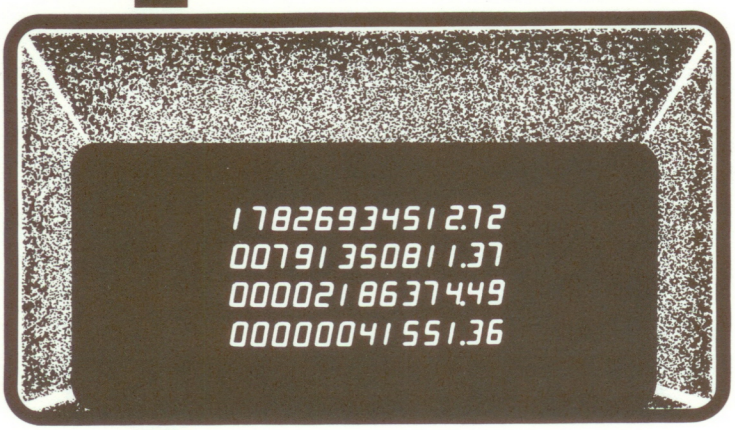
Friden, Inc.



ELECTRONIC CALCULATORS

SERVICE MANUAL

Friden, Inc.
SAN LEANDRO · CALIFORNIA
A SUBSIDIARY OF THE SINGER COMPANY

ELECTRONIC CALCULATOR 132
TEMPORARY SERVICE MANUAL

TABLE OF CONTENTS

FOREWORD

The EC-132 is the basic EC-130 modified for Square Root Function. Servicing the EC-132 necessitates a thorough knowledge of the EC-130. Therefore, the EC-132 Service Manual contains only the correlation of the EC-132 with the EC-130 and a description and explanation of the circuits which have been added or redesigned.

Any particular area of the EC-132 which is not covered, then, indicates that there is no change from the EC-130 and information regarding that area will be found in the EC-130 section of the Electronic Calculators Service Manual.

SECTION 1 GENERAL DESCRIPTION

SECTION 1 - GENERAL DESCRIPTION

A. SIMILARITIES TO THE EC-130

In all but a relatively few respects the EC-132 looks exactly like the EC-130. The external differences are: the addition of a Square Root key to the keyboard; the change of CLEAR ALL key to read CLEAR DISPLAY; the addition of a pilot light located in the lower right portion of the keyboard mask; the optional addition of an Entry Counter to the right side of the CRT mask; and a new Decimal Point Position Thumbwheel.

B. DIFFERENCES FROM THE EC-130

The internal differences of the EC-132 and the EC-130 are relatively minor with the exception of the addition of Square Root Function Logic. All of the logic of the EC-130 has been retained intact with respect to functions other than Square Root with the exception of Clear All, which is now Clear Display and affects Registers 1 through 4 (R1-R4) only.

1. SQUARE ROOT KEY

Depressing the Square Root key on the keyboard initiates certain machine operations that result in the square root of the number that is displayed in R1 being extracted and displayed as an answer in the answer register (R1).

2. CLEAR DISPLAY KEY

Depressing the CLEAR DISPLAY key on the keyboard results in the numbers in all display registers (R1, R2, R3, R4) being discarded. This is done by inhibiting the count in A counter (Increment A) from the delay line circuits. Since only the display registers are affected, RS, the undisplayed storage register, is not cleared.

3. DECIMAL POINT SWITCH

In the EC-130, the Decimal Point switch was installed in the keyboard assembly. In the EC-132, the switch has been removed from the keyboard assembly and installed underneath the keyboard mask in close proximity to the Decimal Point Position Thumbwheel. Also, the switch is a completely new design that permits, with associated logic, decimal point position in any one of the 13 display columns, or operation with no decimal point.

4. DECIMAL POINT POSITION

In the EC-130, the decimal point position was limited to one of five places. In the EC-132, the decimal point may be positioned in any one of the display columns, or operations performed with no decimal point. The Decimal Point Position Thumbwheel has been slightly redesigned to locate these 14 positions (0 and 1-13) and to operate with the new Decimal Point switch.

5. REGISTER 4 (R4) DATA RETENTION

In the EC-130, any operation involving a shift down of data places R4 data in R3 and places zeroes in R4. In the EC-132, an operation involving shift down retains R4 data in R4 and repeats R4 data in R3. In effect, then, R4 becomes another storage register for the retention of data. However, in any operation involving shift up R4 data is discarded, as is the case in the EC-130.

6. ENTRY COUNTER

The Entry Counter (optional) in the EC-132 permits a total to be kept of the number of entries of a group of entries as they are placed in the machine. The Entry Counter is a four-digit counter that is installed on the right-hand side of the CRT display. A RESET button is included to permit reset of the counter to zero before the initiation of a new group of numbers.

7. PILOT LIGHT

In the EC-130 and EC-132, dividing by zero keeps the machine in an operational loop that keeps the Display blanked. In this condition there is no Display indication that the machine is on. If the machine should be covered while On, heat build-up could cause damage to semi-conductors and to other components. In the EC-132, a Pilot Light is installed at the lower, right-hand corner of the keyboard mask that indicates an On condition when power is applied to the machine.

SECTION 2 ANALYSIS of OPERATION

SECTION 2 - ANALYSIS OF OPERATION

A. SQUARE ROOT MATHEMATICS

The EC-130 uses a modified "Sum of the Odd Integers" method of extracting a square root. The method usually taught in school is not easily adapted to digital techniques, thus another system must be used. The sum of the odd integers is but one of many ways of arriving at a square root.

Odd integers are the whole numbers 1, 3, 5, 7, 9, 11, etc. If such a succession of odd integers, (odd-order arithmetic progression), are added, each summation will result in a perfect square. A perfect square is a number whose square root is a whole number. The following table illustrates this.

1) $1 + 0 = 1 = 1^2$
2) $1 + 3 = 4 = 2^2$
3) $1 + 3 + 5 = 9 = 3^2$
4) $1 + 3 + 5 + 7 = 16 = 4^2$
5) $1 + 3 + 5 + 7 + 9 = 25 = 5^2$

By adding the next higher odd order integer to each preceding summation, all existing perfect squares are produced, and none is omitted. Note that the NUMBER of odd order integers in each summation IS the square root of the sum. In example 5, there are 5 parts to the summation, and 5 is the square root of 25. This holds true for any combination of successive odd integers.

As another example:

$1 + 3 + 5 + 7 + 9 + 11 + 13 = 49$
1; 2; 3; 4; 5; 6; 7;

Therefore, $\sqrt[2]{49} = 7$

By reversing this procedure, the square root can be extracted. In this case, we SUBTRACT successive odd integers to arrive at the root, starting at 1.

For example:

16	15	12	7
- 1	- 3	- 5	-7
15	12	7	0
(1)	(2)	(3)	(4)

Therefore, $\sqrt[2]{16} = 4$

The number of successful subtractions is the square root. If a larger number is used, more subtractions are necessary.

64	63	60	55	48	39	28	15
- 1	- 3	- 5	- 7	- 9	-11	-13	-15
63	60	55	48	39	28	15	0
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)

The square root of 64, then, is 8.

This is a satisfactory method for use with perfect squares. A slight change is necessary, however, to alter the method for numbers that do not have an integral square root. Rather than count the subtractions to arrive at the root, we can apply a simple formula.

In the last example, ($\sqrt[2]{64}$), note that the last number successfully subtracted is 15. Adding 1 to this, (15 + 1 = 16), then dividing by 2, (16/2 = 8), gives the square root in another way. Expressed as a formula:

R = (n + 1)/2, when R is the last integer to be successfully subtracted, (no overdraft), and is sometimes called the "partial root".

An example will show the complete method developed so far.

100	99	96	91	84	75	64	51	36	19
- 1	- 3	- 5	- 7	- 9	-11	-13	-15	-17	-19
99	96	91	84	75	64	51	36	19	0

$$R = \frac{n+1}{2} = \frac{19+1}{2} = \frac{20}{2} = 10$$

Thus, the square root of 100 is 10.

The method developed to this point is perfectly valid for small numbers, but it is cumbersome for large numbers. Since the number of subtract cycles is equal to the square root of the number, an excessive length of time is necessary to complete a problem using large numbers. To avoid this, another change must be made.

Starting at the decimal point, the number is divided into pairs of digits, called couplets. Beginning at the most significant couplet, the answer can be derived, one digit being produced for each couplet. However, the couplets must be operated upon in a way that is somewhat different than was done in the previous examples.

If the number that is to have its square root extracted is 441, it appears as 04 41 when divided into couplets. The root of each couplet will become one digit of the final square root, which will have 2 digits to the left of the decimal point. However, the root of 04, and the root of 41 cannot be combined in any way to produce the root of the original number. The following example will serve to illustrate the method used with the couplets.

The first step is to operate upon the most significant couplet by subtracting successive odd integers.

```
 04     03
+ 1   - 3
 03     00
```

Now, we do not apply the formula in its entirity. A 1 is added to the 3, but we do not divide by 2 yet. 3 + 1 = 4, and this is used to begin operation on the second couplet. First, however, since the partial root for the most significant couplet is in the 10's column of the final result it must be altered slightly by multiplying by 10.

The second step is to use this result, (4 x 10 = 40), add 1, and begin operations on the second couplet.

```
 41
-41
 00
```

Now we can apply the formula, and (41 + 1)/2 = 42/2 = 21

Thus, the square root of 441 is 21. In this example, there is no remainder, so the problem is completed at this point.

A number that results in a remainder in the first step is treated in much the same manner. For example, let us extract the square root of 841.

```
Step 1.      08  41  (couplets)
            -01
             07

             07
            -03
             04

             04
             05
        (overdraft)
         (restore)
```

In this case, 3 is the largest odd integer successfully subtracted and so becomes the partial root. Multiplying by 10 to adjust for the columnar position, after incrementing by 1, results in (3 +1) x 10 = 4 x 10 = 40.

Step 2. The remainder above, (04), combines with the other couplet (04 41). Use the partial root, (40), again incremented by one as we repeat the odd integer method.

04 41	04 00	03 57	03 12	02 65
- 41	- 43	- 45	- 47	- 49
04 00	03 57	03 12	02 65	02 16

02 16	01 65	01 12	00 57
- 51	- 53	- 55	00 57
01 65	01 12	00 57	00 00

The last number successfully subtracted is 57. Again applying the formula:

$$R = (n + 1)/2 = (57 + 1)/2 = 58/2 = 29$$

Thus, the square root of 841 is 29.

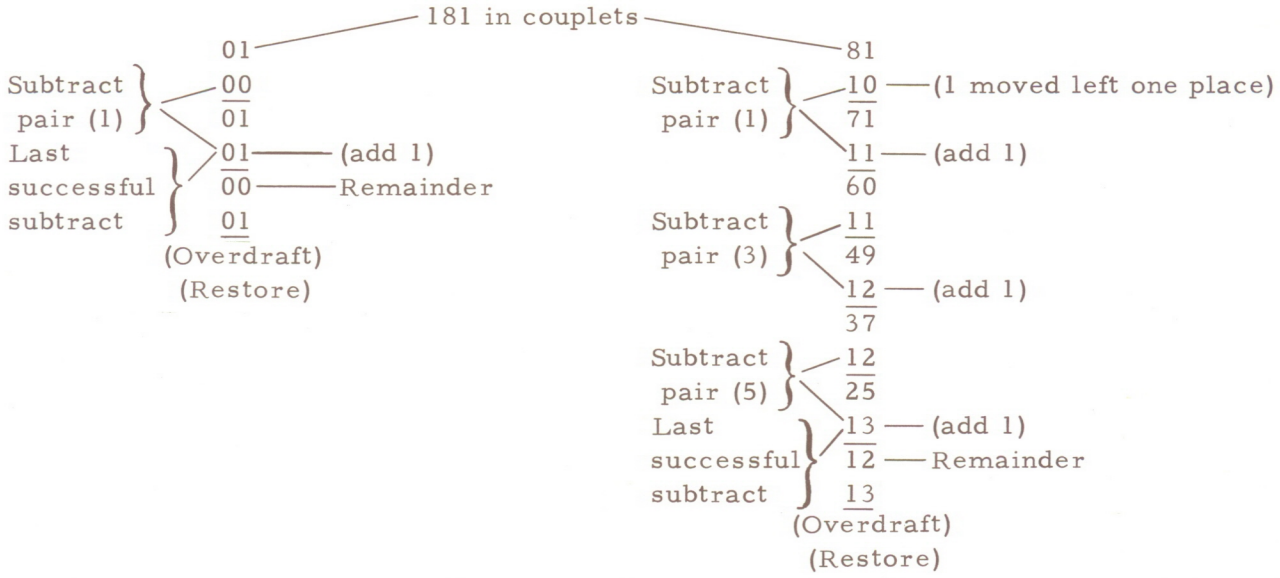
The method that has been developed up to this point is easily implemented by typical digital techniques. For a machine similar to the EC 130, however, it will require too much time to be a practical method. The final step of dividing by 2, which can add as much as one second to the total time required, is to be avoided if possible. The following procedure can be used to eliminate this step.

