ETH zürich

ETH Library

Computerarchitektur aus damaliger und heutiger Sicht

Report

Author(s):
Zuse, Konrad

Publication date:
1992

Permanent link:
https://doi.org/10.3929/ethz-a-000653198

Rights / license:
In Copyright - Non-Commercial Use Permitted

Originally published in:
ETH, Eidgenössische Technische Hochschule Zürich, Abteilung für Informatik, Departement Informatik 180

This page was generated automatically upon download from the ETH Zurich Research Collection.
For more information, please consult the Terms of use.

Eidgenössische Technische Hochschule Zürich

Abteilung für Informatik

Konrad Zuse

Computerarchitektur aus damaliger und heutiger Sicht

August 1992

Authors' addresses:

Prof. Konrad Zuse
Im Haselgrund 21
D-6418 Hünfeld

Prof. Walter Gander
Institut für Wissenschaftliches Rechnen
ETH Zürich
CH-8092 Zürich
e-mail: gander@inf.ethz.ch

© 1992 Departement Informatik, ETH Zürich

Contents

1 Vorwort **3**

2 Computerarchitektur aus damaliger und heutiger Sicht **4**

3 Preface **14**

4 A past and present view of Computer Architecture **15**

1 Vorwort

1991 feierte die Abteilung für Informatik der ETH Zürich ihr 10 jähriges Jubiläum. Bei dieser Gelegenheit wurde dem Computerpionier Professor Konrad Zuse die Ehrendoktorwürde verliehen. Zuse hielt am 25. November 1991 an der ETH Zürich einen Vortrag mit dem Titel "Computerarchitektur aus damaliger und heutiger Sicht". Der Vortrag wurde für die wissenschaftshistorische Sammlung der ETH Bibliothek aufgezeichnet. Wir haben den Vortrag nachträglich mit Hilfe des Vortragenden aufgeschrieben und ins Englische übersetzt. Er ist gerade für die englischsprachige Welt von Interesse, weil dort häufig die Pionierleistungen von Konrad Zuse unbekannt sind. Ich danke Frau R. Oettinger und Dr. Kevin Gates für die Hilfe beim Erstellen und bei der Übersetzung des vorliegenden Berichtes.

Konrad Zuse wurde 1910 in Berlin geboren. Von 1927 an studierte er an der Technischen Hochschule Berlin-Charlottenburg zuerst Maschinenbau, dann Architektur und schliesslich schloss er 1935 als Bauingenieur ab.

Schon während des Studiums hatte er sich über den Stumpfsinn ausgedehnter statischer Berechnungen geärgert. Er begann deshalb schon 1933 mit ersten Überlegungen, wie man solche Berechnungen mechanisieren könnte. 1936 richtete er eine Erfinderwerkstatt in der Wohnung seiner Eltern ein mit dem Ziel, eine programmgesteuerte Rechenmaschine zu konstruieren. Die voll mechanisch arbeitende Z1 wurde 1938/39 fertig und funktionierte im Prinzip - die mangelnde Präzision der mechanischen Teile ergab Probleme.

Eine kürzlich mit modernen computergesteuerten NC-Maschinen gefertigte Rekonstruktion der Z1 funktioniert dagegen im Museum für Verkehr und Technik in Berlin einwandfrei!

Konrad Zuse verbesserte seine Maschine unter Verwendung von Relais. Verschiedene grundlegende Techniken im Bau von Computern wurden von Konrad Zuse erstmals und im Alleingang eingeführt: so z.B. die Verwendung des Dualsystems. Am 12. Mai 1941 konnte Zuse einer Gruppe von Fachleuten seine Z3, den ersten in binärer Gleitpunktarithmetik arbeitenden Computer der Welt, vorstellen.

Nach dem 2. Weltkrieg erkannte Professor Stiefel an der ETH Zürich, die grosse zukünftige Bedeutung der Computer. Es gelang ihm, von Professor Zuse die Z4 für 5 Jahre für die ETH zu mieten. Dies führte zu einer fruchtbaren Zusammenarbeit mit den Professoren Rutishauser und Speiser und war der Grundstein für eine sehr erfolgreiche Entwicklung der Numerischen Mathematik und Informatik an der ETH Zürich.

Zürich, 30. Juli 1992 Walter Gander

2 Computerarchitektur aus damaliger und heutiger Sicht

Ich möchte einige Stichworte bringen. Das Wort Architektur ist auf das Gebiet der Computer erst verhältnismässig spät übertragen worden. Wir können zurückschauen auf Babbage. Sein analytisches Gerät hatte im Prinzip schon eine Architektur. Es hatte ein Programmwerk, einen Lochkartenabtaster, ein Rechenwerk, das von dem Programmwerk gesteuert wurde und das mit einem Speicherwerk verbunden war. Es sollte (wie wir heute sagen) tausend "Worte" speichern. Als ich anfing, meine Ideen zu entwickeln, hatte ich keine Ahnung von diesen Arbeiten, ich verstand kaum etwas von Rechenmaschinen.

Etwa um 1938 führte ich einmal mit einem Rechenmaschinen-Fabrikanten ein kurzes Gespräch, das etwa folgendermassen ablief: "Ich habe gehört, Sie haben eine neue Rechenmaschine erfunden. Ich wünsche Ihnen als jungen Menschen alles Gute, aber eines muss ich Ihnen gleich sagen, auf dem Gebiet der Rechenmaschinen ist bereits alles erfunden! Arbeiten Sie nach dem Prinzip der Einmaleins-Körper oder der wiederholten Addition ?" Das war damals eine Weltanschauung unter den Fachleuten. Ich sagte, dass es bei mir dasselbe wäre. Schliesslich kam er doch zu mir, und ich konnte ihm folgendes erklären:" $0 \times 1 = 0, 1 \times 1 = 1$, wozu brauche ich Einmaleins-Körper? Das ist bei mir dasselbe wie die wiederholte Addition."

Der Umstand, dass ich nichts von Rechenmaschinen verstand, half mir, völlig neue Wege zu gehen. Ich habe also gar nicht erst versucht, die auf dem Markt befindlichen Vierspeziesmaschinen, die zum Teil automatisch arbeiteten, oder Lochkartenmaschinen zusammenzuschalten. Ich sagte mir, dass ich von vorneherein neue Wege gehen müsste. Da war zuerst einmal die Programmsteuerung, das bedeutete im Gegensatz zu bisherigen Rechenmaschinen ein Gerät, das viele aufeinanderfolgende Operationen automatisch ablaufen lassen kann. Das hatte auch Babbage schon vor, wovon ich allerdings nichts wusste. Dann beschäftigte mich die Frage, wie ich die Rechenwerke neu konstruieren könnte. Es war natürlich schon bekannt, dass es nicht nur das Dezimalsystem, sondern unter anderem auch ein Dualsystem oder Zwölfersystem gibt, mit denen man rechnen kann.

Ich erkannte, dass sich hier die Gelegenheit bot, neu anzufangen. Die Idee, Rechenmaschinen im Dualsystem zu bauen, war auch nicht neu, was ich allerdings erst später erfahren habe. Aber lohnte es sich denn, eine Maschine zu haben, bei der jede Zahl zuerst von einem System in das andere System und wieder zurück transferiert werden musste? Dadurch ging der ganze Vorteil des binären Zahlensystems wieder verloren.

Die Programmsteuerung schuf eine völlig neue Situation. Es mussten Tausende von Zwischenwerten, die der Mensch gar nicht zu kennen brauchte, gespeichert werden und so konnte ich die für die Maschine günstigste Form, nämlich das Binärsystem, wählen. Diese Idee hat sich, glaube ich, durchgesetzt, was am Anfang nicht selbstverständlich war. Die ersten amerikanischen Computer arbeiteten nicht im binären Zahlensystem. Aiken war ein strenger Vertreter der dezimalen Systeme und es hat noch lange Dezimal-Maschinen gegeben. Allerdings hat sich durchgesetzt, dass alle mit Bauelementen nur mit zwei Zuständen arbeiten. Dezimalziffern werden mit Ja-Nein-Werten vercoded.

