CONTROL DATA® 6600
COMPUTER SYSTEM

6601A-J, 6613-A/B/C
6604A/B/C, 6614-A/B/C
Central Processor
(Including Functional Units)

Volume 1

DIAGRAMS &
CIRCUIT DESCRIPTION

CONTROL DATA
CORPORATION

CUSTOMER ENGINEERING MANUAL

RECORD of REVISIONS

REVISION	NOTES
A	Equation Lists and Appendix A added. Miscellaneous changes made for purposes of clarification.
B	Central Memory diagrams added, sheets 38-44. Central Processor diagrams added, sheets
	45-114. Clock diagrams added, sheets 115-121. Corrections made to Appendix A.
C	This reprint obsoletes all previous editions. Central Processor completely revised.
(3-29-65)	Miscellaneous changes made for purposes of clarification. This printing includes Change Order
	10946.
D	Volumes 1 and 2 obsolete all previous editions. "Add Unit" diagrams added in Volume 1.
(7-12-65)	Miscellaneous changes made for purposes of clarification. This printing includes Change Order
	11826.
E	Change Order 12006.
F	Change Order 12051.
G	Change Order 12082.
H	Change Order 12182.
J	Change Order 12187.
K	Publication Change Order 12481. The following pages have been revised: 6601/04 Central
(1-27-66)	Processor - 1, 3, 4.1, 4.2, 5, 7, 9, 13, 15, 16, 19, 21, 23, 24.1, 25, 29, 31, 33, 35, 43, 53,
	57, 61, 63, 64, 65, 66, 67, 68.0, 68.1, 68.3, 68.5, 68.7, 68.9, 69, 70, 71, 73, 74, 74.1, 75,
	77, 79, 80.1, 80.3, 81, 82, 83, 85, 86.1, 86.3, 87, 88.1, 89, 91, 99, 101 and 103. 6601/04
	Functional Units - Contents, 2.3, 3, 5, 7, 9, 11, 13, 14.1, 14.3, 14.4, 14.5, 14.7, 14.8, 15,
	16, 17, 19, 23, 25, 26, 29, 31, 33, 35, 37, 43, 44, 47, 48, 49, 51, 53, 55, 57, 59, 61, 63, 64,
	67, 95, 97, 103, 126.1, 126.2, 127, 181, 182.0, 183, 185, 187, 189, 190, 191, 193, 194, 197,
	201, 205, 207, 211, 213 and Comment Sheet. 6601/04 Peripheral and Control Processor - 1, 2,
	3, 5, 9, 11, 19, 21, 39, 47, 55, 57, 58, 59, 61, 63, 65 and 67. 6601 Central Memory (131K) -
	Contents, 1, 3, 5, 11, 12.0, 12.1, 12.2, 13, 14.0, 14.1, 14.3, 15. 6604 Central Memory (65K)
	Contents, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14.1, 15. 6601/04 Clock - 3, 5, 7, 9, 11,
	13, 15, 17. 6601/04 Power Wiring - Contents, 1, 3, 5, 6, 7, 9, 11, 12, 13, 15, 17. 6601/04
	Appendix A - Title page, 2 and Comment Sheet.
L	Field Change Order 13358 which advanced the Product Designation to 6601-H31, 6604-A33 and
(6-28-66)	6605-A12. Central Processor pages 11, 13, 17, 19, 22.1, 24.01, 24.1, 25, 37, 41, 45, 86.01
	and 90.1 revised. Functional Units pages 5, 11, 14.21, 14.3, 14.5, 182 and 185 revised.
M	Publications Change Order 13629 which incorporated Change Orders 11310, 11389, 11467, 11487,
(6-28-66)	11826, 11937, 12006, 12450, 12543, 12655, 12656 and 12761 into this Manual. Vol. 1 Pages
	changed: Cover, Title page, Record of Revisions, Key to Logic Symbols, Central Processor
	Contents, 7, 19, 21, 22.1, 23, 24.01, 24.1, 35, 36.1, 39, 40.1, 43, 44.1, 63, 79, 80.01, 80.3,
	81, 82.1, 85, 86.01, 86.1, 86.3, 87, 88.1, 89, 90.1, 91, 101, 103 and 105. Functional Unit
	Contents, 3, 5, 7, 9, 11, 13, 14.1, 14.21, 14.3, 14.5, 14.7, 15, 39, 63, 99, 101, 103, 137,
	149, 173, 181, 182.01, 182.1, 185, 187, 197 and 213. Vol. 2 Pages changed: Cover, Title page, Record
	of Revisions, Key to Logic Symbols, Peripheral Processor Contents, 5, 6.1 and 6.2. Central
	Memory (131K) Contents, 10.1 and 11. Central Memory (65K) Contents, 11 and 12.1.

FORM CA230 REV. 1-67

Pub No. 60119300

Printed in United States of America

Address comments concerning this manual to:

Control Data Corporation
Technical Publications Department
4201 North Lexington Avenue
St. Paul, Minnesota 55112

or use Comment Sheet in the back of this manual.

RECORD of REVISIONS (CONT'D)