The previous examples used successive odd integers in the subtraction process, as 1, 3, 5, 7, 9, etc. The final step in the process is the division by 2. The same end result can be produced by separating each of the above subtraction cycles into 2 steps. The table below shows how this can be done. Note that for each normal subtract cycle there are 2 subtract cycles in the modified system. The first half of the subtract pair uses the previous integer developed in the prior step. (In the first subtract cycle this is of course 0.) The second half of the subtract pair uses the same integer with 1 added, (incremented by 1).

Thus, in the modified method, the first subtraction is by zero, and the second subtraction is by 1, (0 + 1). The first half of the second pair subtracts a 2, (1 + 1), and so on.

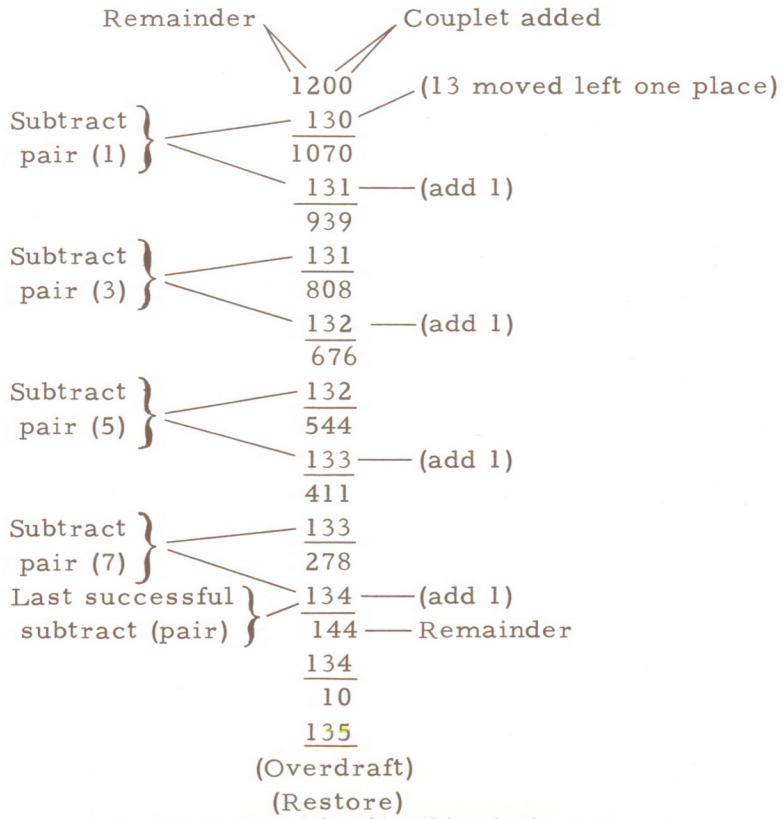
NORMAL METHOD		MODIFIED METHOD		
Subtract cycles	Odd integer	Subtract cycles		Odd integer
1	1	1st pair	1 2	(1) 0 0 + 1 = 1
2	3	2nd pair	3 4	(3) 1 1 + 1 = 2; 2 + 1 = 3
3	5	3rd pair	5 6	(5) 2 2 + 1 = 3; 3 + 2 = 5
4	7	4th pair	7 8	(7) 3 3 + 1 = 4; 4 + 3 = 7
5	9	5th pair	9 10	(9) 4 4 + 1 = 5; 5 + 4 = 9

Since there are twice as many subtract cycles as before, it is not necessary to divide by 2 at the final step. The last successful subtract IS the root by this method. A complete example to 4 significant digits in the root is given below to illustrate the way the method works.



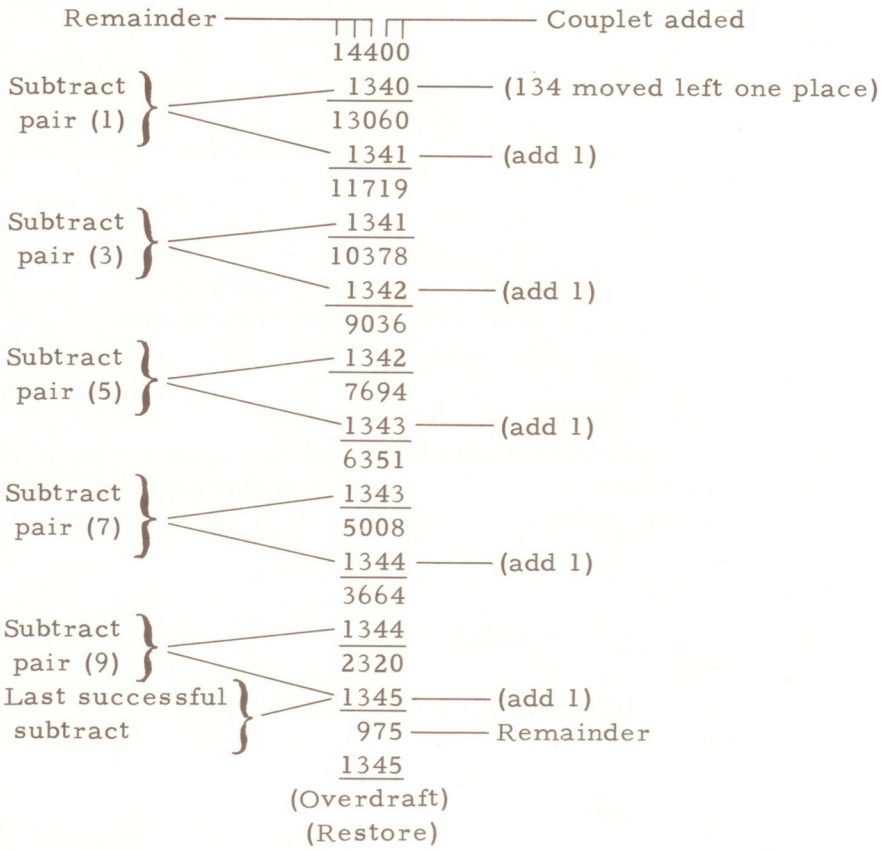
The last successful subtraction yielded the number 13, and the square root of 181 is 13 plus a decimal amount. This decimal amount can be found by continuing the subtractions using another couplet (00) and whatever remainder is left, in this case 12.

To illustrate, using remainder 12:



The last successful subtraction yielded the number 134, which properly decimal aligned would be 13.4. A closer approximation of the square root of 181, then, is 13.4. If more decimal places are desired, then another couplet (00) is used with the remainder each time for each decimal place.

To illustrate, using remainder 144:



The last successful subtraction yielded the number 1345, which properly decimal aligned would be 13.45. A closer approximation of the square root of 181 then is 13.45. If more decimal places are desired, then another couplet (00) is used with the remainder to give another decimal place. Addition of couplets and using the remainder each time can be used to give any desired degree of decimal accuracy.

In the foregoing illustration of square root, the odd integers were found by successively using the last number successfully subtracted as the first step in each subtract pair cycle, then adding 1 for the second step of the cycle. The odd integer can also be found by adding 1 to the last number successfully subtracted in the first step of the new subtract cycle, and for the second step using the last number without adding 1. In other words, mathematically it does not matter in which step of the subtract pair that the 1 is added; the sum of the two steps will still result in the next odd integer.

To illustrate using 25:

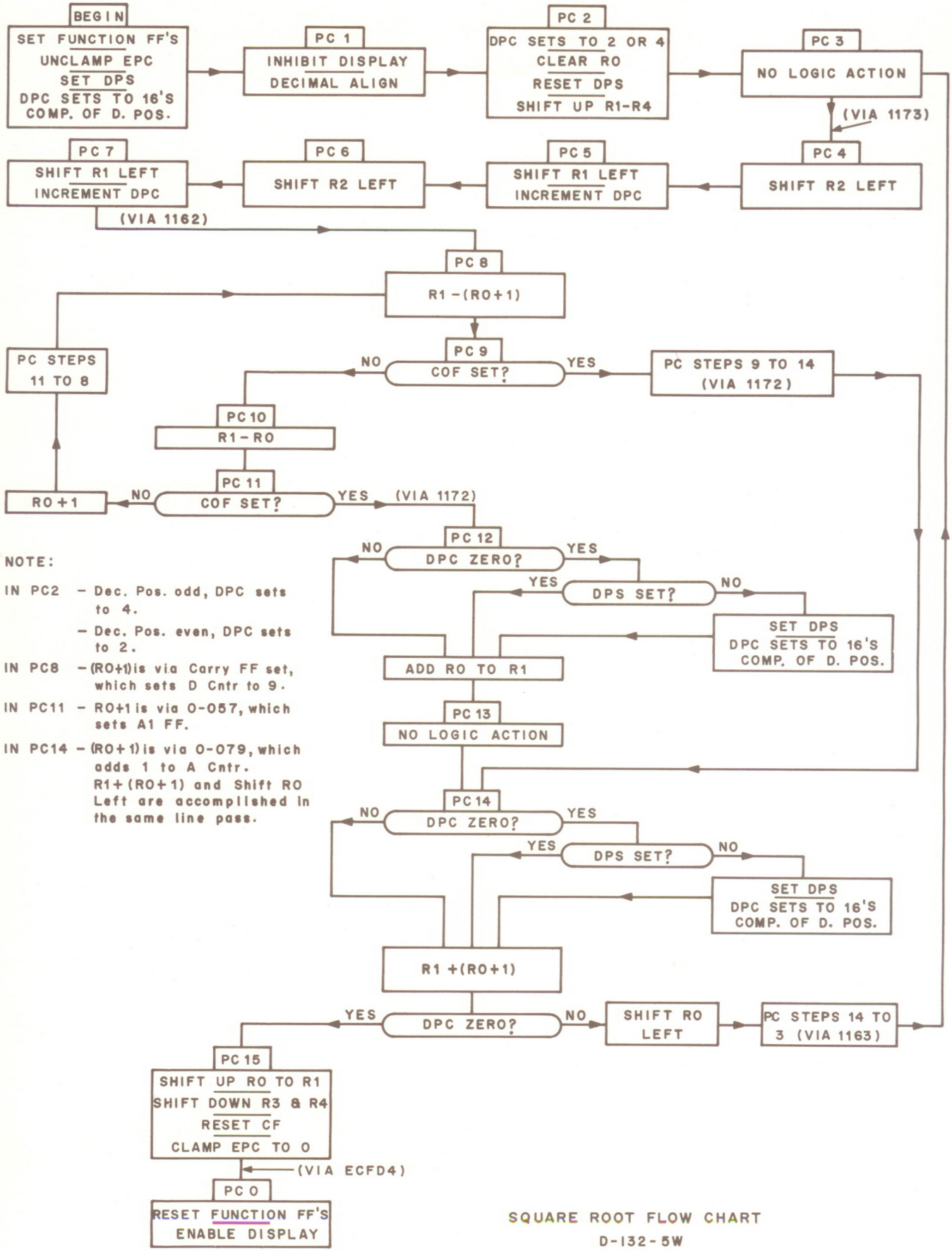
25	24	24	22
- 1	- 0	- 2	- 1
24	24	22	21
Subtract pair (1)		(3)	
21	18	16	12
- 3	- 2	- 4	- 3
18	16	12	9
(5)		(7)	
9	4		
- 5	- 4		
4	0		
(9)			

This is the method used in the EC-132 logic. In the first step of each subtract cycle 1 is effectively added to the last number successfully subtracted, without, however, affecting the retention of the last number. As the second step in each pair, the retained last number is subtracted. If both pairs of subtracts are successful, then 1 is directly added to the last successfully subtracted number, which then is used in the next subtract cycle. It is important to note that each subtract in a pair of subtracts must be successful before the number representing the last successful subtract is actually changed.

In general terms, the EC-132 handles the square root problem as follows.

The number begins in R1 and the Square Root key is pressed. First, R1, R2, R3, and R4 are shifted up to put the number in R2, then R1 and R0 are cleared out. R2 and R1 are then shifted to left twice to put a couplet into R1. Next, the subtract operation begins with the

answer (the odd order integer) being developed in R0. The subtract cycle continues until there is an overdraft, then the overdraft is cleared out by addition, and another couplet is shifted into R1. For every couplet that is shifted into R1, R0 is shifted left once. Thus, there is always one digit in the answer for each couplet in the original number. The answer is thus developed, digit by digit, in R0. At the end of the problem, R3 and R4 are shifted down, and R0 is shifted up into R1; these operations are done simultaneously, so it takes only one pass through the delay line.



