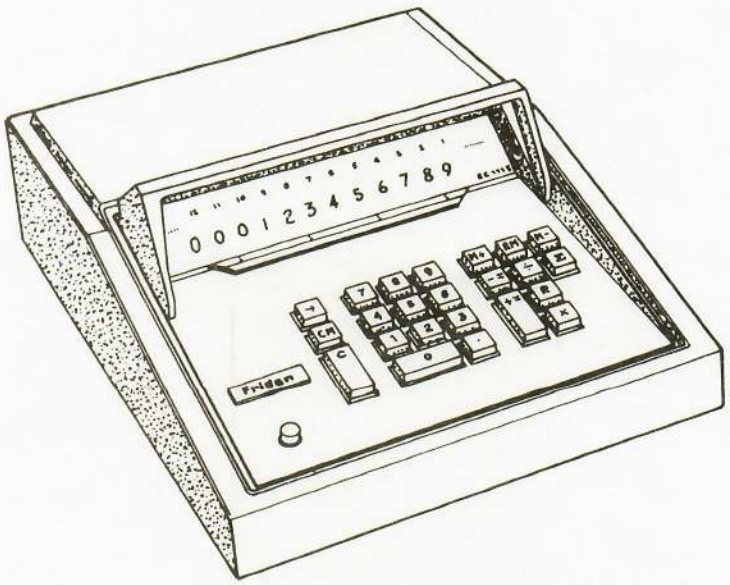
Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

Friden

1113 ELECTRONIC CALCULATOR

SERVICE
MANUAL



Friden
FRIDEN, INC. · DIVISION OF THE SINGER COMPANY
SAN LEANDRO · CALIFORNIA · 94577

TPA-169SL

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

CONTENTS

Page

SECTION 1 - GENERAL DESCRIPTION

GENERAL 1
SPECIFICATIONS 1
KEY FUNCTIONS 2

SECTION 2 - CALCULATING METHODS

GENERAL 1
ADDITION 2
SUBTRACTION 3
MULTIPLICATION 4
CONSTANT MULTIPLICAND 5
DIVISION 6
CONSTANT DIVISOR 7
SIGMA 8
INVOLUTION 10
COMBINED CALCULATIONS 11

SECTION 3 - LOGIC ELEMENTS

LOGIC CIRCUITS USED IN 1113 1
SHIFT REGISTER (HD701M) 1
INVERTER (HD703M) 2
4-INPUT AND GATE (HD704M) 2
J-K FLIP FLOPS (HD705M) 3
SHIFT REGISTER (HD707M) 4
2-INPUT AND GATE; 3-INPUT AND GATES (HD712M) 4
CIRCUITS COMPOSED OF DISCRETE COMPONENTS 5

SECTION 4 - LOGIC CIRCUITS

GLOSSARY OF LOGIC FLIP FLOPS AND REGISTERS i
GENERAL 1
TIMING 2
Data Organization 3
Clock Pulse Generator 3
Bit Counter 3
Digit Counter 3
SHIFT REGISTERS 4
Indicator Register (IR) 4

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

SECTION 4 - LOGIC CIRCUITS (Cont.) Page

Accumulator Register (AR) 4
Memory Register (MR) 4
FULL ADDER AND SUBTRACTER (FAS) 4
Carry Flip Flop (CAF) 5
Compensation 5
Compensation Flip Flop (CDF) 6
Block Compensation 8
SPECIAL FAS GATES 8
FAS TRUTH TABLES 9
ADD OR SUBTRACT ONE (ASO) 10
Add "1" 10
Subtract "1" 11
MASTER FUNCTION FLIP FLOPS (J-K Type) 11
CONDITIONAL CONTROL PHASE COUNTER 12
ENTRY CIRCUIT 12
DISPLAY CIRCUIT 12
POWER SUPPLY 12

SECTION 5 - ELECTRONIC CIRCUITS

POWER SUPPLY 1
MASTER OSCILLATOR 2
DISPLAY CIRCUIT 4

SECTION 6 - FUNCTION LOGIC

GENERAL 1
ENTRY FUNCTION 1
ENTRY FLOW CHART ANALYSIS 1
1st DIGIT SELECTION LOGIC (Figure 6-3) 5
2nd DIGIT SELECTION LOGIC (Figure 6-4) 5
ENTRY FLOW CHART (Figure 6-2) 7
ADD AND SUBTRACT 9
SAMPLE PROBLEM 9
ADD AND SUBTRACT FLOW CHART ANALYSIS 10
ADD OR SUBTRACT (Basic Logic CC1) 14
ADD OR SUBTRACT (Basic Logic CC2) 15
ADD OR SUBTRACT (Basic Logic CC4) 16
ADD OR SUBTRACT (Basic Logic CC5/CC7) 17
ADD OR SUBTRACT (Basic Logic CC6.SCF) 18
ADD OR SUBTRACT (Basic Logic CC9) 19

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

SECTION 6 - FUNCTION LOGIC (Cont.)

Page

ADD OR SUBTRACT FLOW CHART (Figure 6-7) 21
MULTIPLY 23
SAMPLE PROBLEM 23
MULT FLOW CHART ANALYSIS. 24
MULTIPLY (Basic Logic CC0.SCF) 29
MULTIPLY (Basic Logic CC3 and CC9) 30
MULTIPLY (Basic Logic CC4) 31
MULTIPLY (Basic Logic CC6.SCF) 32
MULTIPLY (Basic Logic CC6.SCF) 33
MULTIPLY (Basic Logic CC7) 34
MULTIPLY (Basic Logic CC8) 35
MULTIPLY FLOW CHART (Figure 6-15). 37
DIVIDE 39
SAMPLE PROBLEM 39
DIVIDE FLOW CHART ANALYSIS 40
DIVIDE (Basic Logic CC1) 45
DIVIDE (Basic Logic CC2) 46
DIVIDE (Basic Logic CC4) 47
DIVIDE (Basic Logic CC6.SCF 48
DIVIDE (Basic Logic CC6.SCF). 49
DIVIDE (Basic Logic CC7) 50
DIVIDE (Basic Logic CC8) 51
DIVIDE (Basic Logic CC9) 52
DIVIDE FLOW CHART (Figure 6-24). 53
SIGMA. 55
OUTLINE OF FLOW IN SIGMA OPERATION. 55
FLOW CHART ANALYSIS 55
SIGMA; M-; M- (Basic Logic). 57
OUTLINE OF FLOW IN MEMORY PLUS OR MEMORY MINUS 58
SIGMA, MEMORY PLUS OR MEMORY MINUS FLOW CHART (Figure 6-33) 59
REVERSE KEY (R) 61
MEMORY RECALL KEY (MR) 61
REVERSE KEY AND MEMORY RECALL FLOW CHART (Figure 6-36) . 62
RECALL MEMORY (Basic Logic) 64
SHIFT KEY 65
SHIFT KEY FLOW CHART (Figure 6-38). 66
CLEAR KEY (C) AND MEMORY CLEAR KEY (MC) 67
OVERFLOW. 67

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

SECTION 7 - MAINTENANCE Page

DISASSEMBLY AND ASSEMBLY 1
INSPECTION PROCEDURE 1
OPERATIONAL CHECK 2
TROUBLESHOOTING 4
ADJUSTMENTS 6
COMPONENT IDENTIFICATION AND LAYOUT 7
PC BOARD #1 11
COMPONENT IDENTIFICATION PC BOARD #1 12
PC BOARD #2 13
COMPONENT IDENTIFICATION PC BOARD #2 14
PC BOARD #3 15
COMPONENT IDENTIFICATION PC BOARD #3 16
PC BOARD #4 (Type CD71) 17
COMPONENT IDENTIFICATION PC BOARD #4 (Type CD71) 18
PC BOARD #4 (Type CD79) 19
COMPONENT IDENTIFICATION PC BOARD #4 (Type CD79) 20
PC BOARD #5 21
COMPONENT IDENTIFICATION PC BOARD #5 22
WIRING LIST & SIGNAL LIST CROSS REFERENCE 23
TIMING CHART (Figure 7-9) 26

SECTION 8 - SERVICE LETTERS

SECTION 9 - PARTS

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

GENERAL

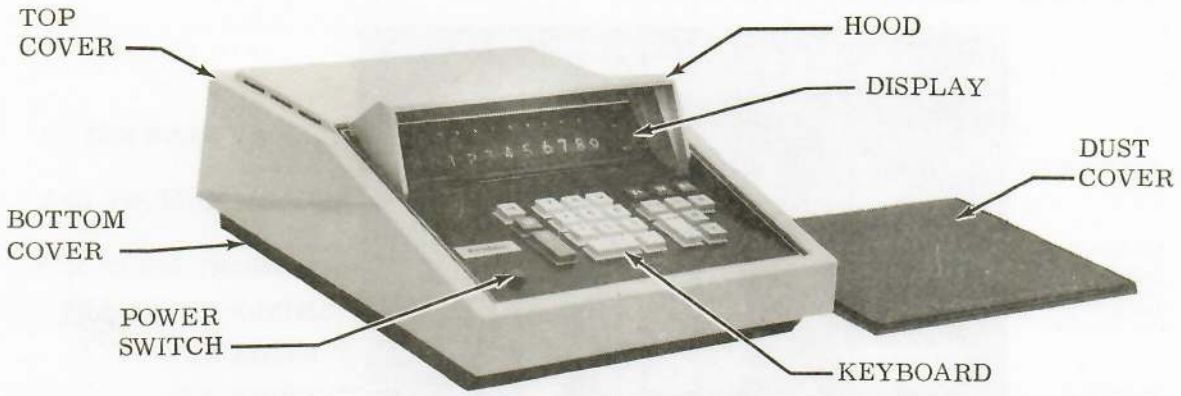


FIGURE 1-1

The 1113 Electronic Calculator (Figure 1-1) is a small, light weight, high speed portable desk calculator featuring: Nixie type, number indicating tube display, 10-key keyboard, and integrated (IC) electronic circuitry for long life and maximum reliability.

SPECIFICATIONS

Size:	11" wide, 14" deep, and 5" high (approx).
Weight:	12 lbs.
Keyboard:	10-key type. Any number can be entered (maximum of 12 digits) using the ten digit keys (0-9) and the Decimal Point key if applicable. The Function keys are arranged around the Digit keys in a manner that provides maximum operator facility and ease of operation.
Decimal Point:	Automatic decimal point alignment
Speed:	The longest arithmetic problem in:

Addition and Subtraction	40 ms.
Multiplication	73 ms.
Division	80 ms.

Reliability:	Use of the latest state of the art integrated circuit chips, transistors, and diodes insures maximum reliability and serviceability.
Power Source:	The 1113 can be connected to any standard, grounded 115 VAC outlet. Transformer taps are provided for 110 or 117 VAC operation. Optional voltages are available on special order.
Power Dissipation:	15 watts
Ambient Temperature:	0-40° C (32-104° F)

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

KEY FUNCTIONS

The Keyboard keys (Figure 1-2) can be grouped by function as: Digit, Arithmetic, and Data Handling. The Digit keys are used in entering numbers; the Arithmetic keys are used in operations which involve computations; and the Data Handling keys shift numbers between registers, or clear numbers from the machine.

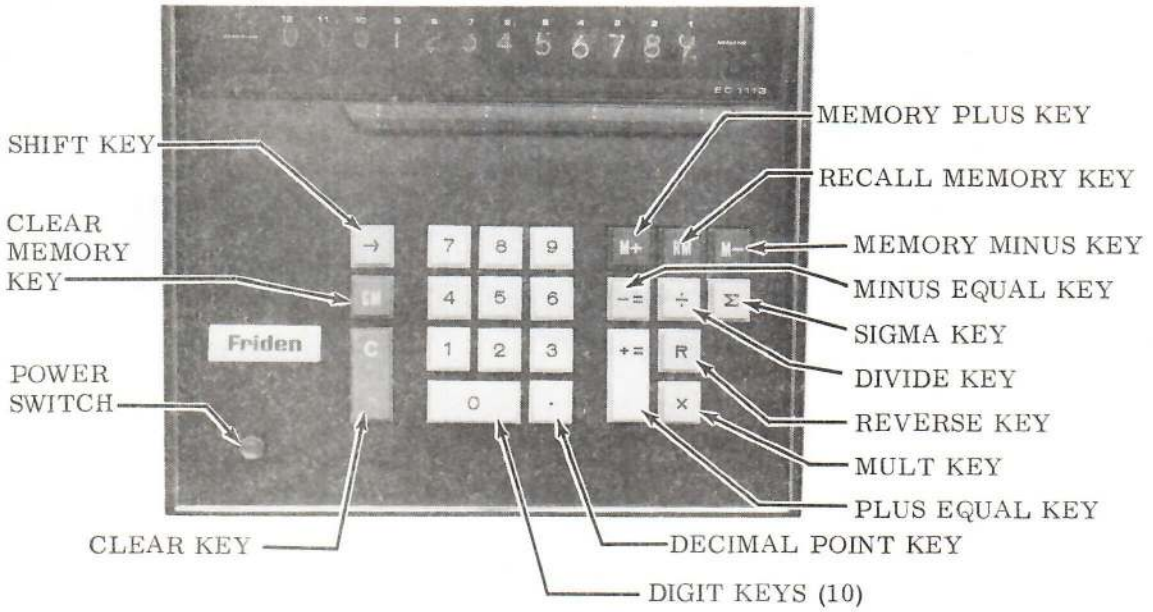


FIGURE 1-2

1. Digit Keys: Numbers are entered into the 1113 data loop digit-by-digit by use of the digit keys. Since numbers can contain decimal digits, the decimal point key is also considered to be in the digit group because it is used for a decimal number entry.

2. Arithmetic Keys: The arithmetic keys enable and initiate the arithmetic operations that will be performed with and on the numbers. Briefly their functions are:

+= (Plus Equal)	Depression of this key initiates the addition, multiplication, and division functions.
-= (Minus Equal)	Depression of this key initiates the subtraction, multiplication, and division functions.
X (Multiply)	Depression of this key enables the multiplication function. Depression of the Plus Equal or Minus Equal key is necessary to initiate the multiplication function.
÷ (Divide)	Depression of this key enables the division function. Depression of the Plus Equal or Minus Equal key is necessary to initiate the divide function.
M+ (Memory Plus)	Depression of this key causes the number in IR to be added to the number in MR.
M- (Memory Minus)	Depression of this key causes the number in IR to be subtracted from the number in MR.
Σ (Sigma)	Depression of this key enables the automatic accumulation of products in MR when a multiply function occurs. This key is self locking and must be depressed prior to the multiplication. To release the key, it must be depressed again.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

3. Data Handling Keys: Using the data handling keys results in the shifting of numbers between registers; the shifting right of numbers in the indicator register (IR); the clearing of numbers from IR, AR, or MR; and the recall of numbers from the Memory Register. The function of the Data Handling Keys is:

R (Reverse)	Depression of this key exchanges the contents of IR and AR.
RM (Recall Memory)	Depression of this key recalls the contents of the memory register (MR) to the indicator register (IR) without destroying the contents of MR.
→ (Right Shift)	Depression of this key results in the right shift of the number in the indicator register (IR) by one digit.
C (Clear)	Depression of this key clears the IR and the AR registers and restores the Function flip-flops to a reset condition.
CM (Clear Memory)	Depression of this key clears the memory register (MR).

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

GENERAL

This section of the manual describes the various calculating methods possible with the EC 1113 electronic calculator. While the operation of the 1113 is basically simple and straightforward, certain aspects of the operation are different from the typical Friden machine. That is, after the initial digit entry in the 1113, the next key depression must be the desired arithmetic function key (Mult (X), Divide (÷), etc.); then the second digit entry is made and this entry is followed by the desired result key Plus equal (+=) or Minus equal (-=). The sample problems included will give the CSRep a working knowledge of the machine operation which is essential to understanding the machine logic and an aid in troubleshooting.

Listed below are several pertinent points that must be observed in the operation of the 1113 Electronic Calculator.

1. This calculator has an automatic decimal point alignment system.

2. When answers of subtraction are minus, they are indicated with a minus sign.

3. In the following instances, the overflow indicator lamp glows to indicate that an answer or an entry number exceeds the capacity of any register.

   a. When the entry number exceeds 12 digits or 11 decimal places.
   b. In any arithmetic function when the answer exceeds 12 digits (whole number plus decimal number).
   c. In multiplication when the sum of the digits in the multiplicand and the multiplier exceeds 12 digits.
   d. In division when the dividend or divisor exceeds 11 digits.
   e. When the contents of MR exceeds 12 digits.

4. The numbers recalled by depressing the reverse key [R] or recall memory key [RM] must be followed by one of the function keys [+=] [-=] [x] [÷] prior to the next entry.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

ADDITION

Addition is accurate up to 12 digits. Depressing the Clear key is necessary before each calculation.

PROBLEMS:

1. Addition without Decimals ..... 946 + 538 = 1484
2. Addition with Decimals ..... 2.63 + 0.574 = 3.204
3. Chain Addition ..... 946 + 2.63 + 0.574 = 949.204

SOLUTION	INDEX IN KEYBOARD	TOUCH CONTROL KEYS IN SEQUENCE	REMARKS
1. 946 + 538 = 1484		C	
	946	+=	
	538	+=	1484
2. 2.63 + 0.574 = 3.204		C	
	2.63	+=	
	.574	+=	3.204
3. 946+2.63+0.574=949.204		C	
	946	+=	
	2.63	+=	948.63
	.574	+=	949.204

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

SUBTRACTION

Subtraction is accurate up to 12 digits. Depressing the Clear key is necessary before each calculation.

PROBLEMS:

1. Subtraction without Decimals ..... 657 - 492 = 165
2. Subtraction and Addition ..... -530 + 829 = 299
3. Subtraction with True Credit Balance ..... 139 - 567 = -428

SOLUTION	INDEX IN KEYBOARD	TOUCH CONTROL KEYS IN SEQUENCE	REMARKS
1. 657 - 492 = 165		C	
	657	+ =	
	492	- =	165
2. -530 + 829 = 299		C	
	530	- =	
	829	+ =	299
3. 139 - 567 = -428		C	
	139	+ =	
	567	- =	-428

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

MULTIPLICATION

Multiplication is accurate up to 12 digits, if the total of the multiplicand and the multiplier does not exceed 12 digits. In multiplication depressing the Clear key is not necessary before each calculation. In chain multiplication, if the number of digits in the product added to the number of digits in the next multiplier exceeds 12, remove the least significant digits from the product by use of the Shift key.

PROBLEMS:

1. Multiplication without Decimals .... 12 x 24 = 288
2. Multiplication with Decimals ..... 6.14 x 87.2 = 535.408
3. Chain Multiplication ..... 369 x 682 x 34 = 8556372

SOLUTION	INDEX IN KEYBOARD	TOUCH CONTROL KEYS IN SEQUENCE	REMARKS
1. 12 x 24 = 288	12	x	
	24	+ =	288
2. 6.14 x 87.2 = 535.408	6.14	x	
	87.2	+ =	535.408
3. 369 x 682 = 251658	369	x	
	682	+ =	251658
251658 x 34 = 8556372		x	
	34	+ =	8556372

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

CONSTANT MULTIPLICAND

A multiplication problem using a constant multiplicand can be accomplished with or without using memory as follows:

PROBLEMS:

359.83 x 514 = 184952.62

359.83 x 45.36 = 16321.8888

SOLUTION	INDEX IN KEYBOARD	TOUCH CONTROL KEYS IN SEQUENCE	REMARKS
a) Not using memory			
359.83 x 514 = 184952.62	359.83	x	
	514	+=	184952.62
359.83 x 45.36 = 16321.8888		R	359.83
		x	
	45.36	+=	16321.8888
b) Using memory.			
359.83 x 514 = 184952.62		CM	
	359.83	M+	
		x	
	514	+=	184952.62
359.83 x 45.36 = 16321.8888		RM	359.83
		x	
	45.36	+=	16321.8888

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

DIVISION

Division is accurate if the dividend or divisor does not exceed 11 digits. The number of digits in the quotient will equal 12 digits minus the number of digits in the divisor. In division, depressing the Clear key is not necessary before each calculation.

PROBLEMS:

1. Division without Decimals ..... 145 ÷ 12 = 12.08333333
2. Division with Decimals ..... 4962.184 ÷ 13.2 = 375.923030
3. Chain Division ..... 759 ÷ 153 ÷ 456 = 0.01087891

SOLUTION	INDEX IN KEYBOARD	TOUCH CONTROL KEYS IN SEQUENCE	REMARKS
1. 145 ÷ 12 = 12.08333333	145	÷	
	12	+ =	12.08333333
2. 4962.184 ÷ 13.2=375.923030	4962.184	÷	
	13.2	+ =	375.923030
3. 759 ÷ 153 = 4.96078431	759	÷	
	153	+ =	4.96078431
4.96078431÷456=0.01087891		÷	
	456	+ =	0.01087891

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

CONSTANT DIVISOR

A division problem using a constant divisor can be accomplished with or without using memory as follows:

PROBLEMS:

297÷1364 = 0.2177419
6428÷1364 = 4.7126099

SOLUTION	INDEX IN KEYBOARD	TOUCH CONTROL KEYS IN SEQUENCE	REMARKS
a) Not using memory.			
297÷1364 = 0.2177419	297	÷	
	1364	+=	0.2177419
6428÷1364 = 4.7126099		R	1364
		÷	
	6428	R	1364
		+=	4.7126099
b) Using memory.			
297÷1364 = 0.2177419		CM	
	1364	M+	
	297	÷	
		RM	1364
		+=	0.2177419
6428÷1364 = 4.7126099	6428	÷	
		RM	1364
		+=	4.7126099

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

SIGMA

The Sigma key is used for accumulating positive or negative products to Memory automatically. Once depressed, the Sigma key will remain depressed until it is depressed a second time.

PROBLEMS:

1. (8.25 x 593) + (186 x 734) - (2.93 x 326) = 140461.07

SOLUTION	INDEX IN KEYBOARD	TOUCH CONTROL KEYS IN SEQUENCE	REMARKS
1. 8.25 x 593 = 4892.25		Σ	
	8.25	x	
	593	+=	4892.25
186x734=136524.00	186	x	
	734	+=	136524.00
-(2.93 x 326) = -955.18	2.93	x	
	326	-=	-955.18 *
Accumulated Answer: 140461.07		RM	140461.07

* Indicates subtraction from accumulated product.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

SIGMA (Cont.)

2. 589 + 610 + (274 x 916) - (59 x 1037) = 191000

3. (26040 ÷ 28) + (153 x 842) = 129756.000000

SOLUTION	INDEX IN KEYBOARD	TOUCH CONTROL KEYS IN SEQUENCE	REMARKS
2. 589 + 610 + (274 x 916) - (59 x 1037) = 191000		C	
		CM	
	589	M+	
	610	M+	
		Σ	
	274	x	
	916	+=	250984
	59	x	
	1037	-=	-61183
191000		RM	191000
3. 26040 ÷ 28 = 930.000000		CM	
	26040	÷	
	28	+=	
		M+	930.000000
		Σ	
153x842=128826.000000	153	x	
	842	+=	128826.000000
Accumulated Answer: 129756.000000		RM	129756.000000

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

INVOLUTION

The process of raising a number to a higher power is indicated in the following problems.

PROBLEMS:

1. $123^2 = 15129$
2. $123^3 = 1860867$
3. $123^4 = 228886641$

SOLUTION	INDEX IN KEYBOARD	TOUCH CONTROL KEYS IN SEQUENCE	REMARKS
1. $123^2 = 15129$	123	C x + =	 15129
2. $123^3 = 1860867$	123	C x + = x R + =	 15129 123 1860867
3. $123^4 = 228886641$	123	C x + = x + =	 15129 228886641

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

COMBINED CALCULATIONS

Examples of combined calculations are shown in the four problems that follow.

PROBLEM:

1. $1.23 + (1.23 \times 456) = 562.11$

2. $\frac{485}{174 + 53 + 102} = 1.47416413$

3. $\frac{15}{2 + 15} = 0.882352941$

SOLUTION	INDEX IN KEYBOARD	TOUCH CONTROL KEYS IN SEQUENCE	REMARKS
1. $1.23 + (1.23 \times 456) = 562.11$		C	
	1.23	x	
	456	+=	560.88
		R	1.23
		+=	562.11
2. $\frac{485}{174 + 53 + 102} = 1.47416413$		C	
	174	+=	
	53	+=	227
	102	+=	329
	485	÷	
		R	329
		+=	1.47416413
3. $\frac{15}{2 + 15} = 0.882352941$		C	
	2	+=	
	15	+=	17
		* R	15
		÷	
		R	17
		+=	0.882352941

* This saves a manual entry of 15.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

COMBINED CALCULATIONS (Cont.)

4. (28.8 x 384 + 7.68 x 192) x 1.2 x 32 = 481296.384

SOLUTION	INDEX IN KEYBOARD	TOUCH CONTROL KEYS IN SEQUENCE	REMARKS
4. 28.8 x 384 = 11059.2		Σ	
	28.8	x	
	384	+=	11059.2
7.68 x 192 = 1474.56	7.68	x	
	192	+=	1474.56
(11059.2000+1474.56000)		RM	12533.76
x 1.2 x 32 = 481296.384		x	
	1.2	+=	15040.512
		x	
	32	+=	481296.384

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

LOGIC CIRCUITS USED IN THE EC 1113

Listed in the Table, Figure 3-1, are the 6 types of IC's and their circuit functions in the EC 1113 Calculator. In addition to the IC circuits, discrete circuits are used for Inverters, OR gates, and Emitter Followers. Each circuit (IC and discrete) is described briefly in the following paragraphs. (NOTE: All J-K flip flops and Shift Register IC's require -24 volt clock pulses for proper operation.)

Type of IC	Circuit Function	No. of circuits
HD701M	Dual 8-bit dynamic shift register	Two 8-bit shift registers
HD703M	Four inverters	Four inverter circuits
HD704M	Dual 4-input AND (OR) gate	Two 4-input AND gate circuits
HD705M	Dual J-K flip-flop	Two J-K flip-flop circuits
HD707M	4-bit shift register	Four D type flip flops (One 4-bit shift register)
HD712M	Dual 2-input AND (OR) gate	Two 2-input AND gate circuits
	Single 3-input AND (OR) gate	One 3-input AND gate circuit

FIGURE 3-1

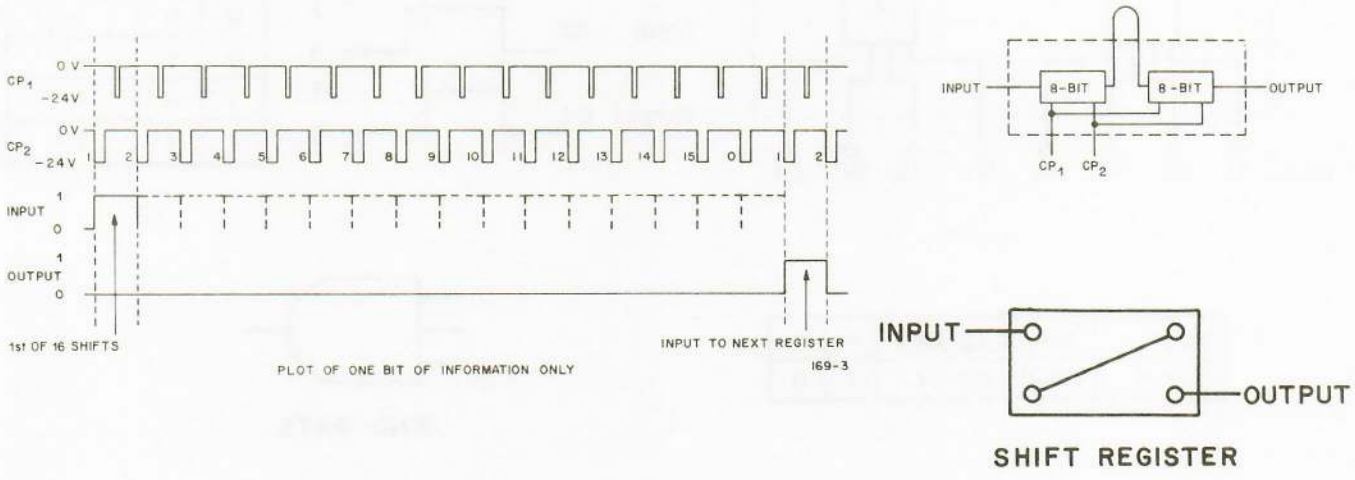


FIGURE 3-2

1. Shift Register (HD701M)

This IC comprises two 8-bit Shift Registers externally connected to form an equivalent 16-bit Shift Register having a BCD capacity of 4 digits. Each register (IR, AR, and MR) will therefore require 3 HD701M IC chips for a 12 digit capacity. Data is dynamically circulated in the register controlled by the clock pulses CP1 and CP2 as shown in Figure 3-2. Note on the timing diagram that 16 CP1 and CP2 pulses are needed to shift one bit of data through the register.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

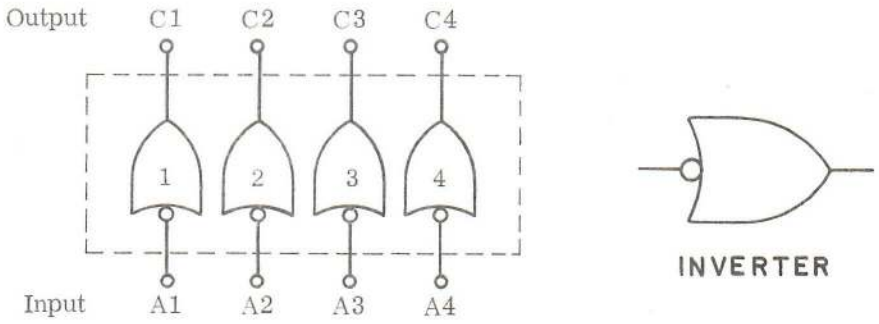
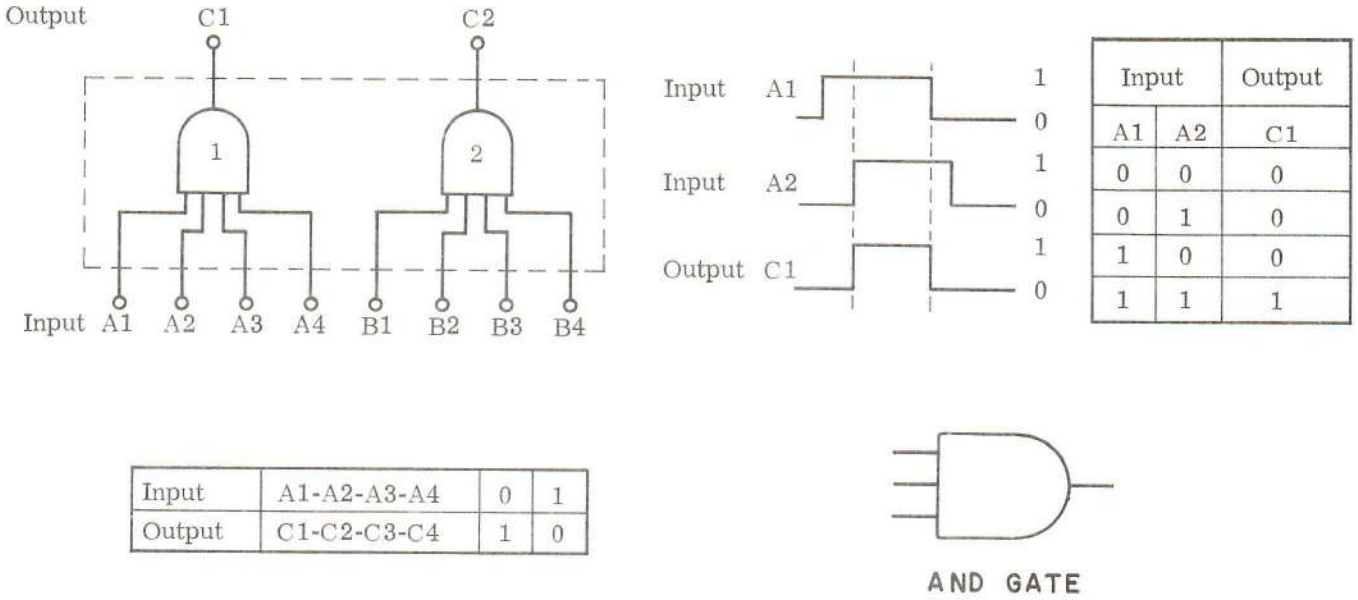


FIGURE 3-3

2. Inverter (HD703M)

This IC comprises four Inverter Circuits with each Inverter operating independently. Logically, it is a NOT circuit used only to invert logic signals. The IC chip with 4 inverters is shown symbolically in Figure 3-3.



Input		Output
A1	A2	C1
0	0	0
0	1	0
1	0	0
1	1	1

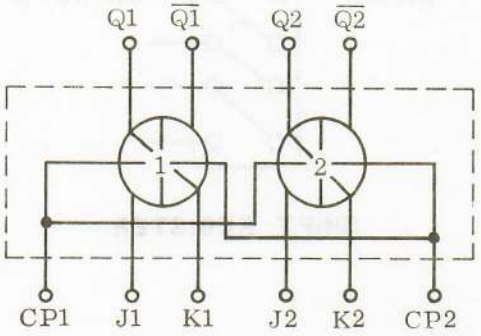
Input	A1-A2-A3-A4	0	1
Output	C1-C2-C3-C4	1	0

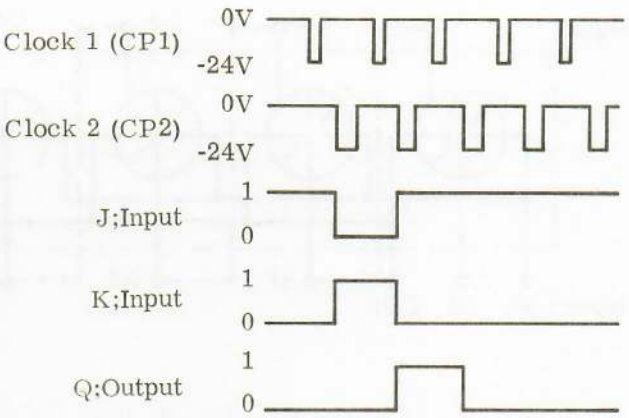
FIGURE 3-4

3. 4-Input AND Gate (HD704M)

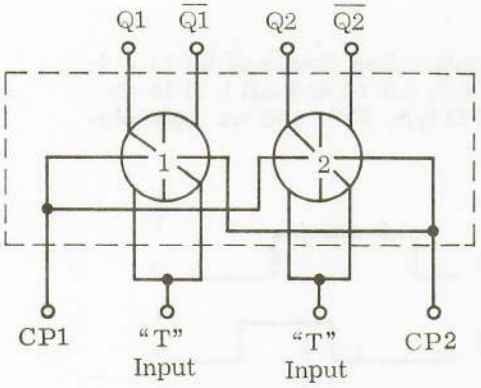
This IC comprises a pair of AND gate circuits, each provided with four inputs, and with each circuit operating independently. The number of inputs used can vary from 2 to 4 depending on the circuit requirements. The IC chip is shown symbolically in Figure 3-4, along with a truth table and input output pulses based on using only 2 inputs.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com





Input		Output
J	K	Q (Set Side)
1	1	No Action
1	0	0 (Reset)
0	1	1 (Set)
0	0	Toggle



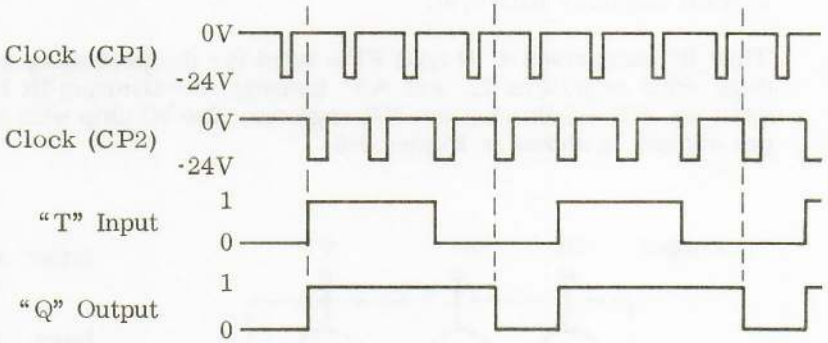
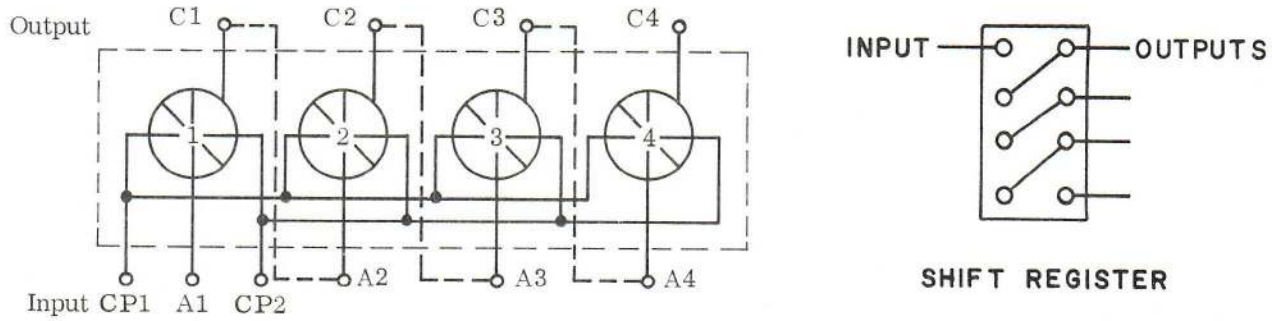


FIGURE 3-5

4. J-K Flip Flop (HD705M)

This IC comprises a pair of J-K FF with each circuit operating independently. However, as shown in Figure 3-5, the clock pulses are common to both flip-flops. In the counter circuits, the J and K terminals are connected together allowing the FF to toggle on each negative input pulse. Note: The J-K (master-slave) FF's used in the 1113 operate on negative logic. That is, the permiss inputs (J or K) must be at -14 volts (logic 0) at the same time the -24 volt clock pulse CPI arrives to set the master FF. Then, when the -24 V CP2 pulse arrives, it will set the slave and the output of the FF will reflect a change of state. Likewise, if the permiss inputs (J-K) are tied together, the FF will toggle in step with the negative input pulses.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com



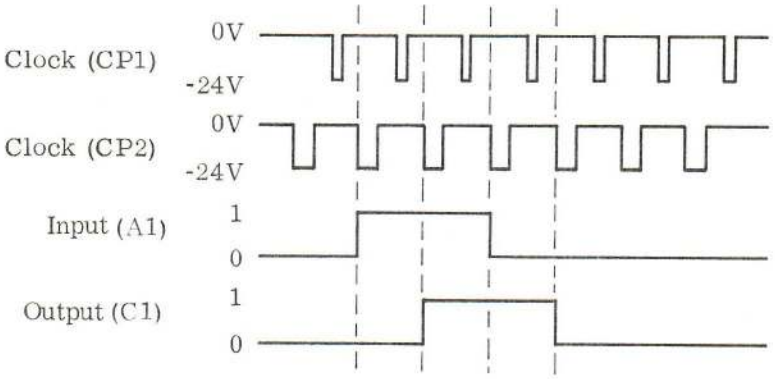


FIGURE 3-6

5. Shift Register (HD707M)

This IC comprises 4 D type FF's used for the preceding and succeeding stages of the two 14-digit shift registers IR and AR; thereby, constituting IR 14, IR 1, AR 14 and AR 1. It is also used as MR 1 following the MR register. The IC chip with its 4 D type FF's and its input output signals is shown in Figure 3-6.

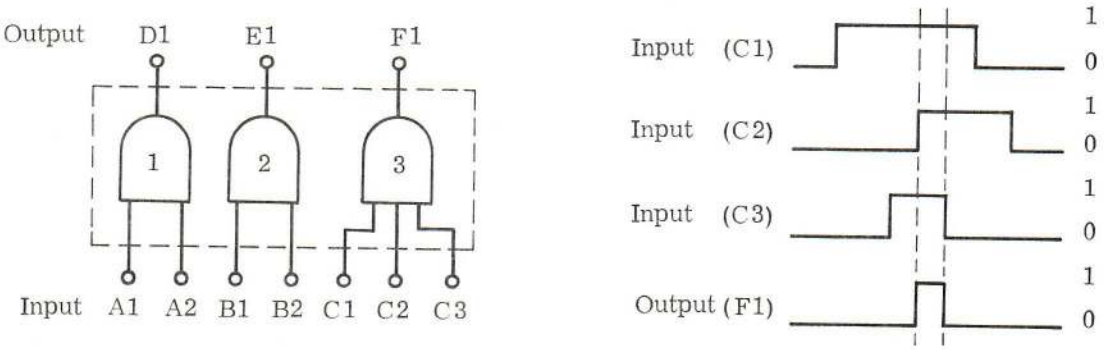


FIGURE 3-7

6. 2-Input AND and 3-Input AND gates (HD712M)

This IC is similar to HD704M except for its number of inputs and gate circuits. That is, two 2-Input AND gates and one 3-Input AND gate circuits are integrated in one module. Relationship between signal input and output is also identical to that described in item 3. The IC chip with its signal relationships is shown in Figure 3-7.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

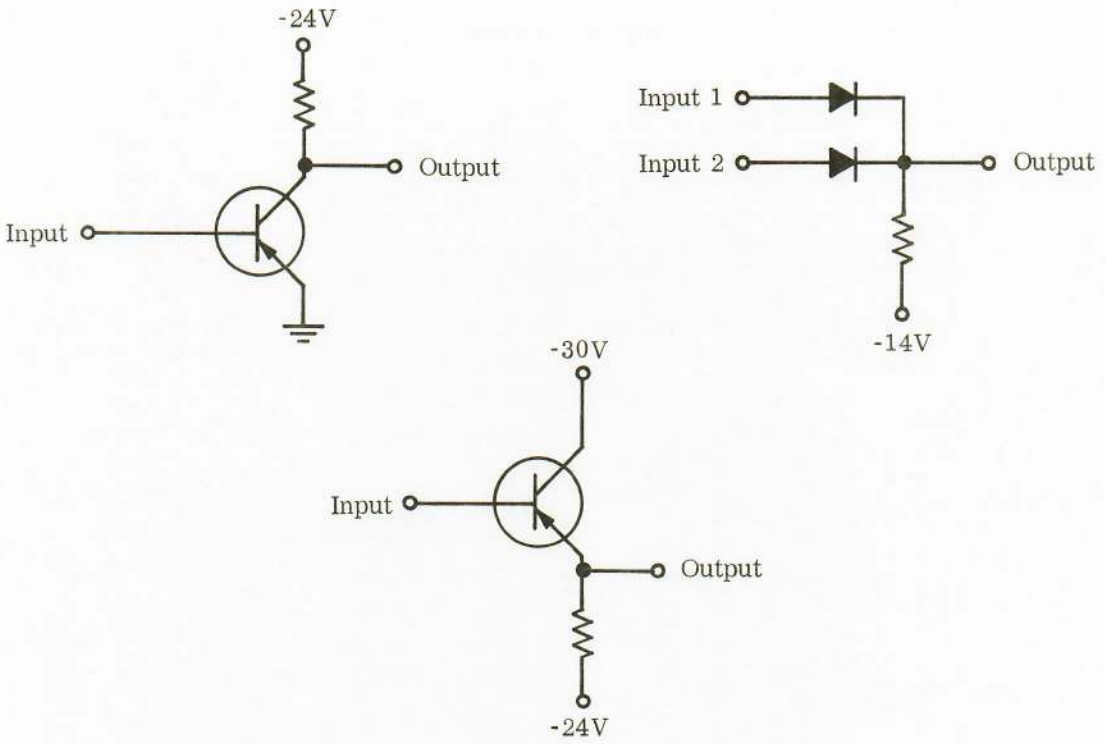


FIGURE 3-8

7. Circuits Composed of Discrete Components

The 1113 uses three basic electronic circuits composed of discrete components as shown in Figure 3-8. They are: Inverters, OR gates, and Emitter Followers.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

LOGIC CIRCUITS

GLOSSARY OF LOGIC FLIP FLOPS AND REGISTERS

Symbol	Name	Function
AR	Accumulator Register	Contains one of the factors in an arithmetic function and accumulates the intermediate answers.
ASF	Accumulator Register Sign FF	Indicates the sign of the number in AR.
ASO	Add or Subtract One	Adds or subtracts "1" from decimal point or shift information in IR.
BCF	Boundary Code FF	Used in various logic functions and it enters the boundary code in Mult and Divide.
CAF	Carry FF	Indicates a carry or borrow in the arithmetic function being performed.
CCF 1 2 4 8	Conditional Control Phase Counter	Divides the arithmetic functions into 10 phases (CC0 through CC9).
CDF	Compensation FF	Enables a compensation during arithmetic function.
DC 1 2 4 8	Digit Counter	Provides signals called digit times.
DIF	Divide FF	Enables the Divide function.
DTF 1 2 4	Bit Counter	Provides timing signals called Bit times.
FAS	Full Adder and Subtracter	Performs additions and subtractions necessary in the arithmetic functions.
FASx	"X" input to FAS	AR input to FAS.
FASy	"Y" input to FAS	IR input to FAS.
IR	Indicator Register	Contains one of the factors in an arithmetic function and stores the final answer for display.
ISF	Indicator Sign FF	Indicates the sign of the number in IR.
KEF	Key FF	Allows only one operation to occur for each key depression of: Entry, Right Shift, and Reverse.
MIF	Memory In FF	Enables an addition or subtraction from memory.
MLF	Mult FF	Enables the Mult function.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

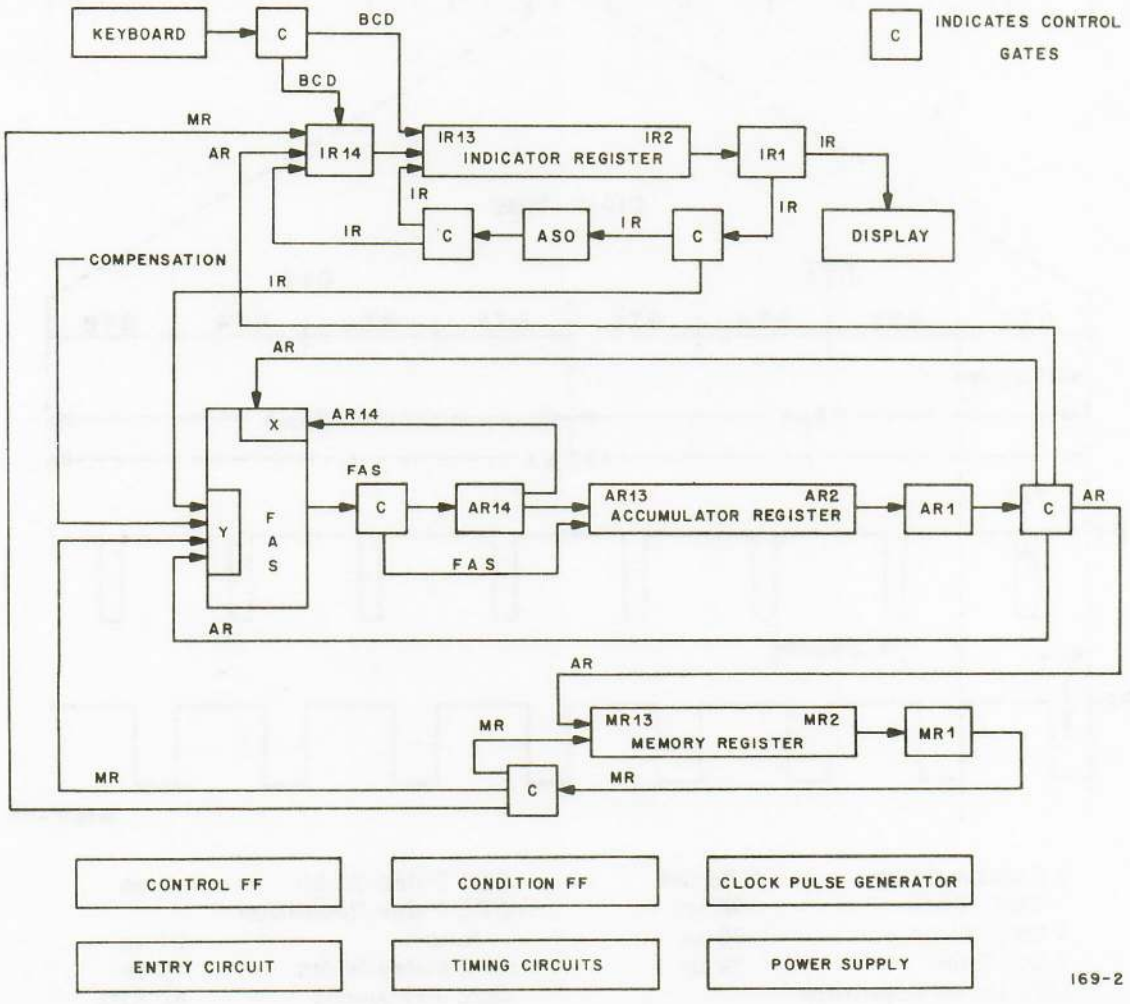
Symbol	Name	Function
MR	Memory Register	Storage register for various arithmetic purposes.
MSF	Memory Sign FF	Indicates the sign of the number in Memory.
OSF	Operation Start FF	Starts all functions.
OVF	Overflow FF	Indicates an overflow condition.
OVF-1	Overflow FF	Used in conjunction with the OVF FF to detect an overflow condition in a mult operation.
±1F	Plus or Minus One FF	Used with ASO to add or subtract "1" from decimal point and shift information.
POF	Decimal Point FF	Indicates a decimal point entry. It is also used as a control FF in arithmetic functions.
RDF	Register Detect FF	Enables a series of additions or subtractions.
SCF	Shift Control FF	Controls Shifting of IR and AR.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

GENERAL

In this section are explanations of the logic circuits and basic machine circuits that are used extensively in the Function Logic. The explanations are designed to give the basic operation of these circuits so that in the Function Logic discussions they need only be referred to. Also included are: A glossary of logic flip flops and registers, logic symbols used in the 1113, and a block diagram (Figure 4-1) indicating the data flow through the registers and associated circuits.

