boundary conditions:

$$\dot{T}_0 = \frac{1}{k\Delta x} H(t) \tag{7.8}$$

$$\dot{T}_{2n} = \frac{1}{k\Delta x}\left(T_{2n-1} - T_{2n-2}\right). \tag{7.9}$$

The resulting quotient terms with indexes $2m$ with $1 \leq m < n$ are

$$\frac{\partial^2 T_{2m}}{\partial x^2} \approx \frac{T_{2m-1} - 2T_{2m} + T_{2m+1}}{\frac{1}{4}(\Delta x)^2}. \tag{7.10}$$

The interjacent elements are

$$T_{2m+1} = \frac{T_{2m} + T_{2m+2}}{2},$$

where T_{2m+1} represents the mean temperature on the edges of the corresponding slice in figure 7.8. Starting with equation (7.10), a system of differential equations involving only time t as free variable can be derived. Assuming $n = 3$ and taking the boundary conditions (7.8) and (7.9) into account yields the following set of differential equations:

$$\begin{aligned}
\dot{T}_0 &= \frac{1}{k\Delta x} H(t) \\
T_1 &= \frac{T_0 + T_2}{2} \\
\dot{T}_2 &= \frac{4}{k(\Delta x)^2}\left(T_1 - 2T_2 + T_3\right) \\
T_3 &= \frac{T_2 + T_4}{2} \\
\dot{T}_4 &= \frac{4}{k(\Delta x)^2}\left(T_3 - 2T_4 + T_5\right) \\
T_5 &= \frac{T_4 + T_6}{2} \\
\dot{T}_6 &= \frac{1}{k\Delta x}\left(T_5 - T_4\right)
\end{aligned}$$

[16] More examples can be found in [Seyferth 1960], [Truitt et al. 1964][pp. 3-66 ff.], [Ameling 1963][pp. 284 ff.], [Sydow 1964][pp. 103 ff.], [Mackay 1962][pp. 243 ff. and pp. 293 ff.], [Telefunken 1963/1], and [Giloi et al. 1963][pp. 255 ff.]. A more complex gas flow example is described in [Mahrenholtz 1968][pp. 143 ff.].

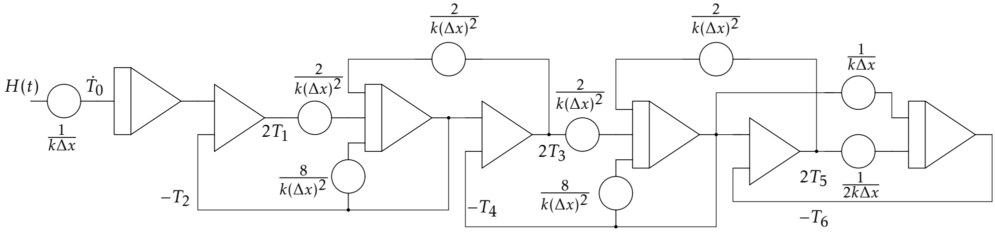


Figure 7.9: Circuit for one-dimensional heat-transfer

These equations can be readily transformed into the computer setup shown in figure 7.9. The high amount of computing elements is obvious.[17] An advantage of this method for solving problems involving PDEs is that the coefficient potentiometers can be set up in groups simplifying the programming task significantly.

When the derivatives not depending on t are of higher degree, the method of difference quotients becomes rather cumbersome. In these cases, a different approach is often used which involves the development of Taylor series approximations for the higher derivatives. Cf. [Ameling 1964] and [Giloi et al. 1963][pp. 270 ff.] for detailed information.

7.4.2 Separation of variables

Another method for the solution of partial differential equations with an analog computer is based on the separation of variables.[18] While the method described above discretizes all variables beside t, the separation of variables allows all variables of a PDE to take on continuous values. As a result, computer setups based on the separation of variables often require substantial fewer computing elements. Nevertheless, this technique has a major drawback: It is much more complicated to derive a computer setup by using the separation approach than by the method of difference quotients.

As an example the one-dimensional heat-transfer based on

$$\frac{\partial^2 T}{\partial x^2} = k\dot{T} \tag{7.11}$$

will be treated again.[19] The task is to determine the temperature $T(x,t)$ at a certain position and a certain time, so the separation of the variables t and x yields

$$T(x,t) = f_1(x)f_2(t). \tag{7.12}$$

From equations (7.11) and (7.12)

$$\frac{\mathrm{d}^2 f_1(x)}{\mathrm{d}x^2} f_2(t) = k\dot{f}_2(t)f_1(x)$$

[17]It was not uncommon for some computer setups, especially in the area of chemical technology, to require hundreds of integrators and summers to model heat transfer problems with a reasonable fine discretization width.

[18]See [Stepanow 1956][pp. 12 ff.].

[19]Similar examples can be found in [Sydow 1964][pp. 107 f.] and [Ameling 1963][pp. 280 ff.].

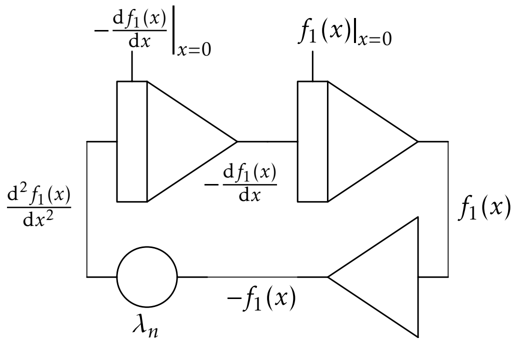


Figure 7.10: Circuit corresponding to (7.16)

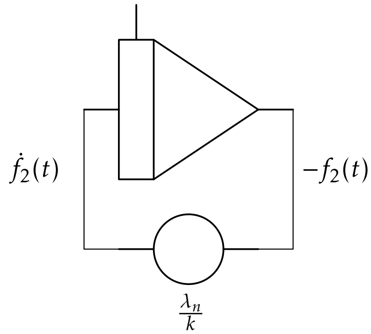


Figure 7.11: Circuit corresponding to (7.17)

follows. Separating x and t yields

$$\frac{\frac{\mathrm{d}^2 f_1(x)}{\mathrm{d}x^2}}{f_1(x)} = k\frac{\dot{f}_2(t)}{f_2(t)}. \tag{7.13}$$

Splitting equation (7.13) and setting both sides equal to $-\lambda_n$ yields the eigenvalue problem

$$\frac{\frac{\mathrm{d}^2 f_1(x)}{\mathrm{d}x^2}}{f_1(x)} = -\lambda_n \quad \text{and} \tag{7.14}$$

$$k\frac{\dot{f}_2(t)}{f_2(t)} = -\lambda_n. \tag{7.15}$$

Solving (7.14) and (7.15) for their highest derivative results in

$$\frac{\mathrm{d}^2 f_1(x)}{\mathrm{d}x^2} = -\lambda_n f_1(x) \quad \text{and} \tag{7.16}$$

$$\dot{f}_2(t) = -\frac{\lambda_n}{k} f_2(t) \tag{7.17}$$

which are readily transformed into the circuits shown in figure 7.10 and 7.11 where the variable x is identified with t.

Initially, for time $t = 0$, the rod is at temperature $T_0 \neq 0$. Then its ends located at $x = 0$ and $x = 1$ are forced to temperature 0 so that $T(0,t) = T(1,t) = 0$. Using the circuit shown in figure 7.11 suitable values λ_n are determined. This is normally performed manually by varying the coefficient potentiometer between successive runs of the circuit with $0 < t \leq 1$ which can be a quite time-consuming process.[20] When all λ_n are

[20]The determination of λ_n is done by trial and error as [Ameling 1963][p. 281] notes.

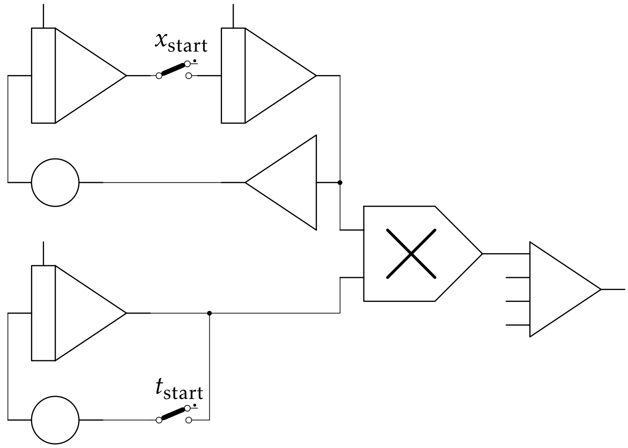


Figure 7.12: One-dimensional heat-transfer with separation of variables

determined, a linear combination of the eigenfunctions is sought that approximates $T(x,t)$ as good as possible or necessary:[21]

$$T(x,t) = \sum_{i \in \mathcal{I}} \lambda_i T_i \tag{7.18}$$

Figure 7.12 shows the mechanization of equation (7.18). Using this setup it is either possible to hold t at a fixed value while x is running from 0 to 1 or vice versa. The two variables are controlled with the two switches labeled t_{start} and x_{start}.

7.5 Scaling

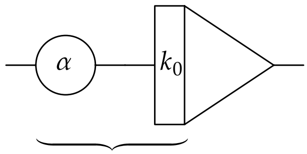
Deriving a circuit corresponding to a given set of differential equations is only one step toward the solution of a given problem by means of an analog electronic analog computer. Since all variables are bounded by ±1 machine unit, it is in most cases necessary to *scale* the problem. An unscaled variable v representing some property of the underlying problem is associated with a *machine variable* $\mathfrak{v}$ that always satisfies

$$-1 \leq \mathfrak{v} \leq 1. \tag{7.19}$$

To achieve this, a set of scaling factors α_v have to be determined that all $\mathfrak{v} = \alpha_v v$ satisfy (7.19). Determining these α_v can be a time-consuming and difficult process since the overall behavior of the underlying differential equations has to be considered.[22] When $\max(|v|)$ is known with regard to a unit U then α_v is determined by[23]

$$\alpha_v = \frac{1 \text{ machine unit}}{\max(|v|)} \qquad \left[\frac{\mathrm{V}}{\mathrm{U}}\right].$$

[21] The T_i denotes the i-th eigenfunction, while $\mathcal{I}$ denotes a suitable index set.
[22] This is even more complex a problem in the case of non-linear differential equations.
[23] Units are noted in square brackets.



$\lambda = \alpha k_0$

Figure 7.13: Time scaling

Determining these α_v is further complicated by the technical requirement of using as much of the range ±1 machine unit of the computing elements to minimize errors.[24] Scaling a problem variable v with a scaling factor α_v always yields a machine variable $\mathfrak{v}$ which is measured in Volts. Such machine variables are normally denoted by square brackets in the literature. Considering a problem variable $\ddot{x}$ representing an acceleration which can assume a maximum value $\max(|\ddot{x}|) = 2 \cdot 9.81\text{m/s}^2$ and a machine unit of 10 V yields a practical scaling factor $\alpha_{\ddot{x}} = 1/2$. The resulting machine variable $\alpha_{\ddot{x}}\ddot{x}$ is denoted by $\left[\frac{\ddot{x}}{2}\right]$.

Time can be scaled, too. Problem time, which is often called *real-time*, is normally denoted by t while machine time is represented by τ. The time-scaling factor is denoted by λ, yielding $\tau = \lambda t$. Since the integrators are the only computing elements depending on time, time-scaling is either done by changing the time constant of the integrators[25] or by changing the input factors of the integrators. Often a combination of both methods is used. Figure 7.13 shows the effect that is caused by selecting a time constant k_0 in conjunction with an coefficient potentiometer placed at the input and set to a value α.

By changing λ it is possible to have simulations running slower or faster than real-time. Large analog computers often feature a switch labeled "10×" that exchanges the feedback capacitors of all integrators by devices with one-tenth of the normal capacity thus speeding up a computer run by a factor of 10 at the press of single button. Such large computers often feature up to five different feedback capacitors per integrator, thus they allow the values k_0 to span five decades easily.

Many methods were devised to facilitate the scaling process, including exemplarily solutions of a problem with a stored-program digital computer to determine the maximum values of the problem variables involved.[26] Other approaches used hybrid computers[27] where the digital part generates scaling values for the analog part.[28]

[24] Especially analog electronic multipliers tend to have rather large errors for small multiplicators and multiplicands.

[25] This is done by selecting appropriate capacitors in the feedback path of the integrators.

[26] A system using digital optimization techniques to generate scaling factors for analog computers is described in [Celmer et al. 1970].

[27] Cf. chapter 9.

[28] See [Hall et al. 1969] and [Giloi 1975][pp. 129 ff.].

8 Programming examples

Learning how to program an analog computer requires quite some practice. Thus, the following sections describe some typical problems and their treatment by means of an analog computer. The complexity of these examples ranges from low to fairly high.[1]

8.1 $y = a\sin(\omega t + \varphi)$

Generating a sine-/cosine-signal pair on an analog computer is a common and useful task. Signals like these are useful to study filter design, the behavior of vibrating mechanical systems, they allow the generation of closed figures for display on oscilloscopes or plotters, etc. Since an analog computer is the ideal tool to solve differential equations, it is natural to start with a differential equation of the form

$$\ddot{y} + \omega^2 y = 0 \tag{8.1}$$

with initial conditions

$$y_0 = a\sin(\varphi) \quad \text{and}$$
$$\dot{y}_0 = a\omega\cos(\varphi).$$

To employ Kelvin's feedback method, equation (8.1) is solved for $\ddot{y}$ yielding

$$\ddot{y} = -\omega^2 y. \tag{8.2}$$

Since the sine-function is a particular solution of this differential equation it can be readily used to generate a pure sine-signal given suitable initial values. In this case, additional scaling of the problem variables is not necessary since $\max(|\sin(\omega t)|) = 1$. Figure 8.1 shows the basic circuit based on equation (8.2). This setup is, of course, missing any initial values and would yield the null-function.

Since ω determines the frequency of the resulting sine-signal, it influences the time-scaling process. Figure 8.2 shows the time scaled circuit. Using two coefficient potentiometers in conjunction with the time-constant k_0 of the integrators, $\omega = \alpha k_0$ can be set.[2] Thus time is identified with τ in this setup. To generate a pure sine-signal the second integrator is fed with an initial value of 0.

Figure 8.3 shows the output signal $[y]$ generated by this circuit. Using the time control system of the analog computer used,[3] the integrators were placed into *operate* mode for

[1] All of the programs described were implemented and run on real analog computers.

[2] Some machines, like the Telefunken RA 770, even have ganged coefficient potentiometers to facilitate setups like this where two potentiometers must be set in a combined fashion.

[3] A Telefunken tabletop analog computer RA 741 was used in this example.

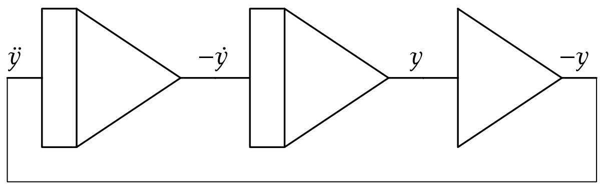


Figure 8.1: Basic circuit for $y = \sin(\omega t + \varphi)$

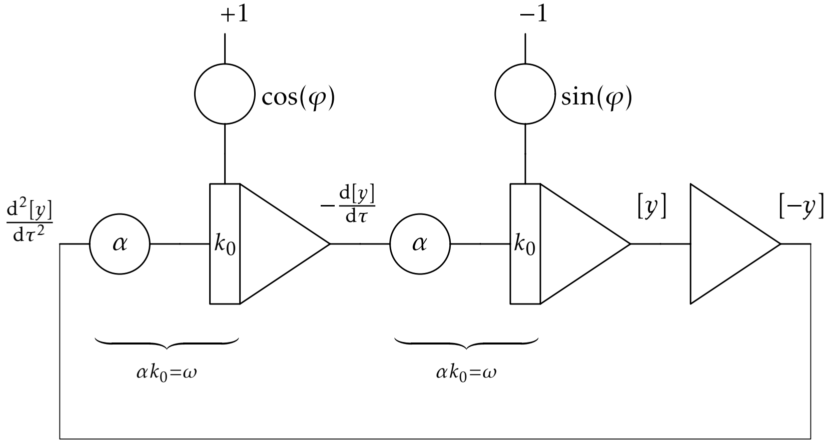


Figure 8.2: Basic circuit for $y = \sin(\omega t + \varphi)$

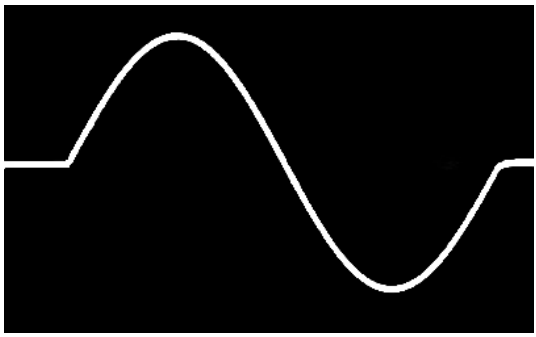


Figure 8.3: A single sine period generated by the circuit shown in figure 8.2

the time it takes to generate one full wave. It should be noted that the output signal of a setup like this will decay or build-up when the program is run for a longer period of time. This is due to the fact that the integrators are never error-free. If a sine-/cosine-signal with a stable amplitude is required over an extended period of time, additional measures have to be taken.[4]

[4]See section 8.5.

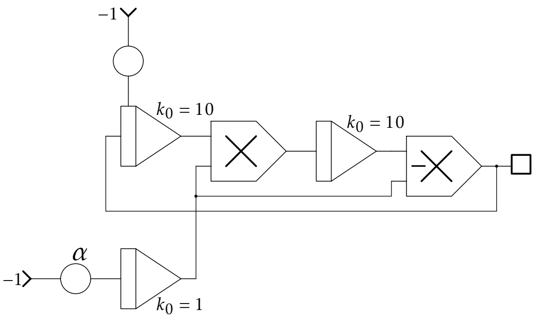


Figure 8.4: Basic circuit for a sweep generator

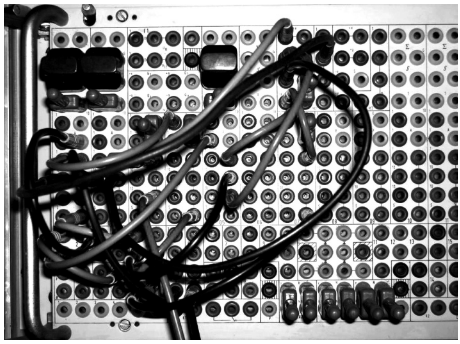


Figure 8.5: Patch panel setup for the sweep generator

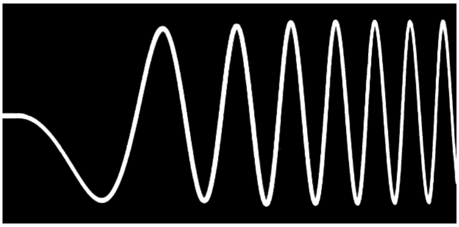


Figure 8.6: Sine oscillation with variable frequency

8.2 Sweep generator

Since $\omega = \alpha k_0$ controls the output-signal's frequency, the circuit shown in figure 8.2 can be easily extended to yield a sweep generator. Sweeped signals are frequently employed to analyze transmission systems or models thereof, the response of vibrating systems, etc. It is just necessary to replace the two coefficient potentiometers feeding the integrators by two multipliers which share a common multiplicator.[5] If only one of these multipliers is sign-changing the inverting summer can be saved.[6]

Figure 8.4 shows the basic setup of such a sweep generator.[7] The frequency controlling signal is derived from integrating over a constant value α. To achieve a sufficient slow frequency change two time constants for the integrators involved are used. The integrator with α as its input has a time scale $k_0 = 1$ while the two integrators which form the basic oscillator loop run with a time scale ten times fast ($k_0 = 10$). The patch panel setup for this sweep generator is shown in figure 8.5. Figure 8.6 shows the output of a typical sweep run with linearly increasing frequency.

[5] In applications like this where two or more multiplicands are to be multiplied with a (slowly changing) multiplier, a servo multiplier or a time division multiplier could be used with advantage.

[6] The sign-inversion can be easily achieved with a quarter square multiplier: Since these multipliers require both input signals with both polarities, $\pm x$ and $\pm y$, the sign of the result can be inverted by interchanging the connections of one of these signal pairs.

[7] It should be noted that this circuit can be used to generate the functions $-\sin(\varphi)$ and $\cos(\varphi)$ if the first derivative $\dot{\varphi}$ is available as input. In this case, $\dot{\varphi}$ is used to feed the two multipliers thus replacing the integrator shown on the lower left.

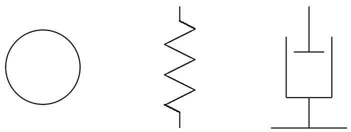


Figure 8.7: Parts of a mass-spring-damper system

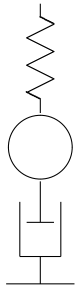


Figure 8.8: Mass-spring-damper system

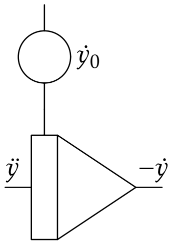


Figure 8.9: First integration step yielding $-\dot{y}$

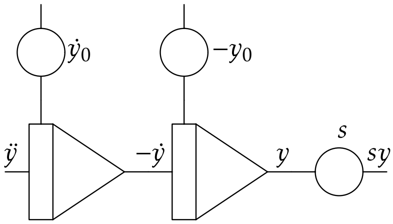


Figure 8.10: Second integration step yielding y

8.3 Mass-spring-damper system

The next problem is a bit more complex. A mass-spring-damper system is to be modeled and simulated with an analog computer. Figure 8.7 shows the basic parts of the underlying physical system: A mass m exerting a force $m\ddot{y}$, a spring with stiffness s resulting in a force sy and a damper with damping coefficient d that exerts a force $d\dot{y}$. Since the forces in a closed physical system must add to 0 the mass-spring-damper system shown in figure 8.8 can be described by

$$m\ddot{y} + d\dot{y} + sy = 0. \tag{8.3}$$

which yields

$$\ddot{y} = \frac{-(d\dot{y} + sy)}{m} \tag{8.4}$$

as a starting point for applying the Kelvin method. Figure 8.9 shows the first step of the resulting program. Using an integrator, $\ddot{y}$ is integrated, yielding $-\dot{y}$ with an initial value of $\dot{y}_0$.

Using a second integrator with initial value $-y_0$ yields y which can be multiplied by s using a coefficient potentiometer as shown in figure 8.10. Adding an inverter and a second coefficient potentiometer to derive $d\dot{y}$ and summing this term an sy results in the circuit shown in figure 8.11. Now, since sy and $d\dot{y}$ have been derived from $\ddot{y}$ their sum $-(d\dot{y} + sy)$ can be multiplied by $1/m$ with a coefficient potentiometer yielding the

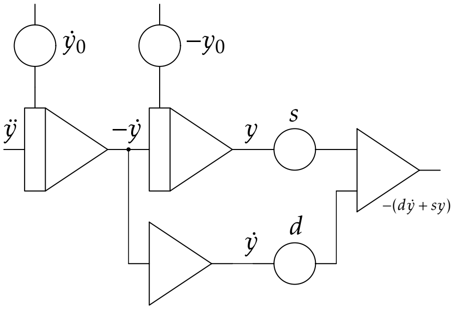


Figure 8.11: Computing $-(d\dot{y}+sy)$

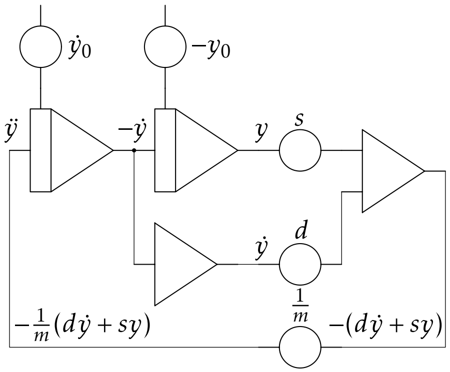


Figure 8.12: Computer setup for the mass-spring-damper system

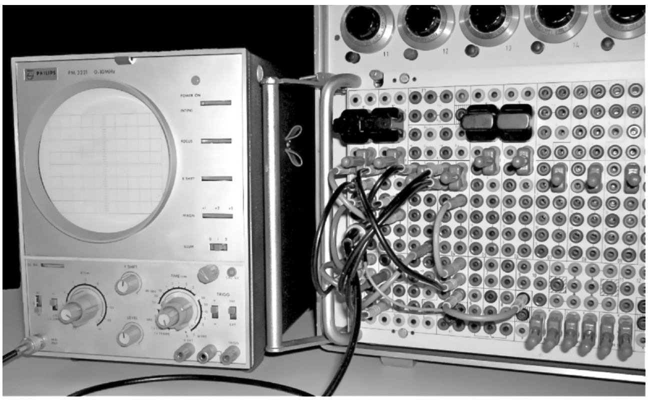


Figure 8.13: Setup of the mass-spring-damper simulation

right side of equation (8.4) thus closing the feedback loop in the final circuit shown in figure 8.12.

The patch panel setup for this program is shown in figure 8.13. The figures 8.14, 8.15, 8.16, and 8.17 show simulation results for various settings of the spring's stiffness and the damping coefficient with constant mass $m = 1$.[8] If the computer is running in repetitive mode and if the time constants of the integrators are selected accordingly, a standing image can be displayed on an oscilloscope screen. The effect of changes to the coefficients s, d, and m can then be seen directly while turning a potentiometer's dial.

These runs were performed on a non-scaled model and thus yielded only qualitative results. To derive quantitative results the following values are assumed: $m = 1.5$ kg, $s = 60\ \frac{\text{kg}}{\text{s}^2}$, $d = 3\ \frac{\text{kg}}{\text{s}}$ and $y_0 = 0.1$ m (initial displacement of the mass).

[8] These still frames were taken from the screen of a digital storage oscilloscope Nicolet 4094C.

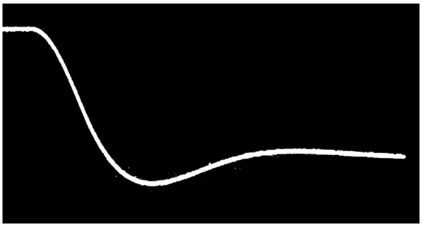


Figure 8.14: $s = 0.2$, $d = .8$

Figure 8.15: $s = 0.6$, $d = .8$

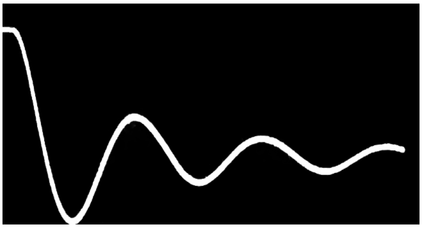


Figure 8.16: $s = 0.8$, $d = .6$

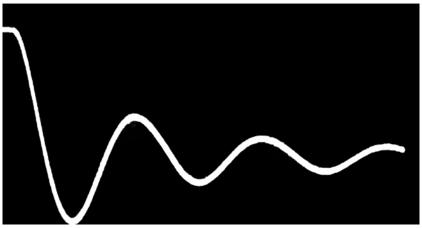


Figure 8.17: $s = 0.8$, $d = 1$

To scale the problem, the maximum values for y, $\dot{y}$ and $\ddot{y}$ must be determined. A good approximation is an undamped harmonic oscillator with an eigenfrequency of

$$\omega = \sqrt{\frac{s}{m}} = \sqrt{\frac{60\ \frac{\mathrm{kg}}{\mathrm{s}^2}}{1.5\ \mathrm{kg}}} \approx 6.3\ \mathrm{s}^{-1}.$$

From $y = y_0 \sin(\omega t)$ it follows that

$$\dot{y} = y_0 \omega \cos(\omega t) \quad \text{and} \tag{8.5}$$

$$\ddot{y} = -y_0 \omega^2 \sin(\omega t). \tag{8.6}$$

Since the mass-spring-damper system in this example represents a damped oscillator,

$$\max_{t>0} |y| \leq y_0$$

holds which yields the following boundary values for $\dot{y}$ and $\ddot{y}$ based on equations (8.5) and (8.6):

$$\dot{y} < y_0 \omega \approx 0.63\ \frac{\mathrm{m}}{\mathrm{s}}$$

$$\ddot{y} < y_0 \omega^2 \approx 4\ \frac{\mathrm{m}}{\mathrm{s}^2}$$

Thus a reasonable scaling for the machine variables would be $[10y]$, $[15\dot{y}]$ and $[\frac{5}{2}\ddot{y}]$ respectively. Time scaling is not performed so that the simulation will yield results in real-time. Based on these scaled machine variables equation (8.3) becomes

$$\left[\frac{5}{2}\ddot{y}\right] = -\frac{5}{2m}\left(\frac{d}{15}\left[15\dot{y}\right] + \frac{s}{10}\left[10y\right]\right).$$

To perform a scaled simulation run the coefficient potentiometers for the problem parameters s, d and $1/m$ must be set to $[s/10]$, $[d/15]$ and $[5/2m]$.

After scaling of a problem is done a so-called *static test* should be performed to make sure that the patch panel setup is correct. The basic idea of such a test which larger analog computers often feature as a special mode of operation, is simple: All integrators are switched into a mode where they act as simple summers. Then test signals which are derived from a static solution of the problem's equations are injected and the resulting values are compared against this static solution which has been prepared manually with the help of a stored-program digital computer. To perform a static test of the program developed in this section values like $y = 0.05$ and $\dot{y} = 0.5$ have been selected. Using equation (8.3) with these values yields

$$\ddot{y} = -\frac{1}{1.5}(3 \cdot 0.5 + 60 \cdot 0.05) = -3.$$

During the static test these values for y and $\dot{y}$ will be injected in the circuit with all integrators acting as summers. The value on the output of the rightmost summer in figure 8.12 should not deviate from the above value for $\ddot{y}$ by more than about 10^{-3} or less.

8.4 Predator and prey

While the mass-spring-damper system could be described by a single differential equation, the following problem of modeling a simple ecosystem containing predators and prey requires two coupled differential equations. Systems like these were first studied in 1925 by Alfred James Lotka[9] and independently in 1926 by Vito Volterra.[10] They developed the so-called Lotka-Volterra differential equations which describe simple ecosystems. This development was caused by statistical data gathered by the *Hudson Bay Company* in the years between 1850 and 1900. Their data sets contained the number of rabbits and lynxes that were captured in these years and showed an interesting periodicity.

In the following a closed-world ecosystem with enough food for the prey is assumed. The number of lynxes and rabbits is denoted by l and r respectively. This ecosystem can be described by the following two coupled differential equations:

$$\dot{r} = \alpha_1 r - \alpha_2 r l \tag{8.7}$$

$$\dot{l} = -\beta_1 l + \beta_2 r l \tag{8.8}$$

The coefficients are: α_1 representing the rate of birth for rabbits, α_2 the rabbit death-rate caused by lynxes catching rabbits,[11] β_1 describing the death-rate of lynxes, and β_2 the increase of the lynx population due to food supply in the form of rabbits. The food supply for rabbits is assumed to be unlimited.

[9] 03/02/1880–12/05/1949
[10] 05/03/1860–10/11/1940
[11] The *capture cross section*, so to speak.

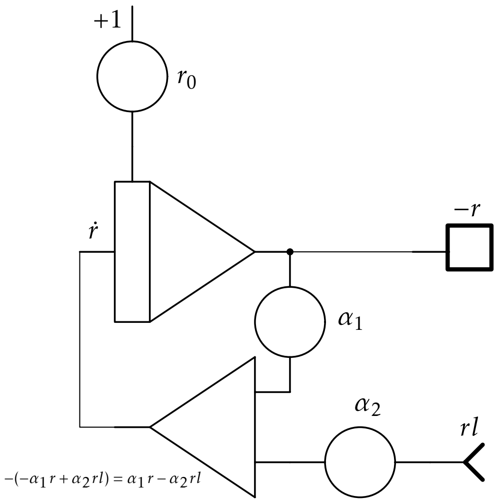


Figure 8.18: Computing $-r$

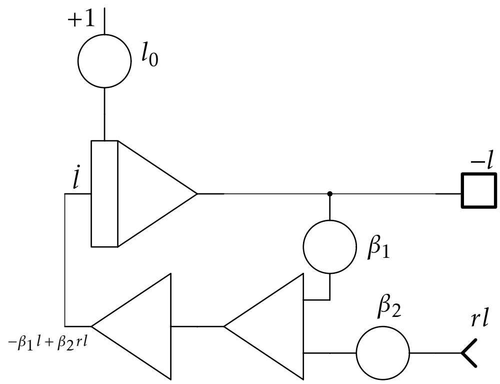


Figure 8.19: Computing $-l$

The increase of the rabbit population is described by two terms: $\alpha_1 r$ which describes the natural birth rate which is assumed to be proportional to the current population size, and $\alpha_2 rl$ describing the number of rabbits killed by lynxes. Analogously the lynx population is decimated by the natural death rate $\beta_1 l$ since there are no predators in this simulation specialized on lynxes. On the other hand, the lynx population increases with $\beta_2 rl$ due to additional food intake.

The two coupled differential equations (8.7) and (8.8) can be converted into the two sub-circuits shown in figures 8.18 and 8.19 by using the Kelvin feedback technique. Without the common input rl, each of these circuits would just model an ever faster increasing rabbit population[12] and a decreasing lynx population. Noting that the two summers connected in series in figure 8.19 just act as a summer without sign-inversion readily yields the simplified circuit shown in figure 8.20 saving both summers and thus not only simplifying the overall setup but also decreasing the error of the simulation. Since both of these circuits share a common input signal, rl, the overall setup for the predator-prey model shown in figure 8.21 combines both sub-circuits by means of a multiplier.

The overall setup of the predator-prey simulation, consisting of a Nicolet 4094C digital storage oscilloscope on the left and a Telefunken tabletop analog computer type RA 741, is shown in figure 8.22. The resulting curves are shown in figure 8.23. In this case only a qualitative study of the simple ecosystem has been performed. Scaling a system of coupled differential equations is rather difficult, so a qualitative study is outside the scope of this example.[13] The typical curve form and the phase leg between rabbit and lynx populations are clearly visible.

[12] Natural death for rabbits has not been modeled.

[13] An example for scaling such a system of coupled differential equations can be found in [Schwarz 1971][pp. 369 ff.].

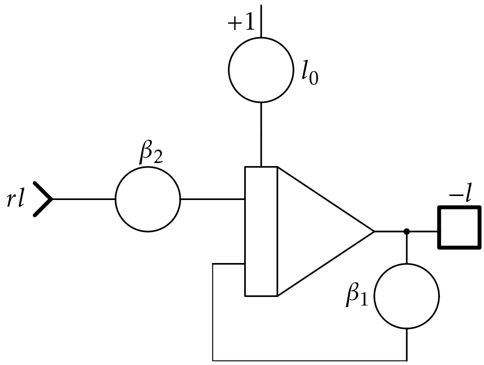


Figure 8.20: Improved circuit for $-l$

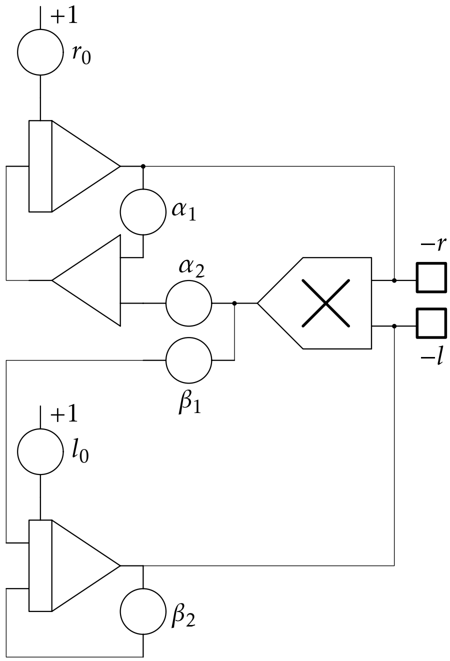


Figure 8.21: Completed predator-prey setup

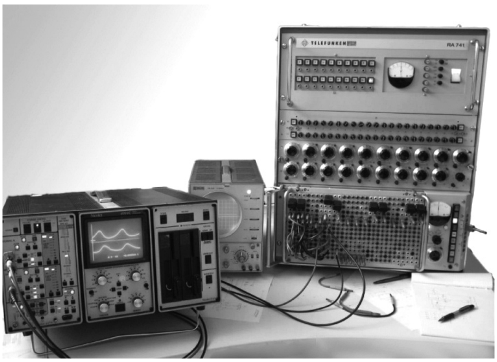


Figure 8.22: Setup for the predator-prey simulation

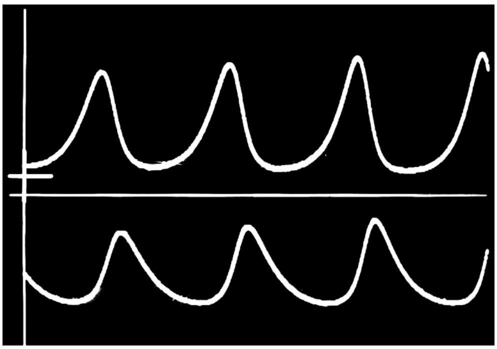


Figure 8.23: Results of the predator-prey simulation

8.5 Bouncing ball

The following example simulates a ball bouncing in a box in real-time on an analog computer with an oscilloscope display of the ball's movement. The setup follows closely an example published by Telefunken in the 1960s[14] which was used back then

[14]Cf. [Telefunken/1] for the original example and [Pfaltzgraff 1969] for a much more complex but also more precise simulation of the same problem. Other vendors, like Heathkit, also used bouncing ball simulations for marketing, although they went with simpler circuits than the one described here. Such a simplified simulation is described in [Winkler 1961][pp. 199 ff.].

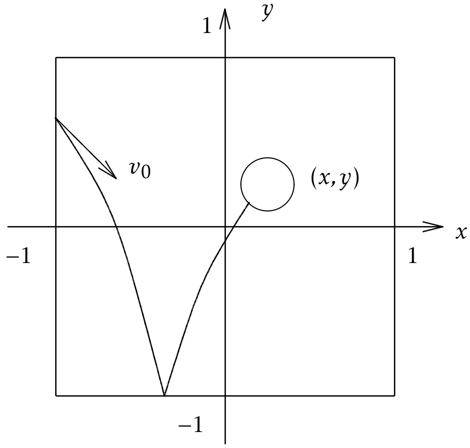


Figure 8.24: Movement of the bouncing ball

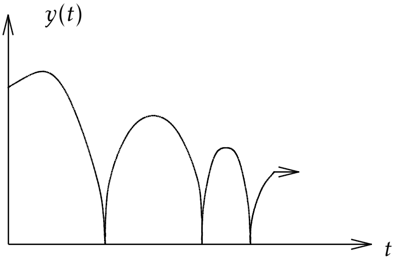


Figure 8.25: y-component of the bouncing ball

as a marketing example showing the power of electronic analog computers. A real-time simulation like this would have required a very expensive stored-program computer whereas the tabletop analog computers of this time were much cheaper. Figure 8.24 shows the basic parts of the simulation. A ball is thrown into a quadratic box with an initial velocity v_0 at a given initial y position. It is then accelerated by the force of gravity until it hits the floor of the box where it rebounces fully elastically. The ball loses energy over time due to air-friction. When the ball hits the left or right wall it rebounds.

Since the x- and y-component of the time dependent ball position are independent from each other, they can be simulated independently, too. Figure 8.25 gives an impression of the approximate y-position of the ball with respect to time. Basis for the computation of y is the acceleration $\ddot{y}$ that consists of the gravitation and the acceleration caused by the elastic collision of the ball with the bottom of the box. $\ddot{y}$ can be described by

$$\ddot{y} = -g + d\dot{y} \begin{cases} +\frac{c}{m}(|y|+1) & \text{if } y < -1 \\ -\frac{c}{m}(y-1) & \text{if } y > 1. \end{cases}$$

Integrating twice over $\ddot{y}$ yields the ball velocity $\dot{y}$ and position y:

$$\dot{y} = \int_0^T \ddot{y}\,\mathrm{d}t + \dot{y}_0$$

$$y = \int_0^T \dot{y}\,\mathrm{d}t + y_0$$

The circuit yielding the y position of the ball is shown in figure 8.26. The integrator on the left integrates over the sum of three terms: The gravitational force which is set by

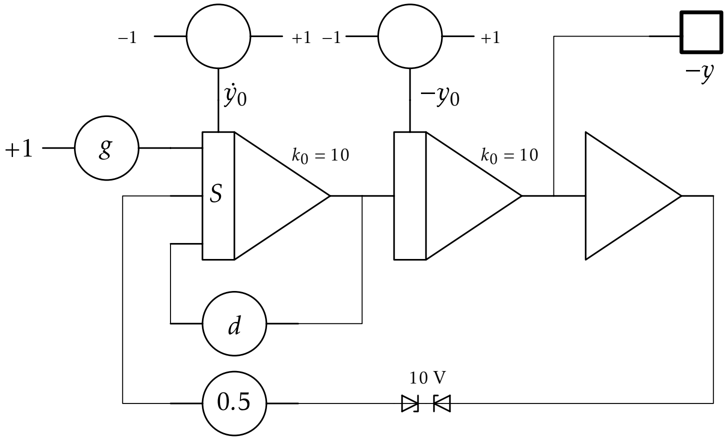


Figure 8.26: Computing the y position of the bouncing ball

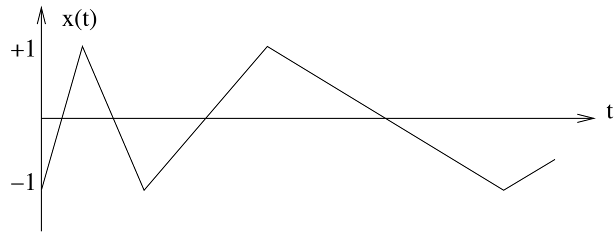


Figure 8.27: x-portion of the ball's movement

the coefficient potentiometer labeled g, the rebound-acceleration which comes from the inverter at the right side, and a damping signal that is proportional with factor d to the ball's velocity $\dot{y}$ that is available at the output of this integrator.

There are several things to be noted about this circuit: First of all, the rebounce-acceleration must by very high to yield a realistic behavior of the ball. Thus an integrator input with a weight far exceeding the standard inputs of 1 and 10 is necessary. Therefore this acceleration is fed directly to the summing junction through a coefficient potentiometer.[15] The acceleration signal representing the rebounce of the ball has be generated whenever the ball hits either the floor or the ceiling of the box. Normally, two comparators would be used to implement this, but the small analog computer chosen in this example did not have enough comparators, so a different approach was necessary. The two Z-diodes in the output line of the inverter have the effect that they pass a signal only when it exceeds their respective ZENER voltage. Both Z-diodes are selected to yield a ZENER voltage of 10 Volts in each direction thus representing the floor and ceiling of the box. The inputs of the both integrators are connected to free coefficient potentiometer thus allowing arbitrary initial values for the ball.

Generating the x-position of the bouncing ball is simpler. It is assumed that its velocity in x-direction is linearly decreasing while changing the direction every time the ball

[15] A better alternative would be to use a free coefficient potentiometer which is ungrounded.

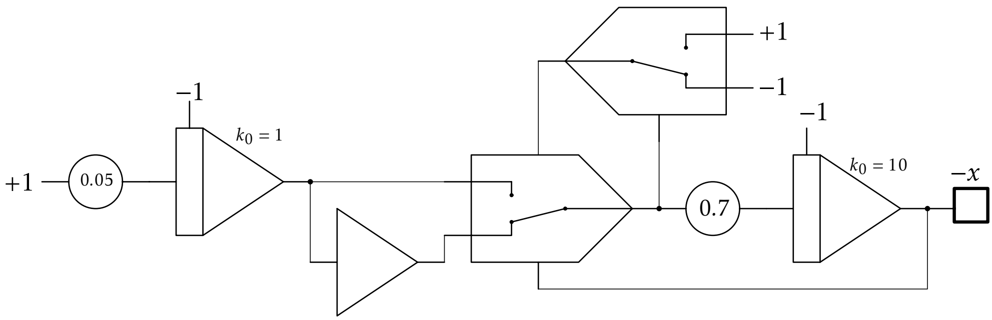


Figure 8.28: Computing the x position of the bouncing ball

hits a wall. Figure 8.27 shows the ball's x-position with respect to time while figure 8.28 shows the associated computer setup.

The leftmost integrator starts with an initial value of −1 and integrates over a small constant value of 0.05 with an integration time constant $k_0 = 1$, so the output of the integrator will linearly decrease, starting at the value +1 representing the maximum velocity of the ball. Its output signal is thus

$$\dot{x}(t) = -\left(\int\limits_0^T 0.05\,\mathrm{d}t - 1\right).$$

In effect, this velocity signal is integrated by a second integrator with a time constant ten times faster to generate the x-position. The two comparators detect the ball hitting the left or right wall, where the left comparator also performs the sign-reversal of the velocity signal.

Now that both, the x- and y-position of the bouncing ball are known, a sine-/cosine-generator circuit is necessary to display an actual ball shape on an oscilloscope screen. Basically this is done as described in section 8.1 but it is a bit more complicated since the generated signal-pair must have a rather high frequency to generate a steady figure on the screen and the amplitudes of the two signals must be constant. Figure 8.29 shows the enhanced sine-/cosine-generator circuit used.

The basic structure of this circuit corresponds to that shown in figure 8.2. Two integrators and an inverter form a loop effectively solving a differential equation of second degree as (8.1). To achieve a high value for ω the integrators need high input weights which is achieved by connecting free potentiometers to their respective summing junction inputs. To assure that the amplitude of the generated sine-/cosine-signal pair does not decay, a positive feedback loop is setup between the output of the inverter and one input of the right integrator. This feedback only has to provide a small amount of signal amplitude, thus it is attenuated by a coefficient potentiometer set to 0.02. Without additional means this would yield an ever increasing signal amplitude which would drive the computing elements into overload. To counteract this effect, a pair of Zener diodes is used as a negative feedback on the right integrator. These connect the inte-

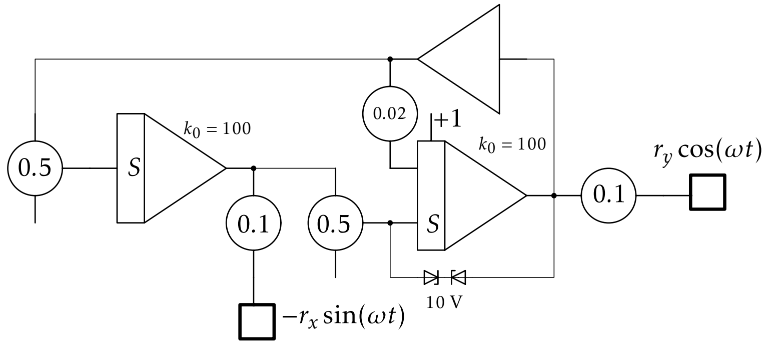


Figure 8.29: Generating a high-frequency sine-/cosine-signal pair for displaying the ball

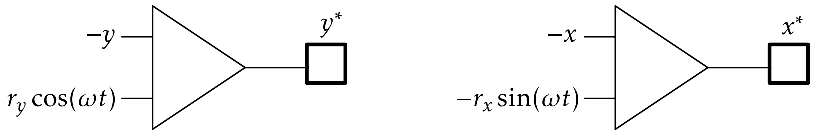


Figure 8.30: Generating the signals to display the ball's outline on the oscilloscope

grator's output directly to its summing junction, effectively limiting its output signal to their Zener voltage.[16]

Using the two summers shown in figure 8.30 the position signals $-x$ and $-y$ of the ball and the attenuated sine-/cosine-signals can now be combined yielding two output signals x^* and y^* which are connected to the x- and y-inputs of the oscilloscope used to display the simulated bouncing ball.

Figure 8.31 shows the setup of the bouncing ball simulation on a Dornier DO-80 analog computer while figure 8.32 shows a time exposure shot of a simulation run.

8.6 Car suspension

The simulation of vibrating mechanical systems was and still is of high commercial value since vibrations in typical transport systems, support structures, machines etc. can cause negative effects ranging from making a train ride uncomfortable to dangerous situations when eigenfrequencies are met and even rigid structures collapse.

In the late 1950s the simulation and analysis of the dynamic behavior of cars gained a lot of interest and most car and railway vehicle manufacturers installed rather large

[16]This circuit or variations thereof are often employed when a high-frequency sine-/cosine-pair is required which does not need to be spectrally clean. "High-frequency" is a bit hyperbolized – a frequency of a couple kHz is sufficient for a flicker-free display. Due to the combination of positive and negative feedback, harmonics are introduced, but for driving a display their amplitude is small enough not to cause any visible distortions.

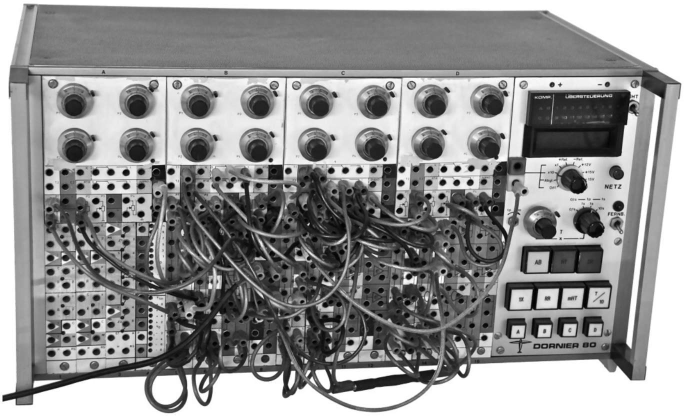


Figure 8.31: Setup of the bouncing ball simulation (Dornier 80)

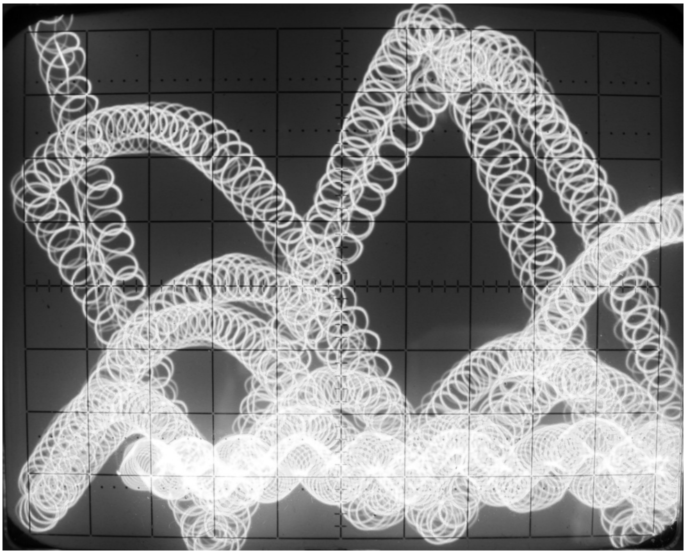


Figure 8.32: Time exposure shot of a bouncing ball simulation run (photo taken by Tore Sinding Bekkedal, reprinted with permission)

analog computing facilities. The rationale behind this is made clear by the following quotation from Robert H. Kohr:[17]

> *"However, the general trend toward heavier cars with softer tires and the increasing adoption of power steering and air suspensions calls for a complete dynamic analysis of the automobile with a view to gaining a basic understanding of the automobile's behaviour on the road."*

[17] Vehicle Dynamics Section, Engineering Mechanics Department of the *General Motors* Research Staff at Warren, Michigan, cf. [McLeod et al. 1958/6][p. 1994].

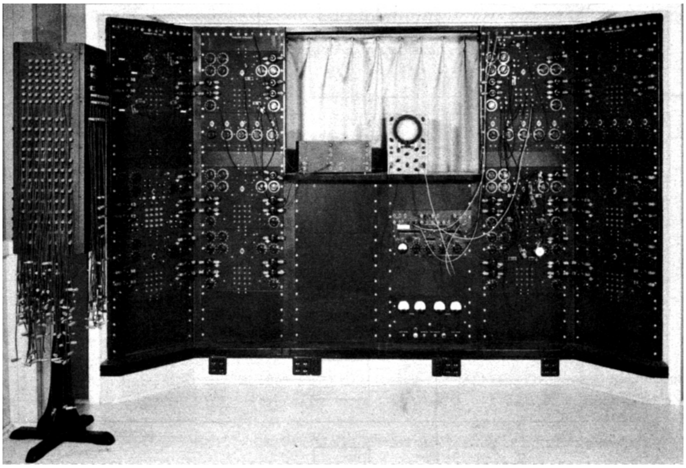


Figure 8.33: Analog computer installed at the Pullman-Standard Car Manufacturing Company in the early 1950s (see [Roedel 1955][p. 42])

Realistic simulations of a car suspension system require up to seven degrees of freedom[18] and a lot of function generators to model the non-linear behavior of wheels, the suspension springs, and other mechanical parts.[19] Simulations like these were also standard for the development of railway vehicles as this citation shows:[20]

> *"A railroad freight vehicle is a complex dynamic system consisting of numerous interrelated physical components [...] Comprehensive models for such a system will generally consist of a large number on nonlinear simultaneous differential equations with complicated functional relationships between the variables to be integrated. The computer implementation of these models will require a significant amount of computer resource to run."*

Due to the complex structure of the bogies of such a vehicle these simulations featured between 17 and 23 degrees-of-freedom.[21] A 5 degree-of-freedom simulation of a single bogie is described in [Malstrom et al. 1977] while [Roedel 1955/2] covers the simulation of the dynamic behavior of railway vehicles in general. Figure 8.33 shows the analog computer installation that was in use at the Pullman-Standard Car Manufacturing Company in the early 1950s. This installation was mainly used to simulate bogies and complete railway vehicles.

[18]Cf. [McLeod et al. 1958/6][p. 1994]. To make the analog model as realistic as possible, actual road profiles were measured and stored on analog tape units to be used as inputs for the simulations. Using multiple read heads, the time delayed excitation of the wheels of the vehicles simulated was implemented.

[19]A thorough introduction to the mathematics of vibrating multi-mass systems can be found in [Macduff et al. 1958][pp. 193 ff.] and [Telefunken/2].

[20]See [Heller et al. 1976][p. 2].

[21]Cf. [Heller et al. 1976][p. 39].

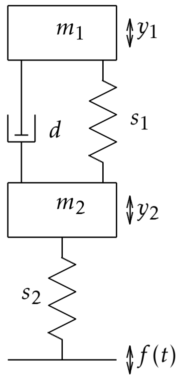


Figure 8.34: Two-mass system

A much simpler simulation was demonstrated in the early 1960s on an exhibition. This setup, which employed a rather small tabletop analog computer[22] generated a lot of interest and turned out to be quite an eye-catcher.[23] This setup is recreated in the following example. A simple couple two-mass system with linear springs and damper as shown in figure 8.34 will be simulated.[24] This mechanical system is described by the following two coupled differential equations

$$\begin{aligned}0 &= m_1\ddot{y}_1 + d(\dot{y}_1 - \dot{y}_2) + s_1(y_1 - y_2)\\ 0 &= m_2\ddot{y}_2 + d(\dot{y}_2 - \dot{y}_1) + s_1(y_2 - y_1) + s_2(y_2 - y_3),\end{aligned}$$

which yield

$$\ddot{y}_1 = -\left(\frac{d}{m_1}(\dot{y}_1 - \dot{y}_2) + \frac{s_1}{m_1}(y_1 - y_2)\right) \quad \text{and} \tag{8.9}$$

$$-\ddot{y}_2 = \frac{d}{m_2}(\dot{y}_2 - \dot{y}_1) + \frac{s_1}{m_1}(y_2 - y_1) + \frac{s_2}{m_2}(y_2 - f(t)) \tag{8.10}$$

as the starting point for applying the Kelvin method to derive a computer setup. The resulting circuit for equation (8.9) is shown in figure 8.35. Basically this circuit is an oscillator – two integrators in series with a sign-inverting summer in the feedback path. The second feedback path containing two summers in series damps the oscillation like $e^{-\alpha t}$ where α is defined implicitly by d/m_1.