NOTE:

IN PC2 – Dec. Pos. odd, DPC sets to 4.
– Dec. Pos. even, DPC sets to 2.

IN PC8 – (RO+1) is via Carry FF set, which sets D Cntr to 9.

IN PC11 – RO+1 is via O-057, which sets A1 FF.

IN PC14 – (RO+1) is via O-079, which adds 1 to A Cntr. R1+(RO+1) and Shift RO Left are accomplished in the same line pass.

SQUARE ROOT FLOW CHART
D-132-5W

B. SQUARE ROOT FLOW CHART

The Square Root Flow Chart is a graphic analysis of the logic operations of the EC-132 Square Root function. The Flow Chart is divided into blocks that contain pertinent major steps in the logic operations. Since the Square Root function is a sequential operation that is controlled by the Phase Counter steps, major areas of the logic are designated as to the PC step in which they occur.

The explanation of the Flow Chart is based upon the fact that the Service Technician is already familiar with the functional operations of the EC-130. Most of the logic operations in Square Root function are the same as in other functions of the EC-132. The differences lie in the sequence of operations, and the Phase Counter step in which they occur. Therefore, those operations that are the same and which are already understood will be mentioned briefly, if at all.

SQUARE ROOT - BEGIN

Square Root begins with depression of the Square Root key which closes Square Root reed switch and Common Function reed switch. The reed switches, in turn, initiate KBD, and enable Square Root and CF FF's. DPS set and DPC set to 16's complement of the decimal position are exactly as in other functions.

SQUARE ROOT - PC STEP 1

Display is inhibited by PSZ3 False at A-256 in the Display Logic. This is the same as in the EC-130 except for the gate number. Decimal Align is controlled by O-075 which controls MA1L3 True (Align R1 left).

SQUARE ROOT - PC STEP 2

DPC sets to either 2 or 4, according to whether the Decimal Point switch position is an odd number or an even number. The logic of this setting is explained in Decimal Point Counter logic discussion.

R0 is cleared of data by two operations. First; R0 data from the Delay Line is inhibited from incrementing A Counter by RCRO3 True at O-181. RCRO3 is developed by I-074 controlled by O-073. PC81, PC41, PC22, and PC11 specify PC step 2, and TRO4 specifies R0 time. When all inputs to O-073 are False, output is False and is inverted True by I-074, designated RCRO3 (Clear R0). Second; RCRO3 True makes CRD9 True via O-169. CRD9 True resets D to zero and destroys R4 data, which otherwise would be placed in R1.

DPS is reset during PC step 2 by RCRO3 True at input 1145 on the reset input of DPS FF.

R1 through R4 shift up, which clears R1 for use in the Square Root operation, is controlled by O-072. RSUQ6 False from O-072 makes RSUL3 True from I-129, which in turn enables A to D and D to B shifts from R1 through R4 time (TFG1 on O-072).

SQUARE ROOT - PC STEPS 4 & 6

The net result of PC steps 4 and 6 is to shift R2 left by couplets (2-digit groups). Shift R2 left is controlled in both PC steps by O-068. PC81, PC42, and PC11 will be False during PC steps 4 and 6. When all inputs to O-068 are False, output is False and is inverted True by I-069, designated ML2Q3 (Shift R2 left in Square Root). ML2Q3 True is applied through O-085 to I-086 and Inverted False, designated M2L4. M2L4 False enables O-126. TFE2 and TFF1 False at O-126 specify R2 time, and TCX5 specifies column times. (For TCX5 logic, see Register Expansion discussion.) When all inputs to O-126 are False, the output is False, designated RS2L6 (Shift R2 left). RS2L6 False applied through A-128 to I-129 is inverted True, designated RSUL3 (Shift Up or Left). RSUL3 controls shifts left of R2 through O-150 and O-154.

SQUARE ROOT - PC STEPS 5 & 7

The net result of PC steps 5 and 7 is to shift R1 left by couplets. Shift R1 left is controlled in both PC steps by O-066. PC81, PC42, and PC12 will be False during PC steps 5 and 7. When all inputs to O-066 are False, output is False and is inverted True by I-067, designated ML1Q3 (Shift R1 left in Square Root). ML1Q3 True is applied through O-087 to I-088 and inverted False, designated M1L4. M1L4 False enables O-123. TR14 specifies R1 time, and TCX5 specifies column times. When all inputs to O-123 are False, the output is False, and is designated RS1L6 (Shift R1 left). RS1L6 False is applied through A-128 to I-129 and inverted True, designated RSUL3 (Shift Up or Left). RSUL3 controls shifts left of R1 through O-150 and O-154.

Each shift of R1 left increments DPC by 1 through O-102 at the toggle input of DC1. DSZ3 False specifies DPC not zero, and M1L4 False specifies shift R1 left. When HOME1 goes True at O-102 with all other inputs False, output of O-102 goes False to True and increments DPC by 1.

SQUARE ROOT - PC STEP 8

To produce the desired mathematical result, R0 data is incremented by 1 before subtracting the data from R1. Since a subtract is a complementary add, subtract is accomplished by counting in D Counter to take the complement. The complement is then shifted D to A and R1 is added. Incrementing R0 data (R0 + 1) is simply accomplished by setting Carry FF via O-064, which controls the development of ASCQ3 True (Set Carry in Square Root). Carry FF being set enables AC gate 1303 at set input of D5 FF, which then is set by CRD9 at C1ROB15 time (before C2RO data is counted in D Counter). The net result, then, is that the count in D starts from 9, or 1 is effectively added to R0 data. It is important to remember, however, that R0 data is counted into A Counter and into D Counter simultaneously. When the D to A shift occurs, which places the complemented R0 data in A, an A to B shift is generated which preserves the R0 data unchanged in the data sequence.

Subtract (R0 - 1) from R1 is partially accomplished by counting R0 data in D Counter and shifting D to A. The final step in the complementary addition is accomplished by counting R1 data in A on top of the complemented R0 data.

SQUARE ROOT - PC STEP 9

The subtract cycle in PC step 8 may leave Carry FF set at Home time, which would set COF FF and indicate an unsuccessful subtract (overdraft). If this occurs, the number in R1 must be restored. Therefore, COF set is used as a decision point at the beginning of PC step 9.

If COF is set, PC steps to 14 at the end of PC step 9 via 1172 on PC4 FF set input. This step jump is covered in the Phase Counter discussion. The logic action in PC step 14 is to restore the number by adding R1 to (R0 + 1), which is the opposite of PC step 8 subtract cycle. This addition will be covered in PC step 14 discussion.

If COF is not set, which indicates a successful subtract cycle, the Phase counter steps to 10.

SQUARE ROOT - PC STEP 10

In the discussion of Square Root Mathematics it is pointed out that the subtract cycles are carried out in groups of two, which eliminates the last step of dividing by two. The first subtract cycle adds 1 to R0 data, and then subtracts. The second cycle subtracts R0 directly. R1 - R0 is a straight complementary addition as in a usual subtract problem.

R1 - R0 is controlled, exactly as in PC step 8, by O-056 which develops MSU4. MSU4 then controls a normal complementary add cycle, which is a subtract. The only difference between PC step 8 and PC step 10 is that in step 8 one (1) is effectively added to R0 data as a part of the subtract cycle, and in step 10 no addition of 1 to R0 data takes place.

SQUARE ROOT - PC STEP 11

The subtract cycle in step 10 may leave Carry set at Home time, which would set COF FF and indicate an unsuccessful subtract (overdraft). If this occurs, the number must be restored. Therefore, COF being set is used as a decision point at the beginning of step 11.

If COF is not set this indicates a successful subtract, which is indicated in the data by a direct addition of 1 to R0 data. This is controlled by O-057, which develops RIAQ6 (Increment A Counter in Square Root). RIAQ6 is inverted True by I-139, designated ASA13 (Set A1 FF). ASA13 enables AC gate 1291 on A Counter A1 FF. When A Counter is reset to zero by CRA9 just before R0 data is counted in, A1 FF becomes set which adds 1 to A Counter. Then, when R0 data counts in on top of the 1 in A Counter, effectively 1 is added to R0 data, which indicates the successful subtract.

When COF is not set this indicates not only a successful subtract but a possibility of more successful subtracts. Therefore, the Phase Counter steps to 8 to perform another subtract cycle. Step 11 to 8 is controlled by gate 1176 on PC4 FF and is explained in the Phase Counter discussion.

If COF is set, this indicates an unsuccessful subtract (overdraft), and the number must be restored. This is accomplished in step 12. Step 11 to 12 is controlled by gate 1172 on PC4 FF, and is explained in Phase Counter discussion.

SQUARE ROOT - PC STEP 12

In step 2, DPC becomes set to 2 or 4 according to the Decimal Point switch setting. When DPC counts to zero from the setting in step 2, it indicates that all data has been shifted out of R2. If DPC is counted to zero in PC steps 5 or 7, then DPC sets again for answer align, if not previously set. Thus, DPC zero is a decision point at the beginning of step 12. If DPC is not zero, R0 is added to R1, which restores the number. Add R0 to R1 is controlled by O-049 which controls MAIQ3 True (Add in Square Root). MAIQ3 True results in the inhibit of reset A, as in a normal add.

If DPC is zero and DPS is set at this time, this means that DPC was previously set for answer align. The number in R1 is then restored by the Add R0 to R1 cycle via O-049 as previously explained. If DPS is not set, then the DPC becomes set for answer align and then R1 is restored. The logic of the answer align setting in Square Root is covered in DPC discussion.

SQUARE ROOT - PC STEP 13

There is no logic action in step 13 and the next Home time steps the counter to 14.

SQUARE ROOT - PC STEP 14

As in step 12, DPC zero is a decision point to indicate the necessity for answer align. If DPC is zero, which could occur in steps 5 or 7 by incrementing DPC, then DPC sets for answer align via DPS set as covered in DPC discussion.

PC step jump 9 to 14 was determined by whether COF was set, which indicated an unsuccessful subtract. The number then is restored in step 14 by adding R1 and (R0 + 1). This is the reverse of the operation that occurred in step 8 which produced the overdraft.

(R0 + 1) is controlled by O-079 which controls the development of AIRQ3 True (Increment A Counter in Square Root). At ROB15 time an increment pulse is applied to A Counter which increments the R0 data by 1, thus (R0 + 1).

Add R1 to (R0 + 1) is a normal add which is accomplished by inhibit of Reset A (CRA9 False) at ROB15 time, which leaves R0 data (incremented by 1) in A Counter. Then R1 data counts on top, which adds R1 data to the R0 data left in A Counter. This is controlled, as in PC step 12, by O-049 which develops MAIQ3 True.

Again, DPC zero is a decision point that decides whether to step to PC15 or to shift R0 left and step to PC3, which starts another subtract cycle. If DPC is not zero, DSZ3 will be False at O-125 and enable the gate. ESQ2 specifies Square Root function; PS144 specifies PC step 14; TCX5 specifies column times; and TRO4 specifies R0 time. When all inputs are False, the output is False, designated ROLQ6 (R0 left in Square Root). ROLQ6 through A-128 is inverted True by I-129, designated RSUL3. RSUL3 controls shifts left through O-150 and O-154 for R0 time, which shifts R0 data one column left.

PC step jump 14 to 3 is controlled by O-008. ESQ2 specifies Square Root function; PS144 specifies PC step 14; and DSZ3 False specifies DPC is not zero. When all inputs to O-008 are False, output is False and is inverted True by I-009, designated PJQ33 (PC jump to 3 in Square Root). PJQ33 True enables an AC gate which is common to the reset inputs of PC8 and PC4. When HOME1 at O-030 increments the Phase Counter to PC step 15, the same HOME1 at the AC gate resets PC8 and PC4. The counter ends up with PC8 reset, PC4 reset, PC2 set, and PC1 set, which is PC step 3. This can only occur if DSZ3 is False at O-008, which means DPC is not zero and the arithmetic is not completed.

If DSZ3 is True at O-008, this means that DPC is zero and the arithmetic is complete. HOME1, then, increments the PC to step 15.

SQUARE ROOT - PC STEP 15

In PC step 15 the answer has already been developed by incrementing R0 and answer align. The answer then must be shifted to R1 for display. Shift up of R0 is controlled by O-124 which develops RO1Q6 (R0 to R1 in Square Root). RO1Q6 through A-128 is inverted True by I-129, designated RSUL3. RSUL3 through O-150 and O-154 controls shifts that place R0 data in D Counter. O-280, which develops R1OQ6, develops shifts that place the R0 data from D Counter to R1. The result of O-124 and O-280 together is to shift up R0 data into R1.

