

MH-1, A COMPUTER-OPERATED MECHANICAL HAND

by

HEINRICH ARNOLD ERNST

dipl. e. l. ing., Swiss Federal Institute of Technology
(1957)

S.M., Massachusetts Institute of Technology
(1959)

SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF SCIENCE

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

December 1961

Signature of Author

Department of Electrical Engineering, December 23, 1961

Certified by

Thesis Supervisor

Accepted by

Chairman, Departmental Committee on Graduate Students

MH-1, A COMPUTER-OPERATED MECHANICAL HAND

by

HEINRICH ARNOLD ERNST

Submitted to the Department of Electrical Engineering on December 23, 1961, in partial fulfillment of the requirements for the degree of Doctor of Science.

ABSTRACT

Digital computers are used in a human environment in almost all of their applications. A man translates the physical problem into numerical form, a programmer provides the machine with all the information needed, and a human user interprets and applies the results to the real world. In this study, instead of using the digital computer as a tool during the execution of human thought processes that are ultimately concerned with the real world we wish to let the computer deal directly with the real world by itself, beginning with the perception of the real world and the appreciation of it and ending with the performance of a purposeful, active task in the real world.

As a vehicle for this experimental study a mechanical servomanipulator has been adapted for operation by the TX-0 computer. The sense organs of this mechanical hand feed information into the computer; the program processes this information, and the computer controls the motors that move the hand. The difference between this system and a conventional control system (for instance, a numerically controlled machine tool) is that the MH-1 system performs more than just the classical control functions, such as position control or speed control, in accordance with a prerecorded program that has been written by a human programmer after a careful analysis of the task and the environment. MH-1 itself performs automatically some of this analysis of the real world with respect to the broad description of the task to be performed. Thus it can select appropriate routines by itself and find out what to do in unexpected situations for which the programmer has not provided an explicit instruction.

To give an example of the performance of the latest programming system, one program consisting of nine statements will make the hand do the following: Search the table for a box, remember its position, search the table for blocks, take them and put them into the box. The position of the objects is irrelevant as long as they are on the table. If as a test for the built-in mechanical intelligence, the box should be taken away and placed somewhere else while the hand searches for blocks, MH-1 will remember the new position of the box and continue to work with it as soon as it has realized the change in the situation, that is, has bumped into the box while looking for blocks. This will be done

automatically, without any need to mention it in the specific program for this block-and-box performance.

The most important aspects of this coupling of the machine and the real world which have emerged during the work concern, first, the interrelation between the sense organs and the programming language, and second, the concepts of awareness and understanding.

Whereas in classical control systems the task of the sense elements is the transformation of some information originating in the real world into analog electrical signals, the tactile sense, which is of prime importance in the operation of MH-1, is of a descriptive rather than a quantitative nature and translates the information contained in the real world into patterns that are used in the language in which programs are stored in the machine. This requires a new approach to the construction of sense elements.

Awareness is defined as the possession of an internal representation of the actual state of the outside world plus the ability to compare this representation with an internal representation of the desired state of the outside world. It is postulated and demonstrated experimentally that a computer displaying artificial intelligence can be coupled successfully to the real world only insofar as it is aware of it.

Understanding is defined as the capability of relating a model to the real world. The experiments, of which a motion picture has been made, demonstrate an elementary kind of understanding on the part of the MH-1 system, achieved through writing the programs in a language that the computer itself can easily correlate with the real world. The computer can understand a program to the extent of looking it over and determining whether the program might be useful in any given situation, but the computer cannot generate appropriate new programs at the present time.

Thesis Supervisor: Claude E. Shannon
Title: Donner Professor of Science

Acknowledgements

I am primarily indebted to Professor Claude E. Shannon for his advice and his interest in my work, as well as for the freedom he allowed me during this somewhat unusual kind of research. Professor Marvin Minsky is his cosponsor of the original idea.

The Research Laboratory of Electronics has provided the means and the congenial environment for this work and I am grateful to all its members who have contributed in some way or other.

Many thanks go to John McKenzie of the TX-0 computer staff for all of his helpful explanations, his assistance even on Sunday mornings, and for not losing his patience when I claimed that the computer was defective when it was really my equipment that was. I could not have written the programs without the aid of the versatile utility routines provided by students of Professor Jack B. Dennis. Particular thanks for advice are due Robert Wagner and Robert Saunders.

John E. Keefe and several members of the machine shop of the Research Laboratory of Electronics were very helpful when parts for the hand had to be machined in a hurry, or when I tried my own skill at their machines.

My thanks go to the known and unknown members of the M.I.T. family who showed their continued interest by inquiring about the progress of "the hand" and who understood patiently when a promised demonstration failed because of trouble with the equipment.

I had many useful discussions with Ivan Sutherland who also corrected the second draft of this thesis. Dr. Helen Thomas of the Publications Office of the Research Laboratory of Electronics translated

by Swiss English into American English, and the scrambled draft was successfully untangled by Margot Fuller and put into neatly typed form.

Finally, the TX-0 and I were on very friendly terms most of the time and I would like to thank it for its contributions and what it has taught me.

TABLE OF CONTENTS

Abstract	2
Acknowledgements	4
Table of Contents	6
List of Figures	7
Chapter 1. The History of Mechanical Hands and Arms	8
Chapter 2. Preliminary Work on Artificial Hands at M.I.T.	12
Chapter 3. Design of Experiments with a Simplified Manipulative System	15
Chapter 4. Other Interesting Aspects of the Project	
4.1 Sensory Instrumentation	18
4.2 The Point of View of the Computer Engineer	18
4.3 Artificial Intelligence	20
4.4 The Human Aspect	20
4.5 Survey of the Areas of Interest	21
Chapter 5. More Methodological Comments	22
Chapter 6. Experimental Equipment	
6.1 The TX-0 Computer	25
6.2 The Manipulator	27
6.3 The Control Unit	29
Chapter 7. The MHI Phase	
7.1 Introduction	33
7.2 Principles and Structure of MHI	34
7.3 Description of Some Programs written in MHI	36
7.4 Programming in the MHI Language	38
7.5 Instrumentation	41
7.6 Inside-out and Outside-in Control	42
Chapter 8. The THI Phase	
8.1 Description of the THI Language	46
Chapter 9. Discussion and Evaluation of the THI Programming Approach	
9.1 A New Source of Errors	57
9.2 Artificial Animals	58

9.3	Programming Stupid Computers	59
9.4	Automatic Control and Automatic Operation	61
9.5	Artificial Intelligence	63
9.6	The Real World as Seen by MH-1	64
9.7	Recommendations for Further Work	73
9.8	Applications	
	i) Mechanical Hands for Industrial Applications	77
	ii) Mechanical Hands in Hostile Environments	80
	iii) Application of Programming Principles	82
9.9	The Study of Cognitive Processes	83
9.10	Summary of the THI Phase	84
Chapter 10. Conclusions		86
Appendix	1) The Experimental Equipment	90
	2) Scenario of the Motion Picture	96
References		99
Biographical Note		101

LIST OF FIGURES AND TABLES

Figure 1.	The modified servomanipulator	28
Figure 2.	Motor and sensor organs	30
Figure 3.	The hand	31
Figure 4.	Sense organs on the hand	31
Figure 5.	The control unit	32
Figure 6.	A typical chassis	32
Figure 7.	Blockdiagram of general real-world coupled automaton	62
Figure 8.	Blockdiagram of the control unit	94
Table I.	Comparison of formal methods	23
Table II.	Comparison between IBM 709 and TX-0 Computers	26
Table III.	Comparison between MHI and THI	56

MH-1, A COMPUTER OPERATED MECHANICAL HAND

Chapter I.

THE HISTORY OF MECHANICAL HANDS AND ARMS

Mechanical hands and arms have a long and fascinating history which reflects most of the preparatory stages on the long road towards our modern machines.

The first mechanical arms that we know of were built by the ancient Egyptians, as parts of statues of Gods, and they were operated by priests who were assumed to be acting under divine inspiration. It is hard to say now exactly what all that meant to the Egyptians, but it does point out the existence of some kind of primordial interest in machines resembling human beings and imitating human actions. I can testify to the fact that this interest still exists, at least in most of the people who have met my mechanical hand.

Many centuries later, in the Hellenistic period, we find several schools of creators of hydraulically operated statues. They were originally (especially under Heron of Alexandria) built to illustrate the science of hydraulics, and their description was patterned after the language used in geometrical proofs, involving axioms and theorems. It was not the intention of the designer to imitate either the internal nor the external aspects of human life. Revolving doors were as interesting as walking statues. They were abstract intellectual design exercises, the models serving as teaching aids.

Since most of these machines were stored in temples, eventually they did not fail to fascinate and entertain the worshipers, and during

the decline of classical Greek culture some might even have been built for that sole purpose.

The most intricate mechanical puppets were built in the eighteenth century in France, Switzerland, and Germany. Some of them had mechanically controlled forearms permitting them to write and draw sketches, some even had moving fingers with which they played small musical instruments such as organs or flutes. All of these automata were built with the intention of presenting a convincingly human performance. The actuating mechanisms were hidden from the spectators and were of no interest to anybody except the designer. These mechanisms consisted of intricate arrangements of cams and levers, or even stacks of cams chosen by a selector, much like a computer disc-memory.

The designers succeeded marvelously in their objective. People travelled literally for days just to see these charming mechanical wonders. The presentation of one of these machines to a monarch or prince would practically guarantee a lucrative existence to the builder for the rest of his life.

The essential properties of these clockwork automata were:

1) They were built to create the impression of being alive. A good job was done of it. People enjoyed the paradoxon of a mechanical device that behaved like a natural living being. They were, however, not interested in the theory of their internal operation, nor in the significance of these creatures.

2) These automata were completely insensitive to their environment. All of the control action goes from the inside, from the hidden control gear, toward the outside, to the limbs or the fingers. There is no information flowing in the opposite direction from the outside world into the control mechanism. There is, to use the terminology of our time, no capacity of discernment, although Droz's puppet could write: "Je ne pense pas, ne serais-je donc pas?", in imitation of Descartes' "Cogito, ergo sum."¹

The next step in the development of automata was the discovery of the possibility of using external information during the control process. This gave rise to what are now called feedback control systems.

The purpose of a control system is to make it possible for the machine to maintain a specified output despite various internal and external disturbances. Whereas originally the specified outputs were single, fixed quantities like the course of a ship, variable, preprogrammed inputs appear later, which are analogous to the cam mechanisms of the clockwork automata. These inputs had the form of either mechanical, optical or magnetic recordings. The function of the servos was to constantly bring the output as close as possible to the specified quantity. This permitted the construction of automatic machine tools that could turn out pieces of a prescribed shape accurately, reliably, and almost without human interference.² All of the human effort went into the construction of the program. In early machines this could be a scale model which a feeler scanned, thereby generating the right motions that were amplified and fed to the machine tool.³ Later, a human operator might machine the first piece himself. His motions are recorded magnetically, and then the machine can repeat the same motions over and over again. And, still later, the stored input was prepared by a digital computer, according to statements in a problem-oriented programming language.²

On several occasions it was felt desirable to replace the metal cutting tool with general-purpose tongs. The machine could then hold and use a hand drill, a spray gun, a brush or any tool intended for human use. Such a system starts to resemble, at least functionally, the human hand and arm. Examples of such machines are: Noman, built by H. W. Nidenburg⁴

and the tape-controlled manipulator by Borg Warner.⁵

Another approach to the problem of artificial hands and arms originated from the need for handling radioactive materials under safe conditions. This problem gave rise to the family of mechanical, and later even motorized, servomanipulators. In both of these cases there is no prerecorded program, but a following of the motions of a human operator, who can see and to some extent feel what the slave arm is doing. Examples of this approach are the machines built by American Machine and Foundry Company⁶, General Electric Company⁷, Hughes Aircraft Company⁸ and General Mills⁹.

A last group involved in the design of artificial arms and hands are the builders of prosthetic devices. Since they are meant to be operated by whatever facilities the human user still commands, they cannot be classified as automata and we shall not consider them here.

Chapter II

PRELIMINARY WORK ON ARTIFICIAL HANDS AT M.I.T.

The idea of building a mechanical hand to be operated by a digital computer was originally presented by Claude E. Shannon and Marvin Minsky during a seminar at M.I.T. in the fall of 1958. In all of the previous automata the decisions that lead to the specification of what to do at each moment were made by human operators ahead of time when they wrote the program, and only the execution of these specifications was left to the machines. The intention now was to let a machine make some of these decisions after the over-all goal has been communicated to it. In other words, the automaton should become autonomous not only with respect to the execution of predetermined actions but also with respect to its own sizing up of the real world and with respect to its decisions about how to proceed in order to reach certain specified goals. This seemed to be a perfectly logical step to take next, and an interesting and challenging one at that. But it was about all we knew and were able to say about the intentions and the expected results of this research at the time of the seminar.

Unlike the final goal, some of the difficulties were immediately obvious:

1) The problem of instrumentation or, as we imagined at the time, the construction of an artificial hand that would resemble the human hand functionally as closely as possible, and the construction of motor organs and sense organs for it.

2) The problem of choosing a control principle and the question of stability, in view of the fact that the human hand and arm have

approximately 35 degrees of freedom. According to what principles should the mass of sensor data be processed in order to generate and control sensible sequences of motions leading to some specified goal?

In an attempt to obviate these difficulties two exploratory studies were started:

1) D. Reich¹⁰ built a mechanical hand, the five fingers of which could be curled and flexed by pulling ten strings coming out of the artificial hand at the wrist. No motor or sense organs were included. The conclusion reached after this study was that it is not too difficult to build a reasonably close kinematic equivalent of our hand. This has been confirmed by several Russian studies on artificial hands for prosthetic purposes, especially Tomovic's work¹¹⁻¹³. However, it turned out that it is much too difficult to operate all of the ten strings manually in order to achieve anything like a sensible, coordinated motion. This has again been confirmed by Tomovic's work. He couples all of the actuating strings together with springs. Thus the number of degrees of freedom in Tomovic's hand is reduced to two: pinching and grasping, pinching being a motion for bringing the finger tips together as used, for example, in lifting a raisin, grasping being a motion involving the whole surface of the hand, as used, for example, in lifting an apple.

2) D. Bobrow undertook a study of a certain way of controlling a mechanical hand by means of a digital computer. He described the motions of the hand in a coordinate system fixed to the hand. His use of an abundance of transformations and similar mathematical operations on the coordinates describing the motions of the hand violates some basic con-

ditions which I feel should be imposed on the work described here. His study will therefore not be considered in any detail here.

Chapter III

DESIGN OF EXPERIMENTS WITH A SIMPLIFIED MANIPULATIVE SYSTEM

One year after the original seminar the work reported in this thesis was begun. Two things had become clear at that time: First, it was imperative to work with a simpler system than one as complex as the human hand. One reason for this was the difficulty of building the equipment, not only the kinematic model, but also the motor and sensor system. We felt that we were neither equipped nor qualified for that task, nor did we feel that this was really the most interesting part of the project. The other reason was that we felt that the difficulty of the control problem would increase approximately quadratically with the number of degrees of freedom, without becoming correspondingly more interesting. We now think that, because of the interdependence of the degrees of freedom in all practical motions, the difficulty increases considerably less than proportionally. Work with a simpler system could be expected to result in indications as to how to go about handling a more complex system if we should ever feel the need of having one.

The second requirement that had become evident was the need for experimentation. The needed experimentation was not for validating some theoretically obtained hypothesis, since we had none, but rather to gain some insight into how to go about tackling the whole unfamiliar problem. We have mentioned that we had only the vaguest ideas about the problems that faced us before we started to build specific equipment and before we started to experiment with it. Only experimentation could show which ways of realizing such a system were promising and which are useless. No

analytic tools were available for that decision. It will become evident during this report how one experimental step clarified and determined the objective of the next step.