Equation (8.10) yields the circuit shown in figure 8.36, which is also basically a damped oscillator circuit. Both circuits are coupled through the variables y_2, $\dot{y}_2$, $-\dot{y}_1$ and $-s_1(y_2 - y_1)/m_1$. Basically these two coupled circuits already describe the dynamic behavior of the two-mass system. To replicate the exhibition setup from the 1960s a pretty real-time display is required. This display should show a car-model viewed from the side with the car's body represented by m_1 and the wheels, which are not modeled separately, represented by m_2.

[22] A Telefunken RAT 700 was used.

[23] This setup is described in [Behrendt 1965].

[24] A more detailed description of such a two-mass system, including scaling, can be found in [Giloi et al.][pp. 48 ff.], [EAI TR-10][pp. 44 ff.] and [Carlson et al. 1967][p. 91].

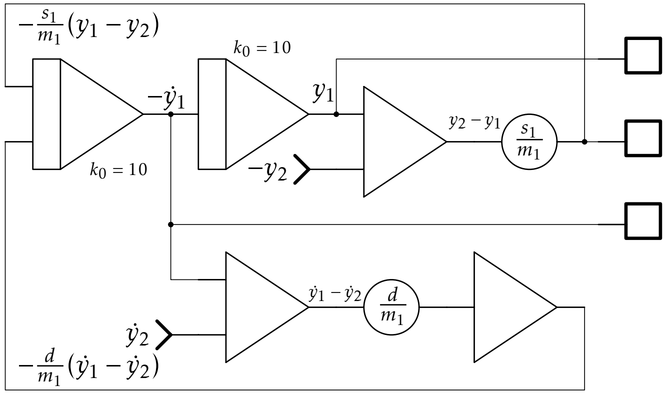


Figure 8.35: Circuit corresponding to equation (8.9)

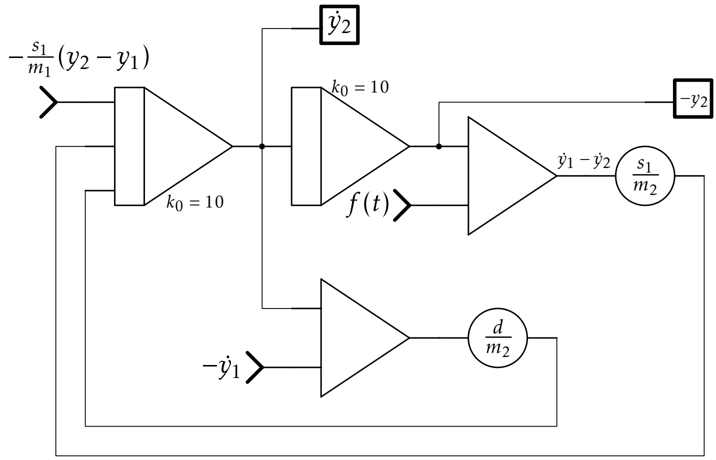


Figure 8.36: Circuit corresponding to equation (8.10)

Whenever closed figures are to be displayed on an oscilloscope's screen, a high-frequency sine-/cosine-signal pair is required. Therefore an additional circuit as shown in figure 8.29 is required. In addition to that, an electronic switch is necessary so that three figures (two wheels and the car frame) can be displayed in rapid succession on the oscilloscope. The two wheels can be directly generated by feeding the x- and y-deflection channel of the oscilloscope with the sine-/cosine-pair. The car frame is also based on this signal pair but the cosine part will be suitably deformed by a diode function generator.

The position signals y_1 and $-y_2$ are then added to the cosine-signal while the sine-portion is fed directly to the oscilloscope after sufficient attenuation by coefficient potentiometers.

The overall program setup is shown in figure 8.38. In addition to the circuits described above, it also contains the three electronic switches to rapidly switch the oscilloscope's

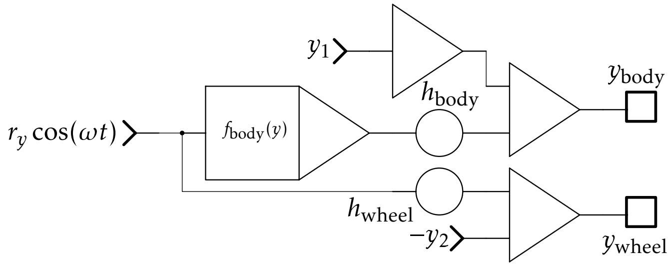


Figure 8.37: Generating the signals for the graphic representation of the two-mass system

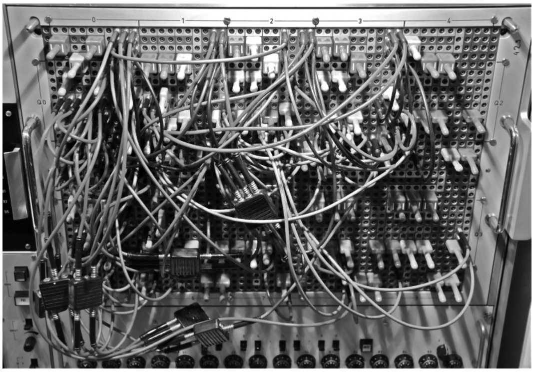


Figure 8.38: Setup of the car suspension simulation (Telefunken RA 770)

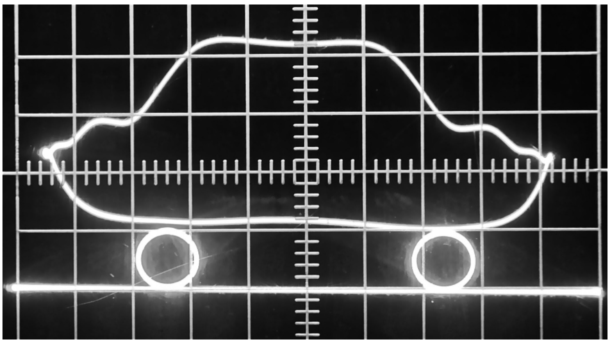


Figure 8.39: Display of the car suspension simulation

inputs between the three x/y-outputs generated by the simulation. Additionally a sweep generator like the one shown in figure 8.4 has been implemented to explore the dynamic response of the two-mass vibrating system to various excitation frequencies and amplitudes. Also, a second excitation source is available in this setup since the analog computer used, a Telefunken RA 770, is equipped with a noise generator. A screen shot of the display generated by this setup is shown in figure 8.39.

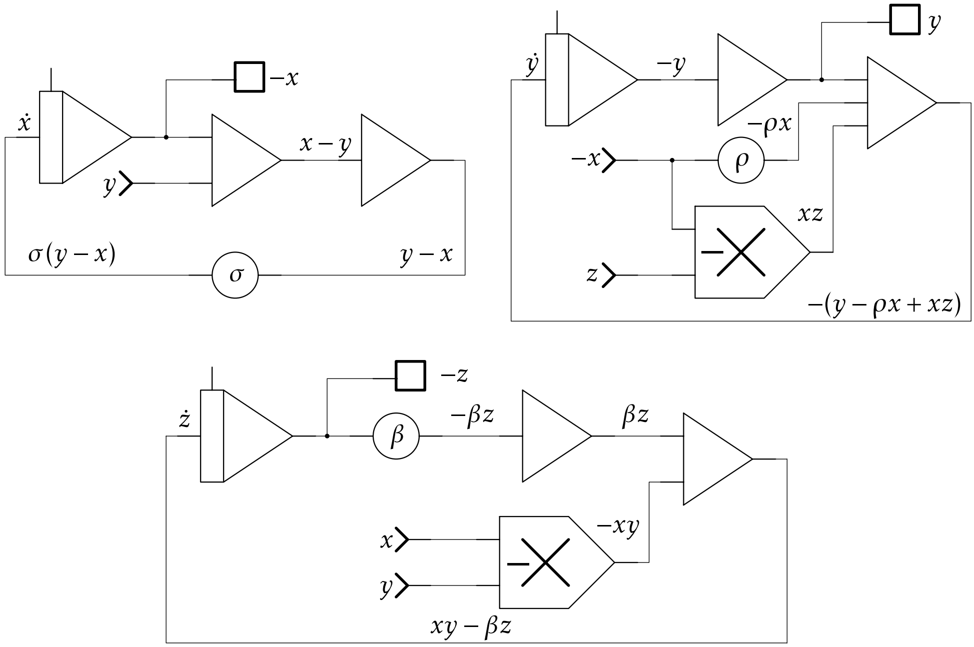


Figure 8.40: Straightforward setup for equations (8.11), (8.12) and (8.13)

8.7 LORENZ attractor

Nearly everybody has seen the fascinating pictures of the so-called LORENZ attractor, discovered by EDWARD N. LORENZ.[25] This attractor is actually a plot of the phase-state of a system of the following three coupled differential equations:[26]

$$\dot{x} = \sigma(y-x) \tag{8.11}$$

$$\dot{y} = x(\rho-z)-y \tag{8.12}$$

$$\dot{z} = xy-\beta y \tag{8.13}$$

Applying the KELVIN method to these three equations yields the circuits shown in figure 8.40.

Obviously, these circuits can be further simplified saving some computing elements and thus improving the precision of the computation. The circuit shown on top in figure 8.40 features two summers in series which can be eliminated by using multiple inputs of the integrator. The resulting circuit is shown on the upper left of figure 8.41. A further change has been implemented as well: Instead of generating $-x$ based on y, the signs have been reversed since $-y$ is readily available from another circuit.

[25] 05/23/1917–04/16/2008

[26] Recent work on this can be found in [TUCKER 2002]. Here a proof is given that the attractor is robust, "*it persists under small perturbations of the coefficients in the underlying differential equations*". It is also proven that the Lorenz equations "*support a strange attractor*" which was conjectured by Lorenz as early as 1963.

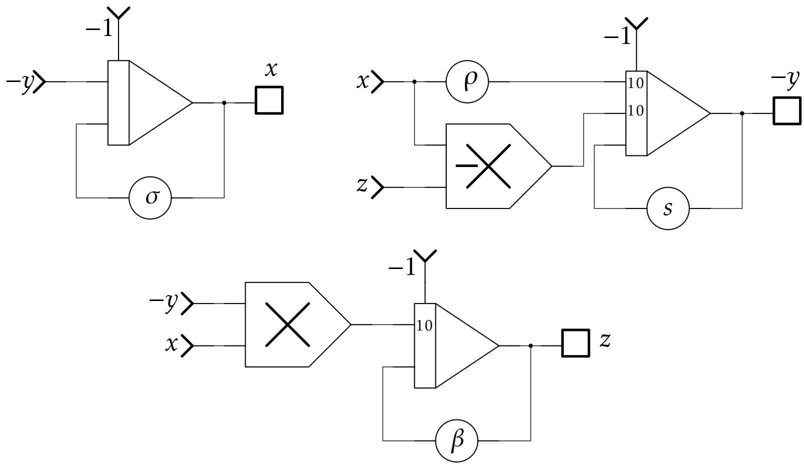


Figure 8.41: Simplified setup for equations (8.11), (8.12) and (8.13)

The circuit shown in the upper right of figure 8.40 can be simplified in the same way: The two summers in series are eliminated by means of multiple inputs of the integrator. The input x is changed to $-x$ which is the output of the circuit described above, so this second circuit will yield y which, in turn, feeds the preceding circuit. The resulting circuit is shown on the upper right of figure 8.41.

Applying the same rationale to the lower circuit of figure 8.40 yields the corresponding circuit of figure 8.41 which saves two summers. It should be noted that the sign-inverting multipliers are most easily setup by using quarter-square multipliers with one of the input-signal pairs reversed with respect to polarity.

Figure 8.42 shows the setup of the program generating a picture of the Lorenz attractor. In this case, a Telefunken RA 770 precision analog computer has been used. With parameters

$$\sigma = 0.357$$
$$\rho = 0.1$$
$$\beta = 0.374$$

a display like the one shown in figure 8.43 can be obtained.

8.8 Projection of rotating bodies

The following example shows how an analog computer operating in fast repetitive mode can be used to generate displays with the projection of three-dimensional figures.[27] The following example, which generates a two-dimensional display of a rotating three-dimensional spiral, is based on [Telefunken/3] and [Hitachi 1967].

[27] Even higher dimensional figures have been projected by means of an analog computer. An example of the projection of a four-dimensional cube is described in [Mackay 1962][p. 137].

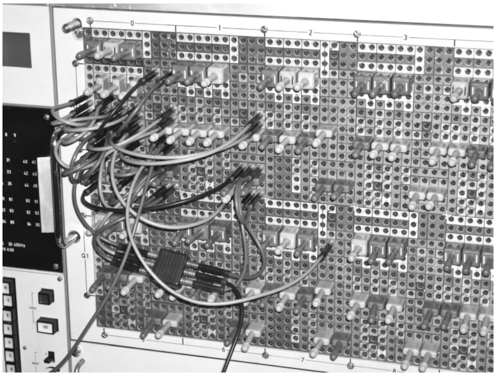


Figure 8.42: Setup for the Lorenz attractor (Telefunken RA 770)

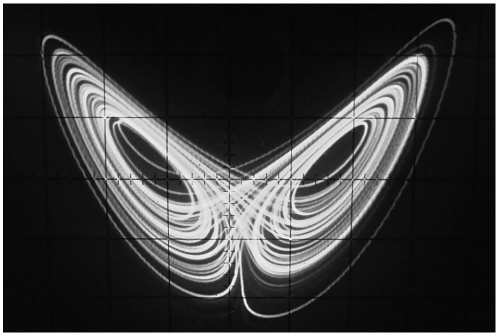


Figure 8.43: Screen shot showing a Lorenz attractor

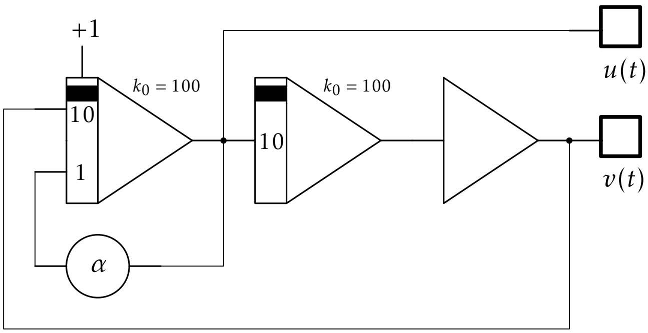


Figure 8.44: Generating $u(t)$ and $v(t)$ for the spiral

At first the three coordinates of the spiral must be generated. This is done by a modified sine-/cosine-generator shown in figure 8.44 which yields the following two signals:

$$u(t) = \cos(\omega_{\text{rep}} t) e^{-\alpha k_0 t}$$
$$v(t) = -\sin(\omega_{\text{rep}} t) e^{-\alpha k_0 t}$$

The attenuating term $e^{-\alpha k_0 t}$ results from the negative feedback path of the leftmost integrator, containing the coefficient potentiometer labeled α. This signal-pair describes a two-dimensional spiral, so a third circuit is necessary to generate the third coordinate signal.

This coordinate is of the form

$$w(t) = -1 + \int\limits_{t=0}^{t_{\text{rep}}} \beta k_0 \, \mathrm{d}t$$

and is easily generated by a single integrator as shown in figure 8.45. It is to be noted that the three integrators shown in figures 8.44 and 8.45 are marked with a black

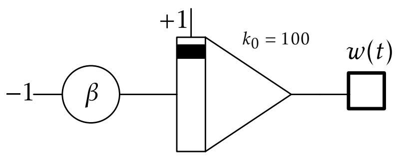


Figure 8.45: Generating $w(t)$ for the spiral

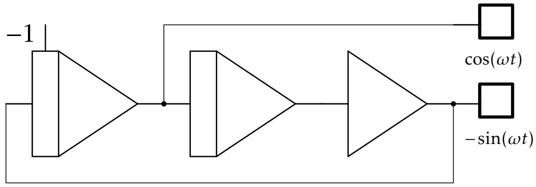


Figure 8.46: Generating a sine-/cosine-signal pair for the rotation of the spiral

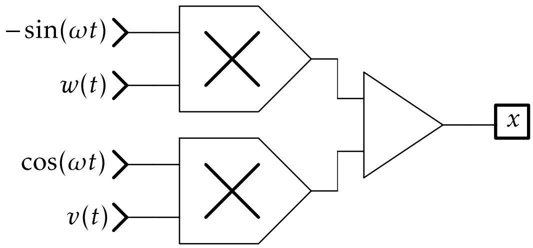


Figure 8.47: Rotation and projection

bar. This denotes that these integrators are part of an integrator-group that is controlled by an external clock generator. To generate the display of a slowly rotating three-dimensional spiral, two time scales are necessary: A very short time scale for the integrator group that generates the spiral's coordinates $u(t)$, $v(t)$, and $w(t)$ a much slower time scale for a second integrator group that yields a sine-/cosine-pair for the rotation of the figure. The three integrators denoted by the black bar have a common time scale factor of $k_0 = 100$ and use their respective inputs with weight 10 yielding an overall speedup of 1000. This integrator group must be run in repetitive mode to yield the necessary three coordinates of the spiral over and over again in rapid succession.

The second integrator group which is controlled by the standard controls of the analog computer is shown in figure 8.46. This circuit is a common sine-/cosine-signal generator which is used to control the rotation of the figure. This is done by the circuit shown in figure 8.47 which implements

$$
\begin{aligned}
&Y = u(t) \quad \text{and} \\
&X = w(t)\sin(\omega t) - v(t)\cos(\omega t).
\end{aligned}
$$

The resulting signals X and Y are used to feed the display oscilloscope.

This computer setup now requires either a second timing-circuit to control the integrator group operating in repetitive mode or some digital control elements to generate the necessary control signals. Every integrator is basically controlled by two signals named *OP* and *IC* controlling the three modes *operate*, *initial condition*, and *halt*. Figure 8.48 shows these two signals. A short period required to pre-charge the integrator's capacitors is followed by a longer period during which the computation takes place. These two modes of operation are repeated rapidly to generate a nearly flicker-free display of the rotating spiral.

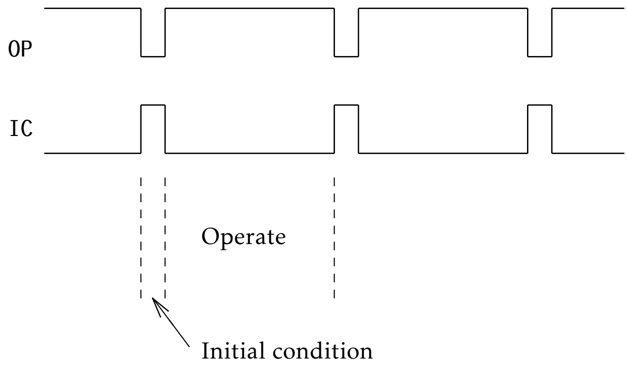


Figure 8.48: Control signals for the integrators operating in repetitive mode

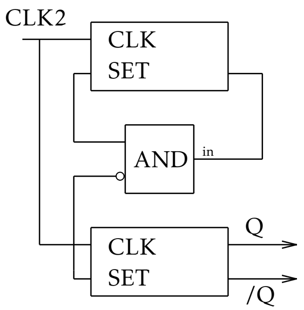


Figure 8.49: Generation of the control signals for the integrators operating in repetitive mode

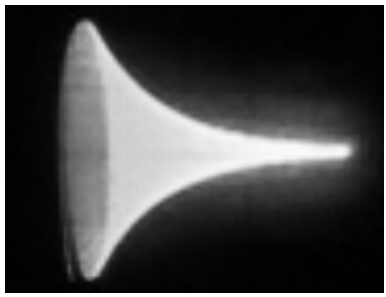


Figure 8.50: Oscilloscope screen capture of a rotating spiral

Figure 8.49 shows the setup of the digital elements to generate this control signal pair. The gate labeled *AND* is unusual as it is not used as an and-gate but as a signal driver with two outputs, one inverted and one non-inverted. A snapshot of the display of the rotating three-dimensional spiral is shown in figure 8.50.

8.9 Conformal mapping

This last and most complex example covered in this chapter is based on [Telefunken/4] and [Sydow 1964][p. 123]. It implements a so-called *conformal mapping* which is an angle-preserving mapping. Such mappings are described either explicitly by analytic functions of a complex variable like

$$w = f(z) = u(x,y) + \mathrm{i}v(x,y) \quad \text{where } z = x + \mathrm{i}y$$

or implicitly by the solution of ordinary differential equations.[28] Implementing such a conformal on an analog computer is quite straightforward and only requires the

[28]Cf. [Heinhold 1959][pp. 46 ff.] for more information on this second variant.

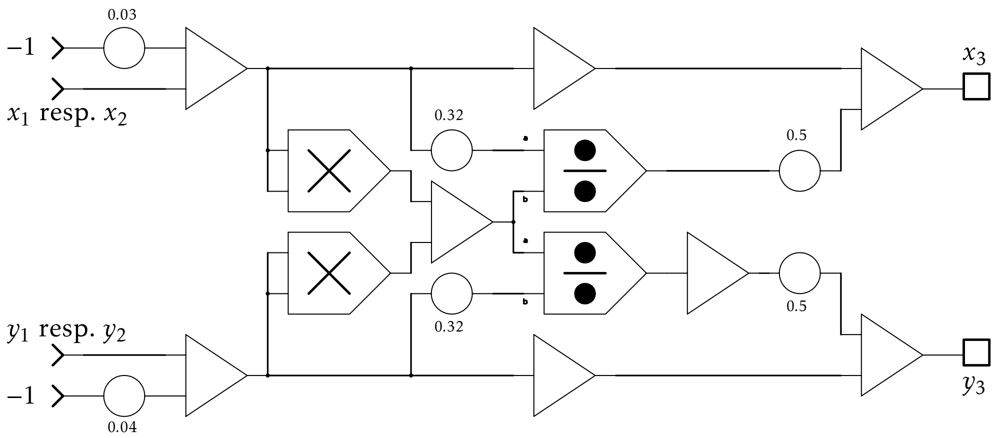


Figure 8.51: Transforming a unit circle into a Joukowsky air foil using a conformal mapping

complex variable to be handled explicitly as a two-dimensional vector. A particularly interesting conformal mapping is

$$w(z) = (z - z_0) + \frac{\lambda^2}{z - z_0} \tag{8.14}$$

which describes a so-called Joukowsky[29] *airfoil*.[30] Applying this mapping to a unit circle yields an airfoil that can then be used to analyze the flow of a medium like air around this structure. Implementing equation (8.14) on an analog computer is based on these two equations which describe the real and imaginary part separately:

$$u(x(t), y(t)) = (x(t) - x_0(t)) + \frac{\lambda^2 (x(t) - x_0(t))}{(x(t) - x_0(t))^2 + (y(t) - y_0(t))^2} \tag{8.15}$$

$$v(x(t), y(t)) = (y(t) - y_0(t)) - \frac{\lambda^2 (y(t) - y_0(t))}{(x(t) - x_0(t))^2 + (y(t) - y_0(t))^2} \tag{8.16}$$

The mechanization of equations (8.15) and (8.16) is shown in figure 8.51. This circuit has been derived by applying Kelvin's feedback method. This circuit is used two times in a time-shared manner: Applied to a sine-/cosine-signal describing a unit circle, it will yield the outline of a Joukovsky airfoil. Applied to a coordinate-pair describing the path of a particle flowing around a rotating cylinder, it will transform this path to the path of a particle flowing around such an airfoil.

[29] Nikolay Yegorovich Joukowsky, 01/17/1847–03/17/1921

[30] A detailed description of this type of airfoil can be found in [Thwaites, ed. 1987][pp. 112 ff.] and [Eck 1954][pp. 237 f.]. Applying suitable changes to equation (8.14) more complex and realistic airfoils can be described. [Ashley et al. 1985][pp. 52 ff.] described two such enhanced mappings based on the Mises-, the Kármán-Trefftz transformation and the Theodorsen mapping. Numerical approaches for the generation of such airfoils and the simulation of particle flows are described in [Zingg 1989].

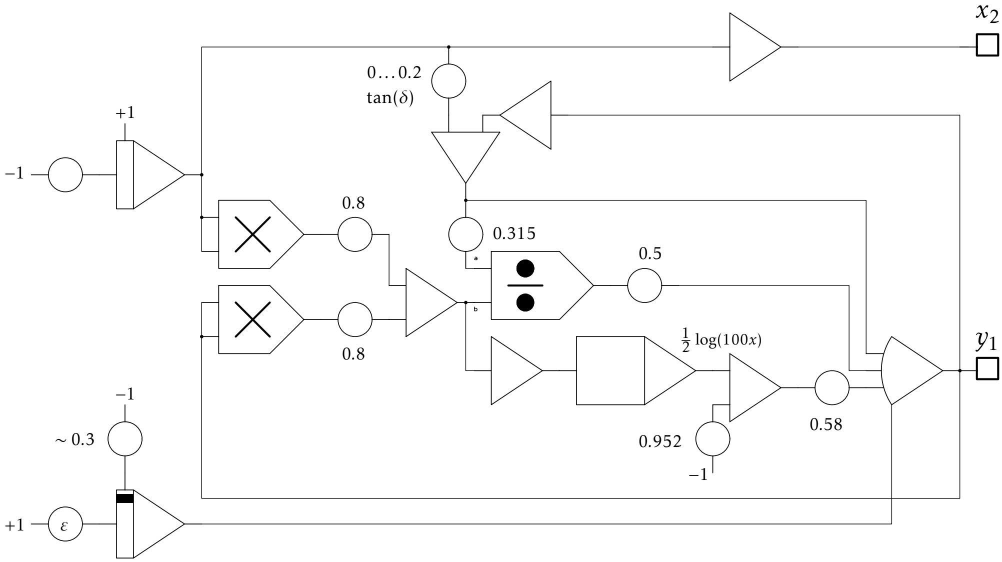


Figure 8.52: Generating the flow lines

The path of a particle flowing around a rotating cylinder is described by the complex velocity potential

$$f(z) = v_0\left(e^{i\delta}z + \frac{e^{-i\delta}r^2}{z}\right) - i\frac{\Gamma}{2\pi}\ln(z),$$

which is realized by the circuit shown in figure 8.52. The lower leftmost integrator yields the x-coordinate of a particle, which path is determined by two parameters: The angle of attack is controlled by the term $\tan(\delta)$ and the y-position of the particle with respect to the rotating cylinder. This last variable is generated by the lower leftmost integrator which is controlled separately from the remaining integrators of the computer setup.[31]

The overall setup of the Telefunken RA 770 precision analog computer is shown in figure 8.53. Figure 8.54 shows a single flow line around a Joukovksy airfoil on top and a group of flow lines spaced equidistantly on the bottom. Such a group can be generated by employing the lower leftmost integrator of the circuit shown in figure 8.52 which is under separate control.

This integrator operating in repetitive mode requires its own dedicated digital control circuit. In addition to this, electronic switches under digital control are necessary to time-share the conformal mapping circuit shown in figure 8.51 between a sine-/cosine-signal pair describing a unit circle and the output signals of the flow line circuit. Neither these switches nor the digital control are shown here since these are implemented pretty straightforward.

[31] This is, again, denoted by a black bar at the top of the integrator symbol.

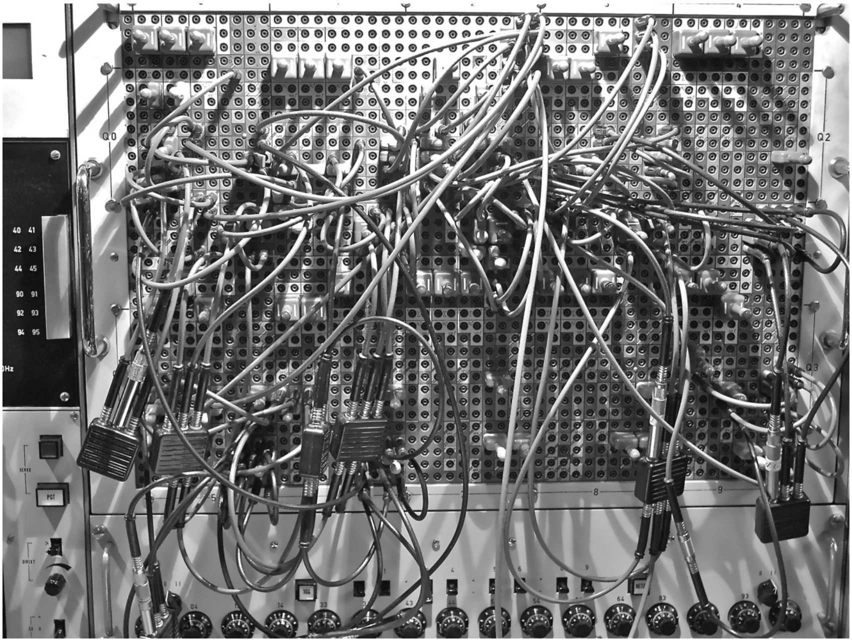


Figure 8.53: Implementation of the conformal mapping and the generation of flow lines

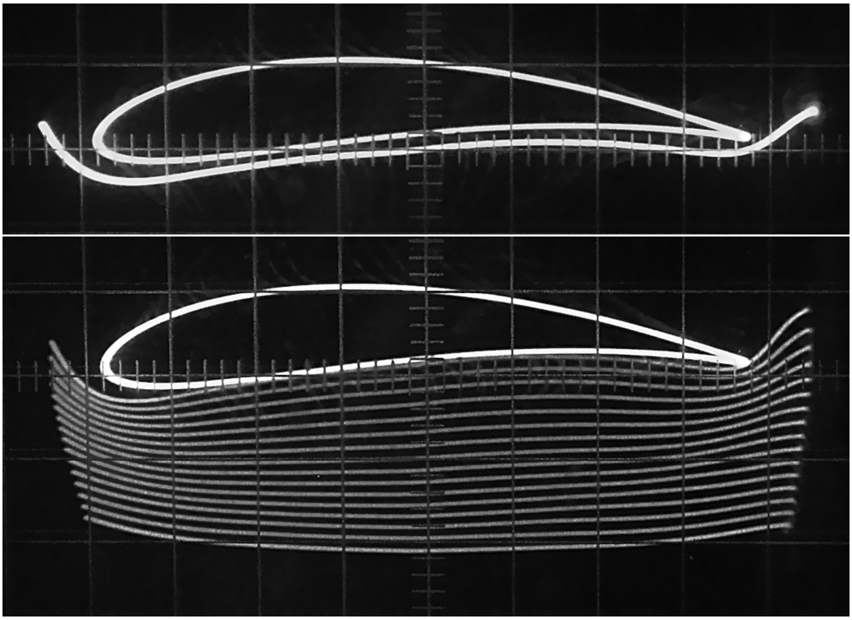


Figure 8.54: Joukowsky air foil with flow lines

9 Hybrid computers

9.1 Systems

As early as in the mid-1950s it became clear that analog electronic analog computers had also their drawbacks that could not easily be overcome by analog electronic means. One of the main problems that were identified was the cumbersome process to setup function generators and, more generally, the generation of functions of more than one variable, a task often being required especially in aerospace applications. These two problems belong to a class which is readily solved with a traditional stored-program digital computer. Thus the idea of coupling analog computers with such digital computers, forming a so-called *hybrid computer*, was born. An early account of this rationale is given in [Waltman 2000][p. 69] which described the application of analog computers for the simulation the X-15 aircraft:

> *"The thought of building up another set of function generators like those already in use was probably considered, but not by me or any other X-15 simulation programmers. We had had enough of those fuses and dinky pots. The idea of using a digital computer to do this job was unanimously and immediately accepted. No discussion was needed. We were going hybrid."*

To couple two such different computing systems, so-called *Analog-Digital-Converters* and *Digital-Analog-Converters* are necessary.[1] These devices translate analog voltages into digital signals and vice versa. In addition to that, typical hybrid computers allow placing the operation of the integrators under program control by the digital computer and feature digital interface lines which can be used to trigger interrupts on the digital system by comparators on the analog side and to control electronic or relay switches on the analog system from the digital computer. Figure 9.1 shows exemplarily the structure of an EAI 690 hybrid computer system.

Figure 9.2 shows a so-called *ADDAVERTER* that was built by *Space Technology Laboratories*[2] in 1956. The system shown features 15 ADC- and 10 DAC-channels[3] with a precision of ±0.1% which quite well matches the precision of large analog computer installations of that time. The development of the ADDAVERTER was the result of previous research efforts of Ramo-Woolridge Corporation and *Convair Astronautics*. Both

[1] *ADC* and *DAC* for short.
[2] A subsidiary of *Ramo-Woolridge Corporation*.
[3] A fully expanded system had up to 15 DAC channels.

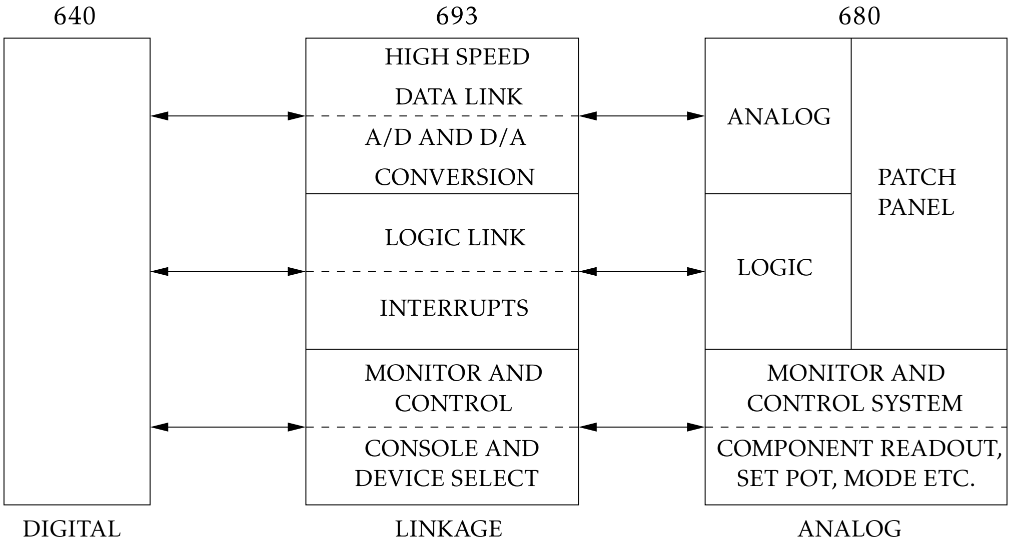


Figure 9.1: Block diagram of the EAI 690 hybrid computer system consisting of an EAI 640 digital computer, the EAI 693 linkage system and an EAI 680 analog computer (cf. [EAI 690])

companies were working on simulations of intercontinental ballistic missiles which involved very large analog computer setups.[4] The special requirements are summarized by [McLeod et al. 1957][p. 1127] as follows:

> *"It was imperative that the simulation be done in real time to allow inclusion of weapon system hardware, and to conserve operating time. This ruled out an all-digital simulation because the large number of individual computations could not be made fast enough, and because analog equipment would be necessary to connect to some of the weapon system components which were to be included. An all-analog simulation was ruled out by accuracy requirements, particularly with respect to the navigational problem. Clearly a combined simulation was necessary to fulfill the requirements [...]"*

The system that was finally put into operation at Convair Astronautics consisted of an IBM 704 stored-program computer, an ADDAVERTER, and a large EAI analog computer. The overall cost for this system was 2.3 million US$ for the IBM 701, 200,000 US$ for the ADDAVERTER and 1.6 million US$ for the analog computer[5] – a very substantial sum at that time. Although this system was never used for its proposed task that had been solved by conventional means in the meantime[6] it served as a demonstration of the efficiency and power of hybrid computers.[7] Subsequently the market

[4] Cf. [Bekey et al. 1968][p. 154].

[5] See [McLeod et al. 1957][p. 1130].

[6] [Bekey et al. 1968][pp. 154 f.] notes that *"[i]n the time that elapsed between the original specification of the hybrid system and the delivery and acceptance of the conversion equipment, it was established that the guidance and control problems associated with missile flight are not closely coupled and can be studied separately. Consequently the basic problem for which the hybrid computing system was designed, vanished."*

[7] A collection of typical problems that would be solved with hybrid computers is given in [Benham 1970].

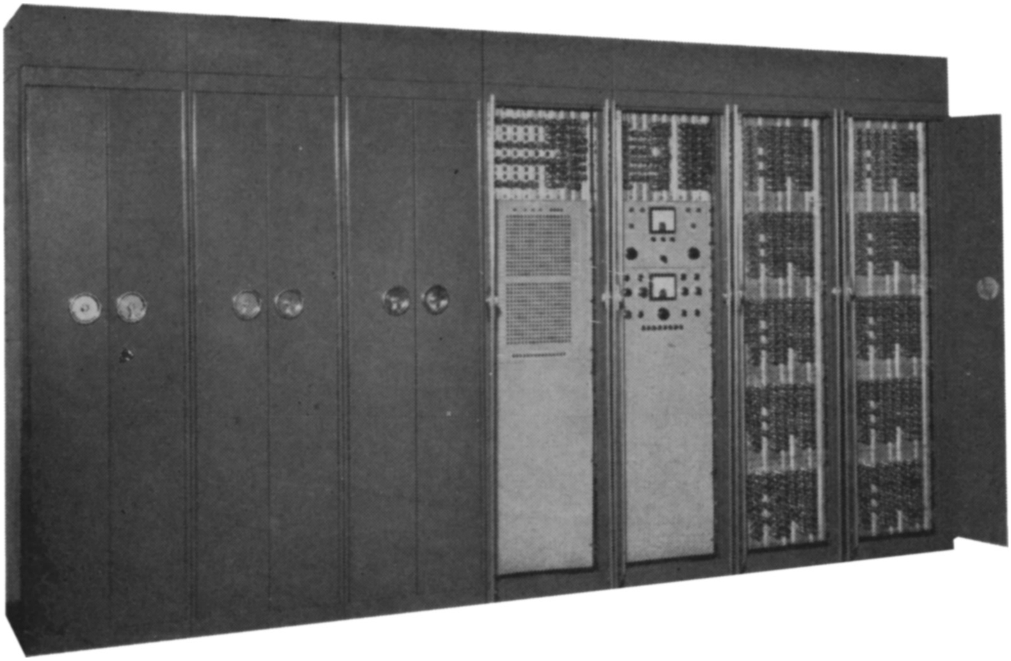


Figure 9.2: Space Technology Laboratories ADDAVERTER (see [McLeod et al. 1957][p. 1129])

demanded more and more hybrid systems which was aided by the development of a fully transistorized precision analog computers featuring a machine unit of ±100 V. This analog computer was developed by *Comcor* in 1964 and offered the same precision as previous vacuum tube based machines but at lower cost, lower power consumption, and longer maintenance intervals.[8]

A typical hybrid system of the 1960s is shown in figure 9.3. This particular system consists of a SDS-9300 digital computer[9] on the left and a Comcor *CI-5000* analog computer on the right with a multi-channel recorder in front of it.

Digital computers suitable as part of a hybrid computer setup must be fast enough to keep up with the analog system in a simulation. Especially simulations of nuclear reactors showed that the analog computer could easily outperform the digital system and was thus slowed considerably by this in a hybrid setup. In some cases a hybrid computer approach was only about three to ten times faster than a pure stored-program solution.[10]

Apart from this obvious idea of coupling analog and stored-program digital computers, other schemes were devised as well. In [Karplus et al. 1972], an analog coprocessor consisting of a complex resistor-network is described, which aids a digital computer in the solution of parabolic partial differential equations.

[8] See [Bekey et al. 1968][p. 155].

[9] See [Scientific Data Systems/2].

[10] See [Frisch et al. 1969][p. 36].

Figure 9.3: Hybrid computer installed at the Department for Electrical Engineering of the Naval Postgraduate School in the late 1960s (with permission of Robert Limes)

Another interesting idea is the implementation of hybrid number systems in which values are represented by a combination of continuous signals like voltages and sequences of bits, thus forming a generalized form of floating point numbers consisting of a mantissa part and an exponent. Here the analog part of a number interpolates between two values that can be described by a digital value.[11]

An interesting recent implementation (2005) of a hybrid computer system is described in [Cowan 2005]. The analog subsystem is integrated on a VLSI[12] chip containing 416 functional blocks, interconnecting switches and the necessary control logic. This highly integrated analog subsystem is also proposed to act as an accelerator for the stored-program digital computer.[13]

9.2 Programming

With respect to the analog part, programming a hybrid computer is not different from programming a stand-alone analog computer. It is the stored-program digital computer as part of a hybrid computer configuration that requires special precautions regarding its programming since it has to work closely coupled to the analog computer. This typically requires fast acting interrupt routines, reentrant code etc. Requirements like these are typical for systems used for process automation tasks, thus basic programming paradigms from this area can be applied to hybrid computers as well. Fur-

[11] More information about this can be found in [Giloi 1963][pp. 268 f.] and [Skramstad 1959].

[12] Short for *Very Large Scale Integration*.

[13] See section 12.

ther complicating is the high amount of interactivity the analog system allows, which the digital computer must equally support.

Basically, two modes of operation must be distinguished in a hybrid system:

***Alternating operation*:** In this mode the analog and digital subsystems work in an alternating fashion. A typical example is the task of value-optimization where the digital processor generates a set of initial values that are then used by the analog subsystem to solve a set of differential equations. The result of this run is then read out by the digital computer to serve as the basis for the next set of initial values and so on. This mode is the easiest from a programmer's perspective since the operation of the digital computer is not time critical. Furthermore the digital computer has complete control of the analog computer characterizing this mode as a master-slave operation.

***Simultaneous operation*:** This mode of operation is by far more complicated than the alternating operation of the digital and analog parts of a hybrid computer. Here both computers operate in a closely coupled fashion and the digital computer must respond to interrupts generated by analog comparators, it must supply functions of more than one variable etc. Often neither the analog nor the digital computer has the role of a master or a slave system.

Historically two approaches for programming the stored-program subsystem of a hybrid computer have been pursued. The first is based on extending traditional programming like *FORTRAN* or *ALGOL* with special library calls and sometimes additional language features to support the control of the necessary interfacing devices as well as the implementation of fast interrupt routines.[14]

A good overview of the various software subsystems included in the EAI 8900 hybrid computer system is given in [Bekey et al. 1968][p. 181]. Apart from an extended FORTRAN dialect, a special purpose language called *HYTRAN* is also available as well as a powerful macro assembler which is often required to implement highly time-critical routines. A very useful feature of typical hybrid system is the possibility to check not only the stored-program digital processor but the analog computer as well by means of diagnostic service routines. Such diagnostics not only include basic tests like rate-checks[15] but also static tests as described in section 5.5.

A complex, yet instructive example of the application of a hybrid computer system to an optimization task can be found in [Witsenhausen 1962]. This paper gives a thorough description not only of the analog computer setup but also describes the necessary control routines written for the digital processor.

[14]A description of an extended ALGOL system can be found in [Herschel 1966] while a more general description of this approach is given in [Feilmeier 1974][pp. 133 ff.].

[15]A *rate check* checks the time-constants of an integrator by applying an input signal of 1 machine units and running the integrator for a precisely determined time-span. The deviation of the output voltage from the expected value at the end of this run is a measure of the accuracy of the integrator.

10 Digital differential analyzers

Since the essence of an analog computer is being an analog, there is no need to limit its implementation to be based on mechanical or analog electronic circuits.[1] In fact, there were analog computers based on digital circuitry, called *Digital Differential Analyzers* or *DDA* for short. It is quite remarkable that the earliest attempts to develop DDAs date back to the late 1940s when *Northrop* started the development of a cruise-missile like system.[2]

Using digital circuits to implement basic computing elements like integrators, summers, and the like, has a number of significant advantages: Drift due to aging of components and temperature gradients etc. is no longer an issue. The precision of a DDA can be substantially better than that of a traditional analog electronic analog computer since it allows to trade time against precision as in a stored-program computer. Sophisticated implementations can even be based on floating point numbers thus virtually eliminating the need for tedious scaling of a computer setup. Power requirements and maintenance costs are substantially lower, compared with analog electronic analog computers etc.

Basically there are two different approaches of implementing DDAs: A straightforward *parallel* DDA consists of a whole complement of typical analog computing elements which can be interconnected by means of a traditional patch panel or the like. An example for this class of machines is *TRICE* which is described in more detail in section 10.4.3. Machines like this exhibit the same amount of fine-grained parallelism as a traditional analog computer, thus essentially being high-performance parallel computers. Nevertheless, this high computational power comes at a rather high cost since the number of computing elements is dictated by the complexity of the problems to be solved. Like an analog electronic analog computer there is no time vs. problem size tradeoff which could be used to tackle complex problems with small machines. Large problems require a vast amount of computing elements which also adds to the complexity of the interconnection scheme.

The second basic approach is based on a time-multiplexed use of a few – in many cases only one – central computing elements. Like a simple stored-program digital computer this *sequential* DDA has a central *arithmetic/logic unit*[3] that is fed with data under control of a sequencing unit that in turn is controlled by a machine readable description of the connections between the virtual computing elements that form a

[1] It is still building an analog, a model, which characterizes a DDA although some definitions like that given in [Michels 1954][p. 2] emphasize the approach to implementing the basic operations with digital elements. The definition given there, "*A digital differential analyzer is an electronic computer which solves differential equations by numerical integration.*" does not grasp the central feature of setting up an analog.

[2] See section 10.4.1.

[3] *ALU* for short.

simulation setup. This second approach is much cheaper due to the significantly reduced complexity and lower number of computing elements required. In addition to that, it scales rather well since larger simulations just take more time whereas they could not be readily implemented on a parallel DDA. The major disadvantage of sequential DDAs is their rather low computational power.

10.1 Basic computing elements

The following section describes the basic computing elements of a DDA regardless of its actual implementation.[4]

10.1.1 Integrators

Central element of a DDA is the integrator that is most easily implemented as an accumulator.[5] This has the distinct advantage over an analog electronic analog computer that every variable can serve as the free variable of integration so that integrals like

$$\int_{y_0}^{y_1} f(x)\mathrm{d}x$$

can be treated directly.

The basic structure of a simple DDA integrator is shown in figure 10.1. In contrast to an analog electronic integrator there are two inputs $(\Delta Y)_i$ and $(\Delta X)_i$. These variables are often represented not as absolute values but as incremental values simplifying the implementation of the DDA and reducing the number of signal lines necessary to interconnect computing elements. These incremental values are often restricted to $\{-1;0;1\}$.[6] Accordingly, $(\Delta Y)_i$ represents the change of the integrand, while $(\Delta X)_i$ is the change of the variable of integration and corresponds to $\mathrm{d}x$.

At the heart of the integrator are two accumulators, the first of which accumulates over the series of $(\Delta Y)_i$ values at its input and yields

$$Y_i = Y_0 + \sum_{j=1}^{i} (\Delta Y)_j$$

where Y_0 is an initial value. The resulting Y_i is now multiplied with the incremental input value $(\Delta X)_i$. Due to the restriction of these incremental values to $(\Delta X)_i \in \{-1;0;1\}$

[4]For further reading see [Forbes 1957], [Winkler 1961][pp. 215 ff.], [Beck et al. 1958], [Klein et al. 1957] [p. 1105], [Goldman 1965], and [Jackson 1960][pp. 578 ff.].

[5]Of course, simple accumulation will not be sufficient for most applications. Nevertheless, this most basic approach to integration sometimes led Rudolph Rutishauser to invent the term *incremental computers* to describe a DDA (see [McLeod et al. 1958/4][p. 1223]).

[6]Since two signal lines are necessary to transmit such values, some DDA implementations restricted the domain of these incremental values to $\{-1;1\}$ which simplifies the hardware implementation but requires constant values to be represented by an alternating sequence of -1 and 1 increments. See [Michels 1954][p. 19].

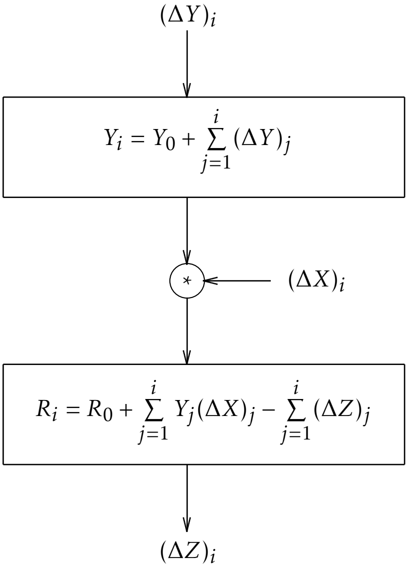


Figure 10.1: Basic integrator of a DDA (cf. [Bendix 1954][p. 1])

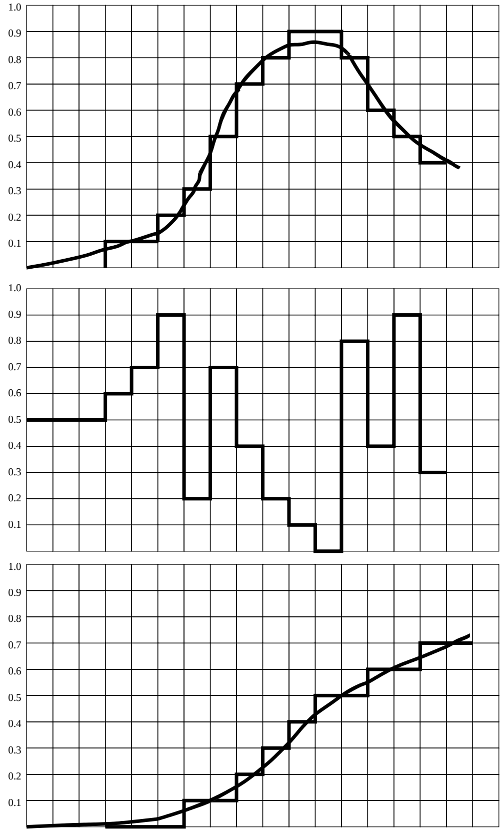


Figure 10.2: Operation of a DDA integrator (cf. [Michels 1954][p. 4])

this multiplication step is extremely simple. The result is then accumulated in the second accumulator[7] yielding

$$R_i = R_0 + \sum_{j=1}^{i} Y_j(\Delta X)_j - \sum_{j=1}^{i} (\Delta Z)_i.$$

The output of this integrator is not the value R_i but instead another incremental value, $(\Delta Z)_i$ representing the over- or underflow of this second accumulator stage. $(\Delta Z)_i$ can now be used as input signal for other computing elements of the DDA. The behavior of such an integrator is depicted in figure 10.2.

The input function and its representation as a sequence of increments and decrements $(\Delta Y)_i$ is shown in the upper graph. The values yielded by the second accumulator

[7] The first accumulator is necessary since $(\Delta X)_i = 0$ is a valid input value. In this case the first accumulator is required to keep track of the incremental integrand. If $(\Delta X)_i$ would be restricted to $\{-1;1\}$ a single accumulator would be sufficient to implement an integrator.

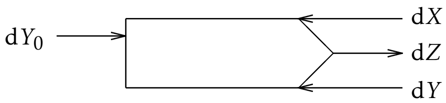


Figure 10.3: Symbol of a DDA integrator

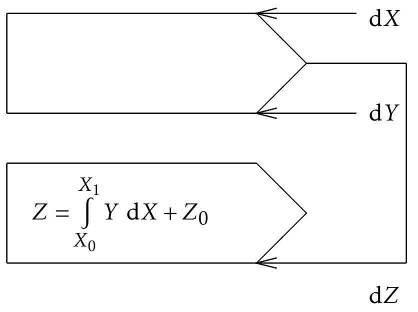


Figure 10.4: Explicit computation of a simple integral (see [Bendix 1954][p. 7])

are shown in the graph in the middle of figure 10.2. Overflows of the accumulator generate a corresponding incremental output signal $(\Delta Z)_i$ which represents the result of the integration and is shown in the lower graph.

Although the use of incremental values throughout a small DDA simplifies its implementation and thus saves cost, it limits the maximum slope of functions that can be represented in a simulation run. To overcome this restriction, additional time scaling of the computer setup is necessary which may result either in non-real-time operation or the requirement of rather high clock rates of the DDA.

The symbol of a DDA integrator is shown in figure 10.3. A typical setup of an DDA to yield the integral over a function is shown in figure 10.4. The first integrator which is fed with the incremental input values $\mathrm{d}X$ and $\mathrm{d}Y$ yields an incremental output signal $\mathrm{d}Z$ that is accumulated in a second integrator to yield the actual value of the integration. The missing second incremental input of this integrator is assumed to be +1.

The integrators of a DDA have an even more central role than the integrators of an analog electronic analog computer. While the latter uses free or open operational amplifiers to create implicit functions by means of a function generator in the feedback path, DDAs use specially configured integrators for this purpose which then are called *Servos*.[8]

10.1.2 Servos

As the name implies, a *Servo* is a computing element that is normally used to minimize some error term thus forming the heart of a servo loop which can be used for example to generate implicit functions. Typical servos are based on DDA integrators but feature only one incremental input $(\Delta Y)_i$ and one accumulator. The output $(\Delta Z)_i$ of this element is defined as

$$(\Delta Z)_i = \begin{cases} +1, & \text{if } Y_i > 0, \\ 0, & \text{if } Y_i = 0, \\ -1, & \text{if } Y_i < 0. \end{cases} \tag{10.1}$$

[8] See [Bendix 1954][pp. 16 ff.].

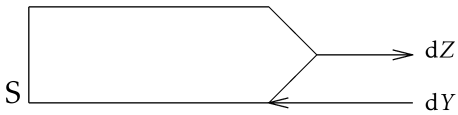


Figure 10.5: Symbol of a DDA servo element

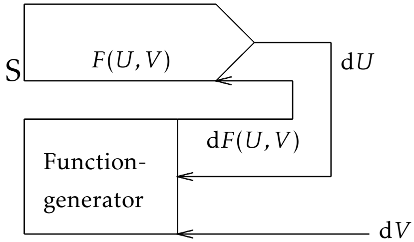


Figure 10.6: Application of a servo element (see [Bendix 1954][p. 18])

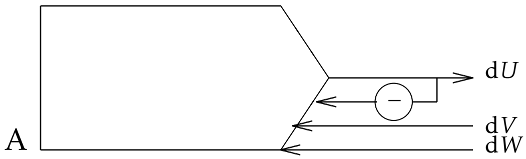


Figure 10.7: Symbol of a DDA summer

Figure 10.5 shows the symbol of a servo element. Some implementations allow an additional input for an initial value, which has been omitted here. Using such an initial value, the threshold level of the servo can be set to an arbitrary value $\neq 0$ which makes it possible to use the servo as a generalized decision element. Figure 10.6 shows a basic application example of a servo element. Using a function generator, an implicit function based on the condition $F(U,V) = 0$ is generated. Usually, $F(U,V)$ is generated in an analytical way by using integrators, summers etc. The input of the servo is the incremental output signal $\mathrm{d}F(U,V)$ of the function generator circuit. The output signal of the servo, defined by (10.1), is in turn used as one input to the function generator while its second input is fed by $\mathrm{d}V$ from some external source. The servo tries to drive the output of the function generator to 0 thus yielding the desired implicit function.