REVISION	NOTES
M (Con't)	Clock Contents 1, 3, 4.1, 11, 12.1.
N thru AP	Publication Change Order 14798 which incoporated the following Change Orders: 13415, 13512,
(10-10-66)	13513, 12930, 13536, 13545, 13569, 12932, 12933, 12931, 13731, 13814, 14136, 13999, 14045,
	14111, 14112, 14153, 13757, 14202, 14203, 13746 and 13756. Vol. 1 Pages changed: Central
	Processor 23, 24.01, 24.1, 31, 55, 71, 86.01, 86.3, 87, 89 and 90.1. Functional Units 19, 31,
	61, 63, 73, 93, 125, 157, 161, 175, 197, 209, 211 and 213. Vol. 2 Peripheral and Control
	Processor page 67. 6601/6613 Central Memory (131K) page 7. 6604/6614 Central Memory
	(65K) pages 7 and 12.1. 6601/6604/6613/6614 Clock page 15 and 17.
AR thru BD	Publication Change Order 16000 which incorporated the following Change Orders: 14247, 14767,
(3-22-67)	15218, 15252, 15338, 15339, 15191, 15562, 15563, 15068 and 15750. Vol. 1 pages changed:
	Central Processor 31, 39, 81, 82.1, 89, 99 and 105. Functional Units 101. Vol. 2 pages
	changed: Peripheral and Control Processor 5, 15 and 23. Clock - Contents and page 3. Power
	Wiring - Pages 7, 9, 13 and 15 revised. Pages 19, 21, 23 and 25 added. Cover and title page
	revised for Vol. 1 and Vol. 2.
BE thru BL	Publication Change Order 17138, which incorporated the following change orders: 14642, 15841,
(11-9-67)	15829, 15968, 15985 and 16269. Vol. 1 pages changed: General Contents (new). Central
	Processor 17 and 91. Functional Units 3, 14.5, 31, 93, 101, 103, 107, 117, 123, 163, 173,
	181, 182.0, 187, 189, 203 and 213. Vol. 2 pages changed: Cover and Title Page, General
	Contents (new), Peripheral and Control Processors 3, 5, 25, 29, 49, 53, 55 and 57. Central
	Memory (131K) 8.1. Central Memory (65K) 8.1. Clock, page 3. E.C.S. Coupler Section
	Pages 1 thru 37 (new). Power Wiring 1, 19, 21, 23 and 25.
BM thru BT	Publication Change Order 18437, which incorporated the following Change Orders: 17014, 17269,
(1-12-68)	17251, 17346 and 17660. Vol. 1 pages changed: Central Processor, pages 7, 9, 11, 13, 19, 22.1,
	24.01, 41, 67, 68.7, 69, 71, 87, 90.1, 91 and 105; Functional Units pages 3, 5, 7, 9, 11, 14.1, 14.3, 14.5,
	14.7, 19, 31, 61, 63, 75, 77, 79, 81, 83, 85, 93, 95, 97, 101, 103, 105, 109, 111, 117, 119, 121, 123, 125, 133,
	135, 137, 139, 141, 145, 147, 149, 151, 153, 155, 159, 161, 163, 165, 167, 169, 173, 175, 180.3, 180.7, 185, 197,
	199, 207, 209, 211 and 213. Vol. 2 pages changed: Peripheral and Control Processors pages 6.1, 7, 9,
	19, 21, 25, 29, 31, 37, 39, 41, 47, 51, 55, 57, 59, 61, 65, and 67; Central Memory (131K) pages 4, 5, 8, 8.1,
	11 and 15. Central Memory (65K) pages 4, 5, 8 and 15. Clock Section pages Clock Contents, 4.1,
	5, 7.1, 7.3 (new), 9.1, 9.3 (new), 12.1, 12.3 (new), 13, 15 and 17. Appendix pages 12 and 14.

FORM CA 230-2

GENERAL CONTENTS

VOLUME 1	Part 1.	Central Processor
	Part 2.	Functional Units
VOLUME 2	Part 3.	Peripheral and Control Processors
	Part 4.	Central Memory 131K
	Part 5.	Central Memory 65K
	Part 6.	Clock
	Part 7.	Extended Core Storage Coupler (Standard Option 10102 and Special Option 60080 for 6601/04, Special Option 60080 for 6613/14/15)
	Part 8.	Power Wiring
	Part 9.	Appendix A

FOREWORD

Logic diagrams contained in this manual do not attempt to show the entire device, nor even depict complete modules within that device. The purpose of the diagrams is to show the logical significance of circuits that may involve parts of many modules on several chassis. Logic hardware that is not pertinent to the particular logic sequence being illustrated is not included. Certain areas may not be shown at all, while others may appear on several drawings. These limitations are important to remember; the logic diagrams do not replace the 6000 Series chassis and cable tabs, but they are a valuable tool in understanding the tabs and the overall operation of the machine.

Central Processor
Pub. No. 60119300
Rev. D

KEY TO LOGIC SYMBOLS

(Standard 6000 Series Card Types)

Logic diagrams represent a symbolic approach to electronic schematics. By using symbols to represent building block circuits, the schematic becomes easy to read if the reader understands the function of the symbols. In CONTROL DATA* logic, two signals, a logical "0" and a logical "1" are the possible input or output conditions of a circuit. For example, "1" is considered "up" and "0" is considered "down" on a timing chart. Detailed descriptions of logic symbols and their associated electronic representations are contained in the Printed Circuit Manual, Cordwood Modules (Pub. No. 60042700).

STANDARD LOGIC SYMBOLS

Standard logic diagram symbols for Control Data equipment using 6000 Series card types are inverters, test points, flip-flops, twisted pair line drivers, and coaxial cable line drivers.

Inverters

An inverter is a logic element which provides an output that is a negation of its input. When more than one input is provided to an inverter, "0's" take precedence over "1's" and therefore drive the output of the inverter to "1". Because all of the several inputs have to be "1" to drive the output of the inverter to a "0", the inverter may be considered an inverting AND (or NAND) gate when more than one input is present. The basic inverter is shown in the logic diagrams as an arrow into either a circle or a square (Figure 1). Both symbols represent the same electronic circuit and have the same logic interpretation. In a logic sequence of inverters, circle and square symbols are usually alternated as an aid in tracing signals, e.g., a "1" output from a square symbol implies a "1" output from subsequent squares in the logic chain.

Figure 1. Inverter Symbols

Certain card types employ variations of the standard inverter building block. These differences are indicated in the logic diagrams by a dot or a cross in the circle or square (Figure 2). Both the chassis tabs containing the card in question and the Printed Circuit Manual, Cordwood Modules (Pub. No. 60042700) contain electronic schematics of these special variations.

*Registered trademark of Control Data Corporation

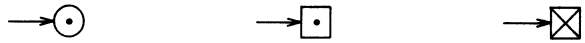


Figure 2. Special Inverters

Acceptable conventions for showing multiple inputs and outputs are given in Figure 3. Note that the output of inverter A is "0" only if inputs X, Y, and Z are all "1". The multiple outputs are identical.

Figure 3. Multiple Inputs/Outputs

Acceptable conventions for showing inverter networks are illustrated in Figure 4. As a general rule, circle inverters alternate with square inverters wherever possible. Because multiple outputs are identical, only one arrow is shown in cases where an inverter (A) serves as the single input to several succeeding inverters. In more complex inverter networks, multiple arrows are used (B to C and D; in this case because B is not the only input to C or D)

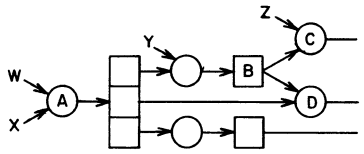


Figure 4. Inverter Networks

Test Points

A test point has no logic function, but is shown in the logic diagrams as a triangle (Figure 5). They are numbered from 1 to 6.

Figure 5. Test Point Symbols

Rev. M

Flip-Flops (FF)

The flip-flop (FF) is a storage device with two stable states--designated as Set and Clear--and is composed of two inverters (Figure 6). The flip-flop is said to be set when the set output (B) is a "1", and clear when it is a "0". Note that the input (A) must be "0" to set the flip-flop and (C) must be "0" to clear it.

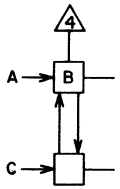


Figure 6. Flip-Flop Symbol

Logic signals are transmitted from one module to another by means of a line driver. Modules on the same chassis are connected with twisted pair lines, and those on separate chassis are connected by coaxial cable.