We realized that there was a big difficulty in exploratory experimentation of this kind. The results would depend, of course, on the equipment, especially on the design idea behind the equipment, and on the kind of experiments that were run. In other words unsuitable equipment or unsuitable experiments would not take us anywhere. But how could we decide what was unsuitable ahead of time? For instance, we decided that we would have to do without a sense of vision, simply because we did not know how to implement one. But what would be the consequences of this decision? They only became evident after nearly one year of experimentation.

There was one principle, however, which seemed safe: The human hand and the human information-processing apparatus provide a proof of the solvability of our problem. Wherever possible we should therefore stay as close as possible to the methods and techniques that nature used to control our own hands, as far as those are known. For instance, in our information processes involved in manipulating objects, transformations of coordinates play no role. Therefore the control approach suggested by Bobrow was not followed up. However, in view of the desired experimental simplification, this principle is more applicable to the information processing aspects and to the design of the sense organs than to the mechanical construction of the hand itself.

These methodological questions turned out to be the most difficult to answer throughout the work. It was always much more

difficult to decide what experiments would be most meaningful and what relevant inferences could be drawn from them. To build the actual equipment, to program the computer, and to run the experiments was then comparatively easy.

In summary, the desired properties of the experimental equipment were, according to these considerations:

- 1) It should have a reasonable capacity for handling objects, but it need not closely resemble the human hand.
- 2) In order to provide as much experimental flexibility as possible, little special hardware should be built for control purposes. All of the functions that can be fulfilled by the computer should be performed by it, so that they can easily be modified. The motor-control feedback loop, for instance, will be closed through the computer, and not through external equipment. Undoubtedly at a later stage in the project it will become clear what to watch out for in this motor control system: Analog external equipment could be built. However, at the beginning it will pay off to be flexible, although this involves some waste of computer time.
- 3) There will be only a limited sense of absolute position in space. Humans do not have a precise conscious knowledge of the angles spanned by all their joints, and therefore this information need not be available to our system. On the other hand, a satisfactory "unconscious" sense of position is required to make possible the feedback loops that have been shown to exist in animals and humans for the control of the motor system.
- 4) The predominant sense input should originate from the tactile sense organs, since organs for vision are too difficult to build at the present time.

It was possible to devise the experimental equipment based on these four considerations, with reasonable assurance of its soundness.

Chapter IV

OTHER INTERESTING ASPECTS OF THE PROJECT

As soon as we had some definite ideas about the experimental equipment it was possible to define more clearly other areas of special interest in addition to the main problem of coupling a digital computer to the real world.

4.1 Sensory Instrumentation

Because of the decision to use a simplified system, not many mechanical or motor problems were expected to arise, and they were not the interesting questions anyway. However, the question of appropriate sense organs turned out to be important in connection with this work. The construction of transducers is a technically important and advanced field. Many sense organs for touch are possible, from strain gauges to simple switches. Experimentation was needed to guide us in the selection of suitable sense elements for our system, and to help us in our understanding of the consequences of the adopted design philosophy for the sensory system.

4.2 The Point of View of the Computer Engineer

Our original analysis was in terms of automata, of which the computer is an important part. But the problem has its interesting side for the computer engineer, too. Computers have been used almost exclusively in a human environment. All of the information fed into the machine is fed in by the programmer or operator, and all of the information going out goes to the operator or user. In this project we put a

computer into a real-world environment. Besides information originating from humans, sense information directly from the real world enters the machine, the computer changes the real world, and these changes in turn modify the sense input. This provides an immediate feedback that is not usually found when the computer is operated in an exclusively human environment (with the exception of re-enforcement in learning experiments). This immediate feedback is present in the application of digital computers in sampled data systems and is therefore nothing new in itself.

But there is another aspect to this real-world environment situation which has not been utilized in the conventional digital control systems. The physical world makes sense; our control processes are formulated with respect to this world. All obstacles and interferences take place in this world. It should therefore be possible to have the computer refer to this real world if difficulties arise. Now, the control computers refer to the human operator if something goes wrong during the execution of a program, or the program has to provide for all possible cases of trouble ahead of time. It should be possible to write programs that do not require perfection in foreseeing possible trouble, but let the system take care of troubles by itself, by referring to the real world for information on how to do this. In other words, now that something separate is available that contains information about the problem at hand, the computer program should be able to take advantage of this, instead of placing all the burden on the programmer's anticipation. This is really just another way of looking at the problem of coupling a digital computer to the

real world directly.

4.3 Artificial Intelligence

Closely related to this is an aspect of artificial intelligence which I always felt to be an important one. "Intelligence" and "understanding" are very often connected with the task of relating two systems, for instance, of relating a mathematical or verbal model with the real world. I think understanding "something" can be usefully defined as the capability of relating "it" to something else, most often reality. We are faced with exactly such a task in our problem: The computer has to relate its internal world (program, information contained in certain registers) with the real world in which the hand works. Little work has been directed at the exploration of such relations.

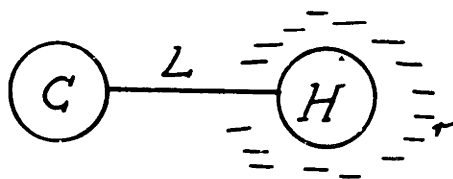
4.4 The Human Aspect

The problem has not only a technical side. Once we have an operating hand-computer system we can try to compare this system with our own information-processing system, and perhaps learn something about ourselves from this comparison. Unlike some computer simulations which were deliberately designed to imitate the properties of our information processes, the leading criterion in this project is feasibility. The subsequent comparison appears therefore in a different light.

The discussions throughout this thesis will take all of these viewpoints into account.

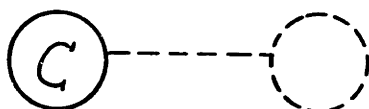
Survey of the Areas of Interest

The mentioned areas of interest follow naturally from the structure of the MH-1 system. This system consists of three parts: the computer C, the hand H working in the real world r, and the linkage hand-computer L.



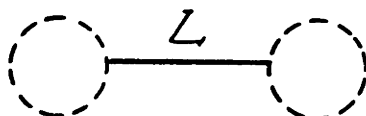
The areas of interest can be graphically represented like this:

1)



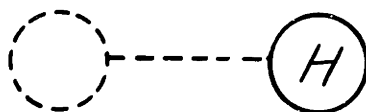
-What can be said about computers used in real-world systems and how does the accessibility of the real world influence the programming? (viewpoint of the computer engineer).

2)



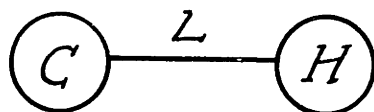
-How are the processes in the computer related to the processes of the hand in the real world? (artificial intelligence).

3)



-How does one build an artificial hand, and how does one build sense organs? (instrumentation).

4)



-What are the properties of the system considered as an automaton? (point of view taken in the introduction).

-What are the properties of the system considered as a model of living systems? (The human aspect).

Chapter V

MORE METHODOLOGICAL COMMENTS

Any casual reader of this thesis will be surprised to note the complete absence of mathematics. This should not suggest to him that this work is therefore less formal. In the "exact sciences" problems can be solved or hypotheses can be tested by formal methods. To that purpose they are stated in a language that can be subjected to the formal procedures that operate on the statements of this language and turn out an answer. For instance, if we want to find out how much an electron is deflected between two charged condenser plates, we set up a mathematical model of this physical situation. This model will permit the formulation of a differential equation for the trajectory of the electron. We then apply the formal procedures used to solve differential equations, and after substituting some numbers we get our answer. Mathematics is the formal procedure that is commonly used, but it is not the only one. Computers can be used for the same purpose, and that is what we are doing here.

Instead of constructing a mathematical model for the behavior of MH-1, we state it in a form that is adapted to the subsequent use of a computer: in terms of programs. Instead of applying the formal procedures of mathematics to see how MH-1 behaves, we use the formal procedures executed by the computer. Table I points out the correspondence between the two methods.

The correspondence of these two formal procedures comes to light in other respects, too. Both formal procedures are rigorous once the

Table I. Comparison of Formal Methods

Description of the situation	Mathematical method, as used for instance, in physics	Method involving MH-1 and Computer
Formal statement of situation	Physical model assumed to describe the behavior of certain bodies	Description of procedure supposed to generate certain desired behavior of MH-1
Exact solution of problem	Mathematical relation derived from model	Program derived from procedure
Evaluation of solution	Solution of the differential equation by the established methods of mathematics	Running the program according to the logic of the computer
	Compare result with the experimental answer, which we try to understand by the use of the model	Comparing actual behavior of MH-1 with desired behavior, which we try to understand by the use of the particular procedure

Physical problem is stated in the appropriate language. The translation of a physical problem into that language is a process that has nothing to do with rigor. There are usually many assumptions, simplifications, and idealizations going into that process. To what extent they are justified can only be determined by comparing the results obtained in a formal way with the experimental results, or, in our case, the observed behavior of MH-1 with the intended behavior.

Whether we can call a science an exact science depends on the availability of exact tools. The digital computers with their novel formal properties can therefore enlarge considerably the class of "exact sciences".

We started with the decision to build a mechanical hand to be controlled by a digital computer with the additional intention of not using the computer as just another device for the human operator to store a detailed program in for the control of the automaton. The computer should be used as a device to relieve the operator of some of the work of analyzing the task of the system and the environment and of programming the system in accordance with that analysis. The computer should deal with the real world directly, it should take over all of the important functions, starting with the perception of the real world and the appreciation of it and ending with the performance of a purposeful, active task in the real world.

We have specified the main characteristics of the necessary experimental apparatus and we have now dealt with the philosophy behind these experiments. We shall now describe the experimental equipment that was actually built.

Chapter VI

EXPERIMENTAL EQUIPMENT

6.1 The TX-0 Computer

We had a choice of computers to be used with the hand. One was the IBM 709, the other the TX-0. The selection was based on the relative advantages and disadvantages of the two as listed in Table II.

Some of these comments require further amplification.

Time requirements. Much more time is usually required for debugging real-time programs than for comparable conventional programs. Total time during which the hand was expected to be connected to the computer (for either standby or actual use) exceeded several hundred hours. The total turned out to be approximately 500 hours, of which 80% was for standby. This shows that the use of the IBM 709 would be feasible only if time-sharing or a sophisticated interrupt scheme is available, so that the standby time and excess time during the operation of the hand itself could be profitably used.

Interrupt feature. The absence of this requirement on the machine may be surprising. It stems from the fact that there is no single sense organ whose response alone is sufficiently meaningful to cause any computer interrupts. Considerable computer activity is required to interpret the sense outputs first. The only conceivable use would be in an interrupt scheme permitting the concurrent running of other programs, which would have to be interrupted periodically in order to determine whether the hand needs any attention.

Table II. Comparison between IBM 709 and TX-0

IBM 709	advantages	disadvantages	TX-0
<p>large memory</p> <p>long words (36 bits)</p>	<p>few in-out facilities</p>	<p>a variety of useful in-out equipment:</p> <ul style="list-style-type: none"> - dig-an. conversion - direct access to all bits of one arithmetic register - programmable control pulse output for external equipment - variable in-out logic by means of DEC equipment 	<p>small memory</p> <p>short words (18 bits)</p>
<p>rigid scheduling</p> <p>difficult to get enough time</p>	<p>easily available</p> <p>whole weekends available</p>		

Memory capacity is usually of prime importance in artificial intelligence programs. However, this situation can be vastly relieved through the construction of special-purpose programming languages. Although I could not know this for certain during the planning phase, it turned out later that I never even came close to exhausting the 8000 word memory of the TX-0. The decision was made to use the TX-0 computer, and I am convinced that the project would have suffocated from lack of time if we had chosen to use the IBM 709 computer of the Computation Center, M.I.T.

Here are the most interesting facts about the TX-0 for the reader who is not familiar with this machine (see the TX-0 Manual¹⁴).

(a) It's a microprogrammable, single address, transistorized machine with a cycle time of 6 usec and two cycles for most instructions.

(b) It has an order code convenient for information processing, but not for numerical work. There is, for instance, no direct multiply.

(c) An excellent set of utility routines is available for both absolute and relocatable programming, and flexowriter interrogation debugging.

More data can be found in Table II. In summary, the TX-0 is an extremely convenient machine for purposes like ours.

6.2 The Manipulator

As we have explained we intended to use a reasonable functional approximation to the human hand instead of developing a complicated piece of apparatus. Therefore we decided to buy an American Machine and Foundry Servo Manipulator (see Fig. 1)⁶ and proceeded to motorize

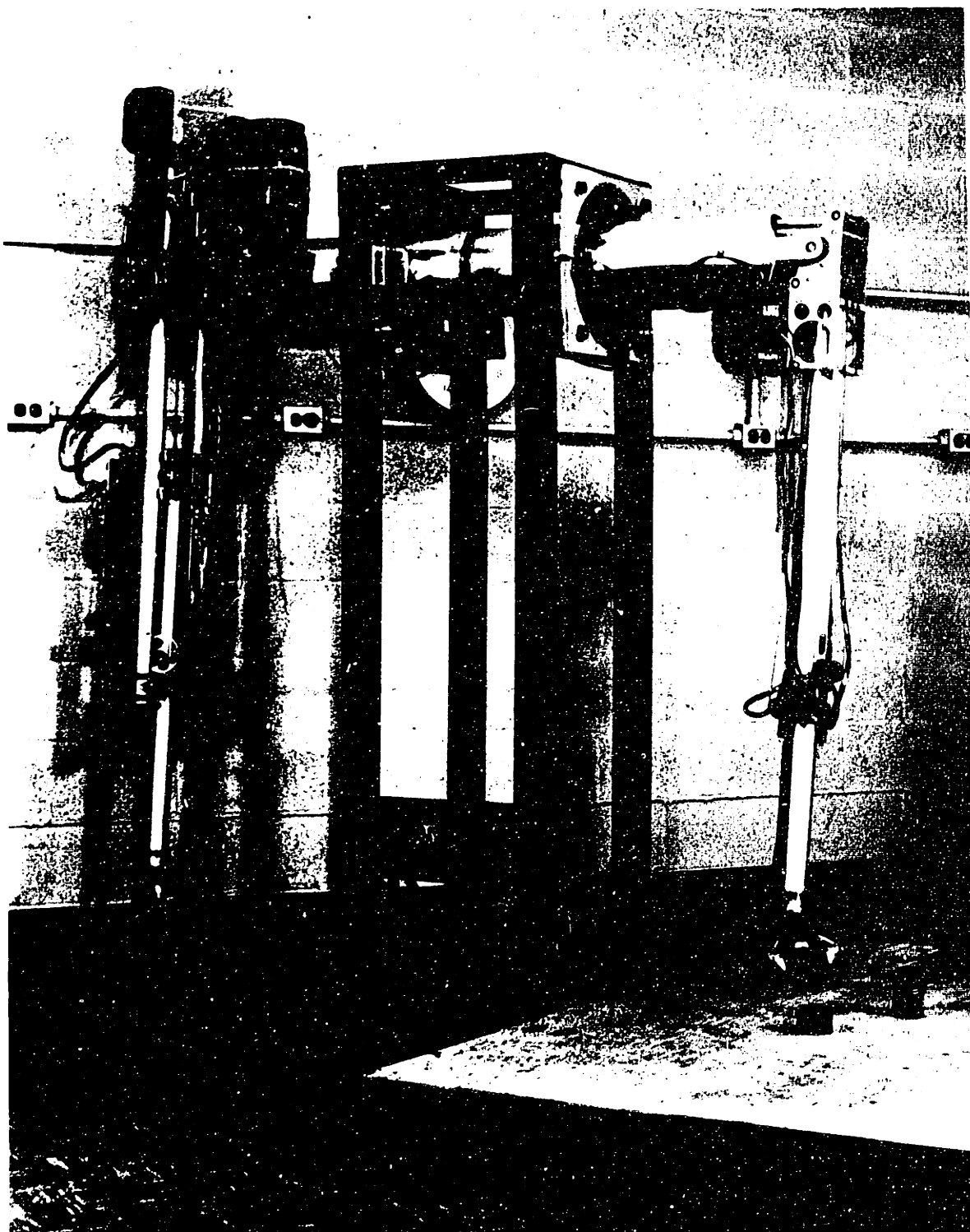


Fig. 1. The modified servomanipulator.

the mechanical unit. Each of the seven degrees of freedom requires one motor and one position sensing potentiometer, as shown in Fig. 2. Figure 3 shows the hand, and Fig. 4 explains the sense organs in more detail. The sense organs are the only piece of equipment that we shall have to talk about again. Further detail about the manipulator can be found in the Appendix. About four months work went into the modification of the servomanipulator and the design and construction of the sense elements.