10.1.3 Summers

Typical incremental DDAs like the "Bendix D-12" described in section 10.4.2 reduce the addition of two incremental values $\mathrm{d}U = \mathrm{d}V + \mathrm{d}W$ to solving the equation

$$V + W - U = 0. \tag{10.2}$$

Thus, a summer can be realized by using a modified servo element which features more than one incremental input. Figure 10.7 shows the symbol of a DDA summer having three inputs $\mathrm{d}W$, $\mathrm{d}V$ and $-\mathrm{d}U$. This last term is in turn determined based on equation (10.2), so this generalized servo effectively yields a value $\mathrm{d}U$ which is the desired sum.

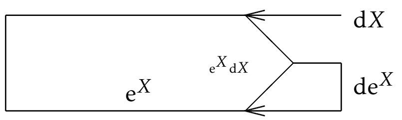


Figure 10.8: Computing e^x with a DDA (cf. [Bendix 1954][p. 9])

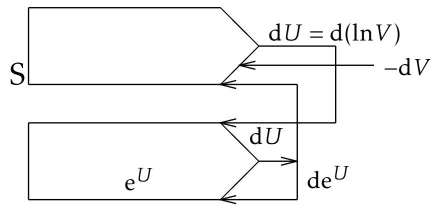


Figure 10.9: Implicitly generating a logarithm function (see [Bendix 1954][p. 19])

10.1.4 Additional elements

In addition to integrators, servos, and summers, typical incremental DDAs feature circuit elements like *output multipliers*, which multiply incremental values by fixed rates thus acting as coefficient units,[9] function generators which are often based on table-lookup techniques, etc.

Since integration can be performed with respect to any variable and not just time t, multiplication in a DDA is most often based on two integrations as described by equation (2.1) in section 2.5.4. An example for this is given in the following section.

10.2 Programming examples

Figure 10.8 shows a DDA-setup to generate an exponential function which is based on a single integrator with its output fed back to its incremental dY-input. Its incremental dX-input is fed with the incremental values comprising the desired exponent value. Using a servo element this function can be used to implicitly generate a logarithm function as shown in figure 10.9. The servo at the top has two inputs: The incremental input value $-dV$ determines the argument of the logarithm function, while the second input is fed with the output of an exponential function generator as described above. At the output of the servo a signal representing $d(\ln V)$ is available. This is due to the fact that the servo tries to ensure $F(U,V) = e^U - V = 0$ holds, so $e^U = V$, which yields $dU = d\ln V$.

Equally simple is the implementation of basic trigonometric functions as shown in figure 10.10 which effectively solves a second order differential equation of the form $\ddot{y} + \omega^2 y = 0$. In contrast to its analog electronic equivalent, the sign reversal operation does not require additional elements since it can be done on-the-fly when incremental values are used.

Figure 10.11 shows a typical DDA setup to perform the multiplication of two incremental variables dV and dU. As already described in section 2.5.4, the product rule can be used to actually perform a multiplication of two values if integration is not limited to only time being the free variable. The two integrators shown on top of the figure perform the actual multiplication of the incremental input values yielding an

[9] See [Bendix 1954][pp. 21 ff.].

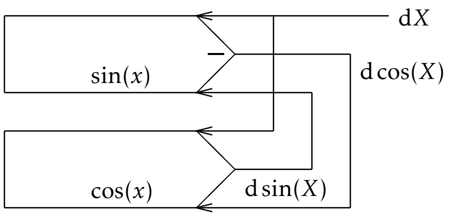


Figure 10.10: Generating a sine-/cosine-signal pair (cf. [Bendix 1954][p. 10])

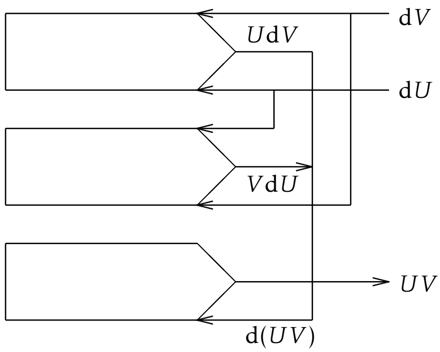


Figure 10.11: Multiplying two incremental values using two coupled integrators (see [Bendix 1954][pp. 10 f.])

incremental output that is fed into a third integrator, which finally yields the desired product.

Additional programming examples of varying complexity for DDAs can be found in [Bendix 1954], [Palevsky 1962] and [Forbes 1972] which focuses on the solution partial differential equations by means of a DDA.

10.3 Problems

The advantages of DDAs, mainly their ability of using any variable as the free variable of integration, and their immunity to drift and other effects decreasing precision as well as the possibility of achieving arbitrary precision by using sufficiently long bit sequences to represent variables, come at a price.

A simple bit-serial incremental DDA working with values -1 and $+1$ only and featuring a word-length of n bit exhibits a basic time-constant of

$$t = \frac{2n}{f}$$

where f denotes its clock frequency. This limits the real-time capability of such a basic DDA considerably.[10] Making n smaller would yield a faster response time but would also diminish the precision of the machine, so increasing f is often the easiest way to speed up an incremental DDA without sacrificing precision. Another approach is to allow incremental values of a larger domain – ideally floating point numbers could be used throughout a DDA. A detailed analysis of DDA specific questions regarding the precision of computations can be found in [Kella 1967], [Kella et al. 1968]. Optimal ratios of the length of the integrator accumulator registers and the incremental values are investigated in [McGhee et al. 1970].

[10] See [Campeau 1969][p. 711].

An additional error source is the integration scheme used in a particular DDA. Using just simple accumulators implementing the rectangle rule is not sufficient for most but the simplest applications. Other techniques, implementing at least a trapezoidal rule, yield better results but complicate the hardware implementation. In addition to that, a DDA cannot speed up a calculation arbitrarily like an analog electronic analog computer where the integrator capacitors can be switched to speed up or slow down a calculation.

Especially the limited precision and speed of incremental DDAs makes them unsuited for many applications as shown by Fred Lesh in [McLeod et al. 1958/3][pp. 488 f.]. There a problem involving the computation of rocket trajectories is treated on a DDA and an analog electronic analog computer. It turned out that the analog computer took only 30 seconds for one solution while the DDA was running for 3 minutes, 30 minutes or up to 5 hours depending on the step width of the integrations. While the analog computer yielded results with a precision of about 98%, the solution obtained in 30 minutes on the DDA only yielded 80%.

10.4 Systems

Over the years quite a lot of DDAs have been implemented. While all implementations are basically of the incremental type, most of them are also based on sequential operation and only one system, namely *TRICE* which is described in section 10.4.3, uses the highly complex parallel implementation approach. Some of these systems are described exemplarily in the following sections.

10.4.1 MADDIDA

Shortly after the end of World War II, in March 1946, *Northrop* started the development of a subsonic cruise missile designated *MX-775* which became known as *Snark*.[11] This system should be able to hit a target at a distance of up to 5,000 miles with a precision of 200 yards much better than the German vengeance weapons V1 and V2.[12] To achieve this level of accuracy, a new guidance system was developed which was based on astro-navigation. Using a built-in telescope, azimuth and elevation angles for sighting predetermined stars were determined to get a navigational fix. This system was coupled with an inertial guidance system.[13]

While Helmut Hoelzer used an analog electronic analog computer in the guidance system of the A4 rocket, the long time-of-flight and the required high precision of Snark required a digital approach to minimize computational errors. Floyd Steele developed the idea of implementing an analog computer using only digital elements which was to be called *DIDA*.[14]

[11] See [Werrell 1985][pp. 82 ff.].
[12] The V2 was admittedly no cruise missile.
[13] See [Ceruzzi 1989][pp. 22 ff.].
[14] Short for *DIgital Differential Analyzer*.

Based on this idea, Donald Eckdahl, Richard Sprague, and Floyd Steele[15] headed a team that finally developed a laboratory prototype of such a DIDA, which was called *MADDIDA*.[16] This system, built by *Hewlett Packard*,[17] was based on a magnetic drum like those used in sophisticated radar systems of World War II. The tracks of this drum housed the bit-serial accumulators of the integrators as well as the necessary address data to represent the interconnection of the computing elements which time-shared a single central ALU. This remarkable prototype is shown in figure 10.12.[18]

Clearly visible on the left is the small magnetic drum with its read/write-amplifier tubes mounted top down. To the right the heart of the system containing the active logic elements implementing the bit-serial arithmetic/logic unit can be seen in its maintenance position.

In 1950 MADDIDA was demonstrated computing Bessel functions to John von Neumann[19] who was excited and characterized the system as *"a most remarkable and promising instrument"*.[20] Although no actual guidance system for Snark resulted from the work on MADDIDA,[21] it became the master pattern of DDAs to follow. A direct successor was built in a small production batch and used mostly for aerospace research.

10.4.2 Bendix D-12

In the early 1950s *Bendix* developed and marketed the *D-12*, a DDA quite similar to MADDIDA. Figure 10.13 shows the DDA on the right with the operator desk on the left. On this desk, from left to right, are the control console, an xy-recorder and a console typewriter.

The D-12, operating on a serial stream of decimal values represented by four bits each, is based on a magnetic drum that contains the integrator registers as well as the necessary information about signal sources and destinations. The system can work in two modes with either 30 integrators operating at 200 iteration steps per second or with 60 integrators running at half the speed.[22] Apart from the simple integration scheme described in section 10.1.1, the D-12 integrators can also work based on

$$\frac{(\Delta Y)_i + (\Delta Y)_{i+1}}{2}$$

yielding better results than those obtained by the rectangle rule.[23]

---

[15]See [Ceruzzi 1989][p. 25].

[16]Short for *MAgnetic Drum DIfferential Analyzer*. Due to reliability problems of this prototype, it was often pronounced as *MAD IDA*.

[17]See [Ceruzzi 1989][p. 25].

[18]See also [Popular Mechanics 1950].

[19]12/28/1930–02/08/1957

[20]Cf. [Eckdahl et al. 2003].

[21]Snark finally employed an analog electronic analog computer for its guidance, see [Ceruzzi 1989][p. 25]. Nevertheless, a DDA was developed for the *Polaris* guidance system, see section 11.15.7.

[22]This is a good example of the capability of a sequential DDA to exchange speed against problem complexity.

[23]See [Bendix 1954][pp. 11 ff.].

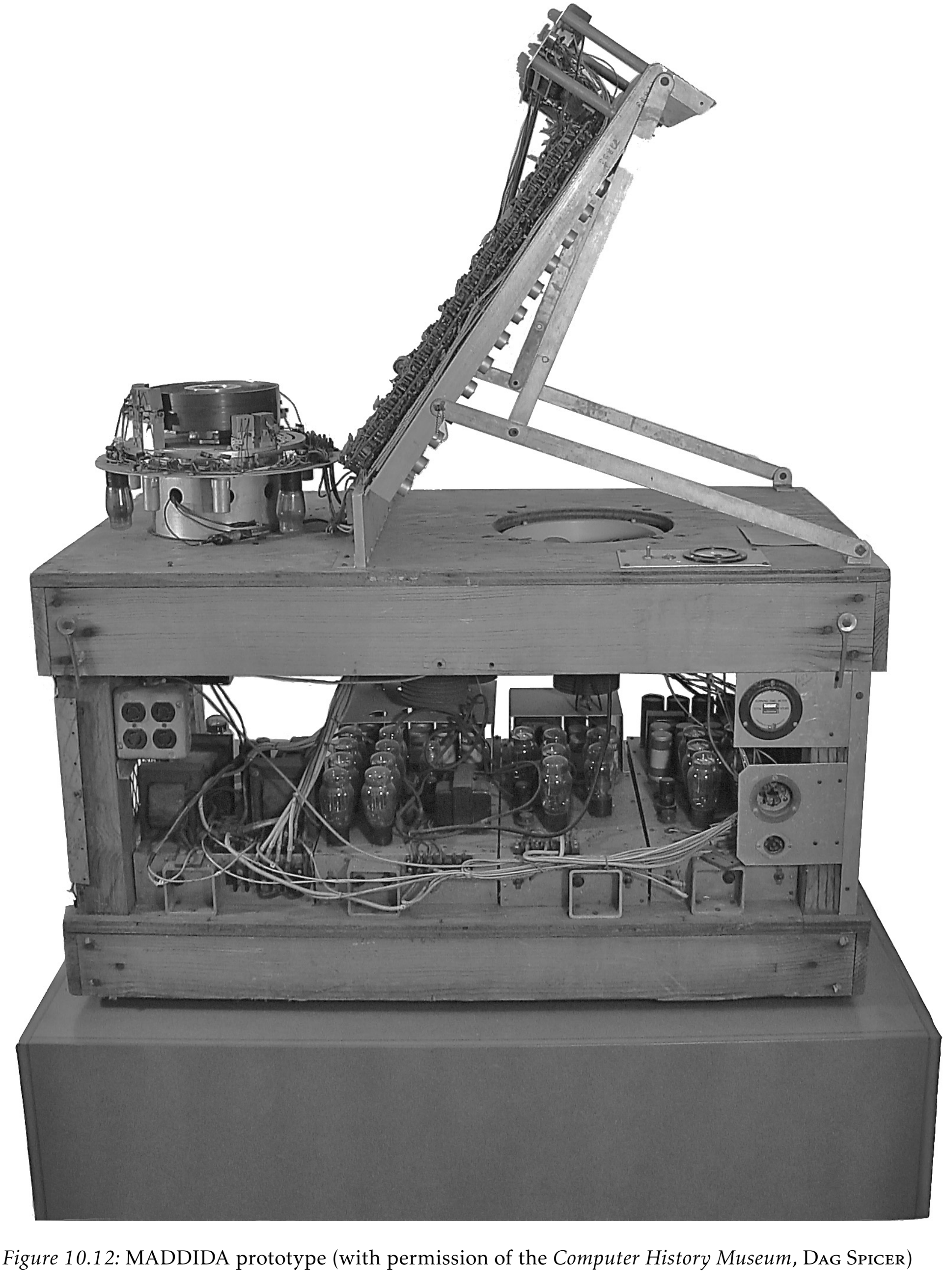


Figure 10.12: MADDIDA prototype (with permission of the *Computer History Museum*, Dag Spicer)

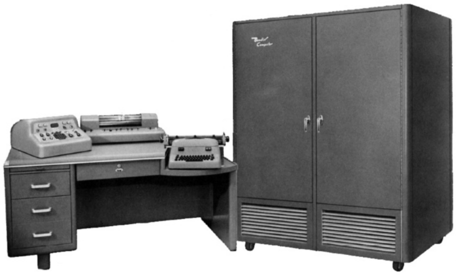


Figure 10.13: Bendix D-12 DDA (cf. [Bendix][p. 2])

Figure 10.14: Bendix D-12 coupled with a Bendix G-15D stored-program computer (see [Klein et al. 1957][p. 1105])

While MADDIDA was intended as a stand-alone system, the D-12 could be coupled directly to a Bendix stored-program digital computer *G-15D* as shown in figure 10.14. On the left hand side the G-15D stored-program digital computer can be seen while the DDA is on the right. The enclosure housing the DDA is considerably smaller than that of the stand-alone version due to the fact that the G-15D and the D-12 share a common magnetic drum which is housed in the cabinet of the G-15D stored program computer, thus saving a considerable amount of hardware.

A typical program for the D-12 is shown in figure 10.15. The four integrators, labeled (4) to (8), solve the differential equation

$$\ddot{x} + k(x^2 + 1)\dot{x} + x = 0.$$

Integrator (9) controls the printout of the values generated by the integrators (3), (4), and (5) on the console typewriter. The integrators (0) and (1) generate the value-pair $(x(t), t)$ that is used to control plotter #1. A phase-space plot is generated on plotter #2 based on the signal-pair $(\dot{x}(t), x(t))$ which is generated by the integrators (18) and (19). The remaining three integrators (10), (11), and (12) perform some post-processing of the signals required for the phase-space plot.

This DDA setup is described by the program shown in figure 10.16. Each line describes one computing element, the number of which is specified in columns 2 and 3. The initial value used for integrators is determined by the numbers written in columns 6 to 14.[24] Integrator control is done with columns 16 to 20:

Column 16: This column controls the mode of operation for the integrator. The values "1" to "4" select integrator-mode with one out of four different integration schemes, while a value of "5" switches the integrator into summing mode whereas "6" specifies a servo.

[24] All values are represented as fixed-point values with one decimal position before and 5 after the decimal point.

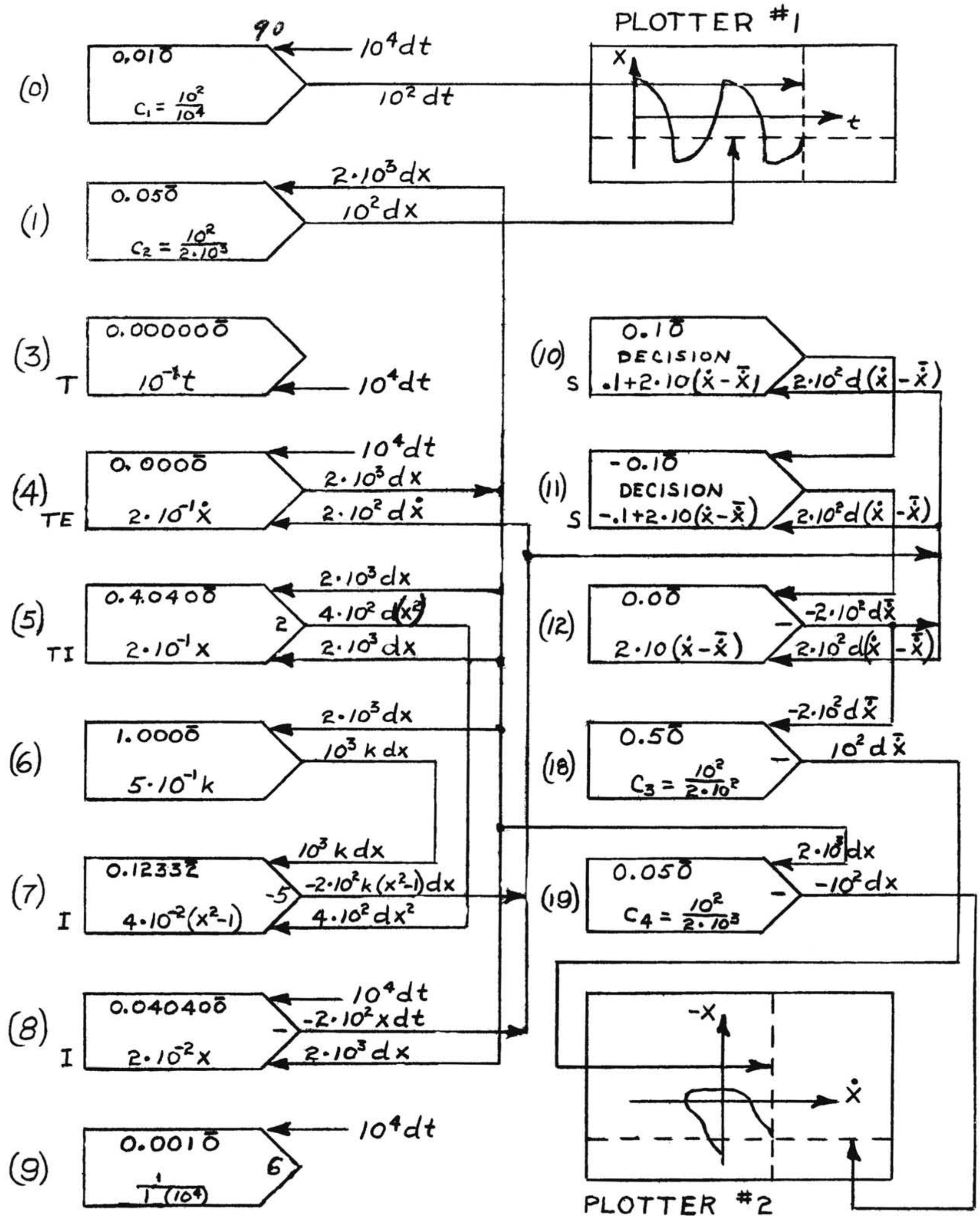


Figure 10.15: DDA setup for solving $\ddot{x} + k(x^2 + 1)\dot{x} + x = 0$ (see [Bendix 1954][p. 65])

```
         11111111112222222222333333333344444444445
12345678901234567890123456789012345678901234567890
--------------------------------------------------
 00  0010      11111 090
 01  0050      11111 04
 03  0000000   11211       90
 04  00000     31211 90    07 08
 05  040400    21212 04    04
 06  10000     11111 04
 07  012332    211-5 06    05
 08  0040400   211-1 90    05
 09  00010     11116 90
 10  010       91111       07 08 12
 11 -000       91111 10    07 08 12
 12  000       111-1 11    07 08 12
 18  050       111-1 12
 19  0050      111-1 04
```

Figure 10.16: D-12 example program (cf. [Bendix 1954][p. 69])

Column 17: The possible values of "1" or "2" select the reset mode of the integrator (normal vs. automatic).

Column 18: Integrators flagged with a value of "1" in this column are running in normal mode while a value of "2" causes the contents of the second accumulator register to be printed on the console printer after each iteration.

Column 19: This can be used to select an automatic sign-reversal for the incremental output signal of an integrator. "1" specifies normal operation while "-" causes a change of sign.

Column 20: Using this column a constant output multiplication factor of 1, 2 or 5 can be selected. The special value "6" selects the corresponding integrator as control circuit for the typewriter.

Columns 22 and 23 specify the source of the dX input while columns 25 and 26 select the source of the initial value of the integrator. Up to 8 addresses of source elements for the dY inputs can be selected by values in the columns starting at 28.[25]

Similar sequential incremental DDAs are the *UNIVAC incremental computer*, see [Remington 1956], the system developed for the *Operational Flight Trainer* described by [Gray 1958], and *STARDAC*, see [Milan-Kamski 1969], to name just a few.

10.4.3 TRICE

A radically different implementation of a DDA was developed by *Packard Bell* in 1958. In contrast to the systems described above which operate purely sequential, this machine, named *TRICE*, was the first parallel DDA. This required a fully transistorized

[25] Each address occupies two columns and successive input addresses are separated by a space character.

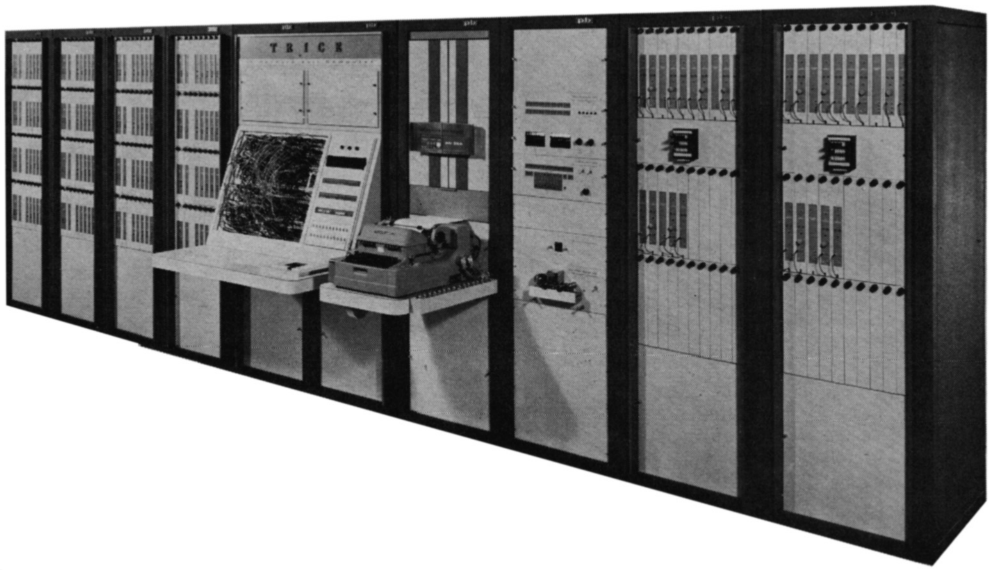


Figure 10.17: Fully expanded TRICE system (cf. [Ameling 1963/2][p. 30])

implementation to allow sufficiently dense packing of the computing elements and minimize waste heat to build a practical system.[26]

Figure 10.17 shows a fully expanded TRICE system. From left to right it consists of four racks housing computing elements, a wide rack containing the central patch panel as well as the operator console, a rack with a bit-serial stored-program digital computer *PB-250* which controls the overall system, loads initial values into integrators, etc. Next to this is a rack containing a paper-tape reader/puncher and various ADCs and DACs which allow coupling TRICE to an analog electronic analog computer or to other analog signal sources. The two racks on the right contain addition computing elements.

TRICE operates at a clock rate of 3 MHz which is quite remarkable for such a large and early transistorized system. Its word length is 30 bit[27] while values are represented in binary and not in decimal form as in the D-12, which results in a maximum iteration frequency of 10^5 s^{-1}. Due to the completely parallel operation of TRICE, repetitive operation is possible with a frequency of up to 100 Hz thus challenging traditional analog computers. TRICE has been described as the *"most advanced DDA that was ever built."*[28]

TRICE features summers, integrators with only one $\mathrm{d}X$ and $\mathrm{d}Y$ input,[29] special summing-integrators, constant value multipliers, variable multipliers, servos, and

[26]See [Ameling 1963/2], [Rechberger 1959], and [Packard Bell].

[27]The internal registers of the basic computing elements are implemented as magnetostrictive delay lines, so word length and clock frequency are fixed.

[28]See [Giloi 1975][p. 23].

[29]If more inputs are required a summer is also required.

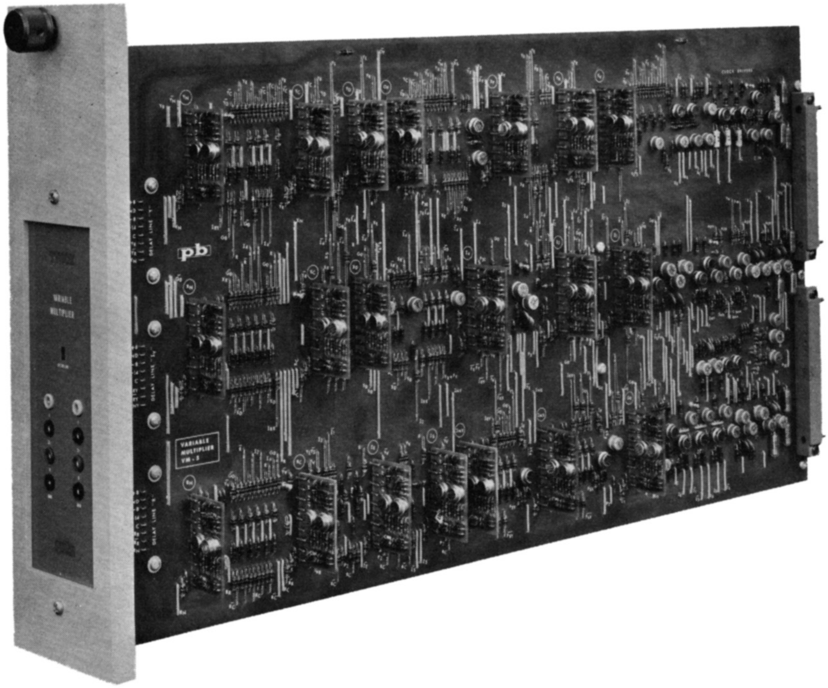


Figure 10.18: TRICE multiplier (cf. [Ameling 1963/2][p. 30])

a variety of ADCs and DACs. A typical TRICE module, a multiplier, is shown in figure 10.18. This multiplier is based on two dedicated integrators implementing the product rule technique described earlier.[30]

A typical application of TRICE is described in [Ameling 1963/2][pp. 40 f.]: A complex flight-dynamics problem is solved on a TRICE system coupled to a large traditional analog computer. While TRICE performs those parts of the simulation that require utmost precision like trajectory calculations and the like, the analog computer implements operations that require extreme high speed but won't introduce errors into integrations like coordinate transformations etc. The TRICE system used in this simulation contained 42 integrators, 11 multipliers, 5 constant value multipliers, 5 servos, 6 ADCs and 13 DACs, while the analog computer featured 200 operational amplifiers, 14 multipliers, 5 function generators and 20 servos for coordinate transformation and rotation.

Due to the high amount of hardware necessary to implement a system like TRICE, only a couple of these were eventually built. According to Tom Graves there were three TRICE installations at the flight simulation laboratory at NASA in Houston. Thus TRICE failed due to its complexity while simpler systems as the D-12 and others eventually failed due to their inherent slow speed.

[30] See [Jackson 1960][pp. 585 f.].

11 Applications

The following sections give an overview of typical analog computer applications. Due to the constrained space and the multitude and complexity of areas where analog computers were used, these descriptions will be quite cursorily. As Ernst Kettel put it, there is next to no scientific field where an analog computer could not be applied successfully.[1]

11.1 Mathematics

Although analog computers do not excel when it comes to accuracy, there were quite some applications in mathematics which formed the basis for many following applications in the fields of engineering and applied sciences.

11.1.1 Differential equations

One of the main areas of application for analog computers in mathematics was to investigate the behavior and properties of differential equations of various types. The easiest case of so-called *linear ordinary differential equations*[2] poses no special problems and can be handled with the Kelvin feedback technique or the substitution method described in sections 7.2 and 7.3. The equations of interest are of the general form

$$\sum_{i=0}^{n} a_i \frac{\mathrm{d}^i y}{\mathrm{d}y^i} = f(t).$$

If the coefficients a_i are constant, the solution of such equations and systems thereof with an analog computer is straightforward. If the a_i are variable, the additional problem of generating these coefficients during a simulation run arises. In most cases the a_i will be generated as solutions of auxiliary differential equations. Normally these auxiliary equations are of the LODE type. The scaling process often becomes extremely difficult for the case of variable coefficients.

Non-linear differential equations are equally challenging with respect to the scaling process. The fact that the structure of the computer setup often depends heavily on the selected scaling factors complicates their solution further. A wealth of information about these problems can be found in [Giloi et al. 1963][pp. 221 ff.], [Mackay 1962][pp. 212 ff.]. The application of hybrid computers in this area is covered in [Valisalo et al.].

[1] Cf. [Kettel 1960][p. 165].

[2] *LODE* for short.

So-called *boundary value problems* where boundary values for a differential equation at $t = 0$ and $t = t_{\max}$ are given, can often be solved rather easily with an analog computer[3] operating in repetitive mode by applying a trial-and-error procedure.[4] Starting with the boundary value at $t = 0$ the equations are solved repeatedly while a human operator or the stored-program computer of a hybrid system[5] varies the parameters until the second boundary value condition is satisfied.

The treatment of partial differential equations has already been described cursorily in section 7.4. More information can be found in [Heinhold et al.][pp. 183–197], [Mackay 1962][pp. 243 ff., pp. 293 ff.], and [Howe 1962]. An interesting method for generating a flicker-free display of the solutions of partial differential equations on an oscilloscope is described in [Ameling 1962/1], while [Feilmeier 1974][pp. 215 ff.] deals with the application of hybrid computers for solving this class of equations. Another interesting technique, involving a magnetic tape for storing analog data, developed by Lawrence Wainwright, is described in [Jackson 1960][pp. 586 f.].

11.1.2 Integral equations

An analog computer is not particularly well suited for treating integral equations. Due to their importance in fields like electrical and communications engineering and the like, some attempts were made to solve this class of equations with an analog computer. Integral equations are of the form

$$f(x) + \int_0^a K(t,x)u(t)\,\mathrm{d}t = 0 \quad \text{or} \tag{11.1}$$

$$f(x) + \int_0^a K(t,x)u(t)\,\mathrm{d}t = u(x). \tag{11.2}$$

These equations are called Fredholm[6] equations of type 1 or type 2, respectively. $u(x)$ is the unknown, while $f(x)$ and $K(t,x)$, the so-called *kernel*, are known. The representation of this kernel is the main problem on an analog computer since functions of more than one variable are rather difficult to implement. A hybrid computer can simplify things considerably. In cases where no hybrid computer is available, x is often restricted to one value so that a simple function generator can be used to implement $K(t,\bar{x})$, where $\bar{x}$ denotes the restricted x.

Using proper time-scaling, the upper boundary of the integral in equation (11.1) or (11.2) can be identified with the computer time τ. The result of such a computer run is

[3]See [Korn & Korn 1956][pp. 142 ff.] for example.

[4]See [Giloi et al. 1963][pp. 249 ff.] or [Heinhold et al.][pp. 108 ff.].

[5]See [Heinhold et al.][pp. 111 ff.] and [Feilmeier 1974][pp. 202 ff.]. More about iterative approaches can be found in [Heinhold et al.][pp. 117 ff.], [Feilmeier 1974][pp. 197 ff.], and [Giloi et al. 1963][pp. 253 ff.]. Special approximation techniques for solving boundary value problems for partial differential equations are described in [Vichnevetsky 1969].

[6]Ivar Fredholm, 04/07/1866–08/17/1927

then compared with either 0 or $u(x)$, depending on the type of equation to be solved. Based on the deviation from the desired result, the parameters defining $u(x)$ are varied manually or under program control. So basically solving integral equations with an analog or hybrid computer is a trial-and-error technique.

Restricting x to a single value is often not sufficient – in these cases x is discretized appropriately, where one function generator is necessary for every discrete value of x to implement $K(t,x)$. Using an analog computer in repetitive operation and electronic switches to select one from the various function generators, it is possible to display an array of curves on an oscilloscope.[7] More information about solving integral equations with analog and hybrid computers can be found in [Sydow 1964][pp. 112 ff.], [Mackay 1962][pp. 317 ff., pp. 331 ff., pp. 352 ff.], and [Chan 1969].

11.1.3 Zeros of polynomials

Determining the zeros of polynomials like

$$P_n(x) = \sum_{i=0}^{n} a_i x^i$$

with an analog computer is a rather straightforward process:[8] The polynomial $P_n(x)$ is generated either by explicit multiplication to generate the x^i, or by repeated integration yielding x, $-\frac{1}{2}x^2$, $\frac{1}{6}x^3$ etc. Since typical analog computers have more integrators than multipliers and since multiplication is not as precise an operation as integration, the latter approach is preferred.[9]

Another approach is to represent the polynomials by differential equations, which is especially useful in the case of Legendre-, Chebyshev- or Hermite-polynomials.[10] In general a polynomial of n-th degree can be represented by a differential equation of n-th degree as the following example[11] shows. The polynomial

$$P(x) = a_3x^3 + a_2x^2 + a_1x + a_0$$

has

$$\begin{aligned} P'(x) &= 3a_3x^2 + 2a_2x + a_1 \\ P''(x) &= 6a_3x + 2a_2 \\ P'''(x) &= 6a_3 \end{aligned}$$

as its first, second and third derivative. So a chain of three integrators with $6a_3$ as input can be used to yield the values $2a_2$, $-a_1$ and a_0 to generate the polynomial.

[7] This requires at least three independently controlled integrator groups which in turn makes a rather complex control circuit necessary.

[8] See [Heinhold et al.][pp. 173 ff.].

[9] If servo or time division multipliers are available, the powers of x can be generated in a straightforward way.

[10] See [Heinhold et al.][pp. 174].

[11] Cf. [Heinhold et al.][pp. 174 f.].

Given a computer setup to generate the required polynomial, the analog computer will then vary x through the value domain of interest. Using a comparator circuit, the zeros of the polynomial can be detected and the machine can be placed into hold mode where the current value x can be read out.

11.1.4 Orthogonal functions

A task often required in control engineering applications and other fields is the approximation of a function $f(t)$ as a sum of orthogonal functions $\varphi_i(t)$:[12]

$$f(t) \sim \sum_{i=1}^{n} a_i \varphi_i(t)$$

Normally, the error square term

$$\int_I \left(f(t) - \sum_{i=1}^{n} a_i \varphi_i(t) \right)^2 w(t) \mathrm{d}t$$

is to be minimized over an interval I, where $w(t)$ denotes a weighting function. This is satisfied if the $\varphi_i(t)$ form an orthogonal system and the coefficients a_i are defined by

$$a_i = \frac{1}{c} \int_I f(t) \varphi_i(t) \mathrm{d}t \tag{11.3}$$

where

$$\int_I \varphi_i(t) \varphi_k(t) \mathrm{d}t = \begin{cases} c \neq 0 & \text{if } i = k \\ 0 & \text{if } i \neq k \end{cases}$$

holds. Using an analog computer, the coefficients a_i can be determined based on (11.3).

11.1.5 Linear algebra

Solving systems of linear equations is a task particularly well suited for stored-program computers and rather ill-suited for an analog computer. Nevertheless, methods were developed to solve such equation systems on analog computers as well, which were applied in cases where no stored-program computer was available. These techniques are of no high value today but of historical interest.

A direct approach using a method like Gauss elimination is unsuited for an analog computer since so-called *algebraic loops*[13] will result in the computer setup. Other

[12] See [Herschel 1962].

[13] An algebraic loop is a positive feedback path involving only summers, i. e. a loop involving an odd number of sign-inverting summers. Such a circuit is inherently unstable and is to be avoided wherever possible.

techniques like the Jacobi method, the Gauss-Seidel method or relaxation approaches are better suited for an analog computer.[14]

As an example a system of linear equations

$$A\vec{x} = \vec{b}$$

is to be solved by the Jacobi method. First $\vec{x} = B\vec{x} + \vec{k}$ is computed, where B represents the Jacobi matrix corresponding to A which yields

$$B = -\begin{pmatrix} 0 & \frac{a_{12}}{a_{11}} & \dots & \frac{a_{1n}}{a_{11}} \\ \frac{a_{21}}{a_{22}} & 0 & \dots & \frac{a_{2n}}{a_{22}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{a_{n1}}{a_{nn}} & \frac{a_{n2}}{a_{nn}} & \dots & 0 \end{pmatrix} \quad \text{and} \quad \vec{k} = \begin{pmatrix} \frac{b_1}{a_{11}} \\ \vdots \\ \frac{b_n}{a_{nn}} \end{pmatrix}.$$

Normally, B and $\vec{k}$ are determined once, aided by a stored-program digital computer. Based on these, iteration steps

$$\vec{x}_{n+1} = B\vec{x}_n + \vec{k}$$

can be performed on the analog computer,[15] which requires two integrators for each element of the vector $\vec{x}$ which are used as storage elements. This requires not only a high amount of computing elements but also a complex control setup which makes this approach rather costly. Based on the same idea, but starting with an identity matrix, matrix inversions can also be handled with an analog computer.

Another approach is to reduce the problem of solving a system of linear equations to solving a corresponding system of differential equations

$$\dot{\vec{x}} = B\vec{x} \tag{11.4}$$

which is treated in detail in [Heinhold et al.][pp. 148 ff.]. The solution of (11.4) converges to the desired solution of the original system of linear equations for $t \to \infty$.

11.1.6 Eigenvalues and -vectors

Determining eigenvalues and eigenvectors for a given matrix $A = (a_{ij}), i, j = 1, \dots, n$ of n-th degree, is a common task and can be reduced to solving a system of linear equations of the form

$$(A - \lambda_i E)\vec{x}_i = 0. \tag{11.5}$$

[14]More detailed information about these methods can be found in [Heinhold et al.][pp. 135–152], [Vocolides 1960], [Giloi et al. 1963][pp. 152 ff.], [Marquitz et al. 1968], [Kovach et al. 1962], [Jackson 1960][pp. 332 ff.], and [Mackay 1962][pp. 192 ff.]. The effects of parameter variations is analyzed in [Kahne 1968].

[15]The convergence criterion regarding the spectral radius must be satisfied to apply this iteration process.

In the case of complex valued eigenvalues, a system of equations of degree $2n$ is required. In either case, the eigenvalues λ_i correspond to the roots of the characteristic polynomial $\det[A - \lambda \mathrm{I}]$ where I denotes the identity matrix.[16]

Although this problem can be readily solved by applying the methods described above, more often the von Mises iteration is used as described in [Popović 1964]. This method works directly on the elements on A and does not require an explicit characteristic polynomial to be derived.

11.1.7 Fourier synthesis and analysis

While transformations from frequency domain into time domain and vice versa are performed nowadays on stored-program digital computers employing so-called *Fast* Fourier *Transformations*[17] or *Wavelet Transforms* neither these methods nor adequately fast stored-program computers were available until the early 1970s. Therefore the decomposition of a signal into its components, specified by frequency, phase and amplitude, was done regularly with analog computers.[18] The basic idea of such a Fourier[19] is that every periodic function $f(x)$ with a period length of l can be represented by

$$f(x) = \frac{a_0}{2} + \sum_{k=1}^{\infty} \left(a_k \cos\left(\frac{2\pi k x}{l}\right) + b_k \sin\left(\frac{2\pi k x}{l}\right)\right) \tag{11.6}$$

under certain circumstances, where the coefficients a_k and b_k are defined by

$$a_k = \frac{2}{l} \int_0^l f(x) \cos\left(\frac{2k\pi x}{l}\right) \mathrm{d}x \quad \text{and} \tag{11.7}$$

$$b_k = \frac{2}{l} \int_0^l f(x) \sin\left(\frac{2k\pi x}{l}\right) \mathrm{d}x. \tag{11.8}$$

In the case of a so-called Fourier *synthesis*, this corresponds to the task of a harmonic synthesizer as described in section 2.6. To generate the sum (11.6), the required sine-/cosine-terms are implemented by analog computer setups as shown in figure 8.2 in section 8.1.

The inverse case of a Fourier *analysis* is more complicated since the coefficients a_k and b_k must be determined, based on a given time varying function $f(t)$. The direct approach is to evaluate the integrals (11.7) and (11.8) explicitly which requires a very exact time base since this determines the bounds of the integration operations.[20]

---

[16] Cf. [Press et al. 2001][pp. 368 ff.].
[17] *FFT* for short.
[18] See [Dick et al. 1967], [Giloi et al. 1963][pp. 306 ff.], [Kovach 1952], and [Ratz 1967].
[19] Fourier, Jean-Baptiste-Joseph, 03/21/1768–05/16/1830
[20] Cf. [Giloi et al. 1963][pp. 307 f.].

Another approach uses a low-pass filter, consisting of a single integrator with negative feedback in the simplest case, to determine the coefficients for a given $f(t)$ which is described in [Giloi et al. 1963][pp. 308 ff.].

Yet another method is based on exploiting a resonance effect in a circuit like that shown in figure 8.2. By means of its time scale parameter αk_0 this circuit is tuned to a specific frequency. The leftmost integrator will be fed with the signal $f(t)$ in addition to the feedback signal from the resonant circuit itself. The coefficients a_k and b_k can then be determined based on the amplitudes of the sine- and cosine-terms generated by this resonator.

11.1.8 Random processes and Monte-Carlo simulations

The study of random processes was an important application area for analog computers since many commercially interesting processes, ranging from chemical technology to the behavior of flight vehicles, nuclear reactors, the creation of steam bubbles in coolant loops etc., are disturbed by external random events.[21]

The analog computer is normally used to model the randomly disturbed system to be analyzed, while the random source is either a true random noise generator as described in section 4.12 or a pseudo-random generator based on a shift-register with multiple feedback lines. This latter signal source can be readily implemented using the digital elements most medium to large analog computers have. The advantage of a pseudo-random source is the repeatability of simulation runs. Its disadvantage is that determining and guaranteeing the required degree of randomness can be quite challenging, depending on the sensitivity of the underlying system which is to be analyzed.[22]

An interesting technique is to identify a partial differential equation with a stochastic model in a way that the parameters of the latter correspond to the solution of the former at a certain location. This allows the application of so-called *Monte-Carlo methods* to a wide class of problems. Most often the necessary stochastic parameters are obtained with the digital portion of a hybrid computer while the analog computer evaluates the model.[23]

11.1.9 Optimization

Optimization problems are of high commercial importance and were typical areas of application for analog computers. In general, a linear optimization problem can be represented by a real-valued matrix $A \in \mathbb{R}^{n,m}$ with $m < n$ and two vectors $\vec{b} \in \mathbb{R}^m$ and $\vec{x} \in \mathbb{R}^n$ satisfying the condition

$$A\vec{x} = \vec{b}. \tag{11.9}$$

[21] Cf. [Giloi et al. 1963][pp. 357 ff.] and [Jackson 1960][p. 367].

[22] More information on this range of topics can be found in [Giloi et al. 1963][pp. 358 ff.], [Meyer-Brötz 1962], [Ott 1964] (computation of correlation coefficients), [Sydow 1964][pp. 250 ff.] (quality control), [Bohling 1970], [Rideout 1962], [Korn & Korn 1956][pp. 140 ff.], and [Feilmeier 1974][pp. 275 ff.].

[23] See [Rideout 1962] and [Feilmeier 1974][pp. 238 ff.].

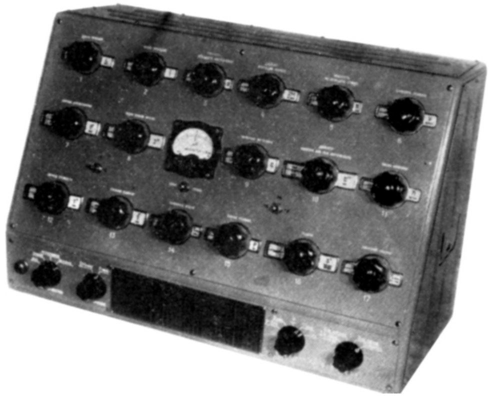


Figure 11.1: BPRR-2 computer used for the optimization of industrial processes (see [Ushakov 1958/2][p. 1961])

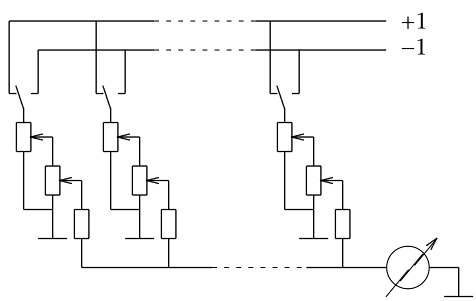


Figure 11.2: Basic circuit of the BPRR-2 (see [Ushakov 1958/2][p. 1961])

The goal of the optimization process is to determine a vector $\vec{x}$ that additionally maximizes the scalar product $\vec{c}^{\mathrm{T}}\vec{x}$ where $\vec{c} \in \mathbb{R}^n$ represents the so-called *value coefficients*.[24]

Basic techniques for tackling this class of problems with analog computers are described in [Korn & Korn 1956][pp. 147 ff.] and [Kovach et al. 1962]. In the early days of analog computer often direct analogies based on conductive sheets which were sampled by a stylus driven with an xy-recorder were employed.[25] Another example for such a direct analog computer is shown in figure 11.1. Its basic circuit, which is shown in figure 11.2, is quite simple: Each parameter of the optimization problem is represented by two potentiometers of which the first one represents the parameter itself which will be varied manually during the optimization process, while the second potentiometer is used to set a fixed weighting factor describing the underlying process. The currents flowing through these potentiometer groups are then summed and displayed on a meter.

Other approaches to optimization with analog computers are based on Monte-Carlo techniques as described above, or implement gradient methods like *continuous steepest ascent/descent*[26] Especially these gradient methods are quite well suited for hybrid computers where the stored-program digital processor controls the ascent/descent-process.[27]

11.1.10 Multidimensional shapes

A fascinating application of analog computers is the display of multidimensional shapes on an oscilloscope. A simple example for such a display has been described in

[24]See [Pierre 1986][pp. 193 ff.].
[25]Cf. [Pierre 1986][pp. 251 ff.].
[26]See [Pierre 1986][pp. 296 ff.], [Levine 1964][pp. 217 ff.], and [Albrecht et al.].
[27]Cf. [Feilmeier 1974][pp. 243 ff.].

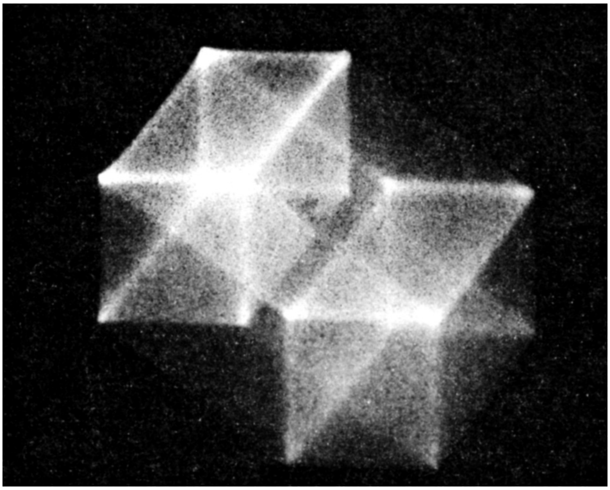


Figure 11.3: Display of a four-dimensional hyper-cube on an oscilloscope (see [Mackay 1962][p. 137])

section 8.8. A far more complex example is shown in figure 11.3 – here the projection of a four-dimensional hyper-cube is generated by a high-speed analog computer.[28] In addition to that, even stereographic projections were implemented with analog computers.[29]

11.2 Physics

Obviously, an analog computer is quite well suited for applications in physics due to its underlying principle of operation. The following sections describe some typical examples of use.

11.2.1 Orbit calculation

Analog computers are quite well suited for solving many-body problems arising in astronomy and space flight. Especially rendezvous simulations were of prime importance during the early days of space flight. A simple example describing the computation of a planet's orbit can be found in [Telefunken 1966].

11.2.2 Particle trajectories

Two typical applications from nuclear physics are the calculation of particle trajectories in a magnetic field and beam control in particle accelerators.

[28]See [Telefunken/3], [Lukes 1967], and [Mackay 1962][pp. 129 ff.] for general information on this topic.
[29][Mackay 1962][pp. 134 ff.].

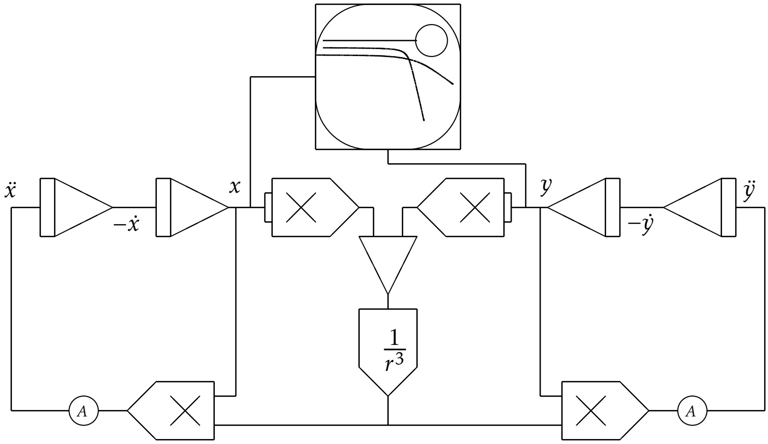


Figure 11.4: Basic circuit for simulating an alpha particle trajectory ([Telefunken/5][p. 2])

[Telefunken/5] describes the simulation of the historic Rutherford experiment – alpha particles being scattered on a thin gold foil – on an analog computer. Basically, an alpha particle approaching an atomic nucleus is subject to a force

$$F = \frac{1}{4\pi\varepsilon_0}\frac{q_\alpha q_k}{r^2},$$

where q_α and q_k denote the charges of the alpha particle and that of the target nucleus. Simplifying this setup by reducing it to a two-dimensional model yields the two components

$$\ddot{x} = A\frac{x}{r^3} \quad \text{and} \tag{11.10}$$

$$\ddot{y} = A\frac{y}{r^3} \tag{11.11}$$

with a suitably chosen constant of proportionality A. Integrating twice over $\ddot{x}$ and $\ddot{y}$ respectively yields a coordinate pair (x,y) describing the position of the alpha particle. The basic circuit for a simulation based on (11.10) and (11.11) is shown in figure 11.4.[30] A noteworthy aspect is that no division by r^3 is performed – instead a function generator, which has been setup to generate $1/r^3$, is used.

The variable parameter of this simulation is the initial condition input of the second integrator from the right that yields y.[31] Using this input, the height at which the particle is injected can be set. Using a simple digital circuit controlling an additional integrator, this initial condition can be stepped automatically to simulate many trajectories starting at various heights which is adumbrated in the screen depicted in figure 11.4. Figure 11.5 shows the overall computer setup for this simulation. On the right hand side, a Telefunken RA 741 tabletop analog computer can be seen while the left

[30] A more detailed circuit can be found in [Telefunken/5][p. 3].

[31] This input is not shown explicitly in figure 11.4.

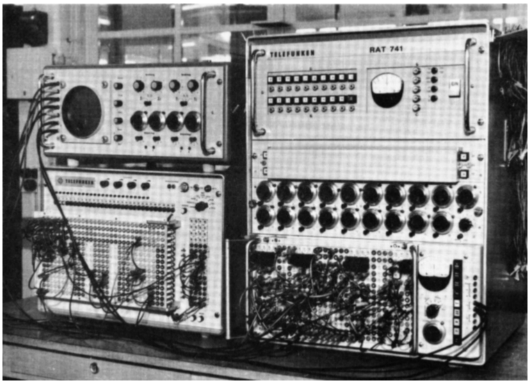


Figure 11.5: Setup of the particle trajectory simulation (see [Telefunken/5][p. 1]

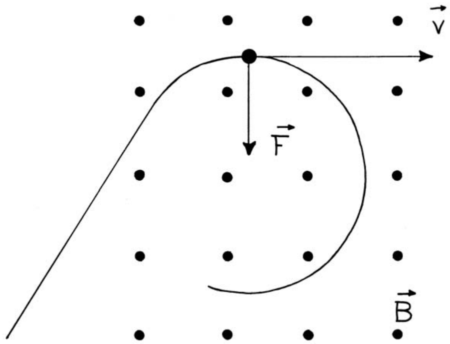


Figure 11.6: Trajectory of a charged particle in a magnetic field (see [Borchardt][p. 1])

half is occupied by a so-called *DEX 100* digital expansion unit. On top of this a two channel oscilloscope of type *OMS 811* can be seen on which the particle trajectories are displayed.

The next example computes the trajectories of charged particles in a magnetic field – a task typical for the development of beam control systems. Simulations like this were performed at *DESY*[32] first using a large EAI 231RV analog computer and later a Telefunken hybrid computer based on the RA 770 precision analog computer.[33]

Such a charged particle moving in a magnetic field is subject to the Lorentz force[34]

$$\vec{F}_\text{L} = q(\vec{v} \times \vec{B})$$

where q represents the particle's charge, $\vec{v}$ its velocity, and $\vec{B}$ the magnetic flux density. Since $\vec{v}$ is constant according to amount, $\vec{F}_\text{L}$ results in a change of direction, so

$$\vec{F}_\text{L} = m\ddot{\vec{r}}$$

holds. This effect is shown schematically in figure 11.6 where a particle is injected diagonally into a field of constant magnetic flux. Figure 11.7 shows the analog computer setup for computing trajectories like this. A typical simulation output is shown in figure 11.8. Using a digital control system or the stored-program computer of a hybrid computer, various trajectories, corresponding to different particle injection speeds $\dot{x}$, can be generated.

11.2.3 Optics

A interesting optics application and its solution by means of an analog computer is described in [Rudnicki]. There the simulation of a laser interferometer based on a

[32] Short for *Deutsches Elektronensynchrotron*.

[33] See [Borchardt et al.], [Borchardt 1965], [Borchardt 1966], and [Borchardt et al. 1965] for more information about these simulations and the techniques developed.

[34] See [Borchardt].