Fast alle Computer sind heute bei Kaufleuten und Verwaltungsleuten eingesetzt. Man muss aber betonen, dass die ersten Computer von Wissenschaftlern und Inge-

nieuren für ihre eigenen Probleme geschaffen wurden. Da kam ein wichtiger Punkt: Der Ingenieur rechnete damals immer noch mit dem Rechenschieber, der immer noch eines der genialsten Rechengeräte ist. Aber die Kommastellung muss man im Kopf rechnen. Die Genauigkeit liegt beim Rechenschieber etwa bei drei Dezimalstellen, was für 90% aller technischen Rechnungen ausreicht. Es musste also jetzt ein anderes Prinzip gefunden werden. Ich wollte zunächst voll logarithmisch arbeiten, ebenso wie der Rechenschieber logarithmisch arbeitet, kam aber bald darauf, dass die Schwierigkeit bei der Addition von Logarithmen liegt. Man kommt in Teufels Küche, wenn man anfängt, gute Geräte für diesen Zweck zu bauen. Ich wählte die halblogarithmische Form, wobei ich nur den vorderen Teil, d.h. den Exponenten logarithmierte, den Rest aber normal darstellte. Das ist dasselbe wie die heute übliche Gleitkommarechnung.

Nach diesem Prinzip entwickelte ich nun verschiedene Geräte. Ich fing an, mechanisch zu bauen, denn damals war eine Rechenmaschine etwas, das man auf den Tisch stellen musste. Ich überlegte mir, dass man das Gerät auch mit elektrischen Mitteln, z.B. den bekannten Fernsprechrelais bauen könnte. Wenn ich aber tausend Worte speichern wollte, würde ich Zehntausende von Relais brauchen. Ich konnte ja nicht einen ganzen Saal mit Relais füllen.

Etwas später war es durchaus selbstverständlich, dass man für ein Gerät, das den ganzen Raum einnimmt, auch ein paar Millionen bezahlte. Zunächst einmal war es aber schwierig, solch einen Gedanken überhaupt zu fassen. Dann habe ich einen Speicher entwickelt, der sehr konzentriert war und die Möglichkeit bot, tausend Worte zu speichern. Dieser Speicher war 1936 im Prinzip fertiggestellt. Ein Nachbau steht im Museum für Verkehr und Technik in Berlin. Er kann dort besichtigt werden und funktioniert auch.

Darauf sagte ich mir, was beim Speicher möglich ist, müsste auch beim Rechenwerk gehen. Ich habe dann angefangen, das Rechenwerk für das binäre Zahlensystem und mit Gleitkommaoperationen in einer von mir entwickelten mechanischen Schaltgliedtechnik zu bauen. Auch dieses Gerät steht als Nachbau in Berlin. Aber sowohl damals als auch heute funktionierte das Rechenwerk nicht gut. Dies ist wahrscheinlich ein Punkt, an den mancher Erfinder einmal kommt. Er ist einen bestimmten Weg gegangen und ist zu 80% fertig und er erkennt, dass es bessere Wege gibt.

Inzwischen hatte ich nämlich von diesem Modell viel gelernt. Die Rechenwerke müssen mit elektrischen Mitteln gebaut werden. Wenn ich eine Nachricht zwischen weit entfernten Stellen übertragen will, genügt bei der elektrischen Lösung ein Draht. Mechanisch erfordert das sehr umständliche Übertragungen. Also ging ich zur Relais-Technik über. Ich habe dann zunächst einmal ein kleines Zwischenmodell, das Modell Z2, gebaut. Dieses Modell erfüllte lediglich den Zweck zu zeigen, dass es überhaupt mit Relais gehen kann. Dann baute ich das Gerät Z3 vollständig in Relaistechnik und tatsächlich ging das verhältnismässig schnell.

Durch die Arbeit an den Geräten Z1 und Z2 kam ich auf neue Ideen. Ich entwickelte eine "Bedingungskombinatorik" und erfuhr erst später, dass diese Bedingungskombinatorik dasselbe wäre wie der Aussagenkalkül der mathematischen Logik. Wir sprechen heute von Bool'scher Algebra. Von meiner Sicht aus gesehen war es eine der wichtigsten Entdeckungen, dass alles mit Ja-Nein-Werten gemacht werden kann, was zunächst gar nicht so selbstverständlich war. Zahlen kann man

im binären System darstellen, aber wie steht es mit all den komplizierten Steuerungen?

Ich musste die Gleitkommaform im Dezimalsystem in eine Gleitkommaform im binären Zahlensystem umrechnen. Das erfordert viele bedingte Befehle, und es war nicht klar, ob das alles nur mit Relais gemacht werden könnte. Die Entdeckung, dass das möglich war, war für mich eine Art Offenbarung und nun konnte eine Relaismaschine gewissermassen vom Papier weg konzipiert werden. Wie ein Mathematiker konnte man seine Schaltungen in logischen Formeln hinschreiben, dann den Lötkolben nehmen und sie sofort verdrahten. Die elektromagnetische Relaistechnik erlaubt das. Die elektronischen Schaltungen, die dann später entwickelt wurden, sind nicht ganz so einfach in den Übersetzungen vom Papier in die tatsächlich arbeitende Schaltung. Jetzt hatte ich also den Aussagenkalkül von der mathematischen Logik übernommen und studierte jetzt weiter den Relationenkalkül, den Prädikatenkalkül usw.

Es zeigte sich, dass hier eine ganz neue Welt des Rechnens erscheint. Heute ist uns diese Verbindung zwischen mathematischer Logik und der Konstruktion von Rechenmaschinen selbstverständlich. Wir müssen uns jedoch einmal zurückversetzen in die damalige Zeit, etwa um 1938. Damals gab es die reinen Mathematiker. Auf das Wort "rein" legten sie ganz besonderen Wert. Es gab links von ihnen die angewandten Mathematiker, die nicht besonders angesehen waren. Rechts von den reinen Mathematikern gab es die mathematischen Logiker. Die waren so superrein, dass man selbst an den Vorlesungen, die damals an den Hochschulen über Mathematik gehalten wurden, kaum etwas darüber erfuhr. Im Gegensatz zu den angewandten Mathematikern waren sie die "abgewandten" Mathematiker, da sie an Problemen arbeiteten, die mit dem täglichen Leben kaum etwas zu tun hatten. Es handelte sich im wesentlichen um den Beweis mathematischer Sätze. Noch weiter links von den angewandten Mathematikern kamen dann die Rechenmaschinenleute. Die waren so kühn, ganz an den rechten Flügel zu gehen, die mathematische Logik zu übernehmen und sie auf die ganz linke Seite, die Konstruktion von Rechenmaschinen anzuwenden.

Aus dieser "Ehe" ist die heutige Informatik entstanden. Es wurden nicht nur die angewandten Mathematiker langsam gesellschaftsfähig, es gab dann unter den Informatikern auch wieder die reinen, theoretischen Informatiker und die angewandten Informatiker. Diese waren zunächst nicht gut angesehen. Heute haben sich die angewandten Informatiker aber durchgesetzt.

Das hatte verschiedene Konsequenzen. Es ist heute Mode, dass man von der Verantwortung des Wissenschaftlers spricht. Man sagt, wenn ein Wissenschaftler oder Erfinder eine neue Idee hat, müsste er eigentlich die Öffentlichkeit fragen: "Darf ich das überhaupt?" Nun, es gab damals schon Institutionen, die sich mit zukünftigen Erfindungen beschäftigten. Es handelte sich dabei um Raketen und so fort, von Computern oder Rechenmaschinen war keine Rede. Das bedeutet, wenn ich zur damaligen Zeit gesagt hätte, ich habe mit der mathematischen Logik eine neue Welt des Rechners erschlossen, vielleicht kann ich eines Tages mit der Maschine Schach spielen, dann hätte man das überhaupt nicht ernst genommen.

In dem Augenblick, in dem eine solche Maschine entsteht, die vielleicht einmal revolutionär sein kann, ist das öffentliche Interesse noch nicht geweckt. Ich selbst habe das jedoch sehr ernst genommen. Es stellte sich heraus, dass alles, was man

heute “Informationen” nennt, in Ja-Nein-Werte aufgelöst werden kann. Da die Programme auch in einer Ja-Nein-Vercodung geschrieben waren (das war damals schon selbstverständlich), war es auch klar, dass ein Programm auf sich selbst einwirken können muss und sich selbst ändern können muss .

