Chapter 6

The Whirlwind I computer[1]

R. R. Everett

Project Whirlwind is a high-speed computer activity sponsored at the Digital Computer Laboratory, formerly a part of the Servomechanisms Laboratory, of the Massachusetts Institute of Technology (M.I.T.) by the Office of Naval Research (O.N.R.) and the United States Air Force. The project began in 1945 with the assignment of building a high-quality real-time aircraft simulator. Historically, the project has always been primarily interested in the fields of real-time simulation and control; but since about the beginning of 1947 most of its efforts have been devoted to the design and construction of the digital computer known as Whirlwind I (WWI). This computer has been in operation for about 1 year and an increasing proportion of project effort now is going into application studies.

Applications for digital computers are found in many branches of science, engineering, and business. Although any modern general-purpose digital computer can be applied to all these fields, a machine is generally designed to be most suited to some particular area. Whirlwind I was designed for use in control and simulation work such as air traffic control, industrial process control, and aircraft simulation. This does not mean that Whirlwind will not be used on applications other than control. About one-half the available computing time for the next year will be assigned to engineering and scientific calculation including research in such uses supported by the O.N.R. through the M.I.T. Committee on Machine Methods for Computation.

These control and simulation problems result in a specialized emphasis on computer design.

Short register length

WWI has 16 binary digits and the control problems are usually very simple mathematically. Furthermore, the computer is almost always part of a feedback rather than an open-ended system. Consequently, roundoff errors are seldom troublesome and the register length can be shortened to something comparable to the sensitivity of the physical quantities involved, perhaps five decimal places or less.

WWI has a register length of 16 binary digits including sign or about four and one-half decimals. The register length was chosen as the minimum that would provide a usable single-address order, in this case five binary digits for instruction and 11 binary digits for address. In a future machine we would probably increase this register length to 20 or 24 binary digits to get additional order flexibility; the increased numerical precision is less important.

For scientific and engineering calculation, greater than 16-digit precision is often required. There is available a set of multiple-length and floating point subroutines which make the use of greater precision very easy. It is true that these subroutines are slow, bringing effective machine speed down to about that obtained by acoustic memory machines. It is much more efficient occasionally to waste computing time this way than continuously to waste a large part of the storage and computing equipment of the machine by providing an unnecessarily long register.

High operating speed

WWI performs 20,000 single-address operations per second. Control and simulation problems require very high speeds. The necessary calculations must be carried out in real time; the more complex the controlled system is, the faster the computer must be. There is no practical upper limit to the computing speed that could be used if available.

Where the problems are large enough, and these problems are, one high-speed machine is much better than two simpler machines of half the speed. Communication between machines presents many of the same problems that communication between human beings presents.

Great effort was put into WWI to obtain high speed. The target speed was 50,000 single-address operations per second, and all parts of the machine except storage meet this requirement. The actual WWI present operating speed of 20,000 single-address operations per second is on the lower edge of the desired speed range.

Large internal storage

WWI now has 1,280 registers. A large amount of high-speed internal storage is needed since it is not in general possible to use slow auxiliary storage because of the time factor. In many cases a magnetic drum can be useful since its access time is short com-

[1]*AIEE-IRE Conf.*, 70–74 (1951)

pared to the response times of real systems. Even with a drum there is considerable loss of computing and programming efficiency due to shuffling information back and forth between drum and computer.

WWI is designed for 2,048 registers of storage. Until recently there has been available only about 300 registers. This number, while small, has been adequate for much useful work. Very recently a second bank of new-model storage tubes has been added. These new tubes operate at 1,024 spots per tube bringing the total WWI storage to 1,280 registers. These tubes have been in the computer and under test for 2 months and in active use for about 2 weeks. In the next few months the tubes in the first bank will be replaced by new-model storage tubes bringing the total storage to 2,048. This number is on the lower end of what the project considers desirable. What the computer business needs, has needed, and will probably always need is a bigger, better, and faster storage device.

Extreme reliability

In a system where much valuable property and perhaps many human lives are dependent on the proper operation of the computing equipment, failures must be very rare. Furthermore, checking alone, however complete, is inadequate. It is not enough merely to know that the equipment has made an error. It is very unlikely that a man, presumably not too well suited to the work during normal conditions, can handle the situation in an emergency. Multiple machines with majority rule seem to be the best answer. Self-correcting machines are a possibility but appear to be too complicated to compete, especially as they provide no standby protection.