NOTE: The EC 1113 uses positive logic with logic 1 = "0" volts and logic "0" = -14 volts. However, all clock pulses are -24 volts and are not considered a logic level. The reason for the -24 volt clock pulses is due to the internal construction of the IC chips (modules) used in this machine.



BLOCK DIAGRAM

FIGURE 4-1

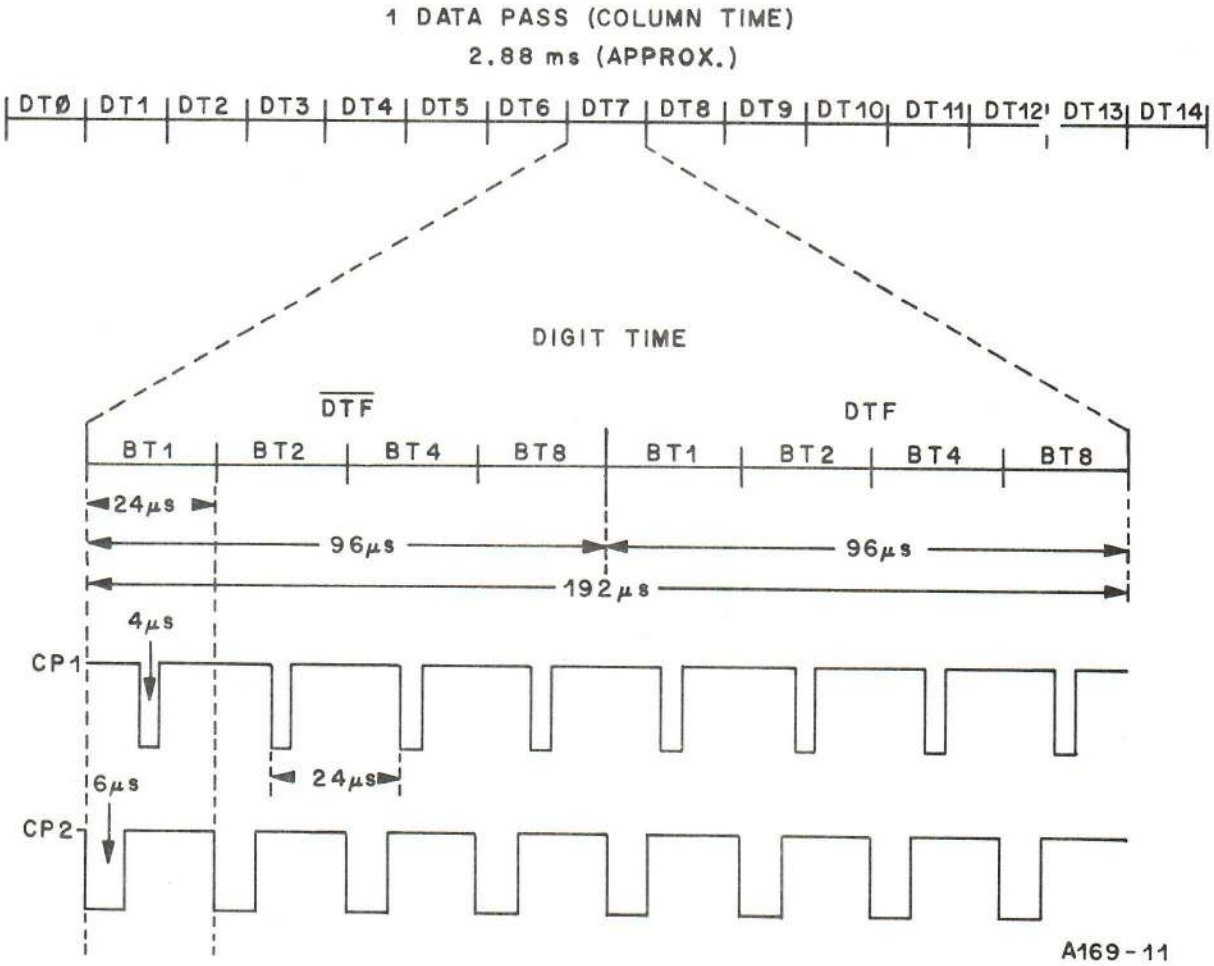
Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

TIMING

Timing circuits supply the clock pulses and timing signals for controlling all operations of the machine. These circuits consist of: Master Oscillator, Clock Pulse Generator, Bit Counter, and Digit Counter. The Clock Pulses derived from the Master Oscillator are CP1 and CP2; while the Clock Pulse Generator supplies seven other Clock Pulses designated: AR14CP1, ARCP1, IRCP1, MRCP1, DFCP1, CCCP1, and CFCP1. Unlike the logic signal levels at "0" volts and -14 volts, the Clock Pulse levels are at "0" volts and -24 volts.

Data Organization

Data flow in the 1113 is dynamic and serial in the form of binary coded decimal (BCD) information. Data is organized into: Bit times; Digit times; and Column times as shown in the Timing Chart, Figure 4-2.



1 Column Time	2.88 ms	CP1 Pulse Width	4 µs
1 Digit Time	192 µs	CP2 Pulse Repetition Rate	24 µs
1 DTF Time	96 µs	CP2 Pulse Width	6 µs
1 Bit Time	24 µs	Osc. Frequency	42 KHz
CP1 Pulse Repetition Rate	24 µs		

NOTE: These values are approximate and will vary from machine to machine.

FIGURE 4-2

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

Master Oscillator. The basic timing is produced by a free running astable (unsymmetrical) multivibrator, operating at a frequency of 42 KHz. The two outputs of this multivibrator are fed to Inverter circuits which shape the outputs and produce clock pulses designated CP1 and CP2. These clock pulses occur every 24 microseconds, but at different time intervals and with different pulse widths. They are used with other timing signals for controlling the logic operation of the machine. A detailed description of the astable multivibrator is covered in the Friden Correspondence Lesson #19. The added circuitry for controlling the wave shaping is described in Section 5 (Electronic Circuits).

Clock Pulse Generator. The seven output signals from the Clock Pulse Generator occur at the same time as the CP1 (Master Oscillator) but are controlled by the timing counters and other logic signals via logic gates. Being clock pulses, they are at a "0" volt to -24 volt level and are used for controlling the shift registers and the J-K flip flops.

Bit Counter. The Bit Counter consists of 3 J-K Flip Flops (scale of 8, pure binary, up counter) controlled by the clock pulses CP1 and CP2 from the Master Oscillator. Its purpose, in conjunction with the Digit Counter is to provide the basic machine timing. The outputs of the J-K Flip Flops are decoded via logic gates to provide repetitive sequential bit counts (BT1, BT2, BT4, and BT8) at 24 microsecond intervals which is the time alloted to the CP1 and CP2 clock pulses. During the first four counts (96 microseconds) the logic signal $\overline{DTF}$ is produced via the Bit Counter logic gates to indicate the first half of a digit time. Then, during the second four-bit-counts, DTF is produced to indicate the second half of a digit time. A complete digit time then, consists of 8 bit times or 192 microseconds. The first four bits of a digit time, when the signal $\overline{DTF}$ is true, are used for shifting data in the registers. During the second half of a digit time, shifting is stopped and this period is used for the arithmetic compensation required in the Full Adder and Subtracter (FAS). Shown in Figure 4-3 is a Bit Counter Truth Table showing the state of the DTF FF's as the count progresses through 8 bit times.

DTF-4	DTF-2	DTF-1	BIT TIMES
0	0	0	BT1
0	0	1	BT2
0	1	0	BT4
0	1	1	BT8
1	0	0	BT1
1	0	1	BT2
1	1	0	BT4
1	1	1	BT8

$\overline{DTF}$ 1 Digit Time = 192 μs DTF

BIT COUNTER TRUTH TABLE

FIGURE 4-3

DC-8	DC-4	DC-2	DC-1	DIGIT TIMES
0	0	0	0	DT0
0	0	0	1	DT1
0	0	1	0	DT2
0	0	1	1	DT3
0	1	0	0	DT4
0	1	0	1	DT5
0	1	1	0	DT6
0	1	1	1	DT7
1	0	0	0	DT8
1	0	0	1	DT9
1	0	1	0	DT10
1	0	1	1	DT11
1	1	0	0	DT12
1	1	0	1	DT13
1	1	1	0	DT14

One Complete Count of Digit Counter = 2.88 ms. (1 Column Time)

DIGIT COUNTER TRUTH TABLE

FIGURE 4-4

Digit Counter. The timing of the machine requires 15 digit times in order to complete one data pass through the registers. The 15 digit times (DT0 through DT14) make up one column time which occurs each 2.88 milliseconds. No shifting occurs during DT14 which is considered as "home" time.

Digit times are generated by the four J-K flip flops that make up the Digit Counter. The Digit Counter being controlled by the two clock pulses CP1 and CP2, and by the Bit Counter; can only advance one step on each complete count of the Bit Counter. As the Digit Counter reaches DT14, a permutation occurs to step it to DT0. In Figure 4-4, is shown the Digit Counter Truth Table showing a count of 15 digit times.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

SHIFT REGISTERS

The EC 1113 has three Shift Registers designated: Indicator Register (IR), Accumulator Register (AR), and Memory Register (MR). These registers are composed of IC flip flops designed for storing numbers in a Binary Coded Decimal (BCD) system. This Binary Coded Decimal information is kept circulating through the registers in a dynamic unidirectional flow. That is, regardless of operating conditions (calculating or not calculating), clock pulses are continuously supplied to the registers so that information is always circulating in a serial bit by bit one direction flow. At timed intervals, the contents of the registers can be sampled for display or other uses, but the dynamic data flow continues uninterrupted under control of the clock pulses.

Indicator Register (IR). The Indicator Register (IR) is composed of a 14 digit shift register with each digit occupying a specific position in relation to time. That is, at a specific time when shifting is stopped momentarily (home time), each digit will have a specific register position designated: IR14 (shift information); IR13 through IR2 (contains the digits 1 through 12 for display in the number indicating tubes); and IR1 (contains decimal point information). At any other time, dynamic data flow keeps the data circulating out IR1 and into IR14, therefore, data cannot be identified during this time.

Accumulator Register (AR). The Accumulator Register (AR) is also composed of a 14 digit shift register whose digit positions in relation to time are designated: AR14 (Control information); AR13 through AR2 (digits 1 through 12); and AR1 (decimal point information). The registers AR13 through AR2 are used for storing the results of the arithmetic calculations developed in the Full Adder and Subtracter (FAS).

Memory Register (MR). The Memory Register (MR) is composed of a 13 digit shift register whose digit positions in relation to time are designated: MR13 through MR2 are used for dynamic storage of 12 digits, and MR1 is used for decimal point information.

FULL ADDER AND SUBTRACTER (FAS)

The Full Adder and Subtracter (FAS) consists of a group of logic gates designed to perform the binary addition and subtraction operations required by the arithmetic functions of: Add, Subtract, Multiply, and Divide. The two registers "IR" and "AR" (Indicator Register and Accumulator Register) are used as the inputs to FAS (y and x) with the contents of IR normally being added to, or subtracted from, the contents of AR as required by the arithmetic function being performed. The answers are accumulated in AR and then IR and AR exchange contents and the final answer appears in IR.

Data flow into and out of FAS is strictly serial in form and consists of the binary bits "0" and "1" associated with the bit times BT1, 2, 4, and 8 as binary coded decimal (BCD) information. As previously described each digit time is divided into two equal parts with each part consisting of four bit times designated BT1, 2, 4, and 8. The first half of a digit time is designated $\overline{\text{DTF}}$ and lasts for four bit times. The second half of a digit time is designated DTF and it too lasts for four bit times. In $\overline{\text{DTF}}$ time (1st half), the normal binary addition or subtraction in FAS occurs. In DTF time (2nd half), a compensation (correction) factor is applied to the output of FAS as required to correct BCD digits greater than "9". (See explanation of "Compensation" on following pages.) Output of FAS is simultaneous with the Input so no time is consumed in performing the binary arithmetic or the compensation.

The function of Adding or Subtracting in FAS is accomplished by means of a group of logic AND gates; a Carry Flip Flop (CAF); and a Compensation Flip Flop (CDF). The normal AR input to FAS is designated FAS(X) and the IR input to FAS is designated FAS(Y). A Carry is designated "C". The Basic FAS Logic Gates and Carry Flip Flop (CAF) are shown in Figure 4-5.

In any arithmetic function of FAS, the data pass occurring in DT0 through DT13 time causes the BCD contents of AR and IR to shift serially into the FAS inputs via logic gates 99, 104, and 111. The output of FAS (X and Y) in combination with the output of the CAF FF (CAF or

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

$\overline{\text{CAF}}$) is then decoded via AND gates 114, 115, 116, 117, and 118 to determine the binary sum or difference. A FAS truth table (Figure 4-10) is used to indicate the possible input/output combinations of the FAS circuit.

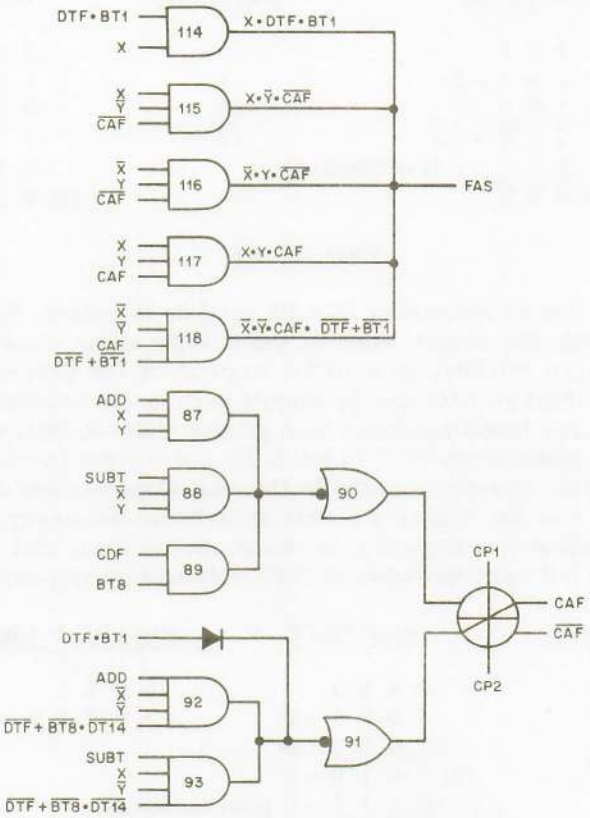


FIGURE 4-5

Carry Flip Flop (CAF)

The Carry Flip Flop (CAF) is used in conjunction with the Full Adder and Subtracter (FAS) to provide the necessary bit carries and digit carries required by the binary arithmetic being performed. The CAF FF is in a reset condition at the start of a FAS arithmetic operation.

The CAF FF set-reset enable will then depend on the function logic (Add or Subtract) and on the FAS (X and Y) logic input on gates 87 and 88; or 92 and 93.

When referring to the FAS truth tables, it should be kept in mind that the heading "Carry Enable Set" and "Carry Enable Reset" indicates the enable signal and not state of the CAF FF. When either the set enable or the reset enable is true (logic 1) at the appropriate inverter gate, the flip flop will not change state until the clock pulse CP2 occurs at the end of the Bit Time in which the enable occurred.

Compensation

Just as in the decimal system, a BCD digit cannot be displayed if it has a value greater than "9". However, since a BCD column has four bits (8-4-2-1) it is capable of retaining counts up to 15 (its 16 possible states are represented by the decimal numbers "0" through "15"). Since only 10 of these 16 states are used (0 through 9), a special correction factor must be applied at times. For example, if the sum of two BCD digits is greater than "9", a compen-

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

sation will occur, adding 6 (0110) to the answer as a correction factor. Figure 4-6, illustrates two examples of this type of correction. Notice that two things happen as a result of the correction: the answer is expressed in its correct binary form, and the carry is made to the next higher order column. The 1 in parentheses (1) denotes the carry.

DECIMAL	BCD BITS	DECIMAL	BCD BITS
	8 4 2 1		8 4 2 1
5	0 1 0 1 = 5	8	1 0 0 0 = 8
+5	+0 1 0 1 = 5	+7	+0 1 1 1 = 7
10	1 0 1 0 = 10	15	1 1 1 1 = 15
	+0 1 1 0 = 6 (Correction)		+0 1 1 0 = 6 (Correction)
	(1) 0 0 0 0		(1) 0 1 0 1 = 5

FIGURE 4-6

The same type of correction is necessary in still another situation. Figure 4-7, illustrates an addition problem in which the binary sum in the column under consideration is not greater than 9, but still requires a correction in order to produce the proper BCD character in this column. It should be noted that even though the binary sum in the column being operated upon is not greater than 9, the carry itself denotes a sum greater than 9. Still referring to Figure 4-7, note that a carry takes place from bit 1 to bit 2. No correction is necessary here. However, when the binary arithmetic causes a carry to the next higher order digit column, the binary sum is greater than 15 and the Carry FF sets to indicate the carry. Therefore, since 15 is greater than 9, a correction is necessary. It can be stated then, that carries between bits do not require a correction, but carries between digit columns do require a correction.

DECIMAL	BCD BITS	BINARY WEIGHT
	8 4 2 1	16 8 4 2 1
9	1 0 0 1 = 9	1 0 0 1 0 = 18
+9	+1 0 0 1 = 9	
18	(1) 0 0 1 0 = 2	
	+0 1 1 0 = 6 (Correction)	
	1 0 0 0 = 8	

FIGURE 4-7

The reason for adding 6 (0110) can easily be explained: it was noted in decimal addition and in regular binary addition that when a column becomes "full" it is returned to "0" and a carry is made to the next higher order column. In the BCD system, the four bits are capable of retaining any count up to 15. Therefore, if a 1 is added to a 9 and a carry is made, 6 more must be added to the column in order to return it to "0". The same conditions apply when doing binary subtraction except that the correction factor 6 (0110) is subtracted instead of being added.

Compensation Flip Flop (CDF)

When the sum of two digits requires a compensation, the Compensation Flip Flop (CDF) must be set in order to provide the enabling signal to gate 112 for the binary 6 (0110) that is added or subtracted during a compensation. Two gates are provided for the set enable of CDF: These are 169 and 171. Gate 171 is enabled when the sum of two digits is greater than "9" but less than 16 as follows: In the 2nd half of every digit time (except DT0 and DT14, or when the BCF FF is set) the signal DTF.BT1 is true and one leg of the gate 171 is enabled. Then, if the addition occurring in FAS during the 1st half of the digit time causes an output of 10 through 15, Bit 2 or 4, and Bit 8 will contain a binary 1. For example, if the sum is 10, then Bit 2 and Bit 8 will contain a binary 1 and these 1'a are sensed at the parallel bit output of AR14. Thus, gate 171 is enabled and the set enable for the CDF FF is true. When the CP2 pulse occurs at the end of BT1, CDF will set. Gate 112 will then be enabled to provide the binary 6 (0110) compensation. Any combination of FAS outputs from 10 through 15 will enable one input to gate 171.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

Gate 169 also enables the CDF FF to set under the following conditions: In the 2nd half of every digit time (except DT0 and DT14 or when the BCF FF is set) the signal DTF.BT1 is true and the bottom leg of gate 169 is enabled. Then, if the addition occurring in FAS produces an output greater than 15, a compensation is required because 15 is greater than 9. However, in this case, the binary sum in the column being operated upon is not greater than 9, but the addition of X and Y in the BT8 column causes a carry (1) to the BT1 column in the next higher order and therefore denotes a binary sum greater than 15 (see example in Figure 4-8). The set enable for the CAF FF is produced in BT8 time and CAF FF will set when CP2 occurs at the end of BT8. Then, in BT1 time of the 2nd half of the digit time, the set enable for the CDF FF is produced via gate 169. When CP2 occurs at the end of BT1, CDF FF will set and the binary 6 (0110) compensation will occur via gate 112.

NOTE: The CAF FF will reset via DTF.BT1 at the same instant that the CDF FF sets to insure proper compensation. At the end of the compensation, the CDF FF will reset and at the same instant, gate 89 will enable the CAF FF to set, indicating a carry to the next higher order.

Compensation example using sum greater than 15. CDF FF will set via gate 169.

	DIGITS IN							
	ADD OR SUBT $\overline{DTF}$				COMPENSATE DTF			
BIT TIME	1	2	4	8	1	2	4	8
X INPUT	0	0	0	1	0	0	0	0
Y INPUT	0	0	0	1	0	1	1	0
CARRY FF	0	0	0	0	1	0	0	0
OUTPUT SUM	0	0	0	0	0	1	1	0

BINARY 6 COMPENSATION
(1) CARRY TO NEXT HIGHER DIGIT
ENABLE CDF FF

SUM GREATER THAN 15

FIGURE 4-8

Compensation example with sum greater than 9, but less than 15. CDF FF will set via gate 171.

	DIGITS IN							
	ADD OR SUBT $\overline{DTF}$				COMPENSATE DTF			
BIT TIME	1	2	4	8	1	2	4	8
X INPUT	1	0	1	0	0	1	0	1
Y INPUT	1	0	1	0	0	1	1	0
CARRY FF	0	1	0	1	0	0	1	1
OUTPUT	0	1	0	1	0	0	0	0

ENABLE SET OF CDF FF

BINARY 6 COMPENSATION

(1) CARRY TO NEXT HIGHER DIGIT

SUM GREATER THAN 9
BUT LESS THAN 15

FIGURE 4-9

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

Block Compensation

Because we never need to compensate during DT0 (decimal point information), nor during DT14 (home time), nor when the BCF FF is set (adding "1" to AR2 data in a Divide operation), gates 169 and 171 are qualified by the signal: $\overline{DT0}.\overline{DT14}.\overline{BCF}$. Therefore, during any of the above conditions, at least one of the above signals will be false and inhibit the gate, thereby preventing a compensation.

Special FAS Gates

FAS GATE 89 (CDF.BT8): Gate 89 enables the CAF FF to set which causes a carry into the next higher digit when the compensation occurs.

FAS GATE 114 (X.DTF.BT1): The purpose of this gate is to insure a "1" output in FAS Compensation BT1 time when the column contains $X.\overline{Y}.C$ (1-0-1). This condition arises only when the FAS "Add or Subtract" output contains a "1" in the BT1 column and the CAF FF is enabled set in BT8 time.

FAS GATE 118 ($\overline{X}.\overline{Y}.C.\overline{DTF}+\overline{BT1}$): The purpose of this gate is to insure a logic 1 out of FAS when the signal $\overline{X}.\overline{Y}.CAF$ is at a logic 1, except during the second BT1 of each digit time.

OR GATE SIGNAL DTF.BT1: The signal DTF.BT1 below gate 93 insures that the CAF FF will be reset before a compensation occurs.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

ADD / SUBTRACT

X	Y	CAF	FAS OUT	ADD Carry SET	ADD Enable RST	SUBTRACT Carry SET	SUBTRACT Enable RST
0	0	0	0	0	1	0	0
0	0	1	1	0	1	0	0
0	1	0	1	0	0	1	0
0	1	1	0	0	0	1	0
1	0	0	1	0	0	0	1
1	0	1	0	0	0	0	1
1	1	0	0	1	0	0	0
1	1	1	1	1	0	0	0

ADD COMPENSATION

X	Y	CAF	FAS BIT BEING ADDED				CARRY ENABLE SET				CARRY ENABLE RESET			
			1	2	4	8	1	2	4	8	1	2	4	8
0	0	0	0			0	0			1	1			0
0	0	1	0			1	0			1	1			0
0	1	0		1	1			0	0	1	1	0	0	0
0	1	1			0				0	1	1		0	0
1	0	0	1			1	0			1	1			0
1	0	1	1			0	0			1	1			0
1	1	0		0	0			1	1	1	1	0	0	0
1	1	1			1				1	1	1		0	0

BLANK AREAS INDICATE IMPOSSIBLE CONDITIONS IN COMPENSATION

SUBTRACT COMPENSATION

X	Y	CAF	FAS BIT BEING SUBTRACTED				CARRY ENABLE SET				CARRY ENABLE RESET			
			1	2	4	8	1	2	4	8	1	2	4	8
0	0	0	0			0	0			1	1			0
0	0	1	0			1	0			1	1			0
0	1	0		1	1			1	1	1	1	0	0	0
0	1	1			0				1	1	1		0	0
1	0	0	1			1	0			1	1			0
1	0	1	1			0	0			1	1			0
1	1	0		0	0			0	0	1	1	0	0	0
1	1	1			1				0	1	1		0	0

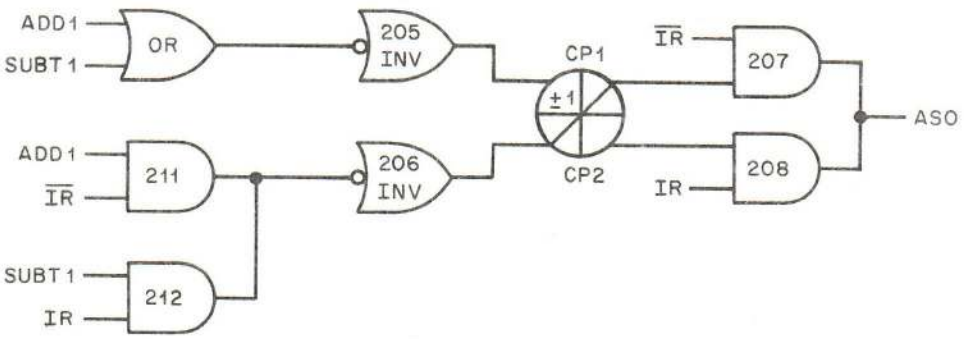
FAS TRUTH TABLES

FIGURE 4-10

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

ADD OR SUBTRACT "1" (ASO)

The Add or Subtract "1" (ASO) circuitry is used in conjunction with the Indicator Register (IR) for Adding or Subtracting "1" to or from the decimal point or shift information in IR. It is also used for normal data shift of IR. The contents of IR flow through ASO in a serial bit by bit manner without change except when enabled to Add or Subtract "1" via machine logic associated with decimal or shift information. The enabling logic for adding or subtracting "1" via ASO causes the ±1F to be set via gate 204. Then, as shown in the basic ASO circuit, Figure 4-11, as the shift or decimal data is gated into ASO a "1" is added or subtracted as follows:



BASIC ASO CKT
FIGURE 4-11

Add "1"

Decimal or Shift data contains four binary bits occurring in the first half of DT0 and DT13 time respectively. Depending upon the number, each binary bit (8-4-2-1) will be a logic 1 or 0 and is gated into ASO via gates 207, 208, 211, and 212 as: $\overline{IR}$ or IR. With the ±1F set and the $\overline{IR}$ true (binary 0 at the input), the output of ASO (gate 207) will be true or logic 1 indicating that a "1" has been added to "0". With $\overline{IR}$ false (binary 1 at the input), the output of ASO (gate 207) will be false or logic 0. However, at the first time gate 207 is enabled, gate 211 will also be enabled and the ±1F will reset before the next bit time. With the ±1F reset, any IR input will go through ASO without change. A Truth Table showing all possible Add conditions is shown in Figure 4-12.

ADD 1

IR	±1	ASO	±1 RESET ENABLE
0	0	0	1
0	1	1	1
1	0	1	0
1	1	0	0

	BT 1	BT 2	BT 4	BT 8
IR	0	0	0	0
±IF	1	0	0	0
ASO	1	0	0	0

SAMPLE PROBLEM

FIGURE 4-12

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

Subtract "1"

A Subtract "1" from IR1 is quite similar to an Add "1" as just described, except that the Subtract "1" only occurs under certain machine logic conditions such as: Mult, Divide, or Shift Right operations. To describe a Subtract "1" operation, we will assume that the Shift key is to be used and that IR contains an entry with four digits to the right of the decimal point as: 2.1234. Before the shift key is depressed, IR1 contains a binary 4 (0010). After the Shift key operation, IR1 will contain a binary 3 (1100) and the contents of IR will appear as: 2.123. A truth table showing all possible conditions for a Subtract "1" is shown in Figure 4-13. Also shown in Figure 4-13, is a table showing the sample problem described in the text.

SUBT 1

IR	±1	ASO	±1 RESET ENABLE
0	0	0	0
0	1	1	0
1	0	1	1
1	1	0	1

	BT 1	BT 2	BT 4	BT 8
IR	0	0	1	0
±IF	1	1	1	0
ASO	1	1	0	0

SAMPLE PROBLEM

FIGURE 4-13

Depressing the Shift key, the logic signal: Key Sh.OSF.$\overline{IR1 = 0}$ will be produced as a true (logic 1) at gate 202 and enable the ±1F to be set at the end of DT14 time via gates 203 and 204. Then in DT0 time, IR1 is shifted through ASO via the IR input and the first bit of the binary 4 (0) produces a binary 1 output from ASO via gate 207. ("0" input is inverted by Inverter gate 210 and the ±1F is in a set condition enabling gate 207 with a true at both inputs.)

At BT2 time, the second bit of the binary 4 (0) is shifted through ASO via the IR input and since the ±1F is still set, gate 207 will again be enabled and the ASO output will be a "1". In BT4 time, the third bit of the binary 4 (1) is shifted through ASO via the IR input, but this time, gate 207 is inhibited by the false output (0) from the Inverter gate 210 and the ASO output will be "0". However, during BT4 time, gate 212 is enabled by the "1" input to ASO and logic 1 being true by: Key Sh.OSF. The ±1F is then reset at the end of BT4 time by the CP2 clock pulse.

In BT8 time, the fourth bit of the binary 4 (0) is shifted through ASO and since the ±1F is now reset, the ASO output will be the same as the input or "0". The binary 4 (0010) passing through ASO has thus been decreased by "1" and IR1 now equals binary 3 (1100). The remaining contents of IR are shifted through ASO without change.

MASTER FUNCTION FLIP FLOPS (J-K TYPE)

Basically the Master Function Flip Flops are used to set up control commands for the entire machine's logic system. Operation of the J-K (master-slave) type flip flops is dependent on a two clock pulse system generated by the master oscillator, and the clock pulse generator.

NOTE: Due to the internal construction of the IC modules containing the J-K flip flops and shift registers, the circuit operation requires -24 volts. This -24 volts is supplied via the clock pulses but is not considered as a logic level.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

CONDITIONAL CONTROL PHASE COUNTER

The Conditional Control Phase Counter has four J-K flip flops whose gated output signals CC0 through CC9 are used to control the logic functions that must occur during each phase of a machine arithmetic function. Starting with the counter at CC0, it will remain in that state via logic signals at the input gates until the required logic action called for in that phase of the machine arithmetic function is completed. The input logic to the counter will then allow it to step to CC1 and again it will be held in this state until the necessary logic action for this phase is completed. However, if no logic action is required in a particular step, the input logic to the counter will cause it to advance to the next higher step. When the counter reaches CC9, the machine arithmetic function being performed is completed and then the input logic to the counter permutes it causing an automatic step to CC0.

ENTRY CIRCUIT

The Entry Circuit is composed of 10 numeral key reed switches, one decimal point key reed switch, and 12 function key reed switches. Depression of any key closes its respective reed switch which is decoded via a diode matrix into the appropriate logic signal or signals required for that particular key. These logic signals are then used to cause the required machine logic action.

DISPLAY CIRCUIT

The Display Circuitry decodes the Binary Coded Decimal (BCD) information arriving from the Indicator Register so the number and/or decimal point will be displayed in the correct number indicating tube. The number decoder circuit is controlled by the output from the Indicator Register IR1. The output from the number decoder matrix energizes the same number cathode in all twelve number indicating tubes simultaneously. The Anode circuitry controlled by the Digit Counter, energizes a single anode in the proper number indicating tube at the same instant its cathode is energized. The number indicating tube will then conduct and display the number associated with a particular cathode. The electronic operation of the circuit is covered in Section 5 (Electronic Circuits).

POWER SUPPLY

The Power Supply converts the AC line voltage into the DC voltages necessary to operate the machine. There are four DC outputs developed directly by the Power Supply circuit on PC card 5. They are: VGG = -24 volts (Clock Pulses), VDD = -14 volts (logic), +103 and -103 volts (Display Circuits). Two other voltages are developed on PC card 4 by means of a voltage divider. They are: VEES = -11 volts and VEE -5 volts used for display circuits. E = "0" volts (ground).

The Power Supply regulation circuitry is covered in Section 5 (Electronic Circuits).

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

POWER SUPPLY

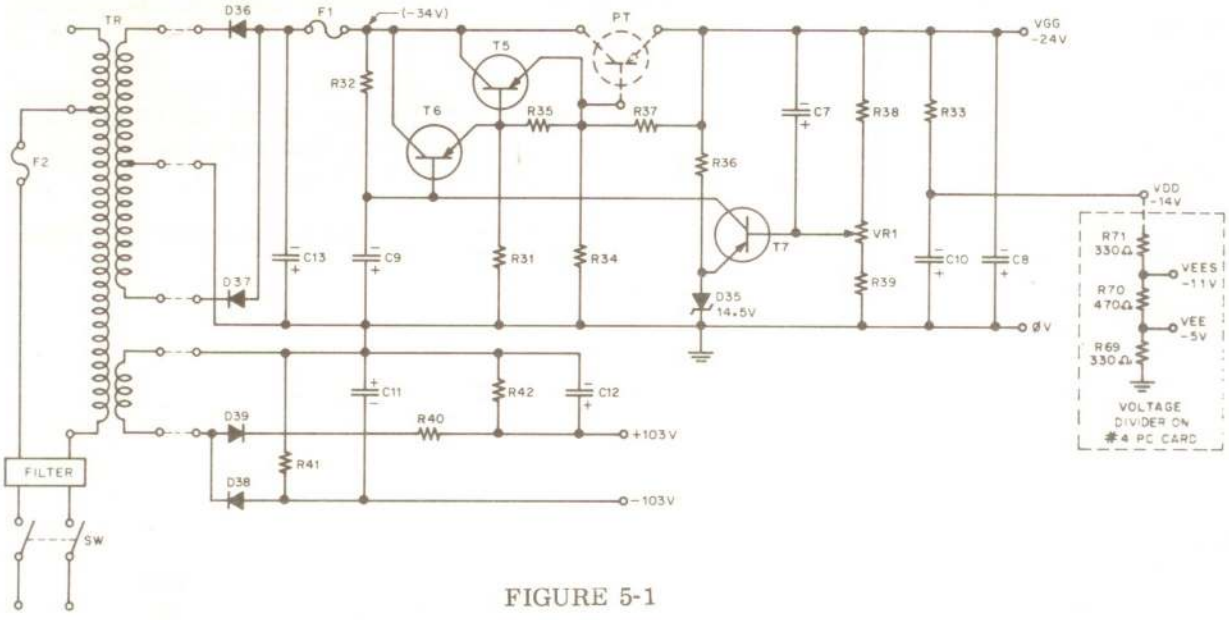


FIGURE 5-1

There are four DC outputs developed directly by the power supply circuit on PC card #5. They are: VGG -24 volts, VDD -14 volts, +103 and -103 volts (used for the display circuitry). As shown in Figure 5-1, a full wave rectifier and filter circuit provides an output of approximately -34 volts DC to the voltage regulator circuit. The power transistor PT (mounted on a heat sink at the rear of the chassis) is in series with the -34 volts and it drops approximately 10 volts, leaving an output of -24 volts at the VGG output. The -14 volts supply (VDD) is developed on PC card #5 via resistor R33. A voltage divider tapped at R33 continues on PC card #4 where two other voltages are developed. They are: VEES -11 volts, and VEE -5 volts.

A dual half-wave rectifier circuit supplied by a separate secondary in the Power Transformer provides the +103 and -103 DC voltages required by the display circuitry.

REGULATION

Because the IC chips used in the 1113 are highly susceptible to damage if they are subjected to overvoltage, the -24 volt (VGG) supply must be highly regulated. Since the -14, -11, and -5 volt supplies are developed via a voltage divider supplied by the -24 volts, regulation of the -24 volts effectively regulates the other voltages as well.

As previously stated, the output voltage of the full wave rectifier ahead of the regulator circuit provides approximately -34 volts DC. Ten volts are dropped across the series regulator transistor PT leaving -24 volts at the VGG output. A 1K potentiometer designated VR1 provides a means for precisely adjusting the -24 volts. The zener diode, D35 connected between the emitter of T7 and ground provides a precise -14.5 volt reference.

In operation, when the output voltage tends to change due to a decrease or increase in load demands, the change is detected by transistor T7. For example, if the load resistance causes a drop toward -23 volts, the base of the transistor T7 goes more positive than before, and this positive going change is inverted by T7 so its collector tends to go more negative. The transistor T6 feels the negative going change on its base and being an emitter follower, its emitter tends to go negative. Likewise, transistor T5 feels the negative going change on its base and it being an emitter follower, its emitter tends to go more negative. Finally the negative going change is felt on the base of the power transistor PT causing it to turn on harder, and it being an emitter follower also, the output on its emitter tends to go more

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

negative which causes an increase in voltage toward the -24 volts. If the values of the components are correct, a drop of 1 volt at the load will cause the power transistor PT to conduct harder which increases the output by 1 volt so the -24 volts remains constant. The emitter followers, T5 and T6 used between T7 and PT are necessary for current gain.

The converse is also true. If the output tends to increase toward -25 volts, transistor T7 is turned on harder and its collector tends to go more positive, which causes T6 and T5 to follow by going more positive. The PT transistor feels the positive going change on its base and it conducts less which causes the output to go more positive toward -24 volts.

NOTE: Power supply rectifier circuits, filters, and regulators are explained in the Friden Correspondence Lesson #17.

MASTER OSCILLATOR

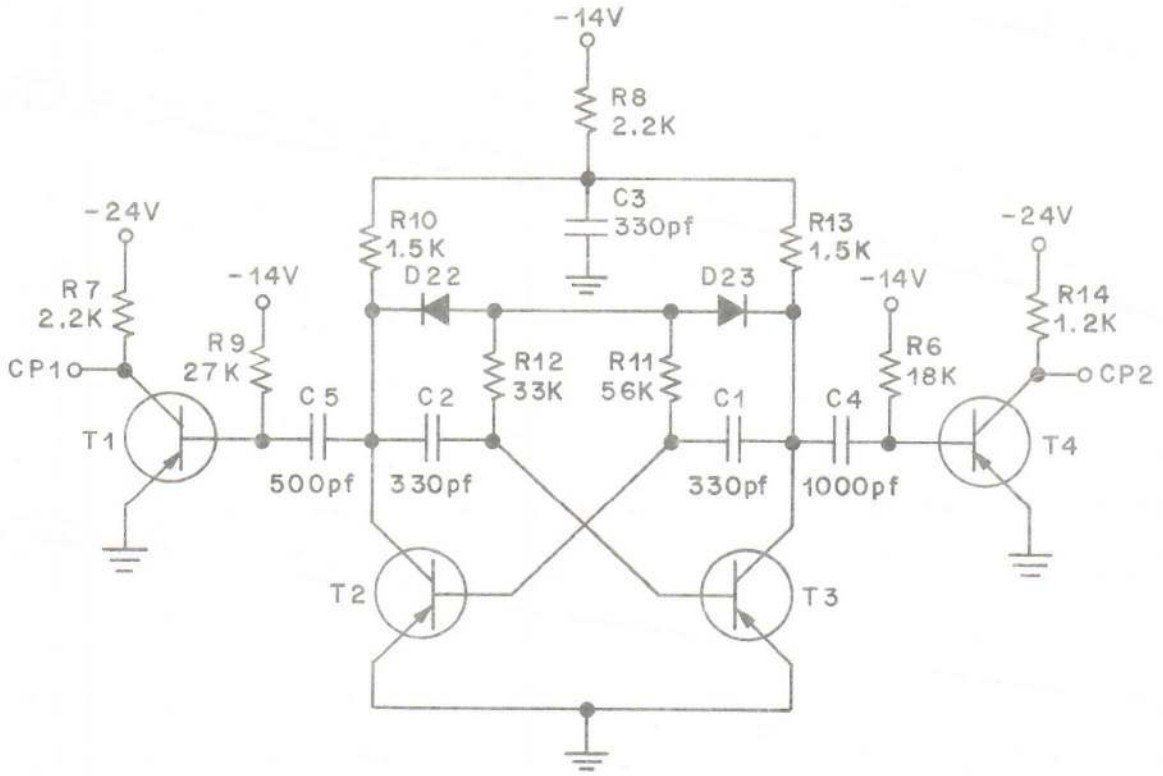


FIGURE 5-2

As shown in Figure 5-2, the basic timing circuit is composed of an unsymmetrical astable multivibrator operating at a frequency of approximately 42 KHz. The operation of this type multivibrator can be found in Lesson 19 of the "Friden Electronics Correspondence Course". The output from each side of the multivibrator is shaped for the proper pulse width and pulse amplitude by the "shaping" circuits consisting of transistors T1 and T4 and associated components. The outputs from the shaping circuits designated CP1 and CP2 occur once every 24 μsec., and are out of phase with each other. They are used to count the Bit Counter which develops Bit Times, with each Bit Time being equal to 24 μsec. They are also used in conjunction with the Bit Counter to count the Digit Counter once every 8 Bit Times. A Digit Time is equal to 8 Bit Times or approximately 192 μsec.

NOTE: The 1113 timing is described in greater detail in Section 4.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

MASTER OSCILLATOR (Cont.)

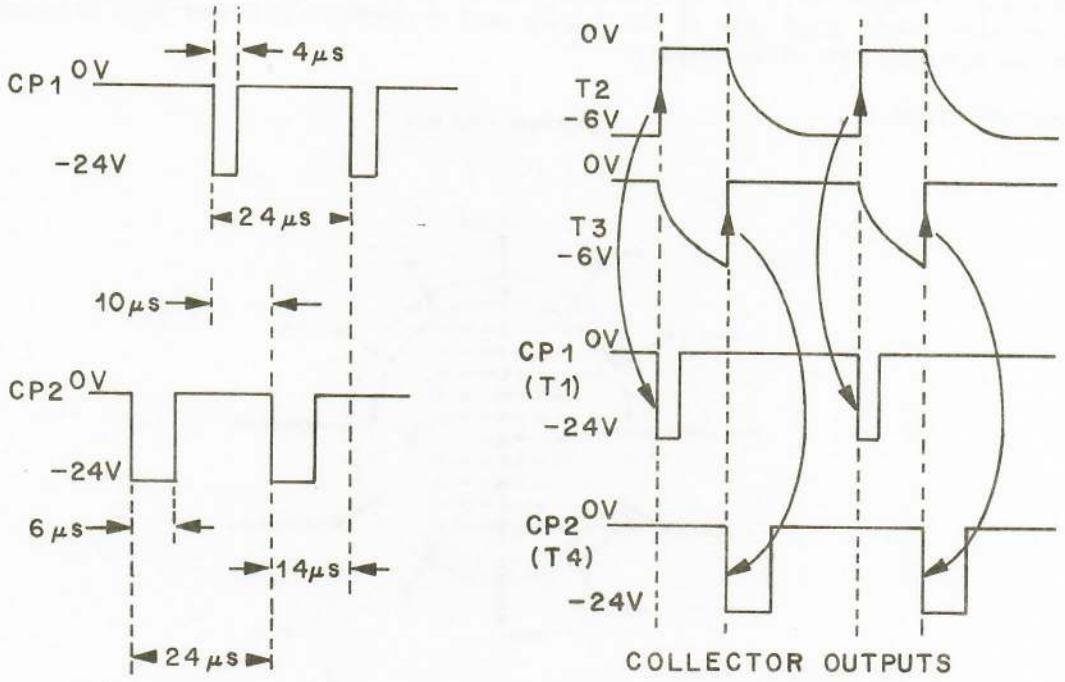


FIGURE 5-3

To further explain the operation of the Oscillator Circuit, assume that the machine has just been turned on and that the two PNP Inverter transistors T1 and T4 are both on at this time due to the negative voltage applied to their respective bases. Also assume that one of the transistors in the multivibrator is turned on and the other is off. For example, T1, T3, and T4 are on and T2 is off. Since the Collector of T2 is at approximately -6 volts due to the voltage divider network, the base of T1 is at approximately -1 volt and therefore capacitor C5 will charge to approximately 5 volts. Then, through the multivibrator action of the oscillator circuit, T2 will turn on causing a positive 4 volt spike to be felt on the base of T1 which will immediately cut T1 off causing its collector to go to -24 volts. T1 will remain cut off until capacitor C5 discharges to approximately -1 volt. This period of time is determined by the RC time constant in the base circuit of T1 (approximately 4 microseconds). When T1 conducts again, its collector will return to zero volts and will remain in conduction until T2 in the multivibrator makes its transition from off to on. Thus, the output designated CP1 has a 4 microsecond pulse width, a -24 volt amplitude, and occurs once every 24 microseconds.

The operation of the inverter circuit T4 is the same as that of T1 except that the RC time constant in the base circuit of T4 is such as to cause CP2 to have a pulse width of 6 microseconds. It should be noted that the inverter circuits (T1 and T4) will cut off only when their associated transistors (T2 and T3) in the multivibrator make their transition from off to on. The clock pulse outputs of CP1 and CP2 will therefore be out of phase with each other because the two outputs of the multivibrator are out of phase.

The timing relationships of CP1 and CP2 (T1 and T4) to their corresponding oscillator transistors T2 and T3 are shown in Figure 5-3. Notice that each positive going collector output of T2 and T3 causes a corresponding negative going collector output on T1 and T4. The relative pulse width and amplitude of CP1 and CP2 are also shown in Figure 5-3.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

DISPLAY CIRCUIT

The 1113 display consists of a row of 12 nixie type, number indicating tubes capable of displaying 12 digits or up to 11 decimal positions. A minus sign indicator light is located at the extreme right hand side of the display and an overflow indicator light is located at the extreme left hand side of the display.

CONSTRUCTION

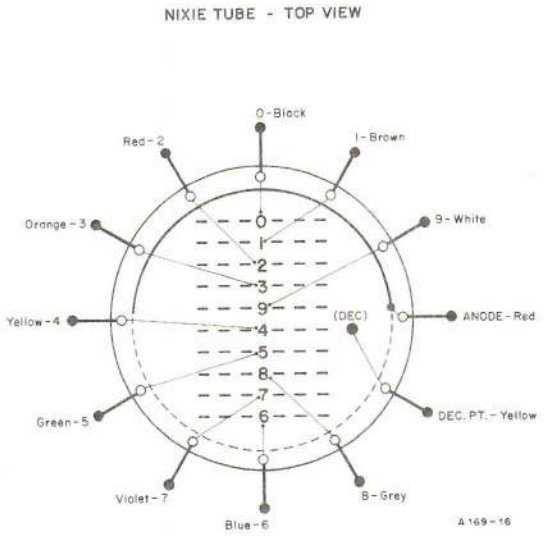


FIGURE 5-4

As shown in Figure 5-4, each number indicating tube contains a single anode element and eleven cathode elements encapsulated inside a glass tube filled with neon gas. The cathodes consist of fine wires formed in the shape of the numerals "0" through "9" and a decimal point (.). The digit cathodes are arranged in a row, one in front of the other, and the decimal cathode is located to the right of the #4 cathode. The anode is designed so it encircles the cathodes and provides the positive potential to the cathodes. The leads from the individual cathodes and the anode are brought out at the bottom of the tube for connection to the #4 PC board.

OPERATION

During each data pass, the entire contents of the indicator register (IR) shifts through IR1 in step with the digit counter, and is sequentially displayed in the 12 number indicating tubes. Since the contents of each register position (IR1 through IR14) is associated with a digit time, then during DTO time, the binary number associated with IR1 (which is decimal point information and is not to be displayed) is shifting out of the IR1 register position. In DT1 time, the binary number associated with IR2 is shifting out of the IR1 register position, and so forth for each of the register positions until finally in DT12 time, the binary number associated with IR13 (most significant digit position) is shifting out of the IR1 register. In DT13 time, the number associated with IR14 (shift information) is shifting out of IR1 but is never displayed. In DT14 time, the digit counter has reached its highest count which is considered as "home" time and no shifting or display occurs during this home time interval.