Diese Idee war mir anfangs offen gestanden ziemlich unheimlich. Denn bis dahin konnte man die von mir entwickelten Geräte Z1–Z4 ziemlich gut überblicken und abschätzen, was sie tun würden. Ebenso konnte man den Ablauf der Rechnungen einigermassen verfolgen. In dem Augenblick, in dem ich den Einfluss der errechneten Daten auf das Programm selbst zulasse – dazu gehört nur ein kleiner Draht, der vom Rechenwerk rückwirkend auf das Programmwerk wirkt – ist diese Kontrolle nicht mehr gegeben. Vor diesem kleinen Draht hatte ich viel Respekt, weil ich ahnte, dass, sobald dieser Draht gelegt ist, Mephisto hinter mir steht. Dann bin ich nämlich nicht mehr in der Lage, stets zu überblicken, was der Computer macht. Tatsächlich ist es so, dass der völlig freie “go to”-Befehl von Mephisto erfunden worden ist. Damit kann man nämlich die tollsten Sachen machen!

Ich versuchte also, eine weitere Grenze zu finden. Ich wollte erreichen, dass man dies und jenes nicht darf, aber ich sah so eine Grenze nicht. Ich überlegte mir weiter, was ich jetzt alles machen kann. Das Schachspiel müsste möglich sein. Ich entwarf Programme für das Schachspiel und etwa um 1938 sagte ich eher scherzhaft zu einigen Freunden: “In fünfzig Jahren wird der Schachweltmeister durch eine Rechenmaschine besiegt!” Ich hatte Pech: Das Jahr 1988 ist vorübergegangen und unser Schachweltmeister Kasparov ist immer noch nicht von einer Rechenmaschine besiegt worden. Aber er sagt heute nicht mehr, dass er **nie** von einer Rechenmaschine besiegt werden wird! Das hat er sich schon abgewöhnt. Und wenn Sie dem Ingenieur 10% Toleranz zugestehen, wie es in der Technik üblich ist, dann kann es durchaus noch passieren, dass meine Aussage stimmt. Tatsache ist, dass die Computer gegen Spieler der Meisterklasse schon sehr gut funktionieren. Es ist dabei bemerkenswert, dass der Computer trotz der enormen Steigerung der Geschwindigkeit (millionen- und milliardenfach) den Schachweltmeister immer noch nicht besiegt hat. Das ist eigentlich eine Blamage der Computerkonstrukteure und der Fachleute der “künstlichen Intelligenz”! Mit dem kleinen Gehirn, das langsamer arbeitet als heutige Rechenmaschinen (es schaltet nicht in Nanosekunden)! Das heisst also, die Leistung des menschlichen Gehirns wird vom Computer heute noch lange nicht erreicht. Das gilt insbesondere für die künstliche Intelligenz. Man arbeitet beispielsweise beim Schachspiel mit den sogenannten “brutalen Methoden”. Man rechnet also sämtliche Möglichkeiten, etwa bis zum vierten Zug, aus. Man kommt sehr schnell dahin, dass es auf diese Weise unmöglich ist, das beste Programm zu errechnen. Damit hat man aber auch nicht den Weg, den der gute Schachspieler intuitiv geht. Ich glaube, wir sollten froh und stolz sein, dass wir in unserem Gehirn Schaltungen haben, die wesentlich flexibler sind als die Schaltungen der besten Computer. Das war die Situation, in der ich mich damals befand und ich merkte, dass ich ein heisses Eisen angepackt hatte.

Jetzt stellt sich die Frage, was man mit dieser “Rückwirkung” machen kann. Da gibt es zum Beispiel bedingte Sprünge und Adressenumrechnung. Es war sehr schnell klar, dass das möglich war. Damals war es noch nicht klar, ob die elektronischen Maschinen wirklich funktionieren würden. Etwa 1937 kam ein guter Freund, Helmut Schreyer, zu mir in meine Werkstatt. Ich dachte, er würde mir helfen, an

den Blechen zu sägen, um meine mechanische Rechenmaschine zu bauen. Da sagte er so einfach daher: "Das musst Du mit Röhren machen!". Er war ein lustiger Bursche und auch das Studentenleben war damals noch verhältnismässig lustig. Wir haben gekneipt und standen auch mit der Polizei gut, wir wurden nicht gleich für jeden Studentenunfug eingesperrt und wenn doch, war es ein grosses Hallo. Ich dachte mir also, das sei sein neuester Studentenulk, dass er mit Röhren arbeiten will! Wir haben uns dann aber überlegt, dass die Idee nicht schlecht ist.

Inzwischen war die Schaltalgebra auch an anderer Stelle erfunden worden (wovon ich nichts wusste, ich habe sie für meine Geräte selbst entdeckt, auch von den Überlegungen von Turing wusste ich nichts). Das ist dann alles etwa gleichzeitig entstanden. Das bedeutete, dass man lediglich die Grundoperationen des Aussagenkalküls schaltungsmässig lösen musste. Wenn das mit Röhren gelang, müsste es möglich sein, die elektromagnetisch arbeitende Relaismaschine in die elektronische Maschine umzusetzen. Dieses Eins-zu-Eins Umsetzen in die Elektronik wurde auch später auf der anderen Seite des Atlantik versucht. Bei dem Gerät ENIAC hat man die traditionellen Ziffernrädchen durch Röhrenschaltungen simuliert. Dadurch entstanden die enormen Zahlen von Röhren. Wir selbst kalkulierten, wieviele Röhren wir brauchen würden. Helmut Schreyer machte erst einmal einige Vorversuche (er hat seine Doktorarbeit darüber gemacht). Wir schätzten nun, dass wir etwa 2000 Röhren brauchen würden, um eine brauchbare elektronische numerische Rechenmaschine zu bauen. Dies haben wir am Institut von Schreyer, am Institut für Strömungsforschung, vorgetragen. Da waren viele Fachleute, die täglich mit Röhren arbeiteten. Und gerade diese sagten: "Ihr wollt mit zweitausend Röhren eine Maschine bauen? Das kann nie funktionieren!". Sie hatten, wie überall auf der Welt, täglich ihren Ärger mit Röhren.

Bei den Plänen zu ENIAC hatte man das gleiche gesagt. Die Amerikaner waren so kühn, gleich mit 18'000 Röhren zu arbeiten. Auf diese Idee wären wir nie gekommen! Als wir nach dem Krieg erfuhren, dass man dort drüben eine elektronische Maschine mit 18'000 Röhren gebaut hatte, haben wir den Kopf geschüttelt und uns gefragt, was die mit den vielen Röhren machen. Aber es war jetzt wenigstens klar, dass ein neuer Weg gefunden worden war. Es stand nun in Aussicht, dass man elektronisch arbeiten kann.

Wir machten den Vorschlag, eine Maschine mit 2000 Röhren zu bauen, bei einer Regierungsstelle. Das war ungefähr 1941. Man fragte uns, wie lange wir dazu brauchen würden. Wir antworteten, dass wir mindestens zwei Jahre brauchen würden, was sehr knapp geschätzt war. Man gab uns zur Antwort: "Was glauben Sie, wie lange es dauert, bis wir den Krieg gewonnen haben?" Damit war der Fall erledigt. Es kommt dazu, dass ich (wie auch Schreyer) für die Dauer des Krieges nicht etwa für die Entwicklung von Computern vom Militärdienst freigestellt war, sondern für militärische Aufgaben. Ich war Ingenieur im Flugzeugbau und Schreyer arbeitete an nachrichtentechnischen Problemen.

Die deutschen Ingenieure waren natürlich an einer solchen Rechenmaschine sehr interessiert. Wir waren sehr bescheiden und sagten, wir könnten tausend Mal so schnell rechnen wie die traditionellen Maschinen. Wir konnten jedoch nur wenig versprechen, da die Umstände nicht gegeben waren, um ein solches Gerät zu bauen. Trotzdem waren die Verlockungen, mit den Möglichkeiten der mathematischen Logik zu arbeiten, sehr gross, da damit eine ganz neue Welt des Rechnens ent-

stand. Die Maschinen, die ich bisher gebaut hatte, waren die Z1–Z4. Die Z4 konnte als einzige Maschine aus dem Krieg gerettet werden. Sie wurde später an der ETH in Zürich aufgestellt. Diese Maschinen konnten, wie man heute sagt, nur "geradeaus" rechnen.