The characteristics of the Whirlwind I computer may be recapitulated as follows:

Register length	16 binary digits, parallel
Speed	20,000 single-address operations per second
Storage capacity	Originally 256 registers Recently 320 registers Presently 1,280 registers Target 2,048 registers
Order type	Single-address, one order per word
Numbers	Fixed point, 9's complement
Basic pulse repetition frequency	1 megacycle 2 megacycles (arithmetic element only)
Tube count	5,000, mostly single pentodes
Crystal count	11,000

There are 32 possible operations, of which about 27 are assigned. They are of the usual types: addition, subtraction, multiplication, division, shifting by an arbitrary number of columns, transfer of all or parts of words, subprogram, and conditional subprogram. There are terminal equipment control orders and there are some special orders for facilitating double-length and floating-point operations.

One way to increase the effective speed of a machine is to provide built-in facilities for operations that occur frequently in the problems of interest. An example is an automatic co-ordinate transformation order. The addition of such facilities does not affect the general-purpose nature of the machine. The machine retains its old flexibility but becomes faster and more suited to a certain class of problems.

From March 14, 1951, at which time we began to keep detailed records, until November 22, 1951 a total of 950 hours of computer time were scheduled for applications use. The machine has been running on two shifts or a total of about 3,000 hours during this interval. The two-thirds time not used for applications has been used for machine improvement, adding equipment, and preventive maintenance.

Of the 950 hours available, 500 have been used by the scientific and engineering calculation group, the rest for control studies. The limited storage available until recently has been admittedly a serious handicap to the scientific and engineering applications people. There has not been room in storage for the lengthy subroutines necessary for convenient use of the machine. The largest part of their time has been spent in training, in setting up procedures, and in preparing a library of subroutines.

A partial list of the actual problems carried out by the group includes:

1. An industrial production problem for the Harvard Economics School
2. Magnetic flux density study for our magnetic storage work
3. Oil reservoir depletion studies
4. Ultra-high frequency television channel allocation investigation for Dumont
5. Optical constants of thin metal films
6. Computation of autocorrelation coefficients
7. Tape generation for a digitally-controlled milling machine

The scientific and engineering applications time on Whirlwind I has been organized in a manner patterned after that originated by Dr. Wilkes at EDSAC. The group of programmers and mathematicians assigned to WWI assist users in setting up their own problems. Small problems requiring only a few seconds or minutes of computer time are encouraged. Applications time is assigned in I-hour pieces two or three times a day. No program debugging is allowed on the machine. Program errors are deduced by the programmer from printed lists of results, storage contents, or order sequences as previously requested from the machine operator. The programmer then corrects his program which is rerun for him within a day or perhaps within a few hours.

Every effort is made to reduce the time-consuming job of printing tabulated results. In many cases a user desires large amounts of tabulated data only because he doesn't really know what answers he wants and so asks for everything. Such users are encouraged to ask only for pertinent results in the form of numbers or curves plotted by the machine on a cathode-ray tube and automatically photographed. If these results prove inadequate or the user gets a better idea of his needs, he is allowed to rerun his program, again asking only for what appear to be significant results. Figure I shows a sample curve plotted by the computing machine showing calibrated axes and decimal intercepts.

Fig. 1. Sample computer output.

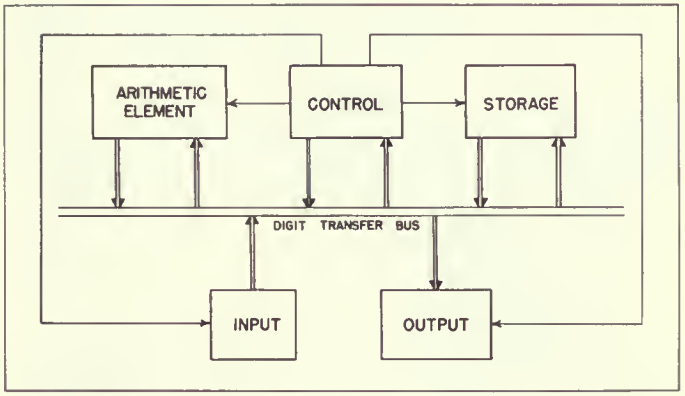

Fig. 2. Simplified computer block diagram.

WWI system layout

Figure 2 shows the major parts of any computer such as WWI. The major elements of the computer communicate with each other via a central bus system.

WWI is basically a simple, straightforward, standard machine of the all-parallel type. Unfortunately, the simple concept often becomes complicated in execution, and this is true here. WW's control has been complicated by the decision to keep it completely flexible, the arithmetic element by the need for high speed, the storage by the use of electrostatic storage tubes, the terminal equipment by the diversity of input and output media needed.

