ON THE DEMONSTRATION OF HIGH-SPEED DIGITAL COMPUTERS

By WALTER F. BAUER and JOHN W. CARR III
The Ramo-Wooldridge Corporation and the University of Michigan

The demonstration of a computer is an important event for the computer staff, whether the computer is to be sold or rented, whether it has been constructed by a non-profit organization, or whether it is simply to be used in an industrial or scientific installation. The organization selling or renting a computer is, of course, anxious to show the computer in its best light. The staff operating a computer in a research, educational, or industrial organization is anxious to impress users of the machine with its speed, flexibility, and large domain of applicability, for it is important that the computer be used to its fullest extent and advantage to justify its considerable cost. Demonstration routines are among the most important tools to gain that objective.

Because computer demonstrations are usually made to persons who do not know the inner workings of the device, their impact is generally dependent on an understanding of the psychological attitude of humans towards machines. A machine that acts like a human will, for a little while at least, impress those humans viewing it for the first time.

Persons arranging demonstrations, however, must not over-emphasize showmanship at the expense of intellectual and scientific integrity. The demonstration should therefore include examples of problems which are standard to the computer's repertoire and are at the same time interesting and intriguing.

The important characteristics of high-speed computers which are new and impressive to the observer are automaticity, speed, and versatility. Despite these impressive characteristics, the demonstration of a computer can be unimpressive to the layman and even, in cases, to the scientist. The following aspects of demonstration routines are important:

1) Complete control by the computer of its own behavior during the entire demonstration.
2) Human-like behavior by the computer in the sense that it responds to external stimuli in a logical fashion.
3) Demonstration of the computer's superiority in speed over conventional techniques.
4) Complete use of the computer's input-output facilities in as thorough and versatile a manner as possible.

Finally, the observers should not be neglected, for the most important parts of any demonstration are the humans watching it. The fifth important aspect is:

5) "Audience participation" by the onlookers in the performance of the machine's set of routines.

Any automatically sequenced computer with satisfactory storage can demonstrate 1). Most digital computers now built have adequate speed for 3); selection of specific problems can emphasize that satisfactorily. Although computers are not in general human-like, automatic sequencing plus the existence of satisfactory input-output allowing communication in a written language and graphical form provides an opportunity for demonstrating 2), 4), and 5). The availability of some form of "single-character" alphanumeric input-output, and cathode ray tubes or plotting tables for curve-plotting, are an important feature in the "audience participation" aspects.

Demonstration routines can be divided into two classes; those which demonstrate the computer in a typical practical problem; and those which solve no practical problem but are of a more recreational nature, as, for example, those in which the computer plays a game. There are many examples of the former type: a system of differential equations is integrated, a system of linear algebraic equations is solved, or a simple pay-roll is computed. The choice of one of the types in preference to the other is usually based on specific circumstances and no general statement can be made about the relative advantages of each. However, much can be said in favor of the recreational type of routine which has a great appeal in promoting general interest, for it is easily understood and the steps which the computer must perform are more comprehensible. The first type of demonstration is often only appreciated by those few who have solved the problem by manual means or on a much slower computer. We shall emphasize here demonstration routines of the second type, that is, those of a more recreational nature.

Among the time-honored demonstration routines are those involving prime numbers, factorials, and factorization; for example, those routines which print out the consecutive prime numbers, a table of roots of integers, or the factorials of integers. The latter routine is, incidentally, a good one for illustrating machine overflow, since the consecutive factorials of integers quickly exceed the capacity of the usual register of 30 to 40 binary digits.

Of a slightly different type is the factorization routine which involves audience participation. In this routine the computer "calls" for a typewriter input, and a member of the audience is invited to type an integer which the machine factors, printing out each prime factor and the number of times it occurs in the original integer.

A common characteristic of these routines is that none of them requires any special input-output methods beyond the ability to accept and print out the decimal digits. These routines are most impressive when single-character, or full typewriter keyboard, input-output is available. It is desirable, for example, to print out decimal points with a routine for finding roots of integers. Similarly, it is better if the computer can accept an integer in its usual form, rather than to have to include initial or final zeros.

An excellent demonstration which involves audience participation is a game of "tick-tack-toe" in which the machine "plays" against a human adversary. Groups intending to use such a program with typewriter input-output should be forewarned: the fact that most commercially available typewriters cannot perform an order which would roll the platen *back* a line is bothersome; either the platen must be rolled back each time manually, or the routine must be so designed as to obviate the difficulty by, say, printing out the entire array with the appropriate marks at each intermediate stage of the game. Two other routines involving games are worthy of mention: the computer can be programmed to play the games of "Nim" or "craps." In Nim, each of two players in turn may withdraw one or more objects from any one of several "piles." The player who draws the last object loses. As a matter of fact, Nim has a convenient mathematical solution for the optimum play at any stage of the game in terms of the binary digits representing the number of objects in each "pile." In the craps game routine the face of each die is selected by a random-number routine or, more precisely, a pseudo-random-number routine. The faces of the dice are printed out in symbolic fashion and the complete rules of the game can, of course, be accurately followed.