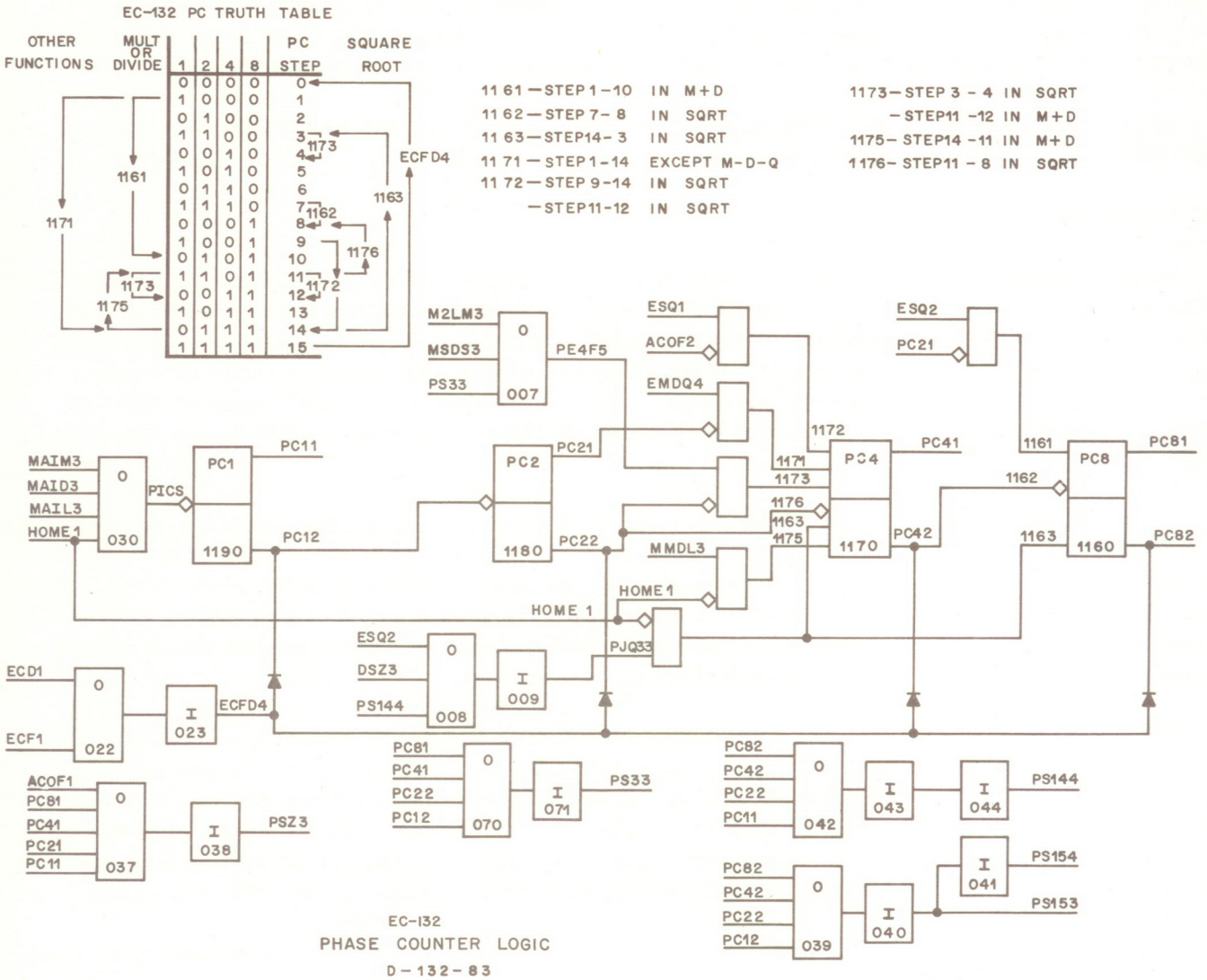
To place significant data in R2 in PC step 15, R4 and R3 are shifted down by one register. Shift down is controlled by O-112, which develops RSDQ6 (Shift down in Square Root). RSDQ6 False through A-113 is inverted True by I-114, designated RESD3 (enable shift down). RESD3 through O-143 and I-144 enables A to C shifts for registers 3 and 4, which shifts R3 and R4 down one. It should be remembered, however, that R4 data is only repeated in R3 and R4 data is not destroyed. (See R4 Data Retention discussion.)

In the EC-132, the only way in which the PC returns to zero is by being clamped to all FF's reset by ECFD4 True. This is accomplished by reset of CF FF at the end of PC step 15. AC gate 1104 at the reset input of CF FF is enabled by PS153 True during PC step 15. HOME1 at this gate at the end of step 15 resets CF, which makes ECF1 False at O-022. ECD1 is also False at O-022. When both inputs to O-022 are False, output is False and is inverted True by I-023, designated ECFD4. ECFD4 True clamps all PC FF's to reset.

When CF FF resets, ECF2 goes True and resets DPS FF, and completes the logic action specified for PC step 15.

SQUARE ROOT - PC STEP 0

Logic action in PC step 0 of Square Root is the same as in all other functions, and is the same as in the EC-130. Reset of function FF's, in this case Square Root FF, is accomplished by ERF3 True. Display is enabled by PSZ3 (Phase Step zero) True at A-256 in the display logic.



EC-132
PHASE COUNTER LOGIC
D-132-83

C. PHASE COUNTER (PC).

The Phase Counter, formerly called the Entry Phase Counter (EPC) in the EC-130, is a group of four FF's connected to give a total binary count of sixteen (16). Reference to the Function Flow Charts in the EC-130 and the Square Root Flow Chart in the EC-132 shows that logic actions are controlled by specific Phase Counter steps. The logic use of the counter settings requires certain step jumps and progressions. These steps and progressions are discussed as follows.

1. UNCLAMP PHASE COUNTER.

During Idle, the Phase Counter is clamped to zero. One of the first actions in any function is to unclamp the counter. This action is exactly the same as in the EC-130 except for the gate numbers. The signal name ECFD4, is the same and the action of ECFD4 in clamping the counter to zero when True and unclamping the counter when False is the same as in the EC-130.

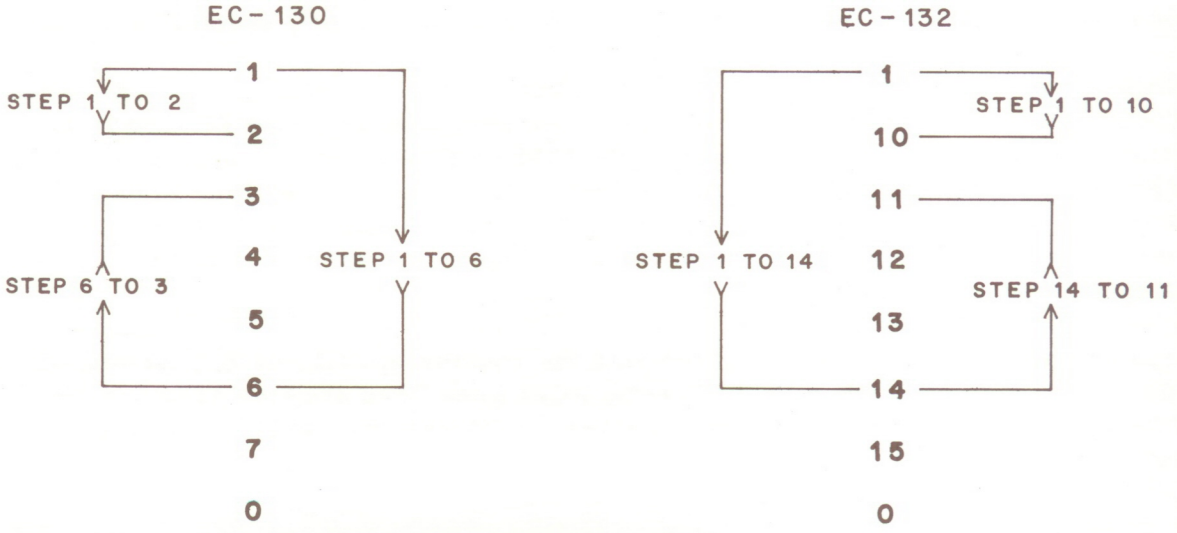
2. INCREMENT PHASE COUNTER.

The Phase Counter in the EC-132 is incremented in the same manner as in the EC-130 except for the gate number that develops PIC5, which in the EC-132 is 0-030. The same logic signals are also used.

3. FUNCTIONS COMMON TO THE EC-130/132.

Because the EC-130 and EC-132 have most functions in common, and the logic of these functions is the same in each machine, the operational sequence is the same. The only differences will be that a given operation will be in a different Phase Counter step. The diagram illustrates the correlation between similar steps in the EC-130 and EC-132.

CROSS-CORRELATION OF EPC STEPS
ALL FUNCTIONS EXCEPT SQUARE ROOT



Since the incrementing and step jumping of the Phase Counter in the EC-132 is not the same as in the EC-130, the similar steps and jumps are explained as follows.

a. PHASE COUNTER STEP 1 to 10. In a Multiply or Divide function it takes six counter steps to complete the function. Since there are 16 steps in the EC-132, in a Multiply or Divide function the counter progression jumps from 1 to 10, which provides six (6) further steps (step 10 to step 15) to complete the function. This step jump corresponds with the counter progression from Step 1 to Step 2 in the EC-130.

Phase Counter step 1 to 10 is controlled by gate 1161 on PC8 FF set input. This is an AC gate with ESQ2, which will be True in any function except Square Root, as the enable and PC21 True as the transition signal. When PC21 goes False to True at AC gate 1161 enabled by ESQ2 True, PC8 FF sets. Therefore, when the counter steps from 1 to 2, this results in PC1 reset, PC2 set, PC4 reset, and PC8 set. This is the condition for step 10.

b. PHASE COUNTER STEP 14 to 11. Logic in a Multiply or Divide function requires a return to a previous step under certain conditions. In the EC-130 this step is from 6 to 3. In the EC-132 this corresponding return step is from 14 to 11.

Step 14 to 11 is controlled by gate 1175 on PC4 reset input. This is an AC gate with MMDL3, which will be True in a Multiply or Divide function, as the enable signal and HOME1 as the transition signal. At the end of step 14, HOME1 at 0-030 makes PIC5 True which increments the counter to PC1 set, PC2 set, PC4 set, and PC8 set. This same HOME1, however, at gate 1175 resets PC4. The new condition, then, of the counter is PC1 set, PC2 set, PC4 reset, and PC8 set, which is step 11.

c. PHASE COUNTER STEP 11 to 12. The design of the counter prevents the usual progression from Step 11 to Step 12. In a Multiply or Divide function this progression is controlled by gate 1173 on PC4 set input. This is an AC gate which is enabled by PE4F5 True. PE4F5 is developed by 0-007. In a Multiply function, M2LM3 True at 0-007 makes PE4F5 True at 1173. In a Divide function, MSDS3 True at 0-007 makes PE4F5 True at 1173. At the end of step 11, PIC5 increments the PC to PC1 reset, which resets PC2. When PC2 resets, PC22 goes True at gate 1173 and sets PC4. The new condition, then, of the counter is with PC1 reset, PC2 reset, PC4 set, and PC8 set, which is step 12.

d. PHASE COUNTER STEP 1 to 14. In all functions other than Multiply, Divide, or Square Root the Phase Counter step is from 1 to 14. This corresponds in the EC-130 to the step from 1 to 6. Step 1 to 14 is controlled by gate 1171 on PC4 FF set input and gate 1161 on PC8 FF set input. AC gate 1171 is enabled by EMDQ4 which will be True in any function except Multiply, Divide, or Square Root. AC gate 1161 is enabled by ESQ2 which will be True in any function except Square Root. PC21, which is the set output of PC2 FF, is the transition signal for both gates.

At the end of step 1, PIC5 True increments the counter to PC1 reset, PC2 set, with PC4 and PC8 reset. However, when PC2 FF sets, PC21 goes True at gates 1171 and 1161 and sets PC4 and PC8 FF's. The new condition, then, of the counter is PC1 reset, PC2 set, PC4 set, and PC8 set, which is step 14.