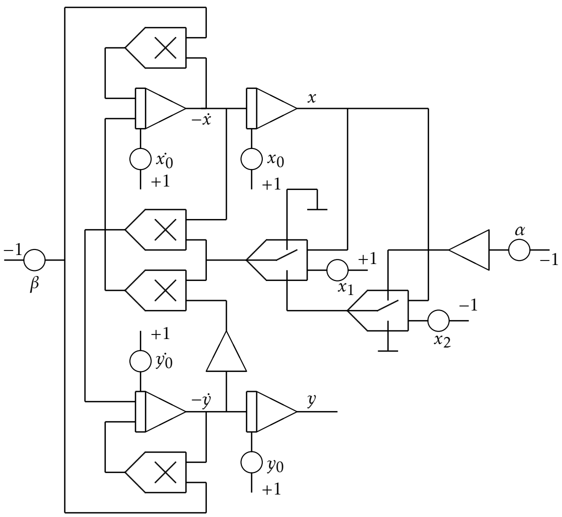


Figure 11.7: Computer setup for simulating the trajectory of a charged particle in a magnetic field (see [Borchardt][p. 6])

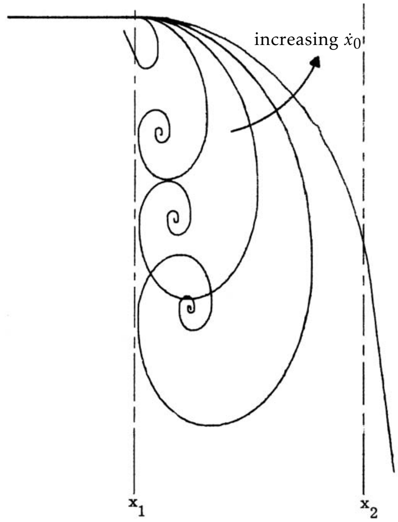


Figure 11.8: Various trajectories of charged particles in a magnetic field (see [Borchardt][p. 4])

modulated HeNe gas laser is treated. Using two basic oscillator setups like the one shown in figure 8.2 the laser beam as well as the modulation oscillator are modeled, thus simulating the interferometer on a wave shape basis.

11.2.4 Heat-transfer

The solution of heat-transfer problems invariably involves partial differential equations which require a rather high amount of computing elements, depending on the degree of discretization as shown in section 7.4.1. A practical application example for a complex heat-transfer simulation is given in [James et al. 1971][pp. 193 ff.] where the design of heat sinks for radioisotope thermoelectric generators for use in unmanned spacecraft is described. The temperature on one side of the heat sink is determined solely by the decay heat of the radioisotopes, the variable parameter of the optimization problem is the thickness of the heat sink. The simulation is implemented in such a way that the computer time τ is associated with the position x on a heat sink cross section.

The simulation of the radial heat distribution in a loaded power cable is covered in [Telefunken 1963/1] using a quotient of differences approach as described in section 7.4.1 to model the cable's geometry. [Valisalo et al. 1982] describes the heat distribution analysis in irregularly shaped two-dimensional areas using a Monte-Carlo approach implemented on a hybrid computer.

Two special purpose direct analog computers for the analysis of heat-transfer problems are shown in figures 11.9 and 11.10. The former shows the *Electronic Analog Frost Computor*, *EAFCOM* for short, that was used by the US Army Corps of Engineers

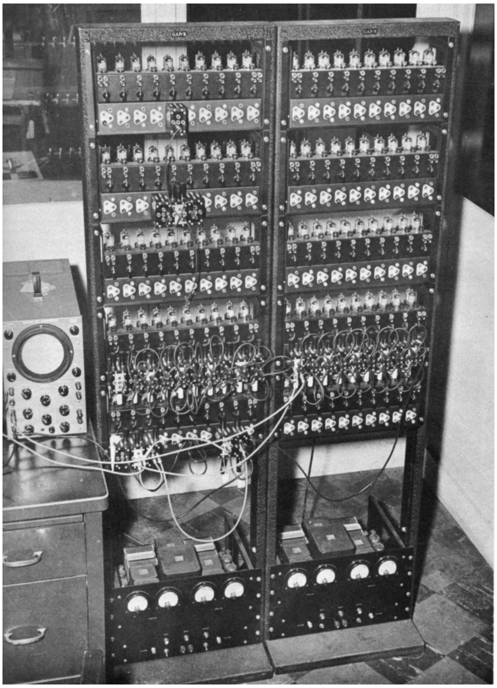


Figure 11.9: The *Electronic Analog Frost Computor*, US Army Corps of Engineers, New England Division (see [Aldrich et al. 1955][p. 259])

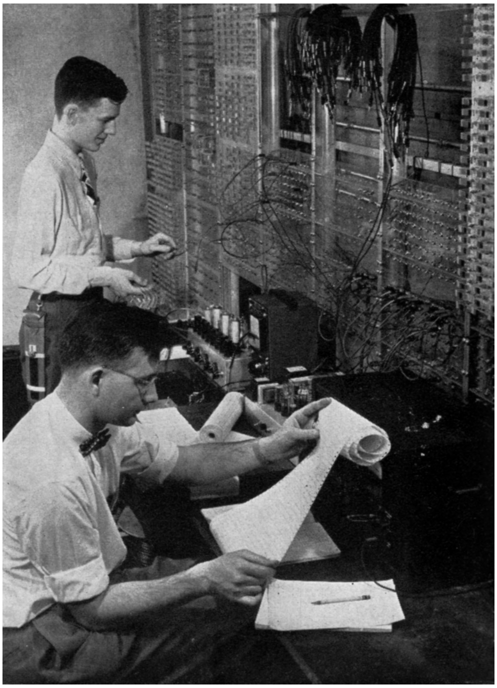


Figure 11.10: The *Heat Exchange Transient Analog Computer*, *HETAC* for short (see [N. N. 1957/5])

to simulate soil freezing and associated effects.[35] This system is a hardware implementation of a two-dimensional quotient of differences.[36] This two-dimensional grid of computing elements contains four different types of building blocks: *A*-elements are simple adders accepting values from their neighbor elements, *C*-elements are basically coefficient potentiometers, *J*-elements represent integrators, and *Z*-elements, which are used to model inert areas like heat sources or sinks.

A sub-circuit, consisting of one A-, J-, Z-, and two C-elements, forms a macro-element representing one layer of soil. These macro-elements are then interconnected with their direct neighbors as well as with more remote elements. Such connections may represent thermal bridges or water flowing between layers.[37]

The latter system shown is the *Heat Exchange Transient Analog Computer*, *HETAC* for short, which was developed at the *University of Virginia* HETAC, too, is a two-dimensional representation of a quotient of differences approach for solving the heat-transfer equation.

[35] See [Aldrich et al. 1955].

[36] In today's notion this machine could be described readily as a two-dimensional automaton with analog representation of its cell states.

[37] A detailed description of these direct analogies and their application to frost and thaw problems can be found in [Paynter].

[Kerr 1978] and [Kerr 1980] describe analog computer models for the analysis of convection currents developing on inclined heat-exchanger surfaces. Using a rather small EAI 380 analog computer solutions for this problem could be generated at a frequency of 500 Hz which allowed to display the solution curves on an oscilloscope while a human operator could change the model-parameters interactively and watch the changing solutions.[38]

A zone melting simulation is described in [Carlson et al. 1967][pp. 318 ff.]. Research like this was essential for the rapid development of semiconductor technology which required high-purity materials like germanium or silicon. These raw materials are usually purified using a zone melting technique. This purification process depends on a number of parameters like the diameter of the germanium or silicon mono-crystal, the width of the melting zone, the speed at which this zone moves through the crystal etc. These parameters were determined using an analog computer simulation.

11.2.5 Semiconductor research

Analog computers were also employed in semiconductor research. [Apalovičová 1979] describes the simulation of the electric field in the depletion zone of a MOS[39] field effect transistor. The target of this research was to predict the behavior of different transistor structures. The solution of the resulting elliptic partial differential equations was done by a hybrid computer.[40]

The analysis of the dynamic response of a so-called *tunnel diode*[41] is described in detail in [Carlson et al. 1967][pp. 332 ff.]. It is noteworthy that the program derived in this work contains an algebraic loop, which is stable due to the rather unique combination of coefficients.[42] Furthermore this is a good example of a problem that is too fast to be analyzed on a real specimen, so an extremely slowed down simulation on an analog computer is necessary to gain the necessary insight.

11.2.6 Ferromagnetic films

Ferromagnetic films were of high interest in the 1960s as storage elements for stored-program digital computers. Accordingly there was some research regarding the simulation of the behavior of such films under various environmental conditions. [Borsei et al.] describe an analog computer approach to simulate the behavior of a ferromagnetic film which is subject to mechanical stress.

[38] Generating solutions at such high speed is still challenging on today's stored-program digital computers.

[39] Short for *Metal Oxide Semiconductor*.

[40] Numerical methods for the simulation of semiconductor junctions and field effect transistors can be found in [Akers 1977].

[41] Also known as Esaki *diode*.

[42] Normally algebraic loops are to be avoided at all costs.

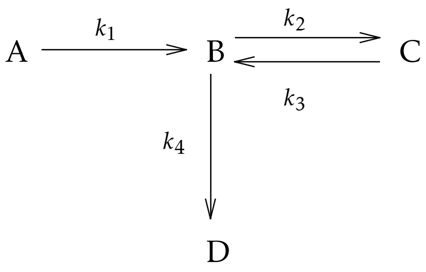


Figure 11.11: Simple reaction kinetics example involving four substances (see [Dornier/2][p. 3])

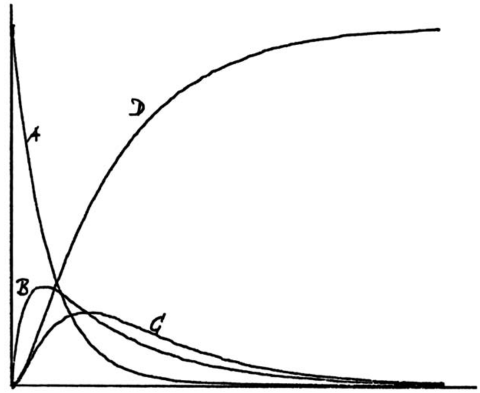


Figure 11.12: Reaction kinetics simulation result (see [Dornier/2][p. 7])

11.3 Chemistry

The following sections describe two typical applications of analog computers in chemistry. For applications in chemical engineering refer to section 11.13.

11.3.1 Reaction kinetics

Most chemical processes involve many reaction steps which are heavily interdependent. Studies in reaction kinetics analyzing such processes have been performed largely on analog computers since the underlying differential equations are well suited for mechanization. A typical example problem is shown in figure 11.11.[43] A substance A is to be converted into a product D which involves an intermediate substance B which is partially transformed into C and vice versa. The rates at which these steps happen are k_1, k_2, k_3, and k_4.

Based on figure 11.11 the behavior of this chemical reaction system can be described by the following four coupled differential equations:

$$\begin{aligned}
\dot{A} &= -k_1 A \\
\dot{B} &= k_1 A - k_2 B - k_4 B + k_3 C \\
\dot{C} &= k_2 B - k_3 C \\
\dot{D} &= k_4 B
\end{aligned}$$

The result of a typical simulation run based on these equations with $k_1 = k_2 = k_3 = k_4 =$ const. is shown in figure 11.12. Starting with 100% of component A the four reactions set in, finally yielding a pure end product D. The simulation program is shown in figure 11.13.

Further examples can be found in [Cheng] where the simulation of polymerization processes using analog computers is described. [Bassano et al. 1976] derive

[43] See [Dornier/2].

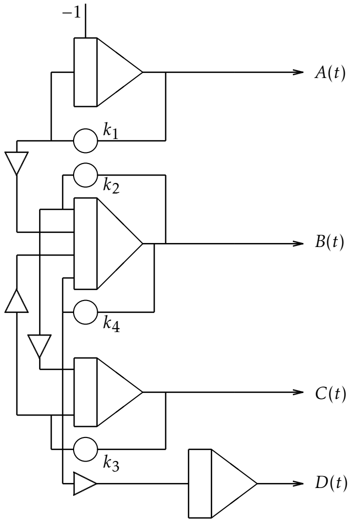


Figure 11.13: Reaction kinetics simulation program (see [Dornier/2][p. 4])

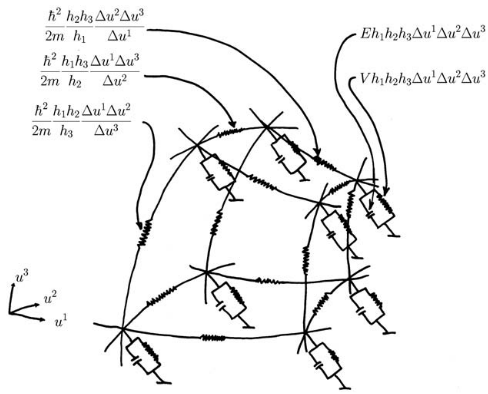


Figure 11.14: Analog model of Schrödinger's equation with three independent variables (see [Kron 1945/1][Fig. 7])

techniques to determine process parameters for the chlorination of methane, and [Ramirez 1976][pp. 74 ff.] describes the simulation of a large-scale benzene chlorination.

11.3.2 Quantum chemistry

Direct analog computers proved very useful in the early days of quantum chemistry,[44] where solutions of the so-called Schrödinger *equation*, which are wave functions, are of prime interest.[45] These solutions determine the *electronic structure* of molecules which determine their properties.

In 1944, Gabriel Kron suggested a direct analog electronic analog computer to model the behavior of the Schrödinger equation under different boundary conditions. A typical circuit derived for this purpose is shown in figure 11.14.[46] Hans Kuhn developed an *acoustical analog computer* which consisted of a multitude of coil springs mounted on a carrier. These springs were interconnected horizontally by leaf springs of equal moduli of resilience. This machine is described in [Preuss 1962][pp. 291 f.] and was used to simulate state transitions of π-electrons in complex shaped molecule chains.

[44] A good description of the historical development of quantum chemistry can be found in [Anders].
[45] See [Preuss 1962] and [Preuss 1965] for the basics of quantum chemistry.
[46] See [Kron 1945/2] and [Kron 1945/1].

11.4 Mechanics and engineering

One of the main areas of application for analog computers are mechanics and engineering with problems ranging from vibrating coupled mass systems to the simulation of surge chambers etc.[47]

11.4.1 Vibrations

The analysis of vibrating systems is a typical application for an analog computer as the examples in sections 8.3 and 8.6 demonstrated. While in some cases vibrations are undesirable ranging from inconveniences to real hazards, they are necessary for the operation of other kinds of equipment. Thus the commercial impact of such studies is rather high. Depending on the complexity of the underlying mechanical system, the simulations can become quite complicated, too, as [Sankar et al. 1979][p. 11.] notes:

> *"The control of vibration in mechanical systems is a serious and challenging design problem."*

A notorious example for unwanted vibrations is the so-called *Pogo effect* that was experienced in large rockets.[48] This effect was caused by pressure variations in the feed lines for the rocket engines. These pressure variations caused unsteady combustion, which in turn resulted in varying accelerations of the rocket which tended to amplify these pressure variations and so on. In extreme cases the resulting forces can reach levels which would destroy the rocket. Other examples involve the stability of building subject to wind forces, earth quakes etc. An example for a system that relies on vibrations for its operation is a vibrating conveyor. Here the isolation of the vibrating machine parts from their mounting etc. is a critical design parameter.

Machines can be considered vibrating multi-mass systems with spring and damper elements coupling the various masses. Treating such systems with analog computers is described in detail in [Dornier/3], [Macduff et al. 1958], [Tse et al. 1964], and [Trubert 1968], while [Jackson 1960][pp. 416 ff.] focuses on the analysis of vibrating truss structures.

11.4.2 Shock absorbers

An interesting optimization problem is described by [Sankar et al. 1979]. A shock absorber is to be designed which will impose minimal accelerations on the attached mass while simultaneously limiting the distance the damped mass travels under pulse excitation. The necessary simulations for determining the parameters of this system are performed on a hybrid computer. This optimization is complicated by implementing an accurate model for dry friction which requires many non-linear computing elements.[49]

[47] A wealth of information and examples can be found in [Mahrenholtz 1968].
[48] See [Bilstein 2003][pp. 360 ff.] and [Woods 2008][pp. 83 ff.].
[49] Cf. [Dornier/4].

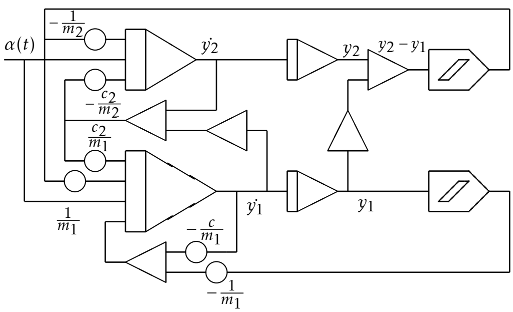


Figure 11.15: Computer setup to simulate a two-story building excited by horizontal ground movements (see [Hitachi 200X][p. 17])

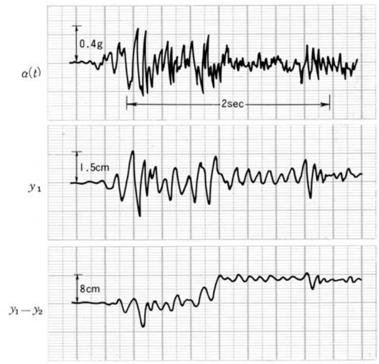


Figure 11.16: Results of an earthquake simulation (see [Hitachi 200X][p. 17])

[Mezencev et al. 1978] describe the development of an adaptive hydropneumatic anti-vibration device for marine propulsion systems by means of an analog computer.

11.4.3 Earthquake simulation

The analysis of the behavior of buildings excited by Love- and Rayleigh-waves of an earthquake is of great importance especially in countries with high seismic activity. The simulation of a multi-story building is described in [Hitachi 200X][p. 17]. The building is modeled as a multi-mass vibrating system where each floor is lumped together as a single mass. The walls act as spring-damper systems connecting adjacent floors.

A simple building with two floors is characterized by the weights of the floors, m_1 and m_2, and the moduli of resilience s_1, s_2 and damping coefficients d_1, d_2. With y_1 and y_2 denoting the horizontal displacements of the respective floors, the building dynamics are described by the following two coupled differential equations:

$$\begin{aligned} m_1\ddot{y}_1 + d_1\dot{y}_1 + d_2(\dot{y}_1 - \dot{y}_2) + s_1y_1 + s_2(y_1 - y_2) &= m_1\alpha(t) \\ m_2\ddot{y}_2 + d_2(\dot{y}_2 - \dot{y}_1) + s_2(y_1 - y_1) &= m_2\alpha(t) \end{aligned}$$

$\alpha(t)$ represents the ground motion induced by the earthquake. In practical applications real data stored on analog magnetic tapes was used for $\alpha(t)$. Based on these equations, the program shown in figure 11.15 can be derived.[50] The two function generators on the right hand side are used to implement two hysteresis curves which are necessary to model the non-linear stiffness of the building structures. The results of a typical simulation run are shown in figure 11.16.[51]

[50] The analog computer used in this example, a Hitachi 200X, features summers and integrators which provide normal and inverted outputs, thus simplifying computer setup considerably, which is reflected in the program shown.

[51] A similar simulation of a three-story building is described in [James et al. 1971][pp. 168 ff.].

Hitachi developed a special purpose analog computer named *SERAC*[52] in the late 1960s.[53] This machine featured a photoelectric curve follower which directly generated the excitation signal $\alpha(t)$ based on seismograms as input.

11.4.4 Rotating systems

Fast rotating systems are of high importance for many areas ranging from centrifuges to jet engines. These systems are often critical in the sense that human lives depend on their faultless operation.

11.4.5 Bearings

Bearings are some of the most critical parts in rotating systems. Especially long shafts which are common in turbines and similar systems are prone to bending vibrations which place additional and critical loads on the bearings. Thus quite some studies were made to simulate the behavior of such systems.[54]

Gas and steam turbines require hydrodynamic bearings since traditional bearing would fail rather quickly in such applications due to disturbances of the necessary oil film. The simulation of such hydrodynamic bearings on analog computers is described in [McLean et al. 1977] and [Rieger et al. 1974].

Some rotating systems achieve extremely high numbers of revolution per minute. A turbomolecular pump with a rotor weight of 1.7 kg, described in [Fremerey][p. 33], operates at 51,600 revolutions per minute. Ultracentrifuges for enrichment of uranium run at even higher speeds. Applications like these require magnetic bearings which are grouped into active and passive systems. Passive magnetic bearings were developed in the 1960s and rely on permanent magnets, while active magnetic bearings are based on electromagnets controlled by a servo loop.[55] Analog computers were not only used to determine the basic parameters of such bearing systems, but were also employed to model the servo circuits for active magnetic bearing systems. Often the analog computer was used in a *hardware in the loop* simulation where the analog computer is part of the system to be analyzed. In this case the analog computer was used to control the current supplied to the electromagnets of a real magnetic bearing in operation. In such a setup the consequences of parameter variations can be directly measured on the real hardware.

[Okah-Avae 1978] focuses on the effects on bearings caused by transverse cracks on turbine shafts. The goal of this work was to develop methods which allow an early detection of such damages. The simulation results were compared with actual data gathered from a damaged turbogenerator and showed a high degree of agreement.

[52]Short for *Strong Earthquake Response Analog Computer*.
[53]See [Hitachi 1969][pp. 6 ff.].
[54]See [Rieger et al. 1974].
[55]See [Fremerey] and [Fremerey 1978].

11.4.6 Compressors

A complex centrifugal compressor simulation is the topic of [Davis et al. 1974]. Interestingly, the simulation described does not only model the macroscopic behavior of the compressor system itself, but also models the gas dynamics. The simulation of another centrifugal compressor system on an analog computer can be found in [Schultz et al. 1974].

11.4.7 Crank mechanisms

Many systems like reciprocating pumps or piston compressors require a crank mechanism like a sliding crank to transform a rotary motion into a linear movement.[56] In the simplest case, this type of transformation can be achieved by a Scotch yoke mechanism as shown in figure 2.13 in section 2.6. In most cases, more complex mechanisms are necessary since a Scotch yoke exhibits rather high forces acting on the bearings of the rotating axis and the pickup. In addition to that many commercial applications require special curve progressions of the linear movement based on the rotary input to the mechanism.[57]

11.5 Materials science

11.5.1 Non destructive testing

Non destructive testing is of high importance due to aspects like cost savings or safety considerations. Typical application areas are mentioned by [Landauer 1975]:

> *"A speeding automobile goes out of control on a test track and crashes violently. A reactor vessel ruptures at a process plant when unstable conditions are reached. Or, a 650-MW generator loses bearing lubricant and goes into uncontrollable vibration. Each of these occurrences represents destructive testing at its best [. . .] or worst."*

In most cases, hybrid computers were used for non-destructive testing since often functions of more than one variable must be generated. Often, the analog computer was closely tied to the systems to be analyzed. More about this can be found in [Landauer 1975].

11.5.2 Ductile deformation

In 1956 *General Electric* performed a complete rolling mill simulation for the *Sharon Steel Corporation* with the aim of optimizing operating and startup procedures for a

[56] Cf. [Hütte 1926][pp. 82 ff.] for basic information about this topic and [Mahrenholtz 1968][pp. 129 ff.] or [Mahrenholtz 1968][pp. 79 ff.] for examples on how to simulate such systems on analog computers.

[57] A simple example can be found in [Dornier/5].

newly planned rolling mill.[58] As a result of this extensive analog computer based simulation, a *tune-up time*[59] of only one day, compared with seven to ten days for conventional rolling mills, was achieved.

Steel rolling processes are described by the Karman differential equation. Its implementation and analysis on analog computers is described in [Hitachi 1968]. A practical example is given in [Golten et al. 1967] where a hybrid computer is used for the actual computations. The mathematical principles for the simulation of cold rolling are delineated in [Heideprim 1976]. A thorough treatment of the simulation of drag rolling processes can be found in [Mahrenholtz 1968][pp. 122 ff.].

The determination of the ideal number of revolutions for roller drives has also been done by means of analog computers as [Rohde 1977] and [Rohde et al. 1981] show. The hybrid computer system used in these simulations is described in detail by [Schloemann-Siemag 1978]. [Mahrenholtz 1968][pp. 117 ff.] focuses on determining optimized parameters for convex dies, while [Ovsyanko] treats the analysis of plastic deformations on an analog computer in general. The simulation of injection molding processes is covered in [Ramirez 1976][pp. 22 f.]. The analysis of the deformation of steel plates subjected to the forces of an explosion by means of an analog computer is described in [James et al. 1971][pp. 175 ff.]. An interesting investigation of the forces exerted by shearing processes is given in [Dornier/6]. This example is a rare case in which an explicit differentiation of a variable is necessary.

11.5.3 Pneumatic and hydraulic systems

Hydraulic elements are of prime importance in many branches of industry. Accordingly much research on these devices was done on analog and hybrid computers. [Cohen 1971] e. g. describes the development of a pneumatic relay based on simulation runs on a hybrid computer, while more complex hydraulic switching elements are analyzed in [Hannigan]. General information about the treatment of hydraulic systems on analog and hybrid computers can be found in [Sankar et al. 1980], while [Nolan 1955] and [Ameling 1962/2] focus on the simulation of liquid flows (the latter places emphasis on complex piping networks).

Large scale hydraulic problems such as flood simulations or the behavior of sewage systems have been treated with indirect as well as with direct analog computers. Examples for such studies can be found in [Hitachi 1969][pp. 1 ff.] and [Paynter, ed. 1955] [pp. 239 ff.].

An interesting application was the simulation of so-called *surge chambers* which are reservoirs placed at the end of barrage pipes or the like to absorb shock waves excited by the closing of valves etc. Simulating the generation, distribution and elimination of these shock waves is of prime importance since these can destroy even large and sturdy machines and piping systems.[60]

[58] See [N. N. 1958/1].

[59] Tune-up time described the period of time that elapses between parameter changes and entering normal operation.

[60] See [Valentin 2003][p. 135].

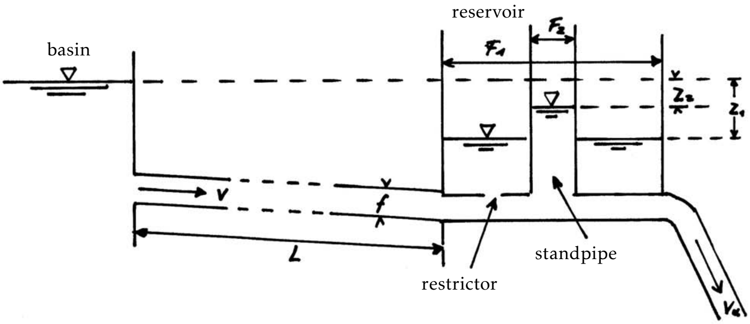


Figure 11.17: Surge chamber (see [Meissl 1960/2][p. 76])

A nice example is given in [Meissl 1960/1] and [Meissl 1960/2] where the focus is on the water level in a so-called *differential surge chamber* as described by [Johnson 1915] and shown in figure 11.17. Such a differential surge chamber can be described by the following differential equation system[61]

$$\begin{aligned} z_2 - z_1 &= \varepsilon_1 \left(\frac{F_1}{t}\dot{z}_1\right)^2 \operatorname{sign}(\dot{z}_1) \\ F_1\dot{z}_1 + F_2\dot{z}_2 &= f\,(v_a - v) \\ \dot{v} &= \frac{g}{L}\left(z_2 - \varepsilon v|v|\right) \end{aligned}$$

where z_1 and z_2 denote the respective water levels. The resulting computer setup is shown in figure 11.18. The results for z_1 and z_2 of a 100 second simulation run are shown in figure 11.19 and 11.20.[62]

A more complex example is given in [Jackson 1960][pp. 403 ff.] where the behavior of the *Appalachian surge tank* operated by the *Tennessee Valley Authority* is analyzed on an analog computer. General information about the use of analog computers for the investigation of water hammer problems is given in [Paynter et al. 1955], [Paynter 1955/1], and [James et al. 1971][pp. 184 ff.]. The design of surge tanks for hydro stations is covered in [Ando 1971]. [Sorondo et al.] describes the simulation of the piping system in a thermal power plant. The focus here is on the effects of the short run-up times exhibited by the pumps. Their spin-up time is only about 3.5 seconds, which places extreme stresses on the piping network, which has a total length of about four miles.

The problem of sloshing in a steel converter is treated in [Barth 1976][pp. 600 f.]. Interestingly, not only the sloshing itself is simulated but also its effect on the tilting control system for the converter. In addition to a direct analogy model which consisted of a scaled down version of the converter, an indirect analog model has been used for the simulations.

[61] See [Meissl 1960/2][p. 76].

[62] These simulations were actually performed on a Telefunken RA 463/2 analog computer as described in section 6.1. Choosing 100 seconds as the duration of a simulation run was due to the fact that the operational amplifiers of the RA 463/2 are not of the chopper stabilized type and integration errors due to drift effects tend to get excessive after about 100 seconds.

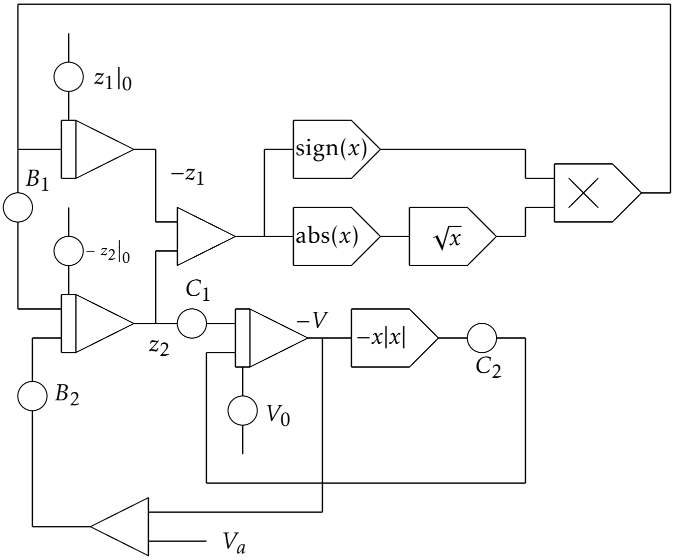


Figure 11.18: Program for the surge chamber simulation (see [Meissl 1960/2][p. 76])

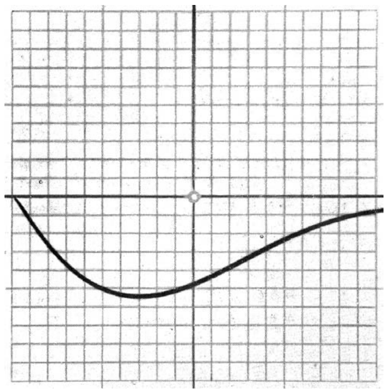


Figure 11.19: Water-level z_1 for $0 \leq t \leq 100$ s (see [Meissl 1960/2][p. 77])

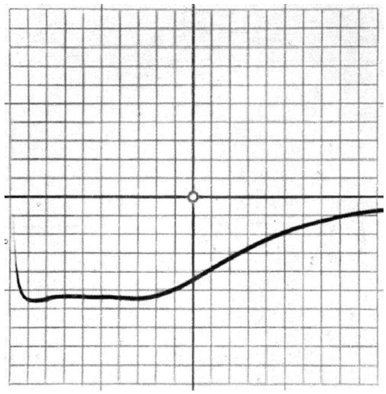


Figure 11.20: Water-level z_2 for $0 \leq t \leq 100$ s (see [Meissl 1960/2][p. 77])

11.5.4 Control of machine tools

Analog computers were also used for the development and implementation of machine tools. Two interesting analog interpolation systems for the control of turbine blade milling machines which were developed at the *Lewis Flight Propulsion Laboratory* are described in [Johnson 1962]: The first system implements a cubic interpolator that employs three mechanical integrators to generate a polynomial of third degree which interpolates a function specified by values given at equidistant sampling points. The second system was based on a direct analogy using a flexible steel ruler to implement a spline interpolation. This ruler was bent into position by a punched card controlled servo system that in turn controlled a number of linear actuators. Using a magnetic pickup in a servo loop the ruler was then used as a model to control the milling process.

11.5.5 Servo systems

Nearly every technical system relies on some kind of a servo system, thus the simulation of such systems using analog computers was of high importance and the number of studies dealing with related topics are abundant. Servo systems are divided into *linear* and *non-linear* systems. Linear servo systems employ actuators that operate in a continuous way while non-linear servo systems often use actuators that support only a limited number of states like most reaction control engines etc. The simulation of such systems is described in [McLeod 1962] and [Korn & Korn 1956][pp. 92 ff.], while [McLeod et al. 1958/2][pp. 297 f.] deals with instabilities of servo systems.

An example for the simulation of a complex servo system can be found in [von Thun] where the attitude control system of a large dish antenna to be used in radio astronomy is modeled. Effects that are covered are the inertia of the antenna itself, wind load, achievable accuracy, and the generation of non-linear antenna movements which are necessary for some tracking tasks.

11.6 Nuclear technology

Although many problems related to nuclear reactors and the like are not well suited for an analog computer due to vastly different time scales.[63] Nevertheless, analog computers were heavily used for research in nuclear technology and as training aids as the following sections show. A lot of information regarding the application of analog computers in this area can be found in [E. Morrison 1962], [Frisch 1971], and [Dagbjartsson et al. 1976].[64]

11.6.1 Research

Analog and hybrid computers were irreplaceable tools in fundamental nuclear research – not least because of the necessary safety constraints which forbid many experiments on real reactors. Typical application examples are the analysis of neutron generation and distribution in reactor cores,[65] the consequences of changes in reactor geometry due to temperature excursions and the generation of delayed neutrons,[66] heat-

[63] A steam bubble simulation for a sodium cooled breeder reactor contains equations spanning eight decades regarding to their time scale. The radius of the steam bubble raises by about five to six decades, resulting in changes spanning 20 to 25 decades in an r^4 term in the simulation (see [Frisch et al. 1969][pp. 19 f.]).

[64] The necessary basics of reactor physics are covered in [Markson 1958].

[65] See [E. Morrison 1962] – the simulations described there make extensive use of complex passive feedback networks to model the neutron flux densities etc.

[66] Cf. [Sydow 1964][pp. 225 f.].

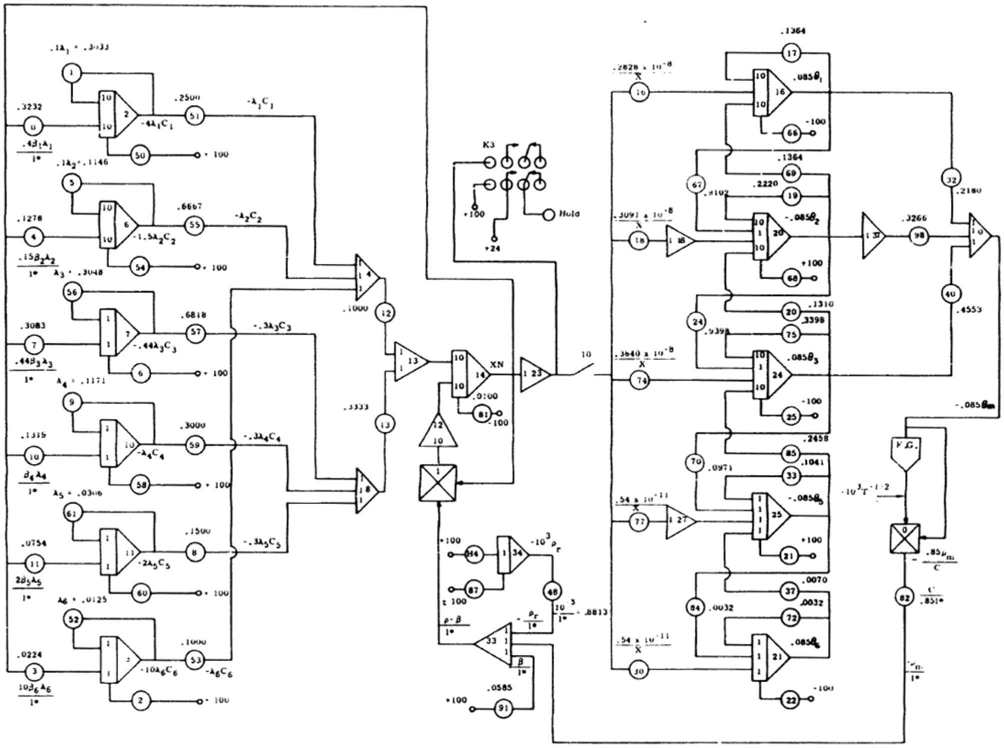


Figure 11.21: Nuclear reactor simulation (see [Jones 1961][p. 17])

transfer problems in pressurized-water reactors,[67] xenon-poisoning,[68] control system development,[69] etc. Most of these applications make heavy use of the typical capability of analog computers to perform a temporal expansion by simply changing the integration constants k_0 within a program since most of the effects under study are happening too fast in reality to be observable in detail.

Figure 11.21 shows a program which was used to analyze the stability of a research reactor in Hanford. Of special interest was the behavior of the reactor in case of a fault during which a rapid depressurization of the primary coolant loop occurs in conjunction with an increase in reactivity. The resulting power gain in output power spans three decades in a one second time frame which is about the maximum that can be handled even on a precision analog computer without either having the variables exceeding the range of ± 1 machine units or introducing exceedingly large errors.[70] Simulations covering longer spans of time normally have to be split into smaller steps, often requiring rescaling.

[67] [Sydow 1964][pp. 230 ff.].
[68] See [Frisch 1971][pp. 72 ff.] and [Sydow 1964][pp. 233 f.].
[69] See [Sydow 1964][pp. 234 ff.], [Frisch 1968], [Brey 1958], and [Scott 1958]. Many control system simulations require the solution of coupled partial differential equations.
[70] See [Jones 1961].

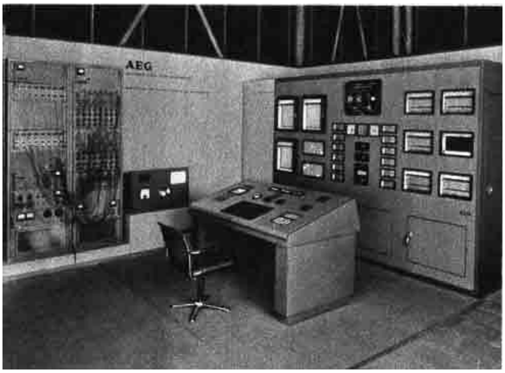


Figure 11.22: AEG reactor simulator based on a Telefunken RA 463/2 (see [Gerwin 1958])

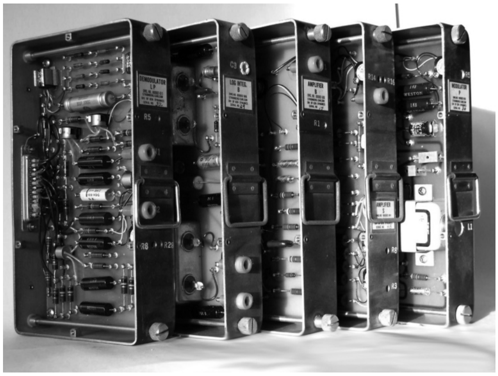


Figure 11.23: Typical reactor control analog computer modules (General Dynamics)

Another interesting simulation is described in [Hansen et al. 1959], where the dynamic behavior of a sodium cooled reactor is studied. The simulation is very detailed and even models hydraulic effects in the piping system and the like.

11.6.2 Training

Training reactor operators is of prime importance and thus requires simulators to be as realistic as possible. One of the first such simulators was built especially to train the operators for the nuclear reactor aboard the *N. S. Savannah* which was the first nuclear powered merchant ship and was launched on 07/27/1959. This simulator is described as follows in [N. N. 1964/3]:

> *"Through the use of an operator's control console identical to the one aboard the SAVANNAH, the two PACE 231R*[71] *General-Purpose Analog Computers [allow…], trainees [to] acquire operating experience just as if they were actually on board the ship.*
>
> *The analog computers, which are programmed to represent the complete reactor kinetics, as well as primary and secondary heat balances, activate all the recording instruments and dials on the control panel and respond to the student's manipulation of the operating controls just as the real reactor would."*

In addition to the operator consoles, this simulator featured an additional console on which instructors could introduce faults to be handled by the operators in training. A much simpler reactor simulator setup is shown in figure 11.22. This particular system was based on a small variant of the Telefunken RA 463/2 analog computer.[72]

[71] See section 6.2.
[72] See section 6.1.

11.6.3 Control

Analog computer elements were also used as elements for actual nuclear reactor control systems which was a natural step since analog computers played such an important role during the development of early nuclear reactors. Typical analog control elements are shown in figure 11.23. These particular modules were used in the control system of two research reactors installed at the University of Virginia.[73]

11.7 Biology and medicine

Applications for analog and hybrid computers in medicine and biology are abundant. The earliest attempts to analyze complex biological systems have been performed by B. Chance who used a mechanical differential analyzer in 1943 to investigate some aspects of enzyme kinetics.[74] The following sections focus on some later analog computer applications in medicine and biology.[75]

11.7.1 Ecosystems

A simple example for the simulation of a closed ecosystem has been shown in section 8.4. A much more complex and interesting simulation is described in [Rigas et al.] where the food chain ranging from phytoplankton to salmon in an aquatic ecosystem is modeled. This study focuses on external influences like the effects of fertilizers like phosphates from detergents washed into rivers and the sea, and seasonal effects due to incident solar radiation etc. Interestingly, this study was initially performed on a stored-program digital computer[76] with a graphical display but was then transferred to a mid range hybrid computer:[77]

> *"[. . .] The initial programming was laborious and the memory requirements were such that the use of the model was restricted to certain hours of the day. Eventually, rental on the CRT[78] was terminated and the interactive capability was lost. At about the same time, a small hybrid computer[79] became available and the development of an interactive hybrid computer model of the same system became a reasonable alternative."*

It turned out that this hybrid approach outperformed its all-digital predecessor and even the restricted precision offered by the hybrid computer had no perceptible influence on the simulation runs.

[73] Modules like these were, of course, also simulated on analog computers prior to their implementation, see [Cameron et al. 1961].

[74] See [Knorre 1971][p. 107].

[75] A wealth of information can be found in [Knorre 1971] and [Röpke et al. 1969].

[76] An IBM 360/67.

[77] See [Rigas et al.][p. 95].

[78] Short for *C*athode *R*ay *T*ube – in this case an IBM 2250 display was used.

[79] An EAI 690 hybrid computer, see [Rigas et al.][p. 95].

11.7.2 Metabolism research

The study of the metabolism of pharmaceuticals is of high importance in modern medicine. Especially the development of radioactive tracers allowed to gather metabolism data from organisms which formed the basis of extensive analog and hybrid simulations.[80] Basis of such studies are so-called *multi-compartment models* in which the way substances are transported between various parts of an organism during their metabolism is modeled. A variety of examples of such studies can be found in [Habermehl et al. 1969]. [Knorre 1971][pp. 201 ff.] describes the simulation of the metabolism of Paracetamol in an organism.

11.7.3 Cardiovascular systems

Multi-compartment models are also used to simulate cardiovascular systems. The effects of lesions can then be studied in detail on an analog or hybrid computer.[81] Direct analogies involving intricate hydraulic systems were also used to model cardiovascular systems as shown in [Stewart 1979][p. 43].

Another area of application of analog computers is the preprocessing and analysis of ECG data. Especially operations like filtering, peak detection, computation of jitter etc. have been implemented using analog computers.[82]

The following example shows how the so-called *respiratory arrhythmia*[83] has been studied using analog computers. The first such study is described in [Clynes 1960], a later study, on which this section is mainly based, can be found in [Albrecht 1968]. Picture 11.24 shows the program used in this latter study. Its purpose is to determine the parameters which control the heartbeat function $a(t)$.

The integrator in the lower left of this figure is used to generate $a(t)_{\text{real}}$ based on real ECG data of a test subject.[84] This function can be compared to a function $a(t)_{\text{simulated}}$ which is generated by the remaining circuit based on the respiratory activity $r(t)$ of the same test person. Based on this, the parameters of the simulator circuit can be adapted until both functions $a(t)_{\text{real}}$ and $a(t)_{\text{simulated}}$ are in good accordance. The resulting parameter set characterizes the behavior of the sinu-atrial node which can be described by $\ddot{y} = -(r_0 - v)y$ and the effects the test person's vagus nerve has on this.

11.7.4 Closed loop control studies

A nice example for the simulation of the inner closed loop control in organisms is described in [Stewart 1979][pp. 46 f.] which focuses on the regulation of CO_2 in the

80 Cf. [Habermehl et al. 1969].

81 An example for a system containing seven compartments can be found in [Benham et al. 1973].

82 These techniques have also been used to process EEG data, see [Feilmeier 1974][p. 30].

83 This term describes the effect that the time Δt between two successive heartbeats is decreased during inhaling and is increased during exhaling.

84 The integrator's mode of operation is controlled by a signal derived from the ECG signal. Normally the R wave of the signal is used for this purpose. Every occurrence of this R spike resets the integrator which yields a ramp function at its output which linearly corresponds to the time span between two successive heartbeats.

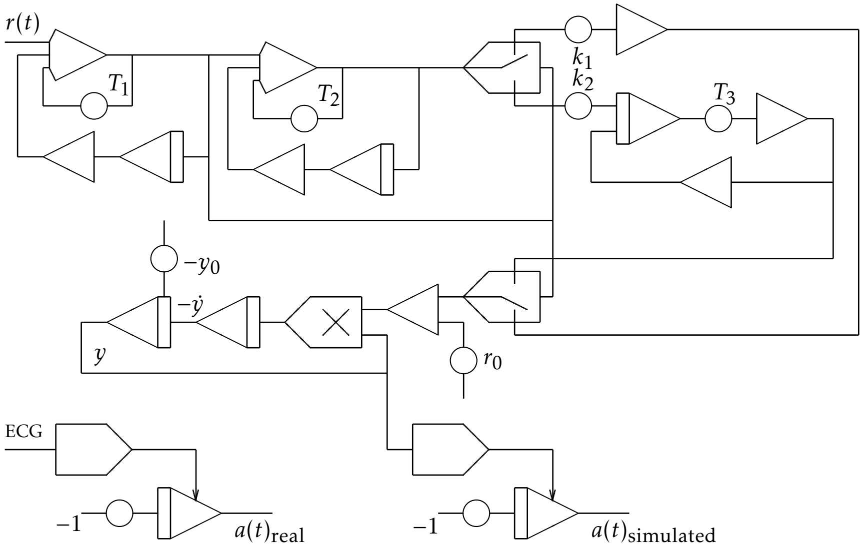


Figure 11.24: Simulation program for the respiratory arrhythmia (see [Albrecht 1968][p. 2])

lung. Another study in which the adaptation process of the pupil to changes in environmental brightness is described in [Giloi 1975][pp. 157 ff.], [Künkel 1961], and [Lunderstädt et al. 1981].

11.7.5 Neurophysiology

A basic question in neurophysiology is how action potentials in nerve cells are generated and propagated. In 1952 Alan Lloyd Hodgkin[85] and Andrew Fielding Huxley[86] developed a mathematical model describing signal generation and propagation in squid giant axons.[87] This model can be described by the following differential equation[88]

$$C\dot{V} = -g_{\text{Na}} m^2 h(V - V_{\text{Na}}) - g_{\text{K}} n^4 (V - V_{\text{K}}) - g_{\text{L}}(V - V_{\text{L}}) + I_a,$$

where g_i represents the conductivity caused by the potassium and sodium ions and the unavoidable leakage current in the axon, while $(V - V_i)$ represent differences to the equilibrium potentials V_i. The coefficients $0 \leq m, h, n \leq 1$ are determined by

$$\dot{m} = \alpha_m(V)(1 - m) - \beta_m(V)m$$
$$\dot{h} = \alpha_h(V)(1 - h) - \beta_h(V)h \text{ and}$$
$$\dot{n} = \alpha_n(V)(1 - n) - \beta_n(V)n$$

[85] 02/05/1914–12/20/1998

[86] 11/22/1917–05/30/2012

[87] These giant axons were often used in early neurophysiology research due to their size which can reach up to 10 cm in length and up to 1 mm in diameter.

[88] Cf. [Brouwer 2007][p. 3].

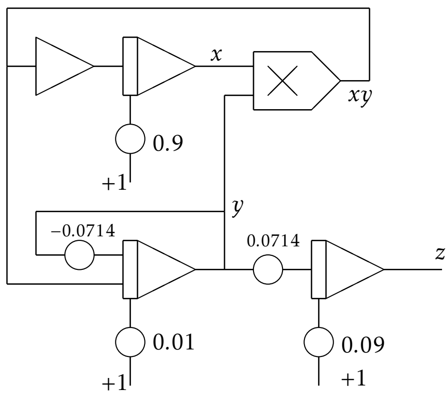


Figure 11.25: Setup for an infection spreading simulation (see [Hitachi 200X][p. 14])

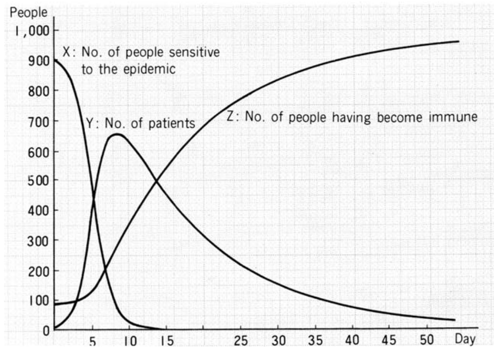


Figure 11.26: Results of a simulation run for the infection spreading simulation ([Hitachi 200X][p. 14])

where $\alpha_i(V)$ and $\beta_i(V)$ are determined through experiments. This model can explain many experimental observations and formed the basis of later research by Richard FitzHugh[89] and J. Nagumo which resulted in the following two differential equations that were studied extensively using analog computers:[90]

$$\dot{v} = v(a - v)(v - 1) + w - I_a$$
$$\dot{w} = bv - \gamma w$$

11.7.6 Epidemiology

A simple example for the application of analog computers in epidemiological research is given in [Hitachi 200X][p. 14]. A closed system consisting of 1000 inhabitants of which 10 are infected with some contagious disease is simulated. Every infectious individual can infect 1/1000 unresistant individuals per day. 90 of the initially 990 healthy individuals were already resistant, the disease, which is not deadly, immunizes its victims. These assumptions lead to the following set of three coupled differential equations

$$\dot{x} = \frac{xy}{1000}$$
$$\dot{y} = \frac{xy}{1000} - \frac{y}{14}$$
$$\dot{z} = \frac{y}{14}$$

which can be transformed into the program shown in figure 11.25.[91] The output of a typical simulation run is shown in figure 11.26.

[89] 03/30/1922–11/21/2007

[90] More information regarding these analog computer studies can be found in [Bekey 1960].

[91] The analog computer used in this study was a Hitachi 200X which features non-inverting integrators and summers which has to be taken into account for the program shown.

11.7.7 Aerospace medicine

Analog computers were also used extensively for aerospace medicine applications. [Paynter, ed. 1955] describes the analysis of the forces acting on a pilot who bails out with an ejection seat. For this study a special-purpose analog computer was developed to model the effects of such accelerations on organisms. The purpose of such *body dynamics* studies is to determine optimal acceleration curves that guarantee a quick separation of the ejection seat from a failing plane while not harming the pilot.[92]

Analog computers were also used to control large scale centrifuges which were used to determine the performance of pilots subject to accelerations as they are experienced during emergency situations, rocket launches or reentry maneuvers. A highly complex control system based on a large-scale analog computer installation is described in [Stone et al.].

11.7.8 Locomotor systems

Analog computers were also used at the boundary of biology and mechanics in locomotor studies. [Tomovic et al. 1961] performed simulations aimed at the question of the degree of autarky in typical locomotor systems. Similar studies are described in [Bekey et al. 1968][pp. 417 ff.]. Based on these a mechanical quadruped was developed.

11.7.9 Dosimetry

Analog computers were also used in nuclear medicine as an example from the *University of Chicago hospital* shows where a *Beckman EASE 2132* analog computer was used from the mid 1960s well into the 1970s to perform dosimetry calculations, a task required for treating cancer patients. In addition to that, this particular machine was also used for basic research involving nuclear and non-nuclear applications.[93]

11.8 Geology and marine science

Applications in geology and marine science often are of rather high commercial value, especially with respect to prospection tasks.

11.8.1 Resources

Oil and gas production rely heavily on hydraulic models of the oil and gas fields. These models are highly complex and require the solution of partial differential equations in at least four variables. Accordingly, direct analog computers have been built to study the effects caused by oil wells etc. to the belowground distribution of oil and gas. An

[92] See [Payne 1988][p. 272]. The term of interest in studies like this is the first derivative of the acceleration, a body experiences, which is called either *jerk* or *jolt*.

[93] Greg Parkhouse, personal communication to the author.

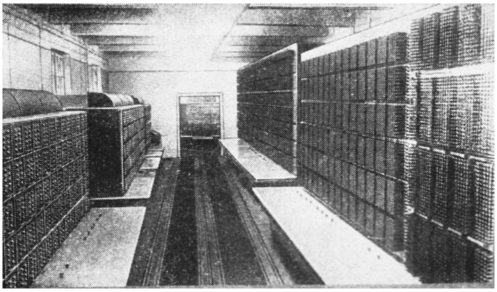


Figure 11.27: Overview of the ZI-S computer ([Ushakov 1958/1][p. 1812])

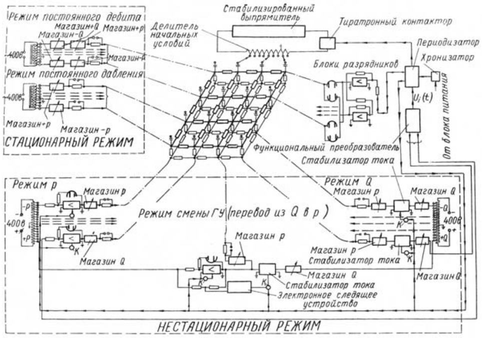


Figure 11.28: Structure of the ZI-S computer ([Ushakov 1958/1][p. 1813])

impressive example for such a direct analog computer is the system named *ZI-S* which was developed in 1958 at the *All-Union Scientific Research Oil-Gas Institute* in the Soviet Union. This machine was basically a three-dimensional grid of passive computing elements featuring about 20,000 of such vertices. In addition to this vast amount of passive elements, a large number of operational amplifiers and other active components was employed.[94] The high amount of components was necessary to achieve a sufficiently good resolution of the model.

This ZI-S analog computer which is shown in figure 11.27 was based on

$$\frac{\partial}{\partial x}\left(A_1(x,y)\frac{\partial P}{\partial x}\right)+\frac{\partial}{\partial y}\left(A_2(x,y)\frac{\partial P}{\partial y}\right)=0$$

and

$$\frac{\partial}{\partial x}\left(A_1(x,y)\frac{\partial P}{\partial x}\right)+\frac{\partial}{\partial y}\left(A_2(x,y)\frac{\partial P}{\partial y}\right)=A(x,y)\frac{\partial P}{\partial t}$$

respectively, where x and y represent coordinates in a single layer of the oil or gas reservoir, while P and A_i denote pressures and hydraulic conductivities.

Using ZI-S, reservoirs with a radius of up to 120 km were studied. These simulations considered up to 750 wells depleting the reservoir. The passive analog computer simulating the reservoir itself can be seen in the middle of figure 11.28. The reservoir is modeled with a three-dimensional mesh of resistors and capacitors which implement lag phenomena and the like. The surrounding modules implement the wells and the instrumentation of the simulation circuits.

11.8.2 Seismology

Prospection relies heavily on the analysis of seismograms to detect belowground reservoirs. This task traditionally requires the fastest stored-program computers available

[94]See [Ushakov 1958/1].

and was thus done or at least supported by analog and hybrid computers in the years before the mid 1970s.