Control

The WW control is divided into several parts, as shown in Fig. 3.

Central control

The central control of the machine is the master source of control pulses. When necessary the central control allows one of the other controls to function. In general there is no overlapping of control operation; except for terminal equipment control, only one of the controls is in operation at any one time.

Storage control

Storage control generates the sequence of pulses and gates that operate the storage tubes. Central control instructs the storage control either to read or to write.

Arithmetic control

Arithmetic control carries out the details of the more complex arithmetic operations such as multiplication and division. The

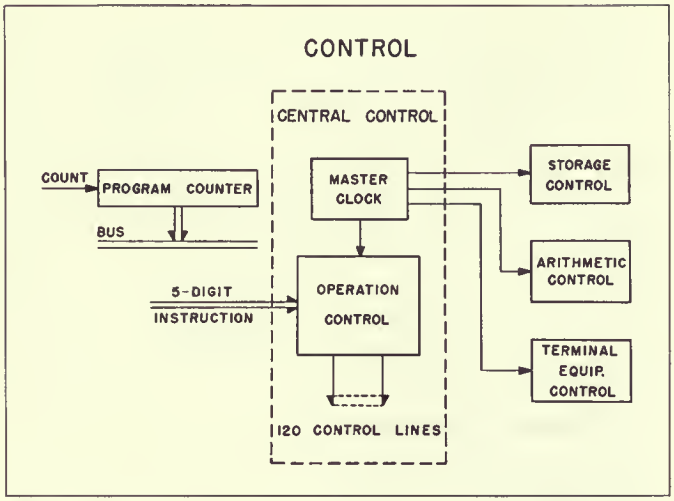

Fig. 3. Control.

setup of these operations plus the complete controlling of the simpler operations such as addition are carried out by central control.

Terminal equipment control

Terminal equipment control generates the necessary control pulses, delay times, and interlocks for the various terminal equipment units.

Program counter

The program counter which keeps track of the address of the next order to be carried out is considered as part of control. This is an 11-binary counter with provision for reading to the bus.

Most of the functions of these subsidiary controls could be combined with the central control. The major reason they are not is that they were designed at different times. The arithmetic element and its control came first, followed by central control. At the time central control was designed, the necessary characteristics of storage control were unknown. In fact, the machine was designed so that any parallel high-speed storage could be used. The form of terminal equipment control was also unknown at this time. Since flexibility was a prime specification, it was felt preferable to build separate flexible controls for the various parts of the computer than to try to combine all the needed flexibility in one central control.

In a new machine we would attempt to combine control functions where possible, hoping to have enough prior knowledge about component needs to eliminate subsidiary controls completely. We would still insist on a large degree of control flexibility.

Master clock

The master clock consists of an oscillator, pulse shaper and divider that generate 1- and 2-megacycle clock pulses, and a clock pulse control that distributes these clock pulses to the various controls in the machine. It is this unit that determines which of the subsidiary controls actually is controlling the machine. This unit also stops and starts the machine and provides for push-button operation.

Operation control

The operation control, see Fig. 4, was designed for maximum flexibility and minimum number of operation digits, and, consequently, minimum register length. It is of the completely decoding type.

The operation switch is a 32-position crystal matrix switch that receives the 5-bit instruction from the bus and in turn selects one of 32 output lines corresponding to the 32 built-in operations.

There are 120 gate tubes on the output of the operation control. Pulses on the 120 output lines go to the gate drivers, pulse drivers, and control flip-flops all over the machine; 120 is a generous number. The suppressors of these gate tubes are connected to vertical wires that cross the 32 output lines from the operation switch. Crystals are inserted at the desired junctions to turn on those gate tubes that are to be used for any operation.

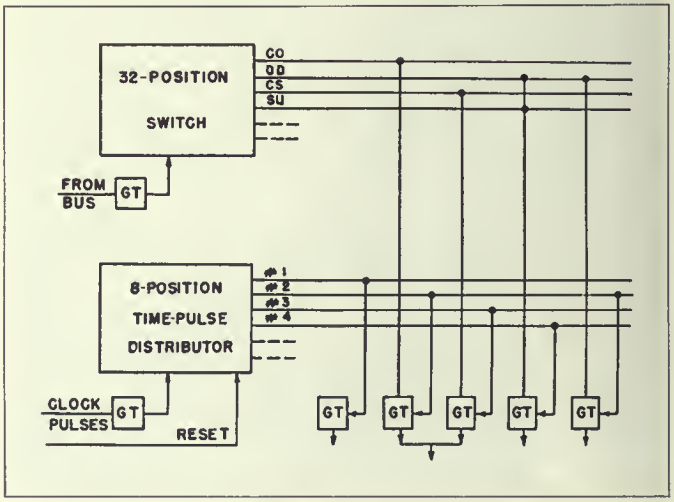

Fig. 4. Operation control.