4. SQUARE ROOT FUNCTION.

The Square Root function is relatively more complex than any of the other functions and utilizes most of the available counter steps, as well as having more step jumps and return steps.

a. PHASE COUNTER STEP 3 to 4. The design of the counter prevents a usual progression from step 3 to 4. In the Square Root function, step 3 to 4 is controlled by gate 1173 on PC4 FF set input. This is an AC gate which is enabled by PE4F5. In Square Root, which is the only function utilizing step 3, PS33 will be True at 0-007, which makes PE4F5 True at 1173 at Phase Counter step 3 in Square Root. PIC5 True at the end of step 3 increments the counter to PC1 reset and PC2 reset. However, when PC2 resets, PC22 goes True at gate 1173 and sets PC4 FF. The new condition, then, of the counter is PC1 reset, PC2 reset, PC4 set, and PC8 reset, which is step 4.

b. PHASE COUNTER STEP 11 to 12, OR 11 to 8. In the logic of the Square Root function there is a decision at the end of step 11 to step to 12, or to step to 8. The decision is made by the state of Carry Overflow FF. At the end of step 11, PIC5 True increments the counter to PC1 reset, PC2 reset, PC4 reset, and PC8 set, which is step 8. However, if Carry Overflow FF became set at the end of step 10 by the arithmetic of the problem, then the same HOME1 that increments the counter at the end of step 11 also resets Carry Overflow FF, which makes ACOF2 transition False to True at AC gate 1172. Gate 1172 at PC4 FF set input is enabled by ESQ1, which will be True in Square Root function. When ACOF2 goes True, PC4 FF sets. The new condition, then, of the counter is PC1 reset, PC2 reset, PC4 set, and PC8 set, which is step 12.

In summary, if the Carry Overflow FF does not reset at the end of step 11, the counter ends up set to 8. If Carry Overflow FF resets at the end of step 11, ACOF2 False to True transition sets PC4 FF via gate 1172, and the counter ends up set to 12.

c. PHASE COUNTER STEP 7 to 8. Step 7 to 8 progression is controlled by gate 1162 on PC8 FF set input. At the end of step 7, PIC5 True increments the counter to PC1 reset, PC2 reset. When PC2 resets PC22 True resets PC4 FF via gate 1176. When PC4 FF resets, PC42 True sets PC8 FF via gate 1162. The new condition, then, of the counter is PC1 reset, PC2 reset, PC4 reset, and PC8 set, which is step 8.

d. PHASE COUNTER STEP 9 to 14. The logic of the Square Root function may require the counter to step to 14 instead of 10. The decision is made by the state of Carry Overflow FF. At the end of step 9, PIC5 True increments the counter to PC1 reset, PC2 set, PC4 reset, and PC8 set, which is step 10. However, if Carry Overflow FF became set at the end of step 8 by the arithmetic of the problem, then the same HOME1 that increments the counter at the end of step 9 also resets Carry Overflow FF, which makes ACOF2 transition False to True at AC gate 1172 on PC4 set input. Gate 1172 is enabled by ESQ1, which will be True in a Square Root function. When ACOF2 goes True, PC4 FF sets. The new condition, then, of the counter is PC1 reset, PC2 set, PC4 set, and PC8 set, which is step 14.

e. PHASE COUNTER STEP 14 to 3. The logic of the Square Root function may require the Phase Counter to step to 3 instead of to 15. The decision is made by whether DSZ3 (Decimal Counter Zero) is True or False at 0-008 at the end of step 14. ESQ2 will be False at 0-008 during Square Root function, and PS144 will be False during step 14. When all inputs to 0-008 are False, output is False and is inverted True by I-009, designated PJQ33 (PC jump to 3 in Square Root). PIC5 True at the end of step 14 increments the counter to PC1 set, PC2 set, PC4 set, and PC8 set, which is step 15. However, if DSZ3 is False at 0-008, which means the Decimal Counter is not zero, PJQ33 is True at AC gate 1163 and HOME1 resets PC4 and PC8. The new condition, then, of the counter is PC1 set, PC2 set, PC4 reset, and PC8 reset, which is step 3.

f. PHASE COUNTER STEP 15 to ZERO. Due to the design of the Phase Counter in the EC-132 there is no way to increment the counter from 15 to 0 as is done in the EC-130. In the EC-132, the counter returns to 0 by being clamped to all FF's reset, by ECFD4 True.

ECFD4 True, which clamps the Phase Counter FF's to reset, is developed by I-023 controlled by O-022. ECD1 and ECF1 will both be False at O-022 if CF and CD FF's are reset. At the end of step 15, CF or CD FF's (whichever is set) will be reset via AC gate 1104, which is common to the reset inputs of both FF's. Gate 1104 is enabled by PS153, which will be True during step 15. At the end of step 15, HOME1 True at 1104 resets either CF or CD FF's, whichever was set, which makes both ECF1 and ECD1 False at O-022. When both inputs to O-022 are False, output is False, and inverted True by I-023 makes ECFD4 True, and clamps all counter FF's to reset, which is Phase Counter step Zero.

5. PHASE COUNTER LOGIC OUTPUTS.

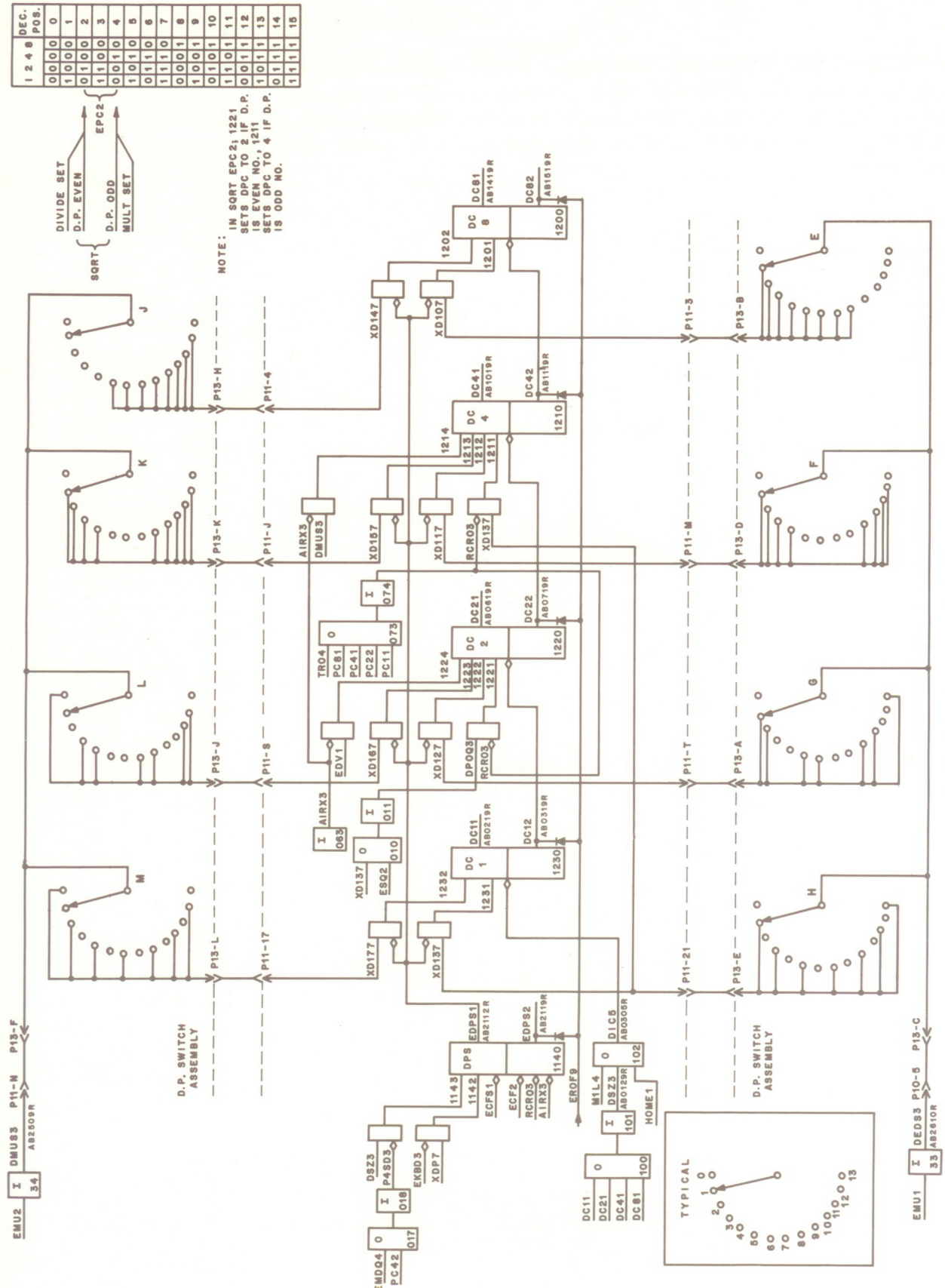
Certain Phase Counter steps are developed separately by logic gates controlling inverters to give necessary logic polarities. They are explained as follows.

a. PHASE COUNTER STEP ZERO. The signal which defines step Zero, which is all PC FF's reset, is developed by I-038 controlled by O-037. PC81, PC41, PC21, and PC11 will be False at O-037 when all counter FF's are reset. ACOF1 will be False at O-037 except when complementing R1 after a Subtract function. When all inputs to O-037 are False, output is False, and is inverted True by I-038, designated PSZ3 (Phase Step Zero, True).

b. PHASE COUNTER STEP 3. The signal which defines step 3 is developed by I-071 controlled by O-070. PC81, PC41, PC22, and PC12 will all be False at O-070 during step 3. When all inputs to O-070 are False, output is False and is inverted True by I-071, designated PS33 (Phase Step 3, True).

c. PHASE COUNTER STEP 14. The signal which defines step 14 is developed by O-042. PC82, PC42, PC22, and PC11 will all be False at O-042 during PC step 14. When all inputs to O-042 are False, output is False, and is inverted True by I-043. The output of I-043 is inverted False by I-044, designated PS144 (Phase Step 14, False).

d. PHASE COUNTER STEP 15. The signal which defines step 15 is developed by O-039. PC82, PC42, PC22, and PC12 will all be False at O-039 during step 15. When all inputs to O-039 are False, output is False and is inverted True by I-040, designated PS153 (Phase Step 15, True). The output of I-040 is inverted False by I-041, designated PS154 (Phase Step 15, False).



EC-132 DECIMAL POINT COUNTER
D-132-82

D. DECIMAL POINT COUNTER (DPC) LOGIC ANALYSIS

The DPC is a group of four (4) FF's connected to give a binary count of sixteen (16). While the DPC is capable of 16 counts (0 through 15), not all of the possible counts are used. The DPC is "preset" to specific numbers in binary form and then counts up to 0 from that setting.

Certain logic operations in the EC-132 are controlled by whether DPC is zero (DCZ3 True), and if not, how many counts it takes for DPC to reach zero. They are Decimal Align, Answer Align, Multiply preset, Divide preset, and Square Root preset. Increment of DPC and each of the presets are discussed separately as follows:

1. INCREMENT DPC

The advance of DPC is controlled by O-102. The logic construction of O-102 specifies that when all inputs are False and then one of the inputs goes True, the output of O-102 transitions from False to True, designated DIC5 (Increment DPC, True). DIC5 applied to the T-input of DPC1 FF causes DPC1 to change state from set to reset, or vice-versa. Each change of state of DPC1 FF causes a change in the DPC setting that corresponds to a binary count of 1 added to the previous DPC setting, or, to state it differently, DPC is counted up by one.

2. DECIMAL ALIGN PRESET

When the Decimal Point switch is on a specific position, for instance 2, then certain of the AC gates at the set inputs of all DPC FF's are enabled through Decimal Point switch wafers E, F, G, and H. Then, when a positive transition is applied to the capacitor input of these AC gates, the enabled gates set the associated DPC FF's. Once the FF's are set, then the advance, or count up, of DPC starts from this preset condition.

When the Decimal Position switch is on 1, the decimal point position indicates one column to the right of the decimal point, meaning that the whole number portion of the entered number must be shifted one column left to be in proper decimal position. These shifts left are controlled by DPC being set to 15, which means that one more count into DPC will bring DPC to zero. As illustrated, when the Decimal Point switch is on 1, DPC8, DPC4, DPC2, and DPC1 set input AC gates are enabled by active connections on the E, F, G and H switch wafers.

The enable signal for Decimal Align is developed by I-033, designated DEDS3 (Entry or Divide set, True). The input to I-033 is the set output (EMU1) of Mult. FF, which will be False until Mult. FF is set. EMU1 False is inverted True by I-033 (DEDS3) and applied to the E, F, G, and H switch wafers and enables those DPC AC gates actively connected through the wafers.

The signal that sets the DPC FF's that are enabled is the False to True transition of DPS FF set output (EDPS1), which occurs when DPS goes from a reset to a set condition. The logic of the setting of DPS FF will be covered as applicable in the Function Logic discussions.

3. MULTIPLY FUNCTION PRESET

Certain logic steps in a Multiply function are controlled by whether DPC is at zero, and, if not at zero, how many counts it takes to bring DPC to zero. In Multiply, the next set of DPC after Decimal Align is to a preset condition of 4, which means 12 more counts to bring DPC to zero.