The routines for the games of Nim and tick-tack-toe certainly involve audience participation since the computer is playing against a human. Similarly, the dice game can be arranged so that bets are made and the computer keeps track of the amount of money in the bank of the two players and which one has the dice (the dice being transferred on a roll of a seven before the point is made).

If the computer is equipped with digital-to-analog converter and oscilloscope visual output, demonstration routines can be contrived which are elaborate and impressive. The display of solutions of differential equations is especially amenable to scope presentation. It is possible, for example, to solve the simple linear differential equation for a mass-spring system which corresponds to the real time oscillations of an actual physical model. The mass of the model is set in motion as the integration process begins on the computer, and the scope presentation of the solution matches with imperceptible error the motion of the model which is seen at the same time. The speed and usefulness of the computer is dramatically portrayed. To the authors' knowledge this scheme has not been tried with a digital computer. However, there seems to be no reason why the demonstration should not be possible or feasible.

Programs have been written for the Whirlwind computer at the Massachusetts Institute of Technology which make excellent use of the oscilloscope for demonstration purposes. The differential equations for a bouncing ball are integrated with the initial conditions changed each time to move the points at which the ball hits the base line; the machine "searches" to find a point where the ball drops through a hole in the base line. The movement of the ball throughout the entire process, as well as the base line with the hole, are shown on the oscilloscope face.

For computers which have the electrostatic type memory certain routines have been used which set up interesting patterns (for example, English words) on the cathode-ray tubes of the memory. Routines which test the read-around ratio of the memory can often be used. This type of demonstration routine does not, however, illustrate the important characteristics of the machine, and is only valuable in case the programmed self-checking feature of the machine is to be stressed, as it can be with the read-around ratio test.

Routines which illustrate the manner in which a machine can check itself by means of a program are important. Here, however, there should be some visible evidence of what is transpiring or what has transpired. The Rand Corporation computer has been programmed to perform arithmetic unit and memory checking routines so that upon completion, the computer prints out a statement such as "The arithmetic unit has just been checked by performing 100,000 multiplications and 100,000 divisions on random numbers without error." The self-checking feature of machines can often be demonstrated in the following way: during the operation of a routine for checking the memory a diode cluster, say, is pulled from a socket, whereupon the computer immediately prints out the number of the memory unit which has failed.

Routines which are certain to evoke interest and provide amusement are those which, during performance by the computer, provide musical tone patterns or perhaps even a melodic line when the computer is "tuned in" on a radio receiver. The SWAC computer of the National Bureau of Standards has a program dubbed "the random symphony" which produces tones of arbitrary pitch and of arbitrary length. Routines have been constructed for other computers which give an arbitrary series of musical notes, the letters for which have been typed into the computer. The routine thus allows for a tune of arbitrary choice to be played.

Recently demonstration routines were written for the MIDAC (Michigan Digital Automatic Computer) and MIDSAC (Michigan Digital Special Automatic Computer) at the University of Michigan. Their preparation was occasioned by the planning of a tour of the computing facilities of the University's Willow Run Research Center for those in attendance at the Annual Meeting of the Association for Computing Machinery held at the University June 23-25, 1954. The MIDAC computer is an improved version of the SEAC computer of the Bureau of Standards and, as auxiliary equipment, has single-character input-output with the input facilitated by a Ferranti high-speed photoelectric (paper tape) reader.* Certain ideas in the formulation of the demonstration routine for MIDAC may be of interest to other groups.

A sample of the Flexowriter output is shown in FIGURE 1 with the format changed only slightly. Certain explanatory remarks are found in parentheses. The routine included the following: a dice shooting program, integer factorization, the solution of the damped spring problem, solution of a linear system of equations, and the game of tick-tack-toe. Before each part began, the computer printed out a brief description of the routine to follow. The entire demonstration ran without any manual intervention by the operator except to signal to the computer the end of certain routines such as the number factorization. Almost all parts of the routine were prepared by students at the University enrolled in courses in digital computation.

Another demonstration routine was prepared for the MIDSAC computer. The MIDSAC is a very high speed computer built for the purpose of real-time simulation or system control. It performs a three-address add operation on signed 30 binary digit numbers in 40 microseconds, multiplication in 88 microseconds, and division in 272 microseconds, giving it a speed on the average program of about 14,000 operations per second. Included in its auxiliary equipment were analog-to-digital and digital-to-analog converters, and a magnetic drum to facilitate the input-output needed in real-time simulation or control.

To illustrate the use of MIDSAC as a simulation or control device, its designers formulated a program by which the computer would "shoot pool." The cue stick and the action of 16 pool balls was illustrated on a cathode ray tube on which was painted the outline of a pool table. In operation each of the players aligns the cue stick by entering voltages by means of potentiometers. Upon pressing a button the cue ball is "stuck" and complete action of the balls is shown as they collide or strike a cushion. Upon reaching a pocket the ball drops from sight or vanishes from the scope. Provision is made for "spotting" the cue ball when the player "scratches." The simulation of the balls is accomplished by moving the balls along their paths in small discrete steps according to their velocities. When two balls collide the new velocity of each ball is computed and used to move the balls further. Similarly the new velocity of a ball (including the effects of inelastic reflection) is computed when it reaches a cushion. The speed of each ball is decreased appropriately at each computation. This demonstration routine illustrates very dramatically the future role of the high-speed digital computer as the heart of a control system, or as a simulation device for complicated systems.