6.3 The Control Unit

This part generates the motor control phase voltages in accordance with the computer specifications, connects selected sense elements to the analog-to-digital converter, and provides a sense of time for the TX-0 computer. It is transistorized except for some thyratrons for the servomotor control.

Besides computer operation the control unit permits manual operation for trouble-shooting purposes. The unit contains approximately 150 transistors, 300 diodes and 32 tubes.

Designing and building this control unit represents approximately four months work. Figure 5 shows the control unit, and Figure 6 shows a typical chassis. A block diagram of the control unit and more details are given in the Appendix.

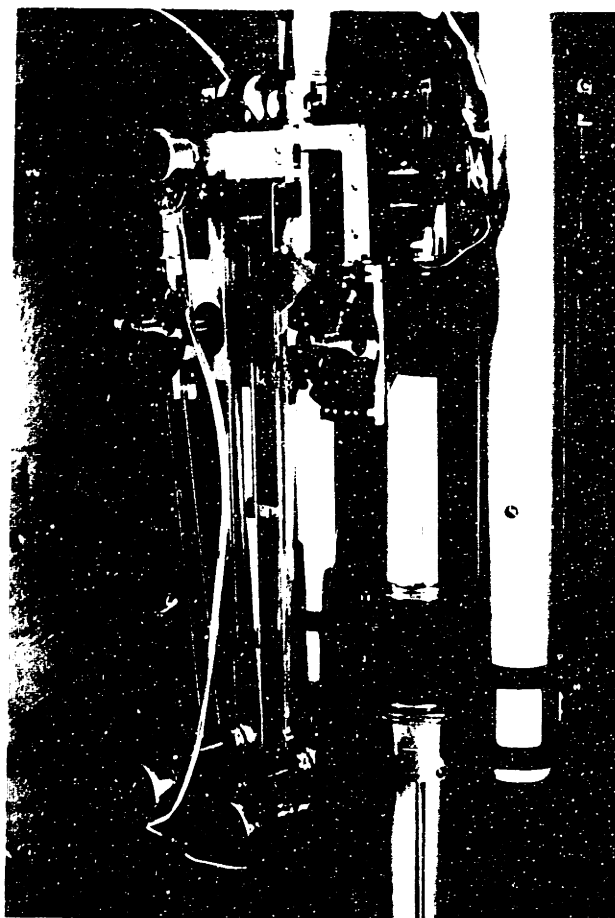


Fig. 2. Motor and sensor organs.

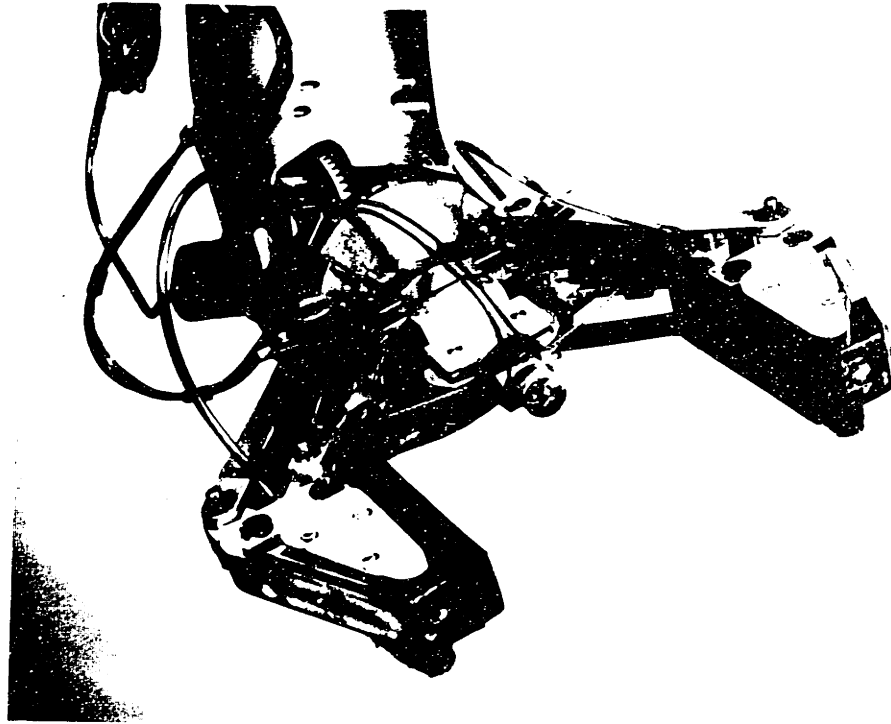


Fig. 3. The hand.

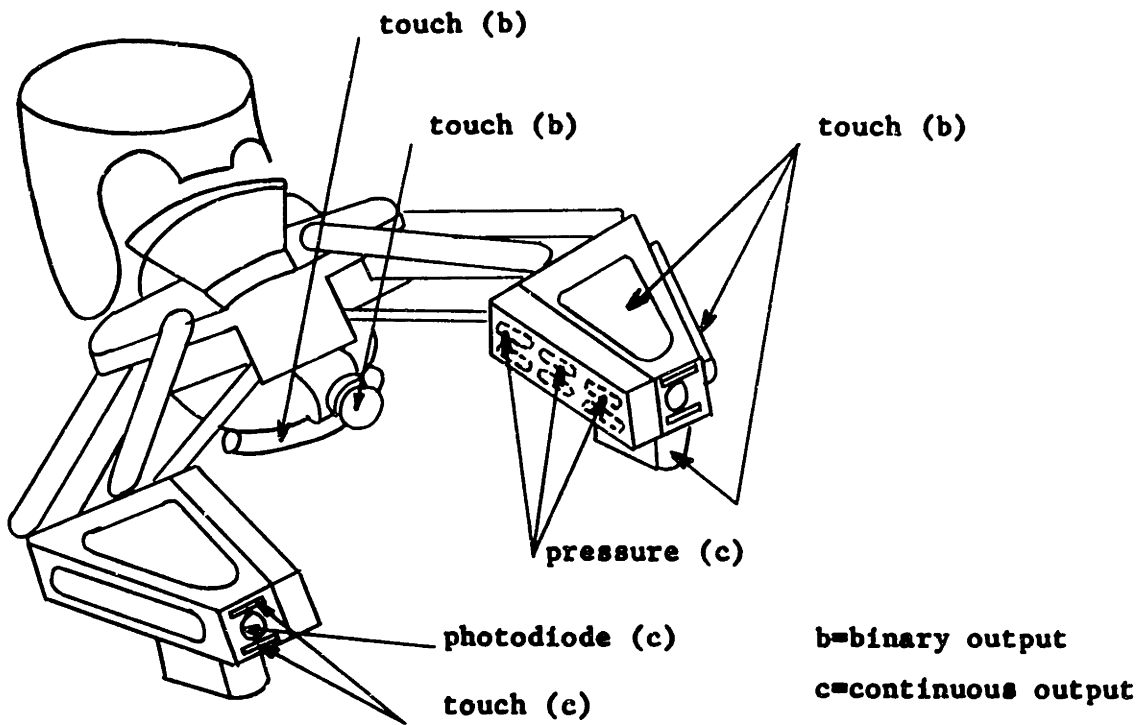


Fig. 4. The sense organs on the hand.

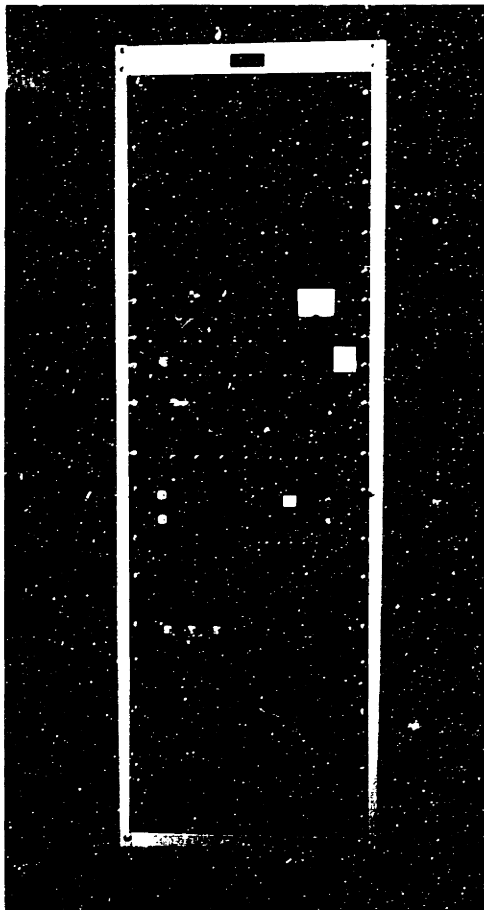


Fig.5. The control unit.

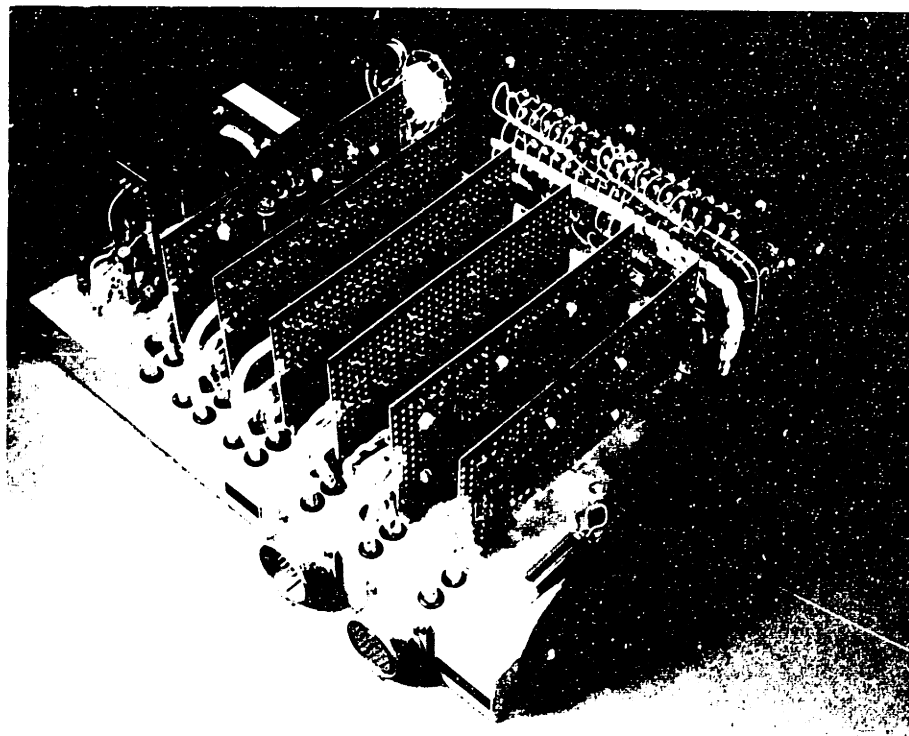


Fig. 6. A typical chassis.

Chapter VII
THE MHI PHASE

7.1 Introduction

We shall now discuss the main part of this work - the programming problems. It has been clear from the beginning that what is required first is a convenient programming language that permits adequate, although not necessarily sophisticated, specification of that which the hand is to do. At this stage we are not concerned with advanced decision making or similar problems. We simply want to get an elementary system going, as a start, in order to gain familiarity with the problems.

For the design of this language there are two things to bear in mind:

1) The speed of operation of MH-1 is determined by the mechanical equipment and not by the computational capacity of the equipment, thanks to the speed of the TX-0 computer. The programs can be expected to be long as compared with the memory of the TX-0 computer. The usual objections to interpretive programming are therefore not valid in this case, and interpretive techniques will be used throughout this work.

2) It is not always realized how much the programming language and its philosophy affect the work itself. First, because the language as such makes the programming easy for some things but difficult for others, so that the programs, and therefore the experiments, will be biased in the direction of the principles used

in the programming language. Second, we come to realize soon how greatly the programming language that is used and its properties pattern our own way of thinking and looking at things. This bias becomes clearly evident when we compare the MHI phase with the THI phase or when we try to draw conclusions from one experiment to find a new approach for the next experiment because by the time of the second experiment one has become thoroughly entrenched in the philosophy behind the programming language used. This first programming language should therefore be as noncommittal as possible.

The objectives of the MHI (=Mechanical Hand Interpreter) programming language were:

- 1) The development of certain routines that are independent of the programming language, such as motor control programs, sense-input sampling, timing, and synchronization routines, and so forth;
- 2) Gaining familiarity with the operation of the system in general by writing some simple programs for MH-1 in the MHI language; and
- 3) The expectation of recognizing in this first attempt the main difficulties of programming MH-1, and thus obtaining the basis for the later design of a better programming language.

7.2 Principles and Structure of MHI

MHI is based upon the assumption that all complex motions can be decomposed into a sequence of steps of the following kind:
"Move in direction z with speed v until sense element s indicates a;
if the sense element s indicates a, go to the next instruction,

otherwise continue the same action."

Accordingly, the source language MHI contains four expressions: move, until, ifgoto or ifcontinue, and several parameters indicating speeds, coordinates, sense organs or instructions. More than one degree of freedom can be varied at a time simply by adding more "move-until" statements and by expressing the "ifgoto" as a combination of conditions on single sense elements. In the actual source language most expressions are abbreviated, but in the following examples the complete "english" version is added to aid the understanding. We shall give an example of a periodically interrupted lateral motion that is continued, with something unspecified(b) happening at these times, until a maximum deflection is reached.

Program No. 1 (MHI)

Interrupted Lateral Motion

a,	move x for 120	Move in x direction forward with speed 120
	until s1 10 rel 101	until sense organ 1 indicates a decrease of 10, relative to the indication at the beginning of this step (condition f1)
	until s1 206 101 abs stop	or until sense organ 1 indicates 206 or less absolute, then stop (condition f2)
	ifgoto f1,b	if condition 1 alone is fulfilled, go to sequence b
	ifgoto t f2,c	if at least condition 2 is fulfilled, go to sequence c
	ifcontinue t, a	in all other cases continue sequence a

The MHI program interprets one such sequence each 1/10 second, according to the indication what to do next at the end of the last sequence interpreted. In the preceding example this is either sequence a, b or c. MHI performs all of the necessary operations for correctly performing the actions and decisions called for by such a sequence; for instance, it samples the required sense elements and compares them with the stated conditions, generates the control signal to the motor control unit in accordance with the difference between desired and actual speed, stops motions if this is desired, maintains the position of some degrees of freedom to which forces are applied, and so on. Also, there are certain checks and error printouts for incorrect statements and undesirable equipment conditions, timing and synchronizing routines to match the speeds of the computer and the external equipment, and programs for some additional MHI features that are merely programming conveniences and of no fundamental interest. The MHI interpreter occupies approximately 2000_{10} registers in the computer.