Ich war damals Bauingenieur und hatte mit Stabwerken und Rahmen zu tun und wusste, dass es über das rein numerische Rechnen hinaus möglich sein müsste, der Maschine Punkt für Punkt die Struktur eines statischen Systems einzugeben. Daraus musste sich automatisch mit Hilfe der logischen Operationen, dem Relationenkalkül oder der Graphentheorie (den Ausdruck gab es damals noch nicht) die Möglichkeit ergeben, den numerischen Plan herzustellen. In dieser Situation glaubte ich damals, dass es richtiger ist, beide Aufgaben zu trennen. Es gab zum einen die sehr leistungsfähige numerische Rechenmaschine, die man damals noch mit Relaistechnik baute. Man wusste nicht, ob die Röhre wirklich funktionieren würde. Zum anderen eine logistische Rechenmaschine. Damals war der Ausdruck "logistisch" für die mathematische Logik reserviert. Das nannte ich das "Planfertigungsgerät", heute würde man Programmfertigungsgerät sagen. Diesem sollte die Aufgabe überlassen bleiben, aus dem Aufbau des Systems, das man durchrechnen will, die numerischen Programme zu entwickeln. Das brauchte man nur einmal zu machen, dann konnte man die numerischen Programme vielfach mit verschiedenen Werten durchspielen. Das war eine Kompromissidee und ich hatte mich damit sehr intensiv befasst.

Das war in etwa die Lage um 1944. Es war die Entdeckung einer neuen Welt des Rechnens mit ungeheuren Möglichkeiten, wobei Adressenumrechnungen und bedingte Befehle nur eine geringe Rolle spielten. Man muss auch wissen, was die technische Seite war. Das Speichern war gelöst und der Gedanke, dass man auch Programme speichern könnte, implizierte eine Unmenge von Möglichkeiten. Ich habe damals ein Gespräch mit meinem Patentanwalt geführt und sagte, ich möchte eine Rechenmaschine anmelden (ein Computer war bereits angemeldet), wo ich die Programme speichern kann. Da sagte er: "Das ist im Sinne des Patentgesetzes keine Erfindung! Die Frage ist auch, ob es zur Architektur gehört. Der Gedanke ist selbstverständlich sehr brauchbar, aber es geht um die Frage des "wie?". Sie müssen mir sagten, *wie* die Maschine aufgebaut ist und wie sie Programme modifiziert. Die Tatsache, dass man Programme speichern kann, ist nicht patentfähig.".

An dieser Anmeldung habe ich während des Krieges gearbeitet, die Unterlagen davon sind aber verloren gegangen. Ich habe einmal eine Taschenbuchskizze gemacht, die anfing bei Babbage und da aufhört, was wir heute John von Neumann Architektur nennen. Ich glaube nicht, dass man dem Gedankengang John von Neumanns gerecht wird, wenn man sagt, er hätte die Speicherprogrammierung erfunden. Die war damals schon bekannt. Die Genialität von Neumanns bestand darin, dass er genau in dem Augenblick dazu kam und mit sicherem Blick aus der Fülle der Möglichkeiten das herauspickte, was wirklich wichtig war. Das ist nämlich sehr wichtig. Bei allen Neuentwicklungen gibt es sehr viele Möglichkeiten und man braucht einen scharfen Blick um zu sehen, was in diesem Moment gerade das Richtige ist.

Folgende Voraussetzungen sind erforderlich: die programmgespeicherte Maschine muss schnell genug sein, um die erhöhte Zahl von Befehlen durchzuführen. Dieser Gedanke, die Relaismaschine eins-zu-eins in die Röhrenmaschine zu übertragen

(damals hatte man nur die Röhren) war im Grunde richtig, aber in sofern auch wieder falsch, als man die neuen Möglichkeiten der Röhre nicht wirklich ausnutzte. Die Röhre kann viel schneller arbeiten. Das, was in der Relaismaschine mit sehr komplizierten Schaltungen gemacht wurde, nämlich die Übersetzung und die Steuerung der Mikroprogramme (die erste Mikroprogrammaschine war übrigens die Z1, die machte das alles mit mechanischen Mikroprogrammen).

Das Eins-zu-Eins Übersetzen der Röhrentechnik war sehr aufwendig. Aber man war jetzt schon tausendmal schneller. Wenn man jetzt also beispielsweise die zehn- oder zwanzigfache Zahl von Befehlen für die Gleitkommaarithmetik brauchte, spielte das keine Rolle. Man war immer noch fünfzig mal schneller als die traditionellen Maschinen. Diese Voraussetzung musste erst da sein, bevor es sich lohnte, eine Maschine mit gesteigerten Kosten zu bauen. Ausserdem musste man ein Speicherwerk mit entsprechend hoher Kapazität haben. Was noch Babbage im Auge hatte und was auch für meine ersten Maschinen galt, war das Speichern von tausend Worten. Damit konnte man natürlich schon viele automatische Umrechnungen machen, aber für eine wirkliche Ausnutzung der Möglichkeiten der Programmspeicherung reichte das noch nicht, weil man sehr viel mehr Befehle brauchte und eine entsprechende Speicherkapazität haben musste. Dieser Punkt war technologisch gerade in dem Augenblick erreicht, als von Neumann zu dieser Entwicklung hinzustiess . Er hat sofort erkannt, welcher Weg gegangen werden muss, und dieser Weg hat sich dann auch bewährt.

Es war bei uns so, dass wir 1945 zunächst einmal nicht an der Hardware weiterarbeiten konnten. Wir waren bis zum Gerät Z4 gekommen, das eine Relaismaschine war, die ausserordentlich kompliziert geschaltet war. Ich glaube, wir haben unter uns Professor Speiser als Zuhörer. Er hat damals an dem Gerät mitgearbeitet. Ich glaube nicht, dass es noch eine zweite Maschine gegeben hat, die die Möglichkeiten der Schaltungsmathematik mit all ihren verschlungenen Vernetzungen usw. derartig ausgenutzt hat wie diese Maschine. Wie bereits gesagt, war es jedoch eine Maschine, die geradeaus rechnete und eine Relaismaschine, die ein verhältnismässig ruhiges Tempo hatte (sie konnte ungefähr 20-30mal so schnell rechnen wie der Mensch). Das war zur damaligen Zeit, also etwa 1940-1945, schon eine ganz ordentliche Leistung.

Ich kann hier vielleicht noch etwas einflechten. Ich war Bauingenieur und wollte die Arbeit für die Bauingenieure erleichtern. Dazu ist es aber erst sehr viel später gekommen. Die ersten, die echtes Interesse an der Maschine hatten, waren die Flugzeugbauer und insbesondere die Flatterfachleute. Die Statik des Flugzeuges war damals schon gut bekannt und man konnte ein Flugzeug auch gut durchrechnen, wenn es in Schalenkonstruktion gebaut war. Es traten jedoch eigenartige Erscheinungen auf. Es kam vor, dass Flügel plötzlich brachen und die Piloten abstürzten, so dass niemand sagen konnte, was eigentlich passiert war. Das war eine tragische Situation.

Es kostete lange theoretische Untersuchungen der Aerodynamiker um herauszufinden, was passierte, wenn ein Flügel mit Leitwerk anfing zu flattern. Wenn ein Flugzeug erst einmal in solch einen instabilen Bereich kommt, ist sehr schnell alles aus. Es war dank dem Verdienst von Küssner und anderen zu diesem Zeitpunkt gerade gelungen, das Problem theoretisch zu lösen und auch den rechnerischen Weg dazu zu zeigen. Das lief im wesentlichen auf die Ausrechnung einer

Determinante mit komplexen Gliedern und Unbekannten in der Hauptidealen hinaus. Das war eine komplizierte Rechnung, wofür der Mensch recht lange brauchte. Genau in diesem Augenblick kam ich mit meiner Maschine. Mit der Z3, die 1941 fertig war, sind solche Rechnungen dann auch schon durchgeführt worden.