[Evans 1959] describes a special-purpose analog computer that was developed and used to generate weathering time corrections. The implementation of this analog computer is quite similar to the BPRR-2 system described in section 11.1.9. Indirect analog computers were used by [Sutton et al. 1963] to perform filtering tasks, as well as spectrum analyses on seismic data sets.[95]

11.8.3 Ray tracing

Ray tracing is a common task in marine applications based on sonar data and of high commercial and military value. This task is rather complicated due to varying amounts of dissolved salt, temperature differences of different water layers etc. Accordingly, a ray tracer based on a small EAI TR-10[96] analog computer has been developed and described by [Light et al. 1966]. The task of tracing a single acoustic wave from a transmitter through a complex medium like the open sea to a pickup involves the solution of a differential equation of second degree. The analog computer described in [Light et al. 1966] uses 17 operational amplifiers and a special function generator to model the structure of the ocean bed reflecting the acoustic waves. A typical simulation run took about 15 minutes to complete.

11.9 Economics

The first one to study economic models by means of analog computers was Alban William Phillips,[97] a New Zealand economist. Since he was trained as an engineer, the idea of applying engineering principles to economic problems was probably quite obvious to him. Accordingly, in 1949 he developed and built a hydraulic analog computer named *MONIAC*,[98] the structure of which is shown in figure 11.29.[99]

This system, of which eventually ten were built, which was affectionately known as *financephalograph* was not only used for early research in mathematical economics but also for teaching purposes due to its structural clearness. The money flow of an economy is represented by colored water which flows through various computing elements. MONIAC features nine control variables which represent tax load, import/export subsidies and the like.[100]

[95]The results of an analog approach to spectral analysis were also compared to those generated by a stored-program digital computer in this study.

[96]Cf. section 6.3.

[97]11/18/1914–03/04/1975

[98]Short for *Monetary National Income Automatic Computer*.

[99]See [Phillips 1950], [Swade 1995], [Care 2006], and [Fortune 1952].

[100]MONIAC inspired some caricatures like that shown in [Swade 1995][p. 16] as well as the analog computer sitting in the basement of the bank in Terry Pratchett's novel *Making Money*.

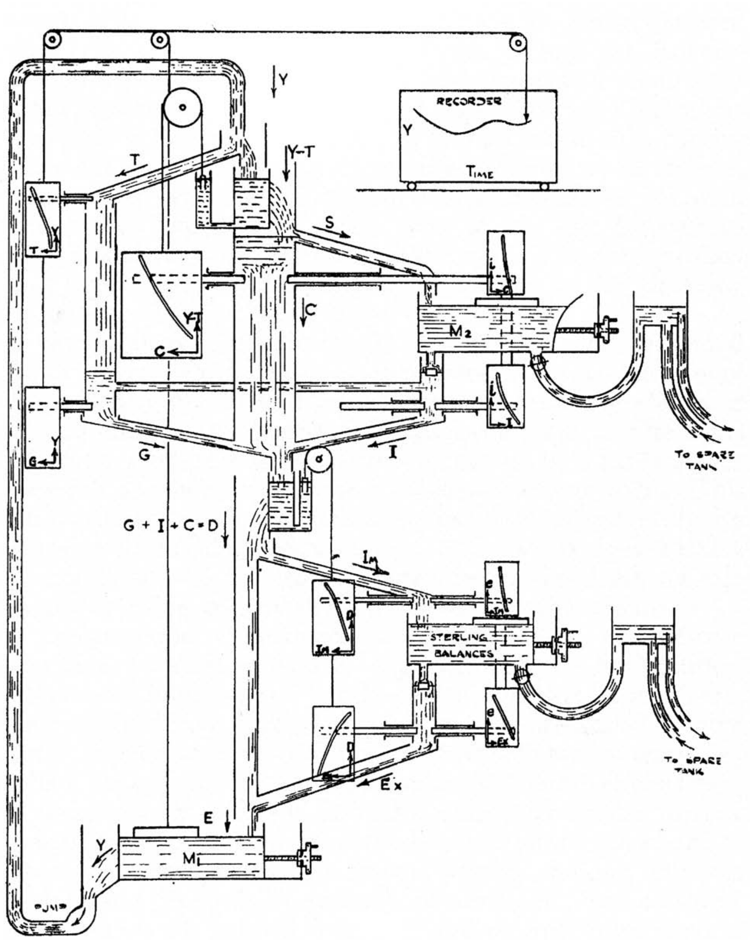


Figure 11.29: Phillips' hydraulic simulator for economic dynamics (see [Phillips 1950][p. 302])

In later years, Philipps used an analog electronic analog computer at the *National Physical Laboratory*[101] as well as a stored-program digital computer.[102] Interestingly, he still preferred the hydraulic MONIAC despite its shortcomings with respect to accuracy due to its high clearness. He characterized his models as being *"exposition[s] rather than accurate calculation[s]"*.[103]

Analog computer models for the study of economic problems were still used in later years as a number of publications like [Hülsenberg et al. 1975], [Herschel 1961], [Kregeloh 1956], and [Jackson 1960][pp. 363 ff.] show.

[101] *NPL* for short.
[102] See [Swade 1995][p. 17].
[103] See [Swade 1995][p. 17].

11.10 Power engineering

Quite early in the 20th century, the area of power engineering developed a demand for computational power that could only be fulfilled by analog and later hybrid computers. As with nuclear reactors, many questions regarding power systems cannot be analyzed on real equipment due to the inherent dangers of such experiments,[104] so only simulation was able to solve a number of vital problems. The following sections describe a number of typical simulations that were done in the past.

11.10.1 Generators

A study regarding the dynamic behavior of generators feeding theoretically infinitely large power grids which only contain asynchronous machines being fed, is described in [Juslin 1981].[105] The main problem of such a power grid is that asynchronous machines exhibit extremely different load characteristics depending on their state of operation. A starting asynchronous machine looks like a transformer with a secondary short to the supply, while this effect vanishes with increasing motor speed. The focus of this study is on severe faults that influence large parts of the simulated power grid and may even yield to oscillatory behavior.

[Gilbert 1970] describes the simulation of a generator with a parasitically loaded speed controller to be used in aerospace applications. The model used consists of the turbine driven alternator, a voltage regulator and exciter, a parasitic load speed controller and the load itself.

11.10.2 Transformers

As simple as a transformer might look at first sight, its behavior is not easy to predict, thus many studies regarding the dynamic characteristics of transformers have been conducted. [Giloi 1960] and [Telefunken 1963/2] cover the simulation of a transformer under mixed ohmic/inductive or ohmic/capacitive loading. The primary and secondary currents are of interest in this study, where also the dynamic magnetization curve of the transformer is taken into account.[106]

11.10.3 Power inverters and rectifiers

Much more complex than this is the simulation of power inverters and (controlled) rectifiers. Such devices are essential for high-voltage direct current systems, the first of which were developed in the early 20th century. Inverters are also commonly used to drive synchronous machines etc.

The analysis of a controlled three-phase rectifier by means of an analog computer model is described by [Eyman et al. 1976], while similar studies can be found in

[104] See [Noronha].
[105] The constraints mentioned are rather realistic in a mainly industrial environment.
[106] This magnetization curve is implemented by a diode function generator. It should be noted that this study is a good example for the implementation of implicit functions.

[Blum et al.], [Tisdale], and [Nava-Segura et al.]. The latter focuses on fault situations like defective thyristors and the like.

The conversion of a direct current into alternating or three-phase current by means of power inverters has been analyzed in analog computer studies like [Krause 1970], where basic approaches to simulation are described, and [Fornel et al. 1981] where a power inverter loaded by an asynchronous machine is modeled. [Bellini et al. 1975] describes the analysis of an inverter driving electromechanical actuators, while [Winarno 1982] focuses on the special requirements posed by large photovoltaic systems.

The analog model of a high-voltage direct current system is described in [Sadek 1976] who notes that such a system cannot be readily described mathematically in closed form, so that simulations based on models are necessary to gain deeper insight in the dynamic behavior of these systems.

[Borchardt et al. 1977] describe the detailed simulation of a high-power inverter system which drives the magnets of a large synchrotron at DESY. This study was necessary since the synchrotron was already in operation when the need for significant changes in the power supply system for the magnets arose. The system itself could not be used to take measurements during parameter changes since this would have disrupted its operation. Thus an extensive simulation was performed on a Telefunken hybrid computer HRS 860 which consisted of a RA 770 precision analog computer and a TR-86 stored-program digital computer. This model even took care of intrinsic effects of heavily loaded thyristors and other parts of the inverter system and proved to be so accurate that the conversion of the system was performed successfully mainly relying on the simulation results of this study.

Another interesting analog computer study regarding the behavior of inverters during severe fault scenarios is described by [Borchardt et al. 1969]. The simulation was focused on the development of suitable protection measures for large inverter systems.

11.10.4 Transmission lines

Long power lines are complex structures exhibiting a complex behavior with phenomena like traveling waves etc. A summary of typical hybrid computer approaches for modeling power lines can be found in [Baun 1970]. A study with special emphasis on traveling wave phenomena is described by [Thomas 1968]. An interesting aspect is the necessity for time delay units to model the delays encountered in long power line systems accurately. Magnetic tape units and capacitor wheels[107] have been used in these simulations instead of Padé approximations.

11.10.5 Power grid simulation

Much more complex than the simulation of a long isolated power line is the study of whole power grid, yet such studies are of even greater importance than the former due to the dependency of our culture on the ubiquitous availability of electric

[107] Cf. section 4.11.

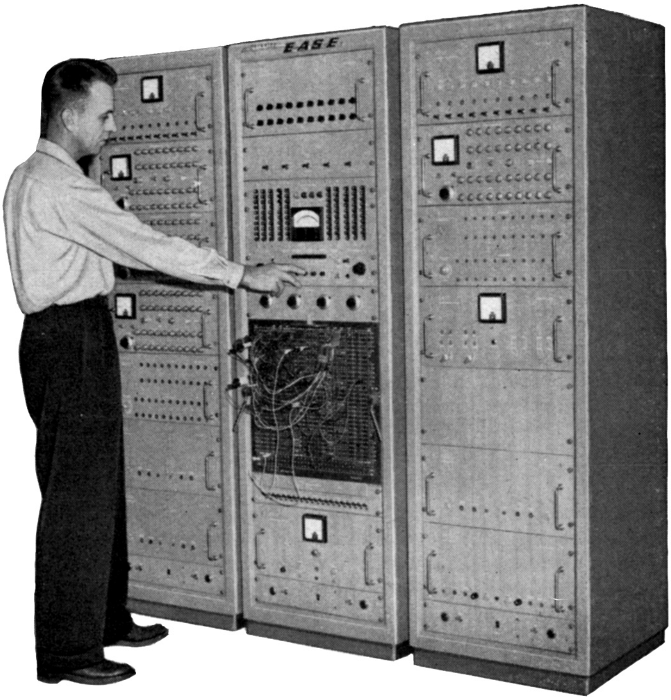


Figure 11.30: EASE 1032 analog computer used at the Bonneville Power Administration for power grid simulations (see [N. N. 1957/2])

energy. Such systems cannot be described mathematically in closed form, so simulations based on analogies are a suitable approach to their study. A good overview of the historical developments of such simulation techniques can be found in [Krause 1971] and [Krause 1974].

Driven by the rapidly rising complexity of the United States' power grid, the first *network analyzer* was conceived in 1925. The purpose of such analyzers has been described by [Nolan 1955][pp. 111 f.] as follows:

> *"Network analyzers have been used to solve quickly the many and various problems concerned with the operation of power systems. They are practical, adjustable miniature power systems. They can be used to analyze results during the progress of a system study and therefore play an active part in system planning as well as checking the performance of completed systems."*

The early network analyzers were direct analog computers, effectively modeling the power grid by means of discrete computing elements. Later studies made extensive use of indirect analog computers as the *Beckman EASE 1032* system shown in figure 11.30 that was used by the *Bonneville Power Administration* for power grid simulations. A typical problem arising in such simulations on indirect analog computers are algebraic loops.[108] Basic techniques for avoiding such loops in power grid simulations can be found in [Giloi et al. 1963][pp. 326 ff.] and [Giloi 1962].

An application example of a power grid simulator can be found in [Michaels et al.] and [Michaels] where a high-voltage distribution system consisting of five power stations and about a dozen transmission lines is simulated. This system was used to train switch room operators. An even more complex system based on a hybrid computer is described in [Enns et al.]. This hybrid approach was able to determine the state of a power grid containing 181 power lines from a given set of initial conditions in 20 seconds while the same computation using only the stored-program digital computer of the hybrid installation took 30 minutes.

The effect of transients caused by breaking conductor ropes, short-circuits, etc. are studied in [Thomas et al. 1968]. These simulations also required time delay units which were implemented with a capacitor-wheel containing 24 capacitors. Transients caused by lightning strikes in electric power transformation substations are covered by [Hedin et al.].

The effects of capacitive and inductive crosstalk between high-voltage power lines have been studied by simulating a complex power grid consisting of one 500 kV section and two 250 kV sections on a large analog computer installation. These studies are described in [Thomas et al./1] and [Thomas et al./2]. The vast amount of computing elements is a good measure for the complexity of the problem. The analog computer used contained 180 operational amplifiers, 350 coefficient potentiometers and 24 time delay units.

A large power grid simulator was proposed in 1976.[109] This *Hybrid Computer Power Simulator*, *HCS* for short, would have consisted of a special purpose analog computer containing specialized computing elements to represent typical analogies in conjunction with such simulations. Instead of a traditional patch panel, an automatic patch panel, a so-called *autopatch* was proposed. Unfortunately this machine was never built and autopatch systems did not prove commercially viable due to the extremely high amount of switching elements necessary.[110]

11.10.6 Frequency control

While power grids are rather complicated dynamic systems, things get even more complicated when multiple power grids are coupled together, as [Paynter 1955/2][p. 229] notes:

> *"[...] the original problem is fundamentally complex due to the multiplicity of generally disparate machines and regulators coupled together by a large number of 'elastic' links."*

[108] Direct analog computers do not suffer from such loops due to their lack of amplifying stages coupling the passive computing elements.

[109] See [Janac 1976].

[110] An interesting study on such autopatch systems was performed by George Hannauer on behalf of NASA, cf. [Hannauer 1968].

Early work on the analysis of coupled power grids using direct as well as indirect analog computers was begun in 1947 at MIT. [Krause et al. 1977] describes a study of shaft distortion in generators caused by misaligned grid frequencies when disjunct grids are coupled. Special questions regarding the synchronization and control of such coupled power grids are dealt with in [Sydow 1964][pp. 214 ff.].

11.10.7 Dispatch computers

Running a power station generates many optimization problems regarding the power station schedule, the supply of combustibles etc. An analog computer study aimed at the generation of optimal schedules for a hydrothermal power plant is described by [Perera 1969].[111]

Special purpose analog computers, so-called *Electronic Dispatch Computers*, *EDC* for short, have been developed to perform such optimization tasks and are described by [Washburn 1962]. The savings possible by the application of analog techniques were remarkable and justified the development of rather large analog computer installations:[112]

> *"Estimated annual fuel savings obtainable with EDCs may approach US$50 per megawatt installed. For a typical 1,000-megawatt system this saving can amount to US$30,000 to US$50,000 per year and thus may warrant an investment of US$250,000."*

Figure 11.31 shows an example of such an electronic dispatch computer, the *GEDA*.[113] power dispatch computer which saved US$ 200,000 per year (see [N. N. 1957/4]). A not too complex optimization example regarding a heating plant is given in [Heinhold et al.][pp. 224 ff.].

11.11 Electronics and telecommunication

The rapid development of electronics and especially telecommunications since World War II resulted in an ever increasing demand for component and circuit simulations. As [Kettel et al. 1967][p. 3][114] put it, the more complex electronic circuits get the more important it is to gain insight in their respective behavior by mathematical models and analogies instead of building breadboard prototypes. The following sections give some examples of such work.

[111] This study is based on the so-called *coordination equations* developed by Chandler, Dandeno, Glimn, and Kirchmayer, see [Chandler et al. 1961].

[112] See [Washburn 1962][p. 5-155].

[113] Short for *Goodyear Electronic Differential Analyzer*.

[114] This work also gives a good overview of typical applications of analog and hybrid computers in electronics and telecommunications development.

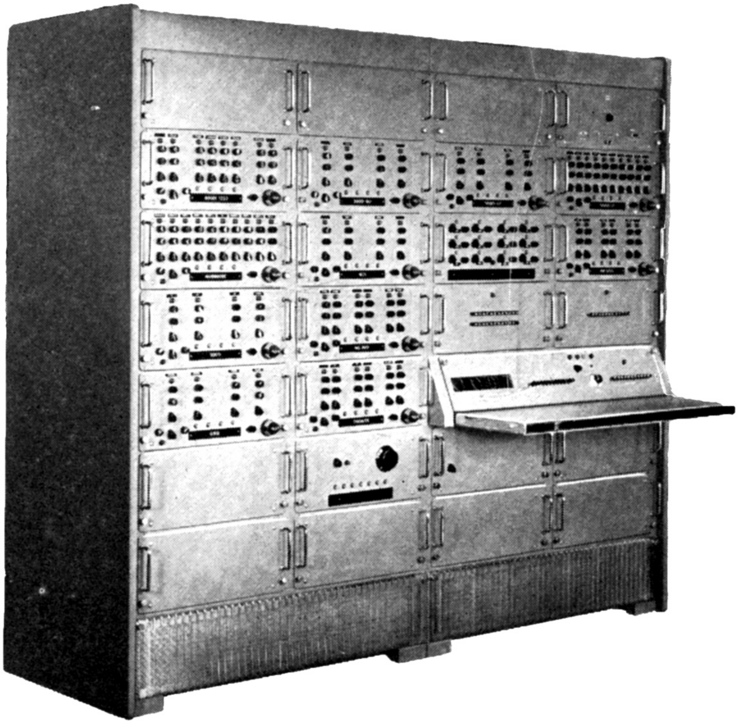


Figure 11.31: GEDA power dispatch computer (see [N. N. 1957/4])

11.11.1 Circuit simulation

An interesting analysis of the behavior of a transistorized circuit subjected to short transients is described by [Ranfft et al. 1977]. This application makes extensive use of the ability of an analog computer to perform temporal expansions since the effects of interest that are to be analyzed on the simulation setup happen in some 10^{-9} seconds. This study strikingly shows the complexity of such circuit simulations on general-purpose analog computers. As [Ranfft et al. 1977][p. 76] put it,

> *"[s]tandard analog computers are normally not well suited for simulation of transistor circuits."*

[Balaban][p. 771] remarks that

> *"a six to eight transistor circuit can be patched on a large analog computer."*

Accordingly Ranfft and Rein set out to develop a special-purpose analog computer aimed at the simulation of electronic circuits:[115]

[115] See [Ranfft et al. 1977][p. 76].

> *"Its advantages, when compared to the usual digital computer simulation, are low costs, simple operation, and short simulation time, allowing a fast man-machine dialog. This simulation method has already been successfully used in the design of several high-speed integrated circuits."*

The main advantage of this system was not only its rather high simulation speed but also the typical high degree of interactivity of an analog computer. The projected costs for an implementation containing 20 transistor models, 12 Schottky diodes, 24 resistors and two current/voltage sources were US$ 3,600. The same amount of money would have had to be spent to perform 100 to 200 transient analyses on a rented stored-program digital computer.

Another simulator, called *HYPAC*, short for *Hybrid PACTOLUS*,[116] is described by [Balaban]. This system is basically a hybrid computer with a special setup on the analog part which simulated basic electronic circuit elements. Using the stored-program digital computer of the hybrid computer, the analog computer's elements are time-shared between the various circuit elements of the circuit to be simulated. This made it possible to simulate circuits of nearly unlimited complexity given enough computer time.

An extensive simulation study on mixer circuits using a Telefunken RA 463/2 analog computer[117] can be found in [Sierck 1963]. This study was unique at its time since it was the first all-embracing simulation of such circuits and it proved that the results achieved by means of analog computers were meaningful and could be used for circuit design.

11.11.2 Frequency response

Before spectrum analyzers and the like were introduced as independent devices, determining the frequency response of circuits was a laborious task that could be simplified by an analog computer. Depending on the complexity of the circuit to be analyzed and the frequency range of interest, either measuring the response of a prototype circuit and processing this data on an analog computer or simulating the circuit itself as well as performing the signal analysis can be done on an analog or hybrid computer. The advantage of an overall simulation and analysis on an analog computer is the possibility to introduce arbitrary time scaling which is not possible with a prototype circuit under measurement.

Typical analog computer setups for performing a spectral analysis are described in [Bard 1965] and [Giloi et al. 1963][pp. 212 f.]. The analog computer programs described there are based on a maximum detector[118] circuit which is in turn based on a delay circuit. Using a sweep generator like that shown in section 8.2 the frequency response of a circuit, be it real or simulated itself, can then be determined.[119] Another

[116] See [Brennan et al. 1964] for more information about *PACTOLUS*, an early analog computer simulator running on an IBM 1620 stored-program digital computer.

[117] See section 6.1.

[118] Cf. [Sydow 1964][pp. 265 f.].

[119] Also the methods described in section 11.1.7 can be used for this purpose.

example for such a technique can be found in [Schüssler 1961] where a sampling process is used to generate a frequency response plot.

11.11.3 Filter design

The design of filter circuits is tightly entangled with the methods described above. An example for such an application can be found in [Larrowe 1966] where the design of a quadrature band-pass filter[120] is described. A very complex example of filter design is given in [Giloi 1961]. This study describes the development of a Wiener[121] filter on an analog computer which involves the solution of integral equations.[122] This requires a technique to have machine time running backwards which was solved by an analog tape drive on which values were recorded in a preliminary step and played back in reverse direction for the actual determination of the filter's parameters.

11.11.4 Modulators and demodulators

Modulator and demodulator circuits were also developed with substantial analog computer application. The design of phase detectors and frequency discriminators is described in [Manske 1968]. In this study the circuit under development as well as the processing of measurement data has been implemented on an analog computer. The analog simulation of a frequency modulation based communication system is the focus of [Hu 1972], while the study of modulators and demodulators for telemetering applications is described in [Kettel 1960][pp. 170 f.].

11.12 Automation

The area of industrial process measurement and control technology offers many applications for analog and hybrid computers. Obviously, the development of new processes, new control systems and the like can profit from studies performed on analog computers. In addition to that analog computers were sometimes even used as integral parts of complex control systems as [Kettel 1960][p. 165] noted in 1960:

> *„Ein vorhandener Analogrechner ist nicht nur ein Rechengerät, das z. B. die Berechnung eines Regelsystems erlaubt, sondern es kann ebensogut, vor allem wenn es klein und transportabel ist, als elektrischer Regler in ein Regelsystem eingefügt werden, damit in Verbindung mit der echten Regelstrecke der Regler optimal dimensioniert werden kann.“*[123]

[120] Such a filter basically consists of two separate filters which yield two signals with a phase difference of $\pi/2$.

[121] Norbert Wiener, 11/26/1894–03/18/1964

[122] See section 11.1.2.

[123] *"Not only can an analog computer serve as a computer to simulate control systems, but can also be part of a control system itself. As such it allows the optimization of the control system in place."*

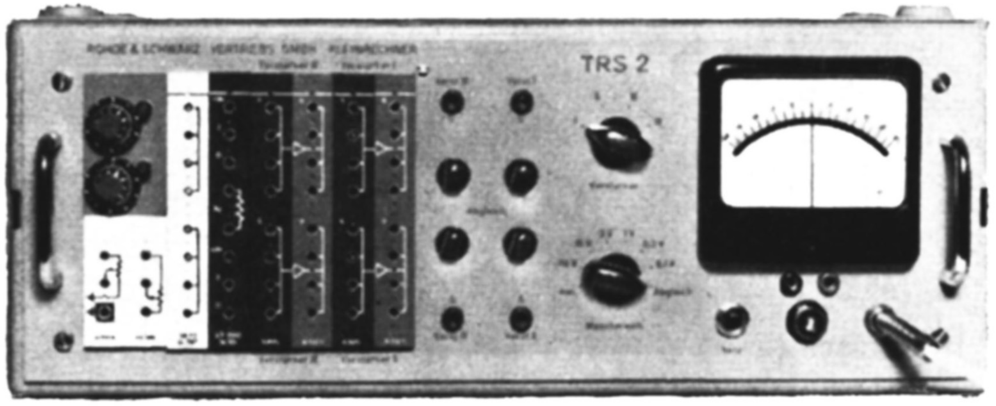


Figure 11.32: Data capturing and preprocessing system TRS-2 (Rhode & Schwarz, see [N. N. 1961])

11.12.1 Data processing

A nice example of an embedded analog computer is the data acquisition and processing system *TRS-2* shown in figure 11.32. This system was developed and manufactured by *Rohde & Schwarz* and contained standard analog computer components manufactured by EAI for use in their TR-10 and TR-20 tabletop analog computers. The system could be equipped with such standard analog computer modules as well as with prewired modules implementing programs for special applications.[124]

11.12.2 Correlation analysis

An interesting application example is given by [Wierwille et al. 1968] where the use of a hybrid computer for performing autocorrelation analyses on vibration data gathered during rocket tests is described. Apart from this analog approach a fully digital implementation is also described and both methods are compared.

11.12.3 Closed loop control and servo systems

The earliest example of an analog computer used to analyze and demonstrate closed loop control systems is George A. Philbrick's Polyphemus, described in section 3.2. A similar early simulation system, called the *Electro-Analogue*, is described by [Janssen et al. 1955]. This process and control simulator consisted of three largely independent subsystems: A model of the control circuit, a model of the process being controlled and an output system containing an oscilloscope display, some panel meters, and recorders. An interesting detail of this system is the implementation of time delays which are often necessary in control system simulations to model transport (including heat) delay times and the like. These time delay units were implemented as a string of capacitors and inductors with taps representing various delay times.

[124] Cf. [N. N. 1961]. The system shown in figure 11.32 contains from left to right two coefficient potentiometers, a couple of fixed resistors and two chopper-stabilized dual operational amplifiers.

Basic examples can be found in [James et al. 1971][pp. 215 ff.] and [Worley 1962] where bang-bang servos, proportional controllers etc. and their simulation on analog computers are covered. The simulation and development of governors is described in detail by [Ammon et al. 1959]. The simulation of a control system for a hydraulic system can be found in [Lotz 1969]. [Sydow 1964][pp. 211 ff.] described a complex example where the control system for a paper machine is simulated. The emphasis of this study is on the control of the 40 steam heated cylinders which must run at precisely controlled speeds. [Koenig et al. 1955] describes the parameter determination for flyweight governors used in a hydroelectric power station by means of an analog computer.

The analysis of damped, non-linear servo systems with analog computers is described by [Caldwell et al. 1955]. Such non-linear servo systems are of high importance for practical applications but are notoriously complex to model. Additional information about such systems can be found in [Hurst]. The determination of parameters for a non-linear control system is covered by [Suryanarayanan et al. 1968] where an EAI 231R analog computer is used.[125]

11.12.4 Sampling systems

Many control systems are, in fact, sampling systems in which input signals are sampled at discrete points and used to control the underlying process. The simulation and analysis of such systems is difficult on pure analog computers since the sampling process has to be implemented by bucket-brigade circuits implemented by chained integrators under individual control. Most often, hybrid computers were used to model sampling systems since the stored-program digital computer can easily implement the sampling process, while the underlying process is modeled on the analog part. Typical examples for this can be found in [Schneider 1960] and [Simons et al.].

11.12.5 Embedded systems

Analog computers were early used as embedded components of complex control systems as [Korn & Korn 1956][pp. 109 f.] describes:

> *"D-c analog representation of automatic control systems has proved to be a powerful aid in the design of prototype models and pilot plants and in the determination of starting procedures and of optimum controller settings after changes in raw materials or other conditions. But the utility of d-c analog techniques for automatic control applications is not restricted to computations of this type. Special-purpose d-c analog computers, which may often be conveniently assembled from the standard components of commercially available machines, can themselves serve as control-system elements in many applications suited to their characteristics."*

An example for such modules has been shown in figure 11.23 in section 11.6.3. Other examples include the computing elements developed by Telefunken for the RA 800

[125] See section 6.2.

analog computer[126] which were also sold as building blocks for industrial applications.[127] The basics of analog process computers are covered in [Ludwig et al. 1974], while [Weitner 1955] describes the implementation of a closed loop model. The design and implementation of an amplifier with a characteristic of $\sqrt[3]{x}$ is described in [Kinzel et al. 1962/1] and [Kinzel et al. 1962/2]. This amplifier was actually used to control injection parameters in large diesel engines.

11.13 Process engineering

Analog computers were invaluable tools for process engineering applications and accordingly most chemical companies implemented their own analog computing centers to aid their development departments. The importance of analog computers in this area can be seen by the following quotation from [Holst 1982][pp. 316 f.]:

> *"Analog computing capacity [at Foxboro] has increased from some 8-10 operational amplifiers in 1938 to more than 150 thirty years later."*

Table 11.1 gives an overview of the analog computing capacity installed at some chemical companies in 1961.[128]

A wealth of information and examples regarding the use of analog computers in process engineering applications can be found in [Wagner 1972], [Worley 1962], [Holst 1982], and [Ramirez 1976]. A small, yet powerful process simulation system based on an EAI *TR-48* analog computer[129] was introduced at the *Systems Engineering Conference* in 1964 by EAI. A highly complex simulation of a solvent recovery process is described by [Lewis 1958]. This simulation consists of 120 summers/integrators, 120 limiters, 48 summers, 28 servo multipliers, 16 photoformers, 300 coefficient potentiometers and 186 external precision capacitors.[130] Another example for a complex process engineering simulation can be found in [Graefe et al. 1974] where a copper melting process is treated, even taking the discontinuous flow of materials into account.

11.13.1 Mixing tanks, heat exchangers, evaporators and distillation columns

One of the basic elements in chemical plants is the mixing tank which can be simulated quite straightforwardly on an analog computer as described in [Ramirez 1976]. The simulation of heat exchangers and evaporators is more complex since these

[126] Cf. section 6.3.
[127] See [N. N. 1960].
[128] Equally large installations were used in other countries as well (see [Ushakov 1958/1] and [Ushakov 1958/2]).
[129] This system contains 48 operational amplifiers.
[130] Such complex simulations were the exception.

Company	Year	Number of amplifiers	Manufacturer
Dow Chemical Co.			
Midland Division	1954	20	Beckman (Berkeley)
	1961	140	EAI
Texas Division	1956	30	Daystrom (Heath)
	1961	80	Philbrick
E. I. du Pont de Nemours & Co.			
Newark, Del.	1950	30	Beckman (Berkeley)
	1955	50	Beckman (Berkeley)
	1958	120	EAI
	1960	300	EAI
Experimental Station, Wilmington, Del.	1960	70	Computer Systems
Monsanto Chemical Co.			
St. Louis, Mo.	1957	116	EAI
	1958	24	EAI
	1959	88	EAI
Ohio Oil Co.			
Denver, Colorado	1957	56	EAI
Humble Oil & Refining Co.			
Baytown, Texas	1960	80	EAI
	1961	80	EAI
Baton Rouge, La.	1959	80	EAI
	1960	40	EAI
Esso Research & Engineering Co.			
Florham Park, N.J.	1959	40	EAI
	1959	40	EAI
	1960	80	EAI
	1960	80	EAI
American Oil Co.			
Whiting, Ind.	1955	–	EAI
	1957	168	EAI
Standard Oil Co.			
Cleveland, Ohio	1955	90	Beckman (Berkeley)
	1957	10	Beckman (Berkeley)
	1961	170	Beckman (Berkeley)
Union Carbide Olefins Co.			
South Charleston	1956	30	EAI
	1958	60	EAI
	1959	60	EAI
Thiokol Chemical Corp.			
Brigham City, Utah	1959	168	EAI
Phillips Petroleum Co.			
Bartlesville, Okla.	1959	80	EAI
	1960	80	EAI
Chemstrand Corp.			
Decatur, Ala.	1960	80	EAI
Shell Oil Co.			
Shell Chemical Corp.	1960	120	EAI
Development Corp.	1956	24	Goodyear
	1957	24	Goodyear
	1960	10	Donner Scientific
	1960	10	Donner Scientific
Hercules Powder Co.			
Wilmington, Del.	1960	44	Beckman (Berkeley)
Daystrom, Inc.			
La Jolla, Calif.	1960	100	Computer Systems

Table 11.1: Analog computer installations for research in petrochemistry in 1961 (see [Carlson et al. 1967][p. 356])

are described by partial differential equations.[131] The simulation of an evaporator on an EAI 590 hybrid computer system is described by [Oliver et al. 1974], while a more complex example involving a heat exchanger reactor can be found in [Carlson et al. 1968].[132] The very complex simulation of a Fischer-Tropsch reactor is treated in [Govindarao 1975]. The main challenge in this type of simulation is the fact that gaseous, liquid and solid phases of the reactants coexist simultaneously in the reactor system.[133] This particular simulation required 33 integrators, twelve sample and hold elements, 13 summers, ten inverters, 18 multipliers, 80 coefficient potentiometers and a digital control system.

[131] Basic information about this can be found in [Billet 1965].
[132] Cf. [Schöne 1976/2] for a treatment of the mathematical modeling of heat exchangers.
[133] Basic information on the simulation of chemical reactor systems can be found in [Starnick 1976].

Even more complex are simulations of fractionating columns containing many column plates. A simulation involving three plates can be found in [N. N. 1958/2], while a more complex system with five trays is described in [Williams et al. 1958].[134] The complexity of such simulations is caused by the complex mixing processes and phase transitions happening at the column plates. A six column plate fractionating column simulation is the focus of [Ramirez 1976][pp.28 ff.].[135]

11.13.2 Adaptive control

The implementation and analysis of an adaptive control system[136] is described in [Powell]. The system covered in this study works in two time scales: A short time scale subsystem that implements a predictor and a slow running subsystem implementing the actual control loop which takes the values generated by the predictor as its inputs.

11.13.3 Parameter determination and optimization

Another important application area for analog computers was the determination and optimization of process control parameters. The complexity of such control problems often exceeds that of other optimization tasks so that most often hybrid computers were used to implement gradient methods. Examples for such techniques and applications can be found in [Kopacek], [Troch 1977], [Woźniakowski 1977], [Gröbner 1961], [Kramer 1968], [Kramer] and [Jamshidi 1976].

[Piceni et al. 1975] describe the evaluation of operating parameters for industrial processes. An interesting topic of this study is the comparison of a priori and a posteriori values to determine a measure for the quality of the underlying model. Optimal process control by analog computer studies is the focus of [Miura 1967].[137] An interesting optimization technique which restricts its parameters to binary values 0 and 1 is described in [O'Grady 1967].

A very interesting optimization scheme based on random generators[138] is developed in [Korn et al. 1970]. The basic idea is to perform random parameter variations and test their performance. If these variations resulted in a better overall performance, this new parameter set is used as the basis for the next variation cycle, otherwise the old parameters are restored.[139] Interestingly, it can be shown[140] that parameters determined by such a stochastic process are generally superior to parameters that are the result of a deterministic process.

134 A similar simulation is treated in [Worley 1962][pp. 5-80 ff.].

135 More realistic simulations involving much more column plates are usually out of the reach of analog computer installations. Using hybrid computers, a time-sharing approach for using the analog computer components and thus simulating higher numbers of plates is possible.

136 Adaptive control systems were developed in the 1960s.

137 Annotations to this can be found in [Rosko 1968], while similar studies are described by [Feilmeier 1974][pp. 259 ff.] and [Michaels et al. 1971].

138 See section 4.12.

139 This method can be seen as a precursor to genetic programming.

140 See [Tacker et al.].

11.14 Transport systems

Transport systems of all sorts can benefit from the application of analog and hybrid computers as the following sections show.

11.14.1 Automotive engineering

Until the late 1980s, analog computers were crucial elements in the engineering departments of all vehicle manufacturers. A basic application was already shown in section 8.6 where the dynamic behavior of a vehicle, modeled as a two-mass system, was simulated. A more realistic but still simplified three-mass simulation is described in [Sydow 1964][pp. 245 ff.] while [McLeod et al. 1958/6][p. 1992] covers the specific problems of the simulation of shock absorbers for heavy lorries and excavators. Similar studies were performed for railway vehicles as well. [Dornier/7] describes the application of an analog computer for data processing in a runout inspection. The ability to perform a spectral analysis is particularly important in this type of application.

11.14.1.1 Steering systems

Steering systems are of prime importance in cars – accordingly, analog computers were used to simulate steering systems giving the freedom to vary and optimize parameters prior to building prototype systems. A special purpose analog steering simulator developed by General Motors is shown in figure 11.33. With this simulator the dynamic behavior of a steering system with respect to yaw, roll and slip moments could be analyzed. The importance of this system is described by [McLeod et al. 1958/6][p. 1995] as follows:

> *"As a result of this work, an understanding has been gained of the effects of the various car parameters on the car's lateral response. In addition, a number of general observations have been made: Yaw and sideslip are strongly coupled, but there is only a weak coupling linking the roll to the yaw and sideslip. However, this weak coupling is often the reason that a particular automobile is stable or unstable, particularly in yaw."*

11.14.1.2 Transmissions

An interesting example for the application of analog computers in automotive engineering, namely the simulation of a four-speed automatic transmission, is described in [Dornier/1]. This example is especially interesting due to the high number of comparators employed to model the discrete gearshift points.[141]

A motor driven car can be described by

$$m\ddot{x} = F_{\text{drive}} - F_{\text{drag}} - F_{\text{rolling drag}} - F_{\text{road slope}}.$$

[141] A similar, although more complex simulation of an automatic transmission was implemented at *Daimler Benz* using a Telefunken RA 800 (see section 6.3). This analog computer system was also used to develop the first antiblocking system (Bernd Acker, personal communication, 08/23/2007).

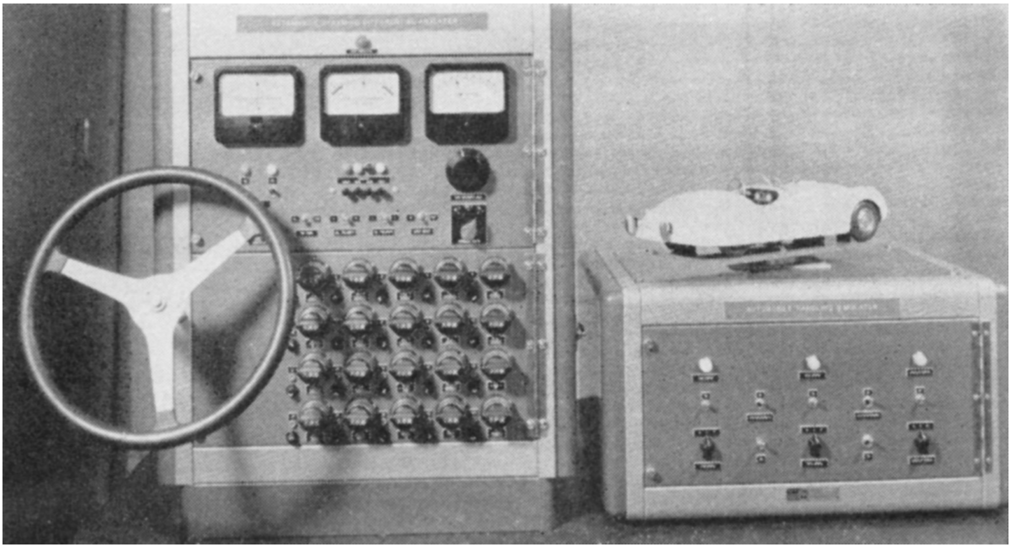


Figure 11.33: Steering simulator (see [McLeod et al. 1958/6][p. 1995])

To simplify matters, $F_{\text{rolling drag}}$ and $F_{\text{road slope}}$ are assumed to be zero, yielding

$$m\ddot{x} - F_{\text{drive}} + F_{\text{drag}} = 0$$

or, more detailed

$$m\ddot{x} - \frac{\eta i J_k}{r} M(n) + Cc_{\text{w}}Av^2 = 0, \tag{11.12}$$

with $m = 1120$ representing the car's mass, $\eta = 0.9$ the efficiency factor, $i = 4.11$ the axle ratio, r the wheel diameter, J_k the four gear transmission ratios, $M(n) = 5.306 + 50n - 81.25n^2$ the rotation speed dependent moment of force, and $Cc_{\text{w}}Av^2$ the drag which can be approximated by $0.003421v^2$. The gear transmission ratios are defined as $J_1 = 3.8346$, $J_2 = 2.0526$, $J_3 = 1.345$, and $J_4 = 1$. The transmission performs an upshift every time the rotational speed of the motor reaches $n = 6000\ \text{min}^{-1}$ which corresponds to the following velocities: $v_1 = 41.4$ km/h, $v_1 = 78.8$ km/h, and $v_1 = 118$ km/h.

Based on equation (11.12) the program shown in figure 11.34 can be derived. A typical simulation output is shown in figure 11.35.

11.14.1.3 Traffic flow simulation

Interestingly, even dynamic systems with strong discrete properties like traffic flows, can be analyzed on analog computers. A rather complex example can be found in [Landauer 1974] where the simulation of a transportation system consisting of 40 stations, 600 vehicles each of which can transport twelve persons, ten crossroads and 23 junction plates is described. The goal of this simulation is the development of an optimal schedule that ideally avoids traffic jams at all or at least minimizes such effects. A much simpler simulation is described in [Jackson 1960][pp. 371 ff.] which deals with a light signaling system and its parametrization.

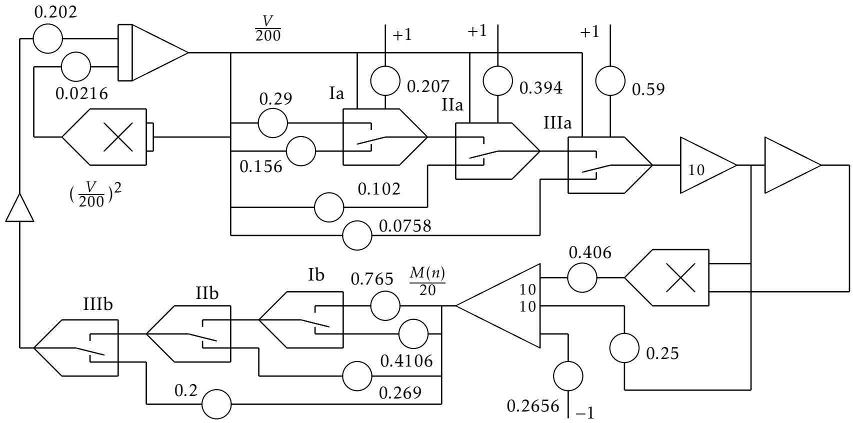


Figure 11.34: Program for the simulation of a four-speed automatic transmission (see [Dornier/1][p. 5])

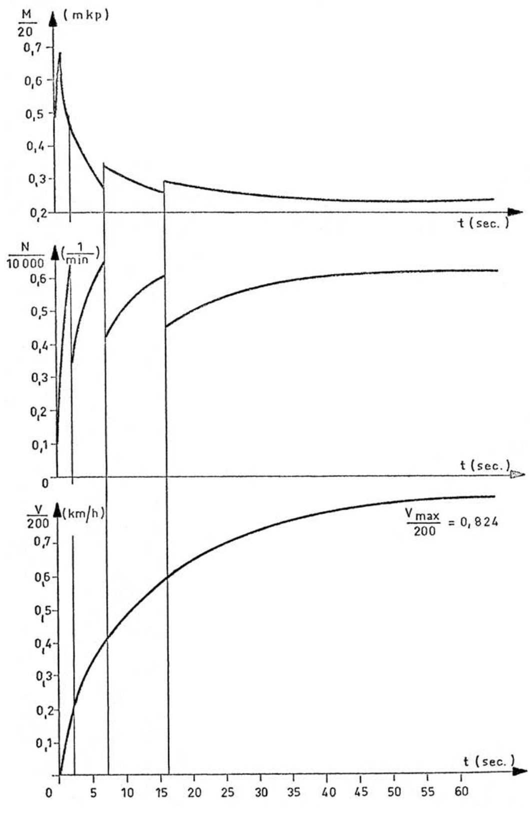


Figure 11.35: Results of a simulation run for the four-speed automatic transmission (see [Dornier/1][p. 4])

11.14.2 Railway vehicles

Many simulation tasks in conjunction with railway vehicles are similar to those encountered in automotive applications. Exceptions are the simulation of *marshaling humps*[142] and the treatment of motor coach trains. [Sydow 1964][pp. 238 ff.] describes the simulation of a direct-current railroad engine with emphasis on breaking procedures. Two different breaking techniques are analyzed: Breaking by placing a resistive load in parallel to the motor that acts as a generator and breaking with recuperation of the energy dissipated by the deceleration.

11.14.3 Hovercrafts and Maglevs

Analog and hybrid computer systems were even used in the development of hovercrafts and maglevs. [Leatherwood 1972] describes the simulation and analysis of a tracked hovercraft[143] with the goal of developing an active control system for this vehicle. [Thaler et al. 1980] describes a complex hovercraft simulation system in which the actual simulation was implemented on three stored-program digital computers while a COMCOR CI-5000 analog computer was mainly used as an interface to the operators of the simulator.

The simulation of maglevs is more complex than that of hovercrafts although their basic principle is not too different. The additional complexity results from the possibility of coupling several cars together to form a train. Coupled systems like these cannot be treated analytically so that large analog computer installations were used in early maglev developments. An example can be found in [Meisinger 1978] where a simplified one-mass maglev is modeled. This maglev hovers over a long track which bends under the weight of the maglev. The simulation not only implements the maglev itself but also these bending effects.

11.14.4 Nautics

The development of ships and vessels also benefitted from the application of analog computers as the following examples show.

11.14.4.1 Dynamic behavior

A rather complex simulation of the dynamic response of a ship subjected to waves is described in [Maslo 1974]. Such studies are of high relevance with regard to loading and unloading procedures as well as military applications. Parameters of interest in such studies are the ship's geometry and dimensions with respect to the wavelengths encountered etc. The partial differential equations involved in this type of problems were most often solved on hybrid computers where the discretization process was left to the stored-program digital computer. This particular simulation required 22 integrators, 42 summers, 28 inverters, 75 coefficient potentiometers, 30 comparators, 15 ADCs, and nine DACs which shows the complexity of this type of problem.

[142] Such a simulation is described in [Giloi et al. 1963][pp. 45 ff.].

[143] Although this sounds like a contradiction, such vehicles were explored since they promised very low frictional losses and offered better stability than many traditional railway systems.

[Glumineau et al. 1982] describes another interesting study in which the dynamic response of an oil tanker is analyzed. This tanker which is to be loaded is anchored and subjected to wind and wave forces. Since the forces of interest in this system have most of their respective energies concentrated at frequencies of about 1/200 Hz,[144] the time scaling ability of the analog computer is used to compress the simulation time into much shorter intervals.

11.14.4.2 Propulsion systems

The analysis of a complex turbine fuel system is described in [Horling et al.]. This task was solved on a hybrid computer which was capable of yielding results in real-time. A comparative study implemented on a *Control Data CDC 6800* stored-program computer[145] only achieved 10 % of real-time while a third implementation on a *UNIVAC 1110* was timed at 1 % of real-time clearly demonstrating the superiority of the hybrid simulation approach.

The simulation of a military vessel powered by twin diesel engines and a gas turbine is described in [Thompson]. Of special interest in this study is the behavior of the vessel while changing from one propulsion system to the other since the diesel engines are used for low and medium speed while the gas turbine provides high speeds. For a military vessel no loss of thrust must be guaranteed in all modes of operations and under all conditions.

11.14.4.3 Ship simulation

While the simulations described above focus on special aspects of ships, some applications, especially the training of skippers and the like, require complex ship simulation systems.[146] A simple submarine simulation with only one degree of freedom can be found in [Jackson 1960][pp. 384 ff.].

The simulation of an assault boat *LCM-6*[147] is described by [Kaplan]. The focus of this study lies on the dynamic behavior of the boat due to forces exerted by waves. The simulation of the waves is done on the digital portion of a hybrid computer while the analog part is used to implement a six degree of freedom simulation of the ship.

11.14.4.4 Torpedo simulation

The development of torpedos relied heavily on analog simulation techniques since tests involving real devices are extremely costly. Analog computers for the simulation of torpedos were used at the *Naval Undersea Center* from the early 1950s on.[148] The complexity of this simulation task is illustrated by the amount of computing equipment in use in the 1970s. All in all two *EAI 8800* hybrid computers[149] with 350 opera-

[144] Higher frequencies can be mostly neglected due to the long time-constant of the oil tanker itself.

[145] This system was later known as the *CDC 7600*.

[146] The basic requirements for such systems are described in [McCallum].

[147] Short for *Landing Craft Mechanized*, colloquially also known as *Mike Boat*.

[148] Cf. [Lowe].

[149] These fully transistorized machines offered a machine unit of ±100 V and were mainly used in applications where precision was of prime importance.

tional amplifiers each, three EAI 231R analog computers[150] with 250 operational amplifiers each, and two stored-program digital computers UNIVAC 1110 and UNIVAC 1230 were used. The former system was coupled to 64 ADCs and 120 DACs while the latter offered 32 ADCs and 24 DACs. This impressive installation allowed two torpedo simulations to be run simultaneously in real-time. These simulations also included models of the various seeker heads as well as the noise emission by the targets and the torpedos.

11.15 Aeronautical engineering

One of the main fields of analog computer application was in the area of aeronautical engineering. Accordingly some of the first commercially available analog computers were developed and built by *Boeing*, *Goodyear*, and *Reeves*, all strong players in aviation and space flight.

Simulations in aeronautical engineering and space flight have some special requirements which have to be kept in mind when analog and hybrid computers are to be used. First of all most typical problems involve variables with large domains, often too large to be handled by an analog computer directly. In addition to this, most simulations require many coordinate transformations and coordinate system rotations which either require special analog computing elements like resolvers[151] or have to be implemented using sine/cosine function generators, multipliers etc. In the end, most aerospace simulations require the generation of functions of more than one variable which either requires a vast amount of analog computer hardware or the use of a hybrid computer.[152] The importance of analog computers in these areas of application has been emphasized by [Levine 1964][p. 2]:

> "Ehricke[153] *believes that the accelerated development of American Missiles would not have been possible without [analog] computers.*"

The following sections show some typical analog computer applications in these areas – basic information on these topics can be found in [Bauer 1962/1].

11.15.1 Landing gears

The development of landing gears for high-performance aircraft is astonishingly complex and dangerous if based only on experiments with real flight-hardware.[154] Thus as early as in 1953 analog computers were used to simulate the dynamic behavior of these aircraft subsystems at the *Wright Air Development Center*. Under the auspices of

[150] See section 6.2.

[151] See section 4.10.

[152] This is one of the reasons why the idea of hybrid computers emerged from companies like the Ramo-Woolridge Corporation and Convair Astronautics, cf. section 9.

[153] See [Ehricke 1960].

[154] A good example for this are the landing gears of the *Me 262* jet which was plagued by a variety of problems, all of which could result in fatal accidents.

professor W. J. Moreland the first simulations of landing gears were performed on analog computers.[155] These simulations were based on systems of differential equations of 7th degree with four degrees of freedom and 14 parameters. An interesting detail of these studies is that the shock absorbers used and modeled often had different dynamic behavior depending on the direction of movement of the piston.[156]

Since no hybrid computers were available at that time, the necessary parameter variations were done manually which quickly yielded feasible landing gear configurations as the following quote shows:[157]

> *"For the past three years all new landing gear designs have been evaluated according to Professor* Moreland*'s method, thus eliminating the dangers that accompany violent shimmy."*

11.15.2 Aircraft arresting gear systems

The aircraft arresting gear systems aboard air craft carriers are highly complex systems since not only they must withstand the immense forces exerted by an aircraft decelerated quickly but also exhibit a force characteristic that does not damage the aircraft structure. Thus the control of the brake system used to tension the arresting cable is a difficult task that has been successfully modeled and analyzed on analog computers. Such a simulation, focusing on two types of aircraft, a jet fighter and a light bomber, is described in [Carlson et al. 1967][pp. 296 ff.].

11.15.3 Jet engines

The simulation of a turbo jet engine with afterburner on an analog computer is described by [Jackson 1960][pp. 426 ff.]. The focus of this study is on the design of a control system for the variable-geometry exhaust of this engine. This simulation requires function generators for functions of more than one variable. A simplified jet engine simulation is covered in [Schweizer 1976/1][pp. 422 ff.]. A study to test the compatibility of a particular jet engine and a special inlet is described in [Costakis 1974].

11.15.4 Helicopter rotor blades

The analysis of bending and torsion effects in helicopter rotor blades by means of an analog computer is described in [McLeod et al. 1958/4][pp. 1222 f.]. Mechanical systems like a rotor blade are described by partial differential equations and are thus not well suited for treatment with an indirect analog computer. Accordingly, the system used in this study was a direct analog computer quite similar in its structure to the HETAC and EAFCOM[158] described as follows:[159]

[155] See [McLeod et al. 1958/6][p. 1995].
[156] Cf. [Johnson 1963][p. 223].
[157] See [McLeod et al. 1958/6][p. 1996].
[158] See section 11.2.4.
[159] Cf. [McLeod et al. 1958/4][p. 1222].

"The direct analog computer consists of an assemblage of passive electrical circuit elements (resistors, capacitors, inductors and transformers), amplifiers, signal generators and control equipment. [...] The electrical analogy for the bending of beams has great importance in the direct analog method of dynamic analysis for lifting surfaces, since many lifting surfaces can be replaced by lifting lines with bending and torsional flexibility. This is certainly true of the helicopter rotor blade which characteristically has a very large span-to-chord ratio."

This direct analog consisted of about 100 inductors and 50 resistors and capacitors which represented seven discrete sections of a rotor blade.

11.15.5 Flight simulation

One of the most fascinating application areas for analog computer are flight simulators in which air planes are modeled with typically five or six degrees of freedom, allowing the study of various aircraft configurations, their behavior, etc.[160] Such analog computer based flight simulation systems are characterized as follows by [Korn & Korn 1956][p. 119]:

"The operation of a flight simulator on the ground is vastly cheaper as well as safer than operation of actual aircraft."

Accordingly, the *US-Air Force* acquired a GEDA analog computer[161] in the early 1950s which would eventually become the first of a long series of increasingly complex simulation systems.[162]

A very simple flight simulation in the vertical plane, which only takes the movements of the elevator into account, is described in [Wass 1955][p. 39]. Figure 11.36 shows the basic setup of the simulation. u denotes the speed along the x-axis, Θ is the inclination, while q represents the angular rate with respect to the y-axis. Based on this a simple simulation with the elevator angle η as the only input can be derived from the following simplified set of equations:

$$\begin{aligned} m(\dot{u}+w_0q) &= -mg\cos(\Theta_0)\Theta + uX_u + wX_x \\ m(\dot{w}+u_0q) &= -mg\sin(\Theta_0)\Theta + uZ_u + wZ_w \\ B\dot{q} &= wM_w + qM_q + \eta M_\eta \end{aligned}$$

Here, Θ_0 is the initial inclination angle, Θ the change of the inclination, u_0 and w_0 represent the initial speed components along the x- and z-axis. B is the moment of inertia, Z_u, Z_w and X_u, X_w represent the aerodynamic forces along the z- and x-axis per unit of u and w. Finally, M_w, M_q, and M_η are the moments about the y-axis per unit of w, q and η.

[160] The basics of such flight simulation systems can be found in [Bauer 1962/1], [Wass 1955][pp. 39 ff.], and [Korn & Korn 1956][pp. 115 ff.].

[161] See section 11.10.7 and [Hansen 2005][p. 147].

[162] The development of analog flight simulation at NASA's flight research center is described in [Waltman 2000]. Typical application examples of analog computers at *Autonetics* are described in [McLeod et al. 1958/1][p. 122].