CIRCUIT DESCRIPTION

The number indicating tube energize circuit consists of applying a positive potential of approximately 200 volts (175 to 200) to the anode of the number indicating tube to be turned on; and at the same time, applying a ground potential to the selected cathode of all 12 number indicating tubes. Therefore, only the number indicating tube whose anode is energized will turn on, displaying the number associated with the selected cathode.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

Anode Circuit: As shown in Figure 5-5, the energize circuit for the anode of each number indicating tube consists of: a capacitor, a transistor switch, and an "Anode Decoder" circuit (which consists of diode OR gates operating as negative AND gates).

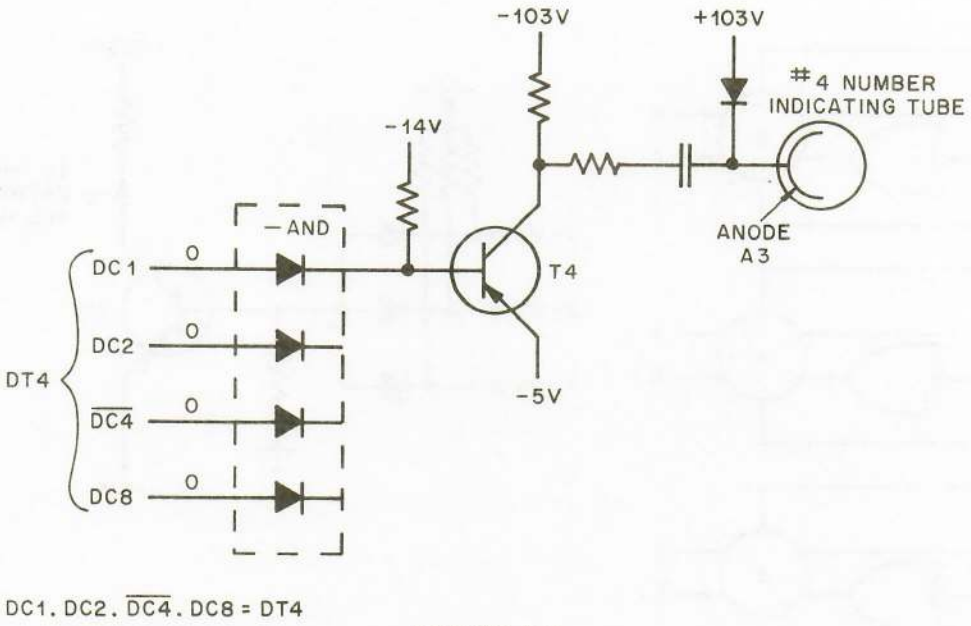


FIGURE 5-5

Any time an input to the Anode Decoder circuit for a particular anode is at a logic 1 ("0" volts), the corresponding transistor associated with that number indicating tube will be off and its capacitor will charge to approximately 206 volts. As the Digit Counter counts, the logic 0 (-14 volts) output of the Digit Counter FF's is applied to the inputs of the Anode Decoder circuits (negative AND gates). Thus, each count of the Digit Counter is decoded via the negative AND gates and each Anode circuit is enabled in sequence and in step with the Digit Counter.

In the example shown in Figure 5-5, the Digit Counter is at a count of four and the logic 0 output from the four Digit Counter FF's (DC1. DC2. $\overline{DC4}$. DC8) is applied to the negative AND gates corresponding to the #4 anode energize circuit. The -14 volts (logic 0) is thus felt on the base of transistor T4 (#4 Anode) causing the transistor to turn on. As the transistor turns on, its collector will go to approximately -5 volts because the emitter is clamped to -5 volts. This amounts to a change of 98 volts in the positive direction being felt on the charged capacitor. Therefore, the other side of the capacitor will feel the same 98 volt positive change which added to the +103 volt charge develops approximately 200 volts positive on the Anode of the #4 number indicating tube causing it to turn on.

Number Cathode Circuit: The energize circuit for the number cathodes consists of: a transistor switch for each number cathode ("0" through "9"), and a cathode number decoding circuit at the output of four J-K flip flops designated as "Number Cathode" FF's.

During the 1st half of each digit time (except DT14), the contents of IR are shifted right one position so each number will eventually arrive in IR1 at a particular digit time. The parallel outputs (IR1-8, IR1-4, IR1-2, and IR1-1) from IR1 are used for the set inputs to the Number Cathode FF's, and the inverted signals ($\overline{IR1-8}$, $\overline{IR1-4}$, $\overline{IR1-2}$ and $\overline{IR1-1}$) are used for the reset inputs to the flip flops.

These flip flops are normally in the set condition. Then, when the DFCP1 clock pulse occurs at BT8 of the 2nd half of every digit time, the Number Cathode FF's are enabled to set or reset depending upon the condition of their enable inputs. The actual transition of the FF's occurs when the CP2 clock pulse occurs at the end of every digit time. This same CP2 pulse

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

also causes the Digit Counter to step to the next higher count (Digit Time) which enables the Anode energize circuit as previously discussed. The number cathode will then be decoded from the true (logic 1) outputs of the number cathode FF's.

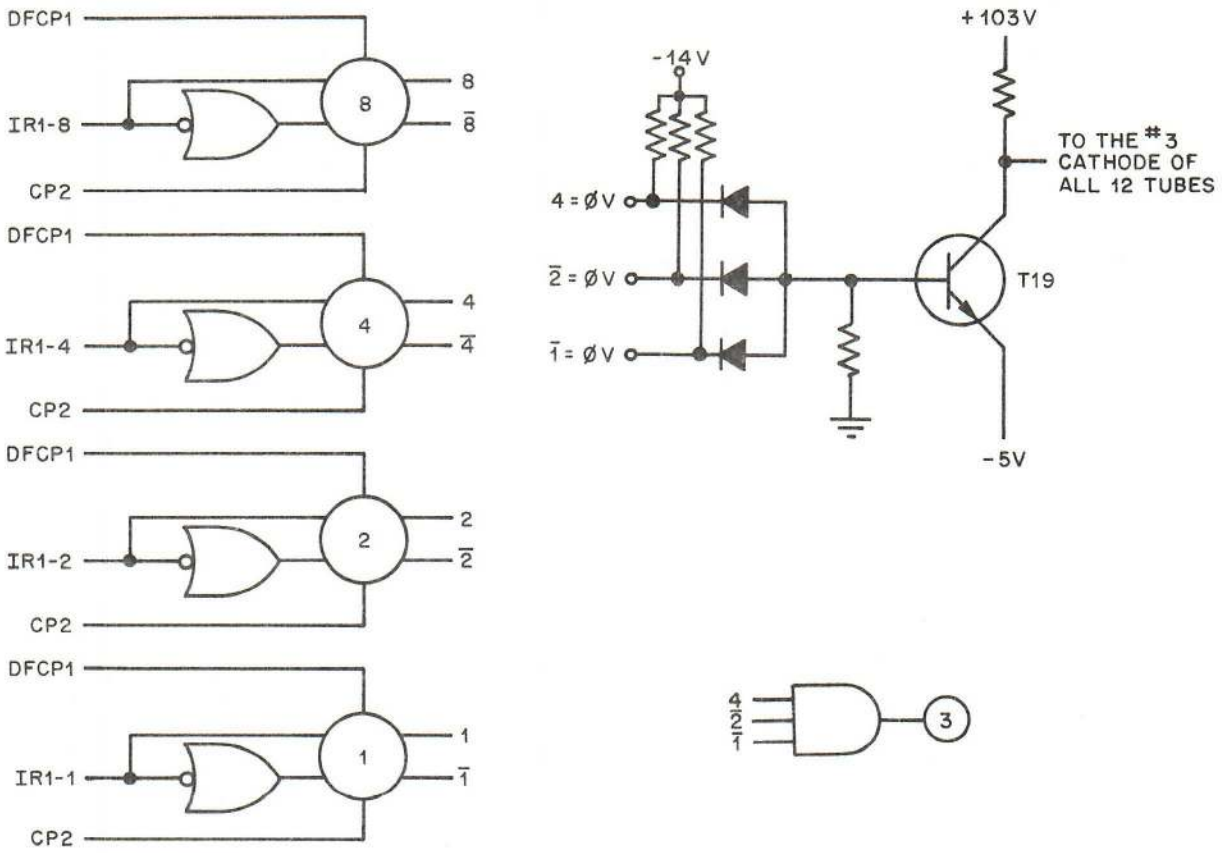


FIGURE 5-6

For example, as shown in Figure 5-6, if the number in IR1 is a "3", then the true output of the four number cathode FF's after the transistion occurs will be #1 FF reset; #2 FF reset; #4 FF set; and #8 FF set; or $\bar{1}$-$\bar{2}$-4-8.

With a true at all inputs of the decoder AND gates, a true ("0" volts) is applied to the base of the transistor switch corresponding to the number cathode to be energized. As the transistor turns on, its collector going to approximately -5 volts is felt on the #3 cathode in all 12 display tubes. At this same instant, the Anode of only one display tube is energized at a potential of approximately 200 volts positive and the neon gas surrounding the #3 cathode glows brightly outlining the numeral 3 inside the tube.

As stated previously, the number cathode FF's are normally in the set condition with no information in the registers. Therefore, after the machine has been cleared, only the "0" cathodes will be enabled via the decoder AND gates. All other transistor switches associated with the cathodes will be held off with (-14 volts) logic 0 on their bases.

Decimal Point Cathode Circuit: The energize circuit for the decimal point cathodes is similar to the number cathode circuit previously discussed. The circuit consists of: a transistor switch for each number indicating tube, and a decimal point decoder circuit at the output of four J-K flip flops designated as "Decimal Point Cathode" FF's.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

At each DT14 time, the decimal point information in IR1 is sensed at the J-K (set-reset) enable inputs to the decimal point cathode FF's which are normally in the set condition. Then when the clock pulse CCCP1 occurs during the last half of DT14, the decimal point flip flops will be enabled to set or reset depending upon the condition of the enable inputs. After the decimal point cathode FF's have made their transition, their combined state (count) will equal the decimal point information in IR1.

The count in the decimal point cathode FF's is then applied to the inputs of the decimal point decoder circuits (AND gates) and the true output (logic 1) from one of these gates is applied to the base of the transistor corresponding to the decimal point cathode to be energized. All other decimal point cathodes are held off because of the false (logic 0) output from the other decimal point decoder circuits.

As previously stated, the decimal point cathode FF's are initially in the set condition. The reason for this is to insure that only the #1 decimal point cathode is energized after the machine is cleared. All other transistor switches associated with the decimal point cathodes are held off with a logic 0 (false) on their bases.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

GENERAL

This section of the manual includes brief, concise descriptions of the logic action of each major function of the EC 1113, keyed to the corresponding Flow Chart of that same function. As applicable, the initiating gate, or gates are included alongside the logic operation on the Flow Chart. These gate numbers correspond to the gate identification numbers shown on the EC 1113 Logic Block Diagram (Rev. 2). The purpose of the flow charts and accompanying explanations is to provide backup to the instruction, study material, and memory refreshment during servicing. They are not necessarily written to the completeness that is normally required for self-teaching of the EC 1113 Calculator.

The Function Logic concerns the operations that take place as a result of the depression of the Function and Digit Keys on the keyboard. Each function is interpreted into a Flow Chart and explained on the basis of the primary actions noted on the chart. Once the function action is specified by the keyboard logic signals, the appropriate master function FF (OSF, MLF, DIF, etc.) in conjunction with the Conditional Control Phase Counter (CC0 through CC9) controls the sequence of the actions which result in the arithmetic function being performed. Specific logic actions of an arithmetic function are controlled by stepping the Conditional Control Phase Counter (CC0 through CC9) and can occur only with respect to those steps. Each logic analysis description is keyed to the corresponding CC step on the Flow Chart.

ENTRY FUNCTION

An Entry Function is the placing of numbers into the Indicator register (IR) via the manual depression of the digit keys. Entry of digits can be as rapid as the operator can depress the keys with the maximum number of digits that can be entered limited to 12. Decimal point entry is by means of depressing the decimal point key. Decimal point alignment in the Indicator Register is automatic via the accumulative count in IR1 of each digit entry following the decimal key depression.

The first digit entry goes into register IR via IR14 and then a series of right shifts occur causing the digit to move through the register until it reached IR2. During this series of right shifts, the contents of IR are being shifted into AR for later use. Any data in AR is shifted out and destroyed by the entry. When the 2nd digit entry occurs, it enters the register IR via IR13. In the meantime, however, the digit in IR2 (1st digit) has shifted around and entered into IR14 so the 1st digit entry is again in IR14 and the 2nd digit entry is in IR13 which is their proper order. A series of right shifts will again occur moving both digits through the register until they reach IR2 and IR3. This sequence continues for as many digits being entered up to the maximum of 12 digits. The decimal point key entry causes a "1" to be added to IR1 for each digit entry following the decimal point entry. IR1 will thus accumulate a total which indicates the decimal position. A decimal point key entry as a first entry will cause the same actions as a first digit key entry.

As shown in Figure 6-1, each digit key depression causes its associated digit to be converted to binary coded decimal (BCD) form via the "Entry Function" OR gates. Also shown is the BCD output from the AND gates 4, 5, 6, and 7.

ENTRY FLOW CHART ANALYSIS

The Entry operations are explained in conjunction with the Entry Flow Chart, Figure 6-2 as follows:

CC0 1ST DIGIT ENTRY

The Conditional Control Phase Counter (CC) will remain at CC0 for the entire Entry Function. In this step, the logic functions are controlled by the digit key logic signals, the Bit Counter, the Digit Counter, and the necessary Clock Pulses.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

BINARY CODED DECIMAL LOGIC

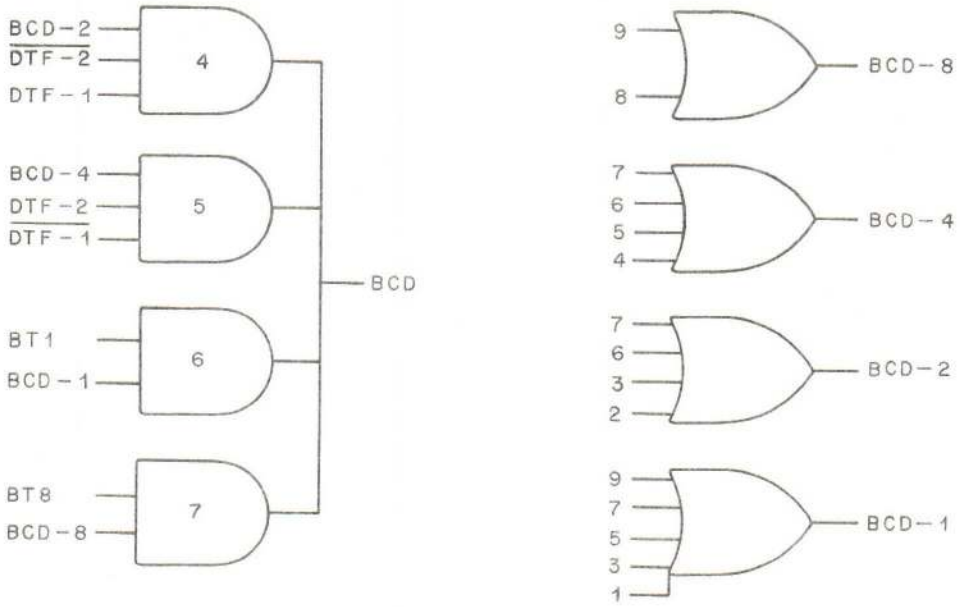


FIGURE 6-1

DEPRESS DIGIT KEY

Depressing a digit key establishes a logic 1 to the logic signals: Key N, Key (N+P), BCD, and Key. These logic signals will be used primarily for control of the Master Function FF's required during the entry operation.

DT14 (Set OSF and KEF)

With the digit key depressed, the $\overline{OSF}$ FF is set via gate 12. The KEF FF is then enabled to set via the logic signal: $\overline{OSF}$ going false (logic 0) after the OSF FF sets.

DTO (Shift AR & IR Right One place; IR1 into AR14)

In DTO, the contents in AR and IR are shifted right one data shift position. This causes IR1 to shift into AR14 and AR1 to shift out and be destroyed because it has no place to go.

Data flow from IR1 to AR14 is via the Full Adder and Subtractor (FAS). IR1 input to FASy is via gates 104 and 111. AR input to FASx via gate 99 is blocked because gate 96 is false; therefore, FAS will add IR1 to "0". The output of FAS goes into AR14 via gates 173 and 174.

DT1 (Shift AR & IR Right One place; IR1 to AR14; New Digit to IR14)

Right shift of IR1 to AR14 same as described for DTO. At this time, however, the 1st depressed digit goes into IR14 via gate 157.

DT2-13 (Shift AR & IR Right One Place Each Digit Time; IR to AR)

During DT2-13, twelve right shifts occur in succession moving the 1st entered digit from IR14 to IR2. At the same time, the contents of IR (from previous entry or answer) are shifted into AR as previously described. Since AR input to FAS is blocked via gate 96, AR information is destroyed.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

DT14 (Block Shifting; Set SCF; Reset OSF; Toggle ISF & ASF if signs are different; Reset ISF).

Shifting stops in DT14 because there are no shift pulses generated by the Clock Pulse Generator at this time. When the first digit entry is shifted into IR2 and DT14 is true, the SCF FF will be enabled to set via gate 15 by: JSCF.BT1. The OSF FF is enabled reset by the signal $\overline{BT1}$ being false when the CCCP1 clock pulse occurs. If the set-reset state of ISF and ASF are different, a toggle will be enabled via gates 37 and 38, or 37 and 39. To insure a positive entry, the ISF FF must be reset via gate 34.

DTO (Reset KEF FF)

After a digit key has been depressed, another selection cannot be made until the key is released. Releasing the digit key allows the logic signal Key to go false, and then at DTO time, KEY.DTO both being false will reset the KEF FF via clock pulse CFCP1. It should be realized that an undetermined number of data passes will occur before the digit key is released and therefore, the exact time of resetting the KEF FF cannot be predetermined. However, once KEF is reset, the next key depression will again set the OSF FF via gate 12 and a normal 2nd digit entry will occur.

2ND DIGIT ENTRY

With the SCF FF set, the 2nd digit entry is accomplished as follows:

DT14 (Set OSF and KEF FF's)

With the 2nd digit key depressed, the OSF FF is set via gate 12; and the KEF FF is then set via logic signal $\overline{OSF}$ being false.

SCF FF SET? YES.

At the end of the 1st entry data pass, the SCF FF was set to indicate the completion of the 1st digit entry, and to block IR data from shifting into AR.

KEY N=1? YES.

Indicates a digit key depression rather than a decimal point key. The logic flow will then follow the normal 2nd digit entry pattern.

KEY N=1? NO.

Indicates a decimal point key depression and since the SCF is set, the logic flow will follow the normal data shift flow sequence. (DTO through DT13, no logic action, normal data shift; DT14, block shifting, reset the OSF FF; DTO, reset the KEF FF.)

POF SET? NO.

With the POF FF not set, the decimal point key has not been depressed and therefore IR1 is equal to zero.

DTO (Shift IR Right One Place; IR1 to IR14)

The contents of IR are shifted right one data shift position by IRCP1. Data flow from IR1 to IR13 is via ASO. Input to IR13 is via gates 162 and 164. As IR1 is shifted into IR13, the number in IR2 (1st digit entry) is shifted into the IR1 position.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

DT1 (Shift IR Right One Place; IR1 to IR14; 2nd digit to IR13)

In DT1, the 1st digit entry now in IR1 is shifted into IR14 via gates 151 and 155. At the same time, the 2nd digit entry goes into IR13 via gate 165.

DT2-10 (Shift IR Right One Place Each Digit Time; IR1 to IR14)

During DT2 through DT10, nine shift pulses (IRCP1) occur moving the contents of IR to the right in a normal data shift flow. IR1 into IR14 is via gates 151 and 155.

DT11 (Block Shift of IR Data)

In DT11, the IR data flow is blocked for one shift by blocking the IRCP1 pulse. IRCP1 is blocked via gate 43 and 44. The result of this block is that the contents of IR are effectively shifted left one position.

DT12-13 (Shift IR Right One Place Each Digit Time, IR1 to IR14)

During DT12 through DT13, the normal data flow moves the contents of IR two places. The 1st digit will now be in IR3 and the 2nd digit will be in IR2. The decimal point information in IR1 is still "0".

DT14 (Block shifting; Reset the OSF FF)

In DT14, shifting is stopped by blocking the IRCP1 clock pulses. (DT14 signal being true at the inverter gate 44 causes the IRCP1 pulse to be "0" volts.) The OSF FF is reset by the $\overline{BT1}$ signal being false at the OSF FF reset enable input.

DTO (Reset the KEF FF)

The KEF FF will be reset only after the key is released. Therefore, KEY.$\overline{DTO}$ being false at the reset input will cause KEF to reset.

CC0 DEPRESS THE DECIMAL POINT KEY

The logic of the Decimal Point key is very similar to the digit key entry logic; therefore, only the differences will be discussed.

Depressing the Decimal Point key establishes a logic 1 (true) to the signals: Key, Key P, and Key (N+P). The POF FF is then set by: Key P at the inverter gate 64.

SCF SET? NO.

If the SCF FF is not set, it indicates that no previous digit entry has been made and therefore the decimal point key is the first entry. The data pass occuring during DTO through DT14 will be the same as described for a 1st digit entry.

SCF SET? YES. KEY N=1? NO.

When the SCF FF is set and the signal Key N is false, it indicates that the decimal point key has been depressed after the 1st digit entry. Therefore, a normal dynamic data shift will take place with no change in IR or AR.

CC0 DIGIT ENTRY FOLLOWING THE DECIMAL POINT KEY

After the Decimal Point key has been depressed and the POF FF is set, each digit entry thereafter will cause a "1" to be added to IR1 as follows:

KEY N=1? YES. POF FF SET? YES. SET THE $\pm$1F.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

With Key N=1 (true) and the POF FF set, the ±1F is set via gates 201, 203, and 204.

DTO (Shift IR Right One Place; Add "1" to IR1; IR1 to IR13)

The contents of IR are shifted right one data shift position by the clock pulse IRCP1 occuring. During this data shift a "1" is added to IR1 via ASO because the ±1F was set during DT14 time as indicated previously. Input to IR13 is via gates 162 and 164. The remaining Digit Entry Logic Flow for a 2nd digit entry (DTO through DT14) is the same as previously described.

ENTRY FUNCTION

1ST DIGIT SELECTION (LOGIC)

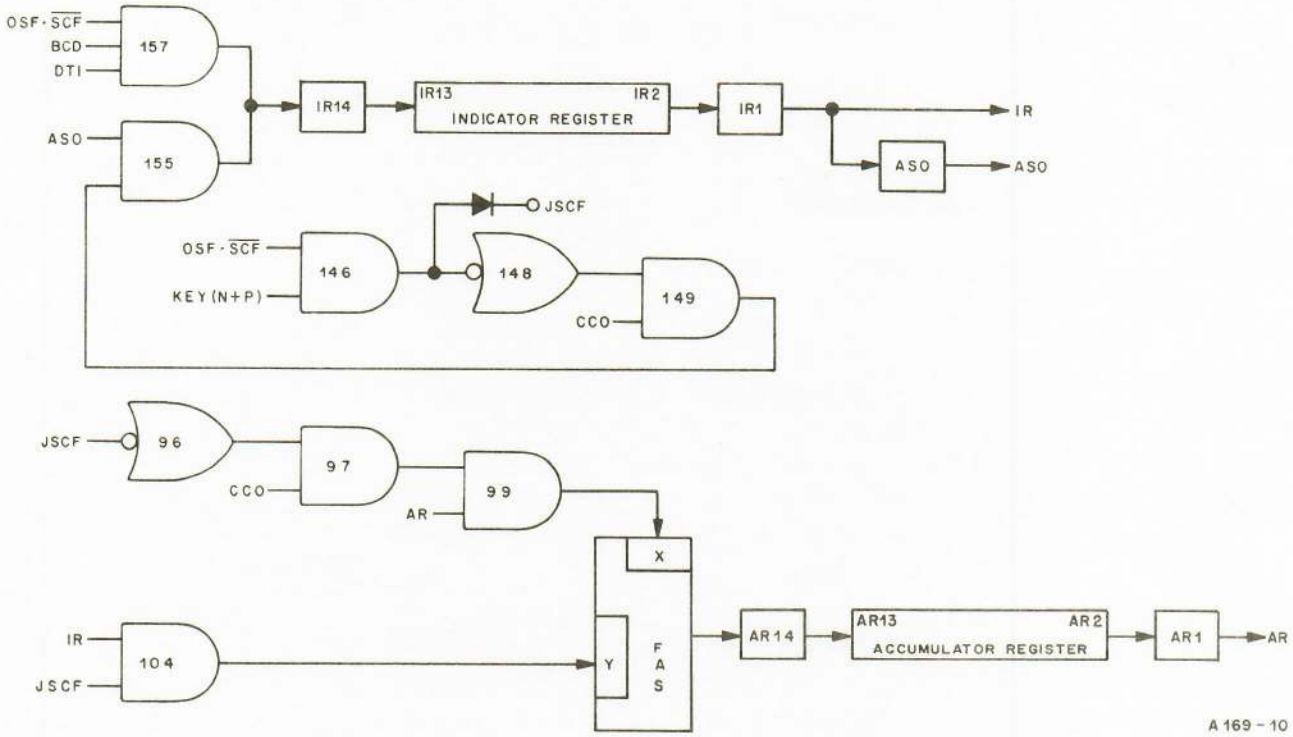


FIGURE 6-3

2ND DIGIT SELECTION (LOGIC)

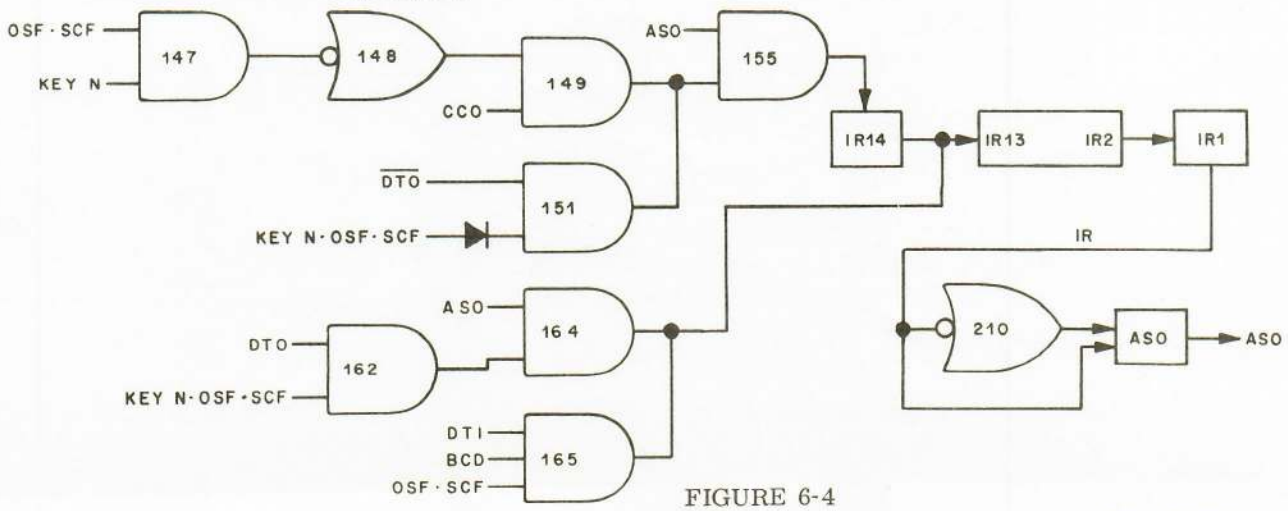


FIGURE 6-4

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

ENTRY FLOW CHART

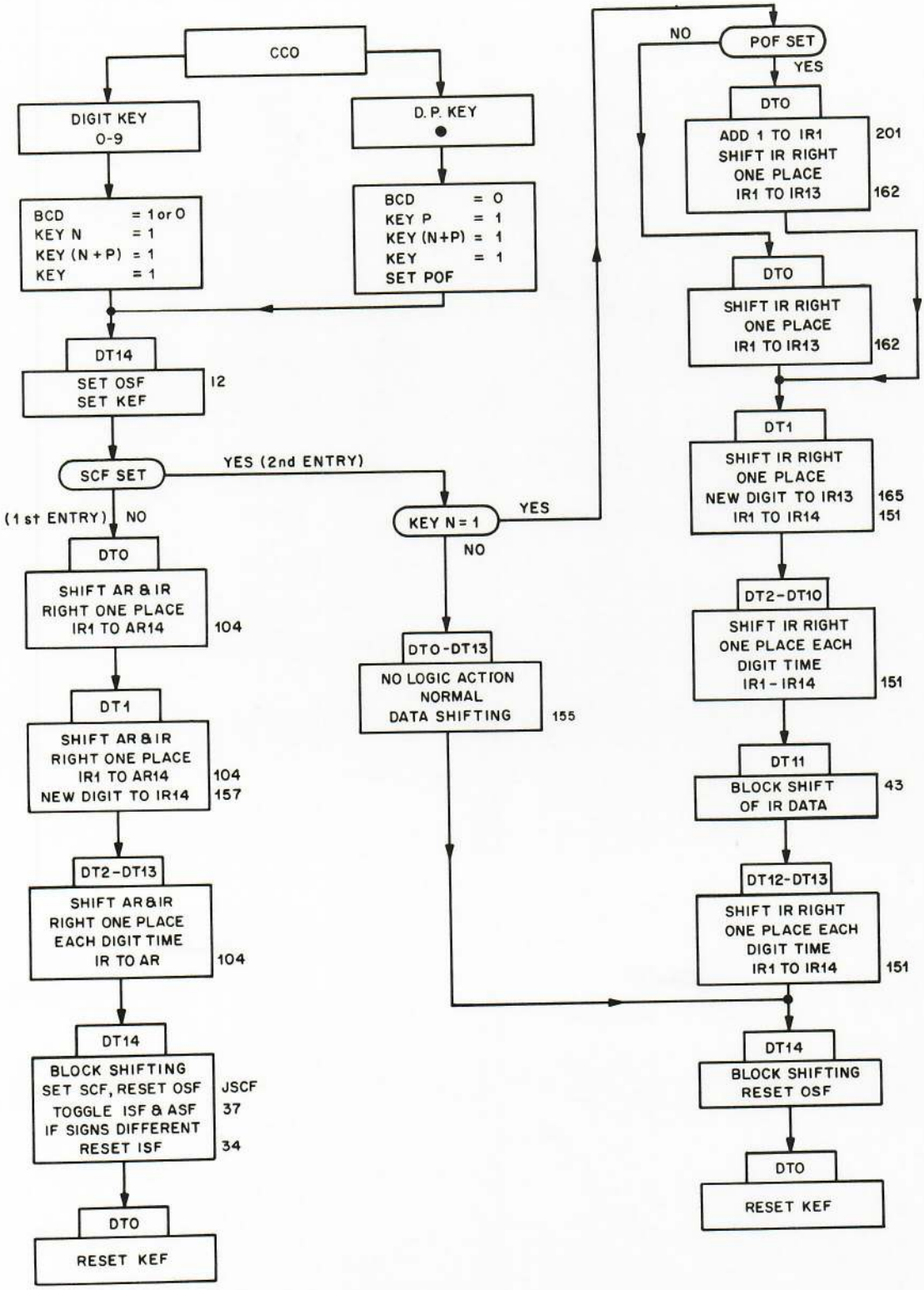


FIGURE 6-2

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

ADD AND SUBTRACT

To Add means to combine two numbers and arrive at a number which is the sum of the two. To Subtract means to combine two numbers and arrive at a number which is the difference between the two. Before an Add or Subtract function can take place, the two numbers must be decimally aligned in order to arrive at the correct answer. Therefore, when the Add or Subtract key is depressed, a comparison of decimal point information in IR1 and AR1 is made to see which is the larger. This will then tell the machine logic which register (IR or AR) must be shifted left to decimally align the contents with the other register. For example, if IR1 is greater than AR1, the contents of AR must be shifted left. Conversely, if AR1 is greater than IR1, the contents of IR must be shifted left.

When the two registers are decimally aligned, an Add or Subtract function occurs in the Full Adder and Subtracter (FAS) and the answer is accumulated in the AR register. During the last step in the operation, the contents of AR and IR are exchanged and the answer now in IR is displayed in the Number Indicating Tubes.

SAMPLE PROBLEM

For the purpose of illustration, the various steps involved in doing an Add or Subtract problem are shown in the work-sheet examples, Figure 6-5 and 6-6. In the Add problem, the two numbers being added are: 45.6+1.23=46.83. In the Subtract problem, the two numbers are: 45.6-1.23=44.37.

ADD
FUNCTION

45.6 + 1.23 = 46.83

INDICATOR REGISTER														ACCUMULATOR REGISTER													CCC	REMARKS
14	13	12	11	10	9	8	7	6	5	4	3	2	1	14	13	12	11	10	9	8	7	6	5	4	3	2	1		
										1	2	3	2											4	5	6	1		0	DEPRESS ADD KEY
										1	2	3	2												4	5	6	15		1	AR1-IR1
										1	2	3	2										4	5	6	0	0		2	AR LEFT; AR1+1
										1	2	3	2										4	5	6	0	0		3	NLA (1 BIT TIME)
										1	2	3	2										4	5	6	0	0		4	NLA (1 BIT TIME)
										1	2	3	2										4	5	6	0	0		5	NLA (1 COLUMN TIME)
										1	2	3	2										4	6	8	3	2		6	AR+IR
										1	2	3	2										4	6	8	3	2		7	NLA (1 COLUMN TIME)
										1	2	3	2										4	6	8	3	2		8	NLA (1 BIT TIME)
									4	6	8	3	2											1	2	3	2		9	EXCHANGE IR & AR
									4	6	8	3	2											1	2	3	2		0	END (SUM IN IR)

ADD PROBLEM
FIGURE 6-5

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

SUBTRACT FUNCTION

45.6 - 1.23 = 44.37

INDICATOR REGISTER														ACCUMULATOR REGISTER												ISF	ASF	CCC	REMARKS
14	13	12	11	10	9	8	7	6	5	4	3	2	1	14	13	12	11	10	9	8	7	6	5	4	3	2	1				
										1	2	3	2											4	5	6	1	0	0	0	DEPRESS SUB KEY
										1	2	3	2											4	5	6	15	1	0	1	AR1 - IR1; TOGGLE ISF
										1	2	3	2										4	5	6	0	0	1	0	2	AR LEFT; AR1 + 1
										1	2	3	2										4	5	6	0	0	1	0	3	NLA (1 BIT TIME)
										1	2	3	2										4	5	6	0	0	1	0	4	NLA (1 BIT TIME)
										1	2	3	2										4	5	6	0	0	1	0	5	NLA (1 COLUMN TIME)
										1	2	3	2										4	4	3	7	2	1	0	6	AR - IR; AR1 + IR1
										1	2	3	2										4	4	3	7	2	1	0	7	NLA (1 COLUMN TIME)
										1	2	3	2										4	4	3	7	2	1	0	8	NLA (1 BIT TIME)
									4	4	3	7	2											1	2	3	2	0	1	9	EXCHANGE IR & AR; (TOGGLE ISF & ASF)
									4	4	3	7	2											1	2	3	2	0	1	0	END (DIFFERENCE IN IR)

SUBTRACT PROBLEM
FIGURE 6-6

ADD AND SUBTRACT FLOW CHART ANALYSIS

The Add and Subtract machine functions are explained in conjunction with the Conditional Control Phase Counter (CC) and with the Add and Subtract Flow Chart, Figure 6-7, as follows:

CC0 DEPRESS ADD or SUBTRACT KEY; RESET POF

Depressing the Add key enables the logic signals: Key.Key (A+S) at logic 1 (true). Depressing the Subtract key enables the above signals plus one other: Key S. The POF FF is reset via gate 65 by: Key (A+S).CC0.SCF. The Conditional Control Phase Counter (CC) steps to CC1 via gate 130.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

CC1 COMPARE DECIMAL INFORMATION; SUB IR1 FROM AR1; IF IR GREATER THAN AR1, SET CAF AND POF.

In CC1 a comparison of decimal point information between AR1 and IR1 is made to determine which register (AR or IR) needs to be shifted left for decimal point alignment prior to an Add or Subtract operation.

The operation of comparing decimal point information is accomplished by subtracting IR1 from AR1 via the Full Adder and Subtracter (FAS). IR1 is gated into FASy via gates 102, 104, and 111 at DTO time. AR1 is gated into FASx via gate 99. Depending upon the numbers in IR1 and AR1, subtracting in FAS will produce a positive number answer if AR1 is greater than IR1; and will produce a complement number (subtract from zero) if IR1 is greater than AR1. If the answer is positive (AR1 greater than IR1) the alignment will occur in CC4. If IR1 is greater than AR1, the CAF and POF FF's will be set via gates 88 and 63; and the decimal alignment will occur in CC2.

KEY S = 1? YES.

In a Subtract operation, Key S being at logic 1 will necessitate toggling the ISF FF via gates 29 and 30. The Conditional Control Phase Counter (CC) will then step to CC2 via gate 139.

KEY S = 1? NO.

The Conditional Control Phase Counter (CC) will step to CC2 via gate 139.

CC2 POF SET? NO. NLA 1 BIT TIME. CC STEPS TO CC3 VIA GATE 131.

CC2 POF SET? YES. DECIMAL ALIGN AR; ADD 1 to AR1; SHIFT AR LEFT.

Because the decimal information in IR1 was greater than AR1 as determined in CC1, the contents of the AR register (except AR1) must be decimally aligned by shifting left. On each left shift, AR1 is incremented by "1" via the Full Adder and Subtracter (FAS) until AR1 is equal to Ø. When AR1 is equal to Ø indicating that the alignment is complete; shifting stops and the Conditional Control Phase Counter steps to CC3.

Adding a "1" to AR1 is accomplished via the Full Adder and Subtracter as follows: As AR1 contents are fed into FASx via gate 99, a binary 1 (0001) is forced into FASy via gate 109. The addition is performed by FAS and the output is gated into AR13 via gates 179 and 181. The remaining data in AR is gated through FAS and into AR14 via gates 173 and 176. Then, at DT11 time, normal data shifting is blocked one time by blocking the ARCP1 clock pulse via gate 42. The effect of blocking a shift right one time displaces the contents of AR one place to the left which is in reality a left shift.

AR1=0? NO. Shift AR left 1 place. Add 1 to AR1.

AR1=0? YES. Step to CC3 via gate 131.

CC3 NLA (1 bit time). STEP TO CC4 VIA GATE 132.

CC4 AR1=0? YES. RESET POF.

When AR1=0, the POF FF is reset via gate 65 by: CC4.AR1=0.DT14. This same signal also enables gate 137 to cause CC to step to CC5.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

CC4 AR1=0? NO. DEC. ALIGN IR; ADD 1 TO IR1; SUB 1 FROM AR1; SHIFT IR LEFT.

If AR1 is not "0", IR must be decimally aligned. Each left shift will cause a "1" to be added to IR1 and a "1" to be subtracted from AR1. When AR1 equals "0", IR will be properly aligned and IR1 will indicate the correct decimal position. Adding a "1" to IR1 is accomplished via ASO at DTO time as follows: During the last half of DT14, the set input to the ±1F is enabled by gates 200, 203, and 204. Then, during DTO, a "1' is added to IR1. IR1 information from ASO is gated into IR13 via gates 160 and 164. Normal shifting of IR2 through IR13 into IR14 is via gates 151, 154, and 155. At DT11, IRCP1 is blocked by gate 43 for one shift causing IR to be shifted left one place. AR1 decimal point information is counted down by "1" at each DTO time by forcing a "1" into FASy via gates 108, 109, and 111. The CC4 "OR" input below gate 85 makes FAS act as a subtracter.

AR1=0? NO.

Repeat the sequence of IR left shift; IR1-1 and AR1-1.

AR1=0? YES.

Reset POF via gate 65. Step to CC5 via gate 137.

CC5 ASF SET? NO.

The contents of AR being positive, no logic action occurs and the Conditional Control Phase Counter steps to CC6 via gate 139.

CC5 ASF SET? YES. COMPLEMENT AR DATA

If AR contains a negative number, the ASF FF will be set and AR must be complemented (subtracted from zero) before the Add or Subtract function occurs in CC6. FAS will act as a subtracter via gate 83. AR input to FASy is via gates 106 and 111. AR input to FASx is blocked by gate 99, therefore, the negative number in AR is subtracted from zero and a complement number is now in AR. The Conditional Control Phase Counter (CC) will then step to CC6 via gate 139.

CC6 ADD OR SUBTRACT

In CC6, an Add or Subtract function occurs depending upon whether or not the ISF FF is set. The RDF FF being set will indicate that only one Add or Subtract operation will occur.

ISF SET? NO. SET RDF; ADD IR TO AR.

The RDF FF is set via gate 50. The arithmetic function of IR added to AR is accomplished in the Full Adder and Subtracter (FAS), with the answer (sum) gated into AR14. IR input to FASy is via gates 103, 104, and 111. AR input to FASx is via gate 99. AR14 input from FAS is via gates 173 and 174. NOTE: During the Add operation, the decimal point information in IR1 is added to AR1 so AR1 will contain the correct decimal point information.

ISF SET? YES. SET RDF; SUBTRACT IR FROM AR.

The RDF FF is set via gate 50. With ISF set, FAS will act as a subtracter via gate 85. IR input to FASy is via gate 103. AR input to FASx is via gate 99. AR14 input to FAS is via gates 173 and 174. Decimal point information in IR1 is added to AR1 so AR1 will contain the correct decimal point information.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

CAF FF SET? NO. RESET SCF AND RDF.

With the CAF FF not set, it indicates that the sign of the number in AR is correct. Therefore, SCF and RDF are reset via gates 17 and 54. The Conditional Control Phase Counter (CC) will then step to CC7 via gate 133.

CAF FF SET? YES. COMPARE SIGNS.

With the CAF FF set, it indicates that the sign of the number in AR is incorrect and, therefore, must be corrected by the setting or resetting of the ASF FF.

ISF SET? NO. RESET ASF AND SCF.

Under the condition of ISF FF being reset, the ASF FF must be reset via gate 70 to indicate a positive number in AR. At the same time, the SCF FF is reset via gate 17, and the CC steps to CC7 via gate 134. NOTE: In this case, the RDF FF will be reset in CC0 because the signal CC6.$\overline{SCF}$ never occurs.

ISF SET? YES. SET ASF, RESET SCF AND RDF.

Under the condition of ISF FF being set, the ASF FF is set via gate 67 to indicate a negative number in AR. At the same time, the SCF FF is reset via gate 17. One bit time later, the logic signal CC6.$\overline{SCF}$ resets the RDF FF via gate 54 and steps the CC to CC7 via gate 133.

CC7 ASF SET? NO.

If the ASF FF is not set, the number in AR does not need to be complemented and after 1 column time, CC steps to CC8 via gate 139.

ASF SET? YES. COMPLEMENT AR DATA.

If the ASF FF is set, the number in AR must be complemented (subtracted from zero) to obtain the correct answer in AR. FAS will act as a subtracter via gate 83. AR input to FASy is via gates 105, 106, and 111. AR input to FASx is blocked by gate 99, therefore, the number in AR is subtracted from zero. Then at DT14 time, the CC steps to CC8 via gate 139.

CC8 NLA 1 COLUMN TIME

No logic action; used only in Mult and Divide. Step to CC9 via gate 135.

CC9 EXCHANGE CONTENTS OF IR AND AR; TOGGLE ASF & ISF IF SIGNS ARE DIFFERENT.

In this step, the answer developed in AR is always exchanged with the contents of IR. IR is gated into AR via FASy (gate 104). NOTE: Y2 is a logic signal developed by CC9 at gate 156 as a control signal for exchanging AR with IR. AR is blocked from FASx at gate 99, therefore, IR is added to zero without changing value, and the output from FAS goes into AR14 via gates 173 and 174. AR is gated into IR via gate 156 and IR is blocked at gate 155.

The toggle of ISF and ASF will occur if the signs are different via the enabling gate 37. The answer to be displayed is now in IR and the second entry is in AR. The CC steps to CC0 via gate 139.