7.3 Description of Some Programs Written in MHI

Two practical programs, GC-2 and BBP-1, have been written in the MHI language for the operation of MH-1.

GC-2 will perform as follows: MH-1 searches for a box that is somewhere on its working table. It assumes that the first object encountered is this box. As soon as it is found, MH-1 explores and remembers its position, and then asks for wooden blocks to be placed on the table.

If the blocks are there and the program is restarted, MH-1 searches the table for these blocks, takes them and drops them into the box. It continues doing this until there are no blocks left on the table.

GC-2 is an example of the feasibility of a preprogrammed activity, with no operator interference necessary or possible short of manual interference with MH-1 or halting the computer.

The program is successful in approximately eight out of ten trials. The most common failure is that occasionally one of the blocks is knocked over and has to be replaced by the operator.

There is usually spontaneous applause by the audience when the first block is dropped into box, and some on-lookers have been heard to comment favorably on the "human" impression conveyed by the performance.

The program uses approximately 2100_{10} instructions, corresponding to approximately 600 source language expressions, plus MHI. Thus it uses approximately half of the TX-0 memory. The TX-0 performs actual control operations during one-third of the time and idles during the rest, waiting for the next period.

The second program BBP-1 can be used by a human operator to build small structures with wooden blocks. To this end, the operator can call for several simple or more complicated motions, such as (simple ones) move forward a little, release block, etc.; (complex ones) put new block on top of last block, get previous block, etc. Once a motion is called for on the optical display unit, it will be completely executed by the machine without further operator influence.

BBP-1 demonstrates the remarkable ease in communication that can be achieved with MHI. The amount of human information necessary for steering the system through these complex tasks is approximately 20 bits/minute. An especially successful demonstration program is the one in which a more or less leaning tower is built out of the blocks. In approximately half of the attempts it succeeds in putting all of the available eight blocks on top of each other before the tower collapses. The precision of the actions is comparable to the precision achieved by a two year old child.

The subroutines for BBP-1 are programmed in MHI, the in-out and control programs in machine (macro) language. The program uses approximately 2200_{10} registers, plus MHI.

We shall now discuss some of the experiences gained during the MHI period.

7.4 Programming in the MHI Language

It is easy to see that MHI is a straightforward application of conventional computer programming techniques and programming language principles. Statements linked by conditional transfers determine the actions to be executed. The reason for this straightforward approach is plain: It is best, at first, to try to apply well-known, conventional techniques to a new problem. As we expected, it turns out that these techniques have several serious drawbacks:

The resulting language is easy to use, but it is uninspiring, too. It is boring to decompose a whole complex activity such as GC-2 into a sequence of the elementary steps described, including the determination of all of the parameters such as distance and speed.

Although in BBP-1 sequences of considerable length can be called as a whole, these sequences must still be written originally. There should be a way to specify a whole action, for instance "put the hand flat on the table", "grasp the object you are touching," so that the decomposition into elementary steps, including (this is most important!) the assignment of parameters such as speed or sense input can be performed automatically by the computer.

One should, however, not misunderstand the explained shortcoming. The flood of recently developed programming languages for numerical and business data processing tends to convince people that all programming troubles arise from the fact that the language in which the problem is communicated to the computer is not sufficiently concise. This may very well be the case for the problems for which these languages are designed, but it is not the cause of trouble in our problem. A statement in a concise language contains all of the information that the equivalent statement in a less concise language contains in a less efficiently coded form. We, on the contrary, want to reduce the amount of information to be communicated to the automaton because we feel that the computer should be able to get this information by itself from the outside world. Our primary problem is not the improvement of the relation man-machine; it is the improvement of the relation machine-real world. A welcome byproduct is the fact that this will result in a better relation man-machine, not because of easier communication between the two but because the computer can take care of things by itself for which it used to need human assistance.

A second shortcoming is that MHI programs show something which is best described as rigidity during their action. Situations that the programmer has not foreseen and explicitly provided for in his program cannot be successfully dealt with. MHI simply ignores these novel situations and goes on doing whatever it was doing, although this may be disastrous. If the tower of wooden blocks collapses it goes right on putting wooden blocks "into place," but now the place is empty space! If the box into which the blocks are dumped is taken away it goes right on dumping the blocks where the box used to be.

The reason for this unsatisfactory kind of behavior is easy to understand in the light of comments already made.

The program, especially during the outside-in control functions, uses information provided by the outside world. But it uses only the information called for by the programmer. MHI really couples MH-1 to the programmer, and then to the outside world indirectly, through himself. What information the programmer has not explicitly cared to gather from the outside world will not find its way into the program. Instead of using the information available in the outside world in emergencies, the program relies upon the original plans and intentions of the programmer. We have pointed out that the programmer could probably be discharged of some of his responsibilities because the physical world could be called upon to help. We have accomplished nothing in this direction thus far.

The MHI-controlled mechanical hand is more flexible than, say, a magnetic tape-controlled mechanical hand. But only because the logical organization of the digital computer permits the programmer

to build more flexibility into his program. He may include many alternatives that are chosen according to the environmental conditions. It is, however, still the programmer who decides what to do in each instance. MH-1 together with MHI is still a normal automaton, placing all of its dependence on the human programmer as far as the recognition of the real world is concerned. But, then, we did not expect anything beyond this from MHI.

Now that the elementary part of MH-1 is working and that we have verified experimentally the earlier discussion of automata and their limitations, we have to start working on the originally proposed more advanced schemes. We now have to program some understanding for the outside world into MH-1, not only the capability of executing the instructions given by the programmer after he has understood what should be done. We hope that understanding of the real world will enable us to specify the desired activity in broader terms, leaving their interpretation in the framework of the real world to the computer. This should relieve the first shortcoming of MHI, the boredom of detailed programming. It should also enable the machine to recognize the unexpected situations that it now ignores and take corrective action instead of simply overriding unexpected obstacles.

7.5 Instrumentation

This can fortunately be limited to a short statement: Our equipment was very adequate for this phase of the work.

7.6 Inside-out and Outside-in Control

We have said that the sense organs for position and for touch have their specific roles: There is only a rough explicit sense of absolute position in space, the exact position of the hand has to be ascertained by touch. Exact positioning is only possible with respect to material points of reference. During the MHI phase, I began to realize how fundamentally different those two modes of using sensory information are.

The sense of position in space is a quantitative kind of information. Among the lower animals, as well as in our artificial system, these signals are directly used in conventional closed-loop control systems, working almost entirely on the "unconscious" level. For two reasons, I came to call this kind of control action based on quantitative information inside-out control:

- 1) The position of the extremities (fingers, etc.) is determined by positioning consecutively the innermost, then the next peripheral joint and so on, until the desired total position is obtained, in a direction from the inside towards the outside.

- 2) The information for this control action is contained completely inside the system. We "know" where to move, or it is contained in the program.

The sense of touch (and if there is one, the sense of vision) is involved in that which I call outside-in control.

- 1) The position as a whole is determined by conditions concerning the extremities, the inner joints are then consecutively moved in such a way that this position is achieved, from the outside toward

one inside.

2) Furthermore, the necessary information is contained in the outside world, in the form of an object to touch or to see.

The following example of the same action controlled by one and then by another principle should help to make this clearer. We shall again use our standard action of putting blocks into a box; in particular, let us assume that a block should be searched for and found on the table.

For an inside-out scheme it is absolutely essential that the machine be told exactly where the block is. The arm will then move to the coordinates representing that location, to coordinates that are stored in the machine. The position of the hand and of the fingers is apparently irrelevant, that is, it follows automatically from the executed motions of the shoulder and wrist. It is incidental that if the arm reaches the specified position, a block will be between its fingers. The machine does not care about this block in the outside world, it simply goes where it is told to go.

Outside-in control, on the other hand, would proceed in this fashion: The position to be reached is indicated by the block itself; the hand will realize it has reached this position when it touches the block. The point of reference is contained in the outside world, and not stored in some form within the machine. The sense organs used are those for touch and not those for absolute position; these are the sense organs in the extremities of the hand, in the fingers, and not the ones in the central joints. Shoulder and wrist-joint

positions are determined by the finger's search for the block; the position of each in itself is irrelevant, they will follow automatically once the fingers touch the block.

Remembering that our original objective is to couple the MH-1 system effectively to the outside world, we can now see how the distinction between these two modes of control fits into our plans. Positioning by outside-in control depends on clues provided by the external world, whereas positioning by inside-out methods depends on machine-internal clues. Introduction of outside-in control into the operation of MH-1 therefore couples the automaton closer to the real world.

Beginning with the next phase, the difference between the two modes of sense information, inside-out and outside-in, will be taken into account on a basic level, and it will be seen that much profit is to be derived by doing so. This clarification of the different functions of the sense organs alone was well worth the effort spent on MHI. It gives an essential clue about how to continue.

For those involved in any analog-versus-digital controversy it is important to realize that our present-day technical servosystems are of the inside-out control type, the type that corresponds to the autonomous animal control system. By means of digital-to-analog conversion techniques digital computers have been used in such systems, usually called sampled data systems. Every effort is made to make the digital computer look like an analog component. The binary, essentially logical, pattern-oriented type of sense information that is typical for the outside-in control principle, is much closer to

the untampered-with nature of digital computers and truly digital techniques, besides achieving another step in the direction of closer coupling between automaton and real world. This property is the connecting point for the application of such really digital techniques as pattern recognition, heuristics, or learning of a more advanced type than simple adjustment of a parameter, in automatic control. It is this truly digital application of digital computers in connection with control and the study of cognitive processes which interests us here.

The main results of the MHI phase are therefore: MH-1 can be operated as a classical automaton, and it will then display the discussed shortcomings of such automata: Its programming is a time-consuming job for the human programmer, and the automaton is not sufficiently in contact with the real world.

MH-1 operated in this way is a very flexible automaton because the computer permits the use of very flexible programs. We can get a good feeling for the kinematical limitations of MH-1 by watching the MHI programs.

There exist two essentially different modes of using sensory information, one used for outside-in control, the other for inside-out control. MH-1, contrary to conventional servomechanisms, stresses the use of the outside-in principle. Outside-in information has a distinctly digital character.

We shall discuss, next, the use of these results during the development of the next programming language, the THI language.

Chapter VIII
THE THI PHASE

8.1 Description of the Language

The first of the changes in MHI, which will eventually lead to the THI programming language, was the desire to express the necessary motion in terms of sense elements rather than in terms of the motors to be used. Thus the emphasis of the instructions is on the "what" to do, in MHI it was on the "how" to do it. In other words, the goal itself is stated instead of the mechanics on how to reach it. MHI represented the Mechanical Hand Interpreter, THI is the Teleological Hand Interpreter. In automatically determining the "how" according to the "what" we have the possibility of introducing the mechanical intelligence and understanding that we have already called for.

Application of this first THI requirement does not result in much change with respect to the lower or inside-out control operation because in this case sense organs and motors are in a strict one-to-one relationship. However, the assignment of motors to the different sense organs of touch changes with the position of the hand. Since these higher order control operations are the ones on which we place most emphasis, the change is significant. In the new language all the parts of a program which are specified in terms of the sense of touch alone are independent of the position of the hand. For instance, a program for grasping an object will work from the top, or the sides, or the back of the object. In order to achieve this, the tactile sense inputs, which are all binary or made binary by comparing them

to a level, are combined into a pattern of one's and zero's in several registers. Then masks that are transformed according to the position of the hand are applied to the sense input registers and to the program statements in order to determine which motors have to be used to achieve a desired sense configuration. For many tasks this results in a much shorter program in THI than in MHI. We shall compare the two programs used to place the hand flat on the working table. Besides being shorter, the THI program is about twice as fast. Approximately 5 per cent of the computer time is now used to execute the sensor interpretation process (see Program No. 2 and Program No. 3).

A few things have to be learned to understand the THI language. There are symbolic statements, which are the ones concerning the outside-in control, that is, the sense organs for touch; the explicit statements, which influence the inside-out or positioning operations, and finally, the transfer statements, which all start with i-. Explicit statements are followed by a series of parameters quite similar to the MHI statements, that indicate the desired output of a specified sense element. Speed indications are now placed behind a comma, and are preceded by a v.

Example: `explic s3 10 rel stp,v10` means: move with speed 10 so that sense element s3 indicates 10 more than at the beginning of this phase, then stop.

Symbolic statements are followed by five arguments, which indicate the expected changes in the sensor indications, and the speed with which these changes should be brought about. Example: `symbol 0,0,btm,0,v20` means: move with speed 20 in the right direction to bring the wrist bottom sense element into contact with something.

Program No. 2 (MHI)

<p>b2, move z don 240 until s8 200 lol cst until s15 200 lol cst until s27 200 lol cst ifgoto f2 f3 f4,b7 ifgoto f3,b4 ifgoto f4,b5 ifgoto f3 f4,b6 ifgoto f2,b3 ifgoto f2 f3,b4 ifgoto f2 f1,b5 ifcont t,b2</p>	<p> z-motion = up and down s8 = btm } s15 = rbt } see below s27 = lbt } f3 = third until condition b5 = new address</p>
<p>b3, move elv don 150 until s15 200 lol cst until s27 200 lol cst ifgoto f1,b4 ifgoto f2,b5 ifgoto f1 f2,b7 ifcont t,b3</p>	<p> elv = flexing wrist</p>
<p>b4, move twi lef 150 until s15 200 upl cst</p>	<p> twi = twisting wrist</p>
<p>c1, move z don 300 until s8 200 lol cst ifgoto t f2,b7 ifcont t,b4</p>	
<p>b5, move twi rit 150 until s27 200 upl cst repeat c1 ifgoto t f2,b7 ifcont t,b5</p>	
<p>b6, move elv up 150 until s27 200 upl cst repeat c1 ifgoto t f2,b7 ifcont t,b6</p>	
<p>b7, move z up 200 until s3 2 rel lol ifgoto f1,b8 ifcont t,b7</p>	
<p>b8, ----</p>	

Program No. 3 (THI)

<p>isymbol 0,0,btm,0,v20 isymbol 0,0,rbt+btm,lbt,v10</p>	<p> btm = sense organ at bottom of wrist rbt = sense organ at bottom of right finger lbt = sense organ at bottom of left finger</p>
<p>symbol rbt+btm,lbt,0,0,v10 iexplic s3 370 rel stp,v10</p>	

There are two kinds of transfer statements: ifgoto al (ifg) and ifcontinue (ifc) al. The contractions iexplicit and isymbol are equivalent to explicit ..., ifc or symbol ..., ifc. al is the indication of the location to which to transfer.

After the interpretive program has been directed to a new location, it concentrates on the string of consecutive statements before the next ifc. It interprets the sense organs with respect to these statements and activates the appropriate motors. Whenever all of the statements between any two transfer statements are true, control is transferred to the location indicated in the second transfer instruction. As long as no transfer has been executed, the interpreter continues to work on the same string of expressions.

Example: symbol 0,0,rbt+btm,lbt,vl0

iexplicit s3 370,rel stp,vl0

will transfer to the next instruction in sequence as soon as both the symbolic and the explicit condition are fulfilled.

symbol 0,0,rbt+btm,lbt,vl0

ifg a0

explicit s3 370,rel stp,vl0

ifc b0

will transfer to a0 if the symbolic condition is fulfilled, and to b0 if the explicit condition is fulfilled. If none is fulfilled, the interpreter will keep trying to fulfill them both.