This is accomplished by an AC gate at the set input of DPC 4 FF that is enabled by the reset output (EMU2) of Mult. FF inverted True by I-034, designated DMUS3. The output of I-063, designated AIRX3, goes False to True at EPC step 2 of Multiply (and Divide) and this transition is used at the enabled AC gate of DPC4 FF to set DPC4. Since DPC was counted to zero in EPC step 1 of Multiply (Decimal Align), the new condition of DPC with DPC4 set and all other FF's reset specifies a preset count of 4 in DPC, and it will take 12 more counts into DPC to bring the counter to zero.

4. DIVIDE FUNCTION PRESET

As in Multiply preset, certain Divide logic steps are controlled by DPC. In Divide, however, the preset is to 2 instead of 4, which means 14 more counts into DPC to bring the counter to zero.

The preset to 2 is accomplished by an AC gate at set input of DPC2 FF which is enabled by set output (EDV1) of Divide FF, which will be True in Divide function. False to True transition of AIRX3 provides the transition in Divide that sets DPC2 FF. Since DPC was counted to zero in Divide EPC step 1 (Decimal Align), the new condition of DPC with DPC2 set and all other FF's reset specifies a preset count of 2 in DPC, and it will take 14 more counts into DPC to bring the counter to zero.

5. SQUARE ROOT PRESET

The DPC setting in PC step 2 specifies the number of register columns to be used in the arithmetic of the problem. In Square Root function, either of two settings is possible and is determined by whether the decimal point column position is an odd number or an even number. Square Root arithmetic is accomplished by couplets (two-digit groups) and, in order to be successful, the digits spaces to the left of the decimal position must be capable of being divided equally into couplets.

If the column position of the decimal point is an odd number, there is an even number of columns to the left of the decimal point. For example, if the decimal position is 7, there are 6 columns to the left of the decimal point (13 - 7 6). If the column position of the decimal point is an even number, there is an odd number of columns to the left of the decimal point. For example, if the decimal position is 2, there are 11 columns to the left of the decimal point (13 - 2 11).

If the decimal position is an odd number, there is an even number of columns to the left of the decimal to be divided into two-digit couplets, and the DPC sets to 4, which means 12 more counts will bring DPC to zero.

If the decimal position is an even number, there is an odd number of columns to the left of the decimal to be divided into two-digit couplets, and the DPC sets to 2, which means 14 more counts will bring DPC to zero.

a. DPC SET TO 2 (DECIMAL POSITION EVEN). DPC set to 2, which is the 16's complement of 14, is controlled by AC gate 1221 at set input of DPC2 FF. DPC is counted to 0 during decimal align in PC step 1 and will be 0 at the beginning of PC step 2. Therefore, if DPC2 FF is set with all others remaining reset, then DPC is set to a count of 2.

AC gate 1221 is enabled by DPOQ3 True, and DPOQ3 will be True if the decimal point position is even in Square Root. DPOQ3 is developed by I-011 controlled by O-010. ESQ2 will be False at O-010 because the Square Root FF is set, and the XD137 input will be open (False) because there is no active connection at Decimal Switch wafer H. When both inputs to O-010 are False, the output is False and is inverted True by I-011, designated DPOQ3. Now, when RCRO3 goes True, DPC2 FF sets. RCRO3 is developed by I-074 which is controlled by O-073. TRO4 input to O-073 will be False at RO time, and PC81, PC41, PC22, PC11 delineate EPC step 2. Therefore, at the first RO time in EPC step 2, all inputs to O-073 are False, the output is False, and is inverted True by I-074, designated RCRO3. RCRO3 going False to True at AC gate 1221 sets DPC2 FF if DPOQ3 is True.

b. DPC SET TO 4 (DECIMAL POSITION ODD). DPC set to 4, which is the 16's complement of 12, is controlled by AC gate 1211 at the set input of DPC4 FF. Since DPC is counted to 0 by the decimal align in EPC step 1, if DPC4 FF is set in DPC step 2 with all other FF's remaining reset, then DPC is set to a count of 4.

AC gate 1211 is enabled by XD137 True through active connections on Decimal Point switch wafer H when the decimal position is odd. Now, when RCRO3 goes True at the first RO time in PC step 2 (as described in DPC set to 2), DPC4 FF sets with all other FF's remaining reset and DPC is set to 4.

6. ANSWER ALIGN - MULTIPLY PRESET

In Multiply function, after the logic has developed the arithmetic answer, the number must be properly decimal aligned in R2. This is accomplished by setting the DPC to the 16's complement of the number of places to the left of the decimal position. For instance, if the column position of the decimal point indicates two places, there are 11 columns to the left of the decimal point and DPC sets to 5. The actual number set into DPC can easily be found by adding 3 to the decimal point position; therefore, this method is indicated on the Flow Chart.

Certain AC gates on the DPC FF's are enabled through switch wafers J, K, L, and M. The enable signal is developed by the reset output (EMU2) of Mult. FF inverted True by I-034, designated DMUS3. The transition signal is the set output (EDPS1) of DPS FF. In PC step 1 of Multiply (and Divide) DPS becomes reset. In PC step 4 of Multiply (and Divide) DPS is set again by the action of O-017. EMDQ4 is False at O-017 because the function is Multiply, and PC42 will go False at PC step 4. True to False output transition of O-017 is inverted False to True by I-018, designated P4SD3, and applied to an AC gate at DPS FF set input. The enable on this gate is DPC zero (DSZ3), which will not be True until DPC has counted to zero from the preset count established in EPC step 2 of Multiply.

When this DPS FF AC gate is enabled by DSZ3 being True, then the next P4SD3 True transition sets DPS, which in turn sets DPC for decimal alignment of the answer according to the position of the Decimal Point switch.

7. ANSWER ALIGN - DIVIDE FUNCTION

In Divide function, the logic action that sets DPC for decimal alignment is very similar to the action in a Multiply function. In Divide, the enable signal applied to the DPC AC gates is EMU1 False inverted True by I-033, designated DEDS3. DEDS3 True is applied through switch wafers E, F, G, and H, and enables DPC FF gates as specified by active connections on the switch wafers for a given decimal location. The transition signal that sets the DPC FF's is DPS set output (EDPS1) which transitions False to True. The logic of the DPS setting will be covered as applicable in Divide Function discussion.

8. SQUARE ROOT ANSWER ALIGN (PC 12 or 14)

In Square Root, the logic action that sets DPC for answer align is similar to that in the Divide function. The enable signal on the DPC FF AC gates is DEDS3 through the Decimal Point switch wafers E, F, G, and H.

The set of DPS, which is the source of the transition signal (EDPS1) that sets the enabled DPC FF's is controlled by AC gate 1143 on set input of DPS FF. The enable on this gate is DSZ3, which will be True when DPC is zero. At PC step 12 or 14, if DSZ3 is True, which means the arithmetic of the problem is complete, P4SD3 True transition sets DPS which sets DPC for answer alignment.

P4SD3 is developed by I-018 controlled by O-017. EMDQ4 will be False at O-017 during Square Root function. PC42 at O-017 is the reset output of EPC4 FF and will be False when EPC4 FF is set. PC4 FF becomes set as a part of binary progression from PC step 11 to 12 or from step 9 to 14, which are the two possible progressions in Square Root. When PC42 goes False both inputs to O-017 are False, the output is False, and is inverted True by I-018, designated P4SD3 (Phase step 4, set decimal, True). P4SD3 False to True transition sets the DPS FF. DPS setting makes EDPS1 go False to True and this signal at enabled AC gates on the DPC FF's sets the DPC to the 16's complement of the decimal position. Therefore, DPS becomes set in Square Root for an answer alignment in PC step 12 or step 14, depending upon when DPC becomes counted to 0.

E. REGISTER EXPANSION

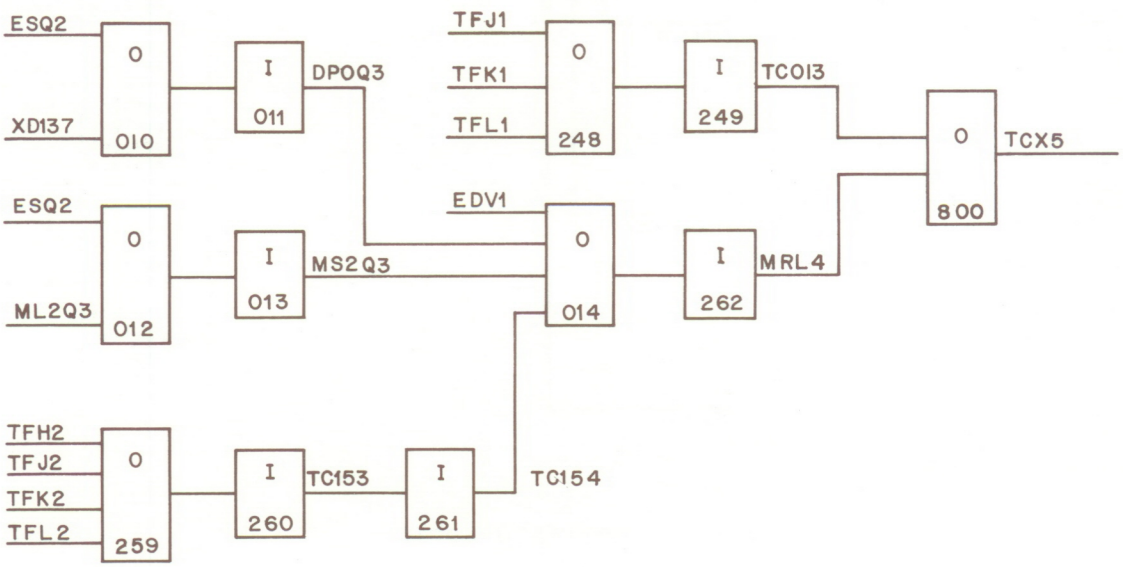
Register expansion to 14 columns is controlled by MRL4 at O-800. The output of O-800, designated TCX5, when False is used to enable OR gates. TCX5, when True, inhibits these gates. TCX5 will be True when either input to O-800 is True. TCO13 will be True for C0 and C1 time, thus TCX5 will be True for these times and any OR gate where TCX5 is used as the enable signal will be inhibited.

MRL4 input to O-800 is developed by I-262 controlled by O-014. If any input to O-014 is True, MRL4 will be False and is used through O-800 as TCX5 False to enable OR gates.

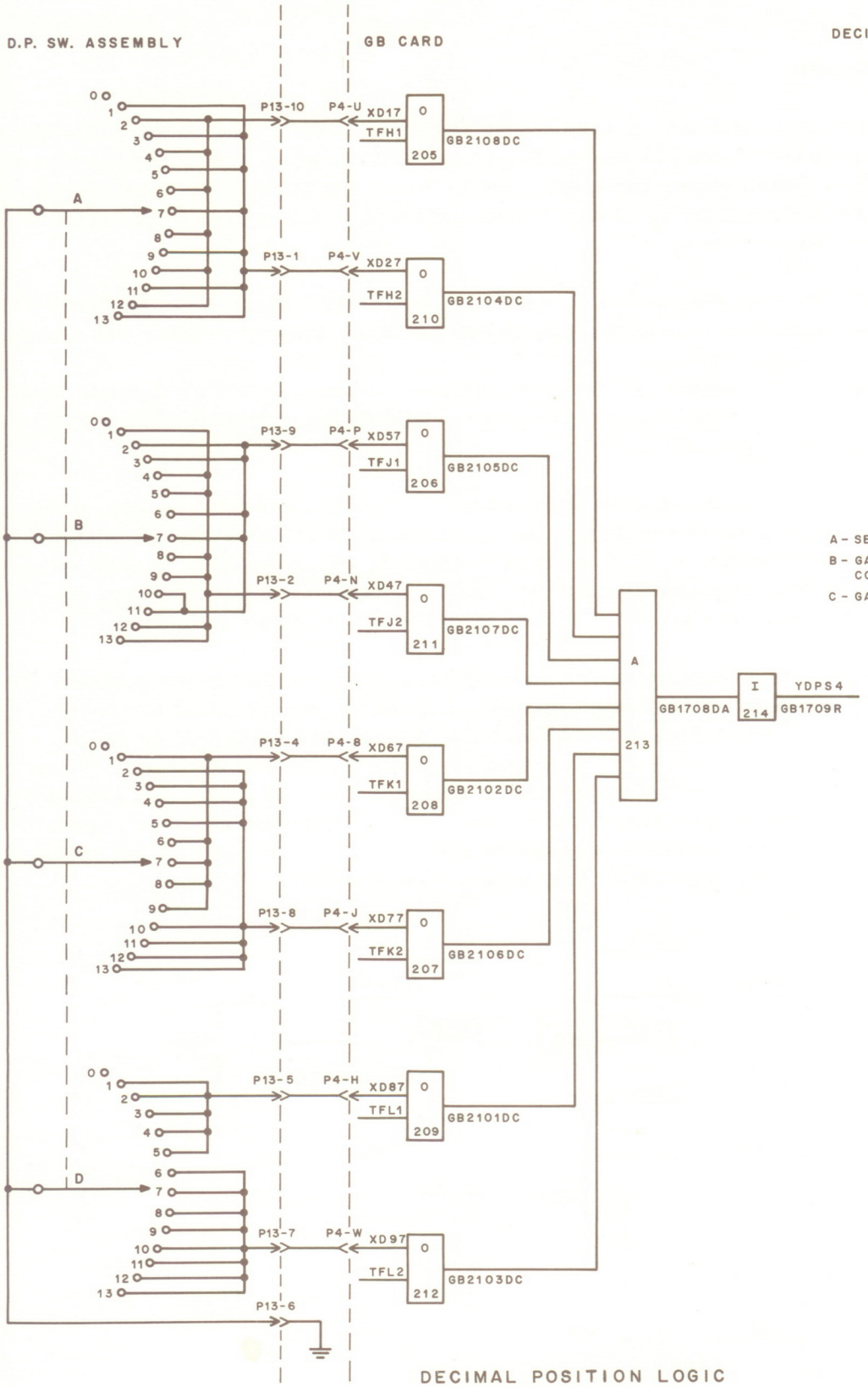
In a Divide problem, the arithmetic necessitates register expansion to 14 columns. This is accomplished by EDV1 True at O-014, which makes MRL4 False at O-800, thus TCX5 False and register expansion occurs.