CC0 RESET RDF FF

As the CC steps to CC0, the RDF FF is reset via gate 54. The Add or Subtract function is completed and the Conditional Control Phase Counter is ready to start another function.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

ADD OR SUBTRACT (Basic Logic for CC1)

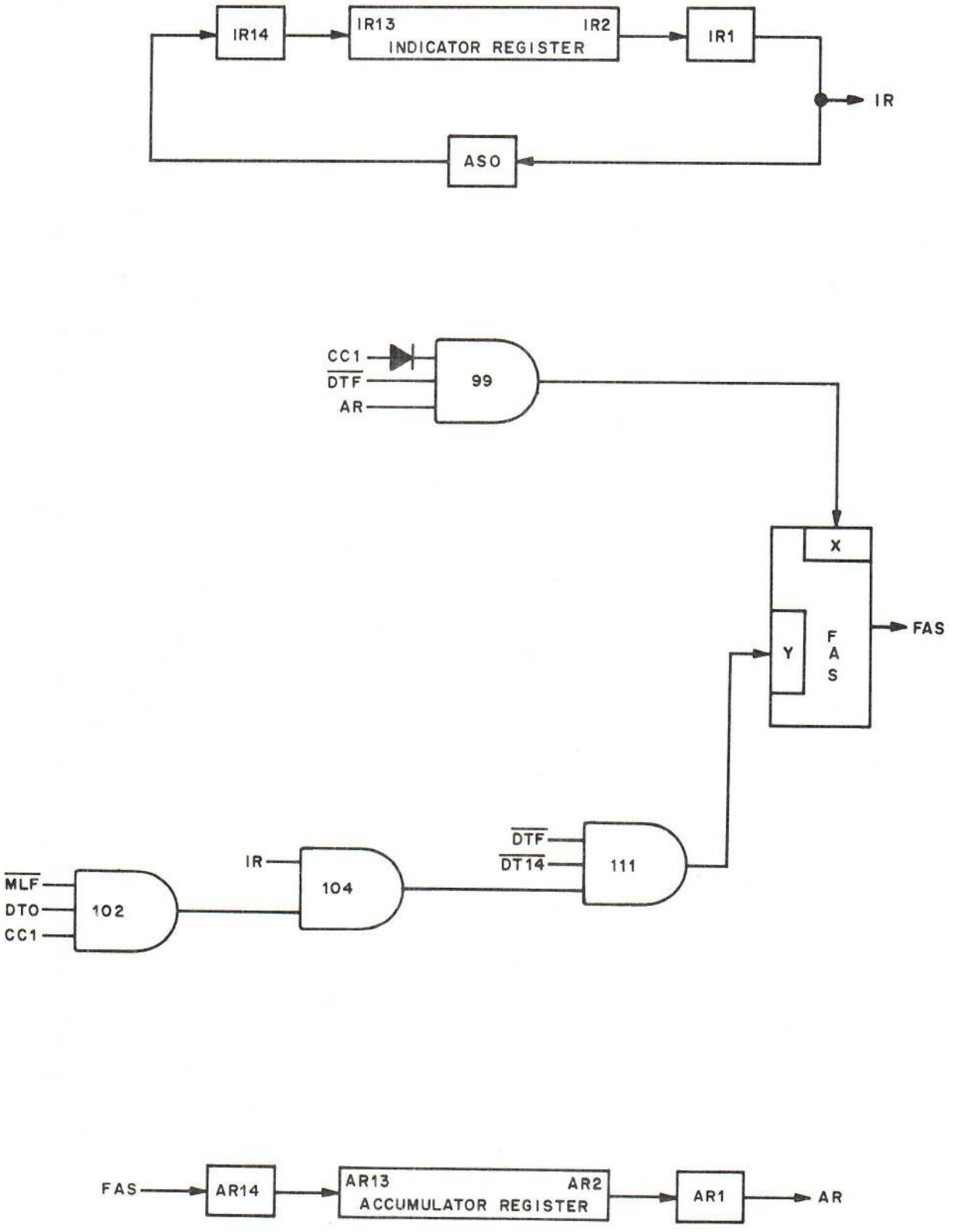


FIGURE 6-8

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

ADD OR SUBTRACT (Basic Logic for CC2)

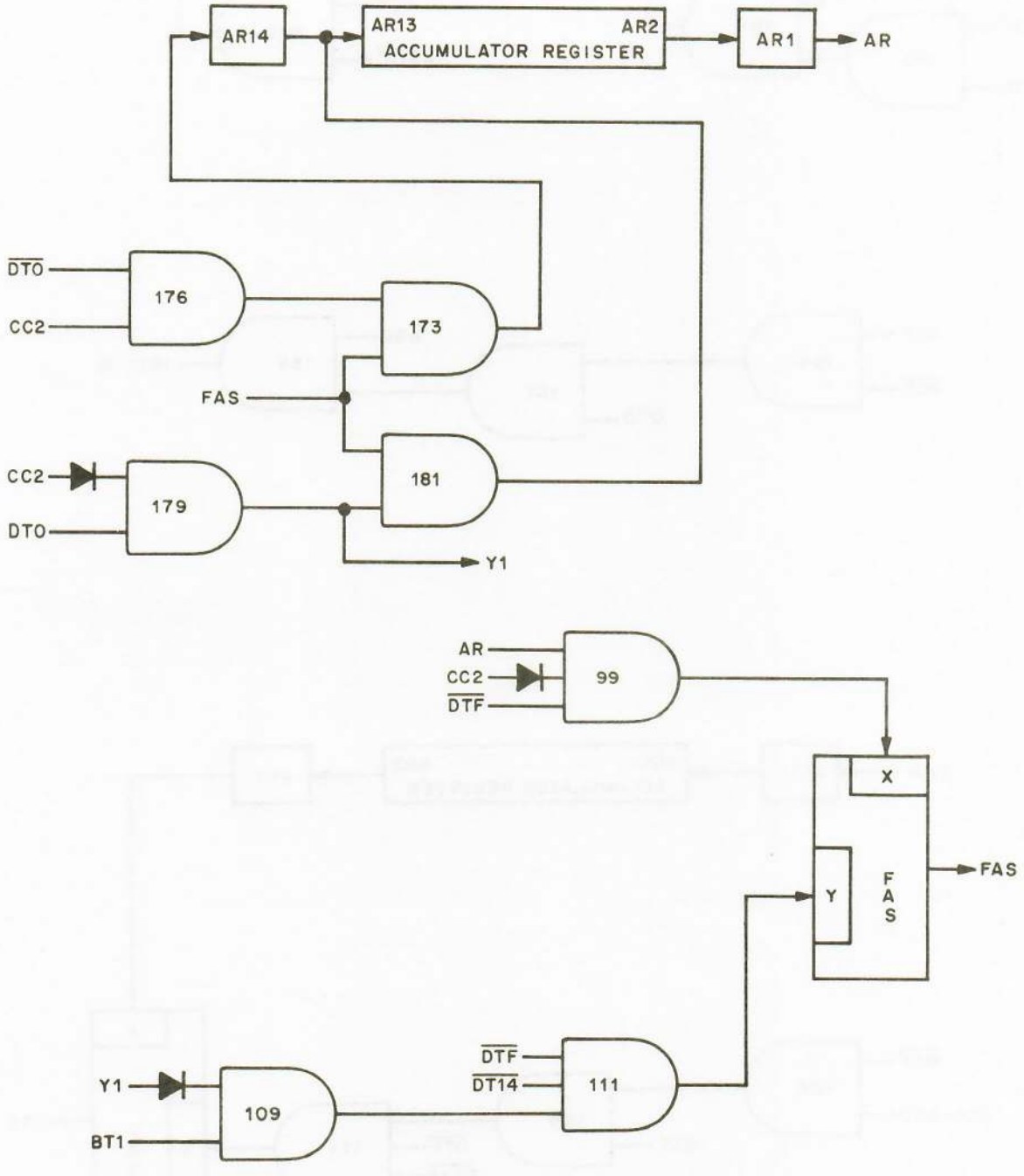


FIGURE 6-9

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

ADD OR SUBTRACT (Basic Logic for CC4)

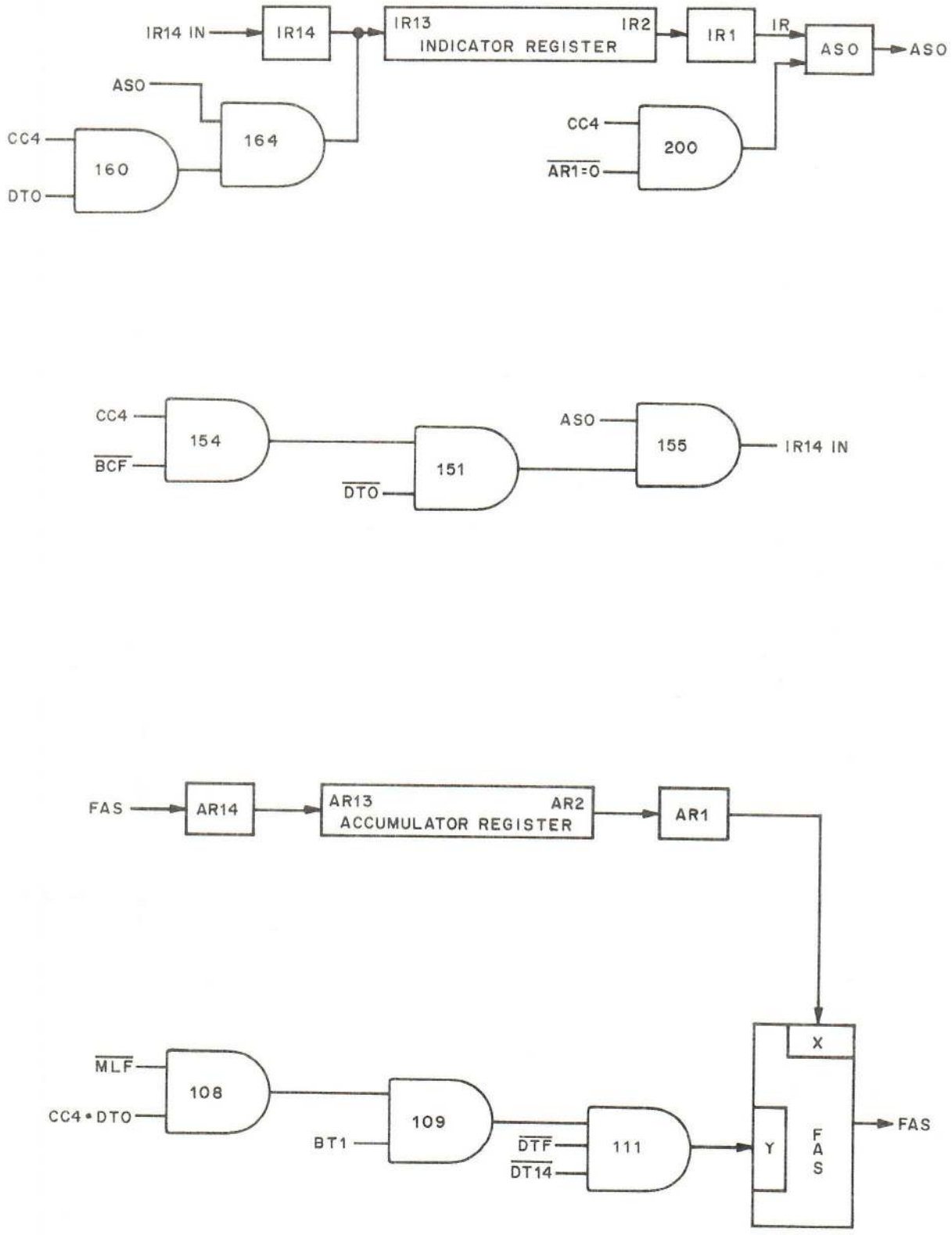


FIGURE 6-10

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

ADD OR SUBTRACT (Basic Logic for CC5/CC7; ASF Set)



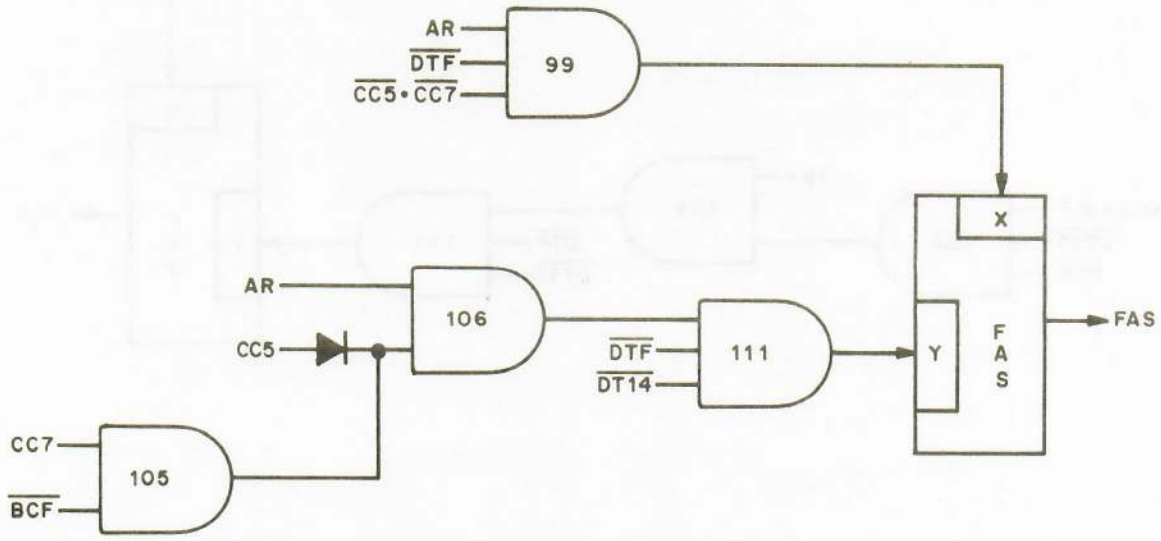
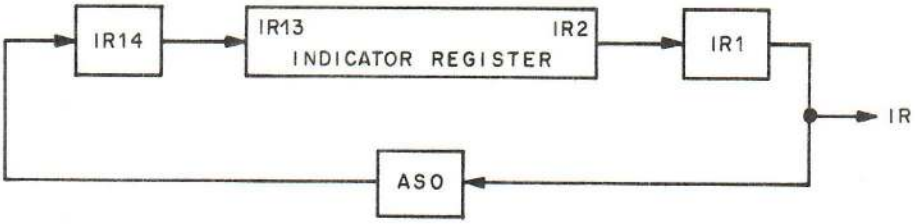
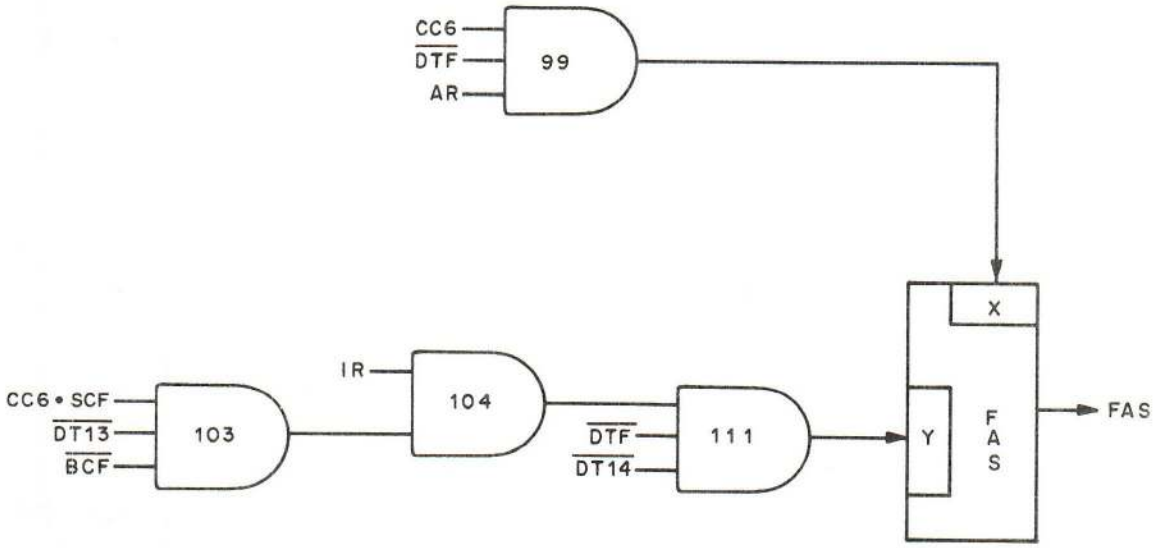


FIGURE 6-11

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

ADD OR SUBTRACT (Basic Logic for CC6.SCF)





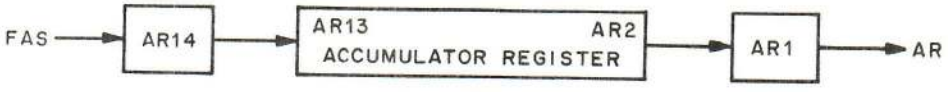


FIGURE 6-12

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

ADD OR SUBTRACT (Basic Logic for CC9)

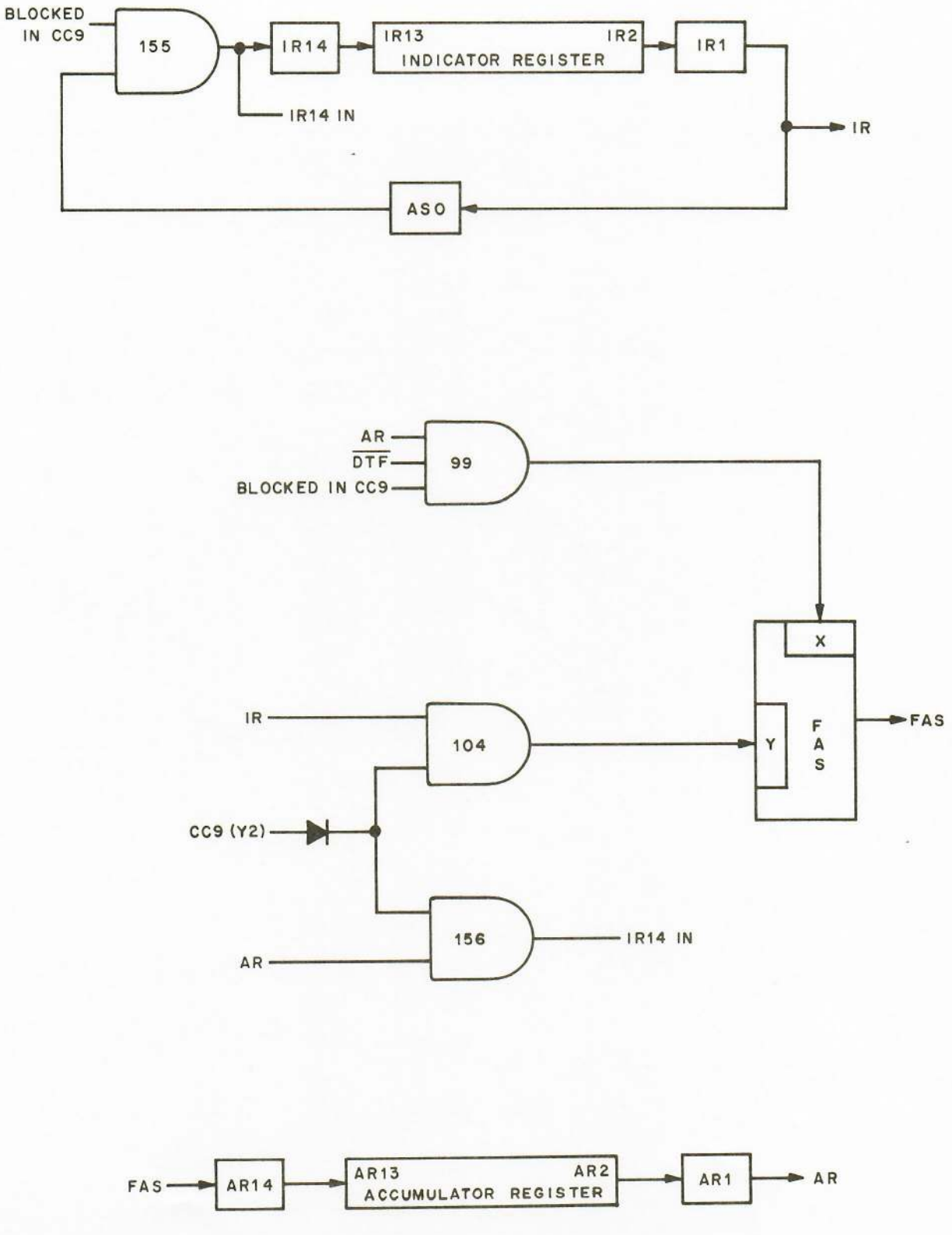


FIGURE 6-13

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

ADD OR SUBTRACT FLOW CHART

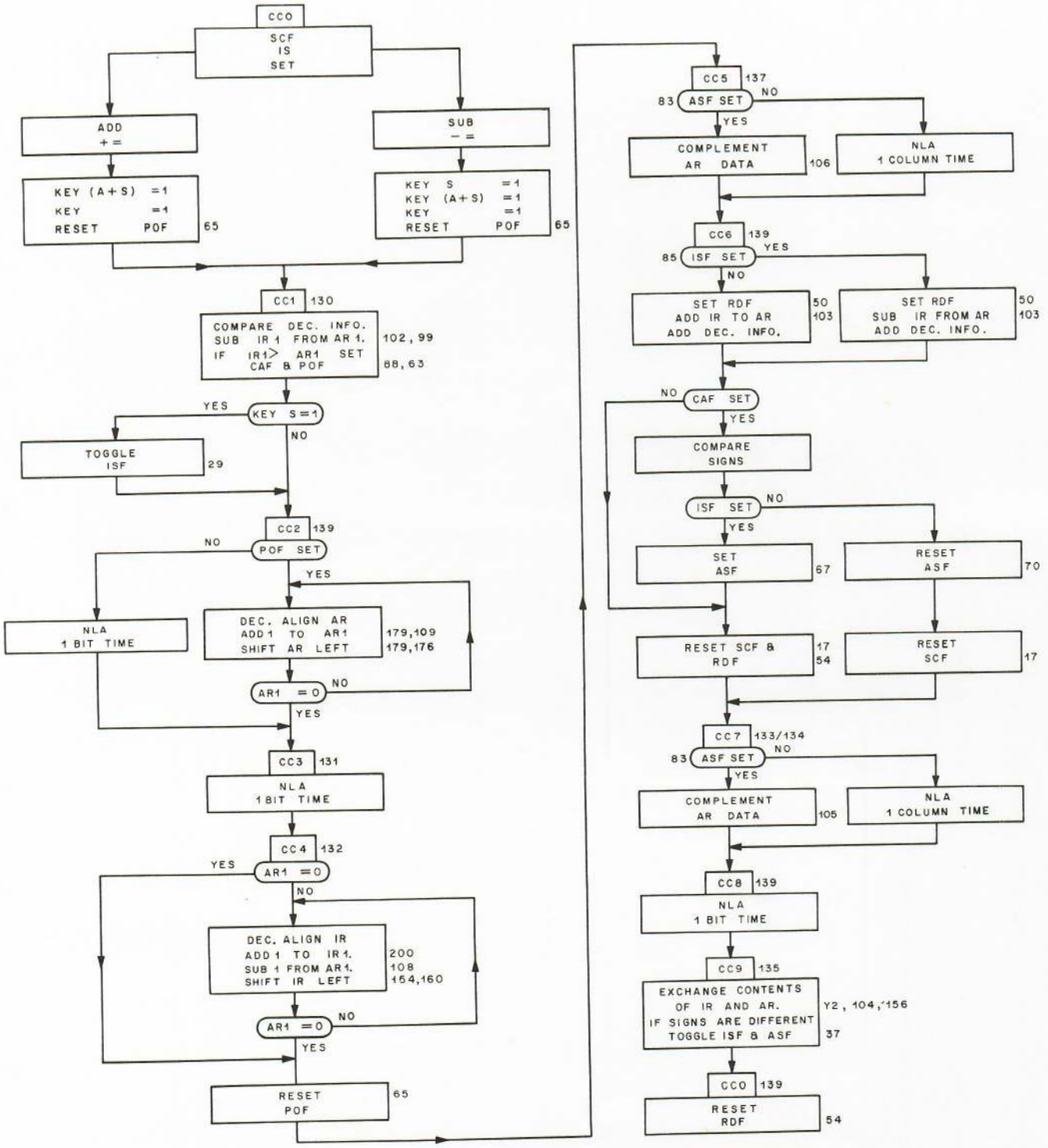


FIGURE 6-7

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

MULTIPLY

Multiplication can be described as a series of additions or subtractions and shifts in which the multiplicand is added or subtracted a number of times as determined by the multiplier. For example, to multiply 135 x 25, the multiplicand (135) is added five times for an intermediate answer of 675. The intermediate answer is then shifted right one position as 0675 and the multiplicand is then added to the intermediate answer two more times for a final answer (product) of 3375.

In the 1113, a multiply operation is performed in a similar manner by adding the number in IR (multiplicand) a number of times determined by the least significant digit of the number in AR (multiplier). The intermediate answer is developed in AR and is finally exchanged with IR at the end of the operation.

SAMPLE PROBLEM

For purposes of illustration, the various steps involved in doing a multiply operation are shown in the work-sheet example, Figure 6-14. The numbers to be multiplied are: 1.44 x 1.2 = 1.728.

MULT.
FUNCTION

1.44 × 1.2 = 1.728

INDICATOR REGISTER														ACCUMULATOR REGISTER												SCF	RDF	CCC	REMARKS
14	13	12	11	10	9	8	7	6	5	4	3	2	1	14	13	12	11	10	9	8	7	6	5	4	3	2	1				
										1	4	4	2															0	0	0	DEPRESS KEY (×)
											1	2	1											1	4	4	2	1	0	0	ENTER 1.2; KEY (+ =)
											1	2	1											1	4	4	2	1	0	1	NLA (1 C TIME)
											1	2	1											1	4	4	2	1	0	2	NLA (1 BIT TIME)
										1	4	4	2												1	2	3	1	0	3	EXCHANGE IR & AR; IR_1+AR_1
	1								1	4	4	(15)	3												1	2	3	1	0	4	ENTER BC; IR_1+1; $IR_{13}+1$; SHIFT LEFT IR_2-IR_{12}.
	2							1	4	4	(15)		4												1	2	3	1	0	4	
	3						1	4	4	(15)			5												1	2	3	1	0	4	SHIFT LEFT IR_2-IR_{12}
	4					1	4	4	(15)				6												1	2	3	1	0	4	AND ADD 1 TO IR_1 &
	5				1	4	4	(15)					7												1	2	3	1	0	4	IR_{13} UNTIL (MSD)
	6			1	4	4	(15)						8												1	2	3	1	0	4	ARRIVES IN IR_{12}.
	7		1	4	4	(15)							9												1	2	3	1	0	4	
	8	1	4	4	(15)								10												1	2	3	1	0	4	
9	1	4	4	(15)									11												1	2	3	1	0	4	IR_1+1; $IR_{14}+1$; SHIFT LEFT IR_2-IR_{13}.
9	1	4	4	(15)									11												1	2	3	1	0	5	NLA (1 C TIME)
9	1	4	4	(15)									11		1	4	4								1	1	3	1	0	6	AR2-1; AR+IR
9	1	4	4	(15)									11		2	8	8								1	0	3	1	0	6	AR2-1; AR+IR

FIGURE 6-14 (Continued on next page)

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

INDICATOR REGISTER														ACCUMULATOR REGISTER												SCF	RDF	CCC	REMARKS
14	13	12	11	10	9	8	7	6	5	4	3	2	1	14	13	12	11	10	9	8	7	6	5	4	3	2	1				
2	1	4	4	(15)									11								1	7	2	8		0	3	1	1	6	(1 C TIME)
1	1	4	4	(15)									11									1	7	2	8	0	3	0	0	6	AR RIGHT; IR14-1
																												1	1	6	(1C TIME)
0	1	4	4	(15)									11										1	7	2	8	3	0	0	6	AR RIGHT; IR14-1
	1	4	4	(15)									11										1	7	2	8	3	1	0	7	NLA (1 C TIME)
		1	4	4	(15)								10										1	7	2	8	3	1	0	8	IR RIGHT; IR1-1
			1	4	4	(15)							9										1	7	2	8	3	1	0	8	" "
				1	4	4	(15)						8										1	7	2	8	3	1	0	8	" "
					1	4	4	(15)					7										1	7	2	8	3	1	0	8	" "
						1	4	4	(15)				6										1	7	2	8	3	1	0	8	" "
							1	4	4	(15)			5										1	7	2	8	3	1	0	8	" "
								1	4	4	(15)		4										1	7	2	8	3	1	0	8	" "
									1	4	4	(15)	3										1	7	2	8	3	1	0	8	IR RIGHT; IR1-1
										1	4	4	2										1	7	2	8	3	1	1	8	IR RIGHT; IR1-1; DESTROY BOUNDARY CODE.
									1	7	2	8	3											1	4	4	2	0	1	9	EXCHANGE IR & AR
									1	7	2	8	3											1	4	4	2	0	0	0	END (PRODUCT IN IR)
																												1	1	6	(1C TIME)
8	1	4	4	(15)									11			2	8	8								1	3	0	0	6	AR RIGHT; IR14-1
8	1	4	4	(15)									11		1	7	2	8								0	3	1	0	6	AR2-1; AR+IR
																												1	1	6	(1 C TIME)
7	1	4	4	(15)									11			1	7	2	8							0	3	0	0	6	AR RIGHT; IR14-1
																												1	1	6	(1 C TIME)
6	1	4	4	(15)									11				1	7	2	8						0	3	0	0	6	AR RIGHT; IR14-1
																												1	1	6	(1 C TIME)
5	1	4	4	(15)									11					1	7	2	8					0	3	0	0	6	AR RIGHT; IR14-1
																												1	1	6	(1 C TIME)
4	1	4	4	(15)									11						1	7	2	8				0	3	0	0	6	AR RIGHT; IR14-1
																												1	1	6	(1 C TIME)
3	1	4	4	(15)									11							1	7	2	8			0	3	0	0	6	AR RIGHT; IR14-1
																												1	1	6	(1 C TIME)
2	1	4	4	(15)									11								1	7	2	8		0	3	0	0	6	AR RIGHT; IR14-1

FIGURE 6-14

MULT FLOW CHART ANALYSIS

The Multiply functions are explained in conjunction with the Mult Flow Chart, Figure 6-15 as follows:

Depressing the (X) key establishes the logic signal Key M at logic 1 which sets the MLF FF via gate 24. As the MLF FF is being set, the Key M signal on the reset input of the SCF FF causes SCF to reset. Then with CCO.$\overline{\text{SCF}}$ both true, the second number (multiplier) is entered in IR at which time the first number in IR (multiplicand) is shifted into AR. During the second number entry the SCF FF is set via the logic signal JSCF as previously described under "Entry". NOTE: In a Square Mult operation, gate 95 will be enabled which causes the SCF FF to set and also enables the gating of IR into AR via FAS, thus avoiding a 2nd digit entry. The number in IR is gated back into IR14 in via gates 149 and 155 the same as in a regular entry.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

CC0 KEY +=; or KEY -=; RESET THE POF FF

Depressing the Plus Equal (+=) key causes the logic signal Key (A+S) and Key to be true (logic 1) and the POF FF is reset via gate 65. Depressing the Minus Equal (-=) key causes the logic signal Key (A+S), Key, and Key S to be true. The POF FF is also reset via gate 65. The Conditional Control Phase Counter (CC) then steps to CC1 via gate 130.

CC1 NLA 1 COLUMN TIME

If Key (+=) Plus Equal is depressed, no logic action will take place in CC1 and after one column time, the CC will step to CC2 via gate 139

TOGGLE THE ISF FF

If Key (-=) Minus Equal is depressed, the ISF FF will be toggled at the end of one column time via gates 29 and 30. The CC will then step to CC2 via gate 139.

CC2 NO LOGIC ACTION. STEP TO CC3 VIA GATE 131.

CC3 EXCHANGE CONTENTS OF IR AND AR; ADD IR1 TO AR1.

In this step, the contents of AR and IR are exchanged so that the multiplier will be in AR and the multiplicand will be in IR. Also, the sum of AR1 and IR1 is accumulated in AR1. The operation is accomplished as follows:

The decimal point information in IR1 is added to the decimal point information in AR1 via FAS. AR1 is gated into FASx via gates 98 and 99; IR1 is gated into FASy via gate 104 (Y2.IR). The output from FAS (AR1+IR1) is gated into AR14 via gates 173 and 174. This decimal point information going to AR will indicate the decimal point position in the final answer.

The contents of AR and IR are exchanged by gating IR2 through IR14 into FASy via gate 104. AR2 through AR14 are prevented from entering FASx because gate 98 is now blocked by DTO being false. The contents of IR going through FAS are therefore unchanged and the output of FAS is gated into AR14 via gate 173. The contents of AR1 through AR14 are gated into IR14 in via gate 156. At DT14 time, shifting is completed and AR and IR are now exchanged. The CC steps to CC4 via gate 139.

CC4 ENTER BOUNDARY CODE; ADD 1 TO IR1 & IR13; SHIFT LEFT IR2-IR12; ADD 1 TO IR1 & IR14; RESET BCF; SHIFT LEFT IR2-IR13; RESET ASF.

Preparatory to doing the additions or subtractions required in the multiplication operation, a series of left shifts are necessary to move the MSD of the multiplicand into IR13. A boundary code consisting of a binary 15 (1111) is also entered to the right of the LSD of the multiplicand to serve as a marker. Each left shift is counted in IR1 and IR13, with decimal point information in IR1 and shift information in IR13. As the last shift occurs to move the MSD into IR13, the shift information in IR13 is routed through ASO and into IR14. The Conditional Control Phase Counter (CC) will then step to CC5.

ENTER BOUNDARY CODE; SET THE BCF FF.

On the first data pass in CC4, the Boundary Code 15 (1111) is gated into IR14 in via gates 158 and 160 during the first half of DTO. Then, when the clock pulse CFCP1 occurs in the last half of DTO, the BCF FF is set via the inverter gate 57. With the BCF FF set, gate 158 will be blocked which prevents entering more than one boundary code.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

ADD "1" TO IR1 AND IR13; SHIFT LEFT IR2 THROUGH IR12.

The ±1F was enabled to be set via gates 200 and 204 during the last half of DT14 time; and is set in DTO time as the Boundary Code is being entered. Then, as IR is gated into ASO, a "1" is added to IR1 as decimal point information and the ASO output is gated into IR13 in via gates 160 and 164.

The contents of IR2 through IR12 are gated into IR14 in via gates 151, 154, and 155. At DT11 time, the IRCP1 clock pulse is blocked via gates 43 and 44 which causes the contents of IR (except decimal and shift information) to be shifted left.

At DT12 time, normal data flow resumes and during the last half of DT12 when BT8 occurs, the ±1F is enabled set via gates 197, 199, and 204. Then at DT13 time, the ±1F is set and as IR is gated into ASO, a "1" is added to IR13 as shift information. The output from ASO is gated into IR13 in via gates 161 and 164.

IR12=0? YES. Repeat left shift operation as described above.

IR12=0? NO. ADD 1 TO IR1 & IR14; RESET BCF; SHIFT LEFT IR2-IR13.

When the MSD of the multiplicand reaches IR12, one more left shift is required to move it into IR13. During this left shift data pass, the decimal point information in IR1 is incremented by "1" via ASO and the output is gated into IR13 as previously explained. During DT1 through DT10, normal data flow of IR2 through IR11 occurs. In DT11, the IRCP1 clock pulse is blocked which causes a left shift of IR.

In DT12 time, normal data flow shifts the MSD of the multiplicand into IR14 where it is sensed as $\overline{IR14=0}$ at logic 1 (true) for the first time. This signal being true at gate 60 enables the BCF FF to reset. Then, in DT13 time, gate 161 will go false because the BCF FF is now reset and IR13 shift data is incremented by "1" via ASO, but since gate 161 is now blocked, IR13 shift data is gated into IR14 in via gates 151, 152, 153, 154, and 155. At the same time shift data is gated into IR14, the MSD of the multiplicand is shifting out of IR14 and into IR13. In DT14 time, shifting stops and the CC steps to CC5 via gate 140. (NOTE: The output of gate 140 also resets the ASF FF via gate 69 to insure that the multiplier number in AR is not complemented in CC5.)

CC5 NO LOGIC ACTION (1 Column Time)

AR and IR circulate dynamically for one column time. Then, the Conditional Control Phase Counter (CC) steps to CC6 via gate 139.

CC6 This step has two phases--an Add or Subtract phase CC6.SCF and a shift right phase CC6.$\overline{SCF}$. In CC6.SCF, a series of additions and subtractions occur in FAS in which the contents of IR (multiplicand) are added or subtracted (depending on which key is used) a number of times determined by the number in AR2 (LSD of the multiplier).

In CC6.$\overline{SCF}$, the contents of AR (multiplier) are shifted right one place in order to bring the next multiplier digit into AR2. Also, a "1" is subtracted from IR14 (shift information) via ASO to indicate that a right shift has occurred in AR.

CC6.SCF (Add or Subtract Phase)

In the Add or Subtract phase, the BCF FF is set via gate 55 to block the FASy gate 103 until the boundary code has shifted out of IR1 and into IR14 via ASO. Then the LSD of the multiplier in AR2 is sensed in AR1 and if it is not "0", the arith-

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

metic function of Add or Subtract via FAS will be enabled and a series of additions or subtractions will occur as follows:

Subtract "1" from AR2. As the 1st digit of the multiplier (LSD) is gated into FASx via gate 99, a "1" is forced into FASy via gates 107, 109, and 111. This "1" is subtracted from AR2 because gate 84 enables FAS to act as a subtracter.

Reset BCF. When the boundary code (1111) is detected in IR14 the logic signal IR14 equals 15 becomes true and the BCF FF will reset via gate 59. The BCF FF being reset will enable the FASy gate 103 so the contents of IR can enter FASy via gates 104 and 111.

Add IR to AR. The multiplicand in IR is gated into FASy via gate 104. The contents of AR corresponding to the same digit positions as IR, go into FASx via gate 99. FASy (multiplicand) is then added to FASx (AR contents) in the normal manner and the sum (intermediate product) is accumulated in AR via gates 173 and 174.

Set BCF. At the end of each data pass at DT14 time, the BCF FF is set via gate 55 in readiness for the start of another Add or Subtract phase as determined by whether or not AR2 is equal to "0".

NOTE: If the ISF FF is set, a series of subtractions will occur, but the operation is the same as described for the Add phase except that the first subtraction will set the ASF FF via gate 67. The purpose of setting the ASF FF will be explained later under "Right Shift".

CC6.$\overline{\text{SCF}}$ (Right Shift Phase)

When the LSD of the multiplier in AR2 is "0", a right shift of the AR contents must occur to bring the next multiplier digit into AR2 position. With AR2 equal to "0", the logic signal AR1 equal "0" is sensed and the RDF FF is set during the 2nd half of DTO via gates 50 and 51. RDF remains set for one complete data pass (column time) and during this time, it blocks the arithmetic function of FAS; and enables the reset of the SCF FF via gate 17.

Reset SCF, RDF, and BCF; Set POF. At DT14 time, the SCF FF is reset via gate 17. Then the RDF and BCF FF's are reset via gates 54 and 58; and the POF FF is set via the logic signal: CC6.$\overline{\text{SCF}}$.$\overline{\text{DIF}}$ inverted by gate 64.

The dynamic data flow of AR contents (AR2 through AR14) is out AR1 and into AR13 via gates 177, 180, and 181. The decimal point information in AR1 enters AR14 via gates 173, 175, and 177. The right shift is accomplished by supplying the normal 14 shift pulses (ARCP1), but since the contents of AR2 through AR14 are being gated into AR13, 14 shift pulses will be one too many; therefore, the "0" in the LSD position will be eliminated.

Subtract "1" from IR14. As each right shift occurs, a "1" is subtracted from the shift information in IR14 via ASO gates 198 and 212.

ASF Set? Yes. Enter "9" in AR14. If the first intermediate answer in AR is a complement number, AR14 will contain a "9" as a result of the subtraction from "0". Then during CC6.$\overline{\text{SCF}}$ when the right shift occurs to bring the next multiplier number into AR2, the "9" in AR14 would shift to the right and leave a "0" in AR14. If this should happen, a wrong answer would occur when the intermediate answer is recomplemented in CC7. To prevent this situation, gate 172 is enabled at the end of the right shift and a "9" (1001) is forced into AR14.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

AFS Set?No. IR14=0? No. Set SCF and BCF FF's via gates 15 and 55 and repeat the CC6 loop.

IR14=0? Yes. When IR14 is equal to "0", it indicates that the product in AR has shifted right the same number of places defined by IR14 and is now decimally aligned. The CC will then step to CC7 via gate 133.

CC7 ASF SET? YES. COMPLEMENT AR DATA

With the ASF FF set, the intermediate product in AR will be a complement number because of the series of subtractions in CC6; therefore, it must be re-complemented by subtracting AR from zero. FAS is enabled to be a subtracter via gate 83. AR is gated into FASy via gates 105 and 106. FASx is blocked via gate 99; therefore, FASy will be subtracted from FASx which is zero. The CC will then step to CC8 via gate 139.

ASF SET? NO. NO LOGIC ACTION (1 Column Time).

After one column time, the CC steps to CC8 via gate 139.

CC8 SUBTRACT "1" FROM IR1; SHIFT IR RIGHT

In CC8 a series of right shifts occur moving the contents of IR to the right until the multiplicand has reached its original entry position. As each right shift occurs, a "1" is subtracted from IR1. During the last shift as the LSD of the multiplicand arrives in IR2, the boundary code is destroyed and the RDF FF is set. Then at the end of the data pass, the Conditional Phase Counter steps to CC9.

Subtract "1" from IR1. In DTO time of each data pass, a "1" is subtracted from IR1 because gate 212 enables ASO to act as a subtracter and the ±1F is set via gate 203. ASO output is gated into IR14 in via gates 150 and 155.

Shift IR Right. In DT1 of each data pass, gate 163 is inhibited blocking the input to IR13 in and therefore, IR2 data during this shift is destroyed because it has no place to go. During DT2 through DT13, the contents of IR3 through IR13 are shifted into IR13 in via gates 163 and 164. At the end of each column time (DT14), the contents IR have been shifted right one place.

IR2=15? NO. Repeat shifting IR right until the Boundary Code enters IR2.

IR2=15? YES. (Sub. "1" from IR1; Set RDF; Shift IR Right; Destroy Boundary Code)

When the Boundary Code arrives in the IR2 position, the ±1F is set at DT14 time via gates 203 and 204. Then in DTO time of the next data pass, a "1" is subtracted from IR1 via ASO and the output of ASO is gated into IR14 via gates 150 and 155. During this same DTO time, the Boundary Code information in IR2 is shifted into IR1 and this is sensed for the first time as $\overline{IR1=0}$. Gate 52 is then enabled and the RDF FF is set.

In DT1 time of CC8, the IR2 data (Boundary Code) is again inhibited by gate 163 and therefore the Boundary Code is destroyed because it has no place to go.

During DT2 through DT13 time, the contents of IR3 through IR13 are gated into IR13 in via gates 163 and 164. At DT14 time, shifting stops and the number in IR is now in its original position. The CC will then step to CC9 via gate 135.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

CC9 EXCHANGE IR AND AR; RESET MLF, POF, AND SCF FF's

In this step, the answer developed in AR is always exchanged with the contents of IR. IR is fed into AR14 in via FAS. AR is fed into IR14 in via gate 156. IR input to FASy is via gate 104 (Y2.IR). NOTE: Y2 is a logic signal developed by CC9 at gate 156, as a control signal for exchanging AR with IR.

AR is blocked from FASx at gate 99, therefore, IR is added to zero without changing value and the output from FAS goes into AR14 in via gate 173 and 174. The product is now in IR and the multiplicand is in AR. The SCF, POF, and MLF FF's are reset because of the CC9 signal at their reset inputs. The CC will now step to CC0 via gate 139. NOTE: If a series of subtractions was done, the signs of IR and AR are negative. If a series of additions was done, the signs of IR and AR are positive.

CC0 RESET RDF FF

As the CC steps to CC0, the RDF FF is reset via gate 54. The Mult function is completed and the Conditional Control Phase Counter is ready to start another function.

MULTIPLY (Basic Logic for CCO.$\overline{SCF}$; Square Function)

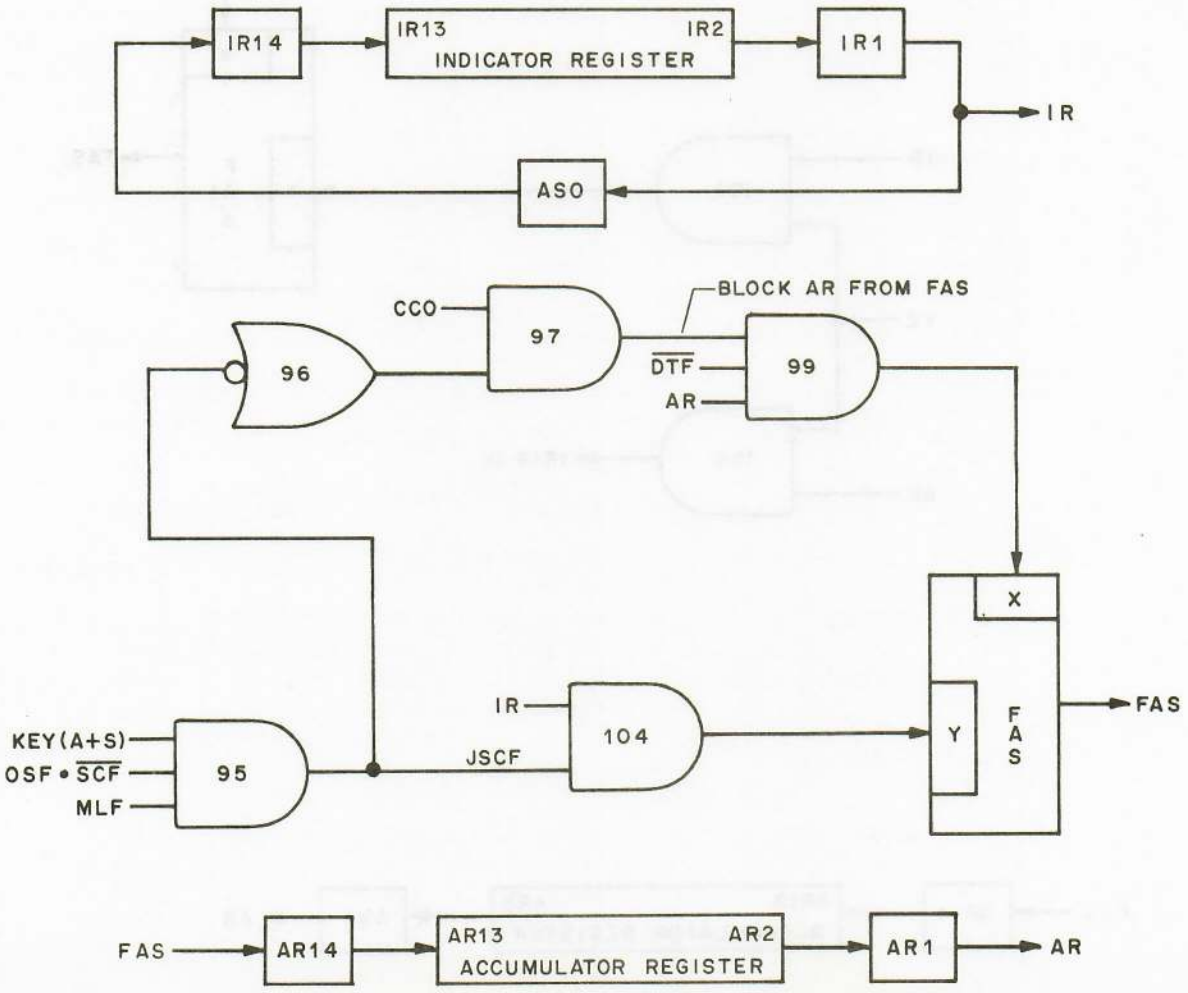


FIGURE 6-16

TDA-169SI

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

MULTIPLY (Basic Logic for CC3 and CC9)



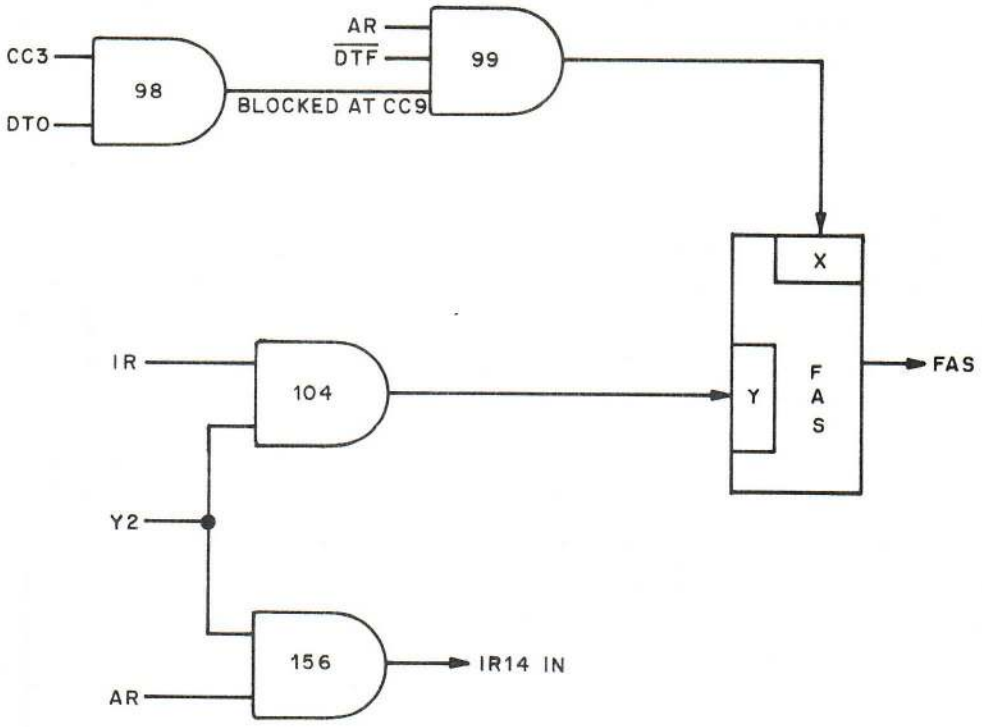




FIGURE 6-17

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

MULTIPLY (Basic Logic for CC4)

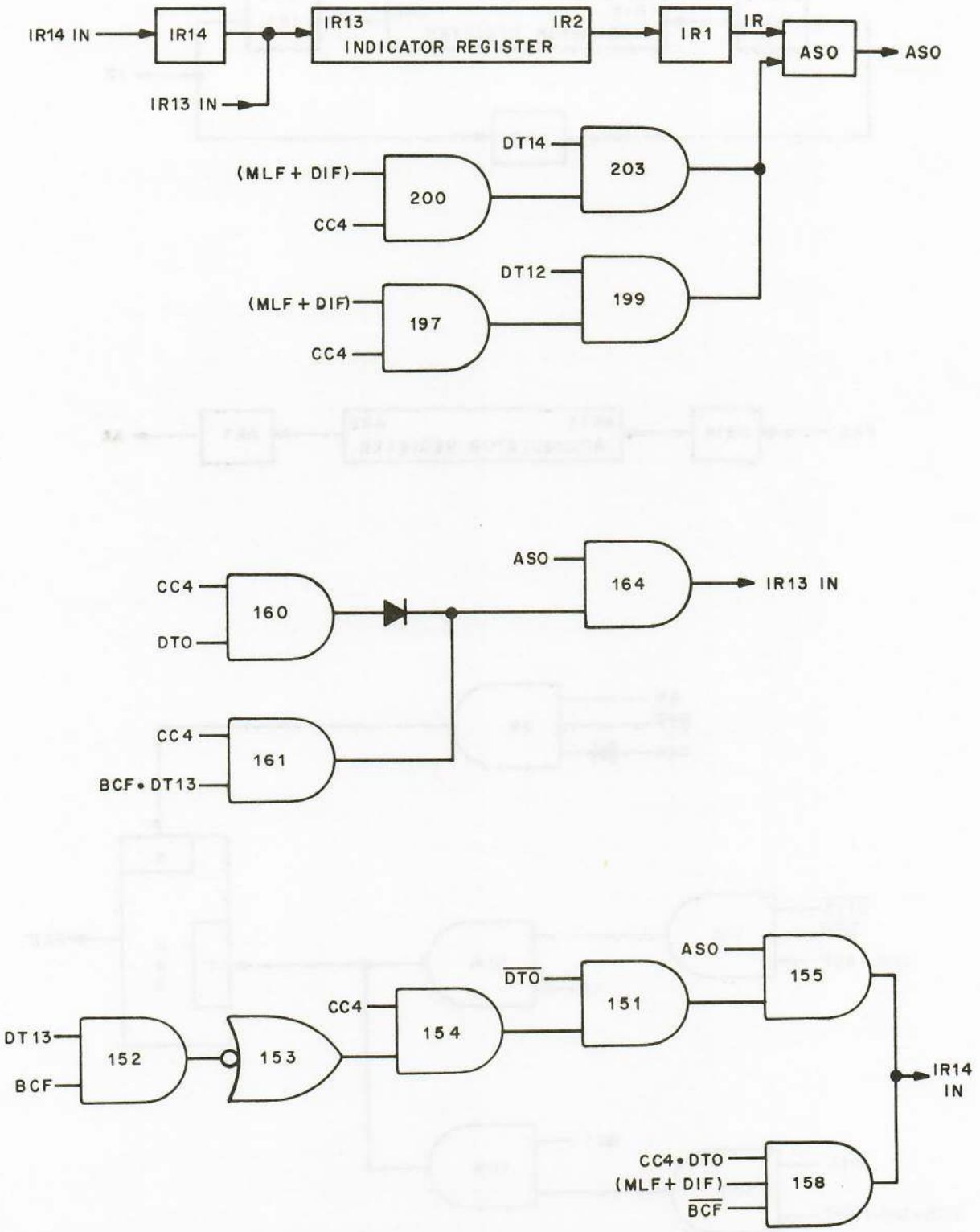


FIGURE 6-18

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

MULTIPLY (Basic Logic CC6.SCF)