The second change by which THI distinguishes itself from MHI is the introduction of an interrupt feature. All of the sense organs are now sampled periodically, not only certain specific ones. The sense organs are constantly divided into two groups: those elements whose changes are expected by the program, which will then influence the transfers, and those elements in which no changes are expected. An unexpected change indicates a situation that was not foreseen by the programmer. As we have mentioned, (and shall explain in more detail later) the programs and the sense organs are not developed enough to permit an automatic generation of more than a few elementary instructions to cope with an unexpected situation. In case of such an interruption, the existing program is searched for instructions in which the now interrupting sense element is expected to change deliberately. Since the THI language consists of expressions in terms of sense organs, this is easy to do. If such an instruction is found, control will be transferred to that part of the program. The search proceeds forward (but not beyond the routine that is used this time) and backward (through the last two or three routines used) in an oscillating motion of expanding amplitude. Searching further away than the limits stated in distant routines does not serve any purpose because it gets more and more likely that the goal there is a different one, and transfer to that location would be meaningless even if a matching instruction is found. The search is stopped when the first appropriate instruction has turned up. The case of no such instruction appearing will be discussed later.

Together with the transformation of the motor assignment for outside-in control in accordance with the position of the hand, this procedure is quite powerful, even more so than expected. It turned out that in the GC-2 program, in which blocks are thrown into a box, with the modification that the box, when found, is not just remembered but lifted and put down at some standard place, the computer has a good chance of being successful in the situation that arises when, instead of placing the blocks on the table after the box has been taken care of, the box is taken away and put down at a different place. In about one out of two cases the interrupt feature will interrupt the block-searching program when the box is at its new place, since during the process of trying to pick it up the box turns out to be much bigger than the expected blocks. There are then a few useless motions that give the impression of confusion, but soon the box is grabbed and put back into the right place. This was quite a surprise when it was tried for the first time.

The method was not expected to be so powerful and is really intended to take care only of minor sequencing troubles. For instance, the program may call for the hand to be put flat on the table, and then to search for a block. If now the hand should hit the block before it reaches the table, it will continue right on. Earlier, in MHI, there would have been a good chance that the hand would have pushed the block away, or even turned it over in an attempt to reach the table first.

The third and last change, from MHI to THI, was the introduction of suitable subroutines for functions more complex than simple statements but still meaningful only if used together with others. These

Subroutines are written in the THI language, but the final program, consisting of a series of these subroutines appears in a language reasonably close to English. There are now four types of these pseudo-English instructions:

1) "goto" brings the hand in touch with an object, either by going to the remembered place if the object has been encountered before, or by going into a systematic search motion.

2) "take" can only be used if an object touches the hand. It will then grasp the object and lift it.

3) "place" positions an object with respect to some material point of reference, on top of it, on its sides or behind it.

4) "scan" explores an object to find out its absolute position, its size with respect to the maximum opening of the fingers and its position with respect to the hand.

There are approximately half a dozen subroutines of each type, each with several points of entry and exit. The choice of a particular subroutine and of its points of entry and exit depends on the conditions of the environment, such as size of the object, location of the object, etc. This information about the outside world is contained in so-called "models." Whenever an object is hit or scanned for the first time it gets its name (m0 - m7), and it is later referred to by this name. Associated with the name are registers for the absolute position, the position relative to the hand and the size. These models are updated at the end of each subroutine; "scan" fills in all these data, and any other routine that touches the object will fill in the information about the relative position. Furthermore, each statement in the

pseudo-English language has two arguments, one refers to the name of the model concerned and the other contains an indication of what the subroutine is supposed to do in terms of model changes that should result from this operation. The choice of a routine of the right type depends then on what the named model indicates and on the mentioned expected model change.

As an example we give a program that finds an object m2 and then puts it on top of the earlier explored object m1.

Program No. 4

Type	Model involved	Model change	Comments
goto	m2	mL	This action should result in an updating of the <u>model</u> for <u>location</u>
scan	m2	md	Scanning results in new <u>model</u> <u>dimension</u>
take	m2	hld	Code for <u>hold</u>
goto	m1	mL	
place	m1	mL xb yh zd	Code for "on top of"

It is easy to see that programming in these expressions is extremely quick and easy. A special compiler permits doing this on the TX-0 on-line equipment. The real work goes, as before, into the subroutines, but the language is considerably more flexible than BBP-1, which was the program to build structures with blocks using MHI principles. It should probably be called a "universal language for building with blocks". It has the same faults as BBP-1 and is therefore only

Justified in connection with the following addition to the interrupt feature.

We can now describe what happens in the interrupt case when no appropriate THI instruction is found. If this happens during a "take" program, the scanning routine is entered and the nature of the obstacle determined. The pseudo-English program is then searched for the mentioning of such an object. If one is mentioned, the control is transferred to that instruction. If none is mentioned, the hand is withdrawn from the disturbance until the disturbance disappears but not more than about 2 inches, and then the program continues to do whatever it did before. Such simple motions can of course be generated automatically.

It is evident that a program search according to this criterion makes sense only if an encountered object does not correspond with the expected object; therefore this kind of search is limited to interruptions in the "take" case. In all other situations the withdrawing action is called for immediately whenever no appropriate THI instruction can be found.

To summarize the differences between MHI and THI a comparative Table III has been drawn up. Both the programming differences and the behavior differences are shown.

In order to illustrate the behavior of MH-1 a motion picture has been made showing the hand in operation with the programming system described as above. The program is first written "on-line" on the Flexowriter and then run directly. The influence of the interrupt

feature is demonstrated by alternatively making it inoperative, thus showing the discussed rigidity and the overriding of unexpected stimuli as usual in MHI. By replacing blocks with different objects the interrupt feature on the pseudo-English level is then demonstrated.

For the benefit of the reader who has not seen the picture, the scenario is included in the appendix, pointing out the places where MH-1 is successful and where it fails.

Table III. Comparison between MHI and THI

Programs	MHI	THI
Programs written in terms of	Motors to be used until indicated sense organs reach given value.	Output of sense organs. Motors are assigned automatically.
Unexpected real world events and conditions are	Ignored. Program goes right on.	Detected. Program determines remedy by selecting appropriate routine.
Performance in general	Rigid. Programs are followed under all circumstances.	Flexible. Adapts itself to the real world.
Most frequent errors	Originate from programs that do not foresee accurately enough the difficulties the hand gets into. These are programmer's mistakes.	Originate from wrong decisions concerning the remedy of unexpected emergencies. These decisions are machine-made.
Flexibility of programming language	Poor. New program required for each new application.	Good. Subroutines are rearranged automatically according to new requirements.
Memory for real world	None explicit. Has to be incorporated into handwritten program.	Essential real world events are remembered automatically in the form of models.
Programming work	Considerable for each new program.	Considerable for the sub-routines. Small once the subroutines exist.
Automatic routine improvement or routine generation	None.	None.

Chapter IX

DISCUSSION AND EVALUATION OF THE THI PROGRAMMING APPROACH

9.1 A New Source of Errors

Disappointingly, the THI program produces nearly as many mistakes as the MHI program on which a similar amount of development time was spent. It turns out, again, that the most frequent mistakes are of a fundamental nature. THI has enough mechanical intelligence built in so that it does not override unexpected stimuli, it decides instead on some better course of action. But evidently it can make mistakes in these decisions, too, and that is what causes most of these blunders. It can choose an improper way out of some difficulty.

The programmer can influence the occurrence and graveness of these mistakes to some extent by organizing the program properly. He does not have to foresee the unexpected stimuli, but he has to follow certain rather broad rules concerning the layout of the program so that the fixed interrupt-search procedure will be directed toward the right places in the program. What the nature of this organization of the program should be depends, of course, on the detailed search pattern, and some experimentation with different patterns has been carried out. No pattern is much better than another; each one provides a good strategy for some (we hope, most) cases. However, there always remain some situations for which a particular strategy is improper. For instance, bumping into the table has different implications according to specific situations. Since the interrupt-search will always do the same thing, these differences have to be reflected in the program, or mistakes will occur.

In other words, delegating responsibility and decision-making to the machine has the same consequences as delegating them to human beings: It relieves the work load, but it also introduces a new source of error. The only solution is to make sure that the subordinate has as much intelligence as possible, and THI really has a bare minimum.

9.2 Artificial Animals

The reader will probably find that the complexity of the operations performed by MH-1 has not increased in proportion to the increase in complexity of equipment, as compared with the more modest artificial animals. There are two reasons for this: First, I did not work with complex performances as a goal; I worked toward a sound and interesting way of calling forth and controlling simple actions. Instead of writing complicated programs in MHI I preferred to investigate the more fundamental questions raised by it and by THI. Second, this not only absorbs one's own time, but it also absorbs computer time and capacity.

Much of the justification for conventional artificial animals lies in the study of human reactions toward them, reactions of either an emotional or intellectual nature. The interior mechanisms of these animals are not really of any fundamental interest. In our case, the situation is exactly reversed. The information processes are of primary interest, and not the actions achieved. In order to make possible internal processes that merit any attention as contrasted with behavior that merits attention, the use of a few dozen active control elements, each of which has one specific function in terms of the behavior of the animal, is not sufficient. We are interested in

The investigation of broader principles and more general mechanisms that require more equipment, although one sample of the final behavior may not look much different from that of a specialized machine.

The task predominantly used, the handling of blocks, was originally suggested by Ivan Sutherland and turned out to be almost ideally suited for this work. It involves in elementary form two out of the three main functions of hands: It uses the hand as a sense organ, and the hand as a manipulating device. The third function, use of the hand as a coupling organ man-tool is too difficult to use in this context, because of the increased demands on sense organs and information processing (It is more difficult to eat with a knife and a fork than to eat with your hands!).

9.3 Programming Stupid Computers

We have mentioned that because of external speed limitations the main objections to interpretive programming are not valid. We now see that one of the main advantages of that method is that programs can be written which permit the machine to "understand" them. By "understanding" we shall denote hereafter the capability of correlating one thing (e.g., a model, or a statement) with another thing (e.g., the real world). In this sense, the computer, during the interrupt search, makes use of a primitive form of understanding of the program it contains and of the real world. It compares the available programs with the unexpected situation in the outside world and chooses a part of a routine which seems to be appropriate to deal with this unexpected situation. Interpretive techniques permit the construction of languages

that are easy to understand, both for the machine and for humans. THI is easy to understand because it consists of sense information patterns, and information about the outside world arrives in these same patterns. The mentioned correlation is therefore easy to achieve in the THI language. MHI is difficult to understand because of the lack of correspondence between the motors, in terms of which the motions are expressed, and the sense organs; that is, between the language used to express the intentions and the language used to express the state of the outside world.

Is the use of the word "understanding" justified here? Perhaps the most popular complaint about digital computers is concerned with their "stupidity": They do exactly what they are programmed to do, whether this makes sense or not. The slightest mistake in the logic of the program will foul up the whole operation. I think that MH-1 with THI is in a somewhat better position. Some things that have been overlooked during the programming, some sequencing errors, and similar situations are corrected automatically, although there still is, of course, a vast number of possible errors that are disastrous.

I think that the responsibility for machine "stupidity" is misplaced. It is not the digital computer that is inherently stupid. In order to understand something, according to our definition, the computer has to have two things to correlate. If a computer should understand its activity it has to be able to compare it with some model of that activity. But we usually neglect to give the computer any access to such a second aspect of its activity. For example, a

computer has no access to abstract mathematics, therefore it cannot possibly show any understanding while running mathematical programs numerically. But the TX-0 has a sense for the outside world, provided by the sense organs of the hand. This direct access to the real world is an absolute requirement for even such a rudimentary form of understanding as the kind that the THI program shows.

9.4 Automatic Control and Automatic Operation

We pointed out several times the ways in which MH-1 goes beyond classical control systems. Its coupling to the real world, by the use of both descriptive rather than quantitative sensory information (outside-in control) and by the programmed understanding of the routines contained in the memory of the TX-0 permits mechanization of the decision functions in what may be called the guidance of the system. We shall summarize this with a sketch of the general organization of a system that is supposed to display intelligence in its activity in the framework of the real world (compare Figure 7). The descriptive sense organs provide the operator with an indication of the state of the real world. By some mechanism involving his memory of possible actions and his set of instructions or intentions as to his activity in the real world the operator decides what to do in each instance. These decisions set goals for the control system that supervises the output of the motor organs by means of feedback involving the inside-out sense organs and tries to reach the specified goals.

The part of the diagram that is underneath the dotted line represents the functions that are carried out by conventional control systems. The remaining functions are performed by a human operator.

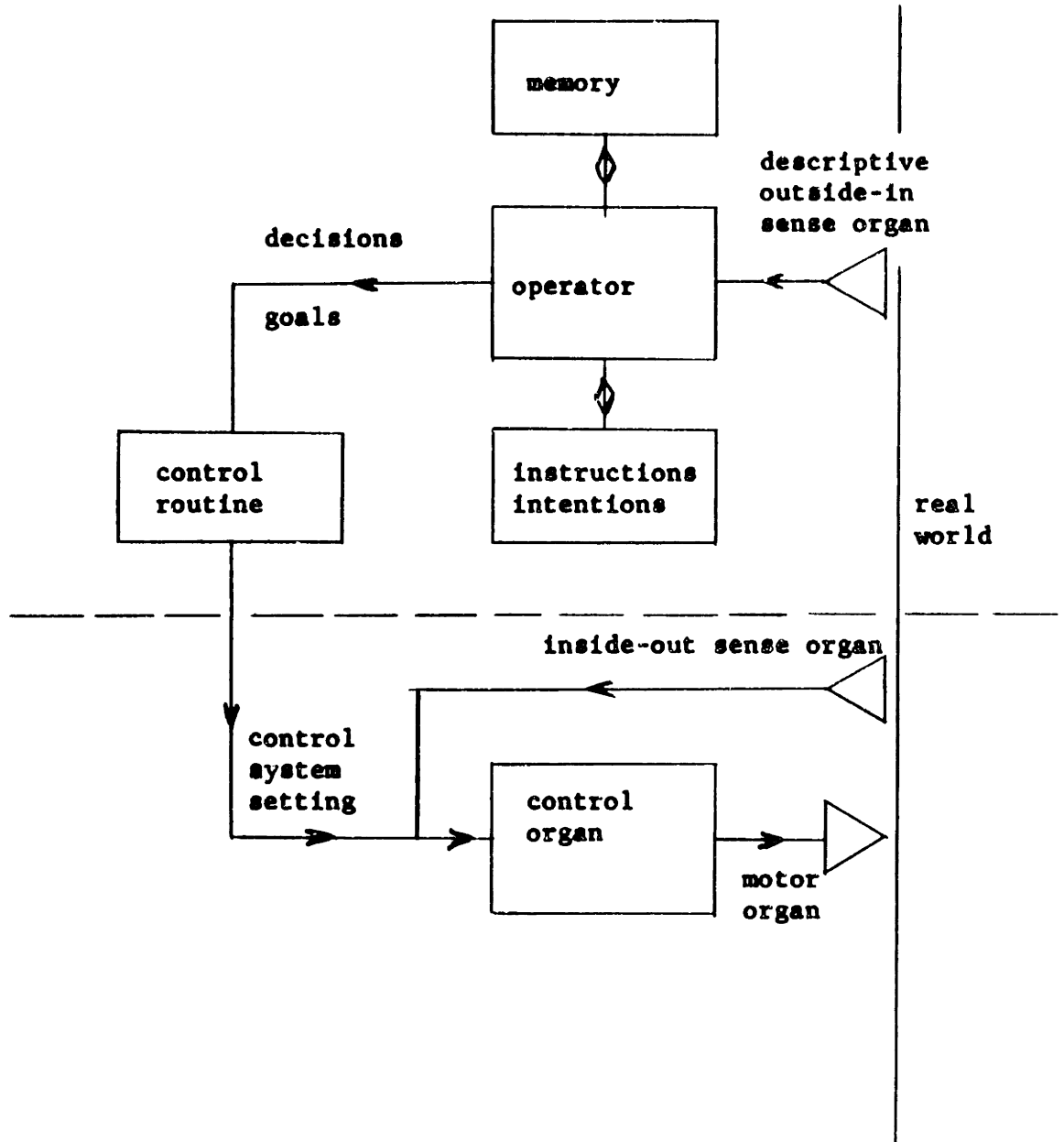


Fig. 7. Blockdiagram of a general, real-world coupled automaton.