Integration of the component parts of the MIDAC routine was performed by Mrs. Jeanne Gardner, Henry Lakin, and Carl Pollmar; the MIDSAC pool-shooting routine was programmed by Edward Lewis and William Brown, all of the Willow Run Research Center, University of Michigan.

*The magnetic drum, undergoing final checkout at time of the demonstration, was not used.

FIGURE 1

MIDAC DEMONSTRATION–ACM MEETING
University of Michigan, Ann Arbor, Michigan
June 23 – 25, 1954

MIDAC was built under the auspices of the U. S. Air Force . . . . (description of MIDAC) . . . . with corresponding times for other operations.

MIDAC will now perform a sequence of problems automatically to demonstrate its speed, flexibility, and complete automaticity.

MIDAC SHOOTS CRAPS

By means of a random number routine, . . . (description of dice game) . . . . Will someone please press the start button?

```
  *      *  *
*          *
  *      *  *

         *
*
             *

  *      * * *
  *      * * *
```

win

lace your bets

.
.
.
.

The dice routine is halted and the next routine automatically begun by setting an external switch.)

.
.
.
.

MIDAC FINDS FACTORS

MIDAC will find all the prime factors . . . (description of factorization routine) . . . . when the process is complete, the MIDAC will ask for another integer.

Number please

18 (number typed in by a member of the audience).

2	1
3	2

Number please

232101 (number typed in by a member of the audience).

3	2
17	1
37	1
41	1

.
.
.

(The factorization routine is halted and the next routine automatically begun by setting an external switch).

.
.
.

MIDAC SOLVES THE DAMPED SPRING PROBLEM

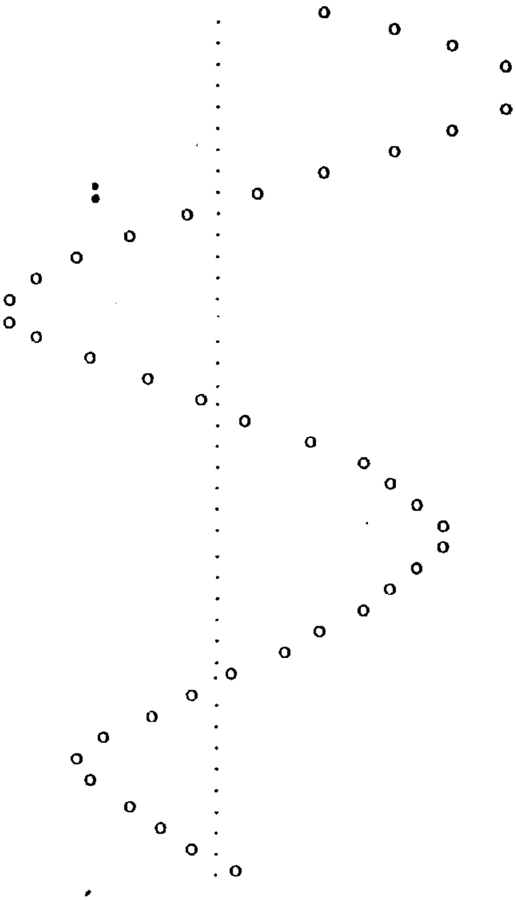
MIDAC generates in graphical form the solution to the boundary value problem of a damped spring. The differential equation being solved is

$$-mX''-cX'-KX \text{ equals } 0$$

with initial conditions

and $X(0)$ equals 0

$X'(0)$ equals a constant



MIDAC SOLVES A SET OF LINEAR EQUATIONS

Using the Gauss-Seidel Method, MIDAC will solve a tenth-order system of linear equations in 20 seconds. The coefficient matrix A and the known vector b are displayed on the side wall. The solution will be typed out on the Flexowriter.
(Eleven numbers are printed out giving the number of iterations required and the 10 answers in decimal form).

MIDAC PLAYS TICK-TACK-TOE

MIDAC will now . . . . (description of game) . . . . When the light on the left hand panel goes on, MIDAC is calling for the first move.

7 (The machine's opponent places a mark in the 7th position, numbering from left to right, and down).

——	——	——	
——	0	——	(The machine places a mark in position number 5. The first two moves are shown).
X	——	——	

8 (Next, the machine's opponent places a mark in the 8th position.)

——	——	——	
——	0	——	
X	X	0	(The machine places a mark in position number 9).

.
.
.

(Play continues in this manner with certain features. For example, if the player tries to place an "X" in a position already occupied, the computer prints out "an error, try again")

.
.
.

X	X	0
0	0	X
X	X	0

It's a tie.

You have just seen a short demonstration of . . . . . We hope that this demonstration has proved both interesting and entertaining.