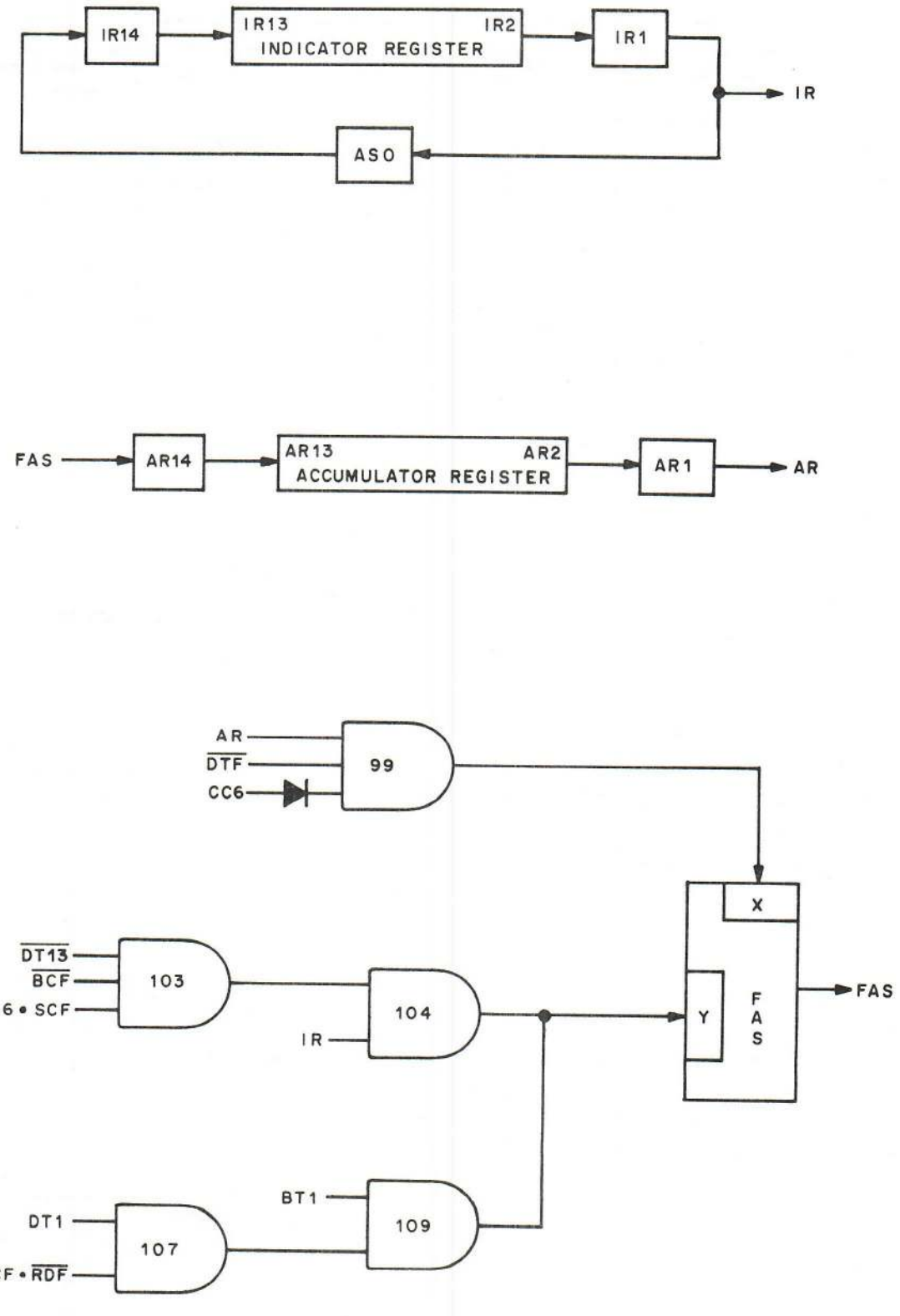


FIGURE 6-19

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

MULTIPLY (Basic Logic CC6.$\overline{SCF}$)

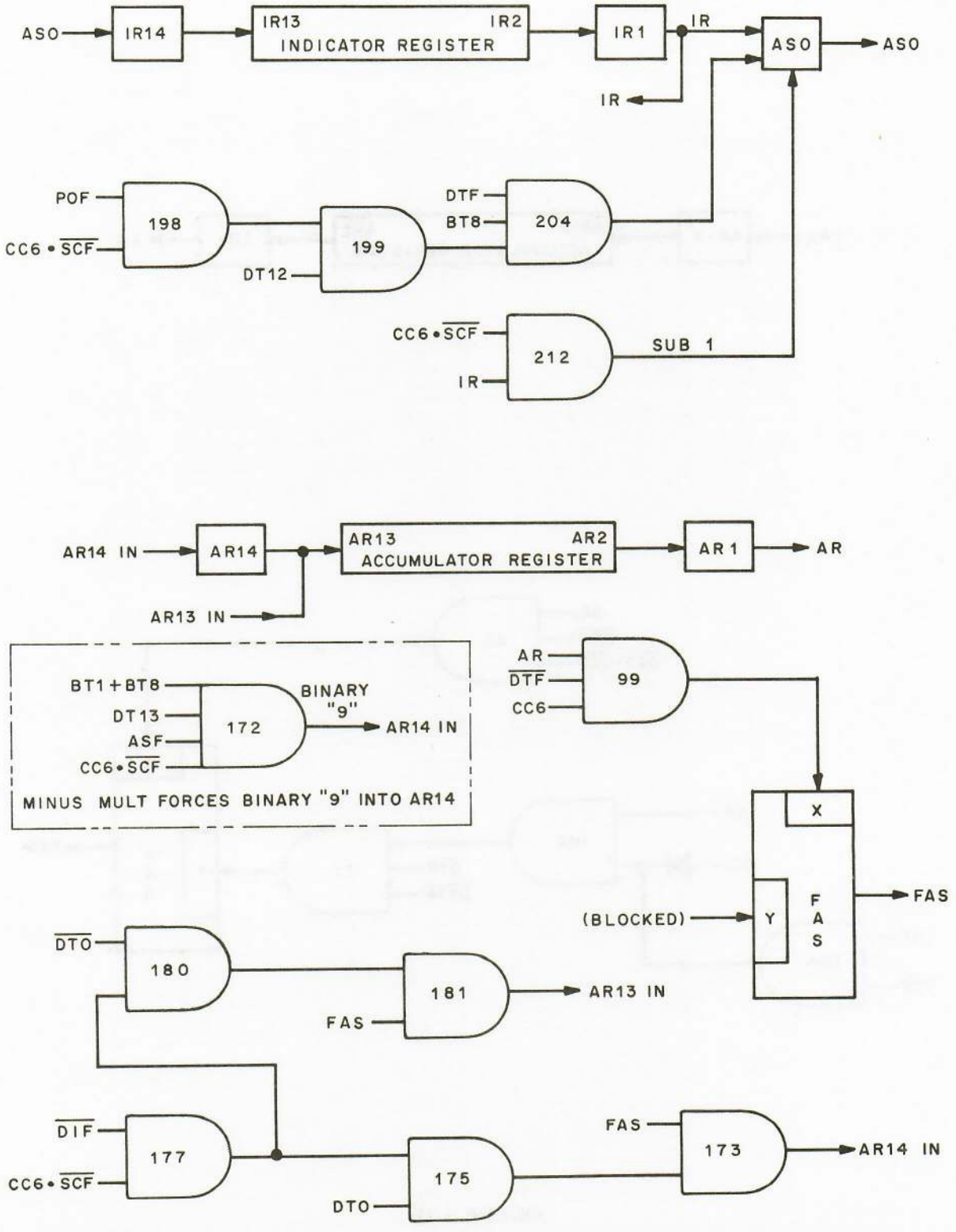


FIGURE 6-20

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

MULTIPLY (Basic Logic CC7; ASF Set)



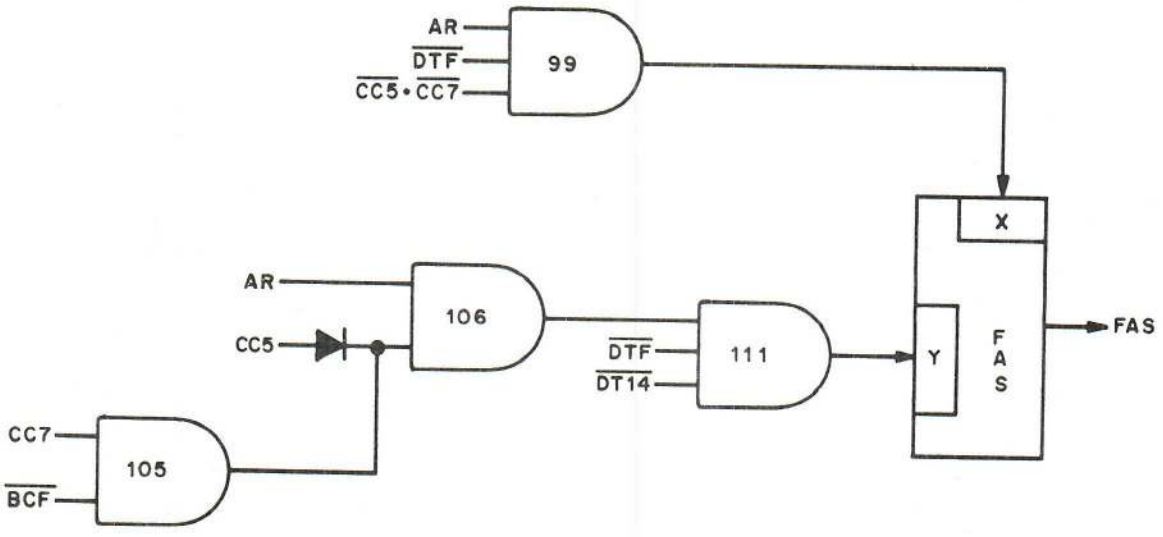
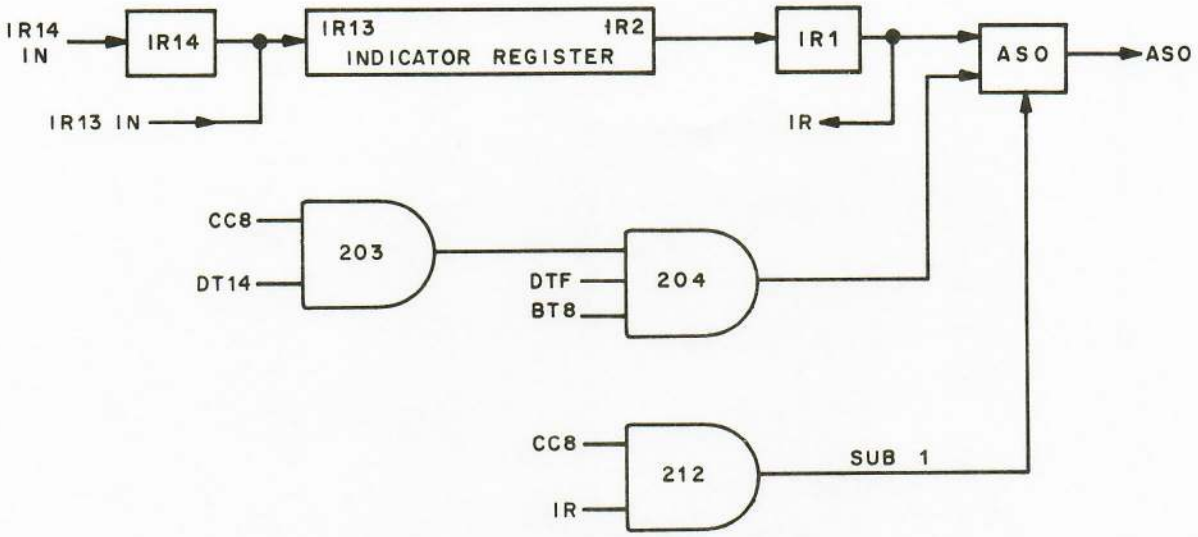


FIGURE 6-21

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

MULTIPLY (Basic Logic for CC8)



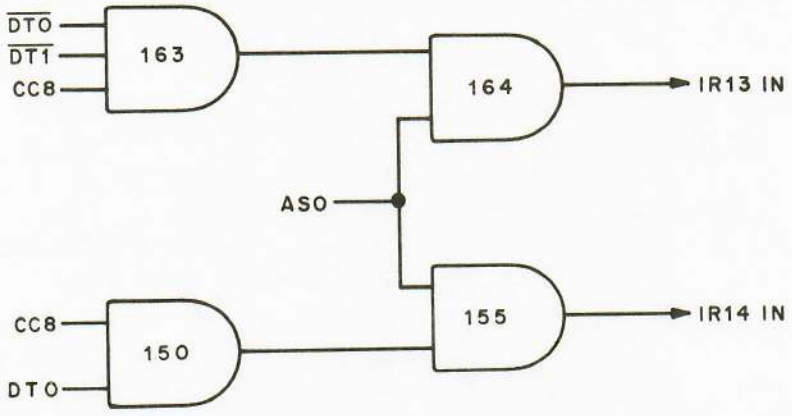


FIGURE 6-22

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

MULT FLOW CHART

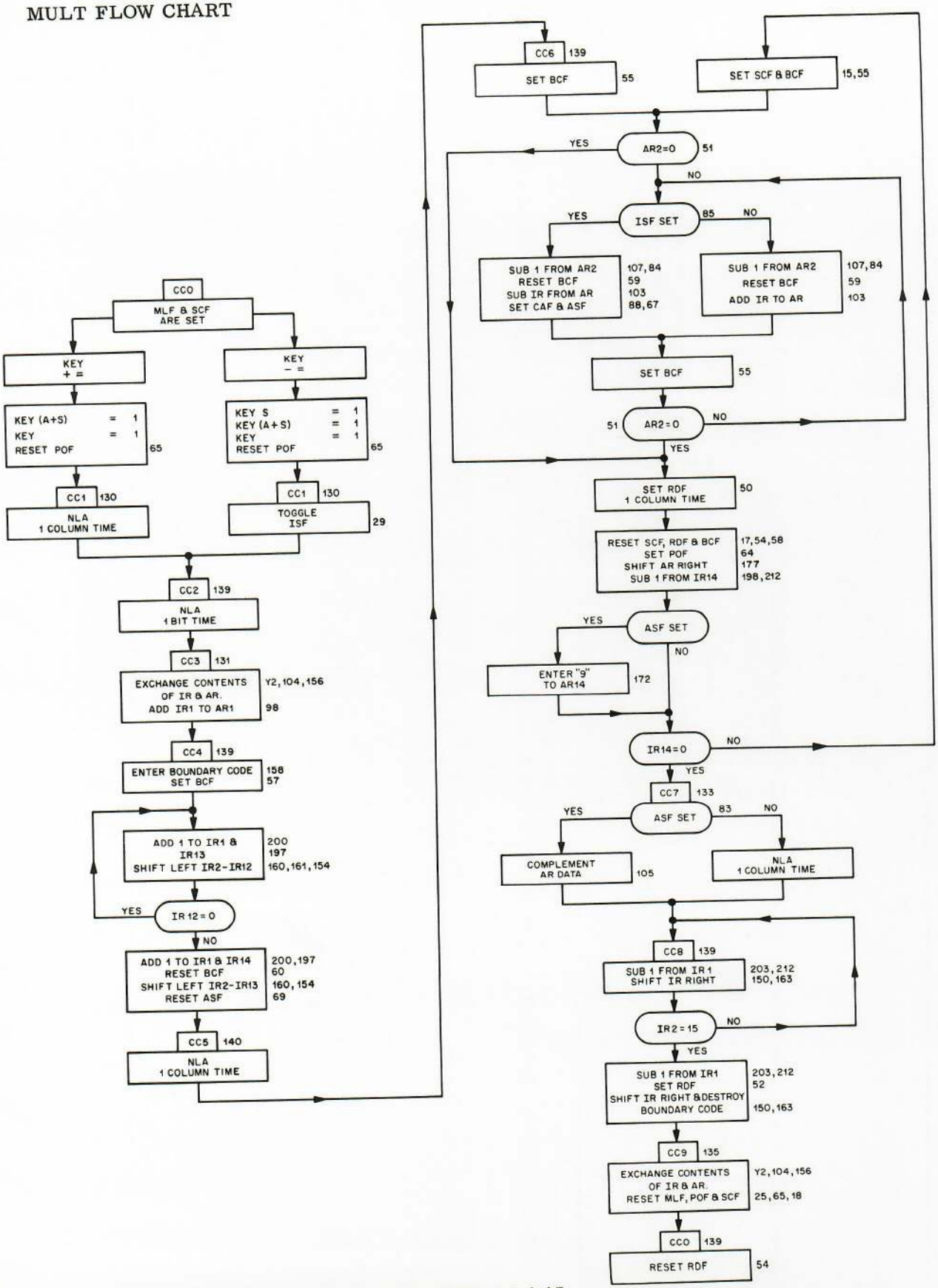


FIGURE 6-15

TPA-169SL

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

DIVIDE

A divide operation can be described as a series of subtractions in which the answer (quotient) is determined by the number of times the divisor can be successfully subtracted from the dividend. For example, to divide 144 by 12, it is found that the divisor (12) can be successfully subtracted from the dividend (144) exactly 12 times. However, depending on the numbers involved, the divisor may not subtract out an equal number of times from the dividend, in which case a remainder would occur which may or may not be a significant part of the answer.

In the 1113, the divide operation is performed by subtracting the number in IR (divisor) from the number in AR (dividend) as many times as possible. Then, an unsuccessful subtraction occurs which causes an overdraft. The overdraft is corrected by an add cycle and then the dividend in AR is shifted one place to the left and another series of subtractions take place. Each successful subtraction is accumulated in AR as an intermediate quotient until the problem is completed. The contents of AR and IR are then exchanged and the final quotient of the divide problem is in IR.

SAMPLE PROBLEM

For purposes of illustration, the various steps involved in doing a divide problem are shown in the work-sheet example, Figure 6-23. The numbers are: 11111.111 divided by 3333.33333 equals 3.3.

DIVIDE FUNCTION

$\underbrace{11111}_{5\text{ DIGITS}} . \underbrace{111}_{3\text{ DIGITS}} \div \underbrace{3333}_{4\text{ DIGITS}} . \underbrace{33333}_{5\text{ DIGITS}} = 3.3$

INDICATOR REGISTER														ACCUMULATOR REGISTER												SCF	ISF	CCC	ENTRIES COMPLETE / REMARKS
14	13	12	11	10	9	8	7	6	5	4	3	2	1	14	13	12	11	10	9	8	7	6	5	4	3	2	1				
				3	3	3	3	3	3	3	3	3	5						1	1	1	1	1	1	1	1	3	1	0	0	DEPRESS KEY +=
				3	3	3	3	3	3	3	3	3	5						1	1	1	1	1	1	1	1	14	1	0	1	AR1 - IR1
				3	3	3	3	3	3	3	3	3	5					1	1	1	1	1	1	1	1		15	1	0	2	AR LEFT; AR1+1
				3	3	3	3	3	3	3	3	3	5				1	1	1	1	1	1	1	1			0	1	0	2	" " " "
				3	3	3	3	3	3	3	3	3	5				1	1	1	1	1	1	1	1			0	1	0	3	NLA (1 BIT TIME)
	1		3	3	3	3	3	3	3	3	3	(15)	6				1	1	1	1	1	1	1	1			15	1	0	4	ENTER BC; IR LEFT;
	2	3	3	3	3	3	3	3	3	3	(15)		7				1	1	1	1	1	1	1	1			14	1	0	4	IR1 ↕ IR13+1; AR1-1
3	3	3	3	3	3	3	3	3	3	(15)			8				1	1	1	1	1	1	1	1			13	1	0	4	IR LEFT; IR1 ↕ IR14+1; AR1-1
3	3	3	3	3	3	3	3	3	3	(15)			8				1	1	1	1	1	1	1	1			13	1	0	5	NLA (1 C TIME)
3	3	3	3	3	3	3	3	3	3	(15)			8	9	6	6	7	7	7	7	7	7	8	1		1	13	1	1	6	AR2+1; AR-IR
																	1	1	1	1	1	1	1	1		0	13	1	0	6	AR2-1; AR+IR
																1	1	1	1	1	1	1	1			0	14	0	0	6	AR1+1; AR LEFT
														9	6	7	7	7	7	7	7	7	8			1	14	1	1	6	AR2+1; AR-IR
																1	1	1	1	1	1	1	1			0	14	1	0	6	AR2-1; AR+IR
2	3	3	3	3	3	3	3	3	3	(15)			8		1	1	1	1	1	1	1	1	1				0	15	0	0	6	AR1+1; AR LEFT; IR14-1

FIGURE 6-23 (Continued on next page)

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

INDICATOR REGISTER														ACCUMULATOR REGISTER												SCF	ISF	CCC	REMARKS
14	13	12	11	10	9	8	7	6	5	4	3	2	1	14	13	12	11	10	9	8	7	6	5	4	3	2	1				
														9	7	7	7	7	7	7	7	7	7			1	15	1	1	6	AR2+1; AR-IR
															1	1	1	1	1	1	1	1				0	15	1	0	6	AR2-1; AR+IR
1	3	3	3	3	3	3	3	3	3	15			8	1	1	1	1	1	1	1	1					0	0	0	0	6	AR1+1; ARLEFT; IR14-1
															7	7	7	7	7	7	7	6	7			1	0	1	1	6	AR2+1; AR-IR
															4	4	4	4	4	4	4	3	4			2	0	1	1	6	" "
															1	1	1	1	1	1	1	0	1			3	0	1	1	6	" "
														9	7	7	7	7	7	7	7	6	8			4	0	1	1	6	" "
															1	1	1	1	1	1	1	0	1			3	0	1	0	6	AR2-1; AR+IR
0	3	3	3	3	3	3	3	3	3	15			8	1	1	1	1	1	1	1	0	1			3	0	1	0	0	6	AR1+1; ARLEFT; IR14-1
															7	7	7	7	7	6	7	7			3	1	1	1	1	6	AR2+1; AR-IR
															4	4	4	4	4	3	4	4			3	2	1	1	1	6	" "
															1	1	1	1	1	1	0	1	1		3	3	1	1	1	6	" "
														9	7	7	7	7	7	7	6	7	8		3	4	1	1	1	6	" "
0	3	3	3	3	3	3	3	3	3	15			8		(1	1	1	1	1	1	0	1	1)		3	3	1	1	0	6	AR2-1; AR+IR
	3	3	3	3	3	3	3	3	3	15			8												3	3	1	0	0	7	DESTROY REMAINDER
		3	3	3	3	3	3	3	3	3	15		7												3	3	1	0	0	8	IR RIGHT; IR1-1
			3	3	3	3	3	3	3	3	3	15	6												3	3	1	0	0	8	" "
				3	3	3	3	3	3	3	3	3	5												3	3	1	0	0	8	IR RIGHT; IR1-1;
																															DESTROY BOUNDARY CODE
											3	3	1					3	3	3	3	3	3	3	3	3	5	0	0	9	EXCHANGE IR & AR
											3	3	1					3	3	3	3	3	3	3	3	3	5	0	0	0	END (QUOTIENT IN IR)

FIGURE 6-23

DIVIDE FLOW CHART ANALYSIS

The divide operations are explained in conjunction with the Divide Flow Chart Figure 6-24 as follows:

Depressing the Divide key establishes the logic signal Key D at logic 1. With Key D at logic 1, the DIF FF is set via gate 26; and the SCF, ISF, POF, and MLF FF's are reset via their respective reset gates.

With CC0.$\overline{SCF}$ both true, the second number (divisor) is entered in IR and the first number (dividend) is shifted into AR the same as described for Entry. Then with both numbers entered, the Plus Equal key is depressed to initiate the Divide operation. NOTE: The Minus Equal key will also initiate a Divide operation but the machine is not designed to do a negative division.

CC0 DEPRESS THE PLUS EQUAL KEY

Depressing the Plus Equal key causes the logic signal, Key and Key (A+S) to be true (logic 1) which resets the POF FF and the Conditional Control Phase Counter (CC) steps to CC1 via gate 130.

CC1 COMPARE DECIMAL POINT INFORMATION; SUB IR1 FROM AR1; IF IR1 GREATER THAN AR1, SET CAF & POF FF'S

In CC1 a comparison of decimal point information between AR1 and IR1 is made to determine which is greater. If IR1 is found to be greater than AR1, then the number in AR must be decimally aligned in CC2 by shifting AR left. If AR1 is found to be greater than IR1, no decimal alignment is required.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

The operation of subtracting IR1 from AR1 is accomplished in FAS the same as described for CC1 Add or Subtract which may be referred to if desired. If IR1 is greater than AR1, the CAF and POF FF's will set via gates 88 and 63. The CC will step to CC2 via gate 139.

CC2 POF SET? YES. DEC. ALIGN AR; ADD 1 TO AR1; SHIFT AR LEFT

If IR1 was greater than AR1 as determined in CC1, a series of left shifts of the number in AR is required in CC2 for the purpose of decimal alignment. Each left shift of AR adds "1" to AR1 until AR1=0; then, the CC steps to CC3. The logic analysis is the same as described for CC2 Add or Subtract which may be referred to if desired.

POF SET? NO. NO LOGIC ACTION 1 BIT TIME; STEP TO CC3 VIA GATE 131

CC3 NO LOGIC ACTION 1 BIT TIME; STEP TO CC4 VIA GATE 132

CC4 ENTER BOUNDARY CODE; SET BCF FF

The functions in CC4 DIVIDE are quite similar to the functions described for CC4 MULTIPLY with the exception that in CC4 DIVIDE, a "1" is subtracted from AR1 each time IR is shifted left.

On the first data pass in CC4, the Boundary Code 15 (1111) is gated into IR14 in via gates 158 and 160 during the first half of DTO. Then, when the clock pulse CFCP1 occurs in the last half of DTO, the BCF FF is set via the inverter gate 57. With the BCF FF set, gate 158 will be blocked to prevent entering more than one Boundary Code.

ADD "1" TO IR1 AND IR13; SUB 1 FROM AR1; SHIFT LEFT IR2-IR12

The $\pm$1F was enabled to be set via gates 197, 200, and 204 during the last half of DT14 time; and is set in DTO time as the Boundary Code is being entered. Then, as IR is gated into ASO, a "1" is added to IR1 as decimal point information and the ASO output is gated into IR13 in via gates 160 and 164. AR1 decimal point information is counted down by "1" at each DTO time by forcing a "1" into FASy via gates 108, 109, and 111. The CC4 "OR" input below gate 85 makes FAS act as a subtracter.

The contents of IR2 through IR12 are gated into IR14 in via gates 151, 154, and 155. At DT11 time, the IRCP1 clock pulse is blocked via gates 43 and 44 which causes the contents of IR (except decimal and shift information) to be shifted left.

IR12=0? YES. Continue shifting the contents of IR (except decimal and shift information) to the left.

IR12=0? NO. ADD 1 TO IR1 & IR14; SUB 1 FROM AR1: RESET BCF FF; SHIFT LEFT IR2 THROUGH IR13; RESET ASF FF

When the MSD of the divisor reaches IR12, one more left shift is required to move it into IR13. As previously explained in Mult, during this left shift data pass, the decimal point information in IR1 is incremented by "1" via ASO and the output is gated into IR13. Also, AR1 decimal point information is counted down by "1" by forcing a "1" into FASy via gates 108, 109, and 111. During DT1 through DT10, normal data flow of IR2 through IR11 occurs. In DT11, the IRCP1 clock pulse is blocked which causes a left shift of IR. In DT12 time, normal data flow shifts the MSD of the divisor into IR14 where it is sensed as $\overline{IR14=0}$ at logic 1 (true)

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

for the first time. This signal being true at gate 60 enables the BCF FF to reset. Then in DT13 time, gate 161 will go false because the BCF FF is now reset; IR13 shift data will then be incremented by "1" via ASO as before; but since gate 161 is now blocked, IR13 shift data is gated into IR14 in via gates 151, 152, 153, 154, and 155. At the same time the shift data is entering IR14, the MSD of the divisor is shifting out of IR14 and into IR13. In DT14 time, shifting stops and the CC steps to CC5 via gate 140. (NOTE: The output of gate 140 also resets the ASF FF via gate 69 which insures that the dividend number in AR will not be complemented in CC5.)

CC5 NO LOGIC ACTION 1 C TIME; STEP TO CC6 VIA GATE 139

CC6 CC6 consists of two phases: CC6.SCF and CC6.$\overline{\text{SCF}}$. In CC6.SCF, the divisor in IR is subtracted from AR as many times as necessary to cause an overdraft; with each subtraction causing a "1" to be added to AR2 as an intermediate quotient. The overdraft is then corrected by adding IR to AR, and the intermediate quotient is corrected by subtracting "1" from AR2.

In CC6.$\overline{\text{SCF}}$, the contents in AR are shifted left one digit position, a "1" is added to AR1, and again the phase CC6.SCF occurs to cause a series of subtractions of IR from AR; add "1" to AR2; overdraft and correction.

After one or more left shifts of AR, the dividend will be decimally aligned with the divisor and the POF FF will set indicating that alignment has been achieved. The two phases of CC6 continue as above, but now each phase of CC6.$\overline{\text{SCF}}$ causes a "1" to be subtracted from IR14. When IR14 equals zero no further left shifting of AR occurs and when the subtract, overdraft and correction function of CC6.SCF are completed, the quotient and remainder if any are in AR, the final decimal point information is in AR1, and CC steps to CC7.

CC6.SCF ADD 1 TO AR2; SUBTRACT IR FROM AR

The ISF and BCF FF's are set via gates 31 and 55 respectively. Then, in DT1 time, AR2 information is shifted into FASx via gate 99 and at the same time a "1" is forced into FASy via gates 107 and 109 and is added to AR2.

IR contents are shifted out of IR1 and into IR14 via ASO in a normal dynamic data flow until the boundary code 15 is detected in IR14 (IR14=15) at which time the BCF FF is reset via gate 59. With the BCF FF reset, the FASy gate 103 is enabled and now the divisor in IR is gated into FASy. In the meantime, the normal data flow of AR has been going through FASx via gate 99, therefore, as the first digit in IR following the boundary code goes into FASy, FAS acts as a subtracter via gates 85 and 88 and the divisor in IR is subtracted from the dividend in AR. (IR14 is blocked from FASy via gate 103 and therefore is not subtracted from AR14.)

At the beginning of the next data pass, the BCF FF is set via gate 55. The setting and resetting of the BCF FF will thus control when the divisor in IR is shifted into FAS to be subtracted from AR.

CAF SET? NO. Continue adding "1" to AR2 and subtracting IR from AR during each column time as described above until finally an overdraft occurs which will cause the CAF FF to set at DT14 time.

CAF SET? YES. With CAF set, the ASF FF will set via gate 67, the BCF FF will set via gate 55, and the ISF FF will reset via gate 36. (ISF FF being reset allows FAS to act as an Adder on the next data pass in order to correct the overdraft.)

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

CORRECT OVERDRAFT. SUBTRACT 1 FROM AR2; RESET BCF; ADD IR TO AR

Because of the overdraft, a correction must be made to restore the overdraft and correct the extra count in AR2. Therefore, on the next data pass, instead of adding "1" to AR2, a "1" is subtracted from AR2. Likewise, instead of subtracting IR from the dividend in AR, IR is added to AR. Subtracting "1" from AR2 occurs at DT1 time by AR2 information going into FASx via gate 99 and at the same time, a "1" is forced into FASy via gates 107 and 109. FAS is made to act as a subtracter via gates 84 and 88. Adding IR to AR occurs when the Boundary Code 15 reaches IR14 (IR14=15) to cause the BCF FF to reset via gate 59. IR will then shift into FASy via gate 103, and as the first digit in IR following the Boundary Code goes into FASy, the divisor in IR is added to AR.

IR14=0? YES. RESET ASF AND STEP CC TO CC7

If IR14=0, this means that the quotient has been completed in AR and that the CC is ready to step to CC7 via gate 134. The ASF FF is reset via gate 70 to insure that the quotient is not complemented in CC7.

IR14=0? NO. RESET SCF, ASF; ADD 1 TO AR1; SHIFT AR LEFT; *SET POF

The ASF FF is reset via gate 70 and the SCF FF is reset via gate 19. During DTO time, a "1" is added to AR1 via gates 179 and 109. Because AR1 is not to be shifted left it is gated into AR13 via gates 178, 179, and 181. The contents of AR2 through AR14 enters FASx via gate 99 and into AR14 via gates 173, 176, and 178. At DT11 time, the ARCP1 shift pulses are blocked via gate 42 causing the contents of AR to shift left one digit position.

*Set POF FF. In CC6.$\overline{\text{SCF}}$ of a Divide, the POF FF is set via gates 61 or 62 when AR1 is sensed at DT11 as $\overline{\text{AR1=0}}$, or at DT14 time as AR1=0 or AR1=15. For example, if the dividend in AR is such that the MSD will shift into AR13 during the AR left shift, $\overline{\text{AR1=0}}$ will be sensed at DT11 time and the POF FF will set via gate 62. Otherwise, one or more AR left shifts will occur, adding "1" to AR1 until AR1=0 or 15 is sensed at DT14 time and then the POF FF will set via gate 61.

POF SET? NO. Set the SCF FF via gate 15 and continue to alternate between the two phases of CC6 until POF is set via gates 61 or 62 as previously explained.

POF SET? YES. SUBTRACT 1 FROM IR14

As explained previously, if the POF FF is set, the decimal alignment of the dividend to the divisor is complete and the final decimal alignment of the quotient will begin by subtracting "1" from IR14 on each left shift of AR until IR14=0.

With the POF FF set, the ±1F is set via gates 198, 199, and 204. Then at DT13 time, a "1" will be subtracted from IR14 via ASO (gate 212). The SCF FF will set via gate 15 and the operation continues to alternate between the two phases CC6 until IR14=0. When IR14=0, the decimal alignment of the quotient and the arithmetic function of the Divide are complete. Therefore the ASF FF will reset via gate 70 and the CC will step to CC7 via gate 134.

CC7 DESTROY REMAINDER

In CC7, the remainder if any, of a Divide problem is in the AR digit positions to the left of the boundary code in respect to IR and must be destroyed. For example, at the end of CC6 as shown in the sample problem the number in AR13 through AR2 is: 111111011033. The quotient is: 03.3 and the remainder is: 111111011. In order to destroy the remainder but not disturb the quotient, the contents of AR

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

are shifted through FASy into AR14 via FASy gates 105, 106, and 111; while IR data flow is into IR14 via gate 155. Therefore, when the Boundary Code 15 arrives in IR14, the signal IR14=15 will be true and the BCF FF will set via gate 56. At this point, the quotient in AR has shifted into AR14 via gate 173 and the remainder is now in position to be shifted into FASy. However, since the BCF FF is now set, the input to FASy via gate 105 is blocked and the remainder is destroyed because it has no place to go. At DT14 time the CC will step to CC8 via gate 139.

CC8 SUBTRACT 1 FROM IR1; SHIFT IR RIGHT; DESTROY BOUNDARY CODE

In CC8, the contents of IR must be shifted right the same number of places that it was shifted left in CC4, and each right shift will subtract "1" from IR1 and destroy IR2 data. When the divisor in IR has shifted right to its original position, the Boundary Code will have been destroyed and IR1 will indicate the original decimal point number entered in CC0. The Flow Chart analysis of CC8 Divide is exactly the same as described for CC8 Mult which can be referred to if desired. The CC will step to CC9 via gate 135.

CC9 EXCHANGE IR AND AR. RESET DIF, POF, AND SCF FF's

In this step, the answer (quotient) developed in AR is exchanged with the contents of IR. IR is gated into AR14 in via FASy and gate 104 (Y2.IR). AR is gated into IR14 in via gate 156. (NOTE: Y2 is a logic signal developed by CC9 at the input to gate 156 and is used as a control signal for exchanging the contents of IR and AR.) AR is blocked from entering FASx at gate 99, therefore, IR is added to zero in FAS without changing value. The output from FAS is gated into AR14 in via gates 173 and 174. The answer (quotient) is now in IR and the divisor is in AR. The SCF and DIF FF's are reset via CC9 and clock pulse CCCP1. The POF FF is reset via CC9 and clock pulse CFCP1. The CC will now step to CC0 via gate 139. (NOTE: The signs of IR and AR are positive due to the resetting of ISF and ASF in CC6.)

CC0 RESET RDF AND BCF

The RDF and BCF FF's are reset by CC0 and clock pulse CFCP1 via gates 54 and 58. The Divide operation is now complete.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

DIVIDE (Basic Logic for CC1)

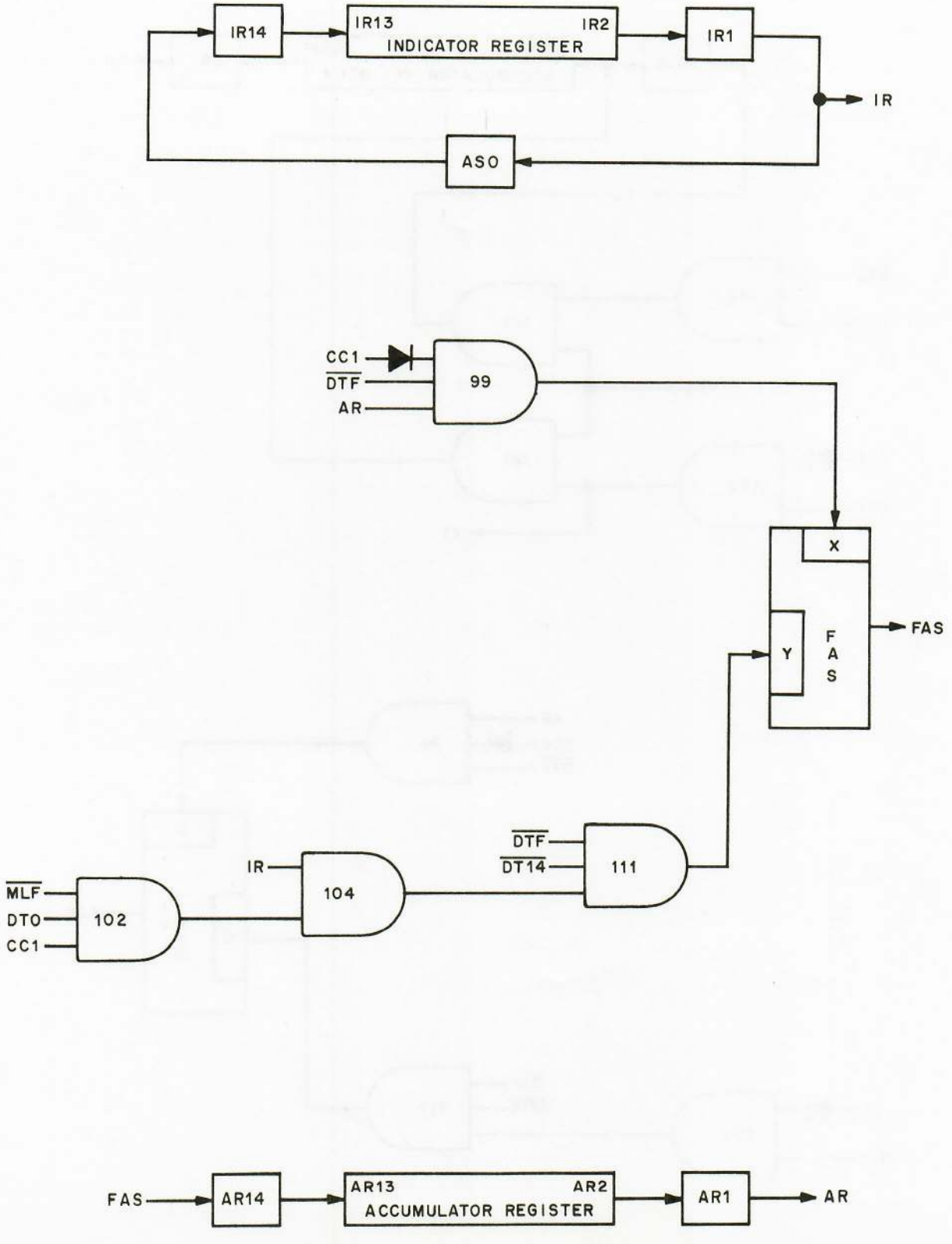


FIGURE 6-25

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

DIVIDE (Basic Logic for CC2)

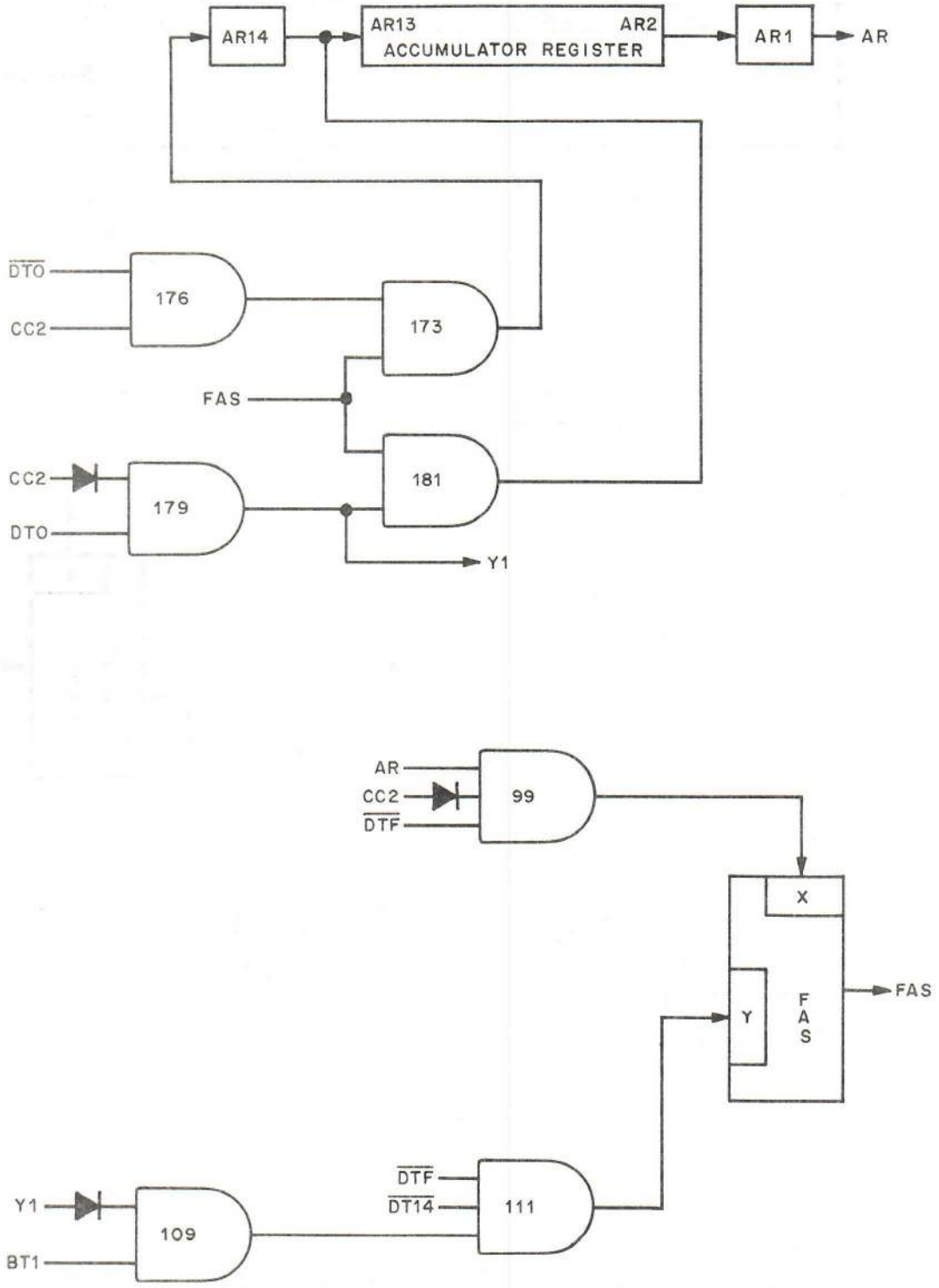


FIGURE 6-26

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

DIVIDE (Basic Logic for CC4)

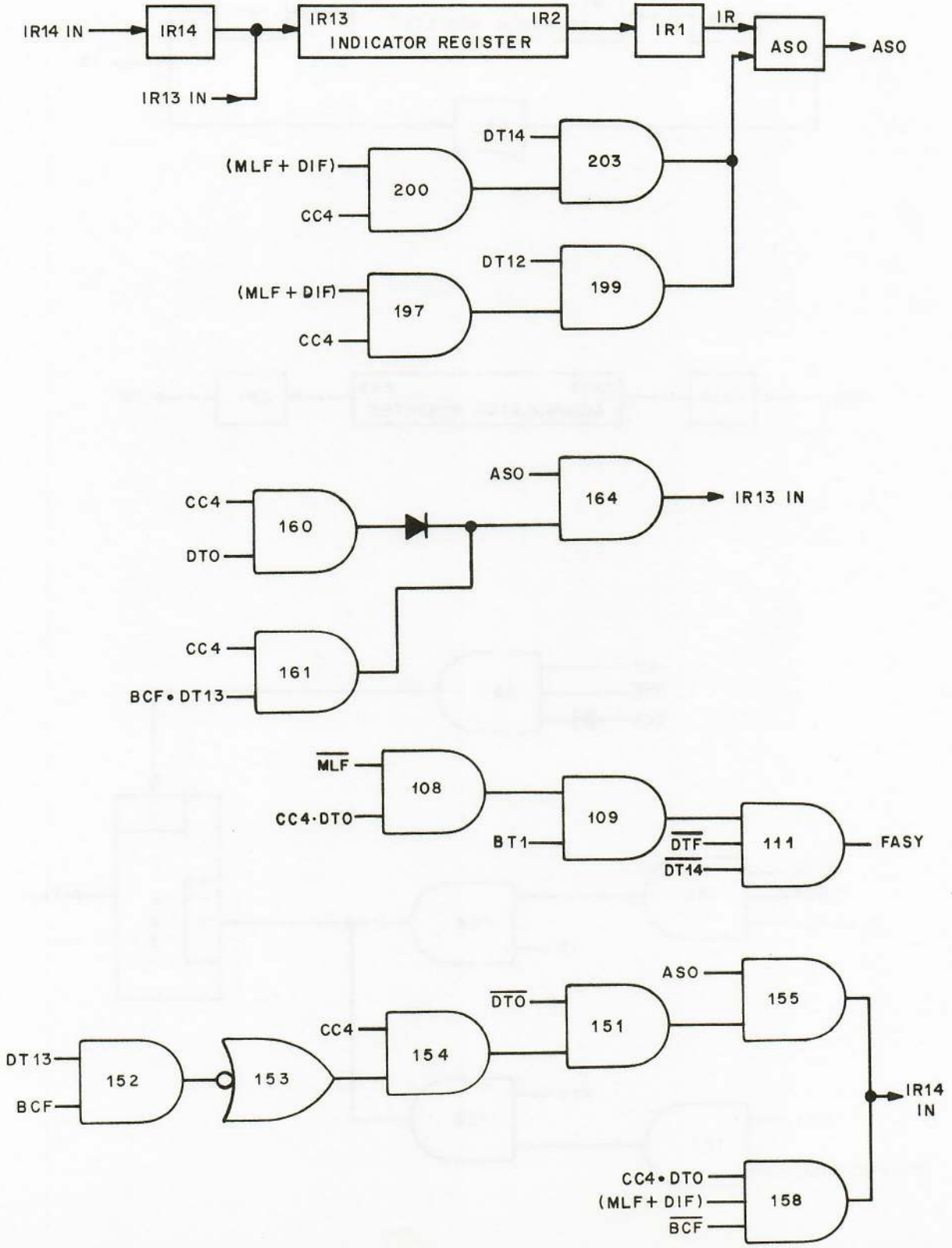


FIGURE 6-27

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

DIVIDE (Basic Logic for CC6.SCF)