9-1 is a simple attempt to mechanize some of the functions above the dotted line, too.

9.5 Artificial Intelligence

We shall now try to compare this work with other recent efforts in the general area of mechanization of cognitive processes. Most of these efforts have been concentrated in the area of problem solving, either in a game situation (chess, checkers) or in a mathematical situation (geometry, symbolic logic, formal integration). Outstanding success has been achieved by "heuristic programming."

In a problem we are given one configuration of data and we have to obtain a well-defined different configuration (solution). There must exist tests that indicate whether some configuration is the desired final one or not. In the most interesting cases there are no algorithms leading from the initial configuration to the end solution. There are, however, possible steps that can be taken at each time, and one is faced with the problem of picking a good sequence of steps which will lead from the initial situation to the solution. It is not possible to decide ahead of time which steps to choose; rather, one tries a step, evaluates the progress made with it, tries another one and finally selects the best step. Often one even considers several steps together and judges their combined effect. The important characteristic is that we are confronted with a lack of knowledge as to how to proceed. We have to find a way more or less by trial and error, aided by some general heuristic rules. This analysis holds for chess playing, as well as for proving theorems in geometry or mathematical logic.

But this analysis does not apply to the "problem" we are faced with if we want to do something with our own hands. In our everyday life there is no uncertainty about how to do something specific with our hands, no trial-and-error procedure is needed to find an appropriate method: We know how to do it. The application of knowledge results in a tremendous increase of speed of operation. We have a perfectly good method available right away, we do not have to discover it painstakingly. This advantage holds for computers, as well as for human beings. It is therefore essential that computers be able to solve problems not only by heuristic methods, but also from knowledge. Heuristic methods are to be used in areas in which ways leading to the solution of a problem have to be found. The main problem is the evaluation of possible ways. Knowledge is used in areas in which such ways are well known. The main problem is therefore to achieve efficient storage, accessibility, and use of this information.

Let us now investigate how this problem is handled in the THI language.

9.6 The Real World as Seen by MH-1

Unless we indulge in metaphysics we analyze our own cognitive processes by scrutinizing our personal impressions of our environment. In order to make it easier to evaluate the cognitive processes performed by MH-1, we shall attempt to describe the real world as seen by the TX-0 when it operates the mechanical hand. A few assumptions and some clarification are necessary before we can do so.

1) In order to enable the TX-0 to express its impressions, we have to attribute to it some form of consciousness. The reader should keep in mind that we do this as a "thought experiment." This is not a description of actual processes in the computer. This whole section should be placed in quotation marks; it should be taken with a grain of salt.

2) In attempting to design that fictitious consciousness we are immediately faced with a mind-body problem for the TX-0 computer: We have to decide how this fictitious consciousness (mind) is related to the physical processes in the computer (body). As a working hypothesis let us assume that the mental impressions are simply projections on a screen called "consciousness," of some of the underlying physical processes in the computer, but that there is no back action of these projections on the physical processes. In other words, we assume that the information processing is controlled completely and exclusively by the physical properties of the processing system. Some of these processes are then projected on a screen, a little like an Eidophor television picture, and the pictures on that screen are what happens in the mind. According to this view, the mind and the conscious impressions are a charming, colorful, but useless, luxury, an intricate shadow play by self-governed incomprehensible actors against the wall of a cave* - or a

* Plato

ing picture generated by a mechanism and by forces that are at present beyond our understanding and beyond our influence, but we sit and watch and enjoy it.

3) We have now to decide which of the processes in the TX-0 are to be projected and become mental, conscious impressions. A comparison with our own conscious impressions indicates that the feedback loops for the control of motions are not conscious; this means that the lowest level of programs should not be projected. The sequencing of simple steps leading to a goal is, however, very often conscious. We may be conscious of the steps taken in picking up a pencil, particularly if we want to steal a gold pencil from somebody's desk. On the other hand, if we pick up our old pencil from our desk in order to play with it while we are talking with somebody, this is not a conscious act. This level of complexity seems to be about the lower level of consciousness, and we shall therefore project on the screen of consciousness of the TX-0 programs that are on this or on a higher level.

4) Next, we shall catalog the possible conscious TX-0 impressions. We shall state them in the first person singular for the TX-0, in order to acquaint the reader with the fictitious personality of the TX-0.

a) Immediate sensory input:

I know what all the sense organs indicate now.

I know whether any sense organs have changed in the past moment.

I know whether such a change is to be expected during my present activity or whether it comes as an unexpected change.

b) Program:

I know what each of my sense organs should indicate at each moment. The necessary processes to bring about this desired state are determined in my unconscious. If, for instance, I want to touch something with my fingertips, I automatically move my hand forward.

I can look up all of the programs in my memory. They are sequences of instructions stating what my sense organs should indicate at each moment.

I possess a table of contents of all of the routines that my memory contains. This permits me to select routines in accordance with the general instructions that I am given (for instance "goto") and with some previous events that I remember.

c) Memory:

I can remember certain occurrences that are pointed out to me by the routines. I can remember the location of these occurrences, I can remember what they felt like (for instance, pressure against the left finger). After such occurrences I usually have to apply a scan routine, and as a result of what happens during that routine I have to remember another few bits that I call properties a, b and c. (Note: these are the bits that indicate the size of the object. This is unintelligible to MH-1 because it does not have a notion of objects with this model of consciousness. There is no process in the computer corresponding to "objects," but there are processes corresponding to property a, b or c, namely the storing of this information in the memory. We remember that according to the definition of consciousness which has been given, only phenomena that are represented by physical

processes in the computer can appear in the mind.)

5) The format of the transcripts of the mental processes of the TX-0 will be: Normally the TX-0 is conscious of its intentions. Each new intention is preceded by a line -, and stated as an order. If the intention is unique, the TX-0 becomes conscious of the next intentions as soon as the sense organs indicate that the first intention is fulfilled. This is indicated by "o.k." If an intention has several possible terminations, the occurrence of the specific terminating condition will be indicated. WAIT indicates the occurrence of an interruption based upon an unexpected response of a sense organ, and the period spent for the program search. This search itself is unconscious, as our own association processes are, but its result is projected into the conscious domain.

In parentheses () we give remarks that facilitate the relating of the mental processes to the actual processes performed by MH-1. We are now ready to proceed to a specific example of the mental processes of the TX-0. We shall use our old block-and-box program. Let us assume that MH-1 has just run into the box, has gone through the scan routine and remembers now the position and the size of the box, in the form of a model. This model is a collection of several bits of information about the event of running into the box and about the results of the scanning process. MH-1 should now proceed to find the blocks (goto m2; scan m2).

Here is the transcript of the impressions of the TX-0 while the hand searches for the block and determines its properties:

- don't make any involuntary forward-backward motions. O.K.
- wait if unexpected changes occur in your sense of touch. O.K.
- go up a foot, and go to the back and to the left (the initial search position). O.K.
- point the hand forward, in a horizontal position. O.K.
- now touch something with the palm (hand moves down to the table). O.K.
- now touch with both your fingertips and the palm (hand flat on the table). O.K.
- relieve the pressure from both fingertips and palm, and lift the hand about one inch. O.K.
- now you want to feel something with the outside of your right finger (search motion to the right). Stop if you feel something with your finger, or if you cannot feel anything stop after you have moved approximately one foot, or if you have gone as far right as you can. O.K., I have moved one foot.
- put hand flat down, move another foot, all exactly as before. (to make sure hand does not search too far above the table). O.K., I tried as you said, I wanted to touch something with my palm, but instead I touched something with the fingertips (while going down the fingertips touched the table before the wrist did).
- WAIT---, that is all right. You should in the next step, when you put down your hand flat anyway. Proceed with that. O.K., I moved a foot.
- do all that again, hand flat down etc. ... O.K., I have moved as far right as I can.
- now go forward almost one inch, then try to feel something with your left finger. Stop if you feel something against your left finger, or if you have moved a foot and found nothing, or if you have gone as far left as you can. (Search proceeds on a somewhat advanced new line on the table, from right to left.) O.K., something is pressing against my left fingertip. (A block is in front of the left finger.)
- now this is a special event. Remember that something is in front of your left finger, call it m2. O.K.

-look up in your set of instructions what to do next. O.K., it says I should scan m2. Since m2 is on my left, routine c25 is appropriate according to my catalog of scan routines. (Control is transferred to routine c25.)

-back up 1/2 inch and point the hand straight forward. (Get in position for scanning motion.) O.K.

-now, as in the old routine, put your hand down flat and lift it again 1/2 inch. O.K.

-touch something with your left finger (begin scanning routine). Stop if something touches the finger or if you have moved 1/2 foot. O.K., I tried but instead I feel something with the bottom side of my fingers.

WAIT---. (The hand has touched the table again because it was not lifted enough). Try to put the hand down flat, lift it as before, and then try to go left again. O.K., I feel something against my left finger.

-go back and go right approximately 3 inches, and then try again. O.K., now the left fingertip responds.

-continue; stop if either your left fingertip loses contact, or if your right finger starts to feel something. (This determines the size of the object. If the left finger loses contact with the object before the right finger makes contact, the object is smaller than the maximum finger opening.) O.K., left finger does not feel anything any more.

-Now this is a special occurrence; remember that m2 is at this position, and that it has property a (Is small with respect to the finger opening.) O.K.

-Back up a little and look up in the program what comes next.

After this bit of TX-0 soul-searching (every psychiatrist should envy us for the accuracy and completeness with which we can do it, even the whole unconscious mechanism is clear), what can we now say about the world as seen by the TX-0?

The salient feature is that MH-1 does not have what we would call a conscious picture of the real world, nor of the objects in it, nor, for that matter, of itself. Its impressions of the real world consist of events, of which certain ones are remembered upon command, and the impression of itself consists of plans of action. Specific actions are chosen in accordance with the instructions received (the highest level of program) and the recollections of previous events (the state of the models). MH-1 is not concerned with the causes of these events, it is not trying to organize them into a meaningful world, it is not trying to attribute them to something called objects. In other words, even with its fictitious mind, the TX-0 does not know that it is playing with blocks.

To summarize, MH-1 operates according to an association mechanism controlled by the highest level programs and by the models. These associations are now established by the programmer, but they could be established by trial-and-error procedures, or more efficiently by what is commonly called learning. Educated people might scorn such a form of "intelligence". They feel that a truly intelligent creature should operate somewhat like this:

There is a table and wooden blocks. Since blocks in general are subject to gravity and since there is a table, the blocks are probably sitting on the table. Therefore if we want to find them we have to search the surface of the table. Now if we just bring the hand down we should feel the table because it is a member of the class of palpable things. The best way to scan it is in TV-like scanning

attern. If there are objects sitting on the table, we should hit them sidewise or conceivably with the fingertips. So if we hit something the chances are that it would be a block,

In other words, this form of intelligence requires a list of all of the properties of all objects and conditions in the real world. The intelligent creature then applies its logic to these properties and derives from them in due order well-founded procedures that will bring about the desired goal.

We have now explained two possible mechanisms of storing and using information concerning a domain in which the relations between the objects are known, and knowledge rather than trial and error decides which method to use in order to solve a problem. We can make use of an associative memory mechanism as THI does. This is realized in a computer by the use of descriptive languages. The information about the relations between objects and the useful methods is contained in the description of the events in the real world and of one's specific intentions. Or we can use the logical approach, in which a set of logical rules governs the behavior of arbitrary objects, and all we have to know is these rules and how to apply them to the objects with given properties.

To establish which of these two systems of intelligence we human beings use has been a philosophical problem ever since David Hume wrote "An Inquiry Concerning Human Understanding." Since this is an Sc.D. thesis and not a Ph.D. thesis there is no reason for us to participate in that dispute, tempting as it may be. We shall be

satisfied to point out only that the association mechanism is the one that has shown most promise in the computer-based study of the imitation of cognitive processes.

Having finally investigated two methods for the storage and use of information in areas in which we know what we want to do, and having turned the evaluation of these mechanisms over to the philosophers, we shall now take away the fictitious mind of the TX-0, degrade it to a conventional digital computer, and discuss some more practical engineering problems.

9.7 Recommendations for Future Work

This question is easy to answer by just following the established procedure, namely to ask what is wrong with the last approach, THI in this case. Although THI makes possible the automatic selection of appropriate routines both during the normal course of a specified action and during an unexpected interruption, these routines have still to be written by a human programmer who analyzes the environment and the task. And then these routines have to be tried out on the computer, and much time has to be spent to adjust the parameters in the single commands. It is often quite difficult to determine which parameter is responsible for some poor motion, and two hours may be required to iron out a routine like "take." In summary, the human operator is still relied upon too much. Too little of the analysis of environment and task is performed automatically, too few of the operating decisions are made by the computer.

It is my opinion that the associative, automatic combination of simple routines into more complex motions, which are then again combined into still more complicated actions, is basically a good solution. The trouble with THI is that the most basic routines, for instance, the "go to" or "take" are already too complex. The basic routines should not contain more than that which is needed for simple situations such as:

Something touches the left finger inside, touch it now with the right finger inside.

Something touches the left finger from above, move the finger in such a way that it is touched from underneath.

Such really elementary routines, which could even be generated by watching the sense output of random motions, just as in the way that children learn to use their hands, will then be automatically combined into more complex motions in the same way as THI now puts together the right routines to find and grasp a block, by associations triggered by both the real-world conditions and the intentions incorporated into the program. The important shift is in the logical complexity of the most basic routines.

This shift will contribute to the remedy of another essential fault of the THI system. The models that are used in THI are not sufficiently related to the next lower level of information. In other words, the model indication that an object is securely grasped is not organically related to the sequence of sense inputs indicating that event. The models are code words written down by the programmer,

according to the branching points in the routines. They are not sufficiently descriptive. The models should be related so closely to the sense input during the execution of the routines that it will be possible to search routines with respect to the information contained in a model, as well as modify the models automatically by going through a routine. A model should therefore be a summarizing pattern standing for a whole sequence of sense input patterns describing in detail a certain motion. This summarizing pattern serves as a label in the association network that is used to find appropriate routines in the memory.

The reader need not be overly concerned if he cannot understand these remarks fully. It seems to me that this is what can be abstracted and extrapolated from a detailed knowledge of the mechanism of THI, and it is what I myself would try to understand if I were to spend another half-year on developing the successor to THI. After all, these are only some hints of what I have in mind to do next, and no explanation of it.

It is fortunately easier to analyze the reason for the present unsatisfactory level of complexity of the basic THI routines: It is the nature of the sense organs for touch. These sense organs are very often satisfactory only if they are interpreted together with the specific program provided by the operator. They will indicate by themselves, for instance, whether something touches the left fingertip, but in order to determine the location of the edge of a block one has to observe the output of the sense organs during a test motion that

is more or less complicated. The sense output makes no sense if it is not connected with a specific test motion, and the logical complexity of the test motion is far beyond the capabilities of the associative mechanism used for the information processing in THI. The most basic routines must therefore contain these test motions ready-made by the programmer, and hence are necessarily quite complex.