Auch später zeigten noch vor den Bauingenieuren die Optiker ihr Interesse. Die Optiker haben schon lange bevor es Computer gab sehr umfangreiche Rechnungen durchgeführt. Sie hatten zum Teil bis zu 60 Rechner (Personen!) in einem Raum und es rechneten bis zu zehn Leuten an einem einzigen Objektiv. Es musste Strahl für Strahl durchgerechnet werden (durch einen "6,8,10-Flächner" wie man es nannte). Sie erkannten als erstes, dass sie den Computer gebrauchen könnten. Das war etwa um 1950 und die deutsche optische Industrie war damals noch sehr stark (die Japaner waren noch nicht auf dem Markt). Sie waren sehr daran interessiert, diese Rechnungen durchzuführen.

Es ist vielleicht für die gesamte Entwicklung typisch (heute ist es uns selbstverständlich), dass ein Computer nach dem von Neumann-Prinzip bedingte Operationen und vieles mehr machen kann. Die Optiker wollten die Strahlendurchrechnung haben durch eine Reihe von Flächen. Jedes Objektiv hat mehrere Flächen. Das konnten wir vorführen und es funktionierte. Wozu ein Mensch ein bis zwei Stunden brauchte, das konnten wir in wenigen Minuten. Wir sagten ihnen, dass wir noch viel mehr könnten. Wir könnten die Kurve des Astigmatismus entwickeln und daraus wiederum Verbesserungen der Linsenanordnung finden. Sie sagten jedoch, dass das nicht die Maschine machen sollte, sondern dass sie das selbst machen wollten. Das ist damals unter den Optikern (und nicht nur unter ihnen) die Einstellung gewesen. Wir haben heute derartig komplizierte Programmsysteme, dass man nach den Leuten suchen muss, die überblicken können, wie sie funktionieren. Das wollten die Ingenieure und Wissenschaftler damals auf keinen Fall aus der Hand geben, grundsätzliche Überlegungen wollten sie lieber selbst machen.

Es gab eine zweite Gruppe, die sehr interessiert war, und das waren die Geodäten, die auch sehr viel zu rechnen haben. Zunächst einmal wollten auch sie nur einen gewissen Satz von Formeln gerechnet haben und die schwierigen Operationen durch eigene Überlegungen machen. Damals war zum Beispiel die geodätische Zeichnung mit Punkten und Polygonen ein Dokument, welches juristische Bedeutung hatte. Wir konnten die neuen Karten mit Strassenkreuzungen usw. rechnen und jetzt kamen die Geodäten und sagten, wir wollen das auch gleich gezeichnet haben. So ist die Computergraphik entstanden. Sie waren die ersten, die mit dem Wunsch an uns herantraten, das berechnet zu haben. Sie stellten äusserste Ansprüche an die Genauigkeit. Dass man Zeichenmaschinen mit Schrittmotoren bauen kann war an sich schon bekannt. Das war aber nicht genau genug. Wir machten auch da zunächst einen Umweg über eine mechanische Konstruktion. Diese Graphomaten haben wir dann in grösserer Stückzahl auf den Markt gebracht. Sie waren wegen ihrer Genauigkeit sehr lange in Betrieb, obgleich es auf dem Markt schon schnellere Geräte gab.

Da wir gerade von Computergraphik sprechen, kann ich Ihnen eine andere tragische Geschichte erzählen. Das war alles schon später, als wir bereits elektronische Maschinen bauten, etwa Ende der fünfziger Jahre. Da kam die grösste Teppichfirma Deutschlands zu uns und fragte an, ob man den Vorgang vom Entwurf des Künstlers bis zu den Karten der Kartenschlagmaschine automatisieren könne. Ich

sagte ihnen, dass wir das machen könnten. Das beste wäre aber, bereits beim Entwurf des Teppichs mit dem Computer anzufangen. Das hörten die dort tätigen Künstler, und damit war die Sache erledigt! Das war vor etwa dreissig Jahren. Wenn man nun Unternehmer ist, dann ist man ein Sklave seines Unternehmens. Die Sache erschien äusserst aussichtsreich, warum also haben wir das nicht weiterverfolgt? Das kann man nicht durchführen, wenn man nicht den Auftrag hat, es zu einem bestimmten Termin fertigzustellen. Man kann nicht einfach zwei Mathematiker daransetzen. Es kamen dann die Stahlfirmen, die andere Probleme hatten, die unbedingt gelöst werden mussten. So blieb die Computergraphik in der Schublade liegen. Heute ist sie jedoch grosse Mode.

Damals hatte ich das Pech des "zu früh Gekommenen". Wenn man zu früh kommt, hat man wenig Aussicht auf Erfolg. Die Kunst ist, im richtigen Augenblick mit einer Idee zu kommen. Nehmen wir zum Beispiel das Anti-Blockier-System des Autos. Es liess sich schon sehr früh voraussehen, dass man mit dem Computer die Bremsstärke sehr fein kalkulieren können müsste. Aber dann stellte sich die Frage, wieviele Transistoren dafür nötig wären. Man erkannte, dass man nicht das ganze Auto mit Transistoren vollpacken könnte und damit war die Sache erledigt. Diese an sich brauchbare Idee musste auch erst reifen. Erst mit der weiteren Entwicklung der Technologie konnten weitere Träume der ersten Computerentwickler verwirklicht werden.

Es besteht immer ein enger Zusammenhang zwischen dem Stand der Technik und den logischen Möglichkeiten. Die Entwicklung läuft über die Mechanik, Elektromechanik, Röhre, Transistor, integrierte Schaltkreistechnik bis zu den modernen miniaturisierten Techniken. Da gibt es heute natürlich ganz andere Möglichkeiten. Dinge, die zu den kühnsten Träumen der ersten Pioniere gehörten, sind heute möglich. So besteht also ein Zusammenhang zwischen der Computerarchitektur und der Entwicklung der Technologie.

Sie wissen, dass die Verkleinerung eigenartigerweise auch mit einer Preisermässigung zusammengeht, was nicht selbstverständlich ist. Die ersten kleinen Computer wurden für Raketenfachleute gebaut, damit sie ihre Computer in die Rakete hineinbauen konnten. Diese Computer waren noch teurer als die grossen. Aber daran wurde gearbeitet und heute haben wir nicht nur eine Verkleinerung und eine Steigerung der Geschwindigkeit und Leistungsfähigkeit, sondern auch eine Ermässigung des Preises.

Durch die Presse gehen Nachrichten über das Megachip (das 4-Megachip, das 64-Megachip usw.). Der Laie fragt sich, wozu man eigentlich so grosse Speicher braucht, ob das wirklich so wichtig ist? Seinerzeit haben Wissenschaftler und Ingenieure die entscheidenden Impulse gegeben, damit Computer überhaupt entwickelt wurden. Nachdem es dann gut funktionierte, kamen die Kaufleute und haben die Entwicklung in die Breite gehen lassen. Und auch heute ist es so, dass die Entwicklung zu immer leistungsfähigeren Chips und Speicherkapazitäten in erster Linie von Wissenschaftlern (in diesem Falle sind es Physiker und Chemiker) vorangebracht wird, da sie diese ungeheuren Kapazitäten brauchen. Die Forschung kommt an Punkte (Differentialgleichungen, Gitternetze usw.), wo man sehr grosse Speicherkapazitäten braucht.

Es ist sicher diesem und jenem die Schrödingergleichung geläufig. Diese ist im Prinzip nicht sehr schwer zu verstehen. Das Problem dabei ist nur, dass man

sie mit vielen Dimensionen im Phasen- und Konfigurationsraum durchrechnen muss. Dadurch kommt man schnell auf beliebig viele Gleichungen. Da können die Computerkonstrukteure die Leistung ihrer Maschinen ruhig noch um einen Millionenfaktor erhöhen. Die Physiker finden immer Probleme, für deren Berechnung sie diese Maschinen brauchen können. Auch das wird dann in die allgemeine Praxis übergehen. Verwaltungsleute werden diese Möglichkeiten aufgreifen.