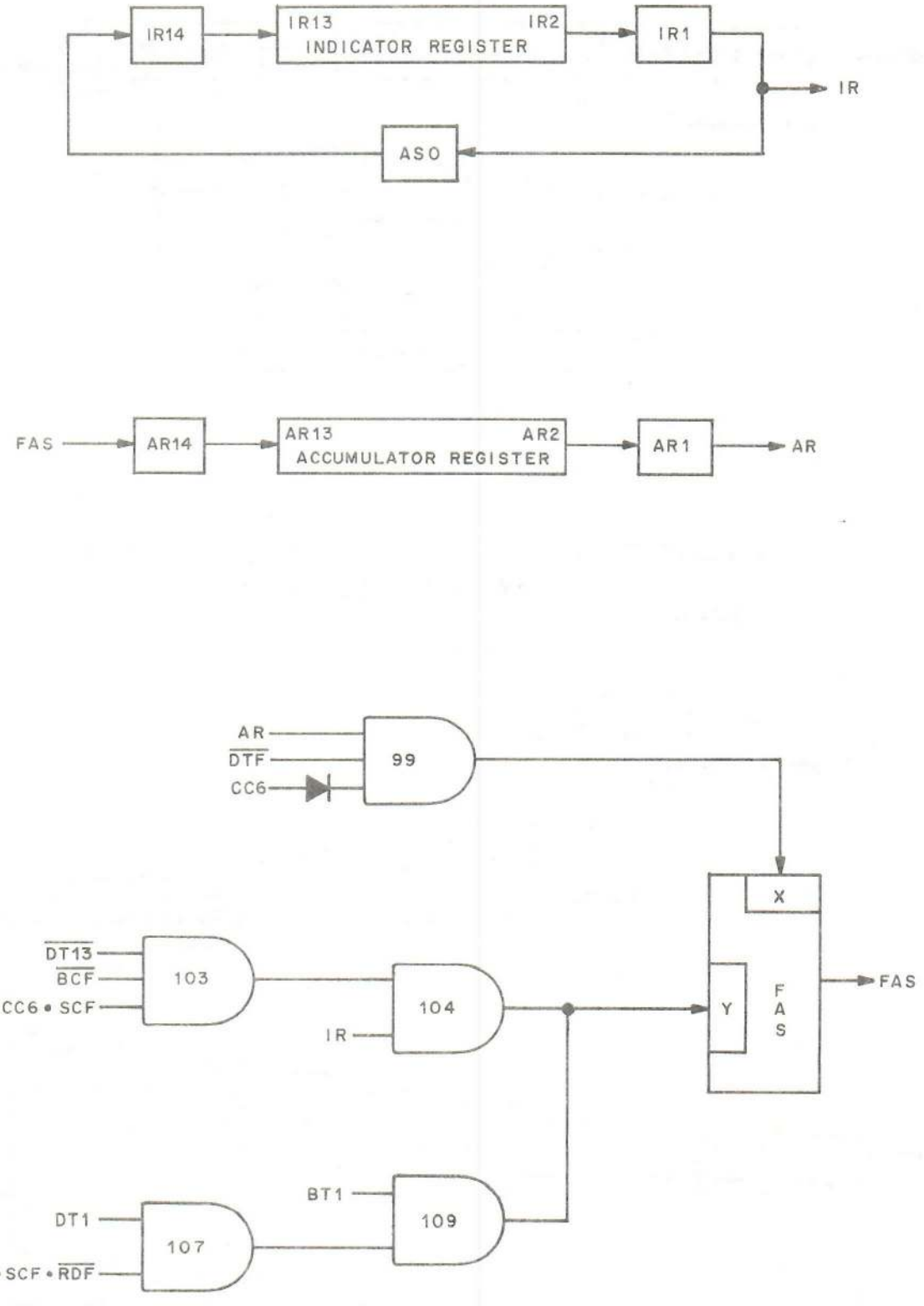


FIGURE 6-28

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

DIVIDE (Basic Logic for CC6.$\overline{SCF}$)

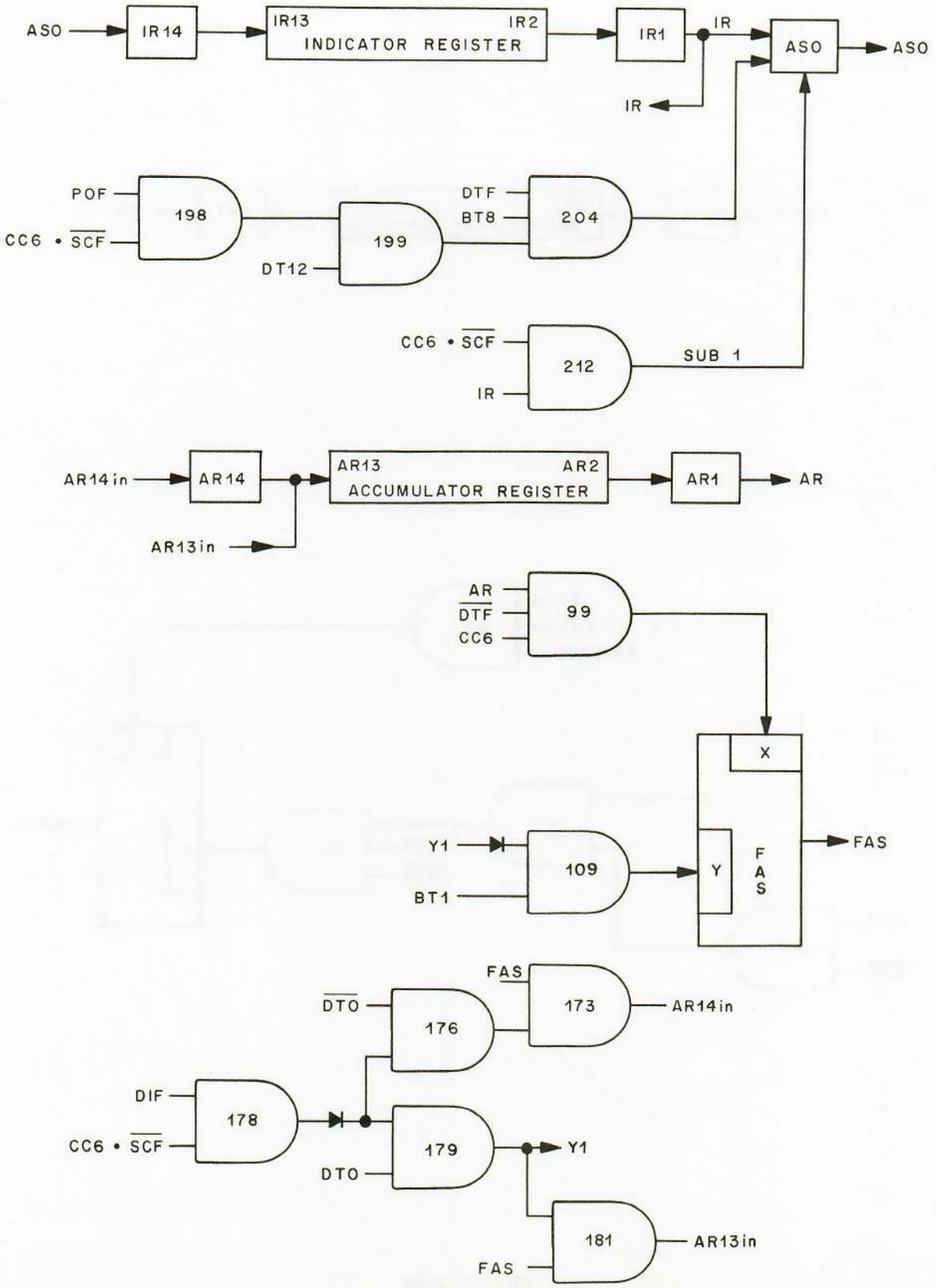


FIGURE 6-29

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

DIVIDE (Basic Logic for CC7)



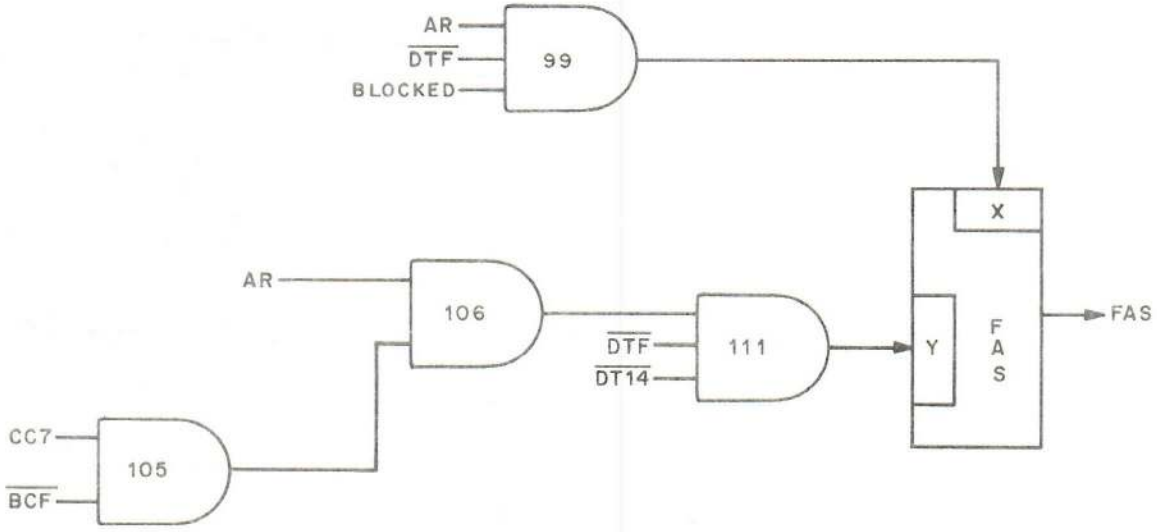


FIGURE 6-30

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

DIVIDE (Basic Logic for CC8)

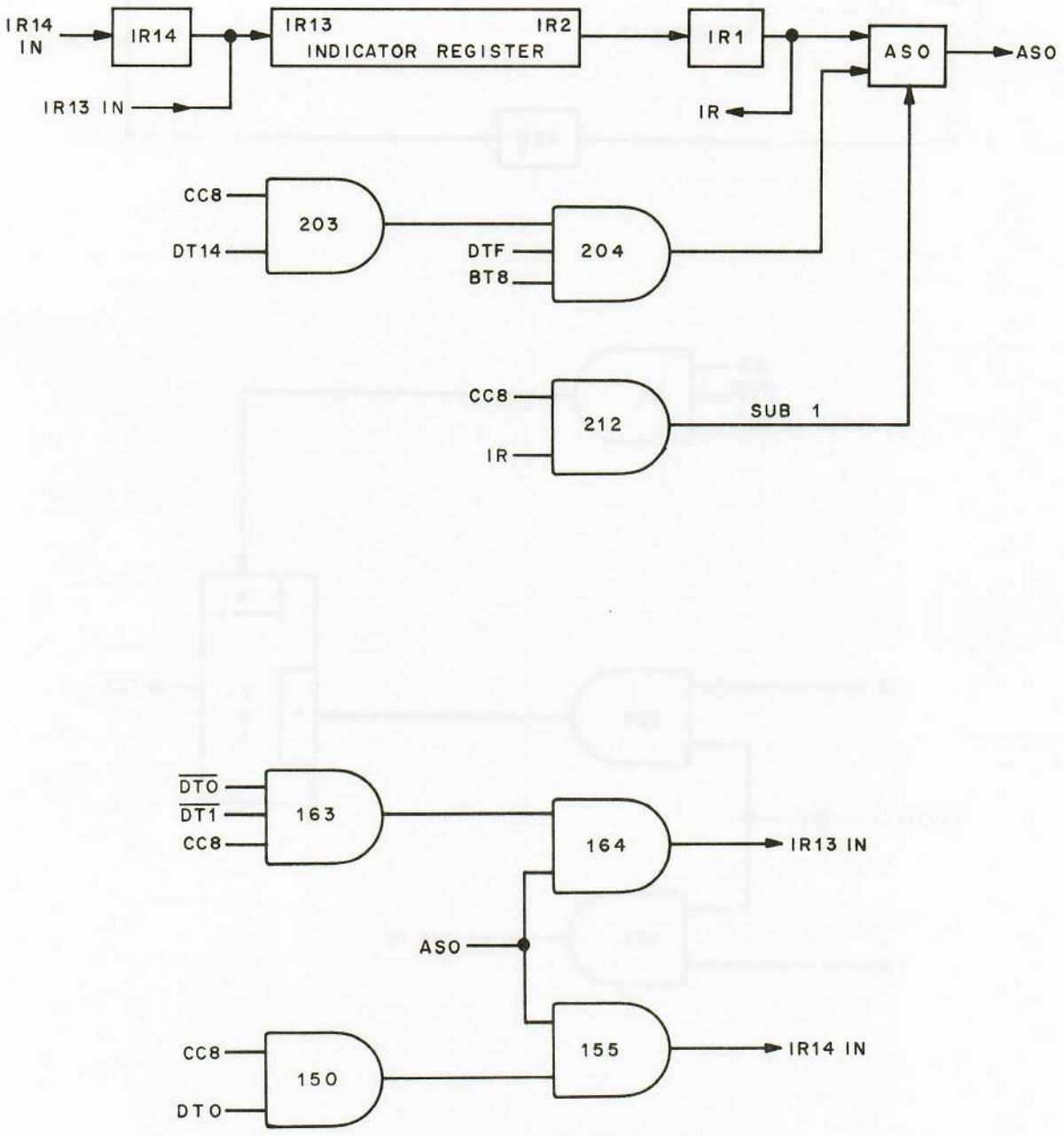


FIGURE 6-31

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

DIVIDE (Basic Logic for CC9)

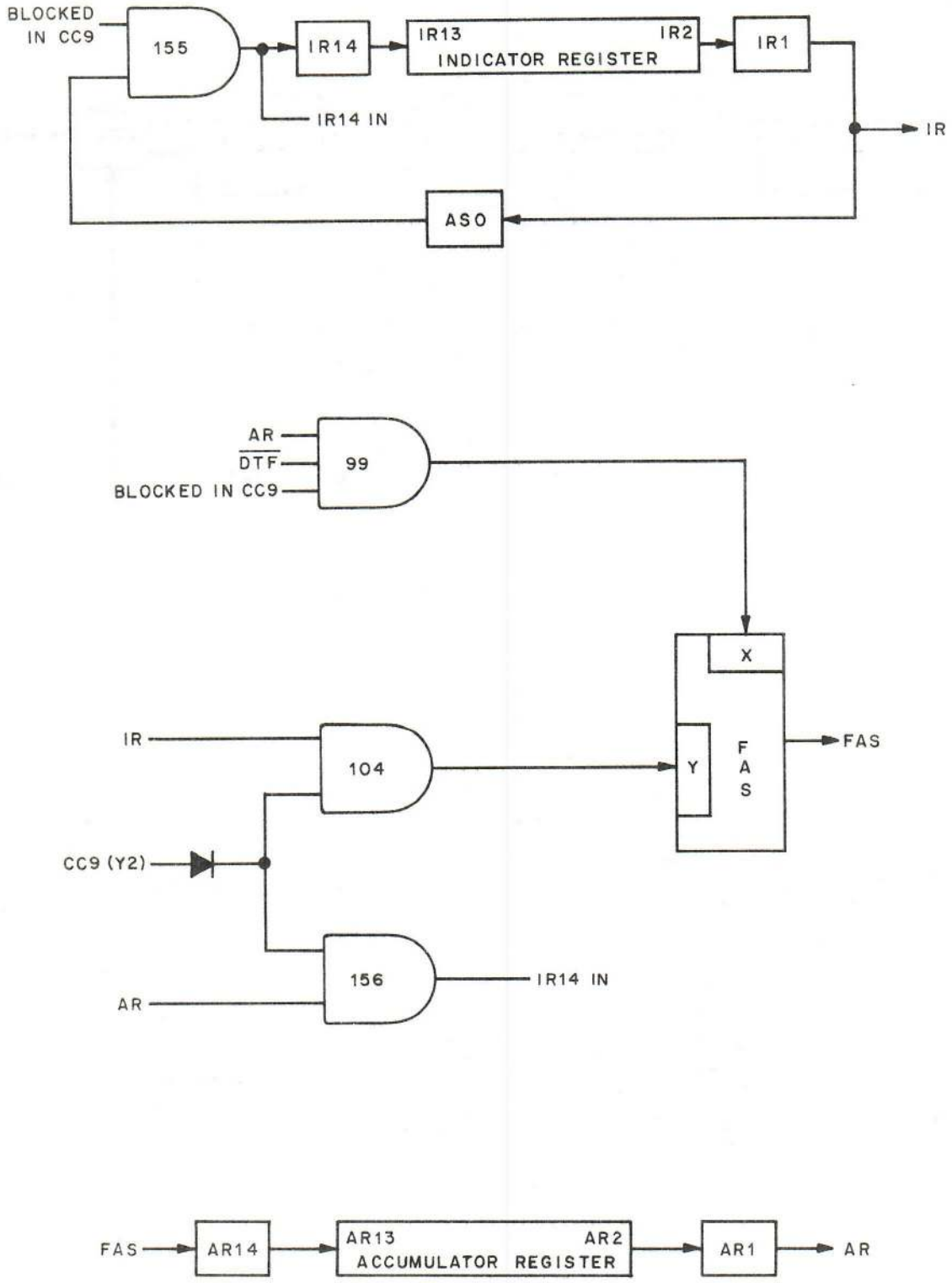


FIGURE 6-32

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

SIGMA

The Sigma key Σ (which means sum or accumulation) is used only in conjunction with a multiplication for the purpose of automatically adding or subtracting the product to the contents of the Memory register (MR). Once the Sigma key is depressed it is latched down and each subsequent multiplication product will be accumulated in MR. The key can only be released by redepressing it.

OUTLINE OF LOGIC FLOW IN SIGMA OPERATION

CCO, CC9 — Same as Multiply

CCO.MIF — Exchange contents of AR and MR, and their respective signs

CC1, CC8 — These functions are the same as an Add or Subtract

CC9 — NLA (1 Bit)

CCO.MIF — Exchange contents of AR and MR, and their respective signs

CCO — End

FLOW CHART ANALYSIS

The Sigma operation is explained in conjunction with the Sigma Flow Chart, Figure 6-33 as follows:

With the Sigma key depressed and latched down prior to a Mult or Mult Square operation, the Mult operation is performed as previously described in CCO. $\overline{SCF}$ to CC9. Therefore, only the logic flow from CC9 will be discussed.

CC9 — END OF MULT; SET MIF AND OSF; RESET MLF

At the end of the mult function, the product of the mult is in IR and the Multiplicand number is in AR.

The MIF FF is set via gates 187 and 188 which also produces the signal "Y3". The OSF FF is set via gate 13 by: Y3. The MLF FF is then reset via gate 25 by: CC9.BT1 and the CC steps to CCO via gate 139.

CCO.MIF — EXCHANGE AR & MR; TOGGLE ASF & MSF IF SIGNS ARE DIFFERENT

In Sigma CCO.MIF the contents of AR (Multiplicand) are exchanged with the contents of MR. AR enters MR via gate 186 and MR enters AR via FASy and AR14in through gates 110, 111, and 173. The FASx gate 99 is blocked in CCO by the false input to gate 97. MR input to itself is blocked by the false signal out of inverter gate 184. As shown in Figure 6-34, at the end of the data pass (DT14), AR contains MR information and MR contains the multiplicand that was in AR. The ASF and MSF FF's are toggled if the signs are different via gates 191, 192,193 and 194. The SCF FF is set via gates 14 and 15; and the OSF FF is reset by: $\overline{BT1}$ being false at the reset input when CCCP1 occurs at BT1 time. The CC steps to CC1 via gate 139 by: MIF.OSF.BT1.

CC1-CC8 — ADD OR SUBTRACT

In this phase (CC1 through CC8), IR and AR are added or subtracted via FAS as previously described (See Logic Analysis for Add or Subtract).

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

However, the contents of IR and AR are not exchanged in CC9 as they would be for a normal Add or Subtract function; therefore, at the end of CC8, IR contains the Product; AR contains the sum of IR and AR (product added to memory); and MR contains the multiplicand number that was in AR. (See Outline of Register Contents in Figure 6-34).

CC9 NO LOGIC ACTION (NLA) 1 BIT TIME

The Conditional Control Phase Counter (CC) steps to CCO via gate 136 and the MIF FF remains set.

CCO.MIF EXCHANGE AR AND MR; TOGGLE ASF & MSF IF SIGNS ARE DIFF; RESET MIF

In this phase (CCO.MIF), the contents of AR and MR are exchanged the same as previously described. Then, if the signs are different, ASF and MSF are toggled via gates: 191,192,193, and 194 as previously described. The MIF FF is reset via the signal $\overline{BT1}.OSF.\overline{CCO}$ being false at the MIF FF reset input.

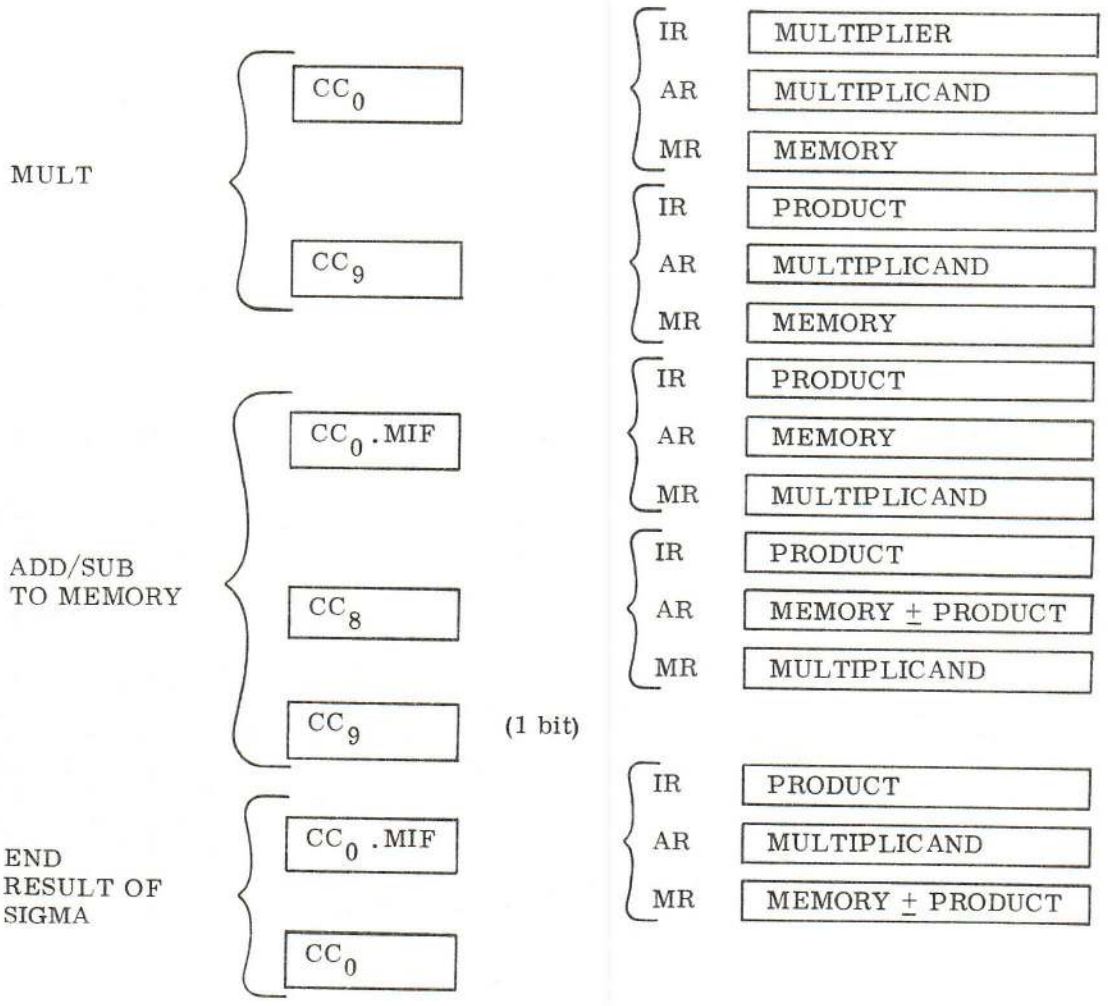


FIGURE 6-34

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

SIGMA; M +; M - (Basic Logic)

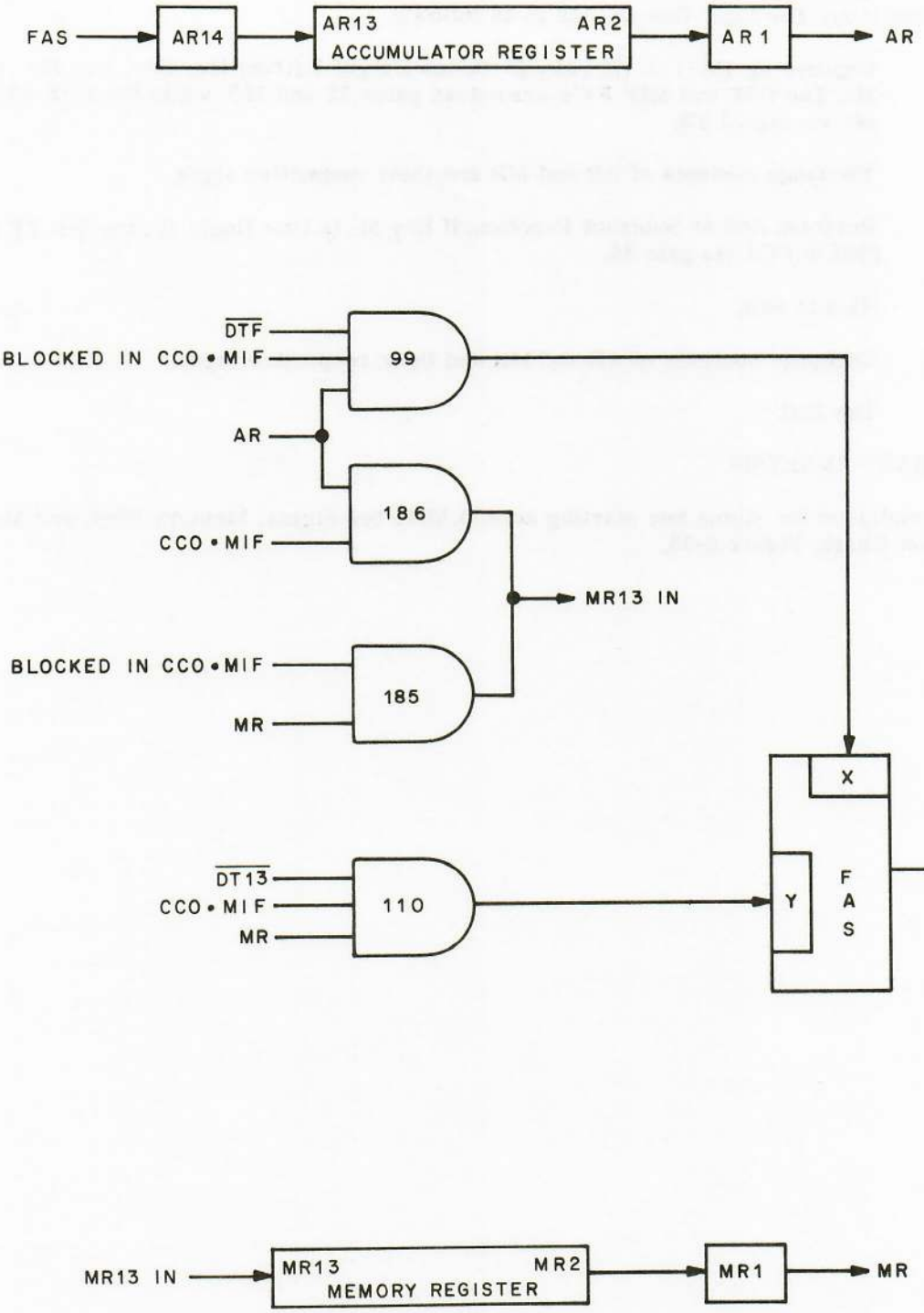


FIGURE 6-35

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

OUTLINE OF LOGIC FLOW IN MEMORY PLUS OR MEMORY MINUS

The logic flow for a Memory operation is identical to that of a Sigma operation except that a mult operation is not required. In other words, the Memory keys can be depressed any time and their respective functions will occcur with each key depression within the limits of the machine capacity. The logic flow outline is as follows:

CCO	Depressing (M+) or (M-) key produces a logic 1 (true) for: Key, Key M+, or Key M-. The OSF and MIF FF's are set via gates 12 and 189, while the POF FF is reset via signal Y3.
CCO.MIF	Exchange contents of AR and MR and their respective signs.
CC1 } CC8 }	Perform Add or Subtract Function. If Key M- is true (logic 1), the ISF FF is toggled in CC1 via gate 30.
CC9	NLA (1 Bit).
CCO.MIF	Exchange contents of AR and MR and their respective signs.
CCO	The End.

FLOW CHART ANALYSIS

Same as explained for Sigma key starting at CCO.MIF. See Sigma, Memory Plus, and Memory Minus Flow Chart, Figure 6-33.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

REVERSE KEY (R)

The function of the Reverse key (R) is to exchange the contents of IR and AR and their respective signs, upon depression of the key.

OUTLINE OF LOGIC FLOW IN REVERSE OPERATION

CCO Depressing Key R causes Key and Key R to be logic 1 (true). Set OSF and KEF FF's. Exchange contents of IR and AR. Toggle ISF and ASF FF's if signs are different. Set SCF and Reset OSF FF's. Release Key R and Reset the KEF FF.

REVERSE KEY FLOW CHART ANALYSIS (FIGURE 6-36)

CCO DEPRESS KEY R

Depressing Key R causes Key and Key R to be at logic 1 (true).

DT14 SET OSF AND KEF FF's

At DT14 time, the OSF FF is set via gate 12 and then $\overline{OSF}$ being false causes the KEF FF to set.

DTO-DT13 EXCHANGE CONTENTS OF IR AND AR

The contents of IR are gated into AR via FASy gate 104 by: JSCF.IR. The output of FAS goes to AR14 via gate 173. The contents of AR are gated into IR via IR14in, gate 156 by: Key R.OSF.AR.

DT14 IF SIGNS ARE DIFFERENT TOGGLE ISF & ASF; SET SCF, RESET OSF.

If signs are different, ISF and ASF are toggled via gate 37. The SCF FF is set via JSCF and gate 15. (NOTE: A digit entry following the Key R function is a valid operation and therefore, the SCF FF must be set by Key R so that a digit entry will enter IR13 instead of IR14 as explained under "Digit Entry". The new digit will then appear in the LSD position of the reversed number in IR.) The OSF FF is reset by: $\overline{BT1}$ being false when CCCP1 occurs.

DTO RESET KEF

Release the Key R and reset the KEF FF at DTO via Key. $\overline{DTO}$ being false.

MEMORY RECALL KEY (MR)

The function of this key is to recall the contents of MR into IR. As MR enters IR it also reenters MR. The original contents of IR are shifted into AR and the AR contents are destroyed. The plus or minus sign of each register is also switched by the necessary toggling of the sign flip flops involved.

OUTLINE OF LOGIC FLOW IN MEMORY RECALL OPERATION

CCO Depressing Key MR causes Key and Key MR to be at logic 1 (true). Set OSF and KEF FF's. Recall contents of MR into IR and shift IR contents into AR. Toggle ISF and ASF FF's if signs are different. Set SCF and reset OSF FF's. IF MSF is set, set ISF. Release Key MR and reset the KEF FF.

KEY MR LOGIC FLOW CHART ANALYSIS (FIGURE 6-36)

CCO DEPRESS KEY MR

Depressing Key MR causes Key and Key MR to be logic 1 (true).

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

REVERSE KEY AND MEMORY RECALL FLOW CHART

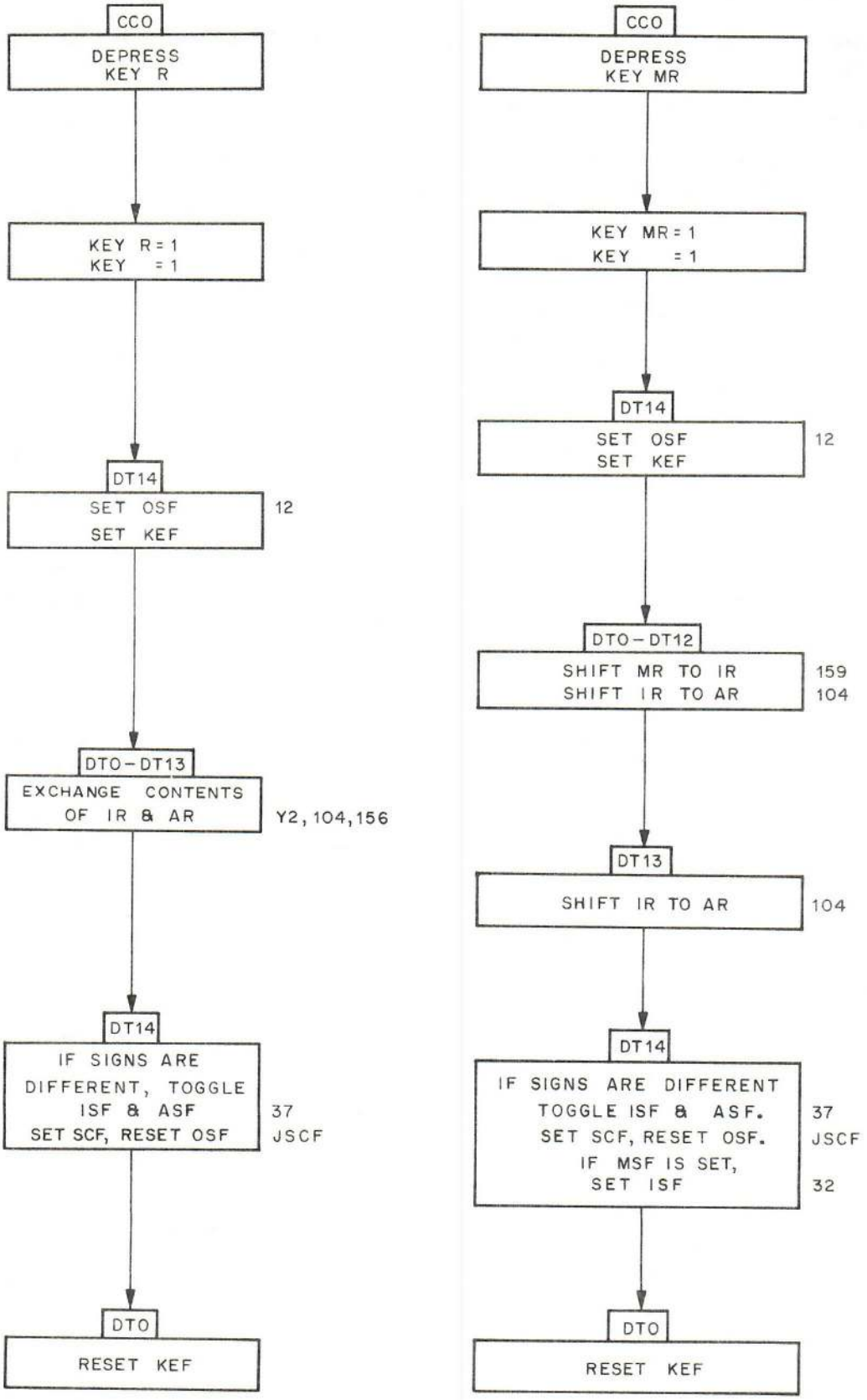


FIGURE 6-36

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

DT14 SET OSF AND KEF FF's

The OSF FF is set via gate 12 and then the KEF FF is set by $\overline{OSF}$ being false.

DTO-DT12 SHIFT MR TO IR; SHIFT IR TO AR

The contents of MR (MR1 through MR13) are shifted into IR via IR14in (gate 159) and MR re-enters itself via gate 185. The contents of IR are shifted into AR via FASy gate 104 and the output of FAS is shifted into AR14in via gate 173.

DT13 SHIFT IR TO AR

In DT13 time, the final IRCP1 and ARCP1 clock pulses occur to complete the data shift of IR and AR.
NOTE: At DT12 time, the MR data pass was complete and therefore, gate 159 must be blocked at DT13 time by $\overline{DT13}$ to prevent MR1 data from entering IR14 a second time.

DT14 IF SIGNS ARE DIFFERENT, TOGGLE ISF AND ASF; SET SCF, RESET OSF; IF MSF IS SET, SET ISF.

If signs are different, toggle ISF and ASF via gate 37. Set the SCF FF via JSCF and gate 15. Reset the OSF FF via $\overline{BT1}$ being false. If the MSF FF is set, then the ISF FF must be set via gate 32.
NOTE: A digit entry following the Key MR function is a valid operation and therefore, the SCF FF will be set by Key MR so that a digit entry will enter IR13 instead of IR14 as explained under "Digit Entry". The new digit will then appear in the LSD position of the recalled number in IR.

DTO RESET KEF FF

Release Key MR and reset KEF FF at DTO time via Key.$\overline{DTO}$ being false.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

RECALL MEMORY (Basic Logic)

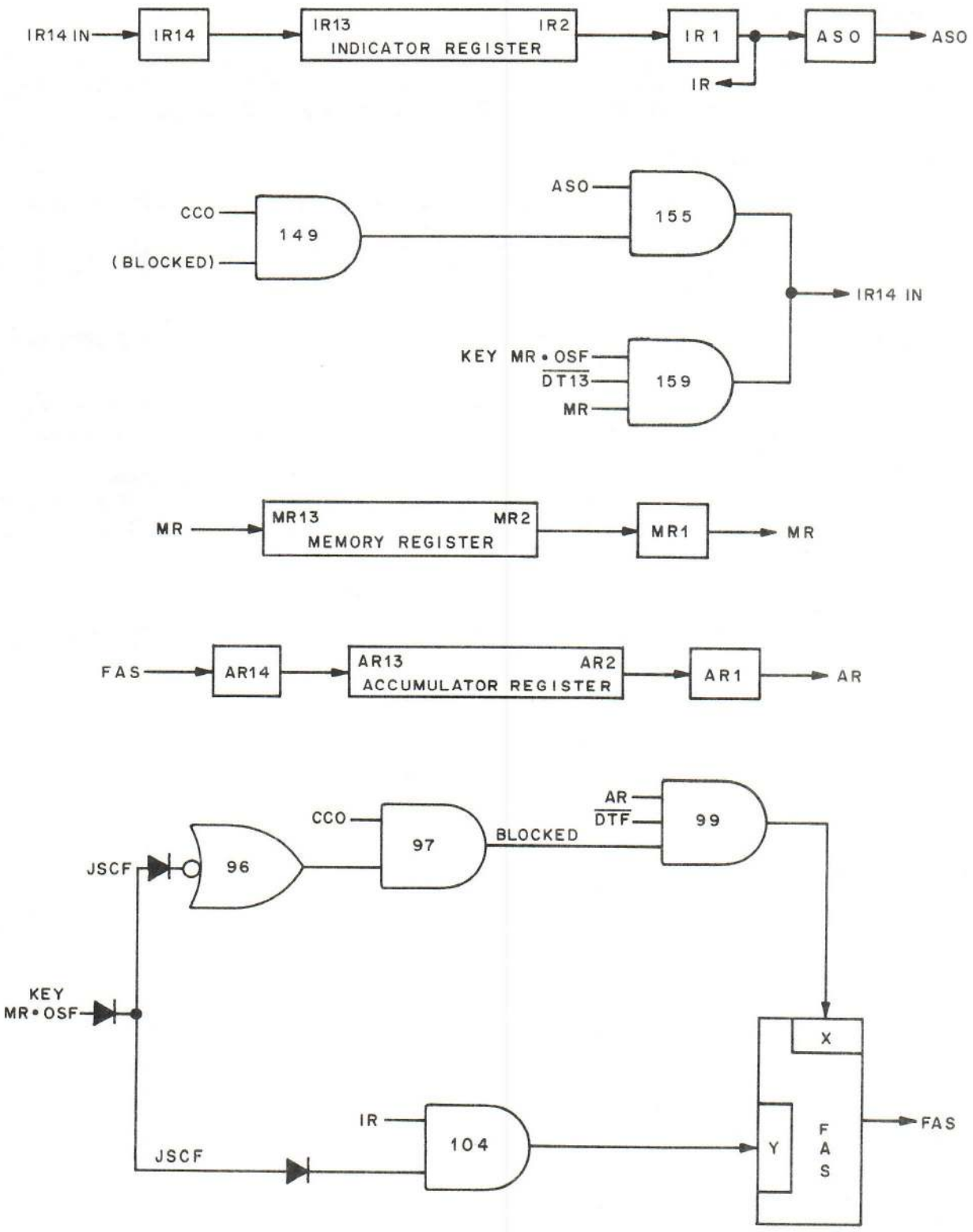


FIGURE 6-37

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

SHIFT KEY (⟶)

The function of the Shift key is to cause the contents of IR (except decimal point information) to shift right one digit position and destroy the previous contents of IR2 (LSD) for each Shift key depression. If the number in IR is a decimal number, the decimal point is also shifted right one place on each IR right shift.

OUTLINE OF LOGIC FLOW IN SHIFT OPERATION

CCO Depressing the Shift key causes Key and Key Sh to be logic 1 (true). Set OSF and KEF FF's. If IR1 is greater than "O", subtract 1 from IR1. Shift IR1 to IR14. Destroy IR2 information (LSD). Shift IR3 through IR14 into IR13. Reset the OSF FF. If IR1=O, reset the POF FF. Release Shift key and reset the KEF FF.

SHIFT KEY FLOW CHART ANALYSIS (FIGURE 6-38)

CCO DEPRESS SHIFT KEY

Depressing the Shift key causes Key and Key SH to be at logic 1 (true).

DT14 SET OSF AND KEF FF's

The OSF FF is set via gate 12 and then the KEF FF is set by $\overline{OSF}$ being false.

IR1=O? YES.

Indicates that the number in IR is not a decimal number and therefore no logic action is required.

IR1=O? NO.

Indicates that the number in IR is not a decimal number and therefore, the $\pm$1F is set via gates 202, 203, and 204 so that a "1" can be subtracted from the decimal information in IR1 at DTO time.

DTO SUBTRACT 1 FROM IR1; IR1 TO IR14

If IR1 is greater than zero, a "1" is subtracted from IR1 in ASO via gate 212. The output from ASO is shifted into IR14 via gates 150 and 155.

DT1 DESTROY IR2 INFORMATION

In DT1 time, the IR2 data is shifted out of IR1 and is destroyed because gate 163 is blocked by: $\overline{DT1}$ being false.

DT2-13 SHIFT IR3 THROUGH IR14 INTO IR13

In DT2 through DT13 time, the contents of IR3 through IR14 are shifted into IR13 via gates 163 and 164.

DT14 RESET OSF; IF IR1=O, RESET POF

Reset the OSF FF by $\overline{BT1}$ being false. If IR1=O, reset the POF FF via gate 66.

DTO RESET KEF

Release the Shift key and reset KEF FF at DTO time via Key.$\overline{DTO}$ being false.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

SHIFT KEY FLOW CHART

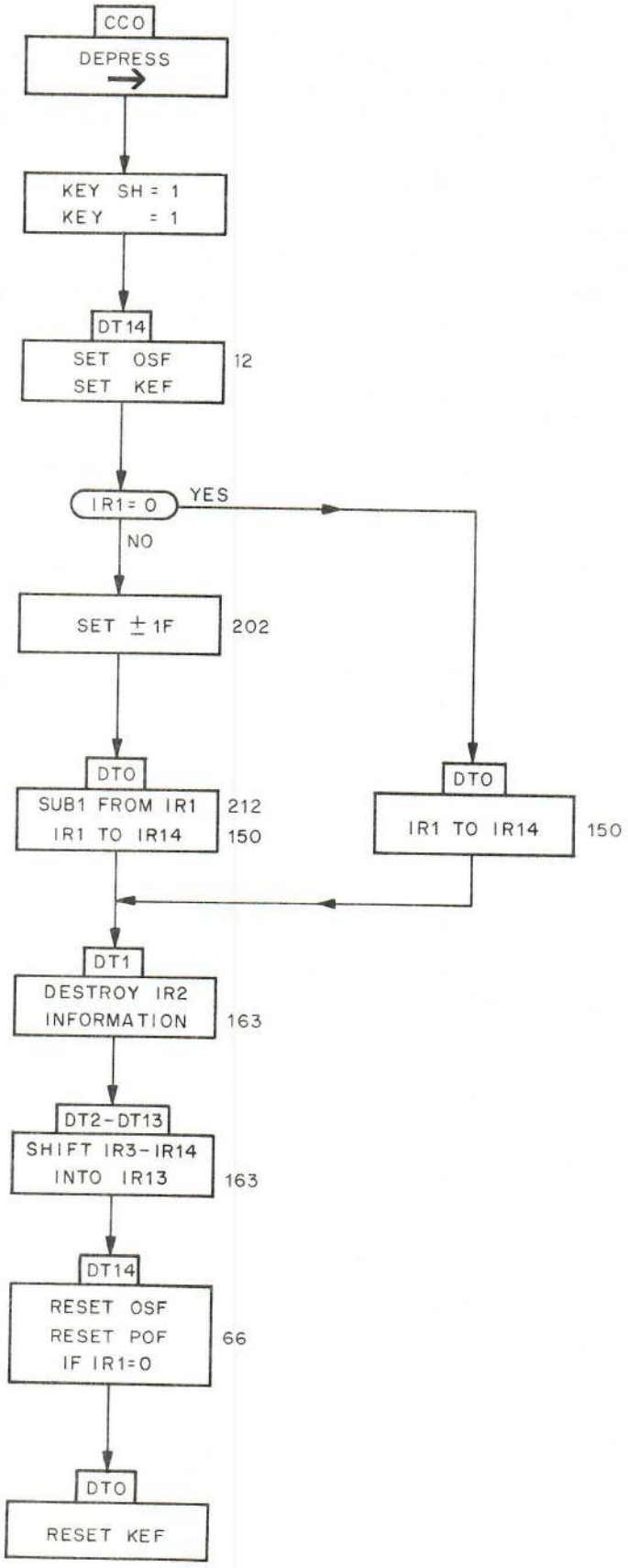


FIGURE 6-38

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

CLEAR KEY (C) AND MEMORY CLEAR KEY (MC)

The Clear key is used for clearing the contents of IR and AR while the Memory Clear key is used only for clearing the contents of MR. Generally, both Clear keys are depressed immediately after turning the power on to insure that all registers are clear, that all flip flops are reset, and that the Conditional Control Phase Counter is at CC0. Then, each Clear key is used separately for the desired clearing action.

OUTLINE OF LOGIC FLOW IN THE CLEAR OPERATION (KEY C)

CC0 Depressing the Clear key causes Key C to be at logic1 (true). The Master Function flip flops and the Overflow flip flop are reset and if the CC is not at CC0, CC will step to CC0. Then, in the first data pass of CC0, IR and AR information is destroyed, and IR is shifted into AR. In the second data pass, the remaining information in AR is destroyed.

CLEAR KEY (C) LOGIC ANALYSIS

Depressing the Clear key causes Key C to be true. Key C being true resets the Master Function FF's: SCF, MLF, DIF, and ISF at DT14 time. Also in DT14 time, the OSF FF is reset by $\overline{BT1}$ being false and the OVF-1 FF is reset by $\overline{DT14}$ being false. The POF and OVF FF's are reset via Key C in the 2nd half of the 1st digit time following the Key C depression, and the KEF FF is reset at DT0 time via Key.$\overline{DT0}$ being false.

Key C.$\overline{CC0}$ via gate 129 causes CC to step 4 times in each DT14 time until it reaches CC0. This will require a maximum of 3 column times because CCCP1 occurs 4 times during each DT14 time.

CC0 When the CC reaches CC0, the RDF and BCF FF's are reset by $\overline{CC0}$ and clock pulse CFCP1. The MIF FF is reset in CC0 at DT14 time via $\overline{BT1}$.OSF.$\overline{CC0}$ being false.

DT0-13 During the first data pass (DT0 through DT13), the contents of IR are shifted into ASO and FAS simultaneously. IR contents are destroyed during this data pass because IR14in is blocked via inverter gate 148. IR input to AR is via FASy gate 104 and 173. The AR input to FASx is blocked by gates 96, 97, and 99 which causes the AR contents to be destroyed. At the end of this first data pass, IR is clear but AR now contains the original IR contents. In the next data pass, the AR input to FASx is blocked as before, which causes the remaining AR contents to be destroyed. At the end of the 2nd data pass of CC0, all clearing of IR and AR is complete.

MEMORY CLEAR KEY(MC) LOGIC ANALYSIS

Depressing the Key MC causes Key MC to go true. At DT14 time, Key MC being true resets the MSF FF via gate 196. During the data pass (DT0 through DT12), the input to MR is blocked via gates 184 and 185 which destroys the contents of MR. At the end of the data pass, only the register MR is clear.

OVERFLOW

The Overflow Lamp will turn on to indicate too large an entry or a wrong answer whenever the arithmetic capacity of the machine is exceeded. The manner in which an overflow (over capacity) can occur varies depending upon the arithmetic operation being performed, or on the number of digits in the entry. Therefore, the overflow logic system must be capable of detecting every possible overflow condition. The basic conditions and logic analysis for causing an overflow are as follows:

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

OVERFLOW LOGIC ANALYSIS

1. In CC0, an entry of 11 digits is the maximum that can be entered after the decimal point. If this is exceeded, the OVF FF will set at DT14 via gates 218, 219, and 220.

2. In CC0, an entry of 13 or more digits will force a digit into IR14 causing the signal CC0.$\overline{IR14=0}$.DT14 to be true at DT14 time. The OVF FF will set via gates 219 and 220.

3. In CC1, mult or divide, the contents in IR13 and AR13 will appear in IR14 and AR14 at DT12 time and should be "0". If they are not "0", it means that one of the factors in the mult or divide problem contains 12 digits which will exceed the capacity of the machine and cause a wrong answer. Then, the OVF FF is set via gate 215 by: CC1.$\overline{IR14=0}$+$\overline{AR14=0}$.(MLF+DIF).DT12.