Twisted Pair Drivers

The twisted pair driver is represented by the standard square or circle. The output of the square or circle, however, is connected to a pin of the module in question and wired from there to a pin on another module (Figure 7). The ground wire of the pair is wired to the connector ground bus of each module. The pins are represented by small circles and are numbered from 1 to 28 (Pins 29 and 30 are ground and +6 volts, respectively, and generally are not shown in logic diagrams). The module location is shown above the card, and the module type is denoted in the upper right corner.

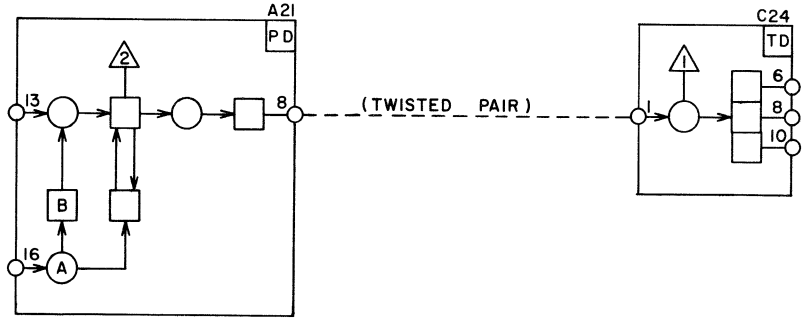


Figure 7. Twisted Pair Line Driver

Coaxial Cable Drivers

The coaxial cable driver is a 25 nsec pulse circuit, and is represented as shown in Figure 8. The pins used are represented by a small double circle.