Wir sind heute zum Glück so weit, dass wir den europäischen Gedanken pflegen können, dass wir nicht mehr quer durch Europa Schützengräben ziehen. Diesen Gedanken haben wir hoffentlich überwunden. Ob daraus wirklich eine echte Zusammenarbeit entsteht, ist ein weiteres Problem. Der Wille dazu ist vorhanden, aber es gibt nicht nur Europa auf der Erde, vielmehr ist der gesamte Globus von Computernetzen überzogen. Es ist typisch, dass der Ingenieur und Wissenschaftler nicht an Dinge wie Viren und Computerkriminalität gedacht hat. In seinem Gehirn existiert das einfach nicht. Es sind heute ganz neue Probleme wie der Datenschutz auf uns zugekommen. Es muss enorm viel getan werden, damit wir nicht Opfer unserer eigenen Ideen werden.

Wie ich bereits am Anfang sagte, ist der freie "go to"-Befehl von Mephisto erfunden worden. Wir merken das heute. Ich kann mit diesem freien "go to"-Befehl in fremde Datenbanken einsteigen und so weiter. Diese Möglichkeiten existieren und es ist gut, dass es Leute gibt, die uns verbotenerweise die Schwachstellen zeigen. Die jungen Informatiker, die hier anwesend sind, tragen dabei mit die Verantwortung. Wir Pioniere haben seinerzeit die Sache in Gang gesetzt, haben viele Gefahren gesehen und waren nicht in der Lage, jeder dieser Gefahren auszuweichen. Es ist heute wohl nicht mehr denkbar, den Schritt zurück zu tun. Sie werden in der Zukunft Netze entwickeln, die sehr unübersichtlich sind und wir brauchen da die Leute mit dem Durchblick, um die komplizierten Verschachtelungen zu entflechten. Man kann nicht einfach alle Schuld auf die Erfinder der Computer schieben. Erst die junge Generation kann uns die neuen Netze und Computersysteme liefern, die wir brauchen und hinter denen dann nicht unbedingt Mephisto steht. Ganz werden wir ihn jedoch nicht los!

3 Preface

1991 the section of Computer Science at ETH Zürich celebrated its 10 anniversary. On this occasion the computer pioneer professor Konrad Zuse was given an honorary degree. Zuse gave a seminar on November 25, 1991 at ETH Zürich with the title: "Computerarchitektur aus damaliger und heutiger Sicht". It was taped by the library of ETH for their section of science history. From this tape we have typed the talk with the help of the speaker and translated it to English. Especially for the English speaking world this talk may be interesting, since often there the pioneering work of Zuse is unknown. I'd like to thank Mrs. R. Oettinger and Dr. Kevin Gates for the preparation and the translation of this report.

Konrad Zuse was born in 1910 in Berlin. 1927 he started his studies at the technical university of Berlin-Charlottenburg. First he was interested in machine engineering, then he switched to architecture and finally got his degree in 1935 in civil engineering.

Already as a student he was annoyed with the tedious calculations one had to perform for statical stress computations. He therefore started already in 1933 to think how such calculations could be performed in a mechanical way. 1936 he installed a laboratory in the living room of the apartment of his parents with the aim, to construct a programmable computing machine. The fully mechanically operating Z1 machine was finished 1938/39 and worked in principle – the precision of the mechanical parts was not good enough and lead to problems.

Recently a replica of the Z1 has been made for the "Museum für Verkehr und Technik" in Berlin. For this replica numerical controlled machines were used to produce the parts. Now the precision is good enough and the machine is operational!

Konrad Zuse improved his machine by using relais. Several fundamental techniques used for building computers were introduced by Zuse for the first time and by himself: e. g. the use of the binary system. On May 12, 1941, Zuse showed to a group of experts his Z3, the first computer of the world working in binary arithmetic.

After the second world war professor Stiefel realized the great importance of computers in the future. He managed to rent for ETH the Z4 from professor Zuse for 5 years. This lead to a very fruitful collaboration with the professors Rutishauser and Speiser and was the beginning of a very successful development of numerical analysis and computer science at ETH Zürich.

Zürich, July 30, 1992 Walter Gander

4 A past and present view of Computer Architecture

I want to start with a few keywords. Architecture has only recently been used in computer science. Consider the work of Babbage, his adding machine had in principle an architecture. It had an instruction stream, a punch card reader, and a computational stream that was controlled by the instruction stream and had access to memory. It should have be able to, as we say today, store a thousand words. As I started to develop my ideas, I had no knowledge of Babbage's work and I knew almost nothing about calculating machines.

Around 1938 I had a short conversation with a manufacturer of calculating machines. It went as follows: "I heard, that you discovered a new calculating machine. Young man, I wish you luck, but I have to tell you, everything about calculating machines is already known. Are you working with the principle of the multiplication table or repeated addition?" That was the world view of specialists at that time. For me, adding and multiplying were the same thing, and I said so. Finally, he came to me and I explained the following: " $0 \times 1 = 0, 1 \times 1 = 1$, why do I need the multiplication tables?"

My lack of knowledge about calculating machines helped me to break new ground. My goal in starting out was not to connect existing partially automatically working mechanical calculating machines or punch card readers. I told myself, that I must go a totally different direction. That direction was instruction streams, which means, in contrast to earlier calculating machines, a machine which automatically executes a string of operations. I didn't know that Babbage had the same idea. I then became involved in the question of how to construct such a computing machine. It was of course already known that besides the decimal system, one could calculate with other number systems like the binary or hexadecimal system.

I realized that this was a possibility for a new start. The idea of building binary computing machines was not new (a fact that I later learned), but was it worthwhile to have a machine where every number is converted from one number system into another number system and then back again? Through this transformation the benefit of the binary system would be lost.

The use of instruction streams created a completely new situation. Thousands of intermediate values, that one does not need to know, must be saved for later use. I selected the binary system for my machine, since it was convenient. I think this idea has proven itself, although in the beginning it was not at all clear. The first American computer did not use the binary system. Aiken was a strong proponent of the decimal system and for a long time there were decimal-based machines. It is now accepted, I think, that all computers work in the binary system where there are only two states. Decimal numbers are stored with an encoding of 0 and 1.

Today most of the computers are used by salespeople, administrative and office personnel. However, I would like to emphasize, that the first computers were created so that researchers and engineers could solve problems. This is an important point. Engineers used to calculate with slide rules. The slide rule is still one of the most ingenious calculating tools; however, the user must calculate the decimal point in his head. The accuracy of a slide rule is around three decimal digits, which is good enough for 90 % of the technical calculations. Another principle had to

be discovered. I first wanted to work totally with logarithms, just like the slide rule does. It was soon clear that the problem with logarithms is addition. A man arrives in the devil's kitchen when he tries to make a machine for this purpose. I chose the half-logarithmic form where I stored only the exponent in a logarithmic form, which is the same as the floating point numbers used in today's computers.

Following this principle I developed new machines. I started with a mechanical calculating machine. It occurred to me that one could also build this machine with electrical parts, for example electrical relays. However, it would have required tens of thousands of relays for a thousand words. I simply could not fill an entire room with relays!

Somewhat later it was clear that people would pay a couple of millions for a machine that filled a room. In the beginning this situation was difficult to imagine. Therefore I developed a storage utility that was very concentrated and enabled the storage of a thousand words. This storage device was in principle completed in 1936. A functioning reproduction of this machine is in the "Museum für Verkehr und Technik" in Berlin.

After building this storage, I thought that what is possible for storage must be also possible for the arithmetic unit. I then started to build a arithmetic unit for the binary system with mechanical switches. A replica of this machine is also in Berlin. However, neither the original nor the replica works well. This is probably a point that most inventors reach. He takes a certain path, has 80% completion level of his idea, and finds out, there is a better way.

In the meantime I learned a lot from this model. I learned that a arithmetic unit must be constructed with electrical parts. When information must be communicated between two widely separated points one wire is sufficient. The same communication with a mechanical technology is very difficult. Therefore I started using electric relay technology. First I built a small computer, the model Z2. The purpose of this model was only to show that a computer can be built with relays. It fulfilled this purpose. I then built relatively fast the computer Z3, which was based entirely on relays.

The work on the computers Z1 and Z2 gave me new ideas. I developed a "combinatorics of conditions" and discovered later that this was the same as propositional calculus. Today we speak of boolean algebra. In my opinion, it was an important discovery that everything can be done with on-off values, which was not evident at that time. It was possible to store numbers in the binary system, but what about complicated instructions?