When the Decimal Point position is an even number in Square Root function, DPC sets to 2 via the action of DPOQ3 True in the DPC. DPOQ3 True at O-014 also makes MRL4 False, thus making TCX5 False during C15 R2 time. Register expansion thus occurs when DPC sets to 2 and the Decimal Position is an even number. Register expansion does not occur when the Decimal Position is an odd number, thus DPOQ3 is False at O-014.

In any case, R1 and R0 are always expanded to 14 columns to provide for the subtract cycle that produces the last digit of a 13-digit answer. Expansion is controlled by O-012, which is enabled by ESQ2 False in Square Root function. ML2Q3 at O-012 will be False except when shifting R2 left in Square Root, thus will be False for R1 and R0 times. When both inputs to O-012 are False, output is False and is inverted True by I-013, designated MS2Q3. MS2Q3 True at O-014 makes MRL4 False, thus TCX5 False and register expansion occurs for R1 and R0. Of course, register expansion also occurs for R3 and R4 but since shifts left do not occur for these registers, it is of no consequence.



REGISTER EXPANSION LOGIC



DECIMAL POSITION LOGIC

D-132-70

DECIMAL POSITION COUNTER
TRUTH TABLE

DPS POS.	H	J	K	L	
	0	0	0	0	C0
	1	0	0	0	C1
	0	1	0	0	C2
1	1	1	0	0	C3
2	0	0	1	0	C4
3	1	0	1	0	C5
4	0	1	1	0	C6
5	1	1	1	0	C7
6	0	0	0	1	C8
7	1	0	0	1	C9
8	0	1	0	1	C10
9	1	1	0	1	C11
10	0	0	1	1	C12
11	1	0	1	1	C13
12	0	1	1	1	C14
13	1	1	1	1	C15

A - SELECTED DECIMAL POSITION
B - GATES TRUE BY SWITCH CONTACTS
C - GATES TRUE BY TIMING SIGNALS

A	B	C	A	B	C
1	210 211 208 209	205 206 207 212	8	205 211 208 212	210 206 207 209
2	205 206 207 209	210 211 208 212	9	210 211 208 212	205 206 207 209
3	210 206 207 209	205 211 208 212	10	205 206 207 212	210 211 208 209
4	205 211 207 209	210 206 208 212	11	210 206 207 212	205 211 208 209
5	210 211 207 209	205 206 208 212	12	205 211 207 212	210 206 208 209
6	205 206 208 212	210 211 207 209	13	210 211 207 212	205 206 208 209
7	210 206 208 212	205 211 207 209			

F. DECIMAL POINT POSITION LOGIC.

The decimal point position logic specifies the column in which the decimal point will appear. YBDP3, which is the decimal point unblanking input to the Matrix, is developed by I-232 controlled by O-231. YBDP3 True enables decimal point unblanking, therefore the input to I-232 must be False for unblanking, and O-231 acts as a negative AND.

1. DECIMAL POINT SWITCH LOGIC

YDPS4 False at O-231 is developed by I-214 controlled by the Decimal Point Switch Logic. The input to I-214 is the output of A-213, and when A-213 inputs are all True the output is True. The inputs to A-213 are a group of eight OR gates which have two (2) inputs each. All OR gate outputs must be True for A-213 output to be True. However, only one input to each OR gate needs to be True for the output of that OR gate to be True.

The action of the decimal point switch is to provide a True input to four of the OR gates, which will make the output of those gates True. The action of H, J, K, and L FF outputs is to provide a True input to the other four OR gates at a specific column time. By the combined action of the decimal point switch and the FF outputs, the inputs to A-213 will be True only for one specific column time in each register. Assume that the Decimal Point switch is on position 5. This specifies a decimal point in column 6 (C6). C7 is the time that C6 information is in the D Counter for display decoding, and C6 is the column in which the decimal point is displayed when 5 decimal places to the right of the decimal point are specified. Ground (True, 0V) is applied through active switch wafer connections (see table) to O-210, O-211, O-207, and O-209, which makes the output of these gates True at A-213. TFH1 at O-205 will be True for C7 time (See Truth Table), which makes the output of O-205 True at A-213 for C7 time. TFJ1, TFK1, and TFL2 are True for C7 time and make O-206, O-208, O-212 outputs True at A-213 for C7 time. Thus, all inputs to A-213 will be True for C7 time, and only for C7 time. Therefore, A-213 True output, inverted False by I-214, designated YDPS4, will enable O-231 for C7 time.

For another example, assume that the Decimal Point switch is on 13, which specifies 13 digit spaces available to the right of the decimal point. Ground (True) is applied through active switch wafer connections to O-210, O-211, O-207, and O-212, which makes these gate outputs True at A-213. TFH1, TFJ1, TFK1, and TFL1 True for C15 time at O-205, O-206, O-208, and O-209, makes these gate outputs True at A-213. Thus all inputs to A-213 are True for C15 time, and only for C15 time. Therefore, A-213 True output inverted False by I-214, designated YDPS4, enables O-231 for C15 time, which is when C14 data is in D Counter for decoding.

By reference to the decimal point switch wafer connections and the Decimal Point Position Truth Table, the OR gate inputs necessary for True outputs at A-213 can be determined for each specific decimal point position. Note, however, that no decimal point display is permitted for switch position 0. Remembering that all inputs to A-213 must be True for decimal point display, O-211 output, for instance, will be False because of TFJ2 False input and no active connection through a switch wafer. Thus, A-213 will be inhibited by O-211. Of course, at this time, O-205 is False because of TFH1 False input; O-208 output is False because of TFK1 False input; and O-209 output is False because of TFL1 False

input. However, any one of the gate outputs being False at A-213 inhibits decimal point display, so if more than one is False, it is of no consequence.

2. DIGIT ENTRY DECIMAL POINT BLANKING

YR1D3 input to O-231 will be False except during digit selection, when the decimal point in R1 is kept blanked until the number is entered by the initiation of a function. YR1D3 is developed by I-230 controlled by O-229 which acts as a negative AND. TFM1 and TFN1 specify R1 display time, and will be False for R1 time. ECFS1 specifies a function, and will be False when the last operation was a digit entry. If all inputs to O-229 are False, and they will be False only during a digit entry, the output of O-229 is False, and inverted True by I-230 acts to inhibit decimal point display for Register one (R1) only. During any other functions, ECFS1 will be True, thus YR1D3 will be False, and O-231 will be enabled for decimal point display.

3. DECIMAL POINT UNBLANKING

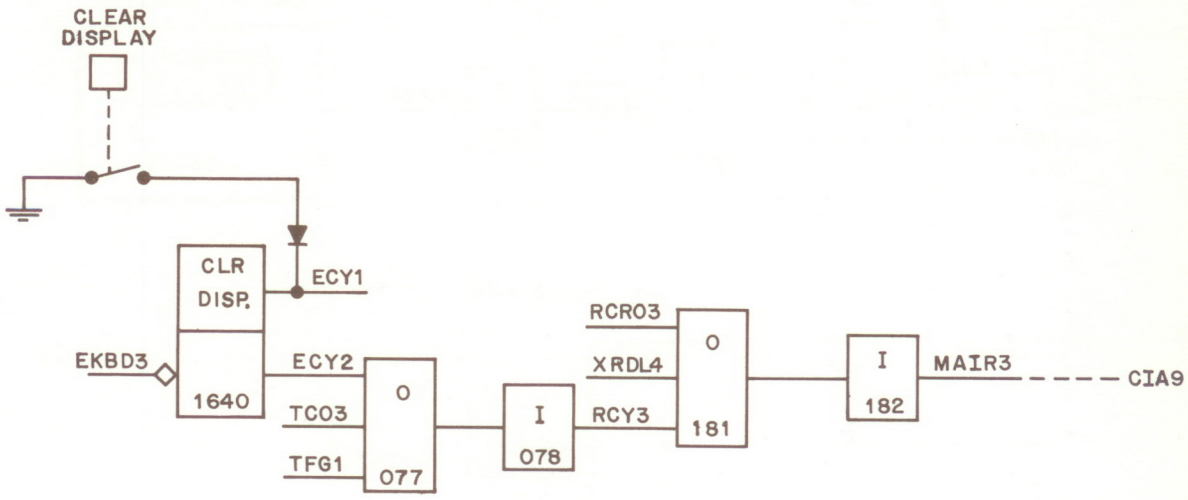
Decimal Point on the display is developed by unblanking segment 13. TFG2, TFD2, and TFE2 will be False at O-231 for segment 13 time only. Therefore, segment 13 is enabled to be unblanked by action of O-231. In summary, the decimal point switch position together with H, J, K, and L FF settings determines the column position of the decimal point; YR1D3 permits disabling the decimal point display for R1 time during a number entry; and TFG2, TFD2, and TFE1 inputs to O-231 define segment 13, which is the decimal point display segment. O-231 acts as a negative AND, and when all inputs are False, the output of O-231 is False. O-231 False output, inverted True by I-232, designated YBDP3, is applied in the matrix to provide unblanking for segment 13.

G. CLEAR DISPLAY LOGIC ANALYSIS

The Clear Display function in the EC-132 corresponds to the Clear All function in the EC-130 with the exception that only Registers 1 through 4 (R1-R4) are cleared, and RS (Store) register is not affected.

Basically, clearing R1 through R4 is accomplished by inhibiting increment A from the delay line for Registers 1 through 4. When the Clear Display key is depressed, 0V (True) is applied through an isolation diode to the set output of Clear Display FF, and places the FF in a set condition. Clear Display set makes ECY2 reset output False. ECY2 False enables O-077. TFG1 at O-077 will be False from R1 through R4 time. TCO3 at O-077 will be False except during C0 time. When all inputs to O-077 are False, output is False and is inverted True by I-078, designated RCY3. RCY3 True inhibits O-181, which is the controlling gate that, when not inhibited, enables XRDL4 (Read Amp output) to increment A Counter.

After the Clear Display function, when any other function at the keyboard initiates Keyboard Delay, EKBD3 True resets Clear Display FF and the Clear Display function is completed.