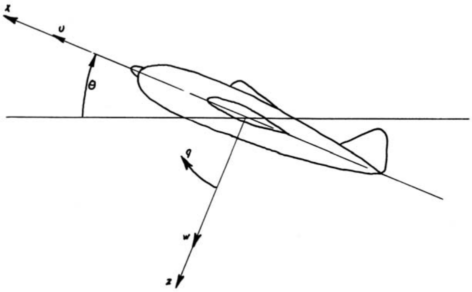


Figure 11.36: Longitudinal motion of an airplane (see [Wass 1955][p. 40])

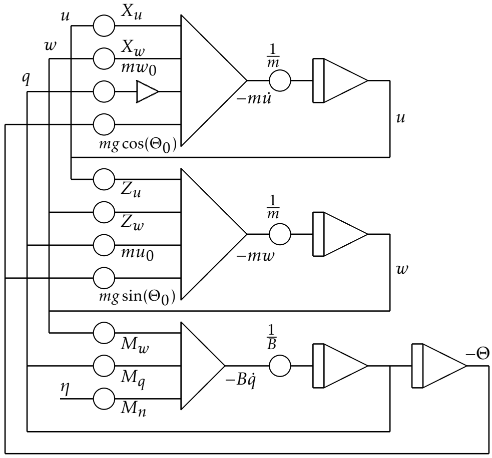


Figure 11.37: Simulation circuit for the longitudinal motion component of an airplane (see [Wass 1955][p. 40])

Based on these equations, the program shown in figure 11.37 can be derived. Using η as its input value the movements of the airplane in the vertical plane can be determined. This computer setup can now be extended to include a simple autopilot as shown in figure 11.38. The autopilot shown will hold the line of flight constant.[163] The term for the required elevator setting is[164]

$$\eta = k \iint \dot{w} - u_0 q \,\mathrm{d}t\,\mathrm{d}t.$$

Of course most of the analog and hybrid computer based flight simulators[165] were much more complex than this simple example. One machine, named *TRIDAC*,[166] is particularly noteworthy due to its sheer size and complexity. TRIDAC[167] was inaugurated on 10/08/1954 and was described as follows in *The Times*:[168]

> *"He was introducing Tridac [...] which is ten times larger than anything else of its kind in this country and one of the biggest computers in the world. It has been installed by the Ministry of Supply at a cost of about £750000 and within its massive form – the equipment would fill six ordinary three-bedroomed houses – there are mechanical computing elements which need 400-horse power*

163 A more detailed description of the simulation of autopilots on analog computers can be found in [Korn & Korn 1956][pp. 128 ff.].

164 See [Wass 1955][p. 43].

165 A good introduction to the mathematical modeling of airplanes and missiles can be found in [Schweizer 1976/5], while [Schweizer 1976/4] describes the implementation of a flight simulation system on a hybrid computer.

166 Short for *Three-Dimensional Analogue Computer*.

167 See [Wass 1955][p. 213] for additional information.

168 See [The Times 1954].

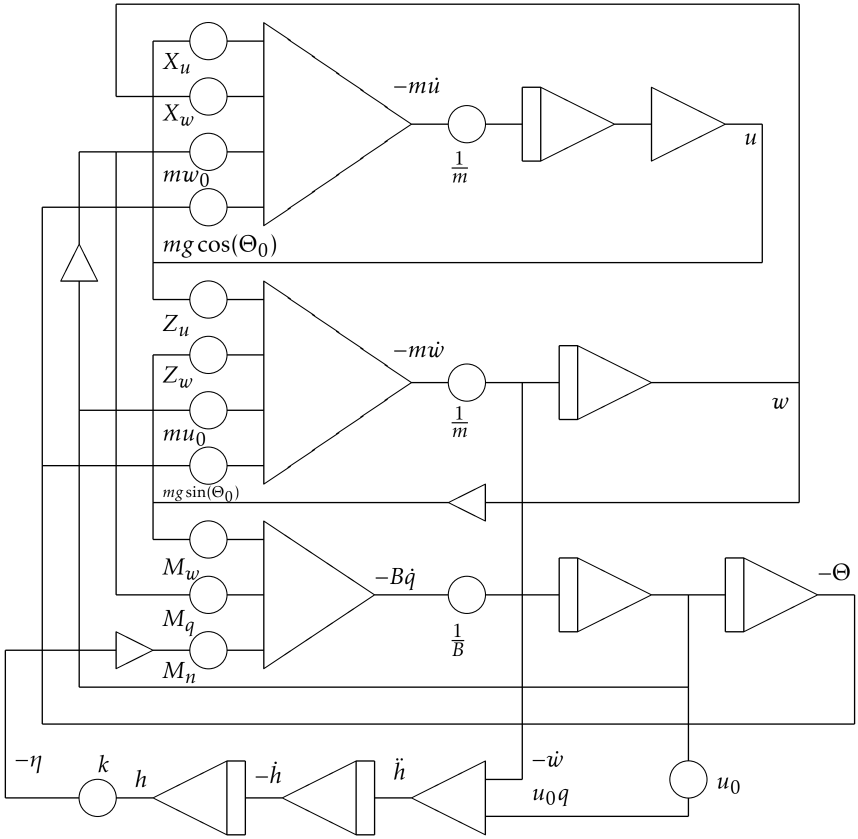


Figure 11.38: Longitudinal motion simulation with simple autopilot (see [Wass 1955][p. 43])

> *to drive them. The total electricity consumption would light a small town. As an example of its capacity, in 20 seconds Tridac could achieve as much as 100 girls using calculating machines in eight hours."*

Figure 11.39 shows the structure of TRIDAC. In contrast to general purpose analog computers, most of TRIDAC's computing elements were grouped into building blocks that performed basic operations necessary for flight simulations. These building blocks could then be configured and coupled together arbitrarily. The central element was the *aerodynamical unit* and surrounding blocks of integrators which integrate rotational and translational movements of the simulated flight vehicle. Other units are a resolver block which derives the necessary signals for a so-called *aircraft position computer*. Units to simulate autopilots, the dynamic behavior of control surface motors, and the like were also available. All in all, TRIDAC contained about 650 chopper stabilized operational amplifiers and a plethora of hydraulic function generators and

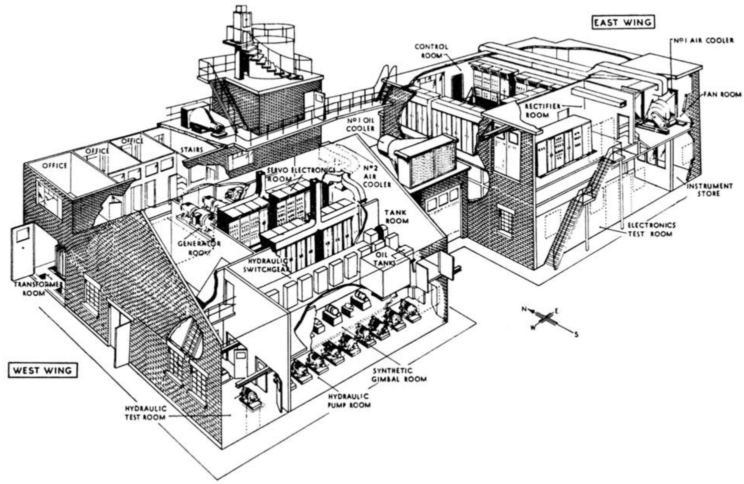


Figure 11.39: Structure of the TRIDAC simulator (see [Wass 1955][Fig. 141])

resolvers.[169]

In cases where an analog or hybrid simulation required hardware-in-the-loop, so-called *flight tables* were employed. The earliest such flight table was developed in Peenemünde during the development of the A4 rocket.[170] The idea was to mount the gyro system on a platform that could be tilted under control of an analog computer to simulate the movements of the rocket and evaluate the performance of the gyro and other guidance system components. While this first flight table, being used later until the mid 1950s at the Redstone Arsenal during the development of the Redstone missile, only offered one degree of freedom, later systems could be moved in three axes.[171]

Not all problems required such a giant analog computer for their solution. Figure 11.40 shows an inertia coupling simulation that was done at the NACA *High-Speed Flight Station* in 1955 during the development of the *Bell X-2* plane. Richard E. Day, one of the pioneers of analog flight simulation at NACA and later NASA, can be seen on the control stick, in the background are two GEDA computers. The high speed of analog computers made it possible to setup such man-in-the-loop simulations where the dynamic response of a simulated aircraft could be experienced directly.

Another large hybrid simulation system, *TASS*,[172] is shown in figure 11.41. This system consisted of a number of EAI 231R analog computers, a *DOS-350* digital subsystem,[173] and some stored-program digital computers like an *EAI 8400* and a *DDP-24* etc. Another large analog computer system, a *Beckman EASE 2133*, was in use at the *MBB Aircraft Division* and is shown in figure 11.42. This system featured 520 oper-

[169]Just the hydraulic subsystem for TRIDAC required 400 hp (about 300 kW) while the overall TRIDAC system was rated at 650 kW.

[170]See [Lange 2006][pp. 249 ff.], [Tomayko 1985][p. 233], and [Hoelzer 1992][p. 18].

[171]A plethora of information about flight tables can be found in [Bauer 1962/2].

[172]Short for *Tactical Avionics System Simulator*.

[173]An EAI 231R coupled with a DOS-350 forms a HYDAC 2000 computer system, see section 6.2.

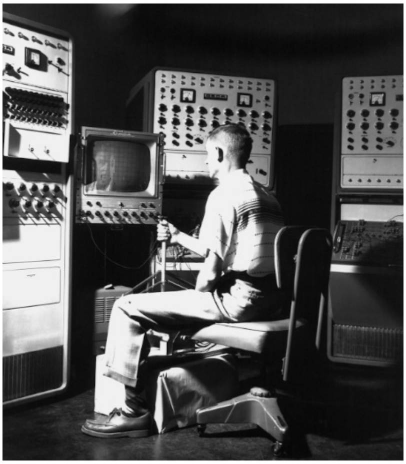


Figure 11.40: Inertia coupling simulation (Richard E. Day at the control stick, see [Waltman 2000][p. 6])

Figure 11.41: The *Tactical Avionics System Simulator* (photo from Bruce Baker, reprinted with permission)

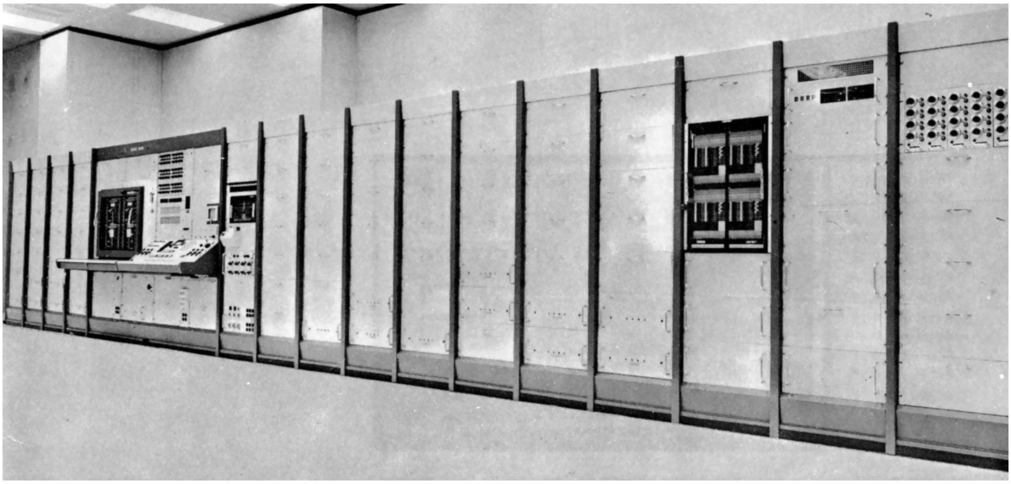


Figure 11.42: Beckman EASE 2133 analog computer installed at MBB Aircraft Division (see [MBB])

ational amplifiers, 72 multipliers, 64 function generators, ten sine-/cosine-function generators, 24 limiters, 240 servo coefficient potentiometers, 112 manual coefficient potentiometers and a variety of digital control elements. The simulation of a complex aircraft like the *VJ 101C-X2* vertical take-off plane required even larger analog computers. During the development of this particular plane, an analog computer installation containing more than 1000 operational amplifiers was used at MBB.[174]

[174] A list of analog and hybrid computer installations in use for aeronautical research in 1971 can be found in [Pirrello et al. 1971].

The accuracy of such analog computer simulations was demonstrated in a rather tragic way in 1956 when Mel Apt[175] died as the result of a stability problem of the X-2 plane that had been predicted previously by a simulation:[176]

> *"We showed him if he increased AOA[177] to about 5 degrees, he would start losing directional stability. He'd start this, and due to adverse aileron, he'd put in stick one way and the plane would yaw the other way [...] We showed* Apt *this, and he did it many times."*

At that time test-pilots did not show much trust in the results of such simulations which clearly were not "the right stuff" as Richard E. Day remembers:[178]

> *"Well, the simulator was a new device that has never been used previously for training or flight planning. Most pilots had, in fact, expressed a certain amount of distrust in the device."*

During Mel Apt's test flight, he encountered the critical region at Mach 3 where roll-coupling set in as predicted by the analog model and the X-2 entered a fatal spin. This fatal accident was in turn analyzed by the very same analog computer model.[179] As a result flight simulation played a central role in the following years as well in airplane development as in pilot training. Eventually, Neil A. Armstrong[180] spent about 50 to 60 hours of simulator training for each of his flights with the *X-15* while every flight only lasted about ten minutes:[181]

> *"[...] our simulators in the space program were so much more sophisticated and accurate, and our preparation was so much more intense, that we convinced ourselves that the pilots could handle whatever situation we might encounter in flight."*

A description of the simulation techniques used during the X-15 project can be found in [Day 1959]. [Mitchell et al. 1966] covers additional details of the simulation implementations. The analog computer studies regarding the X-15's reaction control system are described in [Stillwell 1956], while [Stillwell et al. 1958] deals with basic aspects of such simulations. A problem in the early days of analog flight simulation was how to display information in the cockpit. Simple analog instruments proved to be insufficient since pilots were and still are used to the special instruments found in real cockpits:[182]

[175] 04/09/1924–09/27/1956
[176] See [Waltman 2000][p. 138] and [Hansen 2005][p. 148].
[177] Short for *angle of attack*.
[178] See [Waltman 2000][p. 138].
[179] See [Waltman 2000][p. 140].
[180] 08/05/1930–08/25/2012
[181] [Hansen 2005][p. 148].
[182] See [McLeod et al. 1958/5][pp. 1387 f.].

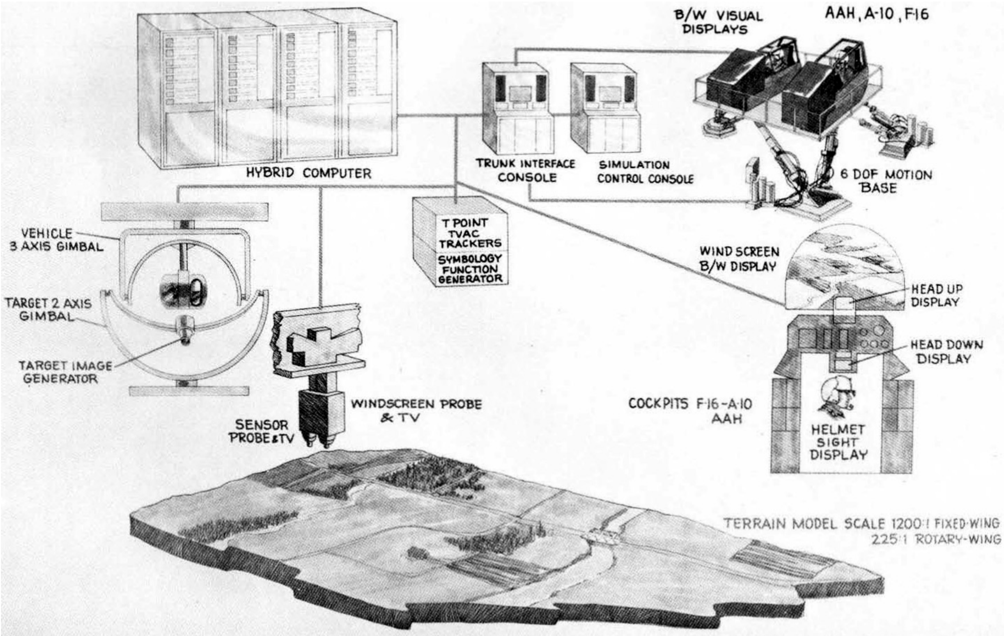


Figure 11.43: Flight simulation with terrain model (see [Baker 1978][p. 9])

> *"Presenting the pilot with the necessary flight instruments was a problem of major concern. Originally galvanometer-type meter movements were used. This type of display seemed satisfactory to the analyst and engineers with no actual flight experience. However, serious objections were raised when experienced test pilots were asked to evaluate the simulator's performance. Even though the necessary information was being presented to the pilot, the unrealistic instruments distracted the pilot so much that they impeded his efforts to truly evaluate the flight condition."*

Accordingly, adapters were necessary to interface the analog and hybrid computers to traditional cockpit instruments. Later systems used universal graphic display units as described in [Schweizer 1976/6][pp. 531 ff.].[183]

To create a realistic impression for the pilot, a flight simulator not only requires a cockpit which is ideally mounted on a hexapod platform allowing the necessary movements, but also features a realistic view of the outside world. Figure 11.43 shows the overall structure of the hybrid flight simulation system installed at *Martin Marietta's Simulation & Test Laboratory*[184] The picture generation is based on a large terrain model, which is probed by a video camera system which can be moved in three coordinates.[185] This model measures 80 by 40 feet and has a scale of 1200:1 for airplane

[183] An interesting study of the impact of instrumentation on a pilot's performance can be found in [Schweizer 1976/3][pp. 561 ff.].

[184] *STL* for short.

[185] The basics of such picture generation systems are described in [Schweizer 1976/7][pp. 394 ff.].

Figure 11.44: Terrain model (see [Baker 1978][p. 10])

simulations while a scale of 225:1 is used for helicopter simulations.[186] Figure 11.44 shows the terrain model in detail with a person standing on the far right giving an impression of the size.[187] The hybrid computer system used at Martin Marietta consisted of a three-CPU *Sigma-5* stored-program digital computer and six EAI 231RV analog computers with combined 1496 operational amplifiers.[188]

11.15.6 Airborne simulators

Even more fascinating than these flight simulators are so-called *airborne simulators*, also known as *in-flight simulations*, in which an analog or hybrid computer aboard an aircraft was used to change the behavior of this plane in a way that made the simulation of a wide variety of other airplanes possible.[189] Such simulators are also known as *variable-stability airplanes*.

A basic requirement for such a simulator is an airplane with a fly-by-wire system. The signals generated by the operator controls in the cockpit are then processed by the analog computer which in turn generates the necessary output signals to control the airplane's actuator systems. Normally two cockpits and two pilots were used with one set of controls wired directly to the plane's control motors and the like while the second set delivered the input signals to the analog computer. One of the earliest such systems is the *General Purpose Airborne Simulator*, *GPAS* for short, that was based on a

186 See [Baker 1978].
187 Bruce Baker remarks on this model (see [Baker 1978][p. 2]): *"Topography is rolling hills modeled after West Germany."*
188 See [Baker 1978][pp. 5 f.].
189 Cf. [Schweizer 1976/7][pp. 396 f.].

Lockheed JetStar. Development of this system began in 1960[190] and many studies were performed on this particular airborne simulator.[191]

Such simulation flights were not without risks as one of the GPAS programmers, Bob Kempel, remembers:[192]

> *"I remember the incident when we were airborne and we [*Larry Caw *and* Bob Kempel*] were looking at different feedback schemes. [...] Well, as you know, signs were sometimes confusing.* Fitz Fulton *was the pilot. The sign on beta was wrong, and we ended up with a dynamically unstable airplane because of it. We turned on the system for* Fitz *to evaluate, and the airplane immediately began an oscillatory divergence!* Larry *and I were in the back hollering to* Fitz *to turn it off, but* Fitz *was intrigued with the thing so he wanted to watch it as it diverged or maybe just teach us a lesson. He finally punched the thing off and* Larry *and I sighed in relief.* Larry *changed the beta-input sign, and we proceeded with the test.*[193] *[...] The JetStar was a fun airplane to fly in, but I always had a feeling of impending doom or something else going wrong."*

11.15.7 Guidance and control

Analog computers and computing elements were also used frequently and for a long time span as integral elements of guidance and control systems aboard aircraft and missiles. An example for such an analog subsystem is shown in figure 11.45 which shows a mechanical resolver module used in the *PHI-4* dead reckoning computer of the *Starfighter*. The device labeled *resolver assembly* basically contains a rotating ball which splits the flight velocity signal into an x- and y-component which are then fed to integrators.

An influential missile guidance system was developed for *Nike* missiles. Due to space constraints, a so-called *command guidance system* was developed. The heart of this system was a large ground based analog computer that was fed with target data from various radar sources and in turn generated steering signals for the missile.[194] This not only made the missiles lighter and cheaper but also allowed for improvements of the guidance system itself without the necessity of performing modifications on the missiles. The analog computer computed a course that was significantly better than a plain tractory curve, thus rendering evasive maneuvers of the target rather ineffective.[195] Such a control system requires some kind of memory to store past movements

[190] See [Waltman 2000][pp. 59 ff.] and [Berry et al. 1966] for a detailed description of the overall system. More information about this type of simulator can be found in [Armstrong et al. 1962], [McFadden et al. 1958], and [Kidd et al. 1961].

[191] [Szalai 1971] describes such a study in which the JetStar was validated for the simulation of the handling qualities of large transport aircraft.

[192] See [Waltman 2000][p. 64].

[193] The vibrations caused an observer in a chase plane to worry about an impending crash: "Stan *told me [...] he just looked out of the window to see where they would crash as he believed the wings would be torn off."* (see [Waltman 2000][p. 64]).

[194] Cf. [Department of the Army 1956].

[195] See [N. N. 1956][p. 4].

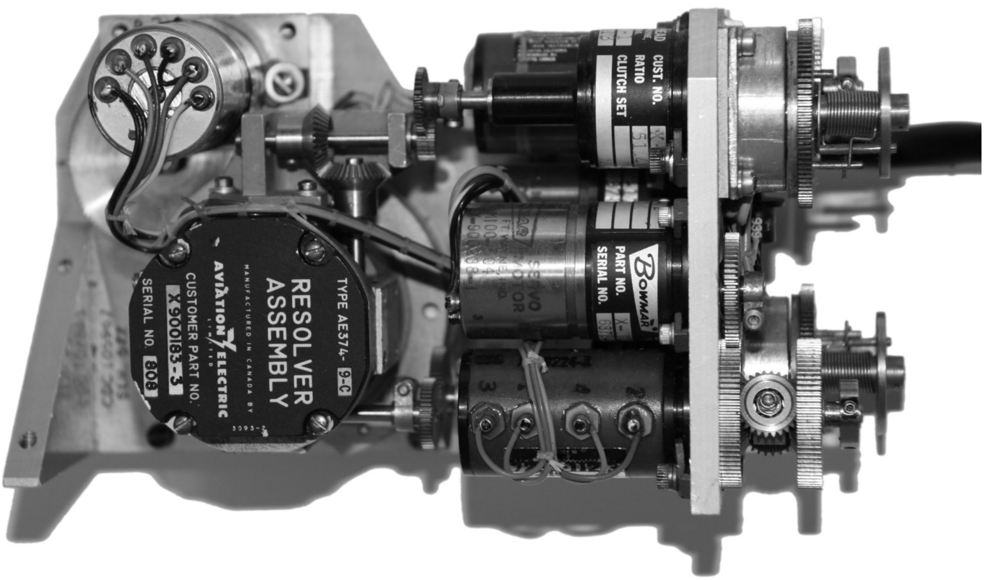


Figure 11.45: Resolver module of the PHI-4 navigation system that was used in the Starfighter

of the target and the missile. Remarkably, the Nike analog computer used differentiators with large time constants to implement these memory functions.[196]

In contrast, the *Polaris* missile, development of which started in 1956, employed an on-board guidance system that was based on a DDA.[197] These missiles were based on ballistic missile submarines which led to the requirement of a self-contained on-board guidance system. The required high accuracy of the missile system forced the development of a digital guidance system. Since the researches at MIT's *Instrumentation Laboratory* had mostly an analog computer background, the decision to develop a DDA was natural.[198]

The resulting bit-serial DDA featured a word length of 17 bit, which corresponds to a resolution of about 10^{-5}, and twelve words of memory implemented as shift registers. The first incarnation of this DDA had a volume of 11,000 cm^3, contained about 500 NOR-gates[199] implemented with germanium transistors and required about 80 W. A later implementation based on silicon transistors and an improved module packaging technology only required a fourth of that volume and about 40 W of power.

[196] See [N. N. 1956][p. 3]: *"[T]he computer needs a memory. [...] It must receive and remember position data for 4 seconds until it knows exactly the direction and speed of the motion involved. The memory of the computer lies in its differentiating circuits."*
[197] See section 10.
[198] See [Hall 1996][pp. 38 ff.].
[199] A NOR-gate is an OR-gate with an inverter connected to its output.

Of particular interest is the method used to perform trigonometric computations on these DDAs which is based on the *CORDIC*[200] system developed by Jack E. Volder.[201]

11.15.8 Miscellaneous

The applications described in the preceding sections are not exhaustive – there were many more, sometimes rather arcane, areas of application for analog and hybrid computers in aerospace engineering. An example for this is a study on parachute stability using analog as well as stored-program digital computers described in [Ludwig 1966]. This paper focuses on the strongly non-linear oscillations induced by parachute systems.

11.16 Rocketry

The development of rockets offered an equally fertile ground for the application of large analog and hybrid computer installations as the following sections show.

11.16.1 Rocket motor simulation

Liquid-propellant rocket motors exhibit an extremely complex dynamic behavior that is impossible to study using analytic methods only. The particular value of (analog) computer simulations for the development of rocket motors has been described by [Szuch et al. 1965][p. 2] as follows:

> *"A computer simulation, when properly used, can be a powerful tool in guiding an engine development program. It provides an easy and economical means for evaluating various design approaches and forewarns the designer of possible problem areas before costly hardware is developed and subsequently scrapped. Once a qualitative design is established, the system may be 'tuned' for optimum performance by varying the system parameters about their design values."*

This study describes the simulation of the *M-1* LH_2/LOX-rocket motor[202] that was intended to be used in *NOVA* rocket that was eventually superseded by the *Saturn*. This study is quite remarkable since apart from the analog computer, a stored program digital computer was used to study details of the dynamic behavior of the rocket motor that were not accessible to the analog simulation due to its limited precision. This simulation required about 250 operational amplifiers, 50 multipliers, 20 variable and five fixed diode function generators demonstrating the complexity of the problem. Especially useful was the time-scaling ability of the analog computer to study short-lived effects during engine startup.

[200] Short for *Coordinate Rotation Digital Computer*.

[201] See [Volder 1959]. The algorithm developed by Volder was also the basis for scientific calculators like the *HP-35*, introduced in 1972.

[202] LH_2 denotes liquid hydrogen, and *LOX* is liquid oxygen.

Other interesting sources are [Fox et al. 1969] who describe an analog computer based study of low-frequency flow dynamics encountered in nuclear rockets, while [Hart et al. 1967] focus on frequency responses and transfer functions of such rockets.

11.16.2 Rocket simulation

The requirements for the simulation and analysis of the dynamic behavior of rockets are quite similar to those encountered in airplane flight simulations, although normally no pilot is necessary in the loop for a rocket simulation. The basics of such simulations are described in [Jackson 1960][pp. 390 ff.]. The amount of computing elements necessary for such simulations is equally impressive as for airplane flight simulators. An early rocket simulation performed on a *REAC* analog computer required 304 operational amplifiers, 369 coefficient potentiometers, 25 servo multipliers, eight resolvers, eight diode function generators and two random generators.[203] The following quotation from [Bilstein 2003][p. 72] emphasizes these vast hardware requirements:

> *"For modifications and installation of new equipment, MSFC spent over \$ 2 000 000 after acquiring the site in the summer of 1962. The array of digital and analog computers for test, checkout, simulation and engineering studies made it one of the largest computer installations in the country."*

The advantage of high speed operation of analog computers compared with their contemporary stored-program digital computer counterparts has been noted by [Biggs][p. 6]:

> *"Because of the random nature of some of the missile system inputs the prediction or extrapolation work may require a large number of runs. AGWAC should compute these runs in real time or something like real time, whereas fast digital machines at present available would be at least one hundred times slower."*[204]

The setup of an early laboratory for three-dimensional guided missile simulation is described by [Bauer 1953]. This development was the objective of *Project Cyclone* at *Reeves Instrument Corporation*. The simulation system was put into operation into 1952 and consisted of 13 REACs, each containing seven integrators, seven summers, six inverters, and 23 coefficient potentiometers. In addition to this, 14 servo units containing a mix of servo multipliers and servo function generators, and three cabinets for additional devices like uncommitted operational amplifiers, limiters etc. were used. The majority of this installation is shown in figure 11.46.

Part of a large scale analog computer system for the simulation of the dynamic behavior of the *Saturn V* rocket is shown in figure 11.47. This *General Purpose Simulator*, *GPS*

[203] See [Jackson 1960][p. 393].

[204] AGWAC contained more than 400 operational amplifiers, 20 fully electronic multipliers, six servo multipliers (each having about 20 ganged potentiometers), and some resolvers and other special computing elements (see [Benyon 1961]).

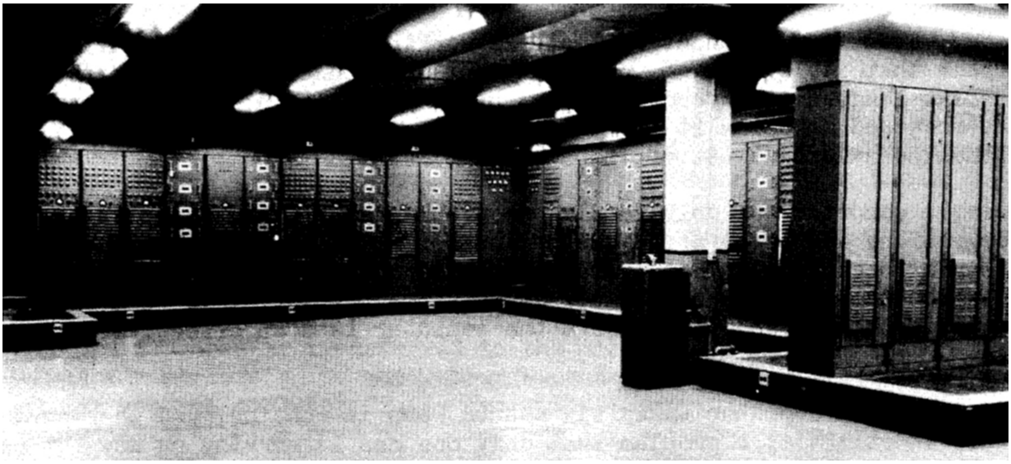


Figure 11.46: Laboratory for three-dimensional guided missile simulation (see [Bauer 1953][p. 195])

for short, was used to implement a twelve degree of freedom simulation of the first stage of a Saturn V rocket, taking effects like dynamic winds, bending of the rocket structure, and fuel sloshing into account.

Using real wind data stored on analog tape drives, this analog computer could simulate the performance of the S-IC stage of a Saturn V rocket up to 3,000 times faster than real-time. The system contained 50 integrators, 50 summers, 350 coefficient potentiometers, 20 quarter square multipliers, and 15 function generators containing additional 70 operational amplifiers. To make simulations with stochastic inputs possible, the system also featured random generators, high- and low-pass filters etc. This system allowed 50 complete flight simulations per second, which made it possible to generate flicker-free displays of solutions on oscilloscopes.[205]

The simulation of a two-stage satellite launch vehicle on an analog computer is described in [Jackson 1960][pp. 261 ff.]. The focus of this study is primarily on possible abort scenarios. The possibility of a piloted rocket flight has been analyzed in a study described in [Waltman 2000][pp. 34 f.]. In these simulations a centrifuge was used to exert gravitational forces of up to 14 G[206] on the pilot while studying his performance in controlling a simulated rocket.

[Kraft et al. 2002][p. 209] remembers that these simulations of the dynamic behavior of rockets and associated systems also resulted in frequent changes to the systems simulated based on results of these simulations:

> *"When we first used the word* dynamic *to describe the simulators, it meant that they acted pretty much like the real thing. But in practice* dynamic *also applied to design because we discovered things in simulations that needed to be changed."*

[205] Cf. [Teuber 1964].
[206] 14 times of earth's gravitation.

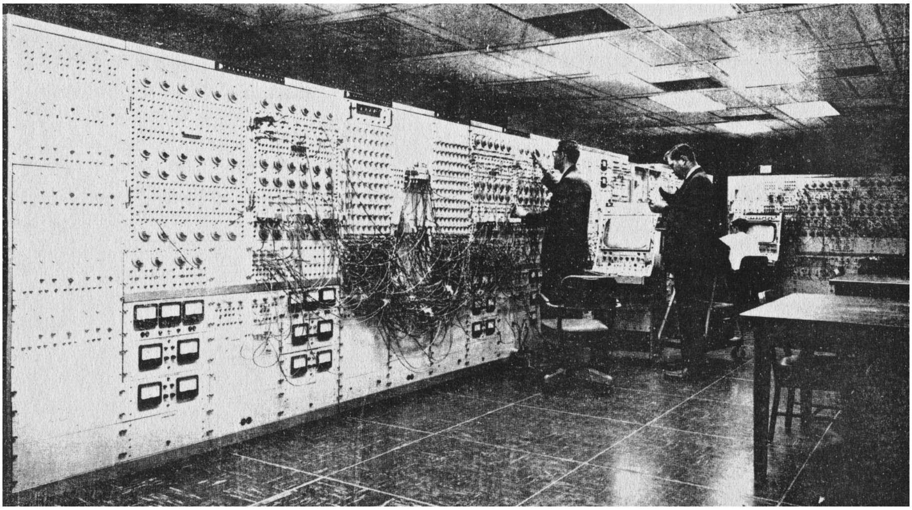


Figure 11.47: The GPS-system, used to simulate the Saturn V rocket's dynamic behavior ([Teuber 1964][p. 26])

11.16.3 Space craft maneuvers

Although the limited precision of analog electronic analog computers places severe constraints on orbital simulations and the like, analog simulations were nevertheless used to obtain at least rough estimates for orbital elements etc. If the orbit radii are restricted to a small range and if the number of circuits is small, 10 e. g., solutions obtained by analog computers are often sufficiently exact to gain valuable insight into space craft maneuvers and the like.[207]

Basic information on such simulations can be found in [Schweizer 1976/2], while [Telefunken/7] describes the computation of the orbital elements for the early passive satellite *Echo-1*. A trajectory optimization for a Mars mission is described in [Giloi 1975][pp. 161 ff.]. A much more complicated example in which the fine-positioning of a geostationary satellite is simulated on an analog computer can be found in [Telefunken/6]. The simulation of the on-board attitude control system of the European communication satellite *Symphony* by means of an analog computer is described in [Reinel 1976]. This simulation required a flight table and a sun and earth simulator to generate proper input signals to the satellite's sensor elements.

A comparative study in which a *reaction wheel* system to be used for attitude control of a satellite was simulated on a hybrid computer as well as on several stored-program digital computers can be found in [Grierson et al.]. While the hybrid computer was able to solve the equations in real-time, the fastest digital computer, a *Cray-1* vector processor, could only achieve a tenth of this speed. Only an *AD-10* stored-program digital computer could compete with the Cray-1. This system was developed by the company *Applied Dynamics* and was optimized for the solution of problems typically solved on analog computers.

[207] See [Telefunken/6][p. 1].

Figure 11.48: Analog computer setup used during the development of the Mercury control stick (author's archive)

11.16.4 Mercury, Gemini, and Apollo

Analog and hybrid computers played a central role in the *Mercury*, *Gemini*, and *Apollo* space flight programs of the United States. Apart from applications like those described above, sophisticated training simulators were built to gain an understanding for the special requirements and problems posed by space vehicles. An example for such a simulator is shown in figure 11.48 which shows the setup used during the development of the control stick used in the Mercury capsules. A complex simulator system for the Mercury capsule and its components had been developed at the Bell Laboratories which was based on an EAI 231R analog computer.[208]

One of the main goals of project Gemini was to work out procedures for docking space craft, a requirement for the following Apollo project. Simple rendezvous simulations on analog computers were performed as early as in the late 1950s.[209] The importance of such simulations cannot be overestimated as [Hansen 2005][p. 248] shows:

> *"Without extensive simulator time, it is doubtful that any astronaut could ever have been truly ready to perform a space rendezvous. 'Rendezvous simulation in Gemini was really quite good',* Armstrong *notes. 'We achieved fifty to sixty rendezvous simulations on the ground, about two-thirds of which were with some sort of emergency.'"*

[208]See [Fagen (ed.) 1978][p. 569].
[209]See [Waltman 2000][p. 107].

The simulation of a pilot controlled rendezvous maneuver is described in detail in [Brissenden et al. 1961]. This study not only focuses on the performance of the pilot during the maneuver, but also aimed at developing suitable display units to be used during the piloted approach. In addition to this, boundary conditions trajectory errors were determined which could make a successful rendezvous maneuver impossible. A highly complex rendezvous simulation system was put into operation at the *TRW Systems Group* in the 1960s. This system featured a TV camera motion unit quite similar to the terrain generator described in section 11.15.5.[210] A similar hybrid simulation system was developed by *McDonnel Aircraft Corporation* and *IBM*, which was also used for reentry simulations during project Gemini.[211]

An interesting in-flight simulator system was developed to train the pilots of the *Lunar Excursion Module*, *LEM* for short. This system, affectionately known as the *flying bedstead* due to its peculiar appearance, was in fact a flight vehicle mimicking the dynamic behavior of the LEM under the constraints of earth's gravitation. This was done by an on-board special purpose analog computer. The resulting system proved very valuable as a research and training device:[212]

> *"The notation of attacking the unique stability and control problems of a machine flying in the absence of an atmosphere, through an entirely different gravity field, 'That was a natural thing for us, because in-flight simulation was our thing at Edwards',* Armstrong *relates. 'We did lots and lots of in-flight simulations, trying to duplicate other vehicles, or duplicate trajectories, making something fly like something else.'"*

11.17 Military applications

Although analog and hybrid computer based flight and rocket simulations were used extensively in a military context, too, there were also problems unique to military applications that were tackled by analog simulation techniques. An interesting example can be found in [EAI Primer 1966][pp. 13 ff.] which describes the simulation of a setback leaf system used as a safety mechanism in a projectile fuse. This mechanism arms the projectile's fuse only after a sufficient acceleration profile has been detected.

Studies of projectile trajectories are described in [Korn & Korn 1956][pp. 110 ff.] and [Johnson 1963][pp. 175 ff.]. Of prime importance is the accurate implementation of aerodynamic properties and forces acting on the projectile which often requires functions of more than one variable, resulting in a rather high amount of necessary computing elements.

The implementation of a simplified naval gunnery problem on an analog computer is covered by [Wass 1955][pp. 57 ff.], while [Demoyer 1980] described a highly complex interactive anti-aircraft gun fire control simulation which was developed in the late 1970s.

[210] See [Bekey et al. 1968][p. 392].
[211] See [Bekey et al. 1968][p. 394].
[212] See [Hansen 2005][p. 314].

11.18 Education

Analog computers were, and in some cases still are, central parts in education. Their main advantage is the basic idea of implementing an analogon without the necessity to worry about numerical integration methods, algorithmic odds and ends in conjunction with the rather unmatched degree of interactivity. Especially the latter allows students to gain insight into the dynamical behavior of complex systems in a very demonstrative way.

More general information about the application of analog computers in education can be found in [Martin 1969] and [Martin 1972]. Educational examples in mathematics are described in [Dornier/8], [Dornier/9], and [Dornier/10]; the simulation of mechanical systems is covered in [Park et al. 1972]; educational applications in process control are the focus of [Nise] and [Medkeff et al. 1955], while [Tabbutt 1967] and [Hamori 1972] contain some examples from chemistry.

11.19 Arts, entertainment and music

Analog computers also experienced rather extensive applications in arts, entertainment and music as the following sections show.

11.19.1 Arts

In the 1940s, Ben Laposky developed the first analog computer solely intended for artistic purposes. In the late 1950s, the computer graphics pioneer John Whitney built an electromechanical analog computer from parts of a M-5 antiaircraft director that was used during World War II and sold for scrap afterwards.[213]

Heinrich Heidersberger[214] was charged with the creation of a large wall painting for the school of engineering in Wolfenbüttel (Germany).[215] His first ideas for this work were based on so-called Lissajous figures named after Jules Antoine Lissajous.[216] These figures are described by

$$x = a\sin(\alpha t + \varphi) \text{ and}$$
$$y = b\cos(\beta t).$$

Based on this idea, Heidersberger built a mechanical analog computer, called *Rhythmograph* which could generate much more complex figures than plain Lissajous figures. Figure 11.49 shows him working on this machine. A typical picture generated by the Rhythmograph can be seen in figure 11.50. Operating the Rhythmograph which exposed photographic paper to a fine spot of light that was deflected by swinging booms and mirrors, was an art in itself. After setting its parameters, influencing the

[213] See [Somers 1980] and `http://en.wikipedia.org/wiki/John_Whitney_(animator)`, retrieved 03/03/2013.

[214] 06/10/1906–07/14/2006

[215] See [Heidersberger][p. 7] and [Hoffmann 2006][p. 35].

[216] 03/04/1822–06/24/1880

Figure 11.49: Heinrich Heidersberger working on the *Rhythmograph* (Archiv Nr. 9179/1, self-portrait, Wolfsburg 1962, reprinted with permission from Benjamin Heidersberger)

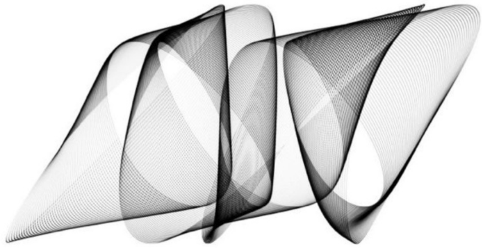


Figure 11.50: Rhythmogramm #229, *Arabeske*, 1950s/60s (reprinted with permission from Benjamin Heidersberger)

periods of the various mechanical oscillators, the Rhythmograph was set into motion by a set of electromagnets and then started to "paint" a picture by this light beam.[217]

An electronic analog computer, built by Franz Raimann, was used in the early 1960s by Herbert W. Franke to generate pictures like that shown in figure 11.51 on an oscilloscope screen. In contrast to Heidersberger's Rhythmograph, this analog computer offered the high degree of interactivity that is characteristic of electronic analog computers which gave the artist more freedom to experiment with the mathematical parameters of an artwork. Figure 11.52 shows Herbert W. Franke working on this analog computer.[218]

In 1979, Benjamin Heidersberger[219] started the development of an electronic analog computer resembling the operation of the Rhythmograph. This machine contains three oscillators which can be controlled separately with respect to frequency, phase and damping. Additional modules implement the necessary control circuitry for repetitive operation, function generators, multipliers, etc. This machine can be seen in figure 11.53 while figure 11.54 gives an impression of the artworks generated with it.

11.19.2 Entertainment

Probably the first special purpose analog computer built for entertainment purposes was the so-called *Cathode-Ray Tube Amusement Device* patented in 1948 and described as follows in [Goldsmith et al. 1948]:

> *"This invention relates to a device with which a game can be played. The game is of such a character that it requires care and skill in playing it or operating the device with which the game is played. Skill can be increased with practice and the exercise of care contributes to success."*

[217] More information about this can be found in [Hoffmann 2006].

[218] See [Hobby 1969][p. 43] and [Deken 1984] for more examples and information.

[219] Son of Heinrich Heidersberger.

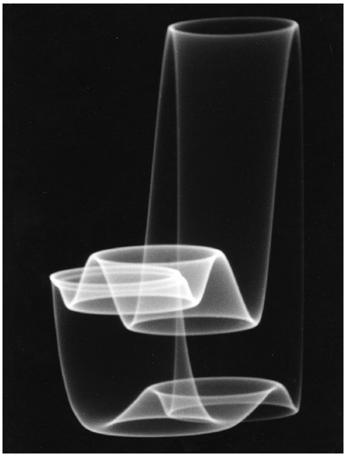


Figure 11.51: "Dance of the electrons" – an artwork of Herbert W. Franke, 1961/1962 (reprinted with permission of the artist)

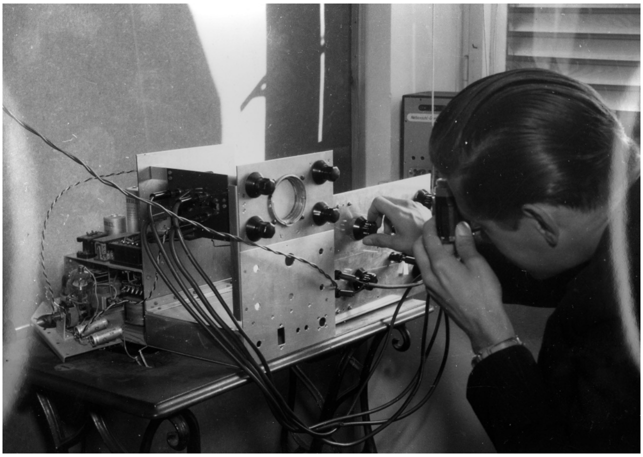


Figure 11.52: Herbert W. Franke with his special purpose analog computer (reprinted with permission of the artist)

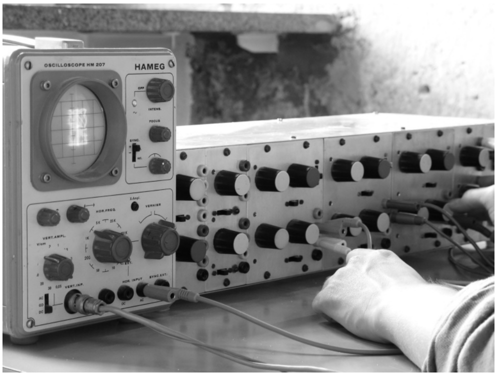


Figure 11.53: Benjamin Heidersberger's analog computer (reprinted with permission)

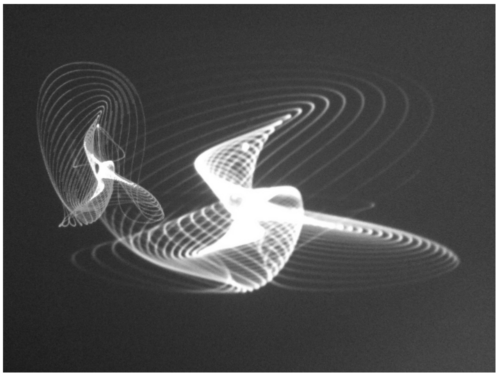


Figure 11.54: Graphic, generated with Benjamin Heidersberger's analog computer (reprinted with permission)

The goal of the game implemented is to hit targets displayed on an oscilloscope screen by firing a beam which can be controlled in its length and angle by the player. To generate a flicker-free picture, this analog computer operates in repetitive mode. Clearly, this device was motivated by analog training devices for gunners. The patent specifications suggest the use of small, adhesive aircraft shaped stickers on the oscilloscope tube to denote the targets.

Ten years later, William Higinbotham[220] independently developed a two-player game aptly called *Tennis for Two* which was demonstrated during an open house presen-

[220] 10/25/1920–11/10/1994

tation of the *Brookhaven National Laboratories* since previous events showed that the public was not too interested in purely static displays as Higinbotham remembers:[221]

> *"I knew from past visitors days that people were not much interested in static exhibits, so for that year I came up with an idea for a hands-on display - a video tennis game. [...] It was wildly successful, and* Higinbotham *could tell from the crowd reaction that he had designed something very special. 'But if I had realized just how significant it was, I would have taken out a patent and the U.S. government would own it!' he said."*

An analog electronic replica of Higinbotham's game, which was originally implemented on a small *Systron Donner* tabletop analog computer, has been developed by the *Museum of Electronic Games & Art*.[222]

A much more complex game based on an electronic analog computer is the "Golf Game Computing System" described by [Russel et al. 1971]. This golf simulator is characterized by the unusual high life-likeness. Therefore effects like the change from laminar to turbulent air flow around the ball, spin, and many more are taken into account for the trajectory computation.

11.19.3 Music

Although analog synthesizers can be justifiably regarded as being special purpose analog computers, not much use has been made of general purpose analog computers in this area. One notable exception of this is the work of the Dutch composer Hans Kulk who started using *Hitachi-240* analog computers in the late 1980 to create music.[223] The high degree of interactivity offered by analog computers is exploited extensively in his work by using special input devices like a three-dimensional manual controller which allows the composer to become part of the composition and sound generation process by changing various parameters of the analog computer setup. This advantage and the possibilities of employing analog computers in musical applications has been summarized by Kulk as follows:[224]

> *"One main conclusion is that the use of an analog [...] computer in analog sound synthesis is very effective, [yet] has been highly ignored by the electronic music community for long, but is slowly gaining interest as the concepts of analog technique become valuable tools in current developments. The 1970s dual Hitachi 240 set-up in this sound synthesis lab will continue to be an inspiring and open system with still many hours of fun to come."*

[221] See [N. N. 2006].

[222] See `http://www.m-e-g-a.org/de/research-education/research/t42-tennis-for-two/`, retrieved 03/03/2013.

[223] This early work was stimulated by [Appleton 1975].

[224] Personal communication to the author.

11.20 Analog computer centers

A typical phenomenon in the heyday of electronic analog and hybrid computers were computer centers set up by the major manufacturers of such machines like Electronic Associates Inc., Beckman etc. These computer centers served a dual purpose: First, potential customers were able to get hands-on experience with the latest machines and could solve practical problems to decide whether an analog computer met their particular needs. Second, these computer centers offered analog computer time on a rental basis for customers who either had no analog computer installation at all or required additional computing power for the solution of some problems.

The first analog computer center in Europe was setup in 1957 in Brussels by EAI.[225] Figure 11.55 shows part of the EAI Pace 96 analog computer which was installed there and contained 96 operational amplifiers, 120 coefficient potentiometers, 20 servo multipliers, 4 quarter square multipliers, five function generators, 2 digital voltmeters, a two-channel X/Y-recorder, two single channel X/Y-recorders, two six-channel recorders and two electronic noise sources.[226] EAI's computer center in Brussels was always equipped with their latest machines. In 1964 a HYDAC 2000 was installed which is shown in figure 6.6.[227]

In 1958, Beckman opened their first analog computer center in Los Angeles which mainly offered computer time for customers. Figure 11.56 shows the EASE 1132 which was the first computer installed there. One year before its inauguration this computer center was announced in [N. N. 1957/6]:

> *"It will be available to business, industry and educational institutions for solution of complex problems relating to aircraft, jet engines, guided missiles and industrial processes."*

[225] This was, in fact, world's third analog computer center – two earlier computer centers were setup in Princeton (New Jersey) and Los Angeles (California).

[226] This model was a precursor of the highly successful model EAI 231R.

[227] See section 6.2.

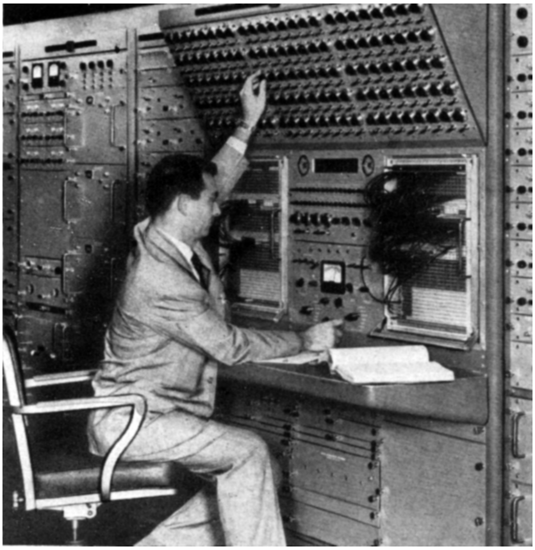


Figure 11.55: EAI Pace 96 analog computer, installed in Europe's first analog computer center in Brussels (see [N. N. 1957/3])

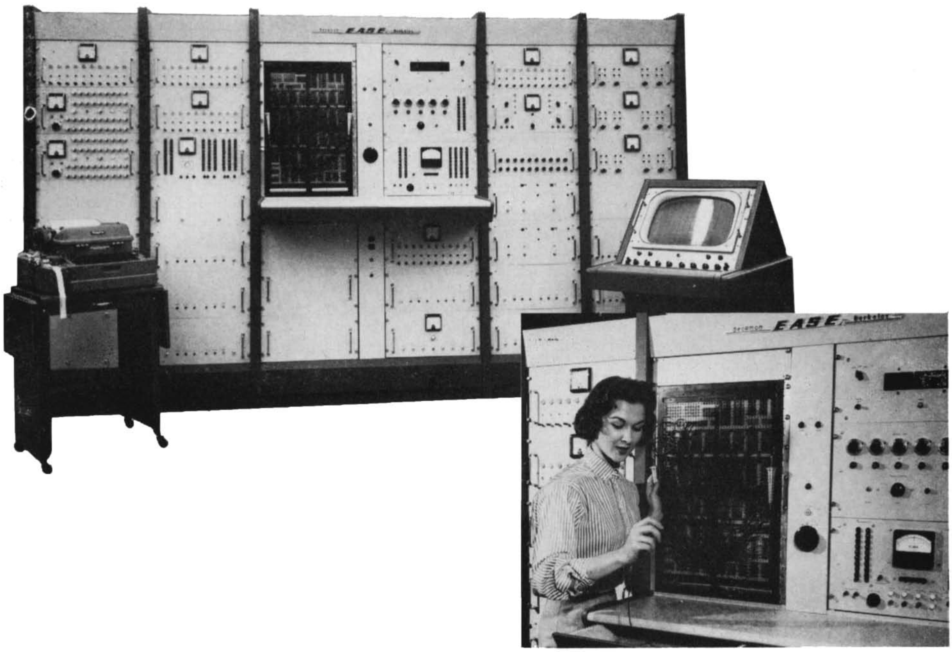


Figure 11.56: EASE 1132 analog computer, installed in the *Beckman/Berkeley Computation Center* which opened on February 28th, 1958 in Los Angeles (see [N. N. 1957/1])

12 Future and chances

Since the 1980s, analog and hybrid computers have become museum pieces and the very idea of analog computing seems to have vanished from the curriculum in universities and the like. This disappearance had many causes: First of all, a paradigm shift occurred in favor of algorithmic approaches instead of analogs. The very idea of using a model, abstract as it may be, to simulate a system which can then by analyzed became obsolete, despite its advantages. Second, stored program digital computers became powerful and cheap enough to replace analog computers in many application areas, starting with flight simulators where the cumbersome configuration process of analog computers became a major obstacle in many installations.[1] Other advantages of stored-program digital computers are their ability to be used in time-sharing mode, the minimized maintenance costs, the possibility to trade in time for complexity etc.

Nevertheless, a renaissance of the idea of analog computing is not only possible but probable since there are some inherent advantages analog computers have in comparison with present day stored-program digital computers. The main advantages of analog computers, be they based on analog electronics or DDAs, amongst others are those:

- Some problems require the solution of differential equations which are quite sensitive to the quadrature algorithm used. Analog electronic analog computers are rather immune to problems like these due to their physical model of integration. In addition to this, analog computers sometimes yield more realistic solutions than other approaches.[2]

- Some problems are rather well suited for analog computers (often based on direct analogies). Introductory examples for this can be found in [Dewdney 1988/1], [Dewdney 1988/2] and [Hoffman 1979].