In order to do an operation I had to convert a decimal floating point number to the binary system. That meant a lot of conditional instructions. It was not clear whether they could be implemented with relays. The discovery that this was possible was a sort of revelation, and now I could design a computer on paper. Like a mathematician I could lay out the switches using logical formulas and then solder them together. The electro magnetic relay technology enabled this. The electronic circuits that were later developed are not as simple in the transcription from paper to the actually working circuits. At this point in time, I had the propositional calculus, which came out of mathematical logic, and I then studied relational calculus and predicate calculus.

This opened a totally new world of computing. Today this connection between

mathematical logic and the construction of computers is a given fact. We need to look back to the time around 1938. At that time there were "pure" mathematicians. They emphasized the word "pure". To the left of pure mathematicians were the applied mathematicians who were not favorably looked upon. To the right were logicians. They were so pure that, even on the university, classes on this subject were not taught. In contrast to the applied mathematicians, they were the "anti-applied mathematicians" and they worked on problems that had almost nothing to do with daily life. The problems were mostly concerned with proving of mathematical theorems. Further to the left of the applied mathematicians were the people who worked on calculating machines. They were bold enough to contact the right wing and to take logic from very pure mathematics and use it in computers.

From this union developed today's computer science. Not only did the applied mathematicians slowly become respectable, but in the new field of computer science we soon had pure, theoretic, and applied scientists. Applied computer scientists were not respected in the beginning. Today their ideas dominate the field.

That has had several consequences. Today it is in fashion to talk about responsibility of a scientist. People say that when a scientist or inventor has a new idea he must ask society: "Should I do that?". There were then offices that investigated new inventions. They were concerned with rockets and so on. They were not concerned with computers or calculating machines. That means, that if I had said at that time, I had opened with mathematical logic a totally new world of computers, with which I could, perhaps, some day play chess, no one would have taken me seriously.

In the period after the invention of such a machine there is no widespread interest in it, even though it may be revolutionary. But I took it very seriously. It turned out that everything which we call today "information" can be coded as yes-no values. Because the programs were also written in yes-no code, which was known at that point of time, it was also clear that a program could have branches and conditionals and modify itself.

This idea often scared me in the beginning, because until then with the computers Z1–Z4 one could understand what was going on. You could even follow the calculations. In the moment that I allowed the computed data to influence the program – for that only a small wire connecting the arithmetic unit and the stored program is required – I could no longer monitor the calculations. I had a lot of respect for that little wire, because I felt as soon as this wire is there, Mephisto stands behind me. I can no longer watch what the computer does. It is actually true, that the fully free "go to" command is invented by Mephisto. With it a programmer can do the most amazing things.

I then attempted to find a further boundary. I wanted to reach a point where one could not do this or that, however, I did not see this point. I thought about what could now be done. Playing chess should now be possible! I developed ideas for a program for playing chess around 1938 I said jokingly to some friends: "In fifty years the world chess champion will be defeated by a machine!" I was wrong. 1988 is past and our world chess champion Kasparov is still undefeated by computers. But he no longer says that he will **never** be defeated by a computer. He is aware of that possibility. When, as is typical in engineering, you give me 10 % tolerance, then my statement could still happen. In fact, computers play world class chess.

Notice that in spite of the enormous increase in speed of computers (a million to a billion) the world chess champion has not been defeated yet. That is somewhat a disgrace for computer constructors and the researchers in artificial intelligence. This happens despite the fact that man has a small brain that works slower than today's computers (it does not switch in nanoseconds)! That means the power of the human brain can not yet be reached by computers. That applies in particular to artificial intelligence. For example, programmers program chess with the so-called "brute force method". The computer figures out all the possibilities until about the fourth move. One can easily see that with this method it is impossible to create the best program. The programs lack what a good chess player has: intuition. I think we should be happy and proud, that we have in our brains a switch, which is more flexible than the best computer! That was the situation in which I found myself and I noticed that I had touched a hot iron.

Now there was the question what I could do with these possibilities to modify the program during execution. For example conditional jumps and calculation of addresses. It was soon clear that they were possible. At that time it was not yet known whether electronic calculating machines would really work. Sometime in 1937 a good friend, Helmut Schreyer, visited me in my workshop. I thought he would help me to saw tin pieces in order to build my mechanical calculating machine. He simply told me: "You should do that with a vacuum tube!" He loved to play jokes and student life at that time was relatively funny. We drank together and were in general good friends with the police. We were not arrested for every student prank and when we were arrested it was a big joke. I thought that he was making a joke, when he said, that I should work with vacuum tubes. We thought about it and came later to the conclusion, that this idea was not bad.

In the meantime boolean algebra was discovered somewhere else (I did not know anything about Turing's work, and what I used for my machine I developed myself). Everything was developed at about the same time. That means, that only the basic instructions of the propositional calculus must be done with switches. If a computer was to work with vacuum tubes it had to be possible to replace the electro magnetic relays with an electronic circuit. This one-to-one replacement by electronics was also tried in the USA. In the ENIAC machine, the traditional number wheels were simulated by vacuum tubes. Because of that, ENIAC had an enormous number of vacuum tubes. We calculated how many vacuum tubes we would need. Helmut Schreyer made first attempts to build such a machine in his doctoral thesis. We then estimated we would have required around two thousand tubes in order to make a useful electronic calculating machine. We gave a lecture over this machine at Schreyer's institute for flow research. The experts there who worked daily with tubes said: "You want to build a machine with two thousand tubes? It will never work!" They had, like everywhere else in the world, daily troubles with vacuum tubes.

The same was said about the plans for the ENIAC. The Americans were so bold as to build a machine with 18'000 tubes. We would have never even thought about doing so! After the war, when we heard, that the Americans had built an electronic machine with 18'000 tubes, we shook our heads and asked, what they did with so many tubes. But it was now at least clear that a new way had been found. The future of the field was to work with electronics.

We made a proposal to a government office to construct a calculating machine with two thousand tubes. That was in 1941. They asked us how long it would take. We answered, that we required at least two years, which was probably not long enough. They answered: "How long do you think it will last until we have won the war?" That killed our proposal. Furthermore, Schreyer and myself were not left free to do computer development during the war, but military work. I was an engineer in airplane manufacture and Schreyer worked on technical problems in communications.

The German engineers were of course very interested in our computing machine. We were very humble and said, that we could calculate a thousand times faster than traditional machines. We could only promise a little, because we were not in the situation to build the machine. In spite of that, the temptation of working with mathematical logic was so big, that a totally new world of computers came out of it. The machines, that I had built so far, were the Z1 to the Z4. The Z4 was the only machine that was saved from the war. It was later installed at ETH Zürich. This machine could do only linear calculations (as one would say today).

I was a building engineer and worked with support frames for buildings. I knew, that beside the numerical calculation it must be possible to input the structure of a system point by point. From this it must be possible, with the help of logical operations, relational calculus, or graph theory, to automatically generate the program. I then believed, that in this situation it was correct to separate the two jobs. On one hand there were calculating machines, built with relays, which were reasonably powerful. It was not clear, whether the tube would really work. On the other hand there existed already a logical calculating machine. At that time "logic" was reserved for mathematical logic. Therefore I named my machine a "plan calculating machine". Today it would be named a "program generating machine". This machine should develop the program for the numerical calculations from the plan of the building. A program must be developed only once and can then be repeatedly executed with different parameter values. That was a compromise idea and I intensively worked on it.

That was the situation around 1944. It was the discovery of a new world of computing with immense possibilities, where address calculation and conditional statements played a little role. You must also know, what the technical situation was. The storage problem was solved and the thought that one could store programs implied a huge amount of possibilities. I had at that time a talk with a patent lawyer and said: "I would like to apply for a patent for a calculating machine, that can store programs". (A computer patent was already applied for). He said: "That idea in patent law is not a discovery. The question is, whether it belongs to the architecture. Your idea is certainly very useful, but here the question is, "how"? You must tell me, **how** the machine is built and how it changes the programs. The fact, that programs can be stored, is not worth a patent."