4. In CC2, the contents of AR are shifted left until decimally aligned with the contents of IR at which time AR14 should be "0". If AR14 is not "0", it indicates that the number in AR is too large to decimally align with IR and the OVF FF will set at DT14 time via gates 217 and 220 by: CC2.$\overline{AR14=0}$.DT14.

5. In CC2, Divide, if the POF FF is set, AR is shifted left until it is decimally aligned with IR. If $\overline{AR14=0}$.DT12 becomes true as AR is shifted left, it means that the dividend cannot be decimally aligned with the divisor in IR. Then, the OVF FF will set via gate 215 by: CC2.$\overline{AR14=0}$.DIF.DT12.

6. In CC3, Mult, the decimal information in IR1 is added to the decimal information in AR1 and the result is the decimal point position of the product. If the sum of IR1 plus AR1 is between 12 and 15, then after the mult is completed, the OVF FF will set at CC0 and DT14 via gates 218, 219, and 220. However, if the result in CC3 of IR1 plus AR1 is 16 or greater, the CAF FF will be set at BT1.DTF of DT0 time and the OVF FF will set via gate 213 to indicate an excess of 11 decimal positions.

7. In CC4, Add or Subtract, if AR1 is not "0", IR is shifted left until it is decimally aligned with AR. However, if the number in IR cannot be decimally aligned with the number in AR and overflows into IR14, the OVF FF will set at DT14 via gates 216 and 220.

8. In CC5, Mult, the boundary code 15 is sensed when it reaches IR14 which causes IR14=15 to be true. This will always set the OVF-1 FF via gate 223 in order to sense the remaining data in AR. Then in the remaining digit times following the sensing of the boundary code, if AR14 does not become "0" ($\overline{AR14=0}$), it indicates that the sum of the multiplier and multiplicand exceeds 12 digits and the OVF FF will be set via gate 214 because the signal OVF-1.$\overline{AR14=0}$ is true.

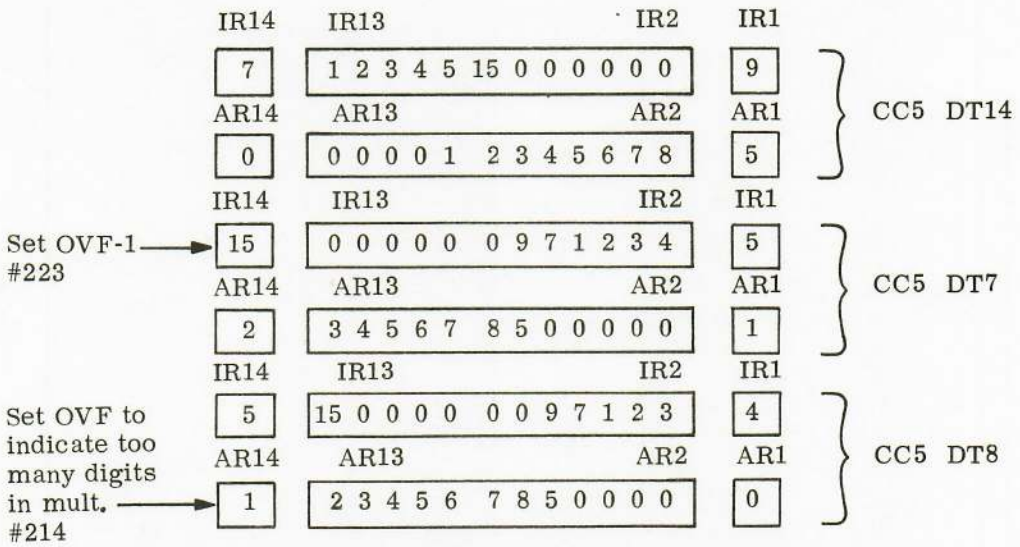
   NOTE: An example of the CC5 mult overflow condition is shown in Figure 6-39. In this example, the two numbers (multiplier and multiplicand) are in their respective IR and AR register positions as indicated at DT14 time. Then, at DT7 time of the data pass, the Boundary Code 15 has arrived in IR14 where it is sensed as IR14=15 and the OVF-1 FF is set via gate 223. Once the OVF-1 FF is set, all remaining data flow through AR14 is monitored to detect a digit other than "0". If a number other than "0" is detected, it indicates that the sum of the digits in the multiplier and multiplicand is greater than 12 and therefore exceeds the capacity of the machine. Referring to the sample problem, a "1" is detected in AR14 at DT8 time which causes the OVF FF to be set via gate 214.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

9. In CC8, Add, Subtract, Memory+, Memory -, or Sigma, the sum or difference of AR and IR is in AR and if AR14 is not "O" at DT14 time, the answer contains too many digits. Therefore, the OVF FF will set via gates 217 and 220 by: CC8.$\overline{AR14=O}$.DT14.

Multiplicand (123.45) x Multiplier (12345.678) contains 13 digits which will cause an OVERFLOW.

123.45 x 12345.678 = OVERFLOW (13 Digits)



(AR14 IS NOT ZERO AFTER OVF-1 SETS)

FIGURE 6-39

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

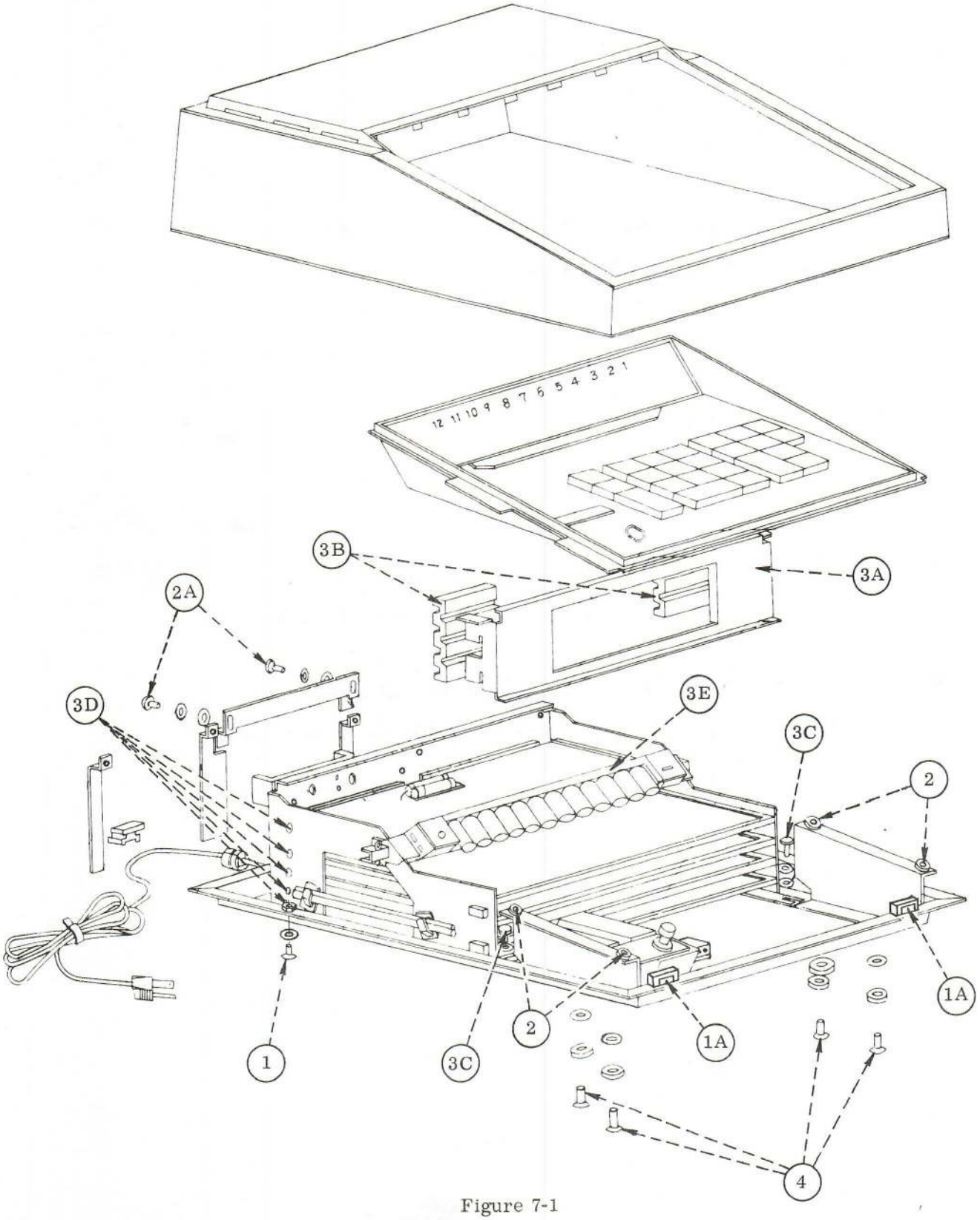


Figure 7-1

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

DISASSEMBLY AND ASSEMBLY

Instructions for disassembly and assembly of the component parts of the 1113 are listed below. Each instruction is keyed to Figure 7-1.

1. Remove the cover by removing the screw (1) on each side of the base at the rear and lifting the back of the cover up so the locking tabs on the cover are released from their respective slots in the base (1A).

2. Remove the keyboard assembly by lifting it up from the rubber bushings (2) in the keyboard supports. The connector on PC board #5 must also be removed by loosening the two screws (2A) and raising the clamp. It is not necessary to remove the screws.

3. To remove the PC boards, proceed as follows:

   a. Lift the front panel (3A) about half way up and then toward the front.

   b. Remove the two rubber separators (3B).

   c. Remove the two screws (3C) which secures the PC card holder frame to the base.

   d. Remove the desired PC board by inserting a flat blade screwdriver into the holes (3D) on either side of the card holder frame and pry gently forward from each side.

   e. To remove PC board #4 on which the number indicating tubes are located, it is first necessary to remove the lamp holder (3E) by removing the cotter pin at the end of the holder and pulling the support rail out the side. Then, carefully lift the lamp holder up and away from the number indicating tubes being careful not to break the wires at the end of the holder. When reassembling the #4 PC board be sure that the lamp holder securely engages the lamps.

   f. To remove PC board #5 (Power Supply Board), the #4 PC board must first be removed as explained in 3e.

4. To reassemble the components, reverse the order of disassembly listed above. Also, the following precautions should be observed.

   a. Use the special flat washer (Part #KTA 12398) on the screws (3C), Figure 7-1, to secure the PC card holder to the base. Conventional washers cannot secure the card holder properly.

   b. To align the keyboard supports to the keyboard, loosen the screws (4), Figure 7-1, and move the supports. Be sure to use the special nylon washers Part #KCA 22026, and special flat washers Part #KTA 12398.

   c. Prior to securing the PC card holder to the base, check to see that the PC boards are in proper position. Then, insert the rubber separators and secure the front panel of the PC card holder.

   d. When installing the top cover on the machine, check to see that all wires and cables are in their respective holders. A wire or cable pinched between the base and the cover could result in a short circuit which could damage the components on the PC boards.

INSPECTION PROCEDURE

1. Perform a complete operational check using the test problems starting on page 7-2. NOTE: Check to see that all numbers in the indicating tubes are operating properly.

2. Visually inspect the machine for worn or loose parts such as: Keystems, Keytops, AC Power Switch, Decimal Cursor, etc.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

3. Check the -24 volt level on test point A (PC card #5) at the negative side of capacitor C7. It should read -24 volts ±1 volt. If not refer to the Power Supply adjustment procedure, page 7-6.

4. Clean and Lubricate keyboard unit as necessary.

   a. Blow out all dust and foreign material.

   b. Lubricate the keystems with grease. (Shell non-stain 71212/Friden MS-136.)

5. Check the adjustment of all reed switches on the keyboard.

   a. Each reed switch should close when its keystem has been depressed approximately half way down. See Reed Switch adjustments, page 7-6, for adjustment procedure.

6. Clean the exterior of the machine with soap and water. Avoid any commercial cleaner that is not recommended for use with plastic.

OPERATIONAL CHECK

The procedures for accomplishing a complete operational check of the 1113 are listed below. The entire sequence should be followed after replacing a PC board or after completion of a repair.

1. Checking the registers and associated circuits.

   a. Entry and Indicator Register "IR" check.

      Index C - 12345678901200000000000

      Note: The decimal point is indicated at random when more than 12 digits have been entered. This does not indicate a machine malfunction. The overflow lamp will light on the 13th digit entry.

   b. Check of reverse key and register "AR".

      Index C - 123.456789012 R R

      Indicator will return to its initial condition.

   c. Check of M+ M- RM keys and Register "MR".

      Index: C CM 123.456789012 M+ C RM

      Index: C CM 123.456789012 M- C RM

   d. → Key (shift function)

      Index: C 123.456789012 → → . . . →

      Indication will shift one digit to the right at each depression of the shift key.

2. Operation Check.

   a. Addition, Subtraction and Decimal Alignment check

      Index: C 123 -= 0.123 -= 0.000123 -= 123.123123 -= 246.246246 -= 492.492492 -= 0.123 += 0.123 += 0.123 += 0.123 += 0.246 += 0.246 +=

      Result: -984.000984

      Index: C 123 += 0.123 += 0.000123 += 123.123123 += 246.246246 += 492.492492 += 0.123 -= 0.123 -= 0.123 -= 0.123 -= 0.246 -= 0.246 -=

      Result: 984.000984

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

b. Subtraction and Decimal Alignment check

Index: C 0.000123 -= 0.123 -= 123 -=

Result: -123.123123

c. Addition and Decimal Alignment check

Index: C 0.000123 += 0.123 += 123 +=

Result: 123.123123

d. Multiplication and constant multiplicand check

Index: C 0.123 x 123 += Result: 15.129

Index: R x 0.123 += Result: 0.015129

Index: C 123 x 0.123 += Result: 15.129

Index: R x 123 += Result: 15129

e. Division and constant divisor check

Index: C 1 ÷ 3 += Result: 0.3333333333

R 1 R += Result: 0.3333333333

f. Division Check

Index: C 123456789 ÷ 1 += Result: 123456789.00

g. Multiplication Check

Index: C 1 x 123456789 += Result: 123456789

h. Check of accumulation of positive and negative products (Sigma Σ)

Index: C CM Σ 0.123 x 123 -= 123 x 0.123 -= 1.23 x 12.3 -= 123 x 0.123 += 0.123 x 123 += 12.3 x 1.23 += RM

Result: 0.000

Index: C CM 9.00000 x 9.00000 -= 7 x 8 += RM

Result: -25.0000000000

Index: C CM 7 x 8 -= 9.00000 x 9.00000 += RM

Result: 25.0000000000

Index: C CM 123.123 M+ RM Result: 123.123

C CM 123.123 M- RM Result: -123.123

i. Check of memory calculations (Release Σ key)

Index: C CM 123 M+ 0.123 M+ 0.000123 M+ 123.123123 M+ 246.246246 M+ 492.492492 M+ 0.123 M- 0.123 M- 0.123 M- 0.123 M- 0.246 M- 0.246 M- RM

Result: 984.000984

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

j. Special Operational Checks

Index: C

(1) 123 x ÷ 3 += Result: 41.00000000

(2) 123 ÷ x 3 += Result: 369.

(3) 123 ÷ += 3 += Result: 41.00000000

(4) 123 ÷ -= 3 += Result: 41.00000000

(5) 123 x ⩾ 3 += Result: 36.

(6) 123 ÷ ⩾ 3 += Result: 4.000000000

TROUBLESHOOTING

Prior to attempting to repair or service the 1113 Electronic Calculator, be sufficiently well informed concerning the operating conditions in the customer's office such as: Excessive high or low temperature (operating temperature 32° F. to 104° F.); proper line voltage (115 volts ±10 volts, 50/60 Hz.); excessively high humidity; and that correct operating procedures are being observed.

Locate and repair obvious problems such as: blown fuses, sticking keys, etc. Any complicated condition which requires a relatively long service time should be accomplished by exchanging subassemblies such as logic cards and keyboard. These can then be returned to the branch office for repair, or sent to the Service Center in your area for repair.

PC Board Replacement

If the PC boards are suspected of causing trouble, special precautions must be observed while replacing or troubleshooting the boards due to the extreme sensitivity of the MOS IC circuits. Replace only one board at a time while observing the following precautions.

1. Turn the Power Switch "OFF" before replacing a PC board. If a board is removed or installed with the power on, possible damage to the components such as IC's, transistors and diodes may occur.

2. Avoid touching the IC's with your bare fingers. Due to their electrical characteristics, the MOS IC's used in the 1113 have an extremely high input impedance and are very sensitive to static charges (especially when the PC boards are removed from the machine). Handle the boards only by the metal side frames.

3. When the PC boards are not in use or while transporting or shipping individual boards, be sure to short circuit the terminals with a protective connector. (Two each, Part #KCA 22035 with all pins soldered together.) Protective connectors with short circuited pins are used to maintain all PC board contacts at an equal potential so that no potential difference exists between the IC leads. However, not all the IC leads are connected to the PC contacts and care must be taken to prevent touching the IC leads with the fingers.

4. Avoid damaging the components mounted on the PC boards. Damage to the components can occur or their insulation may be lowered if the PC boards are allowed to contact each other, or are stacked one on top of the other.

5. Avoid external force or shock to the PC boards.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

6. Do not insert a PC board into a connector for another board. Also, since the connecting terminals of the PC boards are arranged symmetrically, it is possible to insert a board upside down. Therefore, check to see that the component side of the board is facing up, prior to inserting it. The proper arrangement of the PC boards in the machine is 1 through 5 with the #1 board at the bottom.

7. Before connecting any test leads, or using a soldering iron, the logic ground must be connected to the chassis ground. This can be done by connecting a test clip lead wire between the positive (+) side of C13 (470 mfd electrolytic capacitor on card #5) and chassis ground or "Scope" ground.

8. Two #4 PC boards are in use. One is equipped with the old type number indicating tube designated CD71 and the other is equipped with the new, brighter, number indicating tube designated CD79. These two types of tubes are not to be interchanged under any circumstances. However, the complete #4 PC board with either type of tube may be interchanged from machine to machine.

9. Perform the operational check after each replacement of a PC board.

   The basic card functions are listed to help diagnose troubles.

   1. Card #1 - Major timing control, Indicator Register Logic.
   2. Card #2 - Master Control Functions.
   3. Card #3 - Arithmetic Unit, Accumulator and Memory Register Logic.
   4. Card #4 - Display decoding and Number Indicating Tubes.
   5. Card #5 - Entry Functions, Master Oscillator, and Power Supply.

NOTE: Card extender, tool #18770, designed for use on the 1150 Printing Calculator can be modified for use on the 1113 Calculator by removing the card extractor on each side of the card. When using the modified extender on an 1113 PC card, be sure to insulate the exposed terminals with electrical tape or equivalent insulating material. To keep the PC cards properly spaced within their guides, use a heavy rubber band around the sides of the logic card holder to keep it from separating.

Replacement of Components on PC Boards

1. Two #4 PC boards are in use. One is equipped with the old type number indicating tubes designated CD71 and the other is equipped with the new brighter number indicating tubes designated CD79. These two types of tubes are not to be interchanged under any circumstances. However, the complete #4 PC board with either type of tube may be interchanged from machine to machine.

2. On #4 PC boards using the old type CD71 number indicating tubes, the transistors with a black dot on top are NOT to be used in the number cathode circuits (transistors T13 through T22). It has been found that double numbers may occur when the black dot transistors are used in these circuits.

3. When replacing broken or burned resistors, it may be necessary to use insulating tubing on the leads of the stand up type resistors to prevent shorting with adjacent resistors.

4. When replacing IC's, be sure to align the locating tab on the IC with the printed circle on the PC board as shown in Figure 7-5.
   Caution: Do not overheat the IC's when soldering as it may damage them.

5. Transistors used in the 1113 have the normal collector, base, emitter leads, but unlike the transistors we are accustomed to, the 1113 transistors have a mark (see Figure 7-6) to identify the collector rather than the emitter. The PC board is also marked with a "C" to identify the connecting point for the collector.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

ADJUSTMENTS

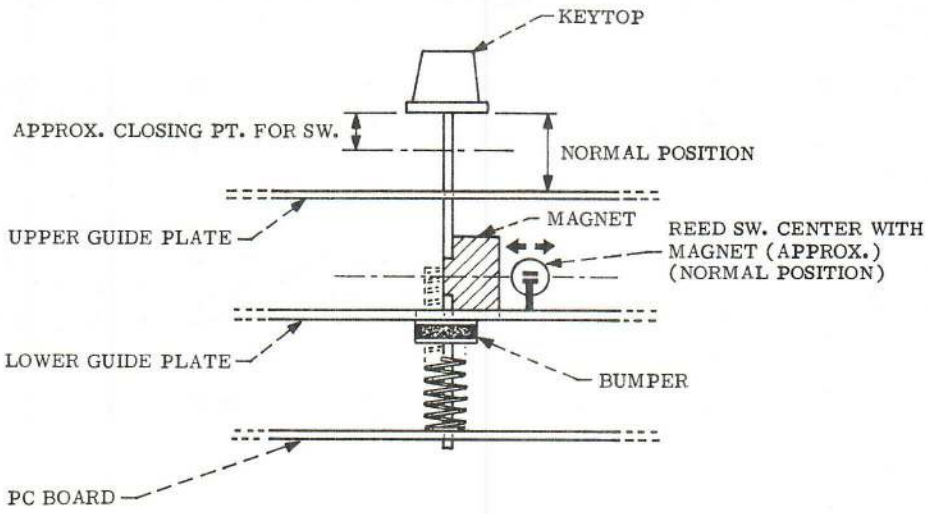


FIGURE 7-2

1. Keyboard Reed Switches: The keyboard reed switches must close when their respective keystems have been depressed approximately half way down. To adjust, move the individual reed switches closer or farther away from their respective magnets as shown in Figure 7-2.
(NOTE: Moving the switches closer to the magnets causes them to close sooner. Moving them farther away from the magnets causes them to close later.)

2. Power Supply: The output from test point "A" on PC board #5 (negative side of capacitor C7) must read -24 volts ±1 volt. To adjust, turn the potentiometer VR-1. After adjusting the -24 volts, check the -14 volt level on test point "D" on PC board #5 (negative side of capacitor C10). It should be -14 volts ±1 volt. Then check test points A, B, C, and D on PC board #4 for: -14 volts, -11 volts, -5 volts, and "0" volts.

   NOTE: Secure the potentiometer setting by applying a suitable adhesive such as locktite, glyptal, nail polish, etc. to the outer rim of the potentiometer.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

COMPONENT IDENTIFICATION AND LAYOUT

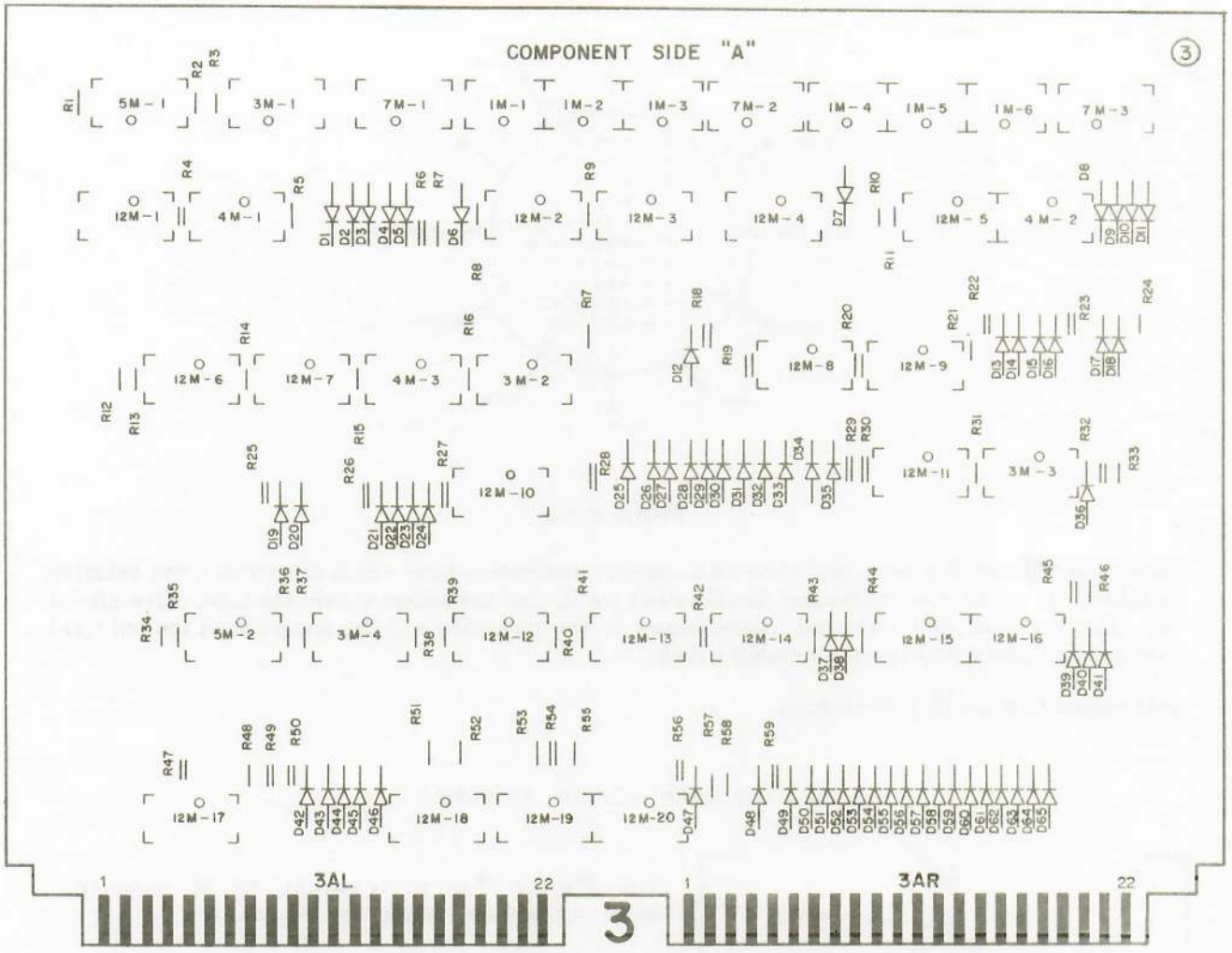


FIGURE 7-3

Printed Circuit Board

For identifying and locating components or terminal pins on the printed circuit boards, the board should be viewed as shown in Figure 7-3. Each component is identified and numbered on the board from left to right starting at the left hand top of the board.

Resistors: Indicated as R1, R2, R3, etc.

Diodes: Indicated on PC board and numbered from left to right.

Capacitors: Indicated on PC board as C1, C2, C3, etc.

IC Chips: Indicated by rectangular outline on board and designated by type (HD705 as 5M, HD703 as 3M, etc.). A number following the type indicates the successive quantity of that particular type. For example, IC chip HD712M as used on PC board #3 in Figure 7-3, starts as 12M-1 at the left end of row 2 and each succeeding use of HD712M on this card is incremented by 1 as 12M-2, 12M-3, through 12M-20.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

Number Indicating Tubes

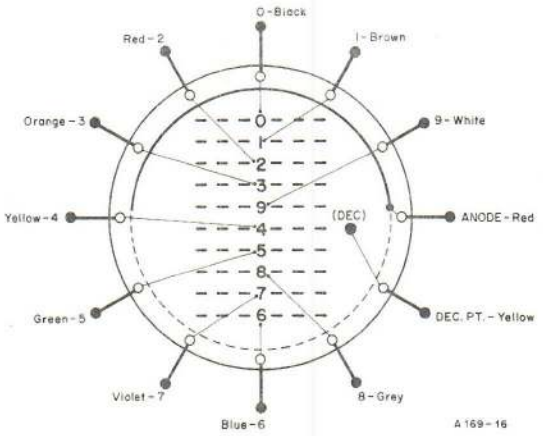


FIGURE 7-4

Shown in Figure 7-4 is a top view of a number indicating tube which illustrates the relative position of the number cathodes, the decimal point, and the anode inside the tube. Also shown are the internal and external connections to the cathodes and the anode, with the external connections identified by color coded wires.

Integrated Circuit (IC) Terminals

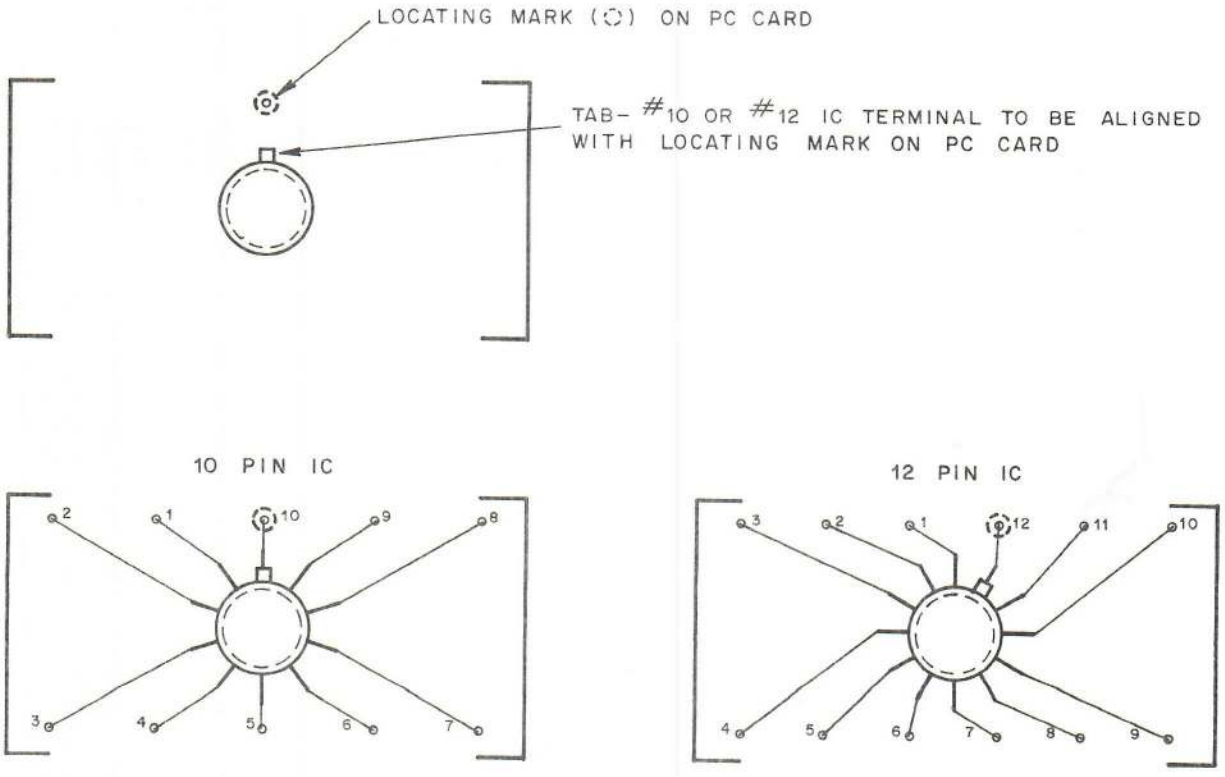


FIGURE 7-5

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

In Figure 7-5 is shown a 10-lead and a 12-lead IC module on their respective PC board positions. Note that the locating tab on the IC identifies either the #10 lead wire or the #12 lead wire, and when the IC is mounted on the PC board, the tab is aligned with the small circle printed on the PC board.

Transistors



FIGURE 7-6

As shown in Figure 7-6, the symbol on the transistors used in the 1113 identifies the collector lead rather than the emitter lead as on most other types of transistors.

NOTE: Two transistors used in the 1113 are the same type (2SC284), but because of differences in their electrical characteristics, they have different part numbers (KCA 22255 and KTA 12342) with KCA 22255 identified with a black dot on top. The two transistors are interchangeable except that the KCA 22255 (black dot) cannot be used in the number cathode circuits on PC board #4 (transistors T13 through T22) which uses the old type number indicating tubes designated CD71.

Diodes

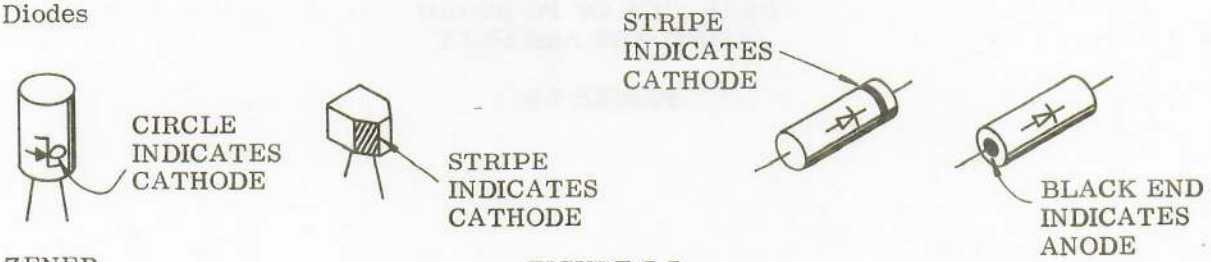
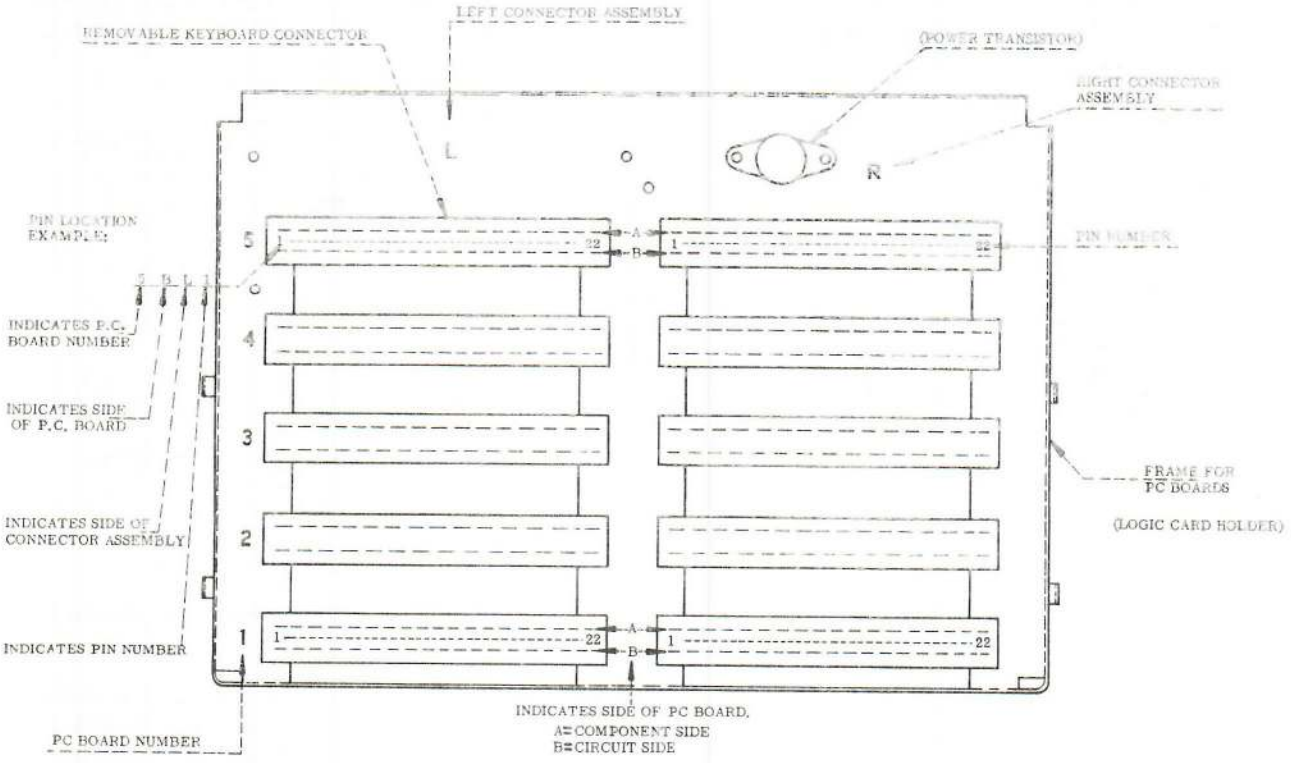


FIGURE 7-7

As shown in Figure 7-7, a small circle is used to indicate the cathode lead on the zener diode; a colored stripe is used to indicate the cathode on two of the diodes; and a black end on the other diode indicates the anode lead.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

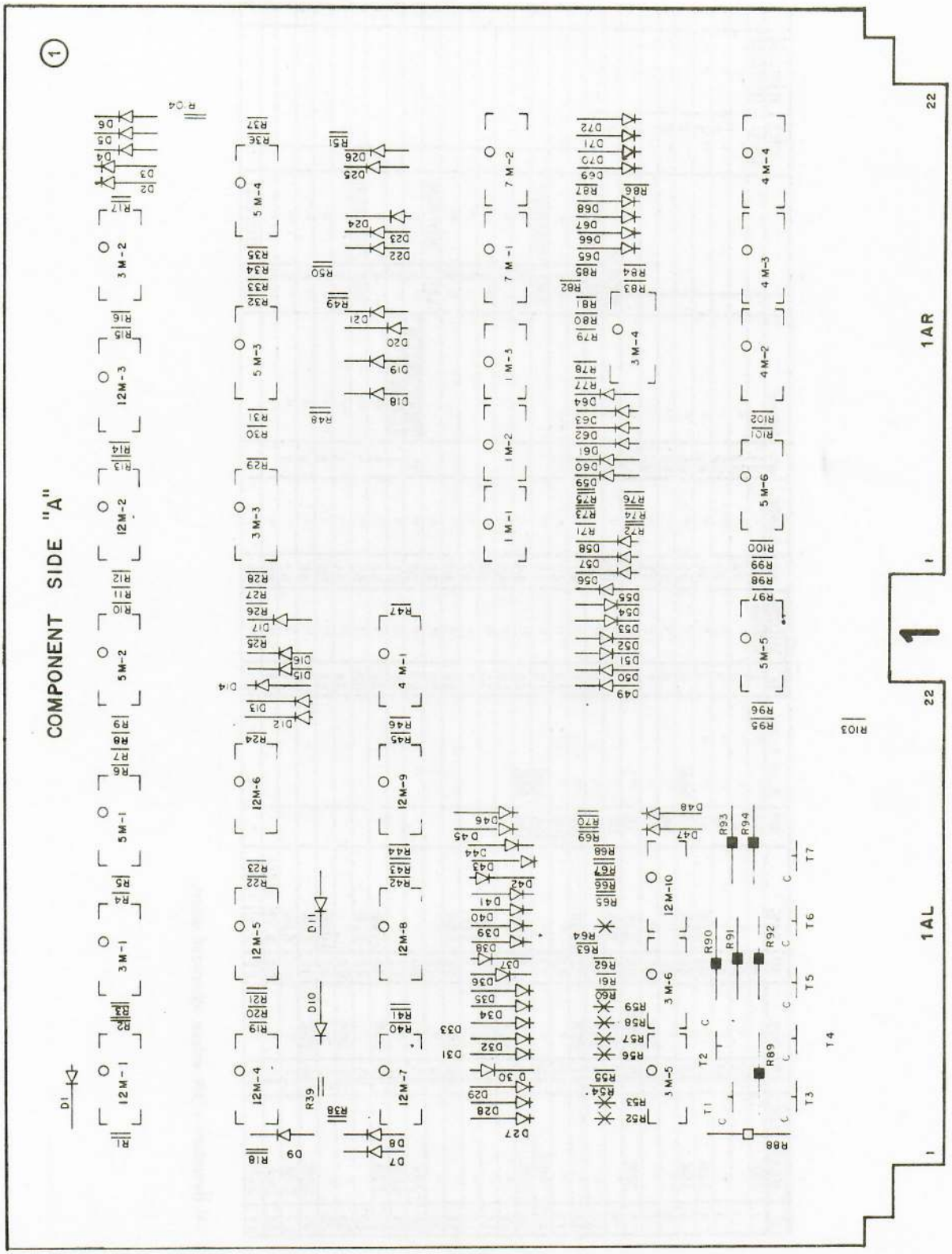


REAR VIEW OF PC BOARD
CONNECTOR ASSEMBLY

FIGURE 7-8

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

P. C. BOARD #1



Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

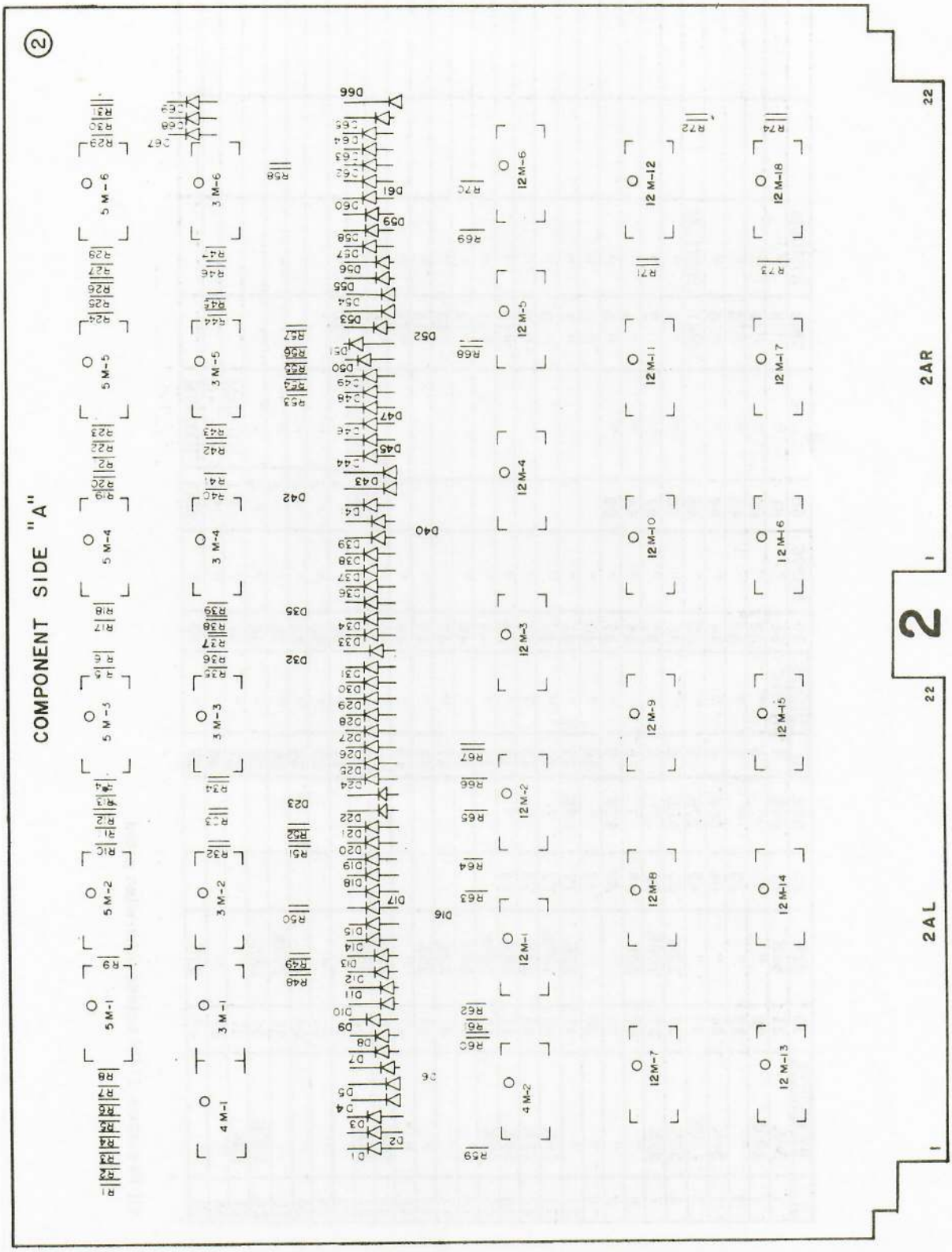
COMPONENT IDENTIFICATION P. C. BOARD #1

R	RESISTOR	30	22K	60	56K	90	3.3K 1/2W	D	DIODE	30	HS996	60	HS996			7M2	HD707M
1	56K	31	"	61	"	91	" "	1	HS996	31	"	61	"			12M1	HD712M
2	"	32	"	62	"	92	" "	2	"	32	"	62	"			" 2	"
3	"	33	"	63	"	93	" "	3	"	33	"	63	"			" 3	"
4	22K	34	"	64	33K	94	" "	4	"	34	"	64	"			" 4	"
5	56K	35	"	65	56K	95	56K	5	"	35	"	65	"			" 5	"
6	"	36	"	66	"	96	"	6	"	36	"	66	"			" 6	"
7	"	37	22K	67	"	97	"	7	"	37	"	67	"			" 7	"
8	22K	38	100K	68	22K	98	"	8	"	38	"	68	"			" 8	"
9	"	39	56K	69	56K	99	"	9	"	39	"	69	"	MOS IC		" 9	"
10	"	40	22K	70	"	100	"	10	"	40	"	70	"	1M1	HD701M	"10	"
11	"	41	56K	71	100K	101	"	11	"	41	"	71	"	" 2	"		
12	"	42	"	72	"	102	"	12	"	42	"	72	"	" 3	"		
13	100K	43	"	73	"	103	22K	13	"	43	"			3M1	HD703M		
14	56K	44	"	74	56K	104	56K	14	"	44	"			" 2	"		
15	"	45	22K	75	"			15	"	45	"			" 3	"		
16	"	46	56K	76	100K			16	"	46	"			" 4	"		
17	"	47	22K	77	56K			17	"	47	"			" 5	"		
18	"	48	100K	78	22K			18	"	48	"			" 6	"		
19	"	49	"	79	"			19	"	49	"			4M1	HD704M		
20	22K	50	"	80	"			20	"	50	"	T	TRANSISTOR	" 2	"		
21	56K	51	"	81	"			21	"	51	"	1	2SA549 K	" 3	"		
22	22K	52	33K	82	100K			22	"	52	"	2	"	" 4	"		
23	"	53	"	83	22K			23	"	53	"	3	"	5M1	HD705M		
24	"	54	56K	84	"			24	"	54	"	4	"	" 2	"		
25	"	55	"	85	100K			25	"	55	"	5	"	" 3	"		
26	56K	56	33K	86	22K			26	"	56	"	6	"	" 4	"		
27	22K	57	"	87	56K			27	"	57	"	7	"	" 5	"		
28	56K	58	"	88	2.2K 1/2W			28	"	58	"			" 6	"		
29	22K	59	"	89	3.3K 1/2W			29	"	59	"			7M1	HD707M		

All Resistors 1/4W unless otherwise noted.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