- Setting up a model is a more natural process than developing a numerical algorithm for the solution of some problem. [Tisdale 1981][p. 3] notes aptly:

  > *"Digital languages [...] obstruct the user's contact with the physical analogy. In other words, the digital programmer becomes preoccupied with the programming task and unfortunately loses sight of the analogy he hopes to create. Hybrid techniques go hand-in-hand with Laplace and Fourier expressions. The analogy is not only established, but the engineer's knowledge and skill grows quickly, giving rise to innovations and breakthroughs."*

[1] See [Tisdale 1981][p. 1].
[2] See [Tisdale 1981][p. 2].

- The degree of parallelism an analog computer exhibits is still unmatched by stored-program digital computers. All computing elements are operating in parallelism without the necessity for complex synchronization schemes and the like.
- However, the main advantage of analog computing is the fact that no stored program is necessary to control the system. There are no memory accesses slowing down the actual work to be done.

The future of the analog computing paradigm will be mainly based on the last two advantages mentioned. In 1975, Wolfgang Giloi predicted, that *"certainly [...] the analog method of hardware parallel processing"* will be preserved. Lee Rubel was even more optimistic as the following quote from a personal note to Jonathan W. Mills of the Indiana University shows:

> *"The future of analog computing is unlimited. As a visionary, I see it eventually displacing digital computing, especially, in the beginning, in partial differential equations and as a model in neurobiology. It will take some decades for this to be done. In the meantime, it is a very rich and challenging field of investigation, although (or maybe because) it is not in the current fashion."*

Mills et al. developed direct analog coprocessors based on ideas of Rubel which can be used with traditional stored-program digital computers. These so-called *Extended Analog Computers, EAC* for short, are similar in their structure to the electrolytic tanks mentioned in section 1.3, although much more modern materials like polymer foils are used.[3] Such coprocessors can speed up simulations and analyses based on partial differential equations significantly.

Glenn Edward Russel Cowan developed a VLSI[4] analog computer that is intended as a mathematical coprocessor for a traditional stored-program digital computer[5] which is shown in figure 12.1.[6] This single-chip analog electronic analog computer features 416 analog functional blocks which can be interconnected dynamically. The advantages of analog computers are clearly demonstrated by this implementation:[7]

> *"This chip is controlled and programmed by a PC via a data acquisition card. This arrangement has been used to solve differential equations with acceptable accuracy, as much as 400× fast than a modern workstation. The utility of a VLSI analog computer has been demonstrated by solving stochastic differential equations, partial differential equations, and ordinary differential equations."*

An analog coprocessor like this can be employed in a variety of ways. In the simplest case, it can act as a traditional analog computer or as the analog part of a hybrid computer. Given the computational power of today's stored-program digital computers,

[3] See [Mills/1], [Mills et al. 2006], [Mills 1995], and [Mills/2] for more information.

[4] Short for *Very Large Scale Integration*.

[5] See [Cowan 2005]. An application for this particular VLSI implementation of an analog computer is described in [Freedman 2011].

[6] See `http://users.encs.concordia.ca/~gcowan/phdresearch.html`, retrieved 03/03/2013.

[7] See [Cowan 2005].

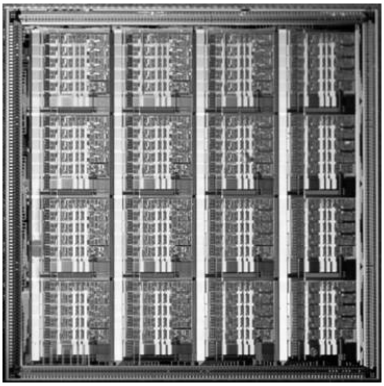


Figure 12.1: VLSI analog computer developed by Glenn Edward Russel Cowan (reprinted with permission)

their role in a hybrid computer configuration should not be restricted to control and function generation tasks. In fact, the analog processor can be used to generate an approximation of the solution for a given task which is then refined using algorithmical methods on the digital part.[8]

Apart from the raw computational power achievable with a hybrid computer setup like this, substantial energy savings might be possible, compared with a purely algorithmic solution of a given problem. This is shown in [Cowan et al. 2009][pp. 13 f.] where different approaches to the solution of a stochastic differential equation are compared with respect to the energy required to reach the solution. The traditional algorithmic approach requires about 400 MFLOPS[9] which corresponds to 1.2–40 J of energy when performed on a typical microprocessor. The same computation performed on a *DSP*[10] requires only 0.04–0.4 J. Using the VLSI analog processor developed by Cowan, the same task can be performed with only 0.008 J – a quite substantial saving. The VLSI analog computer achieves a computing capacitor equivalent to 6.4 GFLOPS while requiring only 300 mW of power to run with all computational units active.

Such low-power high-performance hybrid systems could offer completely new perspectives for mobile computing as well as in the area of traditional high-performance computing.

Another promising approach to apply the analog computing paradigm in future applications are DDAs.[11] DDAs are very well suited as co-processor for stored-program computers since no analog/digital conversion is necessary. In 1969, [Nalley 1969] suggested a small DDA as a dedicated co-processor to accelerate Z-transformations. Other interesting early developments were the *electrically alterable DDA* described by [Hyatt et al. 1968] and a binary floating point DDA described by [Elshoff et al. 1970]. Unfortunately, no commercial viable products resulted from these developments which might have to be attributed mostly to the emerging mini-computers of the

[8] See [Cowan 2005][pp. 182 ff.]. A similar approach is described by [Karplus et al. 1972].
[9] Short for *Million Floating Point Operations Per Second*.
[10] Short for *Digital Signal Processor*.
[11] See section 10.

1970s which offered enough computing power to compete with non-large scale DDAs at lower prices while being more general purpose application oriented.

Nevertheless, using modern technologies like *Field Programmable Gate Arrays*[12] it is now possible to implement DDAs based on either floating point numbers[13] or sufficiently long integers, that would be attractive from a commercial point of view while offering computational power beyond that of general purpose stored-program computers in fields which involve the solution of large sets of coupled (partial) differential equations. Such DDAs could be used to form practical fully digital hybrid computers which would surpass most if not all current high performance computing paradigms with respect to computational power.

Examples for such implementations can be found in [Land] where a DDA implementation on a FPGA is described. This DDA has applied to the solution of differential equations of second degree, the implementation of a FitzHugh-Nagumo model[14] etc.

Maybe no technology of the past has such a great potential for future applications as the analog computing paradigm. This is mostly due to the inherent and unmatched degree of parallelism of analog electronic analog computers and DDAs and the incredibly low power consumption of the former. Quoting the inscription of James Earl Fraser's[15] statue *"Heritage"* at the Federal Triangle, Constitution Ave. & 9th, Washington, DC:

"The Heritage of the Past is the Seed that Brings Forth the Harvest of the Future."

[12] *FPGA* for short.

[13] Using floating point numbers internally, the need for scaling the machine setup could be eliminated or at least reduced significantly. An extended resolution DDA has been proposed by [McGhee et al. 1970].

[14] See section 11.7.5.

[15] 11/04/1876–10/11/1953

Bibliography

[Adler 1968] Helmut Adler, *Elektronische Analogrechner*, VEB Deutscher Verlag der Wissenschaften, Berlin, 1968

[Admiralty 1943] N. N., *Handbook of the Admiralty Fire Control Clock Marks I and I**, Admiralty, S.W.1., Gunnery Branch, 1943

[Akers 1977] Lex A. Akers, "Simulation of semiconductor devices", in *SIMULATION*, August 1977, pp. 33–41

[Albrecht 1968] Peter Albrecht, „Über die Simulation der respiratorischen Arrhythmie auf einem Analogrechner", Technische Mitteilungen AEG-Telefunken, 2. Beiheft Datenverarbeitung, 1968, pp. 13–15

[Albrecht et al.] P. Albrecht, H. Lotz, „Einsatz der hybriden Präzisionsrechenanlage RA 770 zur automatischen Parameteroptimierung nach dem Gradientenverfahren", AEG-Telefunken, AFA 005 0670

[Aldrich et al. 1955] H. P. Aldrich, H. M. Paynter, "First Interim Report – Analytic Studies of Freezing and Thawing of Soils (for the Arctic Construction and Frost Effects Laboratory New England Division, Corps of Engineers)", in [Paynter, ed. 1955][pp. 247–260]

[Ameling 1962/1] W. Ameling, „Die Lösung partieller Differentialgleichungen und ihre Darstellungsmöglichkeiten auf dem elektronischen Analogrechner", in *elektronische datenverarbeitung*, No. 5/1962, pp. 197–215

[Ameling 1962/2] W. Ameling, „Der Einsatz des elektronischen Analogrechners zur Rohrnetzberechnung kompressibler und inkompressibler Stoffströme", in *Elektronische Rechenanlagen*, 4 (1962), No. 3, pp. 109–116

[Ameling 1963] Walter Ameling, *Aufbau und Wirkungsweise elektronischer Analogrechner*, Vieweg Verlag, 1963

[Ameling 1963/2] W. Ameling, „Aufbau und Arbeitsweise des Hybrid-Rechners TRICE", in *Elektronische Rechenanlagen*, 5 (1963), No. 1, pp. 28–41

[Ameling 1964] W. Ameling, „Die Entwicklung verbesserter Ersatzschaltungen mit Hilfe der Differenzmethode", in *Elektronische Rechenanlagen*, 6 (1964), No. 1, pp. 35–41

[Ammon] W. Ammon, *Der elektronische Analogrechner und seine Verwendung in der Industrie*, AEG

[Ammon et al. 1959] W. Ammon, G. Schneider, „Beispiele zur Lösung technischer Probleme mit dem Analogrechner", in *Elektronische Rechenanlagen*, 1 (1959), No. 1, pp. 29–34

[Analog Devices 2006] Analog Devices Inc., *Op Amp Applications Handbook*, Elsevier Newnes, 2006

[Analog Devices 2008] Analog Devices Inc., *Analog Multipliers*, MT-079 Tutorial, 2008

[Anders] Udo Anders, "Early Ideas in the History of Quantum Chemistry", in `http://www.quantum-chemistry-history.com`, as of 24.3.2008

[Ando 1971] Joji Ando, "Simulation Study of Surge Tank System for Hydro Power Plant by Using Hybrid Electronic Computer", in *The Second International JSME Symposium Fluid Machinery and Fluidics*, Tokyo, September 1971, pp. 259–147

[Apalovičová 1979] R. Apalovičová, "Simulation of MOS Transistor Structure on Hybrid Computer Systems", in Tagungsband *SIMULATION OF SYSTEMS '79*, pp. 1013–1019

[Appleton 1975] Appleton, Perera, *The Development and Practice of Electronic Music*, Prentice-Hall, New Jersey, 1975

[Armstrong et al. 1962] N. A. Armstrong, E. C. Holleman, "A Review of In-Flight Simulation Pertinent to Piloted Space Vehicles", AGARD Report 403, July 1962

[Aschoff 1938] V. Aschoff, „Der Sternmodulator als Doppelgegentaktmodulator", in *Telegraphen-Fernsprech-Funk- und Fernsehtechnik*, Bd. 27, No. 10, 1938, p. 379–383

[Ascoli 1947] Guido Ascoli, "Vedute sintetiche sugli strumenti integratori", in *Rend. Sem. Mat. Fis. Milano*, 18:36, 1947

[Ashley et al. 1985] Holt Ashley, Marten Landahl, *Aerodynamics of Wings and Bodies*, Dover Publications, Inc., New York, 1985 (unabridged reprint from 1965)

[Aude et al. 1936] Aude & Reipert, „Gezeitenrechenmaschine", Patentschrift Nr. 682836, Klasse 42m, Gruppe 36, A 78729 IX b/42 m, patentiert vom 6. März 1936 ab

[Bader 1985] Heinz Bader, *Operationsverstärker – Grundlagen und Anwendungen*, Karamanolis Verlag, 1985

[Baker 1978] Bruce Baker, "Martin-Marietta Aerospace Simulation & Test Laboratory", Handout for a talk delivered at a Simulation Councils Conference, San Francisco, 1978, author's archive

[Balaban] Philip Balaban, "HYPAC – A hybrid-computer circuit simulation program", Bell Telephone Laboratories, Holmdel, New Jersey

[Bard 1965] M. Bard, „Schaltung zur automatischen Aufzeichnung von Frequenzgängen mit Analogrechner und Koordinatenschreiber", in *Elektronische Rechenanlagen*, 7 (1965), No. 1, pp. 29–33

[Barth 1976] H.-J. Barth, „Simulation im Maschinenbau zur Festigkeitsermittlung und zur Untersuchung physikalischer Zusammenhänge", in [Schöne 1976/1][pp. 581–602]

[Bassano et al. 1976] J. C. Bassano, Y. Lennon, J. Vignes, "An Identification of Parameters on a Hybrid-Computer", in *Trans. IMACS*, Vol. XVIII, No. 1, Jan. 1976, pp. 3–7

[Bate et al. 1971] Roger R. Bate, Donald D. Mueller, Jerry E. White, *Fundamentals of Astrodynamics*, Dover Publications, Inc., 1971

[Bauer 1953] Louis Bauer, "New Laboratory for Three-Dimensional Guided Missile Simulation", in AIEE-IRE '53 (Western) Proceedings of the February 4–6, 1953, western computer conference, pp. 187–195

[Bauer 1962/1] Louis Bauer, "Aircraft, Autopilot, and Missile Problems", in [Huskey et al. 1962][pp. 5-49–5-64]

[Bauer 1962/2] Louis Bauer, "Partial System Tests and Flight Tables", in [Huskey et al. 1962][pp. 5-64–5-71]

[Baun 1970] P. J. Baun Jr., "Hybrid Computers: Valuable Aids in Transmission Studies", in *Bell Laboratories Record*, June/July 1970, pp. 181–185

[Beck et al. 1958] Robert M. Beck, Max Palevsky, "The DDA", in *Instruments and Automation*, November 1958, pp. 1836–1837

[Bedford et al. 1952] Leslie Herbert Bedford, John Bell, Eric Miles Langham, "Electrical Fire Control Calculating Apparatus", United States Patent 2623692, Dec. 30, 1952

[Behrendt 1965] E. Behrendt, „Wunschautos auf Knopfdruck“, in *hobby – Das Magazin der Technik*, Nr. 6/65, pp. 36–41

[Bekey 1960] George A. Bekey, "Analog Simulation of Nerve Excitation", in [Jackson 1960][pp. 436–444]

[Bekey et al. 1968] George A. Bekey, Walter J. Karplus, *Hybrid Computation*, John Wiley & Sons, Inc., 1968

[Bellini et al. 1975] Armando Bellini, Claudio Cerri, Alessandro de Carli, "A New Approach to the Simulation of Static Converter Drives", in *Annales de l'Association internationale pour le Calcul analogique*, No. 1, Janvier 1975, pp. 3–7

[Bendix] N. N., *Bendix Computer – Digital Differential Analyzer D-12*, Bendix Computer, 5630 Arbor Vitae Street, Los Angeles 45, California

[Bendix 1954] N. N., *Operation Manual – Digital Differential Analyzer – Model D12*, Bendix Computer, 5630 Arbor Vitae Stress, Los Angeles 45, California, Copy No. 2, April 1954

[Benham 1970] R. D. Benham (ed.), "Evaluation of Hybrid Computer Performance on a Cross Section of Scientific Problems", in *AEC Research & Development Report*, BNWL-1278, UC-32, January 1970

[Benham et al. 1973] R. D. Benham, G. R. Taylor, "A PDP 11 Study of the Physiological Simulation Benchmark Experiment", in *DECUS PROCEEDINGS*, Fall 1973, pp. 83–88

[Benyon 1961] P. R. Benyon, "The Australian Guided Weapons Analogue Computer AGWAC", Third International Conference on Analog Computation, Opatija, 4–9 September 1961

[Berkeley et al. 1956] Edmund Callis Berkeley, Lawrence Wainwright, *Computers – Their Operation And Applications*, Reinhold Publishing Corporation, New York, Chapman & Hall, Limited, London, 1956

[Berry et al. 1966] Donald T. Berry, Dwain A. Deets, *Design, development, and utilization of a general purpose airborne simulator*, presented at the 28th Meeting of the AGARD Flight Mechanics Panel, Paris 10–11 May 1966

[Biggs] A. G. Biggs, *Red Duster Acceptance Trials, Scientific Evaluation – A Mathematical Model of the Missile System Suitable for Analogue Computation*, Department of Supply, Australian Defence Scientific Service, Weapons Research Establishment, Report SAD 20, No. 8 J.S.T.U. D3

[Bild der Wissenschaft 1970] N. N., „Das Mathematische Kabinett“, in *Bild der Wissenschaft*, Mai 1970

[Billet 1965] Reinhard Billet, *Verdampfertechnik*, Bibliographisches Institut Mannheim, B.I.-Wissenschaftsverlag, 1965

[Bilstein 2003] Roger E. Bilstein, *Stages to Saturn – A Technological History of the Apollo/Saturn Launch Vehicles*, University Press of Florida, 2003

[Black 1937] Harold Stephen Black, "Wave Translation System", United States Patent 2102671, Dec. 21, 1937

[Blum et al.] Alfons Blum, Manfred Glesner, "Macromodeling Procedures for the Computer Simulation of Power Electronics Circuits", F.B. 12.2 Elektrotechnik, Universität des Saarlandes

[Boghosian et al. 1950] W. H. Boghosian, S. Darlington, H. G. Och, "Artillery Director", United States Patent 2493183, January 3, 1950

[Bohling 1970] Dorothea M. Bohling, Lawrence A. O'Neill, "An Interactive Computer Approach to Tolerance Analysis", in *IEEE Transactions on Computers*, Vol. C-19, No. 1, January 1970, pp. 10–16

[Borchardt] Inge Borchardt, „Demonstrationsbeispiel: Elektrisch geladenes Teilchen im Magnetfeld“, AEG-Telefunken, ADB 007

[Borchardt 1965] I. Borchardt, „Berechnung von Teilchenbahnen in einem magnetischen Horn am Analogrechner“, DESY – H 11, Hamburg, 12/17/1965

[Borchardt et al.] I. Borchardt, P. Maier, F. Hultschig, „Simulation von Strahlführungssystemen auf dem hybriden Rechnersystem HRS 860“, AEG Telefunken, Datenverarbeitung

[Borchardt et al. 1965] I. Borchardt, G. Ripken, „Zur Berechnung der Teilchenbahnen in einem Sextupolfeld“, Hamburg, DESY – H 5, Hamburg, 05/24/1965

[Borchardt 1966] I. Borchardt, „Strahloptische Gleichungen und ihre Verwendung im Analogrechenprogramm“, DESY-Strahloptik, 04/01/1966

[Borchardt et al. 1969] I. Borchardt, P. Zajíček, „Wechselrichter-Schutzprobleme“, Interner Bericht, DESY K-69/3, November 1969

[Borchardt et al. 1977] I. Borchardt, M. Levy, J. Maass, „Untersuchungen über die Regelung des 200 Hz Wechselrichters für das flat-top-System des Synchrotrons mit der hybriden Rechenanlage HRS 860“, Interner Bericht, DESY R1-77/01, June 1977

[Borsei et al.] A. Borsei, G. Estrin, “An Analog Computer Study of the Dynamic Behaviour of Stressed Thin Ferromagnetic Films”, Technical Report No. 62-37, University of California, Los Angeles

[Brennan et al. 1964] R. D. Brennan, H. Sano, “PACTOLUS – A digital simulator program for the IBM 1620”, in *AFIPS Conference Proceedings, Fall Joint Computer Conference 26*, October 1964, pp. 299–312

[Brey 1958] R. N. Brey Jr., “Reactor Control”, in *Instruments and Automation*, Vol. 31, April 1958, pp. 630–635

[Brissenden et al. 1961] Roy F. Brissenden, Bert B. Burton, Edwin C. Foudriat, James B. Whitten, *Analog Simulation of a Pilot-Controlled Rendezvous*, Technical Note D-747, National Aeronautics and Space Administration Washington, April 1961

[Britain 2011] James E. Britain, “Electrical Engineering Hall of Fame: Harold S. Black”, in *Proceedings of the IEEE*, Vol. 99, No. 2, February 2011, pp. 351–353

[Bromley 1984] Allan G. Bromley, *British Mechanical Gunnery Computers of World War II*, Technical Report 223, January 1984

[Brouwer 2007] Jens Brouwer, „Das FitzHugh-Nagumo Modell einer Nervenzelle“, Universität Hamburg, Department Mathematik, 20.8.2007

[Brown 1969] Frank M. Brown, “Comment on Canonical Programming of Nonlinear and Time-Varying Differential-Equations”, in *IEEE Transactions on Computers*, Vol. C-18, No. 6, June 1969, p. 566

[Bückner 1950] Hans Bückner, „Ein neuer Typ einer Integrieranlage zur Behandlung von Differentialgleichungen“, in *Archiv der Mathematik*, 1949/50, Volume 7, Issue 6, pp. 424–433

[Bureau of Ordnance Publication 1944] Bureau of Ordnance Publication (ed.), *Torpedo Data Computer, Mark 3, Mods. 5 to 12 inclusive*, June, 1944

[Bush 1912] Vannevar Bush, “Profile Tracer”, United States Patent 1048649, Dec. 31, 1912

[Cajori 1994] Florian Cajori, *A History of the Logarithmic Slide Rule and Allied Instruments*, Astragal Press, 1994

[Caldwell et al. 1955] R. R. Caldwell, V. C. Rideout, “A Differential-Analyzer Study of Certain Nonlinearly Damped Servomechanisms”, in [Paynter, ed. 1955][pp. 193–198]

[Cameron et al. 1961] W. D. Cameron, R. E. Tiller, “Analog Program in Reactor Speed of Control Systems”, Technical Report, HW-69940, 1961 Jun. 14

[Campeau 1969] Joseph O. Campeau, “The Block-Oriented Computer”, in *IEEE Transactions on Computers*, Vol. C-18, No. 8, August 1969, pp. 706–718

[Care 2006] Charles Care, "A Chronology of Analogue Computing", in *The Rutherford Journal*, Volume 2, 2006–2007

[Carlson et al. 1967] Alan Carlson, George Hannauer, Thomas Carey, Peter J. Holsberg (eds.), *Handbook of Analog Computation*, 2nd edition, Electronic Associates, Inc., Princeton, New Jersey, 1967

[Carlson et al. 1968] Alan M. Carlson, "Hybrid Simulation of an Exchanger/Reactor Control System", in *EAI applications reference library*, 6.4.20h, October 1968

[Celmer et al. 1970] John Celmer, Mary Rouland, *Automatic Analog Computer Scaling Using Digital Optimization Techniques*, National Aeronautics and Space Administration, Washington, D. C., March 1970

[Ceruzzi 1989] Paul E. Ceruzzi, *Beyond the Limits – Flight Enters the Computer Age*, The MIT Press, 1989

[Chan 1969] Shu-Kwan Chan, "The Serial Solution of the Diffusion Equation Using Nonstandard Hybrid Techniques", in *IEEE Transactions on Computers*, Vol. C-18, No. 9, September 1969, pp. 786–799

[Chance et al. 1947] Britton Chance, J. N. Thurston, P. L. Richman, "Some Designs and Applications for Packaged Amplifiers Using Subminiature Tubes", in *The Review of Scientific Instruments*, Volume 18, Number 9, September, 1947, pp. 610–616

[Chance et al. 1949] Britton Chance, Vernon Hughes, Edward F. MacNichol, David Sayre, Frederic C. Williams (ed.), *Waveforms*, McGraw-Hill Book Company, Inc., 1949

[Chandler et al. 1961] W. G. Chandler, P. L. Dandeno, A. F. Glimn, "Short-range economic operation of a combined thermal and hydroelectric power system", in *AIEE Transactions*, Part 3, Vol. 80, May 1961, pp. 1219–1228

[Charlesworth et al. 1974] A. S. Charlesworth, J. R. Fletcher, *Systematic Analogue Computer Programming*, Pitman Publishing, Second Edition, 1974

[Cheng] Shang-I Cheng, "Analog Simulation and Polymerization Kinetics", author's archive

[Clymer 1993] A. Ben Clymer, "The Mechanical Analog Computers of Hannibal Ford and William Newell", in *IEEE Annals of the History of Computing*, Vol. 15, No. 2, 1993, pp. 19–34

[Clynes 1960] Manfred Clynes, "Respiratory control of heart rate: laws derived from analog computer simulation", in *IRE Transaction on Medical Electronics*, Jan. 1960, pp. 2–14

[Coffin 1882] J. Coffin, "Averageometer or Instrument for Measuring the Average Breadth of Irregular Planes", United States Patent 258993, June 6, 1882

[Cohen 1971] E. M. Cohen, "Hybrid Simulation-Aided Design of a Pneumatic Relay", Presented at the Regional Meeting, Eastern Simulation Councils Foxboro, Mass., September 23, 1971

[Costakis 1974] William G. Costakis, *Analog Computer Implementation of Four Instantaneous Distortion Indices*, National Aeronautics and Space Administration, Washington, D. C., March 1974

[Cowan 2005] Glenn Edward Russell Cowan, *A VLSI Analog Computer / Math Coprocessor for a Digital Computer*, Columbia University, 2005

[Cowan et al. 2009] Glenn Edward Russell Cowan, Y. Tsividis, *Analog and Digital Continuous-Time Computation and Signal Processing*, CMOSET 2009

[Dagbjartsson et al. 1976] S. Dagbjartsson, D. Emendörfer, „Simulation in der Kernenergietechnik“, in [Schöne 1976/1][pp. 298–340]

[Hoffman 1979] Dale T. Hoffman, "Smart Soap Bubbles Can Do Calculus", in *The Mathematics Teacher*, Vol. 72, No. 5, Mai 1979, pp. 377–385

[Davis et al. 1974] Frank T. Davis, Armando B. Corripio, "Dynamic Simulation of Variable Speed Centrifugal Compressors", ISA CPD 74105, 1974

[Day 1959] Richard E. Day, "Training Considerations During the X-15 Development", paper presented to the Training Advisory Committee of the National Security Industrial Association, Los Angeles, California, November 17, 1959

[Deken 1984] Joseph Deken, *Computerbilder, Kreativität und Technik*, Birkhäuser Verlag, 1984

[Dehmel 1949] Richard C. Dehmel, "Aircraft Trainer for Aerial Gunners", United States Patent 2471315, May 24, 1949

[Dehmel 1954] Richard C. Dehmel, "Flight Training Apparatus for Computing Flight Conditions and Simulating Reaction of Forces on Pilots", United States Patent 2687580, Aug. 31, 1954

[Department of the Army 1956] Department of the Army, *NIKE I SYSTEMS, NIKE I COMPUTER (U)*, Department of the Army Technical Manual TM 9-5000-3, April 1956

[Demoyer 1980] R. DeMoyer Jr., "Interactive Anti-Aircraft Gun Fire Control Simulation: An Introduction to Hybrid Computation", in *EAI Product Information Bulletin*, February 18, 1980, Bulletin No. 023

[Dewdney 1988/1] A. K. Dewdney, „Rechnen mit Spaghetti – Wie der Spaghetti-Computer und andere kuriose Analoggeräte Probleme im Handumdrehen lösen, an welchen selbst die größten Digitalrechner scheitern.“, in *Computer-Kurzweil*, Spektrum der Wissenschaft, 1988, pp. 198–203

[Dewdney 1988/2] A. K. Dewdney, „Kuriose Analog-Computer – Eine neue Kollektion von Analogrechnern für Heimwerker und eine vertiefte Diskussion ihrer Stärken und Schwächen im Vergleich zu Digitalrechnern.“, in *Computer-Kurzweil*, Spektrum der Wissenschaft, 1988, pp. 204–210

[Dhen 1960] Walter Dhen, „Entwurf und Aufbau eines repetierenden Analogrechners unter besonderer Berücksichtigung der Zusammenhänge zwischen den Rechenfehlern und den Regelkreiseigenschaften in elektronischen

Rechengeräten“, in *Nachrichtentechnische Fachberichte*, Band 17, Vieweg & Sohn, Braunschweig, 1960

[Dick et al. 1967] D. E. Dick, H. J. Wertz, “Analog and Digital Computation of Fourier Series and Integrals”, in *IEEE Transactions on Electronic Computers*, Vol. EC-16, No. 1, February 1967, pp. 8–13

[Dodd 1969] K. N. Dodd, *Analogue Computers*, The English Universities Press Ltd., 1969

[Dornier/1] N. N., „Beschleunigung eines PKW mit automatischem Getriebe (vereinfacht)“, Dornier

[Dornier/2] N. N., „Simulation reaktionskinetischer Probleme auf dem Analogrechner“, Dornier

[Dornier/3] N. N., „Der Analogrechner als Hilfsmittel bei der Untersuchung des Schwingungsverhaltens von Mehrmassensystemen“, Dornier

[Dornier/4] N. N., „Feder-Masse-System mit trockener Reibung“, Dornier

[Dornier/5] N. N., „Kurbeltrieb“, Dornier

[Dornier/6] N. N., „Ermittlung der Scherarbeit bei Materialprüfungen mit Hilfe eines Analogrechners“, Dornier

[Dornier/7] N. N., „Rundlaufprüfungen von Rädern mit Hilfe eines Analogrechners“, Dornier

[Dornier/8] N. N., „Anwendung von Analogrechnern bei der Ausbildung im Rahmen der Oberschulen – Kurvendiskussion“, Dornier

[Dornier/9] N. N., „Anwendung von Analogrechnern bei der Ausbildung im Rahmen der Oberschulen – Approximation von trigonometrischen Funktionen durch Reihenentwicklungen“, Dornier

[Dornier/10] N. N., „Anwendung von Analogrechnern bei der Ausbildung im Rahmen der Oberschulen – Parameterdarstellung von geschlossenen Kurven“, Dornier

[Dungan 2005] Tracy Dwayne Dungan, *V-2 – A Combat History of the First Ballistic Missile*, Westholme Publishing, 2005

[EAI PACE 231R] EAI, *PACE 231R analog computer*, Electronic Associates, Inc., Long Branch, New Jersey, Bulletin No. AC 6007

[EAI 690] N. N., *690 Hybrid Computing System – Reference Handbook*, Electronic Associates, Inc., Long Branch, New Jersey

[EAI Primer 1966] N. N., *Primer on Analog Computations and Examples for EAI-180 Series of Computers*, Electronic Associates, Inc., Bulletin No. 957051, 1966

[EAI TR-10] N. N., *PACE TR-10 Transistorized Analog Computer – operator’s handbook*, Electronic Associates, Inc., Long Branch, New Jersey

[Eck 1954] Bruno Eck, *Technische Strömungslehre*, Springer-Verlag, Berlin/Göttingen/Heidelberg, 1954, 4. verbesserte Auflage

[Eckdahl et al. 2003] Donald E. Eckdahl, Irving S. Reed, Hrant Harold Sarkissian, "West Coast Contributions to the Development of the General-Purpose Computer: Building Maddida and the Founding of Computer Research Corporation", in *IEEE Annals*, January-March 2003 (vol. 25 no. 1), pp. 4–33

[Eggers 1954] K. Eggers, *Über die Integrieranlage 'Integromat' des Instituts für angewandte Mathematik der Universität Hamburg*, Schriftenreihe Schiffbau, 3, Februar 1954

[Ehricke 1960] Krafft A. Ehricke, *Space Flight – Environment and Celestial Mechanics*, D. Van Nostrand Company, Inc., Princeton, N. J., 1960

[Elshoff et al. 1970] J. L. Elshoff, P. T. Hulina, "The binary floating point digital differential analyzer", in AFIPS '70 (Fall) Proceedings of the November 17–19, 1970, fall joint computer conference, pp. 369–376

[Enns et al.] Mark Enns, Theo C. Giras, Norman R. Carlson, "Load Flows by Hybrid Computation for Power System Operation", IEEE, Paper No. 71 C 26-PWR-XII-A

[Ernst 1960] Dietrich Ernst, *Elektronische Analogrechner – Wirkungsweise und Anwendung*, R. Oldenbourg Verlag München, 1960

[Eterman 1960] I. I. Eterman, *Analogue Computers*, Pergamon Press, 1960

[Evans 1959] William T. Evans, "Analog Computer to Determine Seismic Weathering Time Corrections", United States Patent 2884194, April 28, 1959

[Everyday Science and Mechanics 1932] N. N., "Mechanical SUPER-BRAINS – Calculations in Higher Mathematics Performed by Complex Machinery", in *Everyday Science and Mechanics*, June 1932, pp. 625, 678

[Eyman et al. 1976] Earl D. Eyman, Yevgeny V. Kolchev, "Universal Analog Computer Model for Three-Phase Controlled Rectifier Bridges", in *IEEE PES Winter Meeting & Tesla Symposium*, New York, N.Y., January 25–30, 1976, pp. 1136–1144

[Fagen (ed.) 1978] M. D. Fagen (ed.), *A History of Engineering and Science in the Bell System – National Service in War and Peace (1925–1975)*, Bell Telephone Laboratories, Inc., First Printing, 1978

[Feilmeier 1974] Manfred Feilmeier, *Hybridrechnen*, International Series of Numerical Mathematics, Vol. 2, Birkhäuser Verlag, 1974

[Fifer 1961] Stanley Fifer, "Analogue Computation – Theory, Techniques and Applications", Vol. III, McGraw-Hill Book Company, Inc., 1961

[Foote et al. 2007] Robert L. Foote, Ed Sandifer, "Area Without Integration: Make Your Own Planimeter", in *Hands on History – A Resource for Teaching Mathematics*, Amy Shell-Gellash, ed., The Mathematical Association of America (Incorporated), 2007

[Forbes 1957] Georges F. Forbes, *Digital Differential Analyzers*, Fourth Edition, 1957

[Forbes 1972] George Forbes, "The simulation of partial differential equations on the digital differential analyzer", in ACM '72 Proceedings of the ACM annual conference, Volume 2, pp. 860–866

[Fornel et al. 1981] B. de Fornel, H. C. Hapiot, J. M. Farines, J. Hector, "Hybrid Simulation of a Current Fed Asynchronous Machine", in *Mathematics and Computers in Simulation*, XXIII (1981), pp. 253–261

[Fortune 1952] N. N., "The Moniac – 'Economics in thirty fascinating minutes'", in *Fortune*, March 1952, pp. 101–102

[Fox et al. 1969] Harry W. Fox, Ronald J. Blaha, *An Analog Computer Study of the Low-Frequency Dynamics of two Nuclear-Rocket Cold-Flow Engine Systems*, National Aeronautics and Space Administration, Washington, D. C., July 1969

[Freedman 2011] Immanuel Freedman, "System for forecasting outcomes of clinical trials", United States Patent US 2011/0238317 A1, Sep. 29, 2011

[Freeth 2008] Tony Freeth, *The Antikythera Mechanism – Decoding an Ancient Greek Mystery*, Whipple Museum of the History of Science, University of Cambridge, 2008

[Fremerey] Johan K. Fremerey, „Permanentmagnetische Lager", Forschungszentrum Jülich, Institut für Grenzflächenforschung und Vakuumphysik, 0B30-A30

[Fremerey 1978] Johan K. Fremerey, Karl Boden, "Active permanent magnet suspensions for scientific instruments", in *Journal of Physics E – Scientific Instruments*, February 1978, Vol. 11, No. 2, pp. 106–113

[Friedman 2008] Norman Friedman, *Naval Firepower – Battleship Guns and Gunnery in the Dreadnought Era*, Seaforth Publishing, 2008

[Frisch 1968] Willi Frisch, *Stabilitätsprobleme bei dampfgekühlten schnellen Reaktoren*, Dissertation, Universität Karlsruhe, 1968

[Frisch et al. 1969] Willi Frisch, G. Wilhelmi, *Dynamische Simulatoren in der Reaktorentwicklung – ein Vergleich*, Gesellschaft für Kernforschung mbH., Karlsruhe, Januar 1969, 8/69-1

[Frisch 1971] Willi Frisch, *Analogrechnen in der Kernreaktorrechnik*, G. Braun Karlsruhe, 1971

[GAP/R Evolution] N. N., *A Brief History of a Computor-Builder in Terms of Ideas & Instrument – Evolution of a* Model *Kit of Tools*, George A. Philbrick Researches, Inc.

[GAP/R K2W] N. N., *Model K2-W Operational Amplifier*, George A. Philbrick Researches, Inc.

[GAP/R 1959] N. N., *Squaring, Rooting, and the Douglas Quadratron*, GAP/R Application Brief, George A. Philbrick Researches, Inc., No. D1, December 1, 1959

[Gerwin 1958] Robert Gerwin, „Atom-Strom für deutsche Städte", in *Hobby – Das Magazin der Technik*, Nr. 9, September 1958

[Gilbert 1970] Leonard J. Gilbert, *Analog Computer Simulation of a Parasitically Loaded Rotating Electrical Power Generating System*, National Aeronautics and Space Administration, Washington, D. C., November 1970

[Giloi 1960] W. Giloi, „Behandlung von Transformatorproblemen mit dem Analogrechner", in *Telefunken Zeitung*, Vol. 33 (1960), No. 129, p. 50

[Giloi 1961] W. Giloi, „Ein Verfahren zur Berechnung von Optimalfiltern auf dem Analogrechner“, in *Elektronische Rechenanlagen*, 3 (1961), No. 2, pp. 61–65

[Giloi 1962] W. Giloi, „Über die Behandlung elektrischer und mechanischer Netzwerke auf dem Analogrechner“, in *Elektronische Rechenanlagen*, 4 (1962), No. 1, pp. 27–35

[Giloi 1963] W. Giloi, „Hybride Rechenanlagen – ein neues Konzept“, in *Elektronische Rechenanlagen*, 5 (1963), No. 6, pp. 262–269

[Giloi et al. 1963] Wolfgang Giloi, Rudolf Lauber, *Analogrechnen*, Springer-Verlag, 1963

[Giloi 1975] W. K. Giloi, *Principles of Continuous System Simulation*, B. G. Teubner, Stuttgart, 1975

[Giloi et al.] W. Giloi, R. Herschel, *Rechenanleitung für Analogrechner*, Telefunken-Fachbuch, AFB 001

[Gleiser 1980] Molly Gleiser, "Analog Inventor", in *DATAMATION*, October 1980, pp. 141–143

[Glumineau et al. 1982] A. Glumineau, R. Mezencev, "Hybrid Simulation of a Tanker Moored at a Single Point Subjected to Effects of Wind, Current and Waves", in *10th IMACS World Congress on System Simulation and Scientific Computation*, 1982, pp. 98–100

[Goldberg et al. 1954] Edwin A. Goldberg, Jules Lehmann, "Stabilized Direct Current Amplifier", United States Patent 2684999, July 27, 1954

[Goldman 1965] Mark W. Goldman, "Design of a High Speed DDA", in AFIPS '65 (Fall, part I) Proceedings of the November 20–December 1, 1965, fall joint computer conference, part I, pp. 929–949

[Goldsmith et al. 1948] Thomas T. Goldsmith, Estle Ray Mann, "Cathode-Ray Tube Amusement Device", United States Patent 2455992, Dec. 14, 1948

[Golten et al. 1967] J. W. Golten, David Rees, "The Use of Hybrid Computing in the Analysis of Steel Rolling", in *IEEE Transactions on Electronic Computers*, Vol. EC-16, No. 6, December 1967, pp. 717–722

[Govindarao 1975] V. M. H. Govindarao, "Analog Simulation of an Isothermal Semi-batch Bubble-Column Slurry Reactor", in *Annales de l'Association internationale pour le Calcul analogique*, No. 2, Avril 1975, pp. 69–78

[Graefe et al. 1974] P. W. U. Graefe, Leo K. Nenonen, "Simulation of combined discrete and continuous systems on a hybrid computer", in *SIMULATION*, May 1974, pp. 129–137

[Gray 1948] John W. Gray, "Direct-Coupled Amplifiers", in *Vacuum Tube Amplifiers*, Massachusetts Institute of Technology, Radiation Laboratory Series, pp. 409–495

[Gray et al. 1955] John W. Gray, Duncan MacRae, "Bombing Computer", United States Patent 2711856, June 28, 1955

[Gray 1958] H. J. Gray Jr., "Digital Computer Solution of Differential Equations in Real Time", in IRE-ACM-AIEE '58 (Western) Proceedings of the May 6–8, 1958, western joint computer conference: contrasts in computers, pp. 87–92

[Grierson et al.] W. O. Grierson, D. B. Lipski, N. O. Tiffany, "Simulation Tools: Where can we go?", author's archive

[Gröbner 1961] W. Gröbner, „Steuerungsprobleme mit Optimalbedingung", in *mtw – Zeitschrift für moderne Rechentechnik und Automation*, 2/61, pp. 62–64

[Gundlach 1955] F. W. Gundlach, "A new electronic-beam multiplier with an electrostatic hyperbolic field", in *Actes. J. Internat. Calcul Analog Analogique*, Brüssel, 1955, pp. 101–103

[Habermehl et al. 1969] A. Habermehl, E. H. Graul, *Analog Computer Investigations with a Mathematical Model for the Regulation Mechanism of Calcium Metabolism*, National Aeronautics and Space Administration, Washington, D. C., April 1969

[Haeberlin et al.2011] Barbara Haeberlin, Stefan Drechsler, *Wie funktioniert ein Kegel-Reibrad-Planimeter?*, RST 22, Worms, 22.10.2011, `http://www.rechenschieber.org/haedre2011.pdf` (retrieved 11/22/2012)

[Hall et al. 1969] Carroll R. Hall, Stephen J. Kahne, "Automated Scaling for Hybrid Computers", in *IEEE Transactions on Computers*, Vol. C-18, No. 5, May 1969, pp. 416–423

[Hall 1996] Eldon C. Hall, *Journey to the Moon: The History of the Apollo Guidance Computer*, American Institute of Aeronautics and Astronautics, Inc., 1996

[Hamori 1972] Eugene Hamori, "Use of the Analog Computer in Teaching Relaxation Kinetics", in *Journal of Chemical Education*, Volume 49, Number 1, January 1972, pp. 39–43

[Hannauer 1968] George Hannauer, *Stored Program Concept for Analog Computers*, final report, EAI project 320009, NASA order NAS8-21228

[Hannigan] Frank J. Hannigan, "Hybrid Computer Simulation of Fluidic Devices", Electronic Associates, Inc.

[Hansen et al. 1959] P. D. Hansen, J. H. Eaton, "Control and Dynamics Performance of a Sodium Cooled Reactor Power System", MICROTECH RESEARCH COMPANY, Massachusetts, Report No. 171, December 28., 1959

[Hansen 2005] James R. Hansen, *First Man – The Life of Neil A. Armstrong*, Simon & Schuster UK, 2005

[Hart et al. 1967] Clint E. Hart, Dale J. Arpasi, *Frequency Response and Transfer Functions of a Nuclear Rocket Engine System obtained from Analog Computer Simulation*, National Aeronautics and Space Administration, Washington, D. C., May 1967

[Haug 1960] Albert Haug, „Funktionsgeneratoren und Funktionsspeicher der Formen $y = f(x)$ und $z = f(x,y)$", in *Nachrichtentechnische Fachberichte*, Band 17, Vieweg & Sohn, Braunschweig, 1960

[Hedin et al.] Ronald A. Hedin, Kenneth W. Priest, "Progress in Hybrid Simulation of Power Systems", author's archive

[Heideprim 1976] J. Heideprim, „Modelle und Simulation von Produktionsprozessen in der Stahlindustrie", in [Schöne 1976/1][pp. 206–254]

[Heidersberger] Heinrich Heidersberger, *Rhythmogramme*, CARGO Verlag

[Heinhold 1959] J. Heinhold, „Konforme Abbildung mittels elektronischer Analogrechner", in *mtw – Zeitschrift für moderne Rechentechnik und Automation*, 1/59, pp. 44–48

[Heinhold et al.] Josef Heinhold, Ulrich Kulisch, *Analogrechnen – Eine Einführung*, Bibliographisches Institut Mannheim/Wien/Zürich, B.I.-Wissenschaftsverlag, 1969

[Heller et al. 1976] Rainer Heller, Carl W. Malstrom, E. Harry Law, "Hybrid Simulation of Rail Vehicle Lateral Dynamics", presented at *The 1976 Summer Computer Simulation Conference*, Washington, D. C., July 12–14, 1976

[Henrici 1894] O. Henrici, "Report on Planimeters", in *Report of the Sixty-Fourth Meeting of the British Association for the Advancement of Science*, London: John Murray, Albemarle Street, 1894

[Herschel 1961] R. Herschel, „Automatische Optimisatoren", in *Elektronische Rechenanlagen*, 3 (1961), No. 1, pp. 30–36

[Herschel 1962] R. Herschel, „Analogrechenschaltungen für die Entwicklungskoeffizienten nach Orthogonalfunktionen", in *Elektronische Rechenanlagen*, 3 (1961), No. 5, pp. 212–217

[Herschel 1966] R. Herschel, „Zur Programmierung von hybriden Rechenanlagen in ALGOL", in *Telefunken Zeitung*, Vol. 39 (1966), No. 1, pp. 100–109

[Higgins et al. 1982] W. H. C. Higgins, B. D. Holbrook, J. W. Emling, "Defense Research at Bell Laboratories", in *Annals of the History of Computing*, Volume 4, Number 3, July 1982, pp. 218–236

[Hitachi 200X] N. N., *Hitachi Analog Hybrid Computer – Hitachi-200X*, Hitachi Electronics, Ltd.

[Hitachi 505E] N. N., "HITACHI 505E analog/hybrid computer", Hitachi Electronics

[Hitachi 1967] N. N., "Double Integral – Calculation of Volume of Cone", in *Technical Information Series No. 3*, Hitachi Electronics, Ltd., 1967

[Hitachi 1968] N. N., "Analysis of Rolling Theory by Analog Computer – Karman's Differential Equation", in *Technical Information Series No. 8*, Hitachi Electronics, Ltd., 1968

[Hitachi 1969] N. N., "Introduction to Simulator – Part 1", in *Technical Information Series No. 9*, Hitachi Electronics, Ltd., 1969

[Hobby 1969] N. N., „Psychedelic – Explosion der Farben", in *hobby – Das Magazin der Technik*, Nr. 15/69, p. 36–45

[Hoelzer 1946] Helmut Hoelzer, *Anwendung elektrischer Netzwerke zur Lösung von Differentialgleichungen*, Dissertation TH Darmstadt, 1946

[Hoelzer 1992] Helmut Hoelzer, „50 Jahre Analogcomputer“, Rede anlässlich des fünfzigsten Jubiläums des elektronischen Analogrechners im Senatssaal in Berlin, 05/12/1992, Manuskript aus dem Archiv der Familie Hoelzer-Beck

[Hoffmann 2006] Justin Hoffmann / Kunstverein Wolfsburg (Hg.), *Der Traum von der Zeichenmaschine – Heinrich Heidersbergers Rhythmogramme und die Computergrafik ihrer Zeit*, Kunstverein Wolfsburg, Kataloge #1/2006

[Holst 1982] Per A. Holst, "George A. Philbrick and Polyphemus – The First Electronic Training Simulator", in *Annals of the History of Computing*, Volume 4, Number 2, April 1982

[Holst 1996] Per A. Holst, "Svein Rosseland and the Oslo Analyzer", in *IEEE Annals*, Winter 1996, Vol. 18, No. 4, pp. 16–26

[Horling et al.] James E. Horling, Esmat Mahmout, Frank J. Hannigan, "Hardware-in-the-Loop Simulation for Evaluating Turbine Engine Fuel System Components", author's archive

[Howe et al. 1953] R. M. Howe, V. S. Haneman, "The solution of partial differential equations by difference methods using the electronic differential analyzer", in AIEE-IRE '53 (Western) Proceedings of the February 4–6, 1953, western computer conference, pp. 208–226

[Howe 1962] R. M. Howe, "Solution of Partial Differential Equations", in [Huskey et al. 1962][pp. 5-110–5-132]

[Hu 1972] Richard H. Hu, "Analog Computer Simulation of an FM Communication System", American Society for Engineering Education, Annual Conference, June 19–22, 1972

[Hülsenberg et al. 1975] Frieder Hülsenberg, Uwe Kiessling, Hartmut Schönborn, *Beziehung zwischen Produktion, Lagerhaltung und Marktrealisation – dargestellt an einem Analogie-Rechenmodell für den Analogrechner MEDA T*, VEB Deutscher Verlag für Grundstoffindustrie, 1975

[Hütte 1926] Akademischer Verein Hütte, e.V. in Berlin (ed.), „Hütte – Des Ingenieurs Taschenbuch“, 25. neubearbeitete Auflage, II. Band, Berlin 1926, Verlag von Wilhelm Ernst & Sohn

[Hume et al. 2005] Ted Hume, Bob Koppany (ed.), *The Oughtred Society Slide Rule Reference Manual*, Striking Impressions, Los Angeles, California, First Edition, 2005

[Hurst] Charles J. Hurst, "Computer Simulation in a Mechanical Engineering Laboratory Program", author's archive

[Huskey et al. 1962] Harry D. Huskey, Granino A. Korn, *Computer Handbook*, McGraw-Hill Book Company, Inc., 1962

[Hyatt et al. 1968] Gilbert P. Hyatt, Gene Ohlberg, "Electrically alterable digital differential analyzer", in AFIPS '68 (Spring) Proceedings of the April 30–May 2, 1968 spring joint computer conference, pp. 161–169

[Jackson 1960] Albert S. Jackson, *Analog Computation*, McGraw-Hill Book Company, Inc., 1960

[James et al. 1971] M. L. James, G. M. Smith, J. C. Wolford, *Analog Computer Simulation of Engineering Systems*, Intext Educational Publishers, 1971, 3rd edition

[Jamshidi 1976] M. Jamshidi, "Optimization of some Dynamic Industrial Control Processes by Analog Simulation", in *Trans. IMACS*, Vol. XVIII, No. 2, April 1976, pp. 93–100

[Janac 1976] Karel Janac, "Control of Large Power Systems Based on Situation Recognition and High Speed Simulation", presented at 9th-Hawaii International Conference on System Sciences, January 6–8, 1976, University of Hawaii, author's archive

[Janssen et al. 1955] J. M. L. Janssen, L. Ensing, "The Electro-Analogue, an Apparatus for Studying Regulating Systems", in [Paynter, ed. 1955][pp. 147–161]

[Jezierski 2000] Dieter von Jezierski, *Slide Rules – A Journey Through Three Centuries*, Astragal Press, Mendham, New Jersey, 2000

[Johnson 1915] R. D. Johnson, "The Differential Surge Tank", in *Transactions of the American Society of Civil Engineers*, Vol. LXXVIII, 1915

[Johnson 1962] E. Calvin Johnson, "Computers and Control", in [Huskey et al. 1962] [pp. 21-62 ff.]