I worked on this application during the war. Everything from it is lost. I once wrote an essay, that started with Babbage and stopped with what we call today "John von Neumann architecture". I believe, that it is not enough to give John von Neumann only credit for inventing the storage of programs. That was already known. The genius of von Neumann is, that he came to the right time and that he selected out of a lot of possibilities with critical view, what was really important.

This is very important. In all new inventions there are a lot of possibilities, and a critical view is required to see, what is right at that moment.

The following prerequisites are necessary for a successful calculator of the above type: The program memory must be fast enough in order to carry out a large number of commands. The thought to replace the relay machine in a one-to-one fashion with the tube machine (at that time we had only tubes) was basically right, but it was also false in that the new possibilities of the tube were not really used. The tube is able to perform much faster what is done in a relay machine with very complicated switching, i. e. the translation and control of the microprogram(by the way, the first micro programmed machine was the Z1, that did everything with mechanical microprograms).

The one-to-one replacement with tubes was very tedious, but the machine was now a thousand times faster. For example, when a program for floating point arithmetic was ten or twenty times larger, it did not matter. The machine was still fifty times faster than the traditional machine. This prerequisite has to be there before it pays off to build a new machine at higher cost. Besides that, the memory must have a corresponding higher capacity. What Babbage wanted and what I had in mind for my first machine, was storage for a thousand words. With that a lot of automatic calculations could be made. But, for a real use of the possibilities of program storage it did not suffice, because a lot more commands are required and therefore a higher storage capacity. Technology reached this point and exactly at this moment von Neumann came along. He immediately recognized the right way and his way has shown itself to be correct.

We did not have the possibility in 1945 to work on the hardware. We had constructed the Z4, which was a relay machine and was switched in a very complex manner. We have professor Speiser in the audience. He worked on the machine at that time I do not think that there existed a second machine which used all of the possibilities of switching theory as much as this machine. As I already said, it was a calculator that did straightforward calculations and was relatively slow. It could calculate twenty to thirty times faster than a human. For that time, around 1940-1945, that was not bad.

Perhaps I can add something here. I was a building engineer and wanted to make the work easier for building engineers. But this only happened much later. The first people, who had a real interest in my machine, were the people in airplane construction, especially the specialists for vibrations. At that time the static structure of an airplane was quite well known and the stresses could be calculated for some special cases. In spite of this, strange accidents happened. Occasionally a wing would suddenly break off, the pilots would crash, and no one could say what had happened. It was a tragic situation.

In order to do a theoretical investigation of what happened when a wing started to shutter, it required a long investigation. When an airplane reaches such an instable state, the situation develops rapidly. Thanks to the work by Küssner and others the problem was theoretically solved and the correct numerical counterpart to this solution could be formulated. The numerical solution involved essentially the calculation of a matrix determinant with complex elements. That was a complicated calculation, that required by hand a long time. Exactly at that time I developed my machine, the Z3. It was finished in 1941 and was used for such

calculations.

Later, before the building engineers, the opticians expressed their interest. Even before the existence of computers opticians have done complicated calculations. They had up to 60 persons working together in one room, where up to ten people worked on the same lens. The refraction of each ray had to be calculated. Around 1950 they recognized they could use a computer. During this time the German optical industry was still strong (the Japanese were not yet in the market). They were very interested in completing these calculations with the computer.

It may be typical for the entire development (today we take it for granted), that a computer, which follows the von Neumann principles, can perform conditional statements and a lot more. The opticians wanted to do the ray-tracing through a series of facets. Every lens has several facets. We could show our calculations and it worked. In a few minutes we could do with the computer what a human needed one or two hours to do. We told them we could do even more. We could develop the curve of the astigmatism and through that develop a better lens. They said, that they did not want the machine to make a better lens, *they* would like to do that work! This was the attitude at that time within the opticians (as well as other communities). Today we have complex programs which only a few people understand. Originally the engineers and scientists did not want to give away the basic calculations.

There was a second group that was very interested in computers. These were the geodesists. They also had a lot to calculate. First, they only wanted to calculate a set of formulas and do the difficult operations themselves. At that time, the geodesic drawings with points and polygons were of legal importance. We could calculate the street maps with street crossings and so on. After doing so the geodesists said they would like to have the map drawn as well. Computer graphics developed from this. They were the first people who came to us and wanted to have that. They were very demanding about the exactness of the solutions. At that time the construction of drawing machines was done with step motors, they were not exact enough. We first made a detour and developed a better mechanical drawing machine. We sold a lot of these drawing machines ("Graphomat"), which, due to their accuracy, were used for a long time, although faster machines were available.

We just talked about computer graphics. I can tell you another sad story in this field. It happened a lot later, after we had built electronic machines, somewhere at the end of the 1950's. The largest German carpet company approached us and asked, whether we could automate the process from an artist's drawing to the forms for the weaving machine. I told them, that it was possible; however, the best solution would be even to design the carpet with the computer. When the artists heard that, the discussions were ended. That was about thirty years ago. An entrepreneur is a slave of his enterprise! The above mentioned attempt had a lot of possibilities. Why didn't we follow them? You can't follow ideas through unless you have a contract to do so in a certain period of time. You simply can't have two mathematicians working on every problem. The carpet firm was followed by steel firms. They had other problems that had to be solved. Therefore computer graphics remained on the shelf. Today it is high fashion.

I had the bad luck of being too early. When you come too early the chance

for success is small. The trick of the matter is, to come upon an idea in exactly the right moment. Let us consider the example of a car's anti-locking-system. It was evident, even in the beginning, that the braking pressure would have to be very carefully calculated, but then the question was, how many transistors were necessary. It was recognized, that the entire car couldn't be filled with transistors. This useful idea had to be allowed to ripen. Only after a further development of electronics could the dreams of the first computer developers be realized.

There is always a close connection between the technological development and the logical possibilities. Computers developed from mechanics, electro mechanics, tubes, transistors, and integrated circuits to the modern VLSI and ULSI circuit boards. Today there are a lot more possibilities. Things, which belonged to the wildest dreams of the first pioneers, are today possible. This is the link between computer architecture and the development of technology.

We know, that the shrinking of chips is related to price reductions, which is not evident. The first small computers were built for rocket engineers, so that they could include them in the rockets. These computers were more expensive than larger ones. However, people worked on reducing the high cost of these computers, and today we have not only the shrinking of chips but an increase in the speed and capability of these chips as well as a lower price.

Today we can read in the newspaper about the megachip (the 4MByte chip, the 64MByte chip and so on...). Laymen question why we need so much memory. Is it really so important? When the computer was developed, scientists and engineers created the stimulus from which the computer was developed. After it functioned well, the actions of salesmen spread it's usefulness over many fields. Even today is the development of ever faster chips and larger memory driven by scientists (in this case physicists and chemists, who need this huge capacity). Research has reached a point (for example differential equations, discretisation), where a large memory is required.

Certainly the Schrödinger equation is an example of this. In principle it is not hard to understand, the only problem is, that it must be solved in multiple dimensions, in both phase and configuration space. This means many coupled equations. In this case, the computer engineers can easily raise the capacity of their machine a million times. The physicists will always find problems, which will fill this new machine. Also, as these computers become common, the business people will make use of the new possibilities.

We are lucky that today the ideas of a Common Europe have spread widely enough that there are no longer trenches all over Europe. Hopefully we have gotten over this need. Whether a real "common collaboration" will develop, is another problem. The will is there, but there is more than Europe on the earth. The whole world is covered with computer networks. It is typical of engineers and scientists, that they did not think about things like viruses and computer crimes. In their minds these things do not exist. Today we have totally new problems in protecting data. There is a lot, that has do be done, so that we do not become victims of our own ideas.

Like I said in the beginning, the free "go to" command is an invention of Mephisto. Today we see that. With a free "go to" I can break in someone else's data bank and so on. These possibilities exist, and it is good, that there are peo-

ple, that show us the weak points. The young computer scientists, that are in this room, carry the responsibility for computers with us. We pioneers set the process in gear and saw many dangers, but we were not able to avoid them. Today we can't go back to where we were. In the future you will develop the networks, that can't be seen through. We need people with sharp eyes that are able to unnest things. The responsibility for these problems cannot be given only to the developers of the first computers. It is the young generation, who can develop new networks and computer systems that are not in the power of Mephisto. However, we will never totally get rid of him!