The time pulse distributor consists of an 8-position switch driven from a three binary-digit counter. Clock pulses at the input are distributed in sequence on the eight output lines. The control grids of the output gate tubes are connected to these timing lines. The output of the operation control is thus 120 control lines on each of which can appear a sequence of pulses for any combination of orders at any combination of times.

Central control

The Central Control of the machine is shown in Fig. 5. The control switch is in the foreground with the operation matrix to the right.

Electrostatic storage

The electrostatic storage shown in Fig. 6 consists of two banks of 16 storage tubes each. There is a pair of 32-position decoders set up by address digits read in from the bus. There is a storage control that generates the sequence of pulses needed to operate the gate generators, et cetera. A radio frequency pulser generates a high power 10-megacycle pulse for readout.

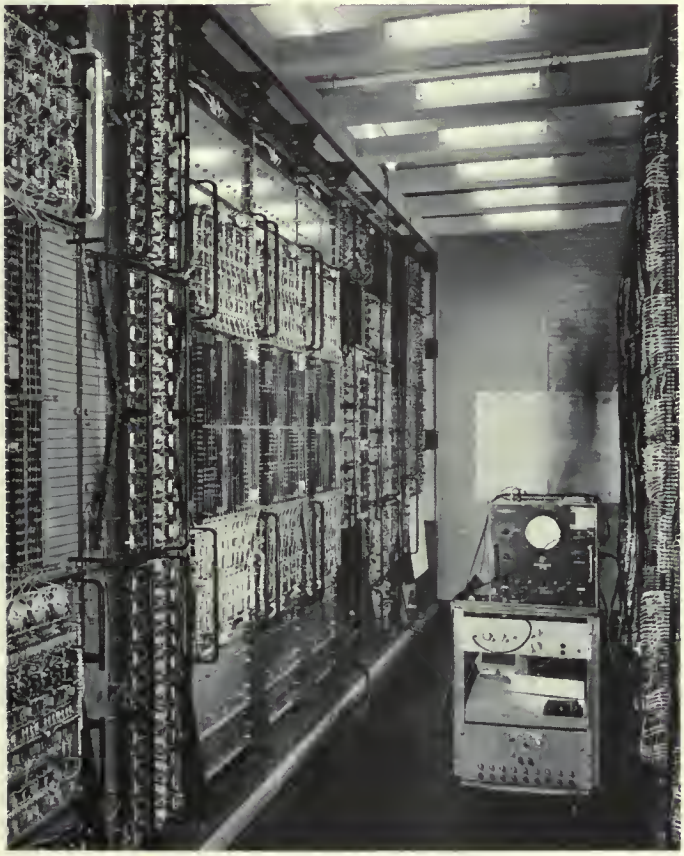

Fig. 5. View of central control.

Fig. 6. View of electrostatic storage.

Each digit column contains, besides the storage tubes, write plus and write minus gate generators and a signal plate gate generator for each tube. Ten-megacycle grid pulses are used for readout in order to get the required discrimination between the fractional volt readout pulses and the 100-volt signal plate gates. For each storage tube there is a 10-megacycle amplifier, phase-sensitive detector and gate tube, feeding into the program register. The program register is used for communicating with the storage tubes. Information read out of the tubes appears in the program register. Information to be written into the tubes must be placed in the program register.

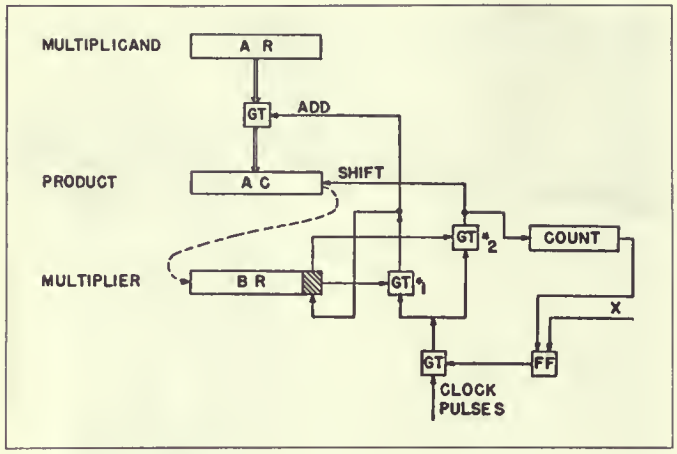

Fig. 7. Arithmetic element.

Arithmetic element

The arithmetic element, see Fig. 7, consists of three registers, a counter, and a control.

The first register is an accumulator (*AC*) which actually consists of a partial-sum or adding register and a carry register. The accumulator holds the product during multiplication.

The second or *A*-register holds the multiplicand during multiplication. All numbers entering the arithmetic element do so through *AR*.

The third or *B*-register holds the multiplier during multiplication. The accumulator and *B*-register shift right or left. A high-speed carry is provided for addition. Subtraction is by 9's complement and end-around-carry. Multiplication is by successive additions, division by successive subtractions, and shift orders provide for shifting right or left by an arbitrary number of steps, with or without roundoff.

The arithmetic element is straightforward except for a few special orders and the high speed at which it operates. Addition takes 3 microseconds complete with carry; multiplication, 16 microseconds average including sign correction.

In Fig. 8 are shown several digits of the arithmetic element. The large panels are accumulator digits. Above the accumulator is the *B*-register, below it the *A*-register.

Test control

Test control, shown in Fig. 9, is used at present both for operating and for trouble shooting the computer. The control includes:

1 Power supply control and meters.

2 Neon indicators for all flip-flops in the machine.

3 Switches for setting up special conditions.

4 Manual intervention switches.

5 Oscilloscopes for viewing wave forms. A probe and amplifier system allows viewing any wave form in the computer on one scope at test control.

6 Test equipment to provide synchronizing, stop, or delay pulses at any step of any order of a program, allowing viewing wave forms on the fly anywhere in the machine.

An important part of the test facilities is the test storage, a group of 32 toggle-switch registers plus five flip-flop registers that can be inserted in place of any five of the toggle-switch registers. This storage has proved invaluable not only for testing control and

Fig. 8. View of arithmetic element.

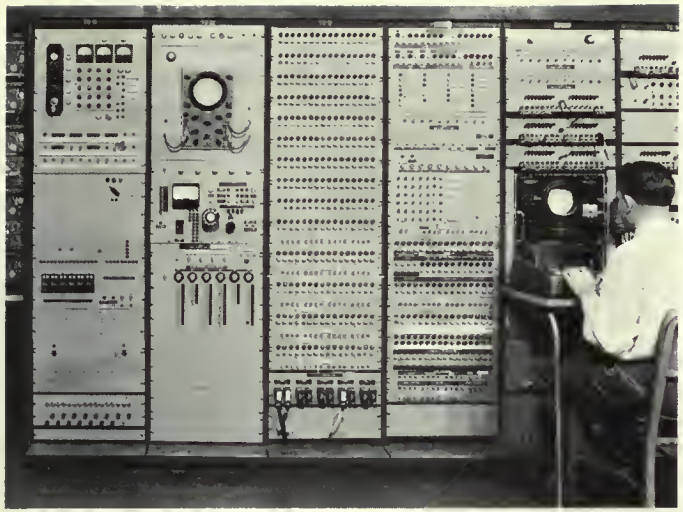

Fig. 9. View of test control.

arithmetic element before electrostatic storage was available but also for testing electrostatic storage itself. When not in use for test purposes test storage earns its keep as part of the terminal equipment system. The toggle-switches hold a standard read-in program; the flip-flop registers are used as in-out registers for special purposes.

Checking

Logical checking facilities built into WWI are rather inconsistent. A complete bus transfer checking system has been provided, duplicate checking of some terminal equipment is permitted, but little else is thoroughly checked. We felt that it was worthwhile to thoroughly check some substantial portion of the machine. This portion would then serve as a prototype for studying the tube circuitry used throughout the machine. We did not feel it was worthwhile to check all the machine, a procedure that requires a great deal of added equipment and logical complexity plus a substantial loss in computing speed.

Operating experience has shown us that it is not worthwhile to provide detailed logical checking of a machine. In a new machine we would leave out the transfer checking. The amount of information and security given by the detailed checking system is not enough to warrant the expense of building and maintaining it.

This decision is based on the expectation that a computing machine should operate 95 per cent of total time or better and that the average time between random failures should be of the order of 5 to 10 hours or approximately 10^9 operations.

In our opinion the way to achieve the extremely high reliability needed in some real-time control problems is to provide three or more identical but distinct machines, thus obtaining error correction as well as detection, plus such features as standby, safety, and damage control. Even so the failure probability of each machine must be kept low by proper design, marginal checking, and preventive maintenance.

Extremely high reliability means a reliability far beyond that achieved in existing machines and not conveniently represented as a per cent. Consider a system consisting of three machines, each operable 98 per cent of the time and each averaging 10 hours between random errors.

One machine will be out of operation $\frac{1}{2}$ hour per day.

Two machines will be out of operation $\frac{1}{4}$ hour per month.

All three machines will be out of operation 4 minutes per year. Furthermore undetected random errors might occur on the average of once a year. Such reliability is needed in some systems.

Our decision to omit detailed checking does not extend to checking devices intended to detect programming errors. Devices to check for overflow from the arithmetic element or for nonexistent order configurations are necessary. Programmers make many mistakes. Techniques for dealing with programming errors are very important and need future development.

Terminal equipment

At the present time, Whirlwind is using the following terminal equipment:

1. A photoelectric paper tape reader
2. Mechanical paper tape readers and punches
3. Mechanical typewriters
4. Oscilloscope displays 5 to 16 inches in diameter with phosphors of various persistencies including a computer-controlled scope camera
5. Inputs from various analogue equipments needed for control studies
6. Outputs to analogue equipment

To be added during the next year:

1. Magnetic Tape (units by Raytheon). One such unit is now being integrated with machine.
2. Magnetic drums (units by Engineering Research Associates, Inc.).
3. Many more analogue inputs and outputs.

This great complexity of terminal equipment requires a flexible switching system. There is a single in-out register (IOR) through which most of the data passes.

There is a switch which is set up by an order to select the desired piece of terminal equipment. Other orders put data into IOR or remove data from IOR. The in-out control provides the necessary control pulses to go with each type of equipment. In general the computer continues to run during terminal equipment wait times; suitable interlocks are provided to prevent trouble. This complete equipment has not yet been fully installed.

References

Whirlwind: EverR51; SerrR62; TaylN51.
EdSAC: SamuA57; WilkM56.

APPENDIX 1 WHIRLWIND I INSTRUCTION CODE[1]

The complete WWI instruction code is given below in tabular form. The notations used in the table, together with their meanings, are as follows:

---- = remains unchanged
IOR = In-Out Register
PC = Program Counter
AC = Accumulator, AC(i) = digit i of AC, $0 \leq i \leq 15$
BR = B-Register, BR(i) = digit i of BR, $0 \leq i \leq 15$
AR = A-Register, AR(i) = digit i of AR, $0 \leq i \leq 15$
$\overline{p}$ = round-off from BR, if $BR(0)=1, p=2^{-15}$; if $BR(0)=0, \overline{p}=0$
Cbd = Core memory

n = address of a storage register $0 \leq n \leq 2047$
n = positive integer $0 \leq n \leq 511$ mod 32
SAM = Special Add Memory
C( ) = original contents of register ( )
C(i) = original contents of digit i of register ( )
F( ) = fractional part of the quantity in [ ]
I ( ) = integral part of the quantity in [ ]
--→ = becomes

x(i) = digit i of register x, $0 \leq i \leq 15$
x(k-l) = digits k thru l of register x, $0 \leq k \neq l \leq 15$
AC◦BR = the composite 32 digit register (including sign) composed of the AC and BR taken in that order
(AC◦BR)(i) = digit i of (AC◦BR), $0 \leq i \leq 31$
⊕ = Boolean "exclusive or" operator
⊗ = Boolean "and" operator

Instruction	Function	Binary Code	Dec. Equiv.	Final Contents of AC	BR*	AR	SAM	n	Comments	Time
ss pqr	select output unit or stop the computer	00000	0	----	----	----	----	----	The unit selected is designated by the digits pqr, and is started. ss 0 will stop the computer. ss 1 is a "conditional" stop. Program always possible if ss selects Magnetic or Photo Electric Tape Reader without necessary rd's	31 μsec
		00001	1	----	----	----	----	----	Illegal instruction.	
bl n	block transfer in (n words to Cbd)	00010	2	n ◦ n	----	n	----	first word of block	$-n \cdot 2^{-15}$ must be stored in AC at the time the computer executes the bl. If n=0 one word will be read but not transferred. It is simply discarded	For drum 8 msec average and 16 msec. max. for first word. 12 μsec for ea. additional word
rd n	read	00011	3	C(IOR)	----	C(IOR)	----	----	After word is transferred from IOR to AC, the IOR is cleared. The address of rd has no significance	4 μsec
bs n	block transfer out (n words from Cbd)	00100	4	n ◦ n	----	n	----	----	$-n \cdot 2^{-15}$ must be stored in AC. If n=0, no recording will take place	same as for bl.
rc n	record	00101	5	----	----	----	----	----	If rc is used as a display instruction, the IOR is cleared	24 μsec
ad n	sum digits	00110	6	$C_i(AC) \oplus C_i(x)$ ----→ AC(i) i = 0, . . .15	----	C[x]	clear	----	Adds digits without carry. ⊕ | 0 1 ; 0 | 0 1 ; 1 | 1 0	24 μsec
ts n	transfer to storage	01000	8	----	----	----	----	C(AC)		24 μsec
td n	transfer digits	01001	9	----	----	----	----	x(0-4) unchanged x(5-15) = AC(5-15)		32 μsec
ta n	transfer address	01010	10	----	----	----	----	x(0-4) unchanged x(5-15) = AR(5-15)	ta normally follows an sp, cp, sf, or ao	32 μsec
ch n	check	01011	11	----	----	----	----	----	Computer stops on "check-register alarm" if C(AC)≠C(x). (Note that +0 ≠ -0)	24 μsec
ab n	add BR	01100	12	C(BR)+C(x)	----	C(x)	clear	C(BR) + C(x)	possible Arith. Overflow Alarm if $\lvert C(x)+C(BR)\rvert \geq 1$	32 μsec
cs n	exchange	01101	13	C(x)	----	C(x)	----	C(AC)	cs 0 will clear AC without clearing BR	32 μsec
cp n	conditional transfer control (conditional program)	01110	14	----	----	y ◦ i (digits 6-15)	----	----	If C(AC) ≥ +0 proceed to next instruction. If C(AC) = -0, execute sp n, y is location of cp n	16 μsec.
sp n	transfer control (sub-program)	01111	15	----	----	y◦i (digits 5-15)	----	----	Take next instruction from register n. PC --→ n. y is location of sp n	16 μsec.
ca n	clear and add	10000	16	C(x)+C(SAM)2^{-15}	clear	C(x)	clear	----	possible Arith. Overflow Alarm if $\lvert C(x)+C(SAM)2^{-15}\rvert = 1$	24 μsec
cs n	clear and subtract	10001	17	-C(x)+C(SAM)2^{-15}	clear	C(x)	clear	----	possible Arith. Overflow Alarm if $\lvert -C(x)+C(SAM)2^{-15}\rvert = 1$	24 μsec
ad n	add	10010	18	C(AC)+C(x)	----	C(x)	clear	----	possible Arith. Overflow Alarm if $\lvert C(AC)+C(x)\rvert \geq 1$	24 μsec
su n	subtract	10011	19	C(AC)-C(x)	----	C(x)	clear	----	possible Arith. Overflow Alarm if $\lvert C(AC)-C(x)\rvert \geq 1$	24 μsec
cau n	clear and add magnitude	10100	20	$\lvert C(x)+C(SAM)2^{-15}\rvert$	clear	$\lvert C(x)\rvert$	clear	----	possible Arith. Overflow Alarm if $\lvert C(x)+C(SAM)2^{-15}\rvert = 1$	24 μsec
as n	special add	10101	21	$F\{C(AC)+C(x)\}$	----	C(x)	$I\{C(AC)+C(x)\}$, if ≠ or = 0	----	Sign of SAM determined by sign of overflow. Previous contents of SAM cleared without alarm	24 μsec
ao n	add one	10110	22	C(x)+$(1n2^{-15})$	----	C(x)	clear	C(x) + $n2^{-15}$	possible Arith. Overflow Alarm if $C(x=1)n2^{-15}=1$	32 μsec
dm n	difference of magnitudes	10111	23	$\lvert C(AC)\rvert - \lvert C(x)\rvert$	C(AC)	$\lvert C(x)\rvert$	clear	----	If $\lvert C(AC)\rvert = \lvert C(x)\rvert$ result is +0	24 μsec.
mr n	multiply and round-off	11000	24	C(AC)·C(x) + $\overline{p}$	clear	$\lvert C(x)\rvert$	clear	----	Sign of AC is determined by sign of product	36-41 μsec
mh n	multiply and hold	11001	25	C(AC)·C(x) (digits 1-15)	|C(AC) C(x)| (digits 15-30) BR(0)-BR(14) BR (0≠ n	$\lvert C(x)\rvert$	clear	----	Sign classified same as for mr. Result in (AC◦BR) is a Booths register product.	Same as for mr
dv n	divide	11010	26	p ◦ (sign of quotient)	|C(AC)| ÷ |C(x)| 0 digits	$\lvert C(x)\rvert$	clear	----	Divide Error Alarm if $\lvert C(AC)\rvert > \lvert C(x)\rvert$. Arith. Overflow Alarm if $\lvert C(AC)\rvert = \lvert C(x)\rvert \neq 0$ and dv n is followed by sir 15. If C(AC)=C(x)=0, the quotient is 0	73 μsec.
slr n	shift left and round-off (n places)	11011 0	27	$F\{C(AC \circ BR)2^n\} + \overline{p}$ (n taken mod 32)	either	----	clear	----	possible Arith. Overflow Alarm if $F\{C(AC \circ BR)2^n\} + p = 1$. The sign digit is not shifted. Digits shifted out of AC 1 are lost. Negative numbers are complemented in AC before shifting and after rounding off. Digit b of slr n must be zero.	16-41 μsec.
slh n	shift left and hold (n places)	11011 1	27	$F\{C(AC \circ BR)2^n\}$ (n taken mod 32)	$F\{\lvert C(AC \circ BR)2^n\rvert\}$ (digits 15-(31-n)) BR(j) --→ 0 j = n-1, . . .15 n≥0	----	clear	----	The sign digit is not shifted. Negative numbers are complemented in AC before and after the shift. Digit b of slh n must be a one.	Same as for slr
srr n	shift right and round-off	11100 0	28	$C(AC)2^{-n} + \overline{p}$ (n taken mod 32)	clear	----	clear	----	possible Arith. Overflow Alarm if n ≠ 0 (this instruction simply causes roundoff and clears BR). The sign digit is not shifted. Negative numbers are complemented in AC before shifting and after rounding off. Digit b of srr n must be a zero.	Same as for slr
srh n	shift right and hold	11100 1	28	$C(AC)2^{-n}$ (n taken mod 32)	$\lvert C(AC \circ BR)2^{-n}\rvert$ (digits 15-31)	----	clear	----	The sign digit is not shifted. Negative numbers are complemented in AC before and after the shift. Digit b of srh n must be a one.	Same as for slr
sf n	scale factor	11101	29	$C(AC \circ BR)2^n$ (digits 1-15)	$C(AC \circ BR)2^n$ (digits 15-(31-n)) BR(j) --→ 0 j = 15-n, . . .15 n≥0	$n \cdot 2^{-15}$	clear	$n \cdot 2^{-15}$ n(5-15)	X(0-4) unaffected, n is such that $\frac{1}{2} \leq \lvert C(AC \circ BR) 2^{n}\rvert < 1$; if C(AC◦BR) = 0, then n = 31. Negative numbers are complemented in AC before and after the multiplication.	33-81 μsec.
clc n	cycle left and clear	11110 0	30	$(AC \circ BR)(i+n)_{32}$ ---→ AC(i) i = 0, . . .15	clear	----	----	----	Sign digit is shifted with all other digits. Digits shifted left out of AC 0 are carried around into BR 15. No roundoff. No complementing of AC either before or after the shift. Digit b of clc n must be a zero. Instruction cla 0 clears BR without affecting AC	Same as for slr
clh n	cycle left and hold	11110 1	30	$(AC \circ BR)(i+n)_{32}$ ---→ AC(i) i = 0, . . .15	$(AC \circ BR)(i+15)_{32}$ ---→ BR(i) i = 0, . . .15	----	----	----	Sign digit is shifted with all other digits. Digits shifted left out of AC 0 are carried around into BR 15. No complementing of AC either before or after the shift. Digit b of clh n must be a one. Instruction clh 0 does nothing.	Same as for slr
tmd n	multiply digits	11111	31	$C_i(AC) \otimes C_i(x)$ ----→ AC(i) i = 0, . . .15	---	-(Final AC)	---	---	Multiplies digits with no carry. ⊗ | 0 1 ; 0 | 0 0 ; 1 | 0 1	24 μsec

Note: In operations mr, mh, dv, sir, srr, srh, sf, the C(BR) is assumed to be the magnitude of the least significant part of AC + BR. For the ab and dm operations, the BR is treated just as any storage register.

[1]Whirlwind I Instruction Code came from "Comprehensive System Manual, A System of Automatic Coding for the Whirlwind Computer," published by Massachusetts Institute of Technology, Digital Computer Laboratory, Cambridge, Mass.