Together with a new programming language, a new sense of touch will be required to resolve this difficulty. The new sense organs would resemble the human skin much more closely. There would be approximately one element every $1/8$ ". This element would simply contribute one bit to the information contained in the composite sense input pattern of tactile information. It would not have a preassigned function like detection of things touching the fingertips, but its output would be used by the computer in the framework of the whole sense pattern according to the circumstances. It is simply impossible to have special sense elements for all conceivable special tasks that the hand might have to perform.

There is one special feature of a general-purpose digital computer that it would be nice to have for similar applications. Many bits (a few hundred) are generated by such sense organs, and their transport into the machine is a matter of some concern. One good way would be to have a set of registers that can be addressed by the computer and can be set by the output of these sense elements directly, so that all of the sense information could be transmitted into the machine by one operation. This scheme is equivalent to a direct transmission of the sense information into the computer memory.

This is the only special feature that can really be advocated for similar computer applications. In general, it pays to beware of special-purpose equipment, simply because one can never be sure that at a later stage one would not like to have an additional feature that is difficult to obtain from the special machine, but which could with a usually small loss of time be obtained by programming a general-purpose machine. This loss of time is more than compensated for by the added flexibility.

It would be a violation of the experimental procedure adhered to throughout this work if we tried to speculate further. In the MHI, as well as in the THI phase, plenty of unexpected but useful suggestions were obtained by interpreting the behavior of MH-1. This would undoubtedly be true for the next step, too. If we could forecast with any accuracy what can probably be done in the more remote future this would imply that we do not really need these exploratory experiments. This is not my intention. The long-range objective of coupling automata directly to the real world, without the continued assistance of a human operator, is in itself bold enough without any additional speculations as to the future of such autcmata.

9.8 Application

This work is of interest with respect to two possible applications: The construction of mechanical hands as useful tools, and the exploitation of the philosophy behind the control problem.

1) Mechanical Hands for Industrial Applications

One can look at this question in two ways. The pessimistic

production engineer will state that, although there is considerable interest in general-purpose programmable tools of the form of a human hand and arm^{2,8}, at present the demand for such a tool falls off rapidly if it should cost more than 15.000\$⁴. A magnetic tape controlled manipulator selling for 50.000\$ did not find any customers and did therefore not go into production⁸. Now U.S. Steel is apparently making another attempt in the 15-20.000\$ class, about which no details are available.

At present, the replacement of the magnetic tape control by a control computer would add at least another 100.000\$ to the cost, so that the prospects appear dim indeed. Furthermore one can calculate that one mechanical hand could work 24 hours a day and therefore replace three workers. If the investment alone, not considering the maintenance and supervision costs, should be made up for by cost savings over two years as is good practice, we again hit that 15-20.000\$ ceiling. Finally, considering the present situation on the labor market, a surplus of unqualified labor forces and a shortage of qualified people, one might have doubts about the justification of a mechanical hand which, as a tool, would probably be limited to rather unqualified operations.

On the other hand the overly pessimistic view of the production engineer is fortunately compensated for by the overly optimistic outlook of the research man. He sees the situation in more glowing colors:

First he argues that the cost of machine computation is coming down rapidly with the introduction of bigger and faster machines. Professor H. Teager quotes the following relative cost per unit of capacity of program execution for automatic computers²³.

IBM 650	(widely used in 1955)	7.5
IBM 709	(widely used in 1959)	1.0
IBM 7090	(widely used in 1961)	0.25
IBM Stretch	("widely used" in 1963)	0.05

This comparison indicates that the cost decreases by a factor of 2 each year. The cost of operating a mechanical hand by computer will therefore reach the quoted ceiling within a few years that will be needed anyway to bring the artificial hands to a state of greater perfection than MH-1.

A company will then have a big, central computer which operates several dozen of these hands. All over the factory there will be connectors, and whenever the need for an additional hand arises somewhere, one is brought to that place and connected to the computer. Then one tells the computer what to do, and that is all.

The optimistic research man will further point out all the advantages of such machines which the pessimistic production man does not seem to realize:

- 1) A mechanical hand is a most flexible tool. The company will have a pool of them and use them for whatever application one has in mind, especially applications for which the demand does not justify the development of special machines. After the short

production run the hand can then be used for something else.

2) The mechanical hand does not get tired and will work 24 hours a day, with the exception of a little time for maintenance.

3) Accidents will damage the hand, but they will not have any of the regrettable consequences human accidents unfortunately have so often.

4) Mechanical hands will not form unions or demand additional fringe benefits. They need no coffee breaks or relief time.

5) There are no pension and health plans for them. They are thrown away after they are worn out. They don't mind being "exploited" in the Marxist sense.

I leave it to the reader to choose his position on the scale between the pessimist and the optimist.

ii) Mechanical Hands in Hostile Environments

We are talking about the very problem that gave rise to the appearance of servomanipulators: Manipulation in radioactive areas. Other adverse environments may be found in deep-sea exploration and space exploration. This situation is radically different from the described production situation because under these circumstances even very simple actions become highly qualified ones. However, the same considerations on the use of a human operator versus the use of a computer to control the servomanipulator apply, except in one case. The manipulator may be so far removed from the human operator that communication with it becomes difficult, either because of delay or because of the power required to ensure a reasonably good transmission.

This situation arises in space exploration. One way to solve this difficulty is to send some mechanical intelligence with the manipulator, in the form of a computer. One can easily see that the transmission of a program like the one in our example is much easier than the establishment of a complete, real-time telecontrol system, besides solving the nasty delay problem. Including the apparatusive delays (scanning rate of the TV system) it is estimated that the delay in such a loop extending to the moon would be approximately 3 seconds. Assume that the manipulator should pick up a rock on the moon. The operator will see that the hand is in the appropriate position only 1-1/2 seconds after this event occurred. Another 1-1/2 seconds will pass until the signal to stop reaches the manipulator. Unless the manipulator works very slowly it will have moved beyond the rock by that time. But if there were sufficient intelligence incorporated into the system on the moon, we would simply line up the manipulator and then tell it to go forward until it touches the rock, similar to the actions that are taken by MH-1 when it searches for objects.

The cost of providing that mechanical intelligence is irrelevant in similar applications. It might even be that the accompanying simplification of some of the other communication equipment more than makes up for the cost of the computer.

Application of the Programming Principles

I am mainly thinking now of the accessibility of some non-human environment to the computer in order to enable it to use the described forms of awareness and understanding. We have only begun to scratch the surface of these problems, and they will probably remain in the research stage for some time to come. This research will have to proceed in two directions:

a) Further clarification of the mentioned phenomena and of similar phenomena. We could call this some form of psychological research, since the ideas come from psychology, and the experimental subjects are computers. This research will be interesting both from a human and from a technical point of view.

b) Development of the necessary internal world for machines, and construction of the required sense organs. This process requires a profound understanding of the environment in which the machine has to work.

Although the exploitation of this research may be even further off than the construction of useful programmable mechanical hands, I think that ultimately it is far more interesting. It seems difficult to build a mechanical hand that will surpass our own human hand. However, I am optimistic enough to think that a better fundamental understanding of our human information processing will open all kinds of doors for interesting technical applications. I therefore consider that aspect of MH-1 to be the more important one.

The Study of Cognitive Processes

It has been pointed out that the advantage of artificial animals in the study of cognitive processes consists in the purity of the experimental object. Physics developed out of the interest in astronomy; here the mechanical principles show up in their purest form. Psychology is in an unfortunate situation in that it is almost impossible to get pure natural systems to experiment with or to observe. The lower animals and humans are so complex that it is very difficult - if not impossible - to control an experiment with them properly. It is, for instance, very difficult to discuss meaningfully the importance of memory on intelligence because it is difficult to separate out the contribution of memory. A natural memory can hardly be switched off in order to find out how the creature behaves without it. With artificial systems, however, this would be extremely easy.

An artificial system is under complete control. We can investigate its behavior as exactly as we want to by modifying components of the system or its principles of operation.

We do not suggest that artificial animals replace natural animals for all purposes of experimentation. There are cases in which one is specifically interested in the mechanisms used by nature, but, on the other hand, it has become evident that the study of natural organisms is too confusing to teach us much about the system organization of the information-processing apparatus. It is in this area that the artificial systems can be of much help

in furthering our understanding of these processes because they are so much simpler to experiment with.

One might ask why we want to build any cognitive capabilities into a machine that serves a technical purpose. The answer is that machines are designed to perform certain operations for us. It is logical to demand that these machines be able to cope with difficulties arising during their operation. They should be able to recognize an error and to understand what they can do about it. It is not always possible to think of all possible emergencies ahead of time, and to incorporate the remedies into the program used by the machine. The required intelligence should therefore be built into the machine itself.

9.10 Summary of the THI Phase

The two characteristics of this programming system are:

- 1) Awareness, achieved through constant surveillance of all of the sense organs and comparing their output to the output that is expected according to the program.
- 2) Understanding of the real world and of the stored programs, achieved by a procedure that chooses appropriate programs in the case of any real-world situation not anticipated by the programmer.

The discussion of these features brings out the following points:

- 1) Certain earlier faults of the behavior of MH-1 have been remedied (rigidity), but the process of understanding is not perfect, and consequently wrong decisions based on that incorrect understanding introduce a new source of errors.

2) Much of the apparent stupidity of digital computers ("they have to be told everything, and they do exactly what they are told to do, even if that makes no sense") springs from the lack of a proper environment for these machines. They have nothing with respect to which they can evaluate their performance. When working in a real world environment with MH-1, the TX-0 has done a few unexpectedly reasonable things, thanks to its limited capabilities of understanding and awareness.

3) It is shown that MH-1 is not working with a logical image of the real world but rather with an associative mechanism. Both are methods of information processing in a well-known, familiar environment, as contrasted with unexplored territories in which heuristic trial-and-error procedures are used.

4) The crucial importance of the outside-in sense elements has been demonstrated by the fact that all of THI is on an improper level of logical complexity because of an improper design philosophy behind these sense elements. The consequences of this are analyzed and remedies suggested.

5) Possible applications are discussed. In the opinion of the author the most promising ones are not those of the results on manipulation, but those that stem from an increased understanding of human information processing in general, an understanding that it is hoped can be furthered by similar studies.

Chapter X

CONCLUSIONS

An analysis of historic and modern automata, using as an example mechanical hands and arms, points out a possible way for the further development of these machines. It has been shown that all of these automata depend on human assistance in their relation to the outside world. The human operator analyzes the environment and the task imposed on the automaton and he then decides upon appropriate programs for the automaton, the execution of which will result in the desired action. Modern machines can gather and use information in order to reach the programmed goals as efficiently as possible. This is called "feedback control." Feedback replaces only part of the human assistances, the decisions that depend on an understanding of the outside world and on an understanding of the task still have to be made by the human operator.

This thesis is concerned with the problem of describing its task to the automaton in general terms, and of providing the automaton with its own means of analyzing that task in relation to the environment and its own capabilities. This understanding of environment and task should then enable the computer itself to make some of the decisions that were formerly made by the human operator.

The first result of this study, reflected in the design of the equipment, is the realization that the controlling quantities must not be purely internal ones, as, for instance, the setting of potentiometers for measuring the absolute position of the arm in

There should be external reference points. For example, the location of an object in the outside world, together with the instruction to touch that object with the right finger, is a very efficient positioning mechanism. Orienting the control functions and sense organs toward the real world rather than basing them on the information generated in the system itself makes any analysis of a task with respect to the situation in the outside world much simpler, and it is the basis for all of the further progress in this work.

The second result is the establishment of a partial analogy between the physical form of the signal used in a conventional servo system and the programming language used in the MH-1 system. In both cases the sense elements translate stimuli of the external world into the respective machine internal form: an analog signal or a word in the programming language. In both cases the form of that machine-internal representation depends on the information-processing procedures used. In the analog system an electric signal or a mechanical signal is chosen most often because there exist both convenient electrical and mechanical procedures for the processing of the information contained in the signals. Both programming languages used for MH-1 point out the corresponding importance of the structure of the processing language. Both languages showed their limitations by the fact that some processes that should be performed on the information contained in these languages are too difficult to perform in that particular language.

The sense organs are closely related to the design of the programming language because they act as transducers whose output should be directly in that language.

Although in its present form MH-1 still requires much human programming of an anticipative nature for the performance of practically useful actions such as finding and grasping objects, the environment can influence the automaton in its selection of programs that are appropriate with respect to the situation in the environment. The automaton can also interrupt by itself the execution of a program in case of an unforeseen emergency and substitute other programs which it finds have a good chance to get around the difficulty. It is in this context that two prerequisites appeared for the achievement of any purposeful communication between the automaton and the real world.

First, the computer can only operate on information that is contained in one of its registers. If the computer has to take the environment into account, it must contain a representation of the state of the environment and a representation of its own intentions. If a computer program has these properties and if the program is able to compare the real world with its intentions, then we say that it has "awareness." A computer can be coupled successfully to the real world without any human interference only insofar as the computer is made aware of the real world. In its dealings with events and conditions of which it is not aware the computer requires human assistance. The amount of human assistance required is therefore a measure of how incomplete the computer-contained representation of the real world is.

Second, if we want the computer to play an active role in the outside world, mere passive awareness is not enough. If the computer has to make decisions about the real world, it has to be able to correlate between the processes performed in its internal world and the processes in the outside world. We call this capacity "understanding." The experimental criterion for understanding is the percentage of correct decisions which the computer makes as compared with all of its decisions because only an efficient mechanism of understanding will enable the computer to generate in its inside world successful decisions about the real world.

The mechanisms of awareness and understanding used in THI are explained in detail and documented in a motion picture.

APPENDIX

1. The Experimental Equipment

The AMF model 8 servomanipulator is a commercial unit that has been described in reference 6. Its normal application is the handling of radioactive material in a hot cell. The manipulator front end (slave arm) is then in the hot cell, visible to the human operator through a lead-glass window. The operator, who is on the other side of the radiation shield, puts his hand on a handle on the master arm of the manipulator. By his own motions he moves the master arm, whose motions in turn are transmitted mechanically to the slave arm. The operator controls his motions according to his visual impressions and according to the forces that he can feel; these forces are transmitted back through the mechanical linkage between master and slave arm.

We have modified the manipulator in the following way: The device was shortened 1 foot in height and length and mounted on a movable rack. The horizontal motion of the through tube is blocked, as well as the indexing motor, because they are not needed. Endstops (rubber bumpers) have been provided for the motions involving considerable mass.

The motors: Each degree of freedom of the manipulator is now controlled by one 75-volt two-phase servomotor (Diehl FPE 25-11). These motors deliver an output of 2.2 in. oz of stall torque. The motors have a slight tendency toward two-phasing, which is harmless and

results only in an occasional slight noise. Each motor drives a reduction gear to generate the required torque. Maximum operating speeds are in the range 10-20 in./sec, the emphasis being placed on acceleration and stopping rather than on speed.

A stainless-steel rope wound around, or pressed against, a pulley transmits the torque to the cables and tapes of the manipulator. Whenever the rope is wound around the pulley we have actually a capstan device. A ball bearing with an excentric shaft generates the necessary preload, which can be regulated.

The sense of position: The coordinate of each degree of freedom is measured by a potentiometer. Since this function is fairly uncritical for the whole system performance, inexpensive wirewound one-turn potentiometers are used. Their coupling to the manipulator is not very rigid and hence they display some drift. This is taken up by proper programming. The potentiometers have a resistance of 1000 ohms and generate output voltages varying between 0 and -15 volts.