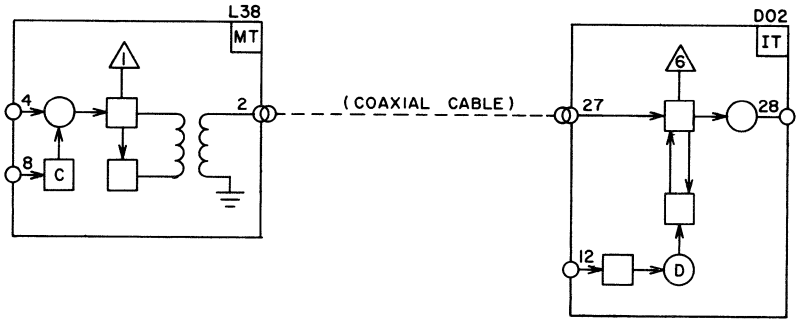


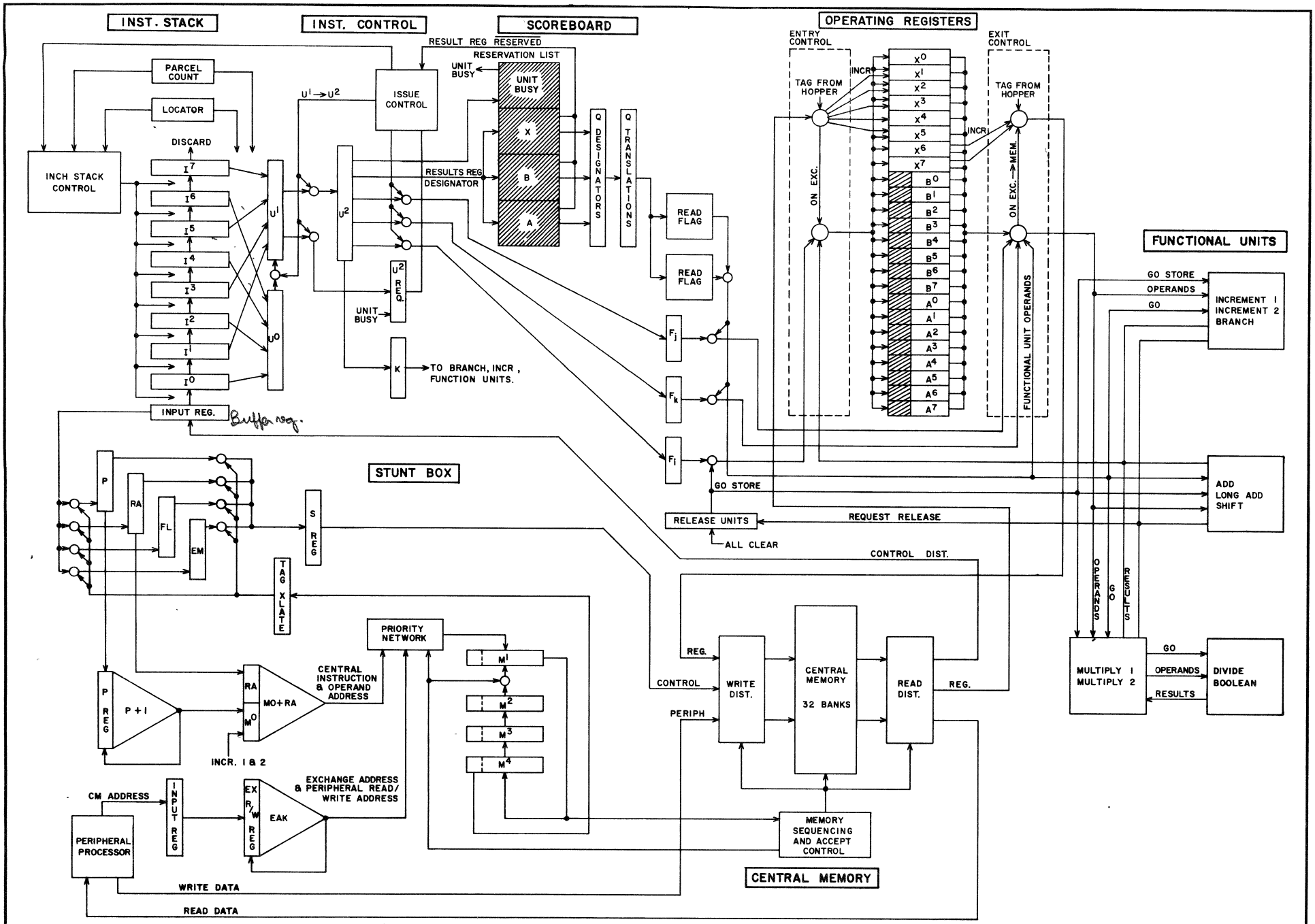
Figure 8. Coaxial Cable Driver

CENTRAL PROCESSOR
CONTENTS

1	Block Diagram
3	Chassis Layout
4.1	Chassis Layout
4.2	Instruction Stack $I_0 \rightarrow I_7$ (Test point tabulations)
5	Organization
7	Inch Counter
9	Parcel Counter & Selectors
11	Parcel Extraction
13	L Counter
15	Instruction Control Block Diagram
16	U^0, U^1, U^2, K, R (Test Points)
17	U^0, U^1, U^2, K, R Registers
19	Issue
21	Unit Request, Unit Busy & Issue, Serials 1-7
22.1	Unit Request, Unit Busy & Issue, Serials 8 and up
23	Instruction Go Control Part I, Serials 1-7
24.01	Instruction Go Control Part I, Serials 8 and up
24.1	Go Control Part II
25	Result Register Reserved & Issue
27	Instruction Issue Timing
28	Scoreboard
29	Overall Block Diagram
30	Designators
31	Designators
32	Placing Scoreboard Reservations
33	Block Diagram, Set F, Q, XBA
35	Set F, Serials 1-7
36.1	Set F, Serials 8 and up
37	Set F (Long Add, F_j)
39	Set Q, Serials 1-7
40.1	Set Q, Serials 8 and up
41	Set Q (Long Add) (X_7 to Q_j)
43	Set XBA, Serials 1-7
44.1	Set XBA, Serials 8 and up
45	Set XBA, Long Add (16) to X_4
47	Block Diagrams, Request Release Circuits
49	All Clear Block Diagram
51	All Clear
53	All Clear X^1
55	Request Release (Boolean) to Go Store (Boolean)
57	Request Release to Clear XBA
59	Request Release (Boolean) to Clear XBA (X7)
61	Request Release to Set Read Flags
63	Request Release (Boolean) to Go Read (Long Add)
64	D Registers (Test points)
65	Operating Registers, Entry and Exit Control Block Diagram
66	X, B & A Op. Registers Bit Locations and Test Points
67	X, B & A Bit Locations
68.0	Go Read, Go Store, MEM D, D X, X MEM (Test Points)
68.1	Entry Control, Chassis 7
68.3	Entry Control, Chassis 8
68.5	Exit Control, Chassis 7 A & B
68.7	Exit Control, Chassis 7
68.9	Exit Control, Chassis 8
69	Register Entry Control Transmit Boolean to X7
70	Go F.U., Transmit, Request Release (Test Points)
71	Register Exit Control, Go Read (X7) to Long Add
73	Data Trunks
74	Data Trunk Table, Part 1
74.1	Data Trunk Table, Part 2
74.2	Stunt Box
75	Block Diagram
77	Exchange/Read/Write Address
79	Exchange Address Counter, Serials 1-7
80.01	Exchange Address Counter, Serials 8 and up
80.1	M^0 + RA Adder
80.3	$M^0 \geq$ FL Address Range Test

CENTRAL PROCESSOR CONTENTS (CONTINUED)

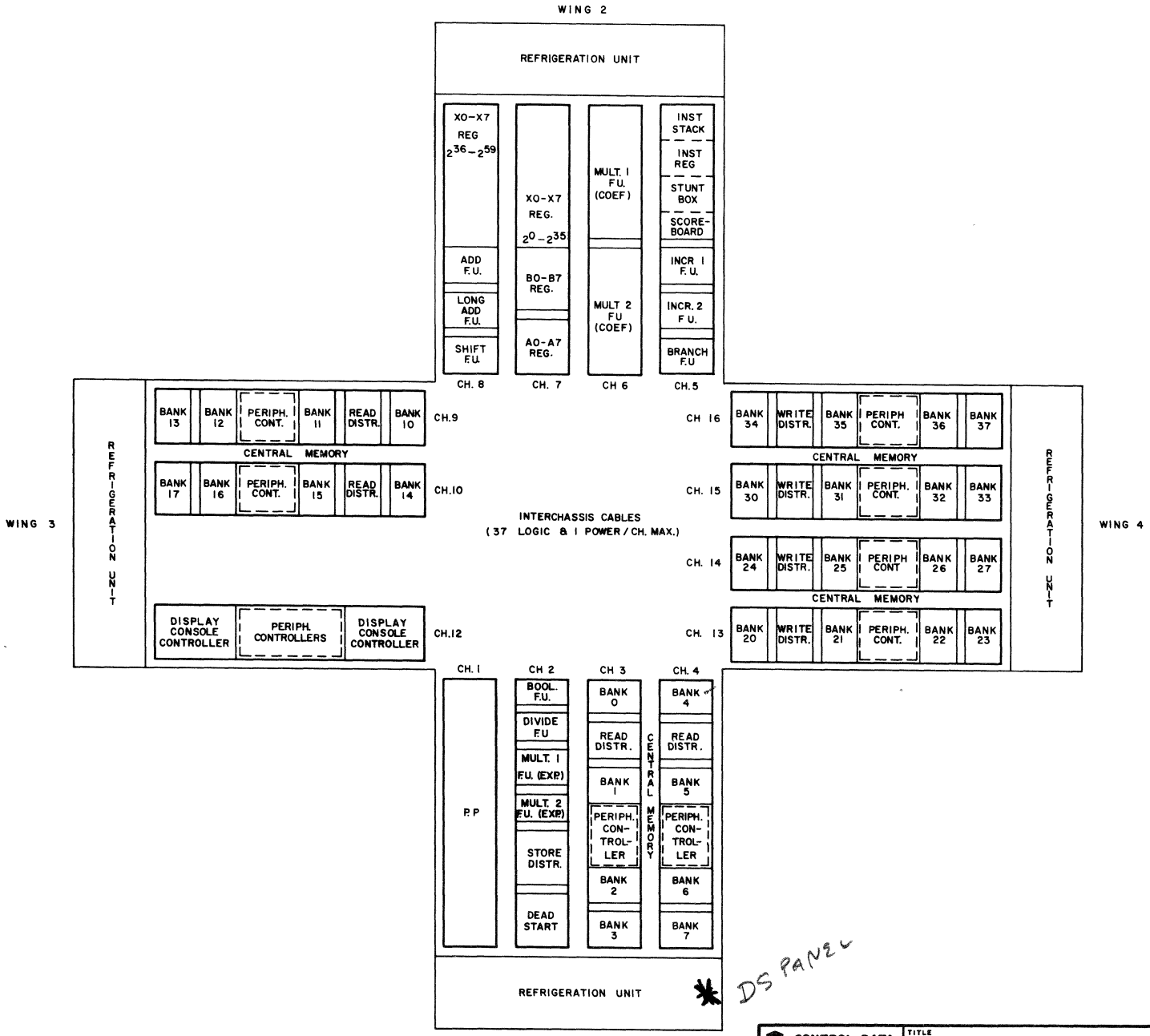
81	Address Issue Priority, Serials 1-7	89	Exchange Jump Control, Serials 1-7
82.1	Address Issue Priority, Serials 8 and up	90.1	Exchange Jump Control, Serials 8 and up
82.2	Hopper M_1 - M_4 Registers (Test Points)	91	Central Processor P, RA, FL & S Registers
83	Hopper Address Distribution	93	Exit Mode; Multiply, Divide, Boolean Block Diagram
85	Exchange Tag Counter (ETK), Serials 1-7	95	X_j Exit Mode Test Multiply, Divide, Boolean
86.01	Exchange Tag Counter (ETK), Serials 8 and up	97	X_k Exit Mode Test Multiply, Divide, Boolean
86.1	Hopper Tag Distribution 1	99	Exit Mode Tests Transmit to Chassis 5
86.3	Hopper Tag Distribution 2	101	Exit Mode Fan In, Branch Range & Indefinite Tests
87	Tag Distribution and Tag Translator		
88.1	All Quiet Network	103	Exit Mode Control, Serials 1-7
		105	Exit Mode Control, Serials 8 and up



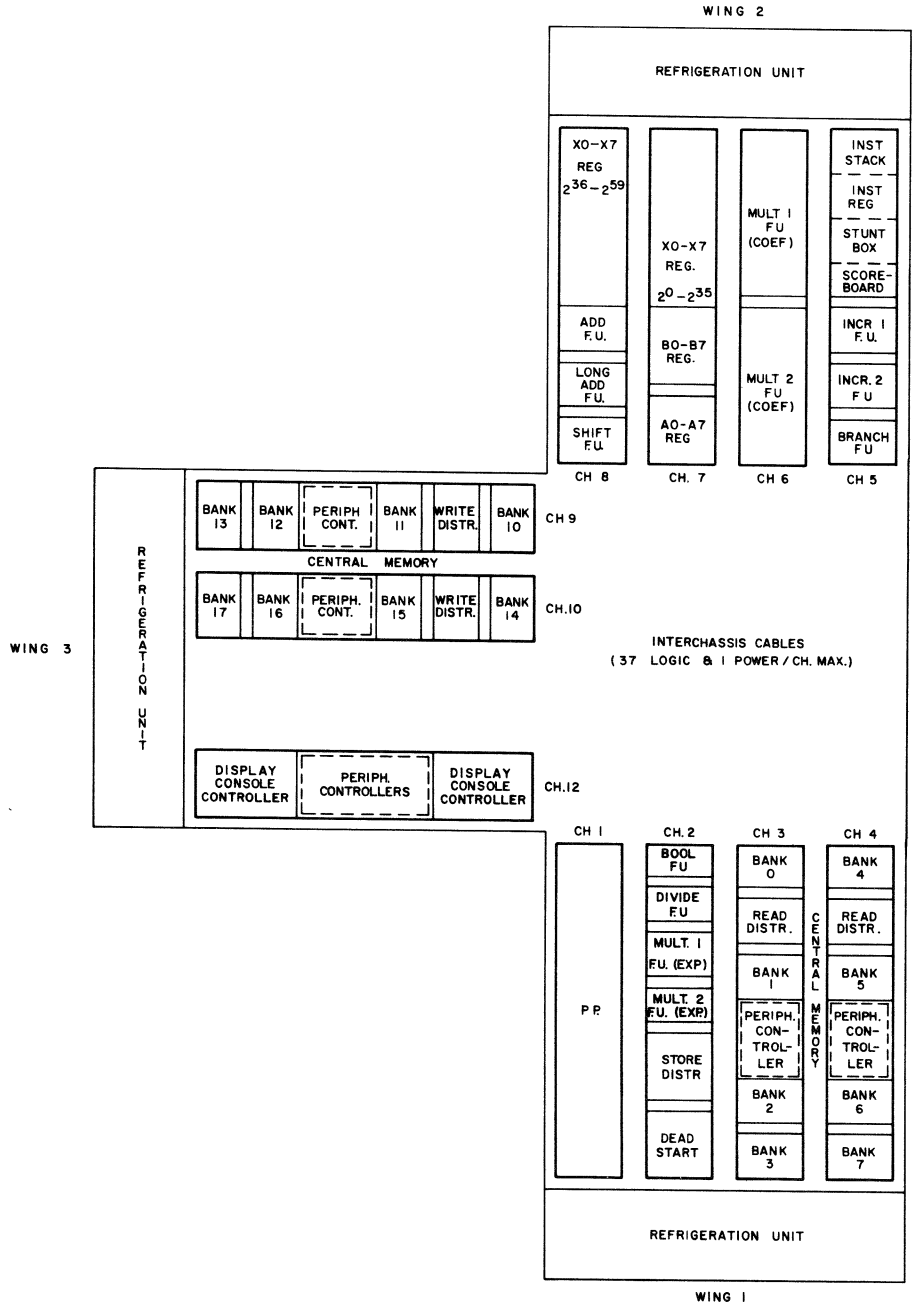
CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE
CENTRAL PROCESSOR
BLOCK DIAGRAM

PRODUCT 6601/04
SIZE C | DRAWING NO. 60119300 | REV K
SHEET 46 | 1



CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR CHASSIS LAYOUT	PRODUCT: 6601	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: K
			SHEET: 47 3

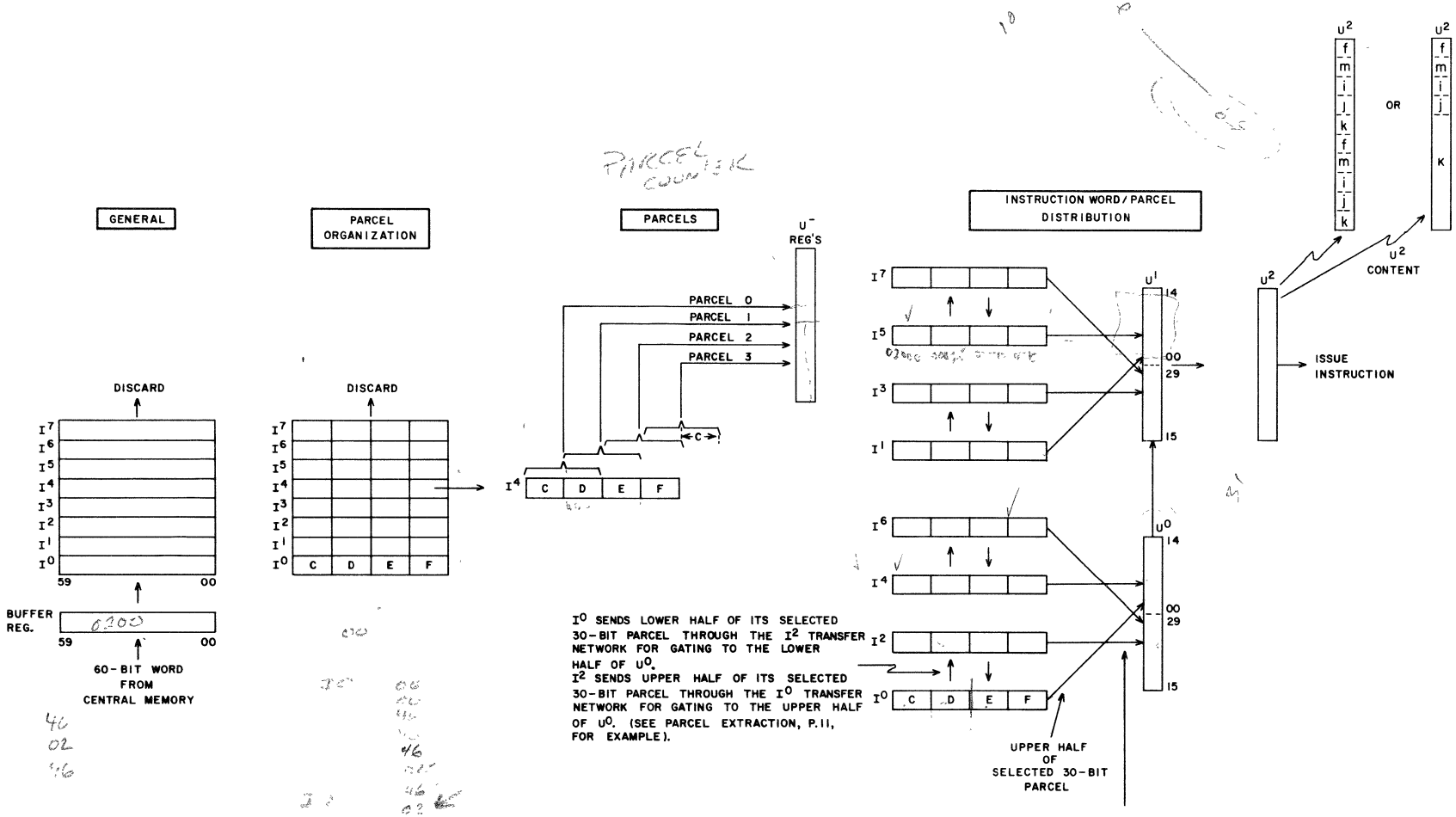


CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR CHASSIS LAYOUT	PRODUCT: 6604	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: K
			SHEET: 236 4.1

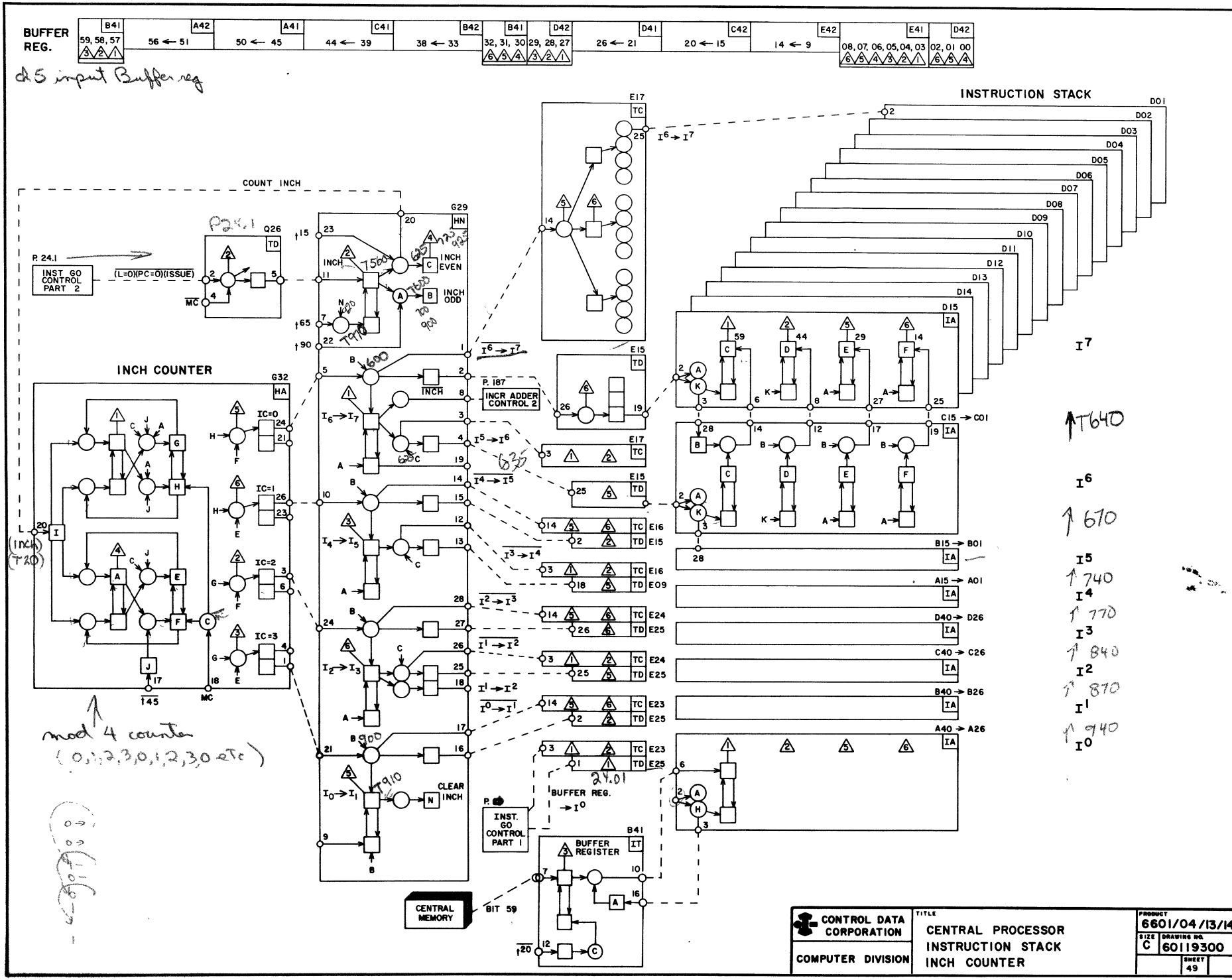
Bit	Module-TP		Module-TP		Module-TP		Module-TP		Bit	Module	TP	Module	TP	Module	TP	Module	TP	Bit
59	5A40	1	5B40	1	5C40	1	5D40	1	59	5A15	1	5B15	1	5C15	1	5D15	1	59
58	39	1	39	1	39	1	39	1	58	14	1	14	1	14	1	14	1	58
57	38	1	38	1	38	1	38	1	57	13	1	13	1	13	1	13	1	57
56	37	1	37	1	37	1	37	1	56	12	1	12	1	12	1	12	1	56
55	36	1	36	1	36	1	36	1	55	11	1	11	1	11	1	11	1	55
54	35	1	35	1	35	1	35	1	54	10	1	10	1	10	1	10	1	54
53	34	1	34	1	34	1	34	1	53	9	1	9	1	9	1	9	1	53
52	33	1	33	1	33	1	33	1	52	8	1	8	1	8	1	8	1	52
51	32	1	32	1	32	1	32	1	51	7	1	7	1	7	1	7	1	51
50	31	1	31	1	31	1	31	1	50	6	1	6	1	6	1	6	1	50
49	30	1	30	1	30	1	30	1	49	5	1	5	1	5	1	5	1	49
48	29	1	29	1	29	1	29	1	48	4	1	4	1	4	1	4	1	48
47	28	1	28	1	28	1	28	1	47	3	1	3	1	3	1	3	1	47
46	27	1	27	1	27	1	27	1	46	2	1	2	1	2	1	2	1	46
45	26	1	26	1	26	1	26	1	45	1	1	1	1	1	1	1	1	45
44	5A40	2	5B40	2	5C40	2	5D40	2	44	5A15	2	5B15	2	5C15	2	5D15	2	44
43	39	2	39	2	39	2	39	2	43	14	2	14	2	14	2	14	2	43
42	38	2	38	2	38	2	38	2	42	13	2	13	2	13	2	13	2	42
41	37	2	37	2	37	2	37	2	41	12	2	12	2	12	2	12	2	41
40	36	2	36	2	36	2	36	2	40	11	2	11	2	11	2	11	2	40
39	35	2	35	2	35	2	35	2	39	10	2	10	2	10	2	10	2	39
38	34	2	34	2	34	2	34	2	38	9	2	9	2	9	2	9	2	38
37	33	2	33	2	33	2	33	2	37	8	2	8	2	8	2	8	2	37
36	32	2	32	2	32	2	32	2	36	7	2	7	2	7	2	7	2	36
35	31	2	31	2	31	2	31	2	35	6	2	6	2	6	2	6	2	35
34	30	2	30	2	30	2	30	2	34	5	2	5	2	5	2	5	2	34
33	29	2	29	2	29	2	29	2	33	4	2	4	2	4	2	4	2	33
32	28	2	28	2	28	2	28	2	32	3	2	3	2	3	2	3	2	32
31	27	2	27	2	27	2	27	2	31	2	2	2	2	2	2	2	2	31
30	26	2	26	2	26	2	26	2	30	1	2	1	2	1	2	1	2	30
29	5A40	5	5B40	5	5C40	5	5D40	5	29	5A15	5	5B15	5	5C15	5	5D15	5	29
28	39	5	39	5	39	5	39	5	28	14	5	14	5	14	5	14	5	28
27	38	5	38	5	38	5	38	5	27	13	5	13	5	13	5	13	5	27
26	37	5	37	5	37	5	37	5	26	12	5	12	5	12	5	12	5	26
25	36	5	36	5	36	5	36	5	25	11	5	11	5	11	5	11	5	25
24	35	5	35	5	35	5	35	5	24	10	5	10	5	10	5	10	5	24
23	34	5	34	5	34	5	34	5	23	9	5	9	5	9	5	9	5	23
22	33	5	33	5	33	5	33	5	22	8	5	8	5	8	5	8	5	22
21	32	5	32	5	32	5	32	5	21	7	5	7	5	7	5	7	5	21
20	31	5	31	5	31	5	31	5	20	6	5	6	5	6	5	6	5	20
19	30	5	30	5	30	5	30	5	19	5	5	5	5	5	5	5	5	19
18	29	5	29	5	29	5	29	5	18	4	5	4	5	4	5	4	5	18
17	28	5	28	5	28	5	28	5	17	3	5	3	5	3	5	3	5	17
16	27	5	27	5	27	5	27	5	16	2	5	2	5	2	5	2	5	16
15	26	5	26	5	26	5	26	5	15	1	5	1	5	1	5	1	5	15
14	5A40	6	5B40	6	5C40	6	5D40	6	14	5A15	6	5B15	6	5C15	6	5D15	6	14
13	39	6	39	6	39	6	39	6	13	14	6	14	6	14	6	14	6	13
12	38	6	38	6	38	6	38	6	12	13	6	13	6	13	6	13	6	12
11	37	6	37	6	37	6	37	6	11	12	6	12	6	12	6	12	6	11
10	36	6	36	6	36	6	36	6	10	11	6	11	6	11	6	11	6	10
9	35	6	35	6	35	6	35	6	9	10	6	10	6	10	6	10	6	9
8	34	6	34	6	34	6	34	6	8	9	6	9	6	9	6	9	6	8
7	33	6	33	6	33	6	33	6	7	8	6	8	6	8	6	8	6	7
6	32	6	32	6	32	6	32	6	6	7	6	7	6	7	6	7	6	6
5	31	6	31	6	31	6	31	6	5	6	6	6	6	6	6	6	6	5
4	30	6	30	6	30	6	30	6	4	5	6	5	6	5	6	5	6	4
3	29	6	29	6	29	6	29	6	3	4	6	4	6	4	6	4	6	3
2	28	6	28	6	28	6	28	6	2	3	6	3	6	3	6	3	6	2
1	27	6	27	6	27	6	27	6	1	2	6	2	6	2	6	2	6	1
0	26	6	26	6	26	6	26	6	0	1	6	1	6	1	6	1	6	0
	I0		I1		I2		I3			I4		I5		I6		I7		

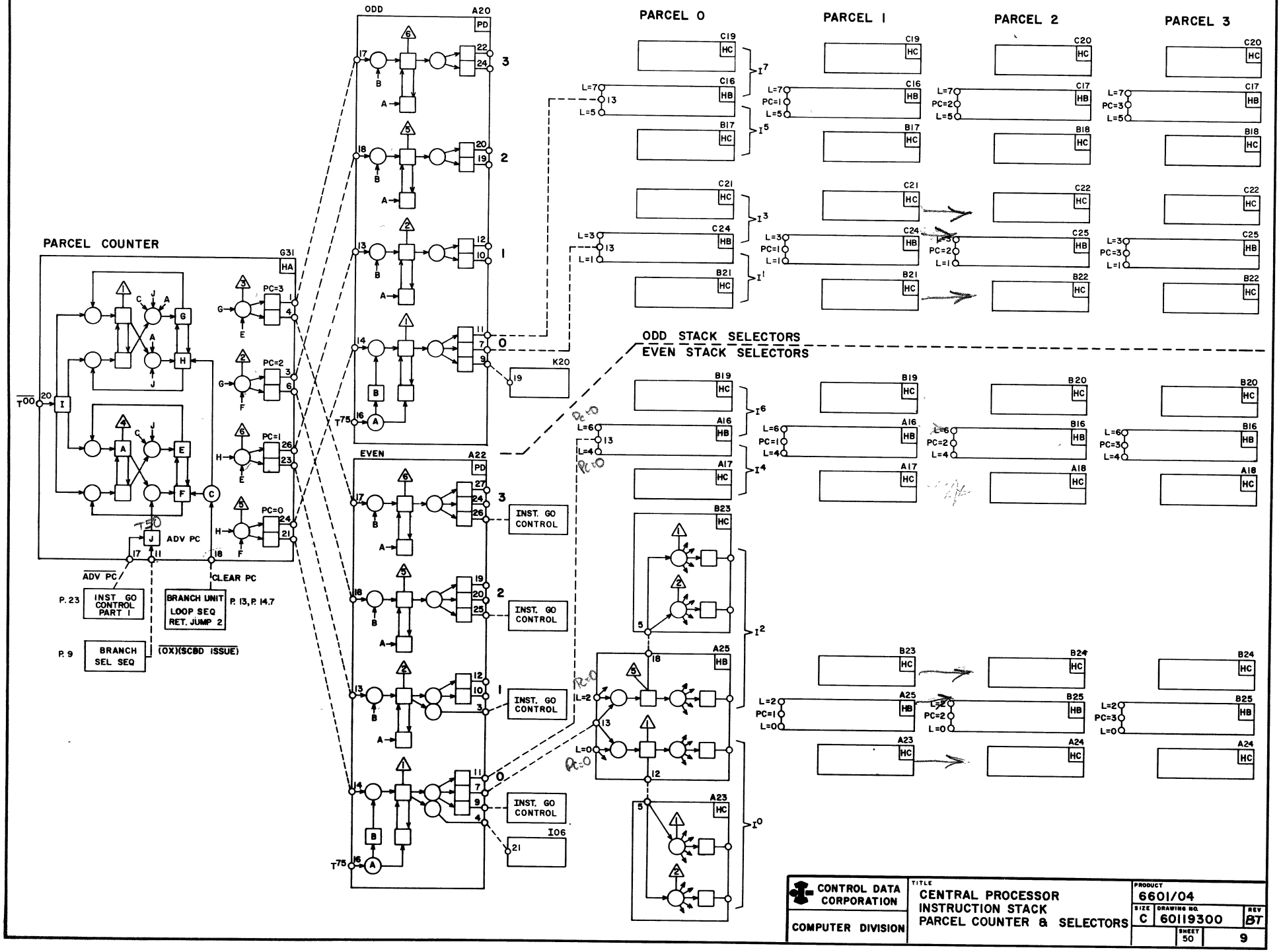
6601/04 CENTRAL PROCESSOR
BIT LOCATIONS & TEST POINTS
$I_0 \rightarrow I_7$ REGISTERS
PUB. NO. 60119300

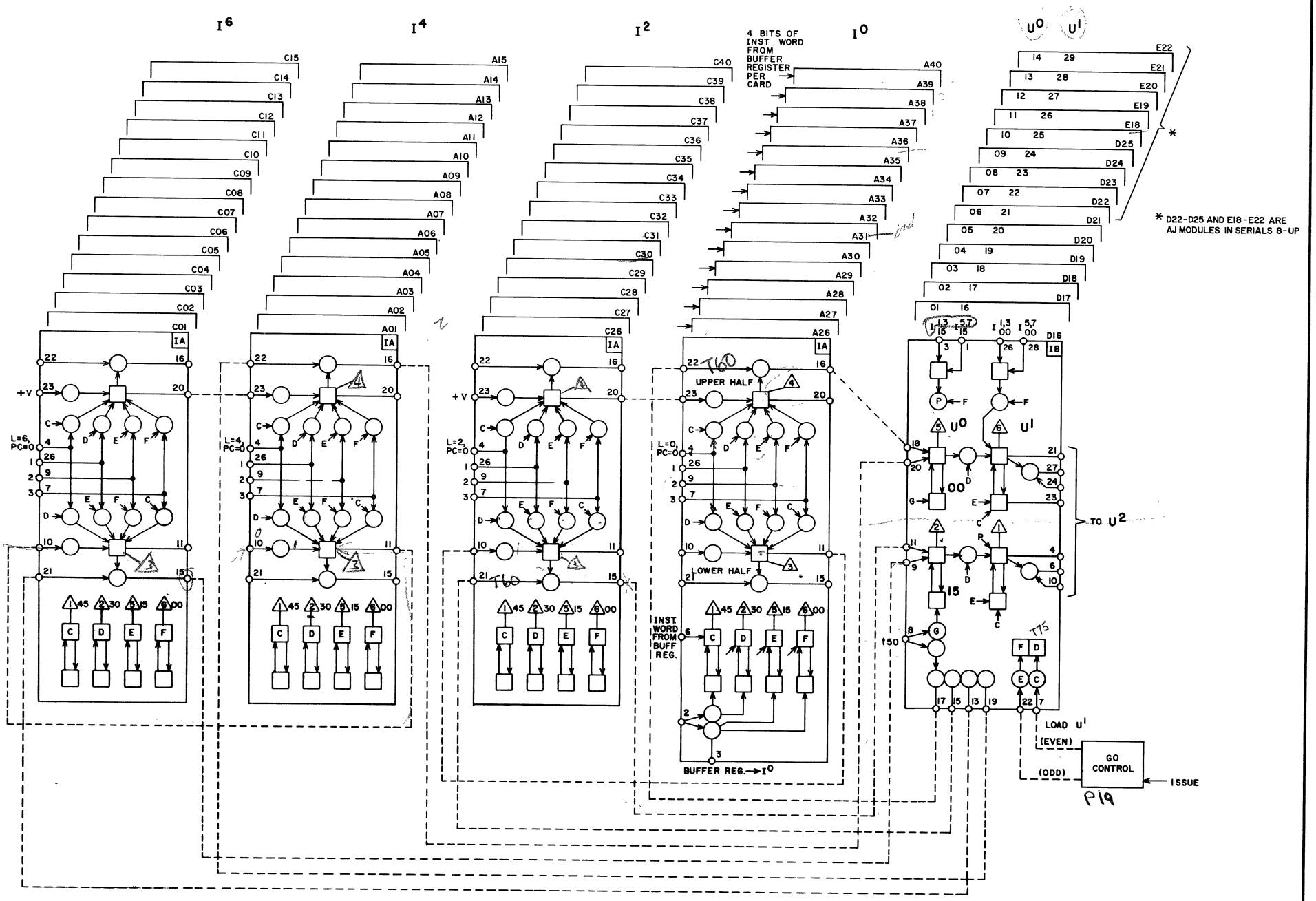
REV. K 4.2