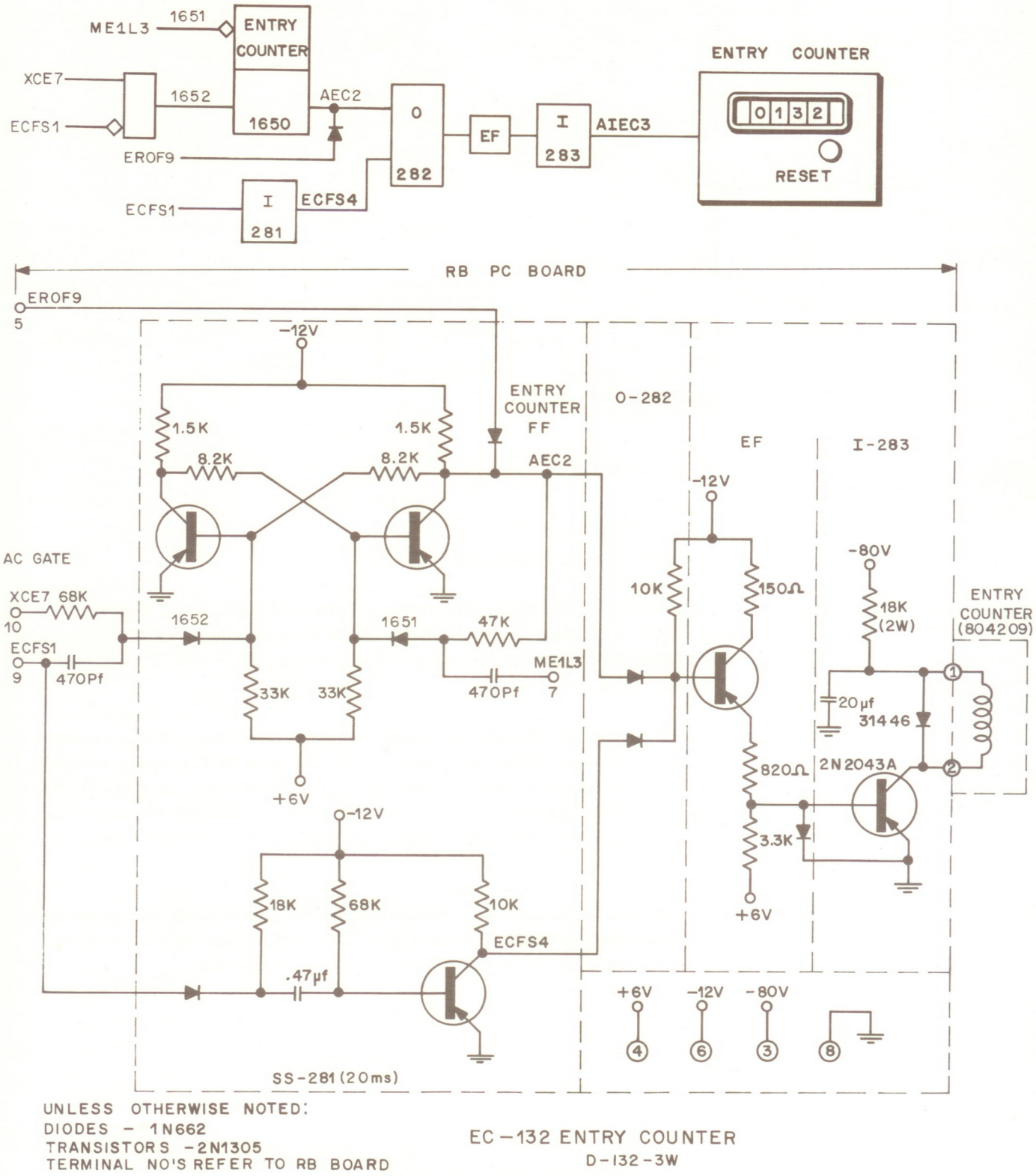


H. REGISTER 4 (R4) RETENTION LOGIC ANALYSIS

A shift down operation in the EC-130 and EC-132 involves shifting specified register data A to C instead of A to B which, in effect, advances the data by one register and displays the data one register sooner on the display. At the same time, in the EC-130, B to C shift is inhibited and D to B shift is enabled.

In the EC-130, the shift D to B, while it occurs for each register shifted A to C, is used primarily for placing zeros in R4 in a shift down operation. At the end of R4 time, R4 data is shifted A to C and the signal A to B is generated. The A to C shift places R4 data in R3 time, and the A to B shift preserves the R4 data. Since B to C shift is not inhibited for RS time, which directly follows R4 time, the new R4 data (all zeros) will shift B to C and appear in R4 time.

In the EC-132, retention of R4 data is easily accomplished by NOT enabling the D to B shift, with the A to C shift enabled and the B to C shift inhibited. Thus, R4 data is not replaced with zeros in B counter and R4 data is retained in the data sequence.



UNLESS OTHERWISE NOTED:
DIODES - 1N662
TRANSISTORS - 2N1305
TERMINAL NO'S REFER TO RB BOARD

EC-132 ENTRY COUNTER
D-132-3W

I. ENTRY COUNTER

The Entry Counter, which will be optional for early production machines, advances by one for each complete number entry into the EC-132 whether completed by depressing ENTER key or the initiation of a function directly after digits have been placed in Register 1. The counting is accomplished by a mechanical counter.

In accounting, an entry is considered to be the recording of a complete number in proper decimal alignment. A number is one or more digits. A digit is a part of a number. In the EC-132, a number is placed in the machine digit-by-digit and then decimal aligned as a part of a function, or by ENTER key. The Entry Counter circuits register only the operation of decimal alignment, which completes the number as a valid entry.

1. ENTRY COUNTER LOGIC

The actual mechanical count in the Impulse Counter is controlled by I-283, whose output is designated AIEC3. I-283 is controlled through an emitter-follower by O-282. The two inputs to O-282 are AEC2 and ECFS4. When both inputs to O-282 are False, the output is False and, inverted True by I-283, increments (advances by one) the Entry Counter.

AEC2 at O-282 is the reset output of Entry Counter FF, and will be False when the Entry Counter FF is in set condition. Entry Counter FF will be set, if reset, by ME1L3 True, which occurs when the first digit of a number is entered. Thus, ME1L3 True ensures that the Entry Counter FF is set at the beginning digit of a number entry. When Entry Counter FF is set, AEC2 is False and enables O-282.

ECFS4 is developed by I-281, which is an R-C coupled inverter that produces a 20 ms long negative (False) output pulse when triggered by the leading edge of an input signal that is longer than 20 ms. The input to I-281 is ECFS1, which is the set output of CFS FF. When CFS sets through the set of CF FF, which occurs in all functions, ECFS1 goes True, and remains True for much longer than 20 ms.

2. ENTRY COUNT INHIBIT

There are four other operations that set CFS FF besides set of CF FF. They are: Power On (Start), Overflow Lock release, Clear Display, and Clear Entry. These operations should not result in an Entry Count because they are not the result of a valid completed number entry. The purpose of the Entry Counter FF is to block an Entry Count when these operations occur. When they occur Entry Counter FF resets and makes AEC2 True, which inhibits O-282.

One set of Entry Counter FF is through EROF9 True applied through an isolation diode to the reset output. EROF9 is developed by O-003. If ESTA3 (Start), or ECY1 (Clear D-play), or XOF7 (Overflow Lock reed switch) is True at O-003, the output is True. True output of O-003 is applied through an emitter-follower as EROF9 to the reset output of Entry Counter FF, and the FF resets, which makes AEC2 True at O-282 and inhibits the gate. Therefore, with O-282 inhibited, there is no count in the Entry Counter for Power On, Clear Display, or Overflow Lock release.

The entry count for a Clear Entry function is inhibited by XCE7 True at an AC gate on reset input of Entry Counter FF, which enables the gate. When ECFS1 goes True at this gate because initiation of Clear Entry set CF FF which, in turn, set CFS FF, Entry Counter FF resets. When Entry Counter FF is reset, AEC2 is True at O-282 and the gate is inhibited, thus no Entry Count for a Clear Entry function.

3. ENTRY COUNT ENABLE

With the entry of the first digit of a number, ME1L3 goes True. ME1L3 True is applied to the set input of Entry Counter FF, and sets the FF. When Entry Counter FF is set, AEC2 is False at O-282 and Entry Counts are enabled.

4. ENTRY COUNTER CIRCUIT

The Entry Counter circuit is composed of a conventional FF, Logic gates, inverters, and an emitter-follower.

a. ENTRY COUNTER FF. The Entry Counter FF is a conventional two-transistor FF such as is generally used in the EC-130 and EC-132. Note, however, the absence of speed-up capacitors, which have been found unnecessary.

b. INVERTER 281. The base circuit of I-281 is unusual with respect to most of the inverters in the EC-132 due to the presence of the .47 ufd capacitor. Remembering that ECFS1 will be True for longer than 20 ms, the purpose of the capacitor is to take the leading edge of ECFS1 signal and cause a 20 ms (approximately) output pulse from the inverter, which will act as the increment signal at O-282 when O-282 is enabled by AEC2 False.

With ECFS1 at about 8V, the diode is partially conducting and the 8V appears at the 18K side of the capacitor. The inverter is biased on by -12V through the 68K and the base side of the capacitor is at about -.2V. Thus the capacitor has a charge of about 8V. When ECFS1 goes True at the anode of the diode, the capacitor sees effective ground at the diode side and must discharge. This is a positive transition of about 8V and is applied to the base of the transistor, which turns off. The transistor will be off until the capacitor discharges through the 68K resistor and starts to charge toward -12V, at which time the transistor turns on.

The effective time from inverter off to inverter on is designed to be about 20 ms and is determined only by the discharge time of the .47 ufd capacitor through the 68K resistor and is not affected by the pulse width of ECFS1 as long as ECFS1 is longer than 20 ms, which it will be.

When the inverter is on, ECFS4 is at effective ground (0V). When the inverter is off, ECFS4 is -12V. Therefore ECFS4 goes False at the time ECFS1 goes True at the input diode and remains False for 20 ms, when it again goes True regardless of ECFS1 True. ECFS4 applied to O-282 provides an entry count when O-282 is enabled by AEC1 False.

c. OR GATE O-282. The OR gate is a conventional two-diode OR gate returned to -12V through a 10K resistor. While the gate configuration is a positive OR, a valid output is produced only when both inputs are False, thus the gate acts as a negative AND.

d. EMITTER-FOLLOWER. The emitter-follower after the OR gate is also a little unusual. First, there is a 150 ohm resistor in the collector circuit which reduces the power dissipated by the transistor. Second, the emitter load has two resistors to obtain proper bias and drive for the next inverter (I-283).

d. IMPULSE COUNTER CONTROL INVERTER. I-283, when on, provides a current path from the coil of the Impulse Counter to ground. When I-283 is on, effective ground appears at the collector which is connected to the bottom of the coil. About 150 ma will flow through the coil and the Counter will advance by one count. This current pulse lasts for about 20 ms, which is the length of time ECFS4 is False at O-282. When ECFS4 goes True, I-283 turns off, which opens the current path for the Counter coil. A 20 ufd capacitor is connected from the top of the coil to ground. When I-283 is off, this capacitor is charged up to -80V. When I-283 turns on, the capacitor discharges through the coil, which provides a large current pulse and positive Impulse Counter advance. The diode connected across the coil is forward biased for the "inductive kick" which occurs when the current flow through the coil is interrupted abruptly by turn-off of I-283, and short circuits this high peak voltage.

f. IMPULSE COUNTER (804209). The Impulse Counter is a mechanical counter of four-digit capacity. The circuitry previously discussed is for the purpose of providing an impulse of sufficient duration and current level to positively advance the counter. The Impulse Counter is located on the right-hand side of the display CRT.

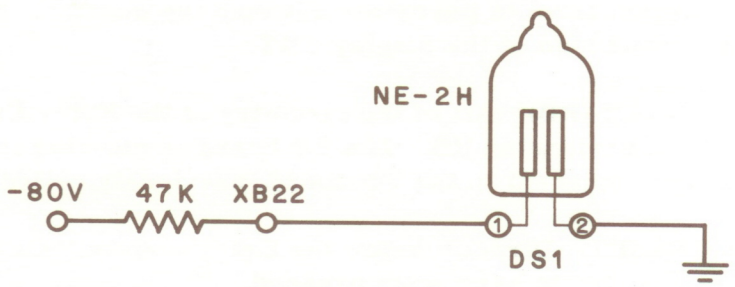
g. PRINTED CIRCUIT BOARD (RB). All of the circuitry of the Entry Counter is contained on one printed circuit board designated RB. The RB board is mounted in the upper right-hand corner of the CRT area adjacent to the PB board, previously called PBA.

h. IMPULSE COUNTER RESET. Directly below the Entry Counter is a Reset button which mechanically returns the counter to zero when pressed.

J. PILOT LIGHT CIRCUIT

When the Power switch is On, the pilot light on the EC-132 provides a continuous external indication that power is applied to the EC-132. The 80V output of both the EC-130 and EC-132 is from terminal XB16. (XB is the new designation for the PWX board). In the EC-132, a 47K resistor is connected from this point to a new terminal XB22. XB22 is connected to one side of an NE-2H neon lamp, with the other side of the lamp connected to ground. This lamp is mounted in a holder underneath the lower right-hand corner of the keyboard mask adjacent to the Decimal Point Position Thumbwheel.

When AC power is applied to the EC-132, the 80V power supply is energized and 80V is applied to one side of the neon lamp. This ionizes the lamp and the lamp glows. The glow of the lamp can be seen through a plastic end cap at the point where the lamp is mounted. The neon lamp will glow during the time AC power is applied, and provides Power-On indication.



SECTION 3 ADJUSTMENTS

No adjustments furnished.

Adjustments are the same as the EC-130.