[Johnson 1963] Clarence L. Johnson, *Analog Computer Techniques*, McGraw-Hill Book Company, Inc., Second Edition, 1963

[Jones 1961] E. D. Jones, *Power Excursion in a Hanford Reactor Due to a Positive Reactivity Ramp*, HW-71119, Hanford Atomic Products Corporation, Richland, Washington, September 20, 1961

[Jung 2006] Walt Jung, *Op Amp History*, in [Analog Devices 2006], pp. 765–829

[Juslin 1981] Kaj Juslin, "Hybrid Computer Model for Synchronous and Asynchronous Motor Interaction Studies", Scandinavian Simulation Society annual meeting, 18–20th May, 1981 at Royal Institute of Technology, Stockholm

[Kahne 1968] Stephen J. Kahne, "Sensitivity-Function Calculation in Linear Systems Using Time-Shared Analog Integration", in *IEEE Transactions on Computers*, Vol. C-17, No. 4, April, 1968, pp. 375–279

[Kaplan] Paul Kaplan, "A Mathematical Model for Assault Boat Motions in Waves", Oceanics Inc., Plainview, New York, author's archive

[Karplus 1958] Walter J. Karplus, *Analog Simulation – Solution of Field Problems*, McGraw-Hill Book Company, Inc., 1958

[Karplus et al. 1958] Walter J. Karplus, Walter W. Soroka, *Analog Methods – Computation and Simulation*, McGraw-Hill Book Company, Inc., 1958

[Karplus et al. 1972] Walter J. Karplus, Richard A. Russell, "Increasing Digital Computer Efficiency with the Aid of Error-Correcting Analog Subroutines", in *IEEE Transactions on Computers*, Vol. C-20, No. 8, August 1972, pp. 831–837

[Kasper 1955] Joseph Emil Kasper, *Construction and application of a mechanical differential analyzer*, Thesis, State University of Iowa, February 1955

[Kella 1967] J. Kella, "A Note on the Accuracy of Digital Differential Analyzers", in *IEEE Transactions on Electronic Computers*, Vol. EC-16, No. 2, April 1967, p. 230

[Kella et al. 1968] J. Kella, A. Shani, "On the Reversibility of Computations in a Digital Differential Analyzer", in *IEEE Transactions on Computers*, Vol. C-17, No. 3, 1968, pp. 283–284

[Kennedy 1962] Jerome D. Kennedy Sr., "Representation of Time Delays", in [Huskey et al. 1962][pp. 6-3–6-16]

[Kettel 1960] E. Kettel, „Die Anwendungsmöglichkeiten der Analogrechentechnik in Meßtechnik und Nachrichtenverarbeitung", in *Telefunken Zeitung*, Vol. 33 (September 1960), No. 129, pp. 164–171

[Kettel et al. 1967] E. Kettel, A. Kley, H. Mangold, R. Maunz, G. Meyer-Brötz, H. Ohnsorge, J. Schürmann, „Der Einsatz elektronischer Rechner für Aufgaben der nachrichtentechnischen Systemforschung", in *Telefunken Zeitung*, Vol. 40 (1967), No. 1/2, pp. 3–9

[Kerr 1978] C. N. Kerr, "Use of the Analog/Hybrid Computer in Boundary Layer and Convection Studies", American Society for Engineering Education, 86th Annual Conference, University of British Columbia, June 19–22, 1978

[Kerr 1980] C. N. Kerr, "Analog Solution of Free Convection Mass Transfer From Downward-Facing Horizontal Plates", in *Int. J. Heat Mass Transfer*, Vol. 23, 1980, pp. 247–249

[Kidd et al. 1961] E. A. Kidd, G. Bull, R. P. Harper Jr., "In-Flight-Simulation – Theory and Application", AGARD Report 368, April 1961

[Kinzel et al. 1962/1] B. Kinzel, L. Sengewitz, „Radizierender Verstärker, insbesondere zur Verarbeitung von Einspritzwerten bei Dieselmotoren", in *Elektronische Rundschau*, Januar 1962, Vol. 16, No. 1, pp. 21–23

[Kinzel et al. 1962/2] B. Kinzel, L. Sengewitz, „Erweiterter radizierender Verstärker", in *Elektronische Rundschau*, Mai 1962, Vol. 16, No. 5, p. 223

[Klein et al. 1957] Martin L. Klein, Frank K. Williams, Harry C. Morgan, "Digital Differential Analyzers", in *Instruments and Automation*, June 1957, pp. 1105–1109

[Kley et al. 1966] A. Kley, E. Heim, „Ein elektronischer Koordinatenwandler", in *Telefunkenzeitung*, Vol. 39, No. 1, 1966, pp. 60–65

[Kline 1993] Ronald Kline, "Harold Black and the Negative-Feedback Amplifier", in *IEEE Control Systems*, August 1993, pp. 82–85

[Klipsch 1981] Paul W. Klipsch, "In Memoriam: Paul G. A. H. Voigt", in *J. Audio Eng. Soc.*, Vol. 29, No. 4, 1981 April, p. 308

[Knorre 1971] Wolfgang A. Knorre, *Analogcomputer in Biologie und Medizin – Einführung in die dynamische Analyse biologischer Systeme*, VEB Gustav Fischer Verlag Jena, 1971

[Koenig et al. 1955] Eldo C. Koenig, William C. Schultz, "How to Select Governor Parameters with Analog Computers", in [Paynter, ed. 1955][pp. 237–238]

[Kopacek] P. Kopacek, "Testing Various Identification Algorithms for Control Systems with Stochastically Varying Parameters by a Hybrid Computer", in *10th IMACS World Congress on System Simulation and Scientific Computation*, pp. 69–71

[Korn & Korn 1956] Granino A. Korn, Theresa M. Korn, *Electronic Analog Computers (D-c Analog Computers)*, McGraw-Hill Book Company, Inc., 1956

[Korn 1962] Granino A. Korn, "Electronic Function Generators, Switching Circuits and Random-Noise Generators", in [Huskey et al. 1962][pp. 3-62–3-84]

[Korn & Korn 1964] Granino A. Korn, Theresa M. Korn, *Electronic Analog and Hybrid Computers*, McGRAW-HILL BOOK COMPANY, 1964

[Korn et al. 1970] Granino A. Korn, H. Kosako, "A Proposed Hybrid-Computer Method for Functional Optimization", in *IEEE Transactions on Computers*, Vol. C-19, No. 2, February 1970, pp. 149–153

[Kovach et al. 1962] L. D. Kovach, H. F. Meissinger, "Solution of Algebraic Equations, Linear Programming, and Parameter Optimization", in [Huskey et al. 1962][pp. 5-133–5-154]

[Kovach 1952] L. D. Kovach, "Solution of Difference Equations", in [Huskey et al. 1962][pp. 6-52–6-56]

[Kraft et al. 2002] Chris Kraft, James L. Schefter, *Flight – My Life in Mission Control*, First Plume Printing, March 2002

[Kramer] H. Kramer, „Parameteroptimierung mit einem hybriden Analogrechner an einem Beispiel aus der chemischen Reaktionskinetik", AEG-Telefunken, AFA 003 0570

[Kramer 1968] H. Kramer, „Optimierung eines Regelkreises mit Tischanalogrechner und Digitalzusatz", in *elektronische datenverarbeitung*, (1968) 6, pp. 293–297

[Krause 1970] Paul C. Krause, "Applications of analog and hybrid computers in electric power research", in *SIMULATION*, August 1970, pp. 73–79

[Krause 1971] Paul C. Krause, "Hybrid Computation Techniques Applied to Power Systems Simulation", Purdue University, School of Electrical Engineering, November, 1971

[Krause 1974] Paul C. Krause, "Applications of Analog and Hybrid Computation in Electric Power System Analysis", in *Proceedings of the IEEE*, Vol. 62, No. 7, July 1974, pp. 994–1009

[Krause et al. 1977] P. C. Krause, W. C. Hollopeter, D. M. Triezenberg, "Sharp Torques During Out-Of-Phase Synchronization", in *IEEE Transactions on Power Apparatus and Systems*, Vol. PAS-96, No. 4, July/August 1977, pp. 1318–1323

[Krause 2006] Christine Krause, „Die Entwicklung der Analogrechentechnik in Thüringen und Sachsen", 3. Greifswalder Symposium zur Entwicklung der Rechentechnik und 12. Internationales Treffen für Rechenschieber- und Rechenmaschinensammler, 2006

[Kregeloh 1956] H. Kregeloh, „Analogrechner und ihre Anwendung auf ein volkswirtschaftliches Modell“, in *Mathematical Methods of Operations Research*, Springer-Verlag, Vol. 1, No. 1, Dezember 1956, pp. 97–106

[Kron 1945/1] Gabriel Kron, “Electric Circuit Models of the Schrödinger Equation”, in *Physical Review*, Vol. 67, No. 1/2, January 1 and 15, 1945, pp. 39–43

[Kron 1945/2] Gabriel Kron, “Numerical Solution of Ordinary and Partial Differential Equations by Means of Equivalent Circuits”, in *Journal of Applied Physics*, 16, 1945, pp. 172–186

[Künkel 1961] H. Künkel, „Beitrag zu einer regeltheoretischen Analyse der Pupillenreflexdynamik“, in *Kybernetik*, 1, 1961, pp. 69–75

[Land] Bruce Land, *DDA on FPGA – a modern Analog Computer*, `https://instruct1.cit.cornell.edu/Courses/ece576/DDA/index.htm`, retrieved 03/03/2013

[Landauer 1974] J. Paul Landauer, “Personal Rapid Transit (PRT) System Design by Hybrid Computation”, in EAI Scientific Computation Report, No. SCR 74-17, November 11, 1974

[Landauer 1975] J. Paul Landauer, “Non-Destructive Destructive Testing”, in *Industrial Research*, March 1975

[Lange 2006] Thomas H. Lange, *Peenemünde – Analyse einer Technologieentwicklung im Dritten Reich*, Reihe *Technikgeschichte in Einzeldarstellungen*, VDI-Verlag, GmbH, Düsseldorf 2006

[Larrowe 1955] Vernon L. Larrowe, “Direct Simulation – Bypasses Mathematics, Simplifies Analysis”, in [Paynter, ed. 1955][p. 127–133]

[Larrowe 1966] Vernon L. Larrowe, “Band-Pass Quadrature Filters”, in *IEEE Transactions on Electronic Computers*, Vol. EC-15, No. 5, October 1966, pp. 726–731

[Leatherwood 1972] Jack D. Leatherwood, “Analog Analysis of a Tracked Air-Cushion Vehicle”, in *INSTRUMENTS and CONTROL SYSTEMS*, April 1972, pp. 81–86

[Leise 2007] Tanya Leise, “As the Planimeter’s Wheel Turns: Planimeter Proofs for Calculus Class”, in *College Mathematics Journal*, January 2007

[Levine 1964] Leon Levine, *Methods for Solving Engineering Problems Using Analog Computers*, McGraw-Hill Book Company, 1964

[Lewin 1972] John Ernest Lewin, “Area Measurement”, United States Patent 3652842, Mar. 28, 1972

[Lewis 1958] Lloyd G. Lewis, “Simulation of a Solvent Recovery Process”, in *Instruments and Automation*, Vol. 31, April 1958, pp. 644–647

[Librascope 1957] N. N., “Ball/Disc Integrator”, in *Instruments and Automation*, April 1957, p. 769

[Light et al. 1966] L. Light, J. Badger, D. Barnes, “An Automatic Acoustic Ray Tracing Computer”, in *IEEE Transactions on Electronic Computers*, Vol. EC-15, No. 5, October, 1966, pp. 719–725

[Light 1975] Leon Henry Light, "Apparatus for Integration and Averaging", United States Patent 3906190, Sept. 16, 1975

[Lilamand 1956] M. Lejet Lilamand, "A Time-Division Multiplier" in *IRE Transactions on Electronic Computers*, March 1956, pp. 26–34

[Linear Technology] Linear Technology, *LTC6943 – Micropower, Dual Precision Instrumentation Switched Capacitor Building Block*, Linear Technology Corporation

[Lotz 1969] Hermann Lotz, „Einsatz des Analogrechners in der Regelungs- und Steuerungstechnik", in *Steuerungstechnik*, 2 (1969) 11, pp. 430–435

[Lovell et al. 1946] Clarence A. Lovell, David B. Parkinson, Bruce T. Weber, *Electrical Computing System*, United States Patent 2404387, July 23, 1946

[Loveman 1962] Bernard D. Loveman, "Computer Servomechanisms and Servo Resolvers", in [Huskey et al. 1962][pp. 3-1–3-40]

[Lowe] William Lowe, "NUC Simulation Facility", Naval Undersea Center, San Diego, California, author's archive

[Ludwig 1966] R. Ludwig, *Stability Research on Parachutes using Digital and Analog Computers*, National Aeronautics and Space Administration, Washington, D. C., November 1966

[Ludwig et al. 1974] Manfred Ludwig, Klaus Kaplick, *Elektronische Analogrechner und Prozeßrechnereinsatz*, Reihe *Programmierung und Nutzung von Rechenanlagen, Teil 6*, Verlag „Die Wirtschaft", Berlin, 1974

[Lukes 1967] Jaroslav H. Lukes, "Oscillographic Examination of the Operation of Function Generators", in *IEEE Transactions on Electronic Computers*, Vol. EC-16, No. 2, April 1967, pp. 133–139

[Lunderstädt et al. 1981] R. Lunderstädt, W. Menssen, „Regelmechanismus der menschlichen Pupille – Stabilität und Simulation", Dornier System GmbH, 1981

[Macduff et al. 1958] John N. Macduff, John R. Curreri, *Vibration Control*, McGraw-Hill Book Company, Inc., 1958

[Mackay 1962] Donald M. MacKay, Michael E. Fisher, *Analogue Computing at Ultra-High Speed*, John Wiley & Sons Inc., 1962

[Macnee 1948] A. B. Macnee, *An Electronic Differential Analyzer*, Technical Report No. 90, December 16, 1948, Research Laboratory of Electronics, Massachusetts Institute of Technology

[Mahrenholtz 1968] O. Mahrenholtz, *Analogrechnen in Maschinenbau und Mechanik*, Bibliographisches Institut, Mannheim/Zürich, 1968

[Malstrom et al. 1977] Carl W. Malstrom, Rainer Heller, Mohammad S. Khan, "Hybrid Computation – an Advanced Computation Tool for Simulating the Nonlinear Dynamic Response of Railroad Vehicles", Pre-Publication Copy of Submission to the Post Conference Proceedings, *Advanced Techniques in Track/Train Dynamics and Design Conference*, Chicago, Illinois, September 27 and 28, 1977

[Manske 1968] R. A. Manske, "Computer Simulation of Narrowband Systems", in *IEEE Transactions on Computers*, Vol. C-17, No. 4, April, 1968, pp. 301–308

[Markson 1958] A. A. Markson, "Introduction to Reactor Physics", in *Instruments and Automation*, Vol. 31, April 1958, pp. 616–623

[Marquitz et al. 1968] W. T. Marquitz, Y. Tokad, "On Improving the Analog Computer Solutions of Linear Systems", in *IEEE Transactions on Computers*, Vol. C-17, No. 3, March, 1968, pp. 268–270

[Martin 1969] George J. Martin, "Hybrid Computation in the Engineering College", in *Engineering Education*, January 1969, pp. 395–400

[Martin 1972] George J. Martin, "Analog and Hybrid Simulation in Science Education", in *Educational Technology*, April, 1972, pp. 62–63

[Maslo 1974] Ronald M. Maslo, "Dynamic Response of a Ship in Waves", in EAI Scientific Computation Report, No. 74-14, September 20, 1974

[Master et al. 1955] R. C. Master, R. L. Merrill, B. H. List, "Analogous Systems in Engineering Design", in [Paynter, ed. 1955][p. 134–145]

[MBB] N. N., *MBB Simulation*, Firmenschrift Messerschmitt Bölkow Blohm GmbH, Unternehmensbereich Flugzeuge

[McCallum] I. R. McCallum, "Horses for Courses: The Mathematical Modelling Requirements of Maritime Simulators", author's archive

[McCarthy 2009] Jerry McCarthy, „Der Mechanismus von Antikythera“, 15. Internationales Treffen der Rechenschiebersammlung und 4. Symposium zur Entwicklung der Rechentechnik, Ernst Moritz Arndt Universität Greifswald, 2009

[McDonal 1956] Frank J. McDonal, "Wave Analysis", United States Patent 2752092, June 26, 1956

[McFadden et al. 1958] Norman M. McFadden, Frank A. Pauli, Donovan R. Heinle, "A Flight Study of Longitudinal-Control-System Dynamic Characteristics by the Use of a Variable-Control-System Airplane", NACA RM A57L10, 1958

[McGhee et al. 1970] Robert B. McGhee, Ragnar N. Nilsen, "The Extended Resolution Digital Differential Analyzer: A New Computing Structure for Solving Differential Equations", in *IEEE Transactions on Computers*, Vol. C-19, No. 1, January 1970, pp. 1–9

[McLean et al. 1977] L. J. McLean, E. J. Hahn, "Simulation of the Transient Behaviour of a Rigid Rotor in Squeeze Film Supported Journal Bearings", 2nd AINSE Engineering Conference, 1977

[McLeod et al. 1957] John H. McLeod, Robert M. Leger, "Combined Analog and Digital Systems – Why, When, and How", in *Instruments and Automation*, June 1957, pp. 1126–1130

[McLeod et al. 1958/1] John H. McLeod, Suzette McLeod, "The Simulation Council Newsletter", in *Instruments and Automation*, Vol. 31, January 1958, pp. 119–124

[McLeod et al. 1958/2] John H. McLeod, Suzette McLeod, "The Simulation Council Newsletter", in *Instruments and Automation*, Vol. 31, February 1958, pp. 297–300

[McLeod et al. 1958/3] John H. McLeod, Suzette McLeod, "The Simulation Council Newsletter", in *Instruments and Automation*, Vol. 31, March 1958, pp. 487–491

[McLeod et al. 1958/4] John H. McLeod, Suzette McLeod, "The Simulation Council Newsletter", in *Instruments and Automation*, Vol. 31, July 1958, pp. 1219–1225

[McLeod et al. 1958/5] John H. McLeod, Suzette McLeod, "The Simulation Council Newsletter", in *Instruments and Automation*, Vol. 31, August 1958, S. 1385–1390

[McLeod et al. 1958/6] John H. McLeod, Suzette McLeod, "The Simulation Council Newsletter", in *Instruments and Automation*, Vol. 31, December 1958, pp. 1991–1997

[McLeod 1962] John H. McLeod, "Electronic-Analog-Computer Techniques for the Design of Servo Systems", in [Huskey et al. 1962][pp. 5-35 ff.]

[Meccano 1934] N. N., "Meccano Aids Scientific Research", in *Meccano Magazine*, Vol. XIX, No. 6, June, 1934, p. 441

[Meccano 1934/2] N. N., "Machine Solves Mathematical Problems – A Wonderful Meccano Mechanism", in *Meccano Magazine*, Vol. XIX, No. 6, June, 1934, pp. 442–444

[Medkeff et al. 1955] R. J. Medkeff, H. Matthews, "Solving process-control problems by ANALOG COMPUTER", in [Paynter, ed. 1955][pp. 164–166]

[Meisinger 1978] Reinhold Meisinger, "Analog Simulation of Magnetically Levitated Vehicles on Flexible Guideways", in *Simulation of Control-Systems*, I. Troch (ed.), North-Holland Publishing Company, 1978, pp. 207–214

[Meissl 1960/1] P. Meissl, „Behandlung von Wasserschloßaufgaben mit Hilfe eines elektronischen Analogrechners, Teil 1“, in *mtw – Zeitschrift für moderne Rechentechnik und Automation*, 1/60, pp. 9–13

[Meissl 1960/2] P. Meissl, „Behandlung von Wasserschloßaufgaben mit Hilfe eines elektronischen Analogrechners, Teil 2“, in *mtw – Zeitschrift für moderne Rechentechnik und Automation*, 2/60, pp. 74–77

[Meyer-Brötz 1960] G. Meyer-Brötz, „RA 800 – Ein transistorisierter Präzisions-Analogrechner“, in *Telefunken Zeitung*, Vol. 33 (September 1960), No. 129, pp. 171–182

[Meyer-Brötz 1962] G. Meyer-Brötz, „Die Messung von Kenngrößen stochastischer Prozesse mit dem elektronischen Analogrechner“, in *Elektronische Rechenanlagen*, 4 (1962), No. 3, pp. 103–108

[Meyer-Brötz et al. 1966] G. Meyer-Brötz, E. Heim, „Ein breitbandiger Operationsverstärker mit Silizium-Transistoren“, in *Telefunken Zeitung*, Vol. 39 (1966), No. 1, pp. 16–32

[Meyer zur Capellen 1949] W. Meyer zur Capellen, *Mathematische Instrumente*, Akademische Verlagsgesellschaft Geest & Portig K.-G., Leipzig 1949

[Mezencev et al. 1978] R. Mezencev, R. Lepeix, "Hybrid Simulation of a Non Linear Hydro Pneumatic Damper for Ships", in *Simulation of Control Systems*, I. Troch (ed.), North-Holland Publishing Company, 1978, pp. 135–137

[Michaels] Lawrence H. Michaels, "The AC/Hybrid Power System Simulator and its Role in System Security", author's archive

[Michaels et al.] Lawrence H. Michaels, William Tessmer, John Muller, "The On-Line Power System Simulator", Electronic Associates, Inc., author's archive

[Michaels et al. 1971] G. C. Michaels, V. Gourishankar, "Hybrid Computer Solution of Optimal Control Problems", in *IEEE Transactions on Computers*, Vol. C-20, No. 2, February 1971, pp. 209–211

[Michels 1954] Lowell S. Michels, *Description of BENDIX D-12 DIGITAL DIFFERENTIAL ANALYZER*, Bendix Computer Division, Bendix Aviation Corporation, 5630 Arbor Vitae Street, Los Angeles 45, California, March 13, 1954

[Milan-Kamski 1969] W. J. Milan-Kamski, "A High-Accuracy, Real-Time Digital Computer for Use in Continuous Control Systems", in 1959 Proceedings of the Western Joint Computer Conference, pp. 197–201

[Miller 2011] David Philip Miller, "The Mysterious Case of James Watt's '1785 Steam Indicator': Forgery or Folklore in the History of an Instrument?", in *Int. J. for the History of Eng. & Tech.*, Vol. 81, No. 1, January, 2011, pp. 129–150

[Mills/1] Jonathan W. Mills, "The Architecture of an Extended Analog Computer Core", Computer Science Department, Indiana University

[Mills/2] Jonathan W. Mills, *Polymer Processors*, Computer Science Department, Indiana University, `http://www.cs.indiana.edu/pub/techreports/TR580.pdf`, retrieved 03/03/2013

[Mills 1995] Jonathan W. Mills, *The continuous retina: Image processing with a single-sensor artificial neural field network*, Computer Science Department, Indiana University, technical report 443, November 13, 1995

[Mills et al. 2006] Jonathan W. Mills, Bryce Himebaugh, Brian Kopecky, Matt Parker, Craig Shue, Chris Weilemann, "'Empty Space' Computers: The Evolution of an Unconventional Supercomputer", in *CF06*, May 3–8, 2006, Ischia, Italy

[Mindell 1995] David A. Mindell, "Automation's Finest Hour: Bell Labs and Automatic Control in World War II", in *IEEE Control Systems*, December 1995, pp. 72–78

[Mindell 2000] David A. Mindell, "Opening Black's Box – Rethinking Feedback's Myth of Origin" in *Technology and Culture*, July 2000, Vol. 41, pp. 405–434

[Mitchell et al. 1966] E. E. L. Mitchell, J. B. Mawson, J. Bulger, "A Generalized Hybrid Simulation for an Aerospace Vehicle", in *IEEE Transactions on Electronic Computers*, Vol. EC-15, No. 3, June 1966, pp. 304–313

[Miura 1967] Takeo Miura, Junji Tsuda, Junzo Iwata, "Hybrid Computer Solution of Optimal Control Problems by the Maximum Principle", in *IEEE Transactions on Electronic Computers*, Vol. EC-16, No. 5, October 1967, pp. 666–670

[Morrill 1962] Charles D. Morrill, “Electronic Multipliers and Related Topics”, in [Huskey et al. 1962][pp. 3-40–3-62]

[E. Morrison 1962] E. Morrison, “Nuclear-Reactor Simulation”, [Huskey et al. 1962] [pp. 5-87–5-93]

[J. E. Morrison 2007] James E. Morrison, *The Astrolabe*, Janus Publishing, 2007

[Nalley 1969] Donald Nalley, “Z Transform and the Use of the Digital Differential Analyzer as a Peripheral Device to a General Purpose Computer”, NASA Technical Memorandum, NASA TM X-53866, August 12, 1969

[Nava-Segura et al.] A. Nava-Segura, L. L. Freris, “Hybrid Computer Simulation of DC Transmission Systems”, author’s archive

[Neufeld 2007] Michael J. Neufeld, *Von Braun – Dreamer of Space, Engineer of War*, Borzoi Book, Alfred A. Knopf, 2007

[Nise] Norman S. Nise, “Analog Computer Experiments for Undergraduate Courses in Network Analysis and Automatic Controls”, Vol. II, No. 2, author’s archive

[N. N. 1945] N. N., *Das Gerät A4 Baureihe B, Teil III, Gerätebeschreibung V2*, OKH/Wa A/Wa Prüf, Anlage zu Bb.Nr 19/45 gK, 1.2.1945 4/64, p. 175

[N. N. 1956] N. N., *Nike I Systems – Nike I Computer, SAM Problem Analysis, Servo Loop Elements and Power Distribution*, TM9-5000-13, Department of the Army, May 1956

[N. N. 1957/1] N. N., “Berkeley opens its new computer facility”, in *Instruments and Automation*, February 1957, p. 288

[N. N. 1957/2] N. N., “Bonneville Power Administration Solves Swing Equations with EASE”, in *Instruments and Automation*, March 1957, p. 498

[N. N. 1957/3] N. N., „Eröffnung des ersten europäischen Analog-Rechenzentrums“, in *Elektronische Rundschau*, August 1957, Vol. 11, No. 8, p. 253

[N. N. 1957/4] N. N., “New GEDA Power Dispatch Computer”, in *Instruments and Automation*, Vol. 30, February 1957, p. 179

[N. N. 1957/5] N. N., “University Research Instrumentation”, in *Instruments and Automation*, June 1957, p. 1120

[N. N. 1957/6] N. N., “New Data Handling Centers”, in *Instruments and Automation*, April 1957, p. 608

[N. N. 1958/1] N. N., “Computer Designed Rolling Mill”, in *Instruments and Automation*, Vol. 31, February 1958, p. 283

[N. N. 1958/2] N. N., “Distillation-Column Dynamic Characteristics”, in *Instruments and Automation*, Vol. 31, August 1958, pp. 1357–1359

[N. N. 1960] N. N., „Funktionsgruppen für die Analogrechentechnik“, in *Elektronische Rechenanlagen*, 2 (1960), No. 1, pp. 43–44

[N. N. 1961] N. N., „Messen – Datenverarbeiten – Auswerten“, advertisement in *Elektronische Rechenanlagen*, 3 (1961), No. 1, p. 44

[N. N. 1964/1] N. N., „Electronic Associates, Inc., Europäisches Rechenzentrum für Analog- und Hybridrechentechnik“, in *Elektronische Rechenanlagen*, 6 (1964), No. 4, p. 214

[N. N. 1964/2] N. N., "EAI awarded contract for Hybrid Computing System", in *mtw – Zeitschrift für moderne Rechentechnik und Automation*, 4/64, p. 175

[N. N. 1964/3] N. N., "Nuclear Power Plant of N. S. Savannah simulated by Analog Computers", in *mtw – Zeitschrift für moderne Rechentechnik und Automation*, 3/64, p. 129

[N. N. 2006] N. N., "Video Games – Did They Begin at Brookhaven?", `http://www.osti.gov/accomplishments/videogame.html`, retrieved 11/20/2006

[Nolan 1955] John E. Nolan, "Analog Computers and their Application to Heat Transfer and Fluid Flow – Part 1, 2, 3", in [Paynter, ed. 1955][pp. 109–126]

[Noronha] Leo G. Noronha, "The Benefits of Analog Computation and Simulation in the Electrical Supply Industry", author's archive

[O'Grady 1967] Emmett P. O'Grady, "Correlation Method for Computing Sensitivity Functions on a High-Speed Iterative Analog Computer", in *IEEE Transactions on Electronic Computers*, Vol. EC-16, No. 2, April 1967, pp. 140–146

[Oliver et al. 1974] W. Kent Oliver, Dale E. Seborg, D. Grant Gisher, "Hybrid Simulation of a Computer-Controlled Evaporator", in *SIMULATION*, September 1974, pp. 77–84

[Olson 1943] Harry F. Olson, *Dynamical Analogies*, D. van Nostrand Company, Inc., 1943

[Okah-Avae 1978] B. E. Okah-Avae, "Analogue computer simulation of a rotor system containing a transverse crack", in *SIMULATION*, December 1978, pp. 193–198

[Ott 1964] A. Ott, „Zur Bestimmung des Korrelationskoeffizienten zweier Funktionen mit dem Analogrechner“, in *Elektronische Rechenanlagen*, 6 (1964), No. 3, pp. 144–148

[Ovsyanko] V. M. Ovsyanko, "The Theory of Synthesis of Electronic Circuits of Linear and Non-linear Object of Structural Mechanics and Applied Elasticity Theory", in *14. seminář MEDA ANALOGOVÁ A HYBRIDNÍ VÝPOČETNÍ TECHNIKA*, Praha 1977, pp. 25–29

[Packard Bell] N. N., *The HYCOMP Hybrid Analog/Digital Computing System*, Packard-Bell Computer

[Palevsky 1962] Max Palevsky, "The Digital Differential Analyzer", in [Huskey et al. 1962][pp. 19-14–19-74]

[Park et al. 1972] William H. Park, James C. Wambold, "Teaching Digital and Hybrid Simulation of Mechanical Systems at the Graduate Level", Delivered during the Joint ACES/ASEE Session No. 3540 at the 1972 Annual Meeting of the American Society for Engineering Education at Texas Tech, Lubbock, Texas, June 19–22, 1972

[Paschkis et al. 1968] Victor Paschkis, Frederick L. Ryder, *Direct Analog Computers*, Interscience Publishers, 1968

[Payne 1988] Peter R. Payne, "An Analog Computer which Determines Human Tolerance to Acceleration", in *39th Annual Astronautical Congress of the International Astronautical Federation*, Bangalore, 8-15 Oct. 1988, pp. 271–300

[Paynter, ed. 1955] Henry M. Paynter (ed.), *A Palimpsest on the Electronic Analog Art*, printed by Geo. A. Philbrick Researches Inc., AD 1955

[Paynter et al. 1955] Henry M. Paynter, J. M. Asce, "Surge and Water Hammer Problems", in [Paynter, ed. 1955][pp. 217–223]

[Paynter 1955/1] Henry M. Paynter, "Methods and Results from M.I.T. Studies in Unsteady Flow", in [Paynter, ed. 1955][pp. 224–228]

[Paynter 1955/2] Henry M. Paynter, "A Discussion by H. M. Paynter of AIEE Paper 53 – 172", in [Paynter, ed. 1955][pp. 229–232]

[Paynter] Henry M. Paynter, "A Retrospective on Early Analysis and Simulation of Freeze and Thaw Dynamics", cf. `http://www.me.utexas.edu/~lotario/paynter/hmp/PAYNTER_Permafrost.pdf`, retrieved 12/04/2008

[Pease 2003] Bob Pease, "What's All This K2-W Stuff, Anyhow?", in *electronic design*, January 2003, `http://electronicdesign.com/article/analog-and-mixed-signal/what-s-all-this-k2-w-stuff-anyhow-2530`, retrieved 12/12/2012

[Perera 1969] K. K. Y. Wije Perera, "Optimum generating schedule for a hydrothermal power system / an analog computer solution to the short-range problem", in *SIMULATION*, April 1969, pp. 191–199

[Petzold 1992] Hartmut Petzold, *Moderne Rechenkünstler – Die Industrialisierung der Rechentechnik in Deutschland*, Verlag C. H. Beck, 1992

[Pfaltzgraff 1969] David J. Pfaltzgraff, "Analog Simulation of the Bouncing-Ball Problem", in *American Journal of Physics*, Volume 37, Number 10, October 1969, pp. 1008–1013

[Philbrick 1948] George A. Philbrick, "Designing Industrial Controllers by Analog", in *Electronics*, June, 1948, pp. 108–111

[Phillips 1950] A. W. Phillips, "Mechanical Models in Economic Dynamics", in *Economica*, New Series, Vol. 17, No. 67, Aug. 1950, pp. 283–305

[Piceni et al. 1975] Hans A. L. Piceni, Pieter Eykhoff, "The Use of Hybrid Computers for System-Parameter Estimation", in *Annales de l'Association internationale pour le Calcul analogique*, No. 1, Janvier 1975, pp. 9–22

[Pierre 1986] Donald A. Pierre, *Optimization Theory with Applications*, Dover Publications, Inc., New York, 1986

[Pirrello et al. 1971] C. J. Pirrello, R. D. Hardin, J. P. Capellupo, W. D. Harrison, *An Inventory of Aeronautical Ground Research Facilities – Volume IV – Engineering Flight Simulation Facilities*, National Aeronautics and Space Administration, Washington, D. C., November 1971, NASA CR-1877

[Popović 1964] D. P. Popović, „Die Automatisierung des von Mieses'schen Iterationsverfahrens auf dem Analogrechner“, in *mtw – Zeitschrift für moderne Rechentechnik und Automation*, 3/64, pp. 104–110

[Popular Mechanics 1950] N. N., “It's Small But Smart, This 'Suitcase Brain'”, in *Popular Mechanics*, 8, 1950

[Powell] Fred O. Powell, “Analog simulation of an adaptive two-time-scale control system”, Advanced Electronic Systems Research Department, Bell Aerospace Division of Textron, Buffalo, New York 14240

[Press et al. 2001] William H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery, *Numerical Recipes in Fortran 77 – The Art of Scientific Computing, Volume 1 of Fortran Numerical Recipes*, Cambridge University Press, Second Edition, 2001

[Preuss 1962] Heinzwerner Preuss, *Grundriss der Quantenchemie*, Bibliographisches Institut, Mannheim, 1962

[Preuss 1965] Heinzwerner Preuss, *Quantentheoretische Chemie*, Bibliographisches Institut, Mannheim, 1965

[de Solla Price 1974] Derek de Solla Price, “Gears from the Greeks: The Antikythera Mechanism – a Calendar Computer from ca. 80 B.C.”, in *Transactions of the American Philosophical Society*, Volume 64, Part 7, 1974

[Ragazzini et al. 1948] John R. Ragazzini, Robert H. Randall, Frederick A. Russel, “Analysis of Problems in Dynamics by Electronic Circuits”, in *Proceedings of the I.R.E*, Vol. 35, May 1947, pp. 444 ff.

[Ramirez 1976] W. Fred Ramirez, *Process Simulation*, D. C. Heath and Company, 1976

[Ranfft et al. 1977] Roland Ranfft, Hans-Martin Rein, „Analog simulation of bipolar-transistor circuits“, in *SIMULATION*, September 1977, pp. 75–78

[Rationalisierungskuratorium 1957] Rationalisierungskuratorium der Deutschen Wirtschaft (Hg.), *Automatisierung*, Carl Hanser Verlag, München, 1957

[Ratz 1967] Alfred G. Ratz, “Analog Computation of Fourier Series and Integrals”, in *IEEE Transactions on Electronic Computers*, Vol. EC-16, No. 4, August 1967, p. 515

[Rechberger 1959] H. Rechberger, „Zweite internationale Tagung für Analogierechentechnik“, in *mtw – Zeitschrift für moderne Rechentechnik und Automation*, 1/59, pp. 18–19

[Redheffer 1953] Raymond M. Redheffer, “Computing Machine”, United States Patent 2656102, Oct. 20, 1953

[Reinel 1976] K. Reinel, „Bewegungssimulatoren für Raumfahrt-Lageregelungssysteme“, in [Schöne 1976/1][pp. 464–475]

[Remington 1956] Remington Rand Univac, *Increment Computer Logic And Programming*, Bomber Weapons Defense Computer Study, Final Engineering Report, Volume 4, October 1956

[Rieger et al. 1974] Neville F. Rieger, Charles H. Thomas Jr., “Some Recent Computer Studies on the Stability of Rotors in Fluid-Film Bearings”, Rochester Institute of Technology, Mechanical Engineering Department, July, 1974

[Rideout 1962] Vincent C. Rideout, “Random-Process Studies”, [Huskey et al. 1962] [pp. 5-94–5-110]

[Rigas et al.] Harriett B. Rigas, Andrew M. Juraszek, “Some Approaches to the Design of a Model for an Aquatic Ecosystem”, author’s archive

[Robinson] Tim Robinson, “Torque amplifiers in Meccano”, `http://www.meccano.us/differential_analyzers/robinson_da/torque_amplifiers.pdf`, retrieved 07/26/2005

[Roedel 1955] Jerry Roedel, “History and Nature of Analog Computors”, in [Paynter, ed. 1955][pp. 27–47]

[Roedel 1955/2] Jerry Roedel, “Application of an Analog Computer to Design Problems for Transportation Equipment”, in [Paynter, ed. 1955][pp. 199–215]

[Röpke et al. 1969] Horst Röpke, Jürgen Riemann, *Analogcomputer in Chemie und Biologie*, Springer-Verlag, 1969

[Rössler 2005] Eberhard Rössler, *Die Torpedos der deutschen U-Boote*, Verlag E. S. Mittler & Sohn GmbH, 2005

[Rohde 1977] Wolfgang H. Rohde, *Beurteilung und Optimierung von Maschinensystemen in der Entwurfsphase – Dargestellt am Beispiel eines drehzahlgesteuerten Walzwerksantriebes*, Dissertation an der Technischen Universität Clausthal, 1977

[Rohde et al. 1981] Wolfgang H. Rohde, Jürgen Stelbrink, „Auslegung und konstruktive Gestaltung von Antriebssystemen schwerer Walzwerke“, in *Stahl und Eisen*, No. 13/14/1981, pp. 164–173

[Rosko 1968] Joseph S. Rosko, “Comments on ‘Hybrid Computer Solution of Optimal Control Problems by the Maximum Principle’”, *IEEE Transactions on Computers*, Vol. C-17, No. 9, September 1968, p. 899

[Rudnicki] Mieczyslaw Rudnicki, „Analogrechner MEDA in der Lasertechnik“, in *14. seminář MEDA ANALOGOVÁ A HYBRIDNÍ VÝPOČETNÍ TECHNIKA*, Praha 1977, pp. 53–55

[Russell 1962] Paul E. Russell, “Repetitive Analog Computers”, [Huskey et al. 1962] [pp. 6-17–6-25]

[Russel et al. 1971] Jack A. Russel, Bradford J. Baldwin, “Golf Game Computing System”, United States Patent 3598976, August 10, 1971

[Sadek 1976] K. Sadek, „Nachbildung einer Hochspannungs-Gleichstrom-Übertragung“, in [Schöne 1976/1][pp. 360–388]

[Sankar et al. 1979] Seshadri Sankar, David R. Hargreaves, “Hybrid computer optimization of a class of impact absorbers”, in *SIMULATION*, July 1979, pp. 11–18

[Sankar et al. 1980] S. Sankar, J. V. Svoboda, “Hybrid Computer in the Optimal Design of Hydro-Mechanical Systems”, in *Mathematics and Computers in Simulation*, XXII (1980), pp. 353–367

[Sauer] Albrecht Sauer, *Gezeiten – Ein Ausstellungsführer des Deutschen Schifffahrtsmuseums*, Deutsches Schiffahrtsmuseum

[Savet 1962] Paul Savet, "Heat-Transfer Computing Elements", [Huskey et al. 1962] [pp. 8-18–8-22]

[Schloemann-Siemag 1978] N. N., „Simulationstechnik im Schwermaschinenbau – Einsatz einer Analogrechenanlage für die Untersuchung und Berechnung von Maschinensystemen", Sonderdruck der Schloemann-Siemag AG, 2/12.78

[Schneider 1960] G. Schneider, „Über die Nachbildung und Untersuchung von Abtastsystemen auf einem elektrischen Analogrechner", in *Elektronische Rechenanlagen*, 2 (1960), No. 1, pp. 31–37

[Schöne 1976/1] Armin Schöne, *Simulation Technischer Systeme*, Band 2, Carl Hanser Verlag München Wien, 1976

[Schöne 1976/2] A. Schöne, „Modelle von Wärmetauschern", in [Schöne 1976/1] [pp. 7–27]

[Schüssler 1961] W. Schüssler, „Messung des Frequenzverhaltens linearer Schaltungen am Analogrechner", in *Elektronische Rundschau*, No. 10, 1961, pp. 471–477

[Schultz et al. 1974] Harold M. Schultz, Roger K. Miyasaki, Thomas B. Liem, Richard A. Stanley, "Analog Simulation of Compressor Systems", ISA CPD 74106, 1974

[Schwarz 1971] Wolfgang Schwarz, *Analogprogrammierung – Theorie und Praxis des Programmierens für Analogrechner*, VEB Fachbuchverlag Leipzig, 1. Ed., 1971

[Schweizer 1976/1] G. Schweizer, „Beispiele für die Simulation von Luftfahrzeugen", in [Schöne 1976/1][pp. 414–426]

[Schweizer 1976/2] G. Schweizer, „Das mathematische Modell für die Echtzeitsimulation von Erdsatelliten", in [Schöne 1976/1][pp. 434–453]

[Schweizer 1976/3] G. Schweizer, „Das Systemglied ‚Mensch' in der Simulation", in [Schöne 1976/1][pp. 555–580]

[Schweizer 1976/4] G. Schweizer, „Der Aufbau der Hybridsimulation für das Strahlflugzeug", in [Schöne 1976/1][pp. 520–526]

[Schweizer 1976/5] G. Schweizer, „Die Aufbereitung der Gleichungen des mathematischen Modells eines Flugzeugs zur Simulation auf dem Analogrechner", in [Schöne 1976/1][pp. 515–519]

[Schweizer 1976/6] G. Schweizer, „Die Sichtsimulation", in [Schöne 1976/1] [pp. 526–533]

[Schweizer 1976/7] G. Schweizer, „Simulationsprobleme aus der Luft- und Raumfahrt", in [Schöne 1976/1][pp. 389–398]

[Scientific Data Systems/2] N. N., *SDS 9300 Computer Reference Manual*, Scientific Data Systems, July 1969

[Scott 1958] Clyde C. Scott, "Power Reactor Control", in *Instruments and Automation*, Vol. 31, April 1958, pp. 636–637

[Seyferth 1960] H. Seyferth, „Über die Behandlung partieller Differentialgleichungen auf dem elektronischen Analogrechner“, in *Elektronische Rechenanlagen*, 2 (1960), No. 2, p. 85–92

[Sierck 1963] Joachim Sierck, *Untersuchung der nach dem Rückmischprinzip aufgebauten Frequenzteiler-Schaltungen unter besonderer Berücksichtigung der anomalen Mischrückkopplung*, Dissertation an der Fakultät für Elektrotechnik der RWTH Aachen

[Simons et al.] Fred O. Simons, Richard C. Harden, Sam J. Monte, "Perfected Analog/Hybrid Simulations of all Classes of Sampled-Data Systems", author's archive

[Skramstad 1959] H. K. Skramstad, "A Combined Analog-Digital Differential Analyzer", in *Proc. EJCC*, Volume 16, December 1959, pp. 94–101

[Snow 1930] L. T. Snow, "Planimeter", United States Patent 718166, January 13, 1930

[Somers 1980] Eric Somers, "Computer graphics for television", in *Video Systems*, June 1980, pp. 11–21

[Soroka 1962] Walter W. Soroka, "Mechanical Analog Computers", in [Huskey et al. 1962][pp. 8-2–8-16]

[Sorondo et al.] Victor J. Sorondo, George D. Wilson, "Hybrid Computer Simulation of a Circulating Water System", author's archive

[Soudack 1968] A. C. Soudack, "Canonical Programming of Nonlinear and Time-Varying Differential Equations", in *IEEE Transactions on Computers*, Vol. C-17, No. 4, April, 1968, p. 402

[Starnick 1976] J. Starnick, „Simulation chemischer Reaktoren“, [Schöne 1976/1] [pp. 51–205]

[Stepanow 1956] W. W. Stepanow, *Lehrbuch der Differentialgleichungen*, VEB Deutscher Verlag der Wissenschaften Berlin, 1956

[Stewart 1979] Peter A. Stewart, "The Analog Computer as a Physiology Adjunct", in *The Physiologist*, 1979, Issue 1, pp. 43–47

[Stillwell 1956] Wendell H. Stillwell, "Studies of Reaction Controls", in *Control Studies, Part B, Studies of Reaction Controls*, NASA, Document ID 19930092438, 1956

[Stillwell et al. 1958] Wendell H. Stillwell, Hubert M. Drake, "Simulator Studies of Jet Reaction Controls for Use at High Altitude", NACA Research Memorandum, September 26, 1958

[Stone et al.] John Stone, Ka-Cheung Taui, Eugene Pack, "Computer Control Study For a Manned Centrifuge", NASA Technical Report, F-B2300-1

[Stubbs et al. 1954] G. S. Stubbs, C. H. Single, *Transport Delay Simulation Circuits*, Westinghouse, Atomic Power Division, 1954

[Suryanarayanan et al. 1968] K. L. Suryanarayanan, A. C. Soudack, "Analog Computer Automatic Parameter Optimization of Nonlinear Control Systems with

Specific Inputs", in *IEEE Transactions on Computers*, Vol. C-17, No. 8, August, 1968, pp. 782–788

[Sutton et al. 1963] George H. Sutton, Paul W. Pomeroy, "Analog Analyses of Seismograms Recorded on Magnetic Tape", in *Journal of Geophysical Research*, Vol. 68, No. 9, May 1, 1963, pp. 2791–2815

[Svoboda 1948] Antonin Svoboda, *Computing Mechanisms and Linkages*, McGraw-Hill Book Company, Inc., 1948

[Swade 1995] Doron Swade, "The Phillips Economic Computer", in *Resurrection – The Bulletin of the Computer Conservation Society*, Issue Number 12, Summer 1995, pp. 11–18

[Swartzel 1946] Karl D. Swartzel, *Summing Amplifier*, United States Patent 2401779, June 11, 1946

[Sydow 1964] Achim Sydow, *Programmierungstechnik für elektronische Analogrechner*, VEB Verlag Technik Berlin, 1964

[Szalai 1971] Kenneth J. Szalai, "Validation of a general purpose airborne simulator for simulation of large transport aircraft handling qualities", NASA TN D-6431, October 1971

[Szuch et al. 1965] John R. Szuch, Leon M. Wenzel, Robert J. Baumbick, *Investigation of the Starting Characteristics of the M-1 Rocket Engine Using the Analog Computer*, NASA Technical Note TN D-3136, December 1965

[Tabbutt 1967] Frederick D. Tabbutt, "The Use of Analog Computers for Teaching Chemistry", in *Journal of Chemical Education*, Volume 44, Number 2, February 1967, pp. 64–69

[Tacker et al.] Edgar C. Tacker, Thomas D. Linton, "Hybrid Simulation of an Optimal Stochastic Control System", Louisiana State University, author's archive

[Telefunken 1958] N. N., „Elektronischer Analogrechner RA 463/2“, Telefunken, AH 5.2 Apr. 58

[Telefunken/1] N. N., „Demonstrationsbeispiel Nr. 5, Ball im Kasten“, AEG Telefunken

[Telefunken/2] N. N., „Schwingungsberechnung eines Zwei-Massen-Systems“, AEG-Telefunken

[Telefunken/3] N. N., „Perspektivische Darstellung von Rechenergebnissen mit Hilfe eines Analogrechners“, AEG-TELEFUNKEN Datenverarbeitung

[Telefunken/4] N. N., „Darstellung von Tragflügeln und ihren Stromlinien mit einem Analogrechner“, AEG Telefunken

[Telefunken/5] N. N., „Kepler und die Atomphysik – Der Beschuß eines Atomkerns mit Alphateilchen auf einem Tischanalogrechner“, Demonstrationsbeispiel 3, AEG-Telefunken, ADB 003 0570

[Telefunken/6] N. N., „Steuermanöver eines Satelliten“, AEG-Telefunken, AB 009/10 70

[Telefunken/7] N. N., „Echo I – Simulation der Umlaufbahn auf dem Analogrechner“, AEG-Telefunken, ADB 004 0770

[Telefunken 1958] N. N., „Angebot Nr. 557/0010 der Telefunken GmbH an die Technische Hochschule München“, 1958, author's archive

[Telefunken 1963/1] N. N., „Anwendungsbeispiele für Analogrechner – Wärmeleitung“, Telefunken, 15. Oktober 1963

[Telefunken 1963/2] N. N., „Anwendungsbeispiele für Analogrechner – Transformator“, Telefunken, 15. Oktober 1963

[Telefunken 1966] N. N., „Demonstrationsbeispiele für Analogrechner RAT 740, Beispiel 2, Einfache Darstellung einer Planetenbahn“, Mitteilungen der Fachabteilung Analogrechner, TELEFUNKEN, 10.1.66

[Teuber 1964] D. L. Teuber, *Nachbildung der Saturn V-Rakete auf elektronischen Analogrechnern*, Tagungsberichte Hermann Oberth-Gesellschaft, 13. Raketen- und Raumfahrttagung vom 25.–28. Juni 1964 in Darmstadt

[Thaler et al. 1980] G. J. Thaler, T. S. Nelson III, A. Gerba, “Real Time, Man-Interfaced Motion Analysis of a 3000 Ton Surface Effect Ship”, in *Summer Computer Simulation Conference*, 1980, pp. 593–598

[The Times 1954] N. N., “The Royal Aircraft Establishment's analogue computer...”, in *The Times*, October 8, 1954

[Thomas 1968] Charles H. Thomas, “Transport Time-Delay Simulation for Transmission Line Representation”, in *IEEE Transactions on Computers*, Vol. C-17, No. 3, March, 1968, pp. 205–214

[Thomas et al./1] C. H. Thomas, D. H. Welle, R. A. Hedin, R. W. Weishaupt, “Switching Surges on Parallel HV and EHV Untransposed Transmission Lines Studied by Analog Simulation”, IEEE, Paper No. 71 TP 128-PWR

[Thomas et al./2] Charles H. Thomas, A. E. Kilgour, D. H. Welle, T. A. Karnowski, “Transient Performance Study of a Parallel HV and EHV Transmission System”, author's archive

[Thomas et al. 1968] Charles H. Thomas, Ronald A. Hedin, “Switching Surges on Transmission Lines Studied by Differential Analyzer Simulation”, IEEE, Paper No. 68 RP 4-PWR, 1968

[Thompson] R. V. Thompson, “Application of Hybrid Computer Simulation Techniques to Warship Propulsion Machinery Systems Design”, author's archive

[Thomson 1876] William Thomson, “Mechanical Integration of linear differential equations of the second order with variable coefficients”, Proceedings of the Royal Society, Volume 24, No. 167, pp. 269–270, 1876

[Thomson 1882] Sir William Thomson, “Tides”, *Evening Lecture To The British Association At The Southhampton Meeting*, source: `http://www.fordham.edu/halsall/mod/1882kelvin-tides.html`, retrieved 11/24/2012

[Thomson 1911] Sir William Thomson, "The tidal gauge, tidal harmonic analyser, and tide predicter", in *Kelvin, Mathematical and Physical Papers,* Volume VI, Cambridge 1911, pp. 272–305

[Thwaites, ed. 1987] Bryan Thwaites (ed.), *Incompressible Aerodynamics – An Account of the Theory and Observation of the Steady Flow of Incompressible Fluid past Aerofoils, Wings, and Other Bodies*, Dover Publications, Inc., New York, 1987

[Tisdale] Henry F. Tisdale, "How a Modern Analog Computer Duplicates Real-World Behaviour of a Thyristor-Controlled Rectifier Bridge", Electronic Associates, Inc., author's archive

[Tisdale 1981] Henry F. Tisdale, "Hybrid Computers Retaining Favor with Controls & Design Engineers", in *EAI OPENERS*, Product Information Bulletin #048 – May, 1981

[Tomayko 1985] James E. Tomayko, "Helmut Hoelzer's Fully Electronic Analog Computer", in *Annals of the History of Computing*, Volume 7, Number 3, July 1985, pp. 227–240

[Tomayko 2000] James E. Tomayko, *Computers Take Flight – a History of NASA's Pioneering Digital Fly-By-Wire Project*, NASA SP-2000-4224, 2000

[Tomovic et al. 1961] R. Tomovic, W. J. Karplus, "Land Locomotion-Simulation and Control", in *Proc. 3rd AICA Conference on Analog Computation*, Opatija, Yugoslavia, 1961, pp. 385–390

[Trenkle 1982] Fritz Trenkle, *Die deutschen Funklenkverfahren bis 1945*, AEG-TELEFUNKEN AKTIENGESELLSCHAFT, 1982, Anlagentechnik, Geschäftsbereich Hochfrequenztechnik

[Troch 1977] Inge Troch, „Eine neue Methode der Parameteroptimierung mit Anwendung auf Randwertaufgaben", in *14. seminář MEDA ANALOGOVÁ A HYBRIDNÍ VÝPOČETNÍ TECHNIKA*, Praha 1977, pp. 121–139

[Trubert 1968] Marc R. Trubert, *Use of Analog Computer for the Equalization of Electromagnetic Shakers in Transient Testing*, Jet Propulsion Laboratory, January 1, 1968

[Truitt et al. 1964] Thos. D. Truitt, A. E. Rogers, *Basics of Analog Computers*, John F. Rider Publisher, Inc., New York, December 1960

[Tse et al. 1964] Francis S. Tse, Ivan E. Morse, Rolland T. Hinkle, *Mechanical Vibrations*, Allyn and Bacon, Inc., Boston, Second Printing, August 1964

[Tucker 2002] Warwick Tucker, "A Rigorous ODE Solver and Smale's 14th Problem", in *Foundations of Computational Mathematics* (2002) 2, pp. 53–117

[Ushakov 1958/1] V. B. Ushakov, "Soviet Trends in Computers for Control of Manufacturing Processes", in *Instruments and Automation*, November 1958, pp. 1810–1813

[Ushakov 1958/2] V. B. Ushakov, "Soviet Trends in Computers for Control of Manufacturing Processes", in *Instruments and Automation*, December 1958, pp. 1960–1961

[Valentin 2003] Franz Valentin, *Hydraulik II – Angewandte Hydromechanik*, Skript des Lehrstuhles für Hydraulik und Gewässerkunde der Technischen Universität München, Oktober 2003

[Valisalo et al. 1982] P. E. Valisalo, D. Bergquist, V. McGrew, "A Hybrid Computer Algorithm for Temperature Distribution Analysis of Irregular Two Dimensional Shapes", in *10th IMACS World Congress on System Simulation and Scientific Computation, 1982*, pp. 17–19

[Valisalo et al.] P. E. Valisalo, J. K. LeGro, "Hybrid Computer Utilization for System Optimization", author's archive

[Van Wauve 1962] Armand Van Wauve, *Automatic Analog Computer Control by Means of Punched Cards – "CRESSIDA I"*, National Aeronautics and Space Administration, Washington, D. C., April 1962

[Vichnevetsky 1969] Robert Vichnevetsky, "Use of Functional Approximation Methods in the Computer Solution of Initial Value Partial Differential Equation Problems", in *IEEE Transactions on Computers*, Vol. C-18, No. 6, June 1969, pp. 499–512

[Vocolides 1960] J. Vocolides, „Über die Behandlung linearer algebraischer Gleichungssysteme mit Analogrechnern", in *Elektronische Rechenanlagen*, 2 (1960), No. 3, pp. 136–141

[Vogel 1977] Fritz Vogel, *Ein elektronischer Koordinatenwandler ohne Diodennetzwerke und seine Anwendung bei der Meßwertverarbeitung mechanischer Größen*, Dissertation, Technische Universität Wien, September 1977

[Volder 1959] Jack E. Volder, "The CORDIC Trigonometric Computing Technique", in *IRE Trans. Electron. Comput.*, EC-8, 1959, pp. 330–334

[von Thun] H. J. von Thun, „Simulation einer lagegeregelten Radiostern-Antenne auf dem hybriden Analogrechner", BBC-Mannheim, Zentrale Entwicklung für Elektronik, Abteilung Systemtechnik

[Wagner 1972] Manfred Wagner, *Analogrechner in der Verfahrenstechnik*, VEB Deutscher Verlag für Grundstoffindustrie, Leipzig, 1972

[Walther et al. 1949] Alwin Walther, Hans-Joachim Dreyer, „Die Integrieranlage IPM-Ott für gewöhnliche Differentialgleichungen", in *Die Naturwissenschaften*, No. 7, 1949, pp. 199–206

[Waltman 2000] Gene L. Waltman, *Black Magic and Gremlins: Analog Flight Simulation at NASA's Flight Research Center*, NASA History Division, Monographs in Aerospace History, Number 20, 2000

[Washburn 1962] R. P. Washburn, "Economic-Dispatch Computers for Power Systems", in [Huskey et al. 1962][pp. 5-155–5-160]

[Wass 1955] C. A. A. Wass, *Introduction to Electronic Analogue Computers*, London, Pergamon Press Ltd., 1955

[Weitner 1955] G. Weitner, „Grundschaltungen elektronischer Regler mit Rückführung", in *Elektronische Rundschau*, September 1955, Vol. 9, No. 9, pp. 320–323

[Werrell 1985] Kenneth P. Werrell, *The Evolution of the Cruise Missile*, Air University Press, Maxwell Air Force Base, Alabama, September 1985

[White 1966] M. E. White, "An Analog Computer Technique for Solving a Class of Nonlinear Ordinary Differential Equations", in *IEEE Transactions on Electronic Computers*, Vol. EC-15, No. 2, April 1966, pp. 157–163

[Wierwille et al. 1968] Walter W. Wierwille, James R. Knight, "Off-Line Correlation Analysis of Nonstationary Signals", in *IEEE Transactions on Computers*, Vol. C-17, No. 5, May, 1968, pp. 525–536

[Williams et al. 1958] Theodore J. Williams, R. Curtis Johnson, Arthur Rose, "Computers in the Process Industries", in *Instruments and Automation*, Vol. 31, January 1958, pp. 90–94

[Willers 1943] Friedrich Adolf Willers, *Mathematische Instrumente*, Verlag von R. Oldenbourg, München und Berlin 1943

[Winarno 1982] H. Winarno, J. Jalade, J. P. Gouyon, "Hybrid Simulation of a Microprocessor Controlled Multi-Convertor", 10th IMACS World Congress on System Simulation and Scientific Computation, 1982, pp. 23–25

[Winkler 1961] Helmut Winkler, *Elektronische Analogieanlagen*, Akademie-Verlag Berlin, 1961

[Witsenhausen 1962] Hans S. Witsenhausen, "Hybrid techniques applied to optimization problems", in AIEE-IRE '62 (Spring), Proceedings of the May 1–3, 1962, spring joint computer conference, pp. 377–392

[Woods 2008] W. David Woods, *How Apollo Flew to the Moon*, Springer, Praxis Publishing Ltd., 2008

[Worley 1962] Charles W. Worley, "Process-Control Applications", in [Huskey et al. 1962] [pp. 5-71–5-86]

[Woźniakowski 1977] Miroslav Woźniakowski, "Hybrid and Digital Simulation and Optimization of Dynamic Systems", in *14. seminář MEDA ANALOGOVÁ A HYBRIDNÍ VÝPOČETNÍ TECHNIKA*, Praha 1977, pp. 159–163

[Zachary 1999] G. Pascal Zachary, *Endless Frontier – Vannevar Bush, Engineer of the American Century*, The MIT Press, 1999

[Zingg 1989] David W. Zingg, *Low Mach Number Euler Computations*, NASA Technical Memorandum 102205

[Zuse Z80 1961] N. N., „ZUSE Z 80 – Ein lochendes und druckendes Transistorzählwerk", in *mtw – Zeitschrift für moderne Rechentechnik und Automation*, 1/61, p. 33

Index