The sense of touch: Considerable time was spent on the development of continuously working sensors for touch. These organs were based upon an electrically conductive elastic material, fabricated by the Coe-Myer Corporation, Chicago, Illinois, and an electrically conductive elastic cement made by Emerson & Cummings (Eccobond 57C).

The main problems here are: first, the cleaning of the involved surfaces, especially critical on the Coe-Myer material, in order to permit adhesion of the cement to the electrodes as well as to the conducting plastic; and second, the embedding of the sense organs into a basis that would take up most of the deformations arising

under external forces, since the deformation of the electrodes should, even with the elastic cement, be kept to a minimum in order to avoid the separation of the conductive material from the electrodes.

However, we found out later that the continuous sense signals are almost always quantitized into two classes by the program, and therefore contributed only one bit of information: They were used just to detect whether something touched the fingers or not. The forces, it turned out, were much better obtained by monitoring the motor input. Some of the sense organs on the fingers were therefore changed to simply open and close contacts. However, no advantage was taken of the possibility of transferring such information much faster into the computer through a direct access to the live register (see Figure 8) instead of using the analog-to-digital converter. This direct, binary sense input would, in connection with the modifications of the sense organs and the introduction of improved feedback loops for the motors, be a necessary change in the equipment if the present work is continued.

Motor control: It has been pointed out that the feedback control loops for the motors are closed through the computer. If the digital position specification is increased each $1/10$ of a second by a small increment, a constant-speed motion will result. If the position specification remains constant, the motor control will generate the necessary tone to maintain the position of the arm. Because of the counterbalancing inherent in the design of the manipulator, little tone is required. It serves, for the most part, to counterbalance

unexpected disturbances. The hand can be operated without any tone, although the performance will be markedly inferior.

The motions of the hand are still of a rather jerky nature. It would therefore be desirable to tighten the control loop by the introduction of rate generators on the motors and by performing most of the stabilizing operations in external control networks instead of in the computer. This would, at the same time, make more time available for the processing of the information stemming from the vastly increased number of sense elements. Nothing of fundamental importance will be gained, however, from such a modification; the original decision has been quite satisfactory for the experiments thus far, and certainly more economical than the proposed modification.

Control Unit: Figure 8 shows a block diagram of the control system which has two main parts: motor part and sensor part.

1) Motor Part

Any 10 bits of the live register of the TX-0 can be selected on the IR plugboard to perform the motor control functions.

The motor selection is triggered by one of the TX-0 "operate" commands, a programmable pulse. The 6 speed bits then go into a digital-to-analog converter; the 3 selection bits go into a motor selection matrix, and the direction bit is stored in a flip-flop.

The motor selector selects one of the 7 motor command hold and timing circuits (one for each motor). The analog speed signal coming from the digital-to-analog converter is stored in the timing unit (it decays to zero with a time constant of 1 sec), and the

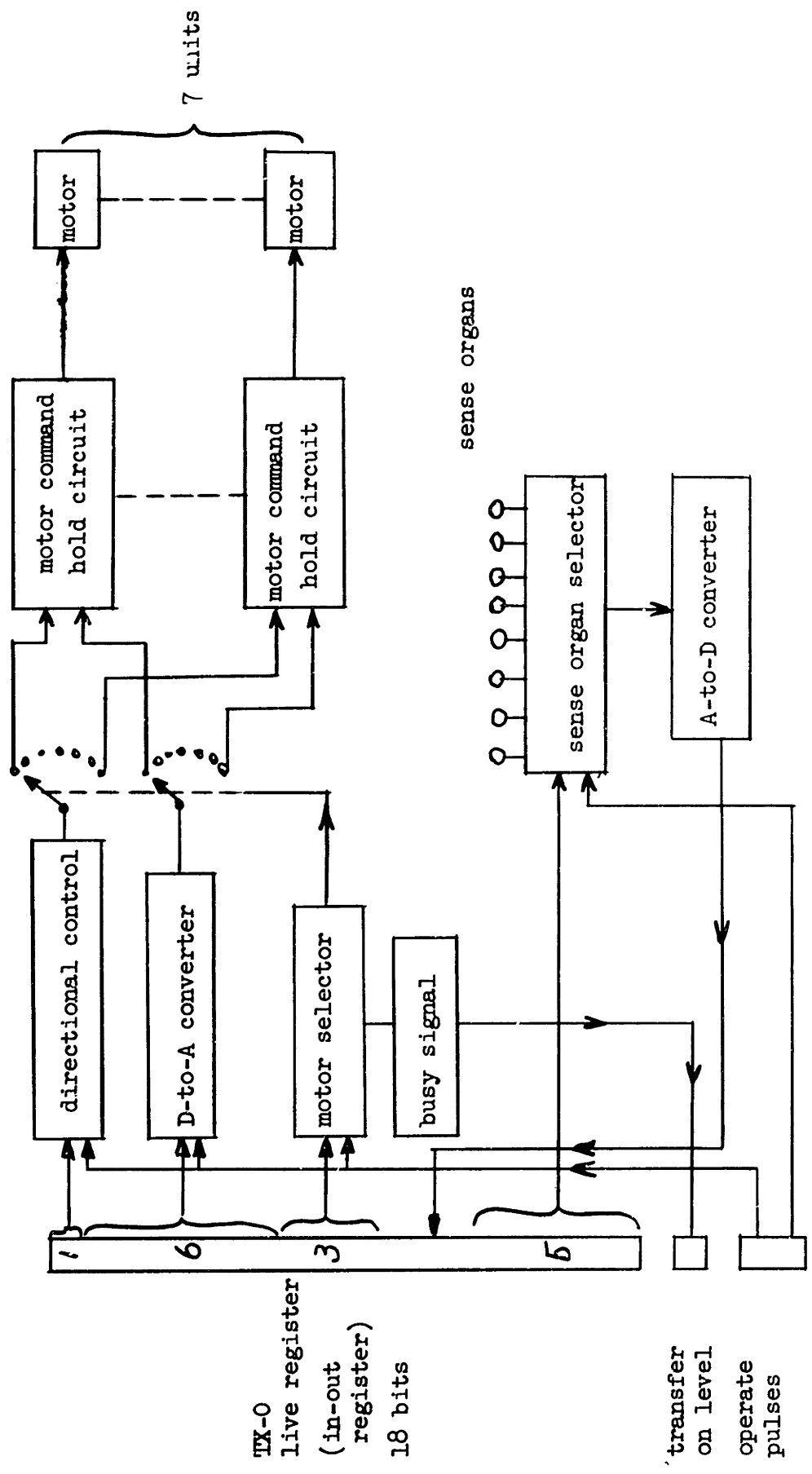


Fig. 8. Block diagram of the MH-1 control unit.

transfer on level operate pulses

direction bit is stored in the timing unit. The storing takes 3 msec, during which a busy signal goes into a "transfer on level" input of the TX-0 computer, indicating that no other motors can be selected during that time. Each motor timing unit is connected to a motor control unit that generates the required ac voltage and phase to drive the servos. Thus we see that for continuous motor operation the computer has to give a corresponding instruction at approximately each 1/10 second. Otherwise the last instruction given decays with a time constant of 1 second. New instructions, with specified speed and direction, are executed within the next 1/60 second (one ac cycle time).

2) Sensor Part

Any 5 bits of the live register (selected on the LR plugboard) are used to select one of the 32 sense organs. These bits are pulsed into the sense-selector buffer by one of the external pulses. The buffer immediately selects the corresponding sense line and holds it until a new line is selected. The analog-to-digital converter connected to the output of the sense-selection matrix can then be triggered and its output goes into the live register. This path is the same for all kinds of sense informations: position and touch. One sense input can also be connected to a real-time clock that provides a saw-tooth waveform of 1/10 sec duration.

2. Scenario of "MH-1, A Computer-Operated Mechanical Hand."

Title of Scene	Comments
<u>PART 1: PICTURES OF EQUIPMENT AND INSTALLATION</u>	
title: MH-1, A COMPUTER-OPERATED MECHANICAL HAND	hand holds board on which title is printed.
total: manipulator	over-all view of servomanipulator.
closeup: hand with sense organs	pointing out of different sense organs on the fingers.
semi-total: master arm	view of motors and potentiometers for wrist and tong control.
title: TX-0 COMPUTER	name shield of the TX-0 computer.
panorama: TX-0 installation	view of arithmetic unit, memory, in-out panel, power supply.
semi-closeup: MH-1 control unit	shows front panel of control unit.
panorama: console of TX-0	
closeup: flexowriter	the computer program as typed on the flexowriter for the first half of the film.
<u>PART 2: BUILDING OF A TOWER WITH WOODEN BLOCKS</u>	
total: search initiation	hand moves into position to start search motion for block on the table. This block is to be the base of the tower.
semi-closeup: search-motion	shows how the hand ascertains the distance from the table during the search motion.
total: search-motion	hand finishes one line of the search pattern and advances into the next line.
closeup: search-motion	hand touches the table.

total: approach of block

closeup: scanning of block

total: putting in second block

total: coffee break

semi-total: get block

semi-close: transport block

closeup: deposit block

total: search motion

semi-total: block presentation.

closeup: block deposition

total: search-motion

total: block presentation

hand approaches block and hits it.

hand passes closely behind block to ascertain its size and position, then starts to go back into search motion for second block which should eventually be placed on the first block.

assistant puts in second block while hand starts to search for it.

assistant can be seen drinking coffee while hand searches for block.

hand hits block, goes behind it and takes it.

hand lifts block and puts it down behind first block where tower is to be built. Hand starts to feel for base block, keeping the second block in its fingers.

hand hits basis block, goes behind it, lifts second block and puts it on the first block.

the hand goes back into the search motion for the third block.

the assistant puts the block the hand is searching for between the fingers of the hand. MH-1 takes it immediately because "it realizes what is going on".

third block is deposited (not quite squarely) on the two previous blocks.

Hand starts to grope for the fourth block, the awareness of the MH-1 system has been cut off for that phase.

the operator tries unsuccessfully (since there is no awareness) to present the block to the hand directly, MH-1 does not react.

closeup: scan motion

if the hand is aware that it bumps into the table it automatically adjusts its height. Without awareness it drags clumsily over the table in that case.

closeup:

MH-1 has found the block and tries to deposit it on the tower. However it bumps into the tower and the tower collapses.

PART 3: PUTTING BLOCKS INTO BOX

semi-total: discover box

hand has gone through search motion (not shown) and now hits box.

semi-close: box exploration

hand feels around box to ascertain its size and position.

total: search motion

MH-1 searches for block to be put into box.

closeup: take block

MH-1 hits block and tries to take it. A sense element is not responding properly, the hand makes little jerks back and forth.

total: repairs

the assistant enters the picture with a voltmeter and traces the trouble to a sense element that is stuck.

closeup: take block

hand is successful in picking up block in this second attempt.

semi-total: go to box

hand transports block into vicinity of box, hits box and goes behind box.

closeup: deposit block

hand lifts block and deposits it in box.

total: search-motion

hand goes back to search for more blocks. Assistant shifts box into way of hand.

semi-close: hit object

hand hits box, thinks it is block, but finds out object is bigger than block when it tries to take it.

semi-close: scanning motion

hand goes into scanning motion to determine nature of obstacle (END). (hand will then find out that it is a box and return to that part of the program which deals with boxes).

REFERENCES

- [1] Alfred Chapuis, Edmond Droz, "Les Automates. Figures Artificielles d'Hommes et d'Animaux", Editions du Griffon, Neuchatel, Switzerland, 1958.
- [2] John E. Ward, "Automatic Programming of Numerically Controlled Machine Tools", Final Report 6873-FR-3, Electronic Systems Laboratory, M.I.T., January 15, 1960.
- [3] An electrically controlled automatic milling machine for the machining of big turbine blades was finished by Escher-Wyss, Zurich, Switzerland, in 1942.
- [4] H.W. Nidenberg, "Programmed Manipulation: Noman", Technical Report, Advanced Manufacturing Engineering Service, March 15, 1960.
- [5] John W. Clark, "The Mobot, a Fully Remote, Mobile Handling Device", Technical Memorandum 627, Hughes Aircraft Company, Culver City, California, November 27, 1959.
- [6] Part catalog for AMF Master Slave Manipulator, AMF Atomics, April 1958.
- [7] R.S. Mosher, "Handyman", Product Engineering, March 30, 1959.
- [8] "Electromechanical Manipulator for Remote Handling", Automation Engineering Department, Borg Warner Corporation, 1956.
- [9] General Mills Catalog, Nuclear Equipment Department, Mechanical Division.
- [10] D.L. Reich, H.A. Ernst, "A Mechanical Hand", Quaterly Progress Report No.56, Research Laboratory of Electronics, M.I.T., January 15, 1960, p.156
- [11] A.E. Kobrinski et al., "Problems of Bioelectric Control", International Federation of Automatic Control, Moscow, 1960, Vol.2. p.1119 (Butterworth's Scientific Publications, London).
- [12] R. Tomovic, "The Human Hand as a Feedback System", International Federation of Automatic Control, op.cit., Vol.2, p.31.
- [13] R. Tomovic, "An Adaptive Artificial Hand", Unpublished Memorandum, University of California, Los Angeles, 1961.
- [14] Programming for the TX-0 Computer. Memorandum M-5001-13, 6/21/60, Department of Electrical Engineering, M.I.T.

[15] Marvin Minsky, "Steps Toward Artificial Intelligence", Proc. IRE, Vol.49, No.1, p.8, January 1961.

[16] P.L. Simmons and R.F. Simmons, "The Simulation of Cognitive Processes, An Annotated Bibliography", IRE Transaction on Electronic Computers, Vol. EC-10, No.3, p.462, September 1961.

[17] A. Newell, J.C. Shaw and H.A. Simon, "Chess Playing Programs and the Problem of Complexity", IBM J. Res. Develop., Vol.3, No.4, p.320, October 1958.

[18] H.L. Gelernter, N. Rochester, "Intelligent Behavior in Problem-Solving Machines", IBM J. Res. Develop., Vol.2, No.4, p.336, October 1958.

[19] A.L. Samuel, "Some Studies in Machine Learning Using the Game of Checkers", IBM J. Res. Develop., Vol.3, No.3, p.210, July 1959.

[20] A. Newell, J.C. Shaw and H.A. Simon, "Empirical Explorations of the Logic Theory Machine", Proc. WJCC, p.218, Los Angeles, California, February 1957.

[21] J.R. Slagle, "A Heuristic Program that Solves Symbolic Integration Problems in Freshman Calculus, Symbolic Automatic Integrator (SAINT)", Lincoln Laboratory Report 5G-0001, May 10, 1961.

[22] H.M. Teager, "MIT Computation - Present and Future", Computation Center, M.I.T., 1961.

Biographical Note

Mr. Heinrich Arnold Ernst was born in Zürich, Switzerland, on February 28, 1933. He attended the Kantonal-Gymnasium Zürich from 1945-1952 and the Swiss Federal Institute of Technology (ETH) from 1952-1957. In December 1957 he received the Diploma in Electrical Engineering. He was awarded the Silver Medal of the Swiss Federal Institute of Technology for his thesis. In February 1958 he came to the Massachusetts Institute of Technology and he received the S.M. degree in June 1959. From June 1958 until December 1961 he was a research assistant at the Research Laboratory of Electronics, with the exception of three terms during which he was teaching Introductory Circuit Theory (6.00,6.01) and Electronic Circuit Theory (6.02).

Mr. Ernst has submitted a paper based on his thesis for the Spring Joint Computer Conference, 1962. He is a member of Sigma Xi, of the Physical Society in Zurich, and a student member of the Institute of Radio Engineers.