P. C. BOARD #2



Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

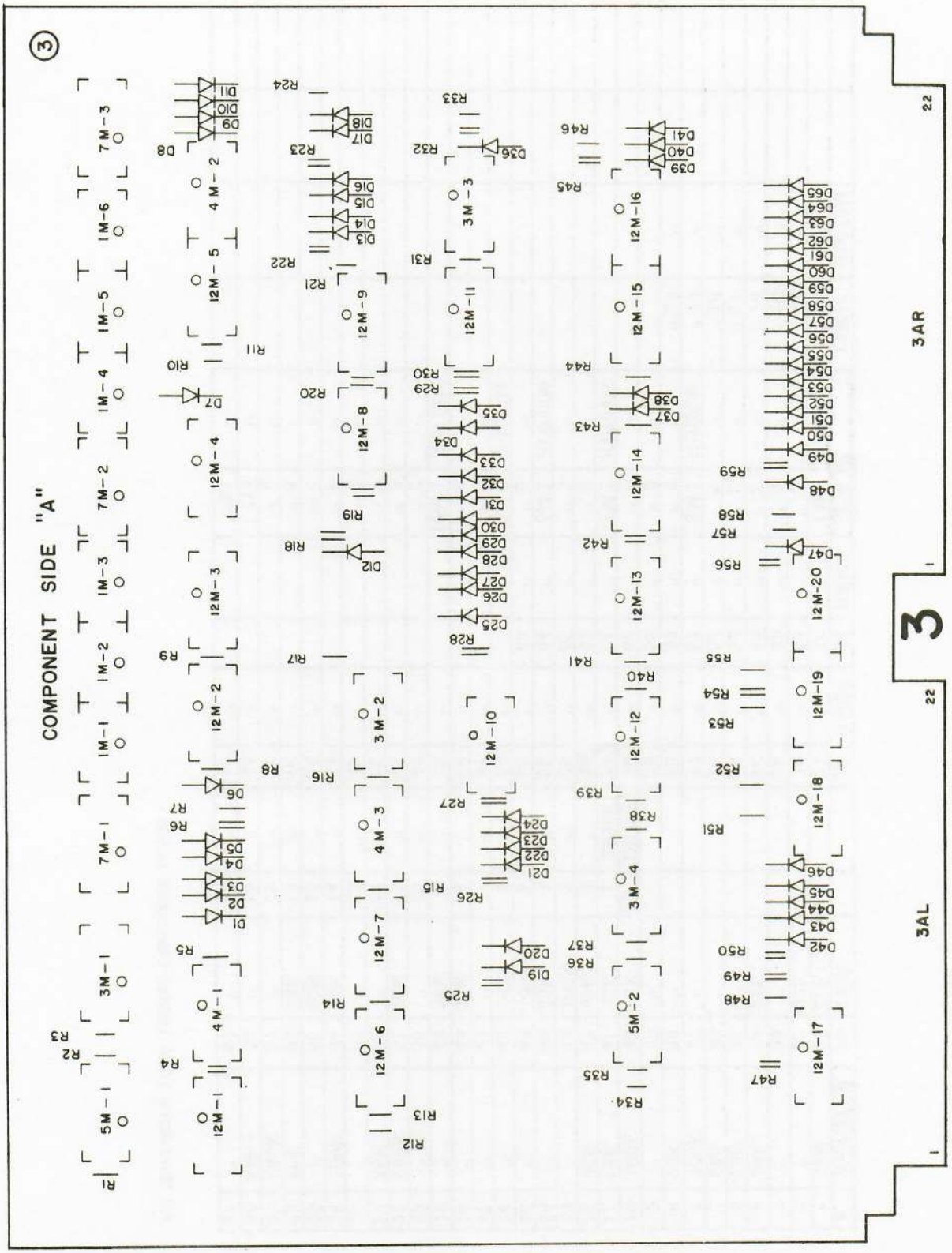
COMPONENT IDENTIFICATION P. C. BOARD #2

R	RESISTOR	30	22K	60	56K	D	DIODE	30	HS996	60	HS996	5M2	HD705M		
1	56K	31	56K	61	22K	1	HS996	31	"	61	"	" 3	"		
2	100K	32	"	62	"	2	"	32	"	62	"	" 4	"		
3	"	33	"	63	"	3	"	33	"	63	"	" 5	"		
4	56K	34	"	64	56K	4	"	34	"	64	"	" 6	"		
5	100K	35	"	65	22K	5	"	35	"	65	"	12M1	HD712M		
6	56K	36	22K	66	56K	6	"	36	"	66	"	" 2	"		
7	22K	37	100K	67	"	7	"	37	"	67	"	" 3	"		
8	"	38	56K	68	"	8	"	38	"	68	"	" 4	"		
9	"	39	"	69	22K	9	"	39	"	69	"	" 5	"		
10	"	40	"	70	"	10	"	40	"			" 6	"		
11	"	41	22K	71	56K	11	"	41	"			" 7	"		
12	56K	42	100K	72	"	12	"	42	"			" 8	"		
13	22K	43	22K	73	"	13	"	43	"			" 9	"		
14	56K	44	56K	74	"	14	"	44	"			"10	"		
15	22K	45	100K			15	"	45	"			"11	"		
16	"	46	56K			16	"	46	"			"12	"		
17	"	47	22K			17	"	47	"			"13	"		
18	"	48	56K			18	"	48	"			"14	"		
19	"	49	"			19	"	49	"			"15	"		
20	56K	50	100K			20	"	50	"	MOS IC		"16	"		
21	22K	51	56K			21	"	51	"	3M1	HD703M	"17	"		
22	"	52	"			22	"	52	"	" 2	"	"18	"		
23	"	53	"			23	"	53	"	" 3	"				
24	"	54	"			24	"	54	"	" 4	"				
25	"	55	100K			25	"	55	"	" 5	"				
26	100K	56	56K			26	"	56	"	" 6	"				
27	22K	57	"			27	"	57	"	4M1	HD704M				
28	"	58	"			28	"	58	"	" 2	"				
29	"	59	22K			29	"	59	"	5M1	HD705M				

All Resistors 1/4W unless otherwise noted.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

P. C. BOARD #3



Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

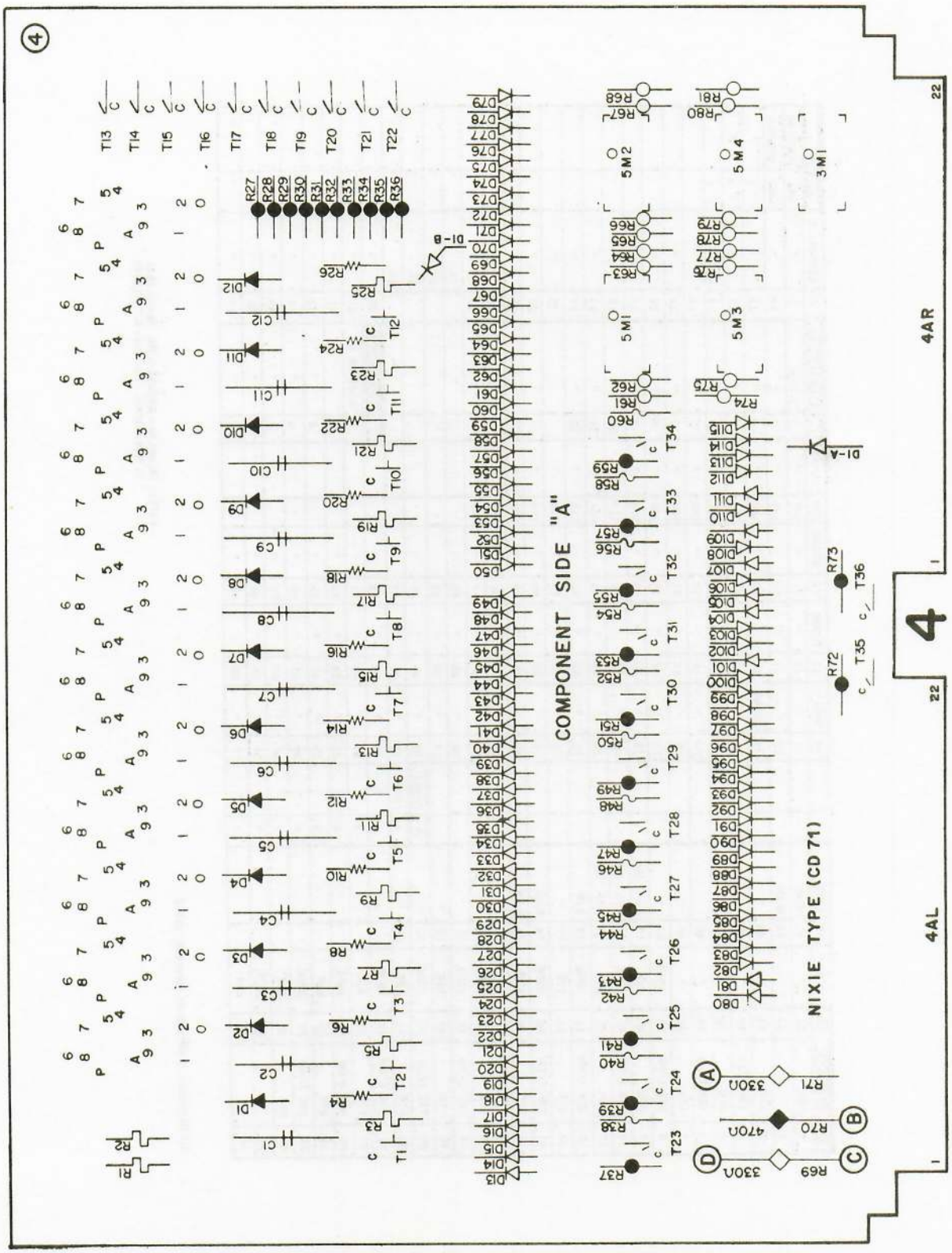
COMPONENT IDENTIFICATION P. C. BOARD #3

R	RESISTOR	30	100K			20	HS996	50	HS996	MOS IC		12M12	HD712M		
1	22K	31	22K			21	"	51	"	1M1	HD701M	" 13	"		
2	"	32	56K			22	"	52	"	" 2	"	" 14	"		
3	"	33	"			23	"	53	"	" 3	"	" 15	"		
4	56K	34	22K			24	"	54	"	" 4	"	" 16	"		
5	22K	35	"			25	"	55	"	" 5	"	" 17	"		
6	56K	36	"			26	"	56	"	" 6	"	" 18	"		
7	100K	37	"			27	"	57	"	3M1	HD703M	" 19	"		
8	22K	38	"			28	"	58	"	" 7	"	" 20	"		
9	"	39	"			29	"	59	"	" 3	"				
10	100K	40	56K	D	DIODE	30	"	60	"	" 4	"				
11	56K	41	22K	1	HS996	31	"	61	"	4M1	HD704M				
12	22K	42	56K	2	"	32	"	62	"	" 2	"				
13	"	43	100K	3	"	33	"	63	"	" 3	"				
14	"	44	56K	4	"	34	"	64	"	5M1	HD705M				
15	"	45	100K	5	"	35	"	65	"	" 2	"				
16	"	46	56K	6	"	36	"			7M1	HD707M				
17	"	47	"	7	"	37	"			" 2	"				
18	"	48	22K	8	"	38	"			" 3	"				
19	"	49	56K	9	"	39	"			12M1	HD712M				
20	"	50	"	10	"	40	"			" 2	"				
21	100K	51	22K	11	"	41	"			" 3	"				
22	56K	52	"	12	"	42	"			" 4	"				
23	"	53	"	13	"	43	"			" 5	"				
24	100K	54	56K	14	"	44	"			" 6	"				
25	"	55	22K	15	"	45	"			" 7	"				
26	56K	56	56K	16	"	46	"			" 8	"				
27	100K	57	"	17	"	47	"			" 9	"				
28	56K	58	"	18	"	48	"			"10	"				
29	"	59	"	19	"	49	"			"11	"				

All Resistors 1/4W unless otherwise noted.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

P. C. BOARD #4 (Type CD71 Number Indicating Tubes)



Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

COMPONENT IDENTIFICATION P. C. BOARD #4 (Type C971)

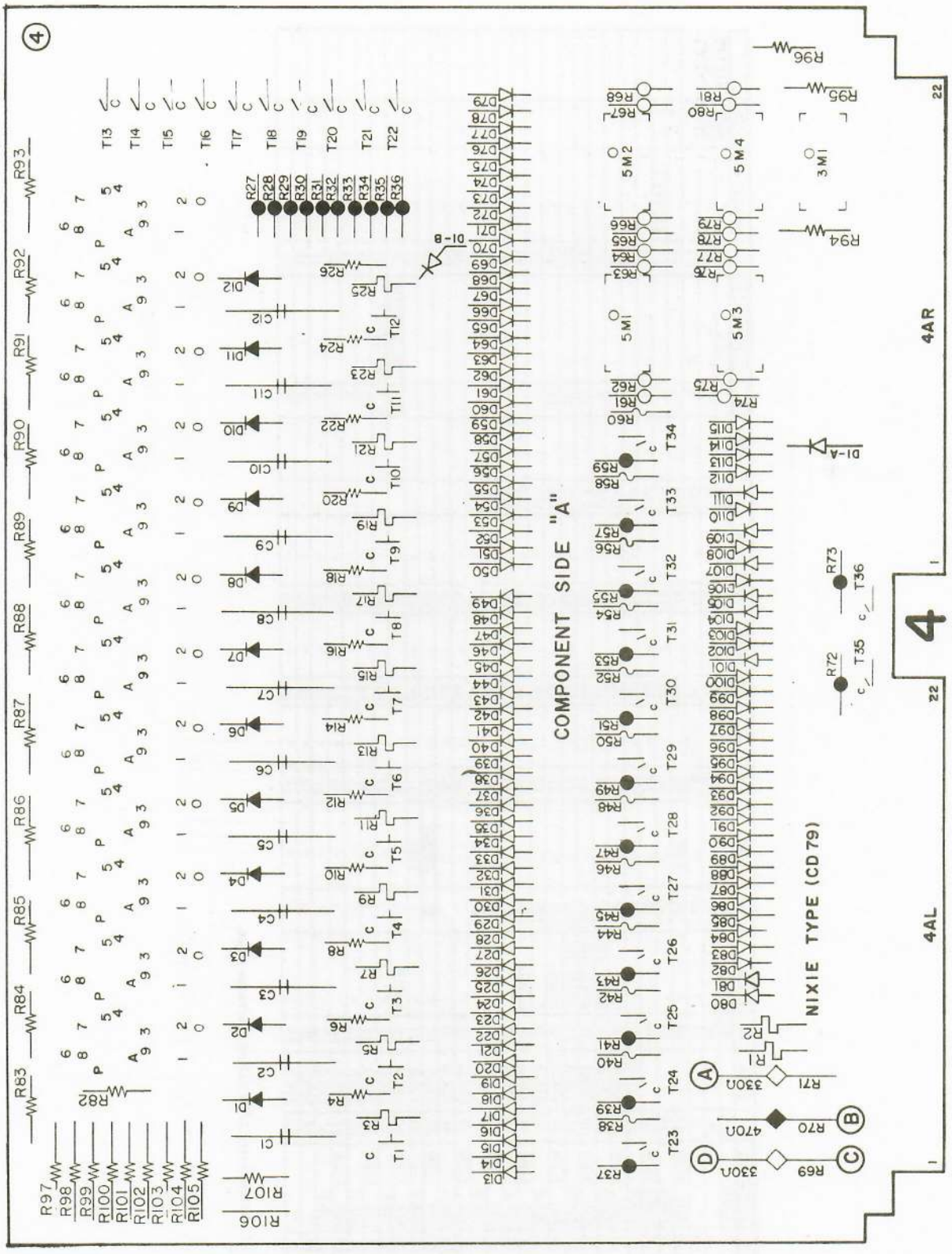
R	RESISTOR	30	22K	60	100K				10	1S84	40	HS996	70	HS996	100	HS996	C	CAPACITOR	8	2SA549A K	MOS IC	
1	33K	31	"	61	8.2K				11	"	41	"	71	"	101	"	1	8.15μf	9	"	3M1	HD703M
2	"	32	"	62	8.2K				12	"	42	"	72	"	102	"	2	"	10	"	5M1	HD705M
3	47K	33	"	63	"				13	HS996	43	"	73	"	103	"	3	"	11	"	" 2	"
4	33K 1/2W	34	"	64	"				14	"	44	"	74	"	104	"	4	"	12	"	" 3	"
5	47K	35	"	65	"				15	"	45	"	75	"	105	"	5	"	13	2SC284 K	" 4	"
6	33K 1/2W	36	"	66	"				16	"	46	"	76	"	106	"	6	"	14	"		
7	47K	37	"	67	"				17	"	47	"	77	"	107	"	7	"	15	"		
8	33K 1/2W	38	100K	68	"				18	"	48	"	78	"	108	"	8	"	16	"		
9	47K	39	22K	69	330 1W				19	"	49	"	79	"	109	"	9	"	17	"		
10	33K 1/2W	40	100K	70	470 1W				20	"	50	"	80	"	110	"	10	"	18	"		
11	47K	41	22K	71	330 1W				21	"	51	"	81	"	111	"	11	"	19	"		
12	33K 1/2W	42	100K	72	22K				22	"	52	"	82	"	112	"	12	"	20	"		
13	47K	43	22K	73	"				23	"	53	"	83	"	113	"			21	"		
14	33K 1/2W	44	100K	74	8.2K				24	"	54	"	84	"	114	"			22	"		
15	47K	45	22K	75	8.2K				25	"	55	"	85	"	115	"			23	"		
16	33K 1/2W	46	100K	76	"				26	"	56	"	86	"	1-B	"			24	"		
17	47K	47	22K	77	"				27	"	57	"	87	"	1-B	"			25	"		
18	33K 1/2W	48	100K	78	"				28	"	58	"	88	"								
19	47K	49	22K	79	"				29	"	59	"	89	"								
20	33K 1/2W	50	100K	80	"	D	DIODE	30	"	60	"	90	"									
21	47K	51	22K	81	"	1	1S84	31	"	61	"	91	"									
22	33K 1/2W	52	100K			2	"	32	"	62	"	92	"			T	TRANSISTOR	30	"			
23	47K	53	22K			3	"	33	"	63	"	93	"			1	2SA549A K	31	"			
24	33K 1/2W	54	100K			4	"	34	"	64	"	94	"			2	"	32	"			
25	47K	55	22K			5	"	35	"	65	"	95	"			3	"	33	"			
26	33K 1/2W	56	100K			6	"	36	"	66	"	96	"			4	"	34	"			
27	22K	57	22K			7	"	37	"	67	"	97	"			5	"	35	"			
28	"	58	100K			8	"	38	"	68	"	98	"			6	"	36	"			
29	"	59	22K			9	"	39	"	69	"	99	"			7	"	37	"			

All Resistors 1/4W unless otherwise noted.

NOTE: Do not use 2SC284 K BK (Black Dot) in this circuit. Use only KTA-1234Z.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

P. C. BOARD #4 (Type CD79 Number Indicating Tubes)



Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

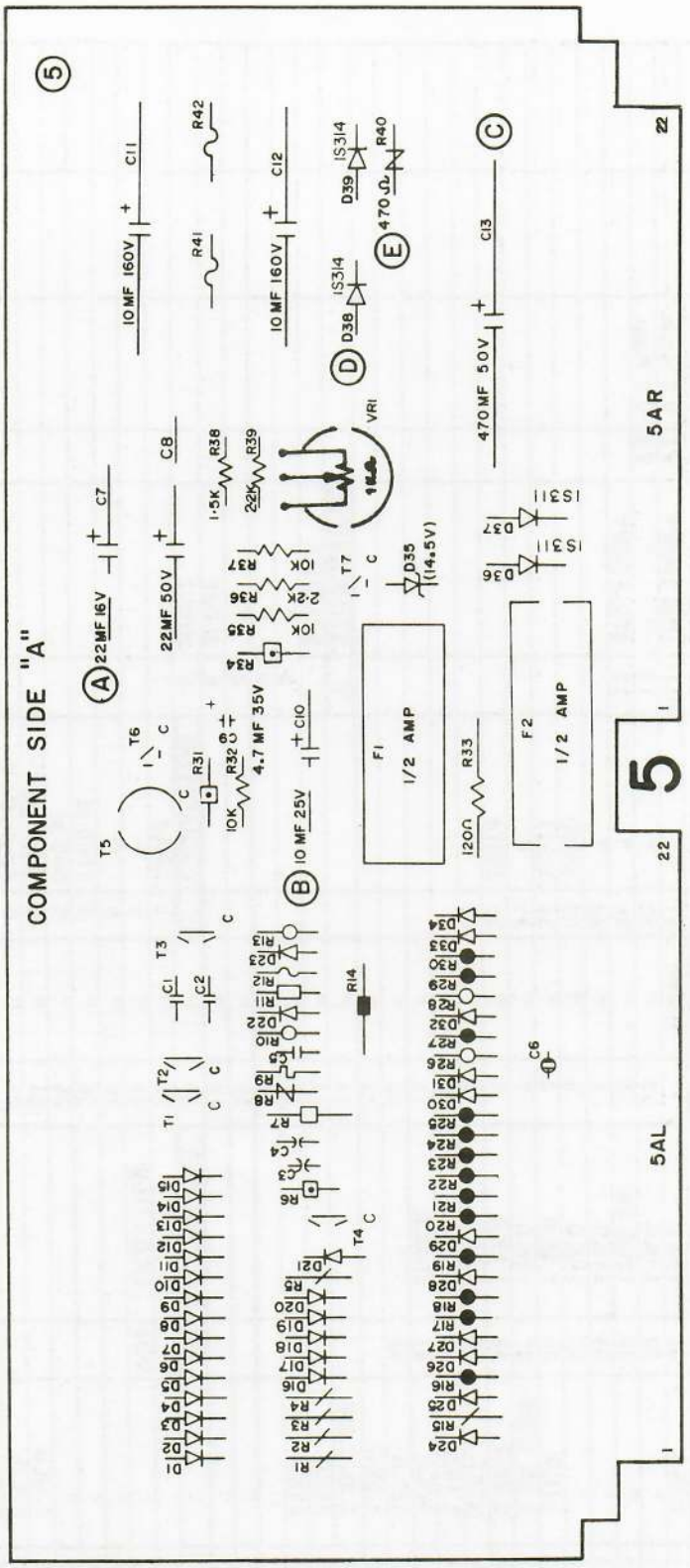
COMPONENT IDENTIFICATION P. C. BOARD #4 (Type CD79)

R	RESISTOR	30	15KΩ	60	39KΩ	90	3.9K	10	1S84	40	HS996	70	HS996	100	HS996	C	CAPACITOR	8	2SA549A K	MOS IC	
1	33KΩ	31	"	61	5.6KΩ	91	"	11	"	41	"	71	"	101	"	1	0.15μf	9	"	3M1	HD703M
2	"	32	"	62	"	92	"	12	"	42	"	72	"	102	"	2	"	10	"	5M1	HD705M
3	"	33	"	63	"	93	"	13	HS996	43	"	73	"	103	"	3	M	11	"	"	2
4	33K 1/2W	34	"	64	"	94	56K	14	"	44	"	74	"	104	"	4	"	12	"	"	3
5	33KΩ	35	"	65	"	95	"	15	"	45	"	75	"	105	"	5	"	13	2SC284 K	"	4
6	33KΩ 1/2W	36	"	66	"	96	"	16	"	46	"	76	"	106	"	6	"	14	"		
7	33KΩ	37	22KΩ	67	"	97	"	17	"	47	"	77	"	107	"	7	"	15	"		
8	33K 1/2W	38	39KΩ	68	"	98	"	18	"	48	"	78	"	108	"	8	"	16	"		
9	33KΩ	39	22KΩ	69	330Ω 1W	99	"	19	"	49	"	79	"	109	"	9	"	17	"		
10	33K 1/2W	40	39KΩ	70	470Ω 1W	100	"	20	"	50	"	80	"	110	"	10	"	18	"		
11	33KΩ	41	22KΩ	71	330Ω 1W	101	"	21	"	51	"	81	"	111	"	11	"	19	"		
12	33K 1/2W	42	39KΩ	72	22KΩ	102	"	22	"	52	"	82	"	112	"	12	"	20	"		
13	33KΩ	43	22KΩ	73	"	103	"	23	"	53	"	83	"	113	"			21	"		
14	33K 1/2W	44	39KΩ	74	8.2KΩ	104	"	24	"	54	"	84	"	114	"			22	"		
15	33KΩ	45	22KΩ	75	"	105	"	25	"	55	"	85	"	115	"			23	"		
16	33K 1/2W	46	39KΩ	76	"	106	Jumper	26	"	56	"	86	"	1-A	"			24	"		
17	33KΩ	47	22KΩ	77	"	107	56K	27	"	57	"	87	"	1-B				25	"		
18	33K 1/2W	48	39KΩ	78	"			28	"	58	"	88	"					26	"		
19	33KΩ	49	22KΩ	79	"			29	"	59	"	89	"					27	"		
20	33K 1/2W	50	39KΩ	80	"	D	DIODE	30	"	60	"	90	"					28	"		
21	33KΩ	51	22KΩ	81	"	1	1S84	31	"	61	"	91	"					29	"		
22	33K 1/2W	52	39KΩ	82	3.9KΩ	2	"	32	"	62	"	92	"			T	TRANSISTOR	30	"		
23	33KΩ	53	22KΩ	83	"	3	"	33	"	63	"	93	"			1	2SA549A K	31	"		
24	33K 1/2W	54	39KΩ	84	"	4	"	34	"	64	"	94	"			2	"	32	"		
25	33KΩ	55	22KΩ	85	"	5	"	35	"	65	"	95	"			3	"	33	"		
26	33K 1/2W	56	39KΩ	86	"	6	"	36	"	66	"	96	"			4	"	34	"		
27	15KΩ	57	22KΩ	87	"	7	"	37	"	67	"	97	"			5	"	35	"		
28	"	58	39KΩ	88	"	8	"	38	"	68	"	98	"			6	"	36	"		
29	"	59	22KΩ	89	"	9	"	39	"	69	"	99	"			7	"	37	"		

All Resistors are 1/4W unless otherwise noted.

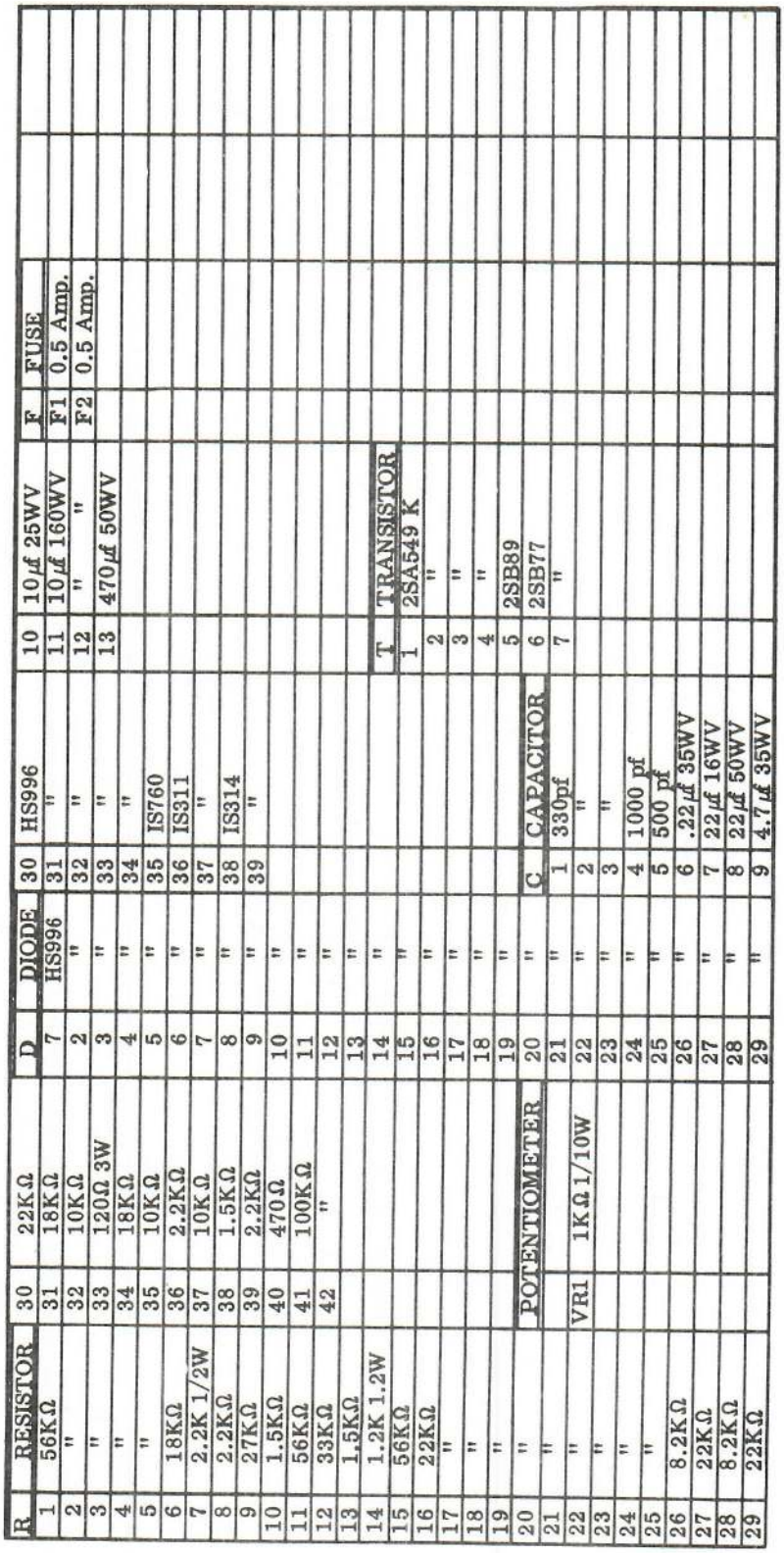
Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

P.C. BOARD #5



Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

COMPONENT IDENTIFICATION P.C. BOARD #5



R	RESISTOR	30	22KΩ
1	56KΩ	31	18KΩ
2	"	32	10KΩ
3	"	33	120Ω 3W
4	"	34	18KΩ
5	"	35	10KΩ
6	18KΩ	36	2.2KΩ
7	2.2K 1/2W	37	10KΩ
8	2.2KΩ	38	1.5KΩ
9	27KΩ	39	2.2KΩ
10	1.5KΩ	40	470Ω
11	56KΩ	41	100KΩ
12	33KΩ	42	"
13	1.5KΩ		
14	1.2K 1.2W		
15	56KΩ		
16	22KΩ		
17	"		
18	"		
19	"		
20	"	POTENTIOMETER	
21	"	VR1	1KΩ 1/10W
22	"		
23	"		
24	"		
25	"		
26	8.2KΩ		
27	22KΩ		
28	8.2KΩ		
29	22KΩ		

D	DIODE	30	HS996
7	HS996	31	"
2	"	32	"
3	"	33	"
4	"	34	"
5	"	35	IS760
6	"	36	IS311
7	"	37	"
8	"	38	IS314
9	"	39	"
10	"		
11	"		
12	"		
13	"		
14	"		
15	"		
16	"		
17	"		
18	"		
19	"		
20	"	C	CAPACITOR
21	"	1	330pf
22	"	2	"
23	"	3	"
24	"	4	1000 pf
25	"	5	500 pf
26	"	6	.22μf 35WV
27	"	7	22μf 16WV
28	"	8	22μf 50WV
29	"	9	4.7μf 35WV

10	10μf 25WV
11	10μf 160WV
12	" "
13	470μf 50WV
T	TRANSISTOR
1	2SA549 K
2	"
3	"
4	"
5	2SB89
6	2SB77
7	"

F	FUSE
F1	0.5 Amp.
F2	0.5 Amp.

All Resistors 1/4W unless otherwise noted.

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

WIRING LIST & SIGNAL LIST CROSS REFERENCE (TERMINAL TO SIGNAL)

CONNECTOR ASSY. SIDE / PIN NUMBER		P.C. BOARD #1		P.C. BOARD #2		P.C. BOARD #3	
		SIDE A	SIDE B	SIDE A	SIDE B	SIDE A	SIDE B
L	1	*BT8	*IR	*BCF	*POF	CP1	AR14CP1
	2	*BT1	KSH·OSF	DT14	*DIF	DTF·BT1	BT8
	3	*BT8	(AR1=0)	CHASF·MSF	*OSF·KR	*CAF	BT8
	4	*CC2	POF	CC4·DTO·BCF·(MLF+DIF)	* OVF	CP2	DT0·DT14·BCF·DTF·BT1
	5	*DT14	*DT13	KMΘ	*OSF·KMR	*MSF	*(AR14=0)
	6	*DTF	*DTF	BT8	*MLF	DIF	*CC6·SCF·DIF
	7	*CC3	(MLF+DIF)	MSF	*ASF	DIF	DTF+BT8·DT14
	8	CC6·SCF	*CFCP1	KS	*ISF	*CC6·SCF·DIF	CC6·SCF
	9	*DFCP1	*CCCP1	K(N+P)	MIF	MLF	*MIF
	10	*AR14CP1	*MRCP1	CC5	KC	*MIF	ASF
	11	———	*CC1	KN	CC0	CCCP1	*CHASF·MSF
	12	*CC5	*DT11	CC2	*OSF·SCF·K(N+P)	*DTF+BT1	KMC
	13	*IRCP1	*CC4	DTO	———	(BT1+BT8)	CC0
	14	*ARCP1	OSF·SCF·KN	*CC6·SCF	K(A+S)	ASF	BT1
	15	CP1	*CC8	DT12	*ISF	*Y3	BT1
	16	*ASO	*CC7	*OSF·SCF	*OSF·SCF	DT0	OSF
	17	(KSH·OSF·CC8)	*CC6	CC3	*OSF·SCF·KN	(BT2+BT4)	K(M⊕+M⊖)
	18	CC6·SCF·DIF	*CC9	KSH	*CC6·SCF·RDF	CC0	KΣ
	19	———	*DT11	*OSF	*CC6·SCF·DIF	DT14	*CC9·BT1
	20	———	*DT14	———	*MLF	DTF	DT13
	21	———	*DT1	(AR1=0)+(AR1=15)	(IR1=0)	*CC4·DT0·BCF·(MLF+DIF)	BCF
	22	*(DC-2)	(BCD-1)	KR	KMR	CC6·SCF·RDF	(MLF+DIF)
R	1	*CC0	CP2	CC9·BT1	KM	CC4	BCF
	2	*(IR14=15)·DT0	*(DC-1)	(AR1=0)	(IR14=0)	DT0	DT1
	3	(BCD-8)	*(DC-8)	DT11	CC6	MLF	CC2
	4	*(DC-1)	*DT0	(AR1=0)	(IR14=15)·DT0	K(A+S)	KN·OSF·SCF
	5	IR13in	*(DC-8)	*(CC6·SCF)	CC6·SCF·DIF	OSF·SCF	*CC8+KSH·OSF
	6	*DT0·DTM·BCF·DTF·BT1	BCF	DT0	*DIF	*IR13 in	KSH·OSF
	7	*(DC-4)	DTF+BT1	*CC-IN	*ASF	IR14	DTF
	8	———	*CC0	CC8	CC9	IR	CC8
	9	*DT13	*DT1	(IR1-4)	DT14	ASO	CC3
	10	*DT0	*(DC-4)	(IR14=0)	(IR1-8)	*IR14 in	CC9
	11	*(DC-2)	*(IR1-8)	KD	*OSF·KSH	CC6·SCF	KR·OSF
	12	*DT12	*(IR1-4)	*(MLF+DIF)	MIF	BCD	KMR·OSF
	13	*(IR1=0)	*(IR1-2)	CAF	BT1	DT1	OSF·SCF·K(N+P)
	14	*(IR14=0)	IR14 in	BT1	CC1	OSF·SCF	KC
	15	*IR14	*IR	(AR14=0)	———	ISF	CC6
	16	CC-IN	(BCD-2)	CC0	CC7	DT13	CC1
	17	*(IR14=0)	*(BT1+BT8)	*BCF	CC4	*(AR1=0)	*AR1=15+AR1=0
	18	*(DTF·BT1)	DT-VDD(-14V)	KEY	JSCF	CC5	*(AR1=0)
	19	(BCD-4)	*BCD	CFCP1	KP	CC7	MRCP1
	20	*BT1	VDD(-14V)	CCCP1	VDD(-14V)	*JSCF	VDD(-14V)
	21	*(BT2·BT4)	VGG(-24V)	Y3	VGG(-24V)	ISF	VGG(-24V)
	22	*DTF·BT8·DT14	E(ØV)	CP2	E(ØV)	ARCP1	E(ØV)

CONNECTOR ASSY. SIDE / PIN NUMBER	P.C. BOARD #4		P.C. BOARD #5			
	SIDE A	SIDE B	SIDE A	SIDE B		
L		-103V	5	8	1	L
		+103V	1	0	2	
			9	4	3	
			7	*KP	4	
		*ISL-P	3	———	5	
			2	*KMR	6	
		*OFL-P	6	*KΣ	7	
			KEY EO(ØV)	*KR	8	
			CP2I	———	9	
			CP1I	———	10	
			CP10	———	11	
			CP20	*KSH	12	
		DT1	*KM	*KMC	13	
		(DC-8)	*KD	KEY Ei(ØV)	14	
		(DC-4)	*KC	KEY 0	15	
		(DC-2)	*KMΘ	*K(A+S)	16	
		(DC-1)	*K(M⊕+M⊖)	*KS	17	
		(DC-1)	VGGi	VDD0	18	
		(DC-2)	DT-VDDi	VGG0	19	
		OVF	DT0i	VDDi	20	
		*OFL-K	KEYi	IRouti	21	
		*ISL-K	DT13i	(DTF·BT1)i	22	
R			*DT13	*DTF·BT1)	1	R
		ISF	*KEY	*IRout	2	
		CP2	*DT0	*VDD(-14V)	3	
		(DC-4)	*DT-VDD0	AC32V-1	4	
		(DC-8)	*VGG(-24V)	AC32V-2	5	
			*KS	———	6	
		DT11	*K(A+S)	F2-in	7	
			*K(M⊖+M⊖)	*F2 out	8	
		(IR1-2)	*KMΘ	PT-E	9	
		DFCP1	*KC	PT-C	10	
		CCCP1	*KMC	———	11	
		IR	*KD	———	12	
		IR	*CP2	*-103V	13	
			*KM	PT-B	14	
			*(BCD-4)	———	15	
			*(BCD-1)	*(BCD-2)	16	
		DT12	*CP1	*KSH	17	
		(IR1-8)	*KR	AC103V	18	
		(IR1-4)	*KΣ	*+103V	19	
		VDD(-14V)	*KMR	*KN	20	
		VGG(-24V)	*(BCD-8)	*K(N+P)	21	
		E(ØV)	*KP	*E(ØV)	22	

* Indicates Output Signal

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

WIRING LIST & SIGNAL LIST CROSS REFERENCE (SIGNAL TO TERMINAL)

SIGNAL	TERMINAL				SIGNAL	TERMINAL			
OSF	2AL19	3BL16			BT1	1AL2	2BR13	3BL15	
POF	1BL4	2BL1			$\overline{BT1}$	1AR20	2AR14	3BL14	
OVF	2BL4	4BL20			BT8	1AL3	2AL6	3BL3	
BCF	1BR6	2AL1	3BL21		$\overline{BT8}$	1AL1	3BL2		
$\overline{BCF}$	2AR17	3BR1			(BT1 + BT8)	1BR17	3AL13		
DTF	1AL6	3BR7			(BT2 + BT4)	1AR21	3AL17		
$\overline{DTF}$	1BL8	3AL20			(BCD-1)	1BL22	5AR16		
MIF	2BR12	3AL10			(BCD-2)	1BR16	5BR16		
$\overline{MIF}$	2BL9	3BL9			(BCD-4)	1AR19	5AR15		
MLF	2BL6	3AR3			(BCD-8)	1AR3	5AR21		
$\overline{MLF}$	2BL20	3AL9			(DC-1)	1AR4	4BL17		
DIF	2BR6	3AL7			$\overline{(DC-1)}$	1BR2	4BL18		
$\overline{DIF}$	2BL2	3AL6			(DC-2)	1AL22	4BL19		
ISF	2BL8	3AR21	4BR2		$\overline{(DC-2)}$	1AR11	4BL16		
$\overline{ISF}$	2BL15	3AR15			(DC-4)	1AR7	4BR4		
ASF	2BL7	3AL14			$\overline{(DC-4)}$	1BR10	4BL15		
$\overline{ASF}$	2BR7	3BL10			(DC-8)	1BR5	4BL14		
MSF	2AL7	3AL5			$\overline{(DC-8)}$	1BR3	4BR5		
CAF	2AR13	3AL3							
					K(A + S)	2BL14	3AR4	5AR7	
CC0	1BR8	2AR16	3AL18		KS	2AL8	5AR6		
$\overline{CC0}$	1AR1	2BL11	3BL13		KC	2BL10	3BR14	5AR10	
CC1	1BL11	2BR14	3BR16		KMC	3BL12	5AR11		
CC2	1AL4	2AL12	3BR3		KR	2AL22	5AR18		
CC3	1AL7	2AL17	3BR9		KMR	2BL22	5AR20		
CC4	1BL13	2BR17	3AR1		KSH	2AL18	5BR17		
CC5	1AL12	2AL10	3AR18		K(M ⊕ + M ⊖)	3BL17	5AR8		
CC6	1BL17	2BR3	3BR15		KM ⊖	2AL5	5AR9		
CC7	1BL16	2BR16	3AR19		KM	2BR1	5AR14		
CC8	1BL15	2AR8	3BR8		KD	2AR11	5AR12		
CC9	1BL18	2BR8	3BR10		KP	2BR19	5AR22		
					KN	2AL11	5BR20		
DT0	1AR10	2AL13	3AL16	5AR3	K(N + P)	2AL9	5BR21		
$\overline{DT0}$	1BR4	2AR6	3AR2		KEY	2AR18	5AR2		
DT1	1BR9	3AR13			KE	3BL18	5AR19		
$\overline{DT1}$	1BL21	3BR2	4BL13						
DT11	1BL19	2AR3							
$\overline{DT11}$	1BL12	4BR7							
DT12	1AR12	2AL15	4BR17						
DT13	1BL5	3BL20	5AR1						
$\overline{DT13}$	1AR9	3AR16							
DT14	1BL20	2BR9							
$\overline{DT14}$	1AL5	2AL2	3AL19						

SIGNAL		TERMINAL			SIGNAL	TERMINAL			
IR	1BR15	3AR8	4BR13	5BR2	CC9.BT1	2AR1	3BL19		
$\overline{IR}$	1BL1	4BR12			Y3	2AR21	3AL15		
IR13in	1AR5	3AR6			(IR14=15).$\overline{DT0}$	1AR2	2BR4		
IR14in	1BR14	3AR10							
IR14	1AR15	3AR7			CP1	1AL15	3AL1	5AR17	
(IR1-2)	1BR13	4BR9			CP2	1BR1	2AR22	3AL4	4BR3
(IR1-4)	1BR12	2AR9	4BR19			5AR13			
(IR1-8)	1BR11	2BR10	4BR18		CCCP1	1BL9	2AR20	3AL11	4BR11
(IR1=0)	1AR13	2BL21			CFCP1	1BL8	2AR19		
(IR14=0)	1AR17	2BR2			DFCP1	1AL9	4BR10		
$\overline{(IR14=0)}$	1AR14	2AR10			ARCP1	1AL14	3AR22		
(AR1=0)	2AR2	3AR17			AR14CP1	1AL10	3BL1		
$\overline{(AR1=0)}$	1BL3	2AR4	3BR18		MRCP1	1BL10	3BR19		
$\overline{(AR14=0)}$	2AR15	3BL5			+103V	4BL2	5BR19		
					-103V	4BL1	5BR13		
CC-IN	1AR16	2AR7			VDD(-14V)	1BR20	2BR20	3BR20	4BR20
ASO	1AL16	3AR9				5BR3			
BCD	1BR19	3AR12			VGG(-24V)	1BR21	2BR21	3BR21	4BR21
JSCF	2BR18	3AR20				5AR5			
OFL-K	4BL21	(OVF Lamp)			E (ØV)	1BR22	2BR22	3BR22	4BR22
OFL-P	4BL7					5BR22	(Transformer Pin #3)		
ISL-K	4BL22				DT-VDD (-14V)	1BR18	5AR4 (To Bit Counter)		
ISL-P	4BL5								
(DTF.BT1)	1AR18	3AL2	5BR1		AC32V-1	5BR4	(Transformer Pin #1)		
$\overline{DTF}$+BT1	1BR7	3AL12			AC32V-2	5BR5	(Transformer Pin #2)		
(AR1=15)+(AR1=0)	2AL21	3BR17			F2 IN	5BR7	(To AC Fuse)		
CH.ASF.MSF	2AL3	3BL11			F2 OUT	5BR8	(Transformer Primary)		
(MLF+DIF)	1BL7	2AR12	3BL22		PT-C	5BR10	(PT-Collector)		
CC6.$\overline{SCF}$.DIF	1AL18	2BR5	3AL8		PT-B	5BR14	(PT-Base)		
CC6.$\overline{SCF}$.$\overline{DIF}$	2BL19	3BL6			AC103V	5BR18	(Transformer - Terminal 103V)		
CC6.SCF.$\overline{RDF}$	2BL18	3AL22			PT-E	5BR9	(PT-Emitter)		
CC6.SCF	2AL14	3AR11							
CC6.$\overline{SCF}$	1AL8	2AR5	3BL8						
OSF.SCF	2AL16	3AR5							
OSF.$\overline{SCF}$	2BL16	3AR14							
OSF.SCF.KN	1BL14	2BL17	3BR4						
OSF.$\overline{SCF}$.K(N+P)	2BL12	3BR13							
OSF.KSH	1BL2	2BR11	3BR6						
OSF.KR	2BL3	3BR11							
OSF.KMR	2BL5	3BR12							
(OSF.KSH+CC8)	1AL17	3BR5							
CC4.DT0.$\overline{BCF}$.(MLF+DIF)	2AL4	3AL21							
DT0.DT14.$\overline{BCF}$.DTF.BT1	1AR6	3BL4							
$\overline{DTF}$+$\overline{BT8}$.$\overline{DT14}$	1AR22	3BL7							

NOTE: PT = Power Transistor

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

WIRING LIST & SIGNAL LIST CROSS REFERENCE

SIGNAL	COLOR OF WIRE	CONNECTOR	TERMINAL	NUMBER
0	black	5BL2		
1	brown	5AL2		
2	red	5AL6		
3	orange	5AL5		
4	yellow	5BL3		
5	green	5AL1		
6	blue	5AL7		
7	purple	5AL4		
8	gray	5BL1		
9	white	5AL3		
KP	pink	5BL4		
KR	light blue	5BL8		
KMR	black	5BL6		
KΣ	brown	5BL7		
KM	red	5AL13		
KD	orange	5AL14		
KC	yellow	5AL15		
KMC	green	5BL13		
KSH	blue	5BL12		
K(M⊕ + M⊖)	purple	5AL17		
KM⊖	gray	5AL16		
K(A+S)	pink	5BL16		
KS	light blue	5BL17		
KEY Eo (ØV)	white	5AL8		
E (ØV)	white	5AL8	5BL14	
VGG (-24V)	blue	5BL19	5AL18	
VDD (-14V)	green	5BL18	5BL20	5AL19
CP1 (OSC.)	pink	5AL11	5AL10	
CP2 (OSC.)	light blue	5AL12	5AL9	
KEY	orange	5BL15	5AL21	

KEYBOARD CONNECTOR WIRING - 5AL, 5BL

Courtesy the Old Calculator Museum
http://oldcalculatormuseum.com

TIMING CHART

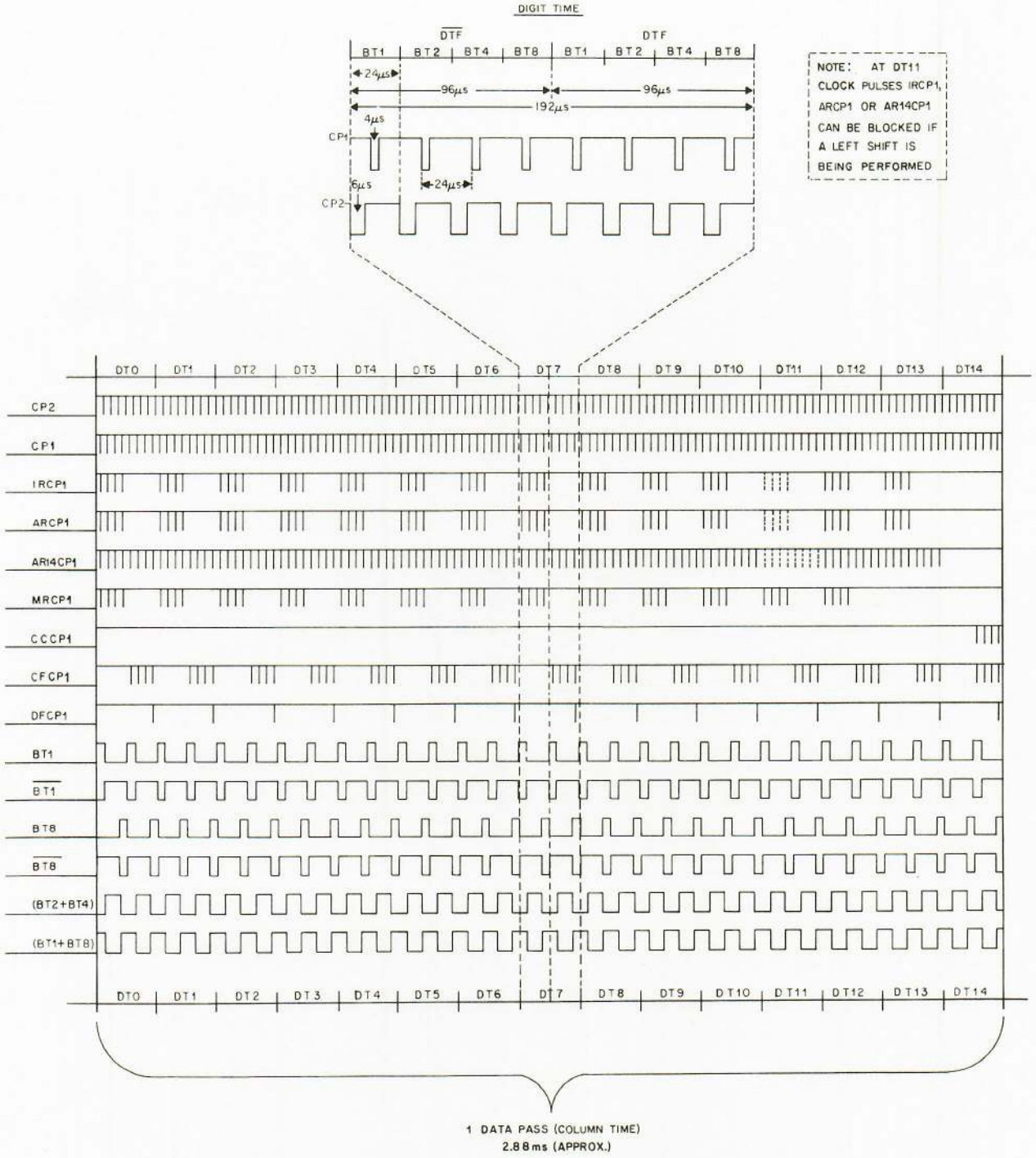


Figure 7-9