IBM electronics

applied science department

technical newsletter no. 2

IBM

APPLIED SCIENCE DEPARTMENT

Technical Newsletter No. 2

February, 1951

CONTENTS

Part 1

Part 2

The Calculation of Variance, Covariance, and Averages

Bruce G. Oldfield

U.S. Naval Ordnance Test Station

Using the following procedure the 604 electronic calculator will compute and punch the covariance, the average of both of the variables concerned, and n which is the number of terms in the summation, all in a single run. By introducing a single variable x, into both the x_i and the x_j read in fields, the control panel will compute the variance, the average, and n.

The variables x_i and x_j can have a maximum of five digits. The control panel is set up for a five-digit number having four decimal places. The variables should always be entered into the highest order columns of the read in field in order to have the maximum number of figures in the covariance and averages. The number of terms in the summation may go up to 999 in all cases and can go higher in most cases, depending on the numbers involved.

The speed of computing two averages and a covariance will be one hundred terms per minute minus three master cards per complete summation. The third master card is used to eliminate a probable unfinished program.

(Stored) FS 1	(Read) FS 2	(Read) FS 3 and 4	(Stored) GS 1 and 2	(Stored) GS 3 and 4	(Stored) Ctr.
n	x_i	x_j	$\Sigma x_{i-1} x_{j-1}$	Σx_{i-1}	Σx_{j-1}
xxx	x.xxxx	x.xxxx	xxxxx.xxx	xxxx.xxxx	xxxx.xxxx

---

(1) FS 3 and 4(x_j) → ctr.

(2) FS 3 and 4 → MQ

(3) ctr. R and R → FS 3 and 4 (Σx_j)

(4) FS 2(x_i) RO - Multiply

(5) $\frac{1}{2}$ adj. → ctr. into 5th

(6) GS 1 and 2 ($\Sigma x_{i-1}x_{j-1}$) → ctr. into 6th

(7) ctr. R and R → GS 1 and 2 out of 6th ($\Sigma x_i x_j$)

(8) GS 3 and 4 (Σx_{i-1}) → ctr.

(9) FS 2 (x_i) → ctr.

(10) ctr. R and R → GS 3 and 4 (Σx_i)

(11) FS 1 (n-1) → ctr.

(12) 1 → ctr. (emitted)

(13) ctr. R and R → FS 1 (n)

(14) FS 3 and 4 (Σx_j) → ctr.

Normally active through Calculate Selector #1

---

(15) FS 1 RO - Divide

(16) ctr. R and R → FS 3 and 4

(17) FS 3 and 4 → ctr. into 6th

(18) MQ → FS 3 and 4

(19) FS 1 RO - Divide

(20) ctr. R and R

(21) MQ → ctr.

(22) FS 3 and 4 → ctr. into 6th

Normally suppressed through Calculate Selector # 2

(23) ctr. → MQ out of 3rd

(24) MQ → ctr. neg. into 3rd

(25) ctr. R and R → FS 2 out of 6th

(26) GS 3 and 4 RO - Multiply

(27) ctr. R and R → FS 3 and 4 out of 6th

(28) FS 3 and 4 → ctr. neg. into 3rd

(29) MQ → FS 3 and 4

(30) FS 2 → MQ

(31) GS 3 and 4 RO - Multiply neg.

(32) GS 1 and 2 → ctr. into 6th

(33) Ctr. R and R → GS 1 and 2 out of 6th $\left(\Sigma x_i x_j - \frac{\Sigma x_i x_j}{n} = \text{xxxxx.xxx}\right)$

(34) FS 3 and 4 → ctr.

(35) FS 2 → ctr. into 4th

(36) ctr. R and R → FS 3 and 4 (Σx_j=x.xxxxxxx)

Normally suppressed through Calculate Selector #2

FS 1	FS 3 and 4	GS 1 and 2	GS 3 and 4
n	$\bar{x}_j$	$\Sigma x_i x_j - \frac{\Sigma x_i \Sigma x_j}{n}$	Σx_i

(37) GS 1 and 2 → ctr.

(38) FS 1 → FS 2 into 2nd

(39) FS 2 RO - Divide

(40) ctr. R and R → GS 1 and 2

(41) GS 1 and 2 → ctr. into 6th

(42) MQ → GS 1 and 2 into 4th

(43) FS 2 RO - Divide

(44) MQ → ctr. into 4th

(45) GS 1 and 2 → ctr. into 6th

(46) $\frac{1}{2}$ adj. → ctr. into 5th

Normally suppressed through Calculate Selector # 3

(47) ctr. R and R → GS 1 and 2 out of 6th $\left(\frac{\Sigma x_i x_j - \frac{\Sigma x_i x_j}{n}}{n} = xxx.xxxxx\right)$

(48) GS 3 and 4 → ctr.

(49) FS 1 RO - Divide

(50) ctr. R and R → GS 3 and 4

(51) GS 3 and 4 → ctr into 6th

(52) MQ → GS 3 and 4

(53) FS 1 RO - Divide

(54) MQ → ctr.

Normally suppressed through Calculate Selector # 3

(55) GS 3 and 4 → ctr into 6th

(56) FS 3 and 4 → GS 3 and 4 $\left(\frac{\Sigma x_j}{n}\right)$

(57) ctr. R and R → FS 3 and 4 out of 4th $\left(\frac{\Sigma x_i}{n}\right)$

(58) FS 3 and 4 → ctr. into 4th, Balance test

(59) FS 1 → ctr. pos. (suppress on - Balance)

(60) FS 1 → ctr. neg. (suppress on + Balance)

$\underbrace{x.xxxxxxx}_{\frac{\Sigma x_i}{n}}$ $\underbrace{xxx}_{n}$

Punch Out GS 1 and 2	Punch Out GS 3 and 4	Punch Out Ctr.
xxx. xxxxx	x. xxxxxxx	x. xxxxxxx/xxx
ℓ_{ix}	$\bar{x}_j$	$\bar{x}_i$ n

521 Wiring

The first card is a master card with an x in column 1 to pick up Pilot Selector number one, which reads out and resets the counter; reads in FS 1,and GS 1,2, 3, 4.

The next to last card is a master card with an x in columns 2 and 3. An x in column 2 picks up co-selector # 1 immediately; an x in column 3 picks up co-selector # 2 and pilot selector number 2, which on the next card cycle stops FS 3 and 4 from reading in and picks up co-selector number three.

The last card is a master card with an x in columns 2 and 4. As before, the x in column 2 picks up co-selector # 1 immediately; an x in column 4 picks up pilot selector # 3, which picks up punch selectors on the next card cycle and also, impulses the counter to read out (not read out and reset or there is a back-circuit), and General Storages 1, 2, 3, 4 to read out.

Numerical Integration by Simpson's Rule

Bruce G. Oldfield

U.S. Naval Ordnance Test Station

A forty program 604 control panel, wired as described below, enables the computation of the $\int f_{(x)}\, dX$, using Simpson's rule, and the punching of the values of x and $f_{(x)}$. The enclosed control panel will automatically instruct the machine to take care of the weighting factors by which the ordinates are to be multiplied, take care of the correction for the first and last ordinate, and in addition, automatically take care of any number of changes of X.

Using this 604 control panel, one can integrate by Simpson's rule at a rate of 100 ordinates per minute minus two master cards per complete integration. The intergration technique is greatly simplified, eliminating a number of steps and possible sources of error. Since the control panel automatically supplies the correct weighting factor to the ordinate, corrects for end value ordinates, and takes care of all interval changes ΔX, the only requirement is that the independent variable be in order with no missing values.

The fact that a change in ΔX is no additional trouble for the 604, will allow the mathematician to change ΔX frequently in order to maintain a specified accuracy. This should eliminate a large number of values of the function that are now computed unnecessarily because, with former methods it has been easier to evaluate the function than to introduce frequent interval changes.

(Read) FS 1 and 2	(Stored) FS 3 and 4	(Stored) Counter	(Stored) GS 1 and 2	(Stored) GS 3	(Read) GS 4
Y_{n+1}	Y_n	X_n	Σ	$-\Delta X_n$	X_{n+1}
xx.xxxxx	xx.xxxxx	xx.xxx	xx.xxxxxx	.xxx	xx.xxx

	(1) GS4 (X_{n+1}) → ctr. neg. - - - -	$(-\Delta X_{n+1})$
	(2) GS3 ($-\Delta X_n$) → ctr. neg. :B. T. - - -	$(-\Delta X_{n+1} + \Delta X_n)$
	(3) Ctr. R and R → MQ	
(S on - bal.)	(4) MQ → ctr. neg. :B. T.	
	(5) Ctr. R and R	
	(6) MQ → ctr. - - -	$(-\Delta X_{n+1} + \Delta X_n)$
	(7) GS 3 ($-\Delta X_n$) → ctr. and MQ - -	(ctr. has $-\Delta X_{n+1}$)
(S on + bal.)	(8) Ctr. R and R → GS 3 - -	$(-\Delta X_{n+1})$
(S on + bal.)	(9) FS 3 and 4 R.O. - Mult. - -	$(-\Delta X_n \cdot Y_n)$
(S on + bal.)	(10) GS 1 and 2 → ctr. into 3rd - -	$(3\int_0^n YdX)$
(S on + bal.)	(11) GS → MQ	$(-\Delta X_{n+1})$
(S on + bal.)	(12) FS 3 and 4 R.O. - mult. neg. - -	$(3\int_0^n YdX + Y_n \Delta X_{n+1})$
(S on + bal.)	(13) Ctr. R and R → GS 1 and 2 out of 3rd	
Programs 1 to	(14) Ctr. R and R	
23 Normally	(15) FS 1 and 2 R.O. - mult. neg. - -	$(-Y_{n+1})(-\Delta X_{n+1})$
active: Cal.	(16) Ctr. R and R → FS 3 and 4 out of 2nd	
Selector # 1	*(17) Emit 2 or 4 → MQ	
and # 2.	(18) FS 3 and 4 R.O. - mult.	$(\frac{2}{4} \cdot \Delta X_{n+1} Y_{n+1})$

*See Selector System

Ed. note: Programs 8 through 13 may be supressed by zero test if available.

	(19) GS 1 and 2 → ctr. into 2nd $(3\int_0^n YdX + Y_n \triangle X_{n+1} + 4Y_{n+1} \triangle X_{n+1})$	
	(20) $\frac{1}{2}$ adj. → ctr.	
	(21) Ctr. R and R → GS 1 and 2 out of 2nd	
	(22) FS 1 and 2 → FS 3 and 4	
	(23) X_{n+1} → ctr.	
<u>First card.</u>	(24) GS 3 → MQ	$(-\triangle X_o)$
Normally Sup.	(25) FS 1 and 2 R.O. - mult. neg.	$[-(-\triangle X_o)(Y_o)]$
Cal. Selector	(26) Ctr. R and R → GS 1 and 2 out of 3rd	$(\triangle X_o Y_o)$
# 3	(27) FS 1 and 2 → FS 3 and 4	(Y_o)
	(28) GS 4 → ctr.	(X_o)
	(29) Ctr. R and R	
<u>Last card.</u>	(30) GS 3 → MQ	$(-\triangle X_{n+i})$
Normally Sup.	(31) FS 3 and 4 R.O. - mult.	$(-Y_{n+i} \triangle X_{n+i})$
Cal. Selector	(32) GS 1 and 2 → ctr. into 3rd	$(3\int_0^{n+i} YdX)$
#4	(33) 3 → FS 1 and 2 into 6th	
	(34) FS 1 and 2 R.O. - Divide	
	(35) MQ → FS 3 and 4 into 3rd	
	(36) Ctr. R and R → GS 1 and 2	
	(37) GS 1 and 2 → ctr. into 6th	
	(38) FS 1 and 2 R.O. - Divide	
	(39) MQ → ctr. into 3rd	
	(40) FS 3 and 4 → ctr. into 6th	$(\int_0^{n+i} YdX)$

Punch from Counter

xx. xxxxxxxxx00

Positions 5 to 12

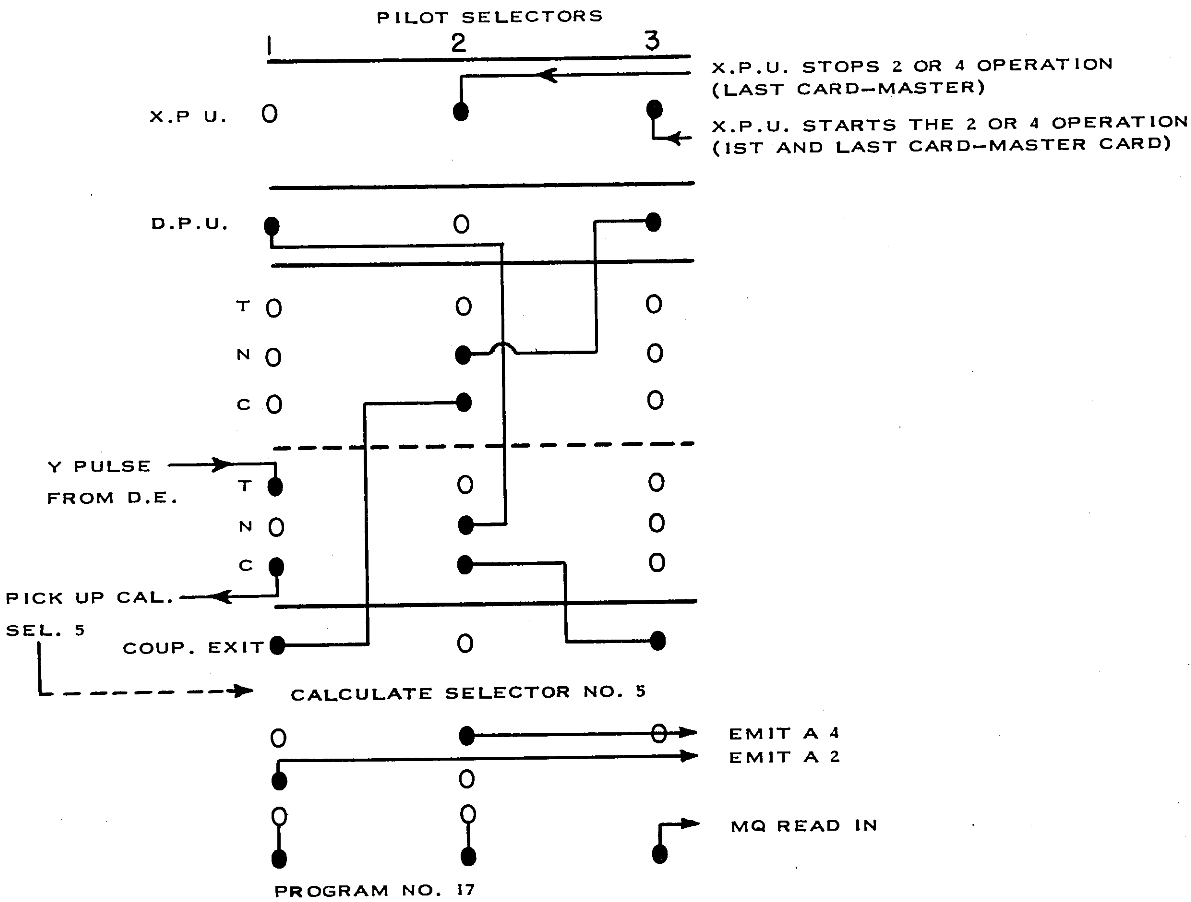
The first card is a master card with an "X" in column one, a "Y" in column two and minus $\triangle X_0$ on the card to read into GS 3. The last card is a master card with an "X" in column one and an "X" in column three. The integral value will be punched on this final master card.

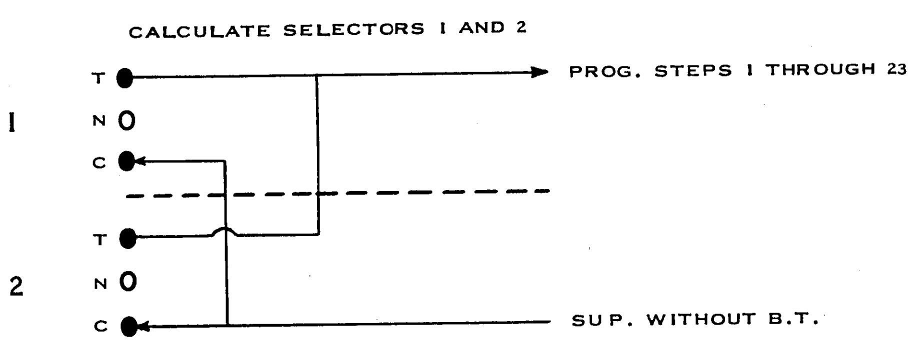
An "X" in column one picks up calculate selector number one, picks up calculate selectors two and three through the coupling exit of pilot selector four, and picks up pilot selector three.

A "Y" in column two goes directly to FS 3 and 4 read in, GS 1, 2, 3, read in, and counter read out and reset.

An "X" in column three picks up calculate selector number four, picks up pilot selector number five which picks up punch selectors through the coupling exit and which lets a card cycle pulse through to counter read out, and picks up pilot selector number two.

The following is the selector system which supplies the two or the four to program # 17.





Notes on using the integration control panel.

(1) The decimal point and numbers carried are as shown, this was set up for a particular problem and can be easily changed as desired.

(2) If more than forty programs are available the half adjusts can be added if desired. If only forty programs are available the half adjusts could still be added and compute $3 \int f_{(x)} dX$ instead of the true integral.

(3) Program seven will not work if Δ X is negative. If X is not taken as an increasing function, program seven must be separated into two programs, that is GS 3 → ctr. and GS 3 → MQ.

(4) Identification can be punched on the final master card by wiring from the second reading brushes through a punch selector.

(5) In checking a control panel remember that the two or four selector system must be stopped by an X in column three to pick up pilot selector number two. If the integration is stopped in the middle the selector system will go on until stopped.

(6) All "X" and "Y" control punches in columns one, two and three should naturally go through column splits.

(7) Minus ΔX must be punched on the first master card and read into GS 3.

Ed. note: Care must also be taken in changing the interval only after an even number of intervals, since Simpson's formulae is only valid then.

Graphical Aids for Determining Shift Codes

Amy McAvory and William D. Bell

Telecomputing Corporation

A recurring problem when using the Card-Programmed Electronic Calculator is locating the decimal point in any given operation. Using an eight column channel entry into the 604, we have assumed that both factors are three whole numbers and five decimals entering on channel A and B. The results of any arithmetical operation are in the electronic counter as a number with six whole numbers and seven decimals. With this particular control panel wiring a column shift of three will select the same decimals as were put in, that is, three whole numbers and five decimals. No shift will select the extreme eight positions to the left, and a five shift will select the extreme eight positions to the right.

The two graphical devices below are for determining the decimal point for multiplication or division. The use is quite simple. The number of decimal points in the two factors and the number of decimals desired in the answer enables the proper shift to be read directly. We have found this to be an extremely useful device.

Combination

Multiplication Shift Control

Shift

MULTIPLICAND														
8									0	1	2	3	4	5
7								0	1	2	3	4	5	
6							0	1	2	3	4	5		
5						0	1	2	3	4	5			
4					0	1	2	3	4	5				
3				0	1	2	3	4	5					
2			0	1	2	3	4	5						
1		0	1	2	3	4	5							
0	0	1	2	3	4	5								

MULTIPLIER														
8	0	1	2	3	4	5	6	7	8					
7		0	1	2	3	4	5	6	7	8				
6			0	1	2	3	4	5	6	7	8			
5				0	1	2	3	4	5	6	7	8		
4					0	1	2	3	4	5	6	7	8	
3						0	1	2	3	4	5	6	7	8
2							0	1	2	3	4	5	6	7
1								0	1	2	3	4	5	6
0									0	1	2	3	4	5

DECIMALS ANSWER

For example:

If there are 3 decimals in the multiplier, 4 decimals in the multiplicand, and 4 decimals are desired in the answer, then by following the arrows shown in the diagram, it is seen that there will be a shift of 5.

Combination

Divide Shift Control

Shift

DIVIDEND	8	5	4	3	2	1	0								
	7		5	4	3	2	1	0							
	6			5	4	3	2	1	0						
	5				5	4	3	2	1	0					
	4					5	4	3	2	1	0				
	3						5	4	3	2	1	0			
	2							5	4	3	2	1	0		
	1								5	4	3	2	1	0	
	0									5	4	3	2	1	0
DIVISOR	8	7	6	5	4	3	2	1	0						
	7	8	7	6	5	4	3	2	1	0					
	6	9	8	7	6	5	4	3	2	1	0				
	5	10	9	8	7	6	5	4	3	2	1	0			
	4	11	10	9	8	7	6	5	4	3	2	1	0		
	3	12	11	10	9	8	7	6	5	4	3	2	1	0	
	2	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	1	14	13	12	11	10	9	8	7	6	5	4	3	2	1
	0	15	14	13	12	11	10	9	8	7	6	5	4	3	2

DECIMAL ANSWER

"Floating Decimal" Calculation on the IBM Card - Programmed Electronic Calculator

Donald B. MacMillan and Richard H. Stark

There is a wide variety of problems occurring in run-of-the-mill computing in which it is extremely helpful to represent numbers x as $x_0 \cdot 10^p$ where x_0 has the same significant digits as x within the capacity of the machine used, $1 \leq x_0 < 10$, and p is an integer. Three types we might mention are:

1) Problems in which certain computed quantities have magnitudes difficult to estimate.
2) Problems in which one or more quantities have such a wide range of magnitude that no single fixing of the decimal point will suffice for the entire range.
3) Problems in which the setup time involved in estimating magnitudes is not justified by the saving in machine time made possible by fixed decimal calculation.

For application of the IBM Card-Programmed Electronic Calculator to such problems, we have devised a calculator programming of general purpose type based on the "floating decimal" representation of numbers. A brief explanation of the way the machine is instructed will help in an understanding of what follows.

The calculating unit (IBM 604) of the CPC is, for one step in calculation, instructed to receive numbers A and B, to perform on them an operation f_i (A, B), and to leave the result C where it can be used for further calculation or can be stored. The numbers A, B, and C may have up to ten digits and a sign. The choice of operations f_i is determined to suit the needs of the group using the machine. The more calculation one can put into one operation, the faster the CPC can do the problem. Hence, for a large problem, it is sensible to design the operations f_i with this in mind. A more versatile, but slower setup, is one in which the same set of operations will suffice for any problem. It will rather obviously include among its operations addition, subtraction, multiplication, and division. What additional operations are included depends upon the needs of the group using the machine. The choice of operations is put into effect by wiring of control panels. Most operations are completely instructed by wiring of the control panels for the 604, but on occasion, it is useful to alter A or B before they are delivered to the calculator, or C as it comes from the calculator. This implies special wiring on the accounting machine control panel.

In our programming of the CPC for general-purpose floating decimal calculation, we represent a number x by a ten digit number X. In the representation $x = x_0 \cdot 10^p$, we carry eight significant digits in x_0 and allow p the range $-50 \leq p < 49$. The first eight digits of X are the digits of x_0. The last two digits of X are those of p + 50. The sign of X is that of x_0. Examples are:

x	X
$4.1752139 \cdot 10^{4}$	4175213954
$-1.9920367 \cdot 10^{-20}$	-1992036730

We shall seldom be careful to distinguish between the number and its "floating decimal" representation.

The set of operations which is available in our setup is as follows:

$$f_0\ (A,B) = A,$$
$$f_1\ (A,B) = A + B,$$
$$f_2\ (A,B) = A - B,$$
$$f_3\ (A,B) = A \cdot B,$$
$$f_4\ (A,B) = A/B,$$
$$f_5\ (A,B) = \sqrt{A},$$
$$f_6\ (A,B) = |A|\ /B,$$
$$f_7\ (A,B) = |A| \cdot B$$

Corresponding to i = 3, 4, 6, 7 there is an operation f'_i such that $f'_i = -f_i$.

If no number is delivered to the calculator to be used as A or B in a calculation, then the number already in that position is used. Because numbers delivered to storage cannot be read from storage for use as A or B immediately, it is often convenient to store the result of a calculation in A or B position in the calculator before the next instructions are given. For all operations except f_o, we automatically store C in A and B positions. For f_o, C is stored in B position, and the unused B in A position.

Operations 6 and 7 are made possible by subsidiary wiring on the accounting machine control panel which replaces A by $-|A|$. Hence, neither of these operations can make use of holding the previous result within the calculator in A position.

Our floating decimal programming for the calculator relies heavily on the program repeat feature of the 604 used in the CPC. This feature allows the machine, given a set of instructions, to repeat them several times in one operation and, if desired, to use one subset of these instructions on one time through a set, a different subset the next time.

This use of program repeat will make addition or subtraction take two or three times normal computing time if A and B are different by several orders of magnitude or if C has magnitude much smaller than that of A and B. As is to be expected, square root operation will always be slow. The other operations will always be finished in the minimum time for a single operation.

Although it is rather expensive in selectors on the accounting machine control panel, one can easily instruct the machine to make a reasonable (within a factor of 4) guess at the qth root of a number which then is improved by iteration.

The guess at the qth root of $x_0 \cdot 10^p$ is $3 \cdot 10^{\left[\frac{p}{q} + 50\right] - 50}$ where $\left[\frac{p}{q} + 50\right]$ is the largest integer less than $\frac{p}{q} + 50$.

The zero of our "floating decimal" setup cannot conveniently have all its digits zero. An explanation of our addition will demonstrate the origin of our representation. Suppose two numbers differing in absolute value only in the eighth significant digit, but of opposite sign, are added. Then the digits of the result will lead off with zeroes. In order to change the answer back to standard representation, we examine the left hand digit. If it is zero, we shift the digits of the answer to the left one position and decrease the calculated p by one. This shifting is continued until the left hand digit is non-zero. To enable the machine to stop shifting, if all eight digits of the answer are zero, we subtract 1 in the right hand digit of the answer each time a shift is made after the first. Thus on our machine,

$$x_0 \cdot 10^p - x_0 \cdot 10^p = -1.1111111 \cdot 10^{p-9}$$

This shifting device gives an obvious means for converting fixed decimal numbers of eight digits or less to floating decimal. Suppose the number of largest magnitude in a column of figures is $x_0 \cdot 10^p$.

We punch $x_0 \cdot 10^p$ into the card with standard "floating decimal" representation and all the other numbers with the same last two digits and the same decimal position. Then transferring these numbers through the 604 will convert them to standard "floating decimal" representation.

Details of Control Panel Wiring

When Channels A and B are transmitted to the 604, the significant digits of Channel A are entered in General Storage 1 and 2, its exponent in Factor Storage 4; the significant digits of Channel B are entered in Factor Storage 1 and 2, and its exponent in Factor Storage 3. The sign entries of FS 1 and 2 and GS 1 and 2 are wired from the channel sign hubs, in normal fashion.

The sign wiring of FS 3 and FS 4 is unusual, and important to the operation of the control panel. Its effect is that these two units always read true figures with negative sign. Two pilot selectors are used. The first is picked up whenever a spread read-in card is at the third brushes. The second is picked up immediately by the impulse from "Transfer and S.P. x Control Plus" hub for counter 2A, which is wired to total on all cycles. FS 3 sign hub is wired to common of one position of the first selector. The transferred hub of that selector is wired to the 10 impulse emitter, and the normal hub is wired to common of one position of the second selector. Transferred of the second selector is wired to the 10 impulse emitter, normal is wired to the sign of Channel B. FS 4 is wired similarly in the other position of each of the two pilot selectors, with the final connection to sign of Channel A.

The significant digits of Channel C are in GS 3 and 4, and the exponent in the two lowest order positions of the 604 counter. Channel C sign is to be taken from GS 3 and 4.

The absolute value device is this: an x-punch is used to instruct the machine; it picks up a pilot selector, which transfers the sign of GS 1 and 2 from Channel A sign hub and connects it to the sign of FS 4. Then GS 1 and 2 contains the negative of the absolute value of A.

The coding that has been used for this control panel is:

No Instruction	Addition	(f_1)
2	Subtraction	(f_2)
3	Multiplication	(f_3)
4	Division	(f_4)
4 and 5	Square Root	(f_5)
7	Transfer	(f_0)

The effect of instruction 2 is just to substitute -B for B. Therefore we also have

2 and 3	Negative Multiplication	(f'_3),
2 and 4	Negative Division	(f'_4).

If we use the x-punch for absolute value in addition to operation codes, we have

x, 2 and 4	(f_6)
x, 2 and 3	(f_7)
x and 4	(f'_6)
x and 3	(f'_7)

Calculate selectors 2, 3, 4, 5, and 7 may be picked up in the ordinary way. Selector 1 is to be picked up whenever either operation code 3 or 4 is present. Selector 6 is to be coupled to selector 5. Selector 8 is to be coupled to selector 4.

On the programming chart, some positions are marked with an asterisk. These are positions where the instruction must be selected according to which operation is being performed. The selection is indicated in the "remarks" column. In that column the symbol 012, for instance, refers to program 01, exit 2.

The significance of the symbols at the program positions is determined by the heading of the column in which they appear. For instance, GS 1 and 2 in the Exit 1 column means General Storage 1 and 2 read-out. The same symbol in the Exit 2 column means General Storage 1 and 2 read-in. The abbreviations R and R and RO refer to the counter. We have used the symbol # to help distinguish between abbreviations for General Storage and those for Group Suppression. Thus we have GS 3 and 4 (General Storage 3 and 4) and GS # 4 (Group Suppression number 4). We have written miscellaneous orders (Balance Test for Step Suppression, group suppression controls, etc.) in Exit 3 where possible.

Editor's Note:

An alternative method of controlling the conversion of the exponents in factor storage 3 or 4 has been successfully accomplished. The wiring is shown below, and involves the use of only one pilot selector. Counter group one may be used for other purposes.

<u>Pilot Selector</u>

digit 8 (every cycle) →	0	Immediate P U
digit 5 (every cycle) →	0	T
channel A sign →	0	N
factor storage 4 →	0	C
digit 5 →	0	T
channel B sign →	0	N
factor storage 3 →	0	C

The positions of FS 3 and FS 4 not used for the exponents digits must be wired to the appropriate channel A or B 9's emitter.

Suppression Types for General Purpose Floating Decimal Programming for CPC 604

Types of Suppression:

I Group suppress #1.

II Group suppress #2.

III Group suppress #3.

IV Group suppress #4.

V Suppress on negative balance.

VI Suppress on positive balance.

VII Suppress on non-zero balance.

X Suppress if selector 5 is normal.

XI Suppress if selector 5 is up.

XII Suppress on negative balance if selector 5 is up; Suppress if selector 5 is normal.

XIII Suppress on negative balance; Suppress if selector 1 is up.

XIV Suppress on negative balance if selector 1 is normal; Group suppress # 1 if selector 1 is up.

XV Group suppress # 1 if selector 4 is up.

XVI Suppress on negative balance if selector 1 and 5 are normal; Suppress on positive balance if selector 5 is up.

XVII Suppress if selector 4 is up.

XVIII Group suppress # 1; Suppress if selector 5 is up.

XIX Group suppress # 1 if selector 4 is normal; Suppress if selector 4 is up.

XX Group suppress # 4 if selector 5 is normal.

XXI Group suppress # 1 if selector 4 is up; Suppress if selector 4 is normal.

XXII Suppress on zero balance; Suppress if selector 3 is up; Group suppress # 1 if selector 7 is up.

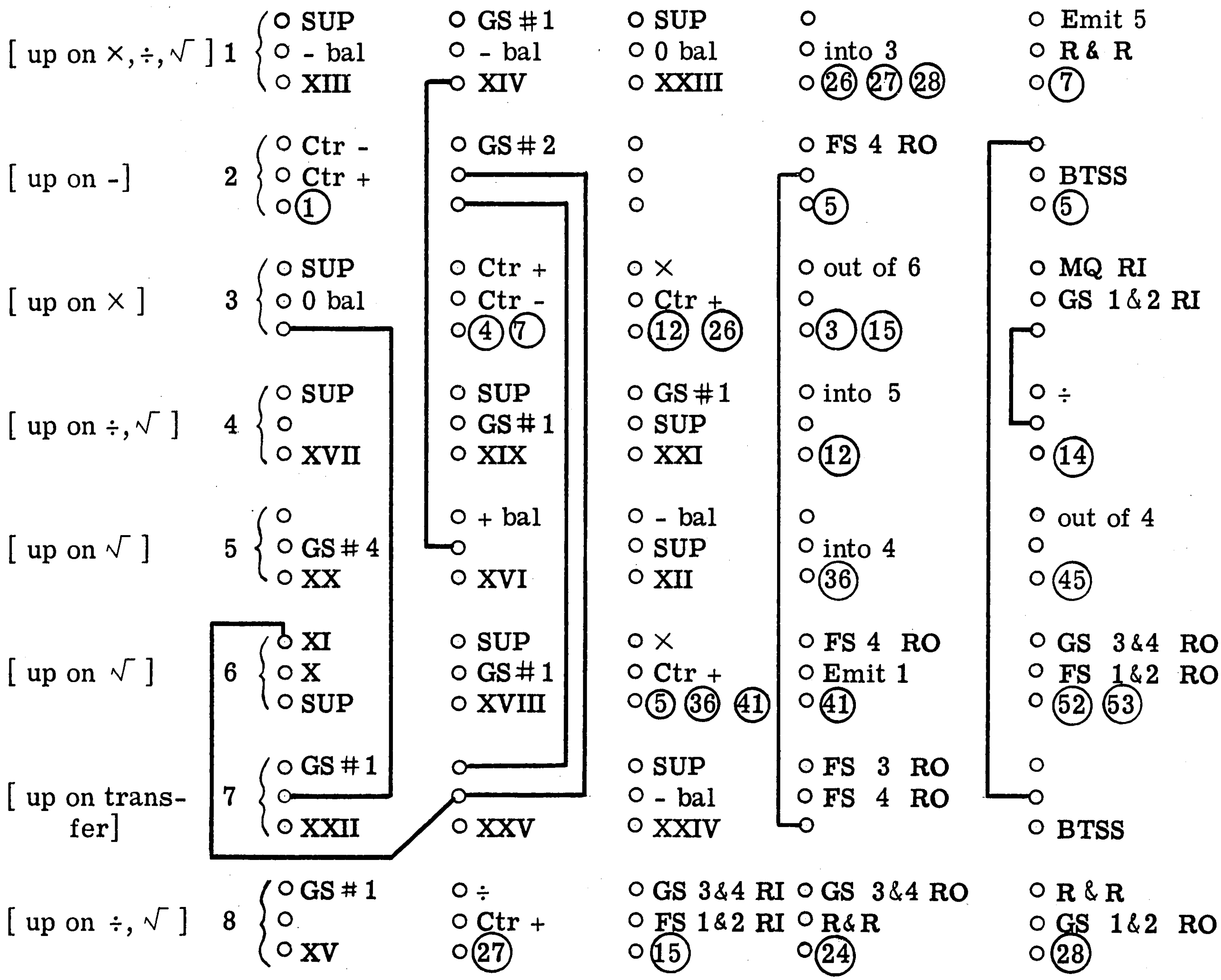
XXIII Suppress on zero balance if selector 1 is normal; Suppress if selector 1 is up.

XXIV Suppress on negative balance; Suppress if selector 7 is up.

XXV Suppress if selector 6 and 7 are up and selector 2 normal; or selector 6 up and 7 normal; Group suppress # 2 if both selector 2 and 7 are up.

Note: Calculate selectors 4 and 8 (also 5 and 6) are coupled so that when calculate selector 4 (5) is mentioned above it may appear on calculate selector 8 (6) on the calculate selector diagram.

CALCULATE SELECTOR
Number



Note :
④ denotes program step 4

GENERAL PURPOSE FLOATING DECIMAL PROGRAMMING FOR C.P.C. 604

	READ OUT EXIT 1	READ IN EXIT 2	SHIFT EXIT 3	TYPE SUP.	COMMENT(S)
1	FS 1&2	CTR + *	GS #4 PU	II	012 CTR - IF SEL 2 IS UP
2	R O	MQ		XVII	
3	R&R	FS 1&2	*	II	033 OUT OF 6 IF SEL 3 IS UP
4	FS 3	CTR - *		XVIII	042 CTR + IF SEL 3 IS UP
5	FS 4	CTR + *	*	none	051 FS 3 IF SEL 7 IS UP
6		B.T.S.S.	ZERO TEST	XXIII	052 x + IF SEL 5 IS UP
7	R&R *	CTR - *	INTO 2	XI	053 B.T.S.S. IF SEL 2 or
					SEL 7 DOWN
8	FS 4	CTR +		XIII	071 EMIT 5 IF SEL 1 IS UP
9	FS 3	FS 4		XIII	072 CTR + IF SEL 3 IS UP
10	R&R	FS 3		XIV	
11	FS 3	CTR +	B.T.S.S.	X	122 x + IF SEL 3 IS UP
12	GS 1&2	CTR + *	*	XVI	123 INTO 5 IF SEL 4 IS UP
13	GS 1&2	CTR +	INTO 4	XII	142 MQ IF SEL 3 IS UP
14	FS 1&2	GS 1&2 *		XIV	142 ÷ IF SEL 4 IS UP
15	R&R	FS 1&2 *	*	XIV	152 GS 3&4 IF SEL 4 IS UP
16	FS 1&2	CTR +		XXIII	153 OUT OF 6 IF SEL 3 IS UP
17	HALF ADJ.			XXIII	
18	R&R	FS 1&2	OUT OF 2	XXIII	
19	FS 3	CTR +		XXIII	
20	EMIT 1	CTR -		XXIII	
21	R O	FS 3		XXIII	
22	FS 4	CTR -		XXIII	
23			ZERO TEST	XXIII	
24	R&R *	CTR +	INTO 6	XV	241 GS 3&4 IF SEL 4 IS UP
25	MQ	GS 3&4	INTO 2	XXI	262 x + IF SEL 3 IS UP
26	GS 1&2	CTR + *	INTO 3 *	XIX	263 INTO 1 IF SEL 1 IS UP
27	FS 1&2	CTR + *	INTO 3 *	I	272 ÷ IF SEL 4 IS UP
28	GS 1&2 *	CTR -	INTO 3 *	XXII	273 INTO 1 IF SEL 1 IS UP
29	EMIT 1	CTR -	INTO 3	XX	281 R&R IF SEL 4 IS UP
30	GS 3&4	CTR +	INTO 6	XXI	283 INTO 1 IF SEL 1 IS UP
31	MQ	CTR +	INTO 2	XXI	
32	HALF ADJ.	GS #3 PU	INTO 2	I	
33	R&R	GS 3&4	OUT OF 3	X	
34	EMIT 5	MQ	INTO 3	none	
35	FS 1&2	x +	GS #3 DO	X	
36	GS 3&4	CTR + *	INTO 4 *	III	362 x + IF SEL 5 IS UP
37	R O	GS 3&4	OUT OF 3	XI	363 INTO 1 IF SEL 5 IS UP
38	HALF ADJ.		INTO 3	none	
39	R&R	FS 1&2	OUT OF 4	none	
40	EMIT 5	FS 3	INTO 2	X	
41	EMIT 1 *	CTR + *	GS #4 DO	III	411 FS 4 IF SEL 5 IS UP
42	EMIT 1	CTR -	GS #2 DO	XIX	412 x + IF SEL 5 IS UP
43	FS 3	CTR +		XI	
44	FS 3	x -		X	
45	R&R	FS 3	*	none	453 OUT OF 4 IF SEL 5 IS UP
46	FS 1&2	CTR -	GS #1 PU	VII	
47	GS 3&4	CTR +	GS #1 DO	X	
48	R O	GS 3&4	OUT OF 3	X	
49	R&R		B.T.S.S.	none	
50	MQ	CTR -	INTO 5	XXV	
51	MQ	CTR -	INTO 5	XXV	
52	FS 1&2 *	CTR +		VI	521 GS 3&4 IF SEL 5 IS UP
53	FS 1&2 *	CTR -		V	531 GS 3&4 IF SEL 5 IS UP
54					
55	R&R	GS #3 DO	B.T.S.S.	none	
56		GS #2 PU	PROG. RPT	VI	
57	FS 1&2	GS 1&2		XXIV	
58	FS 3	FS 4		XXIV	
59	FS 3	CTR +		V	
60	FS 1&2	GS 3&4		V	

† AND SEL 2 IS DOWN

IBM

TRADE-MARK

INTERNATIONAL BUSINESS MACHINES CORPORATION
World Headquarters Building • 590 Madison Avenue, New York 22, N. Y.

Form No. 22-8477-0-3M-BB-P

Litho. in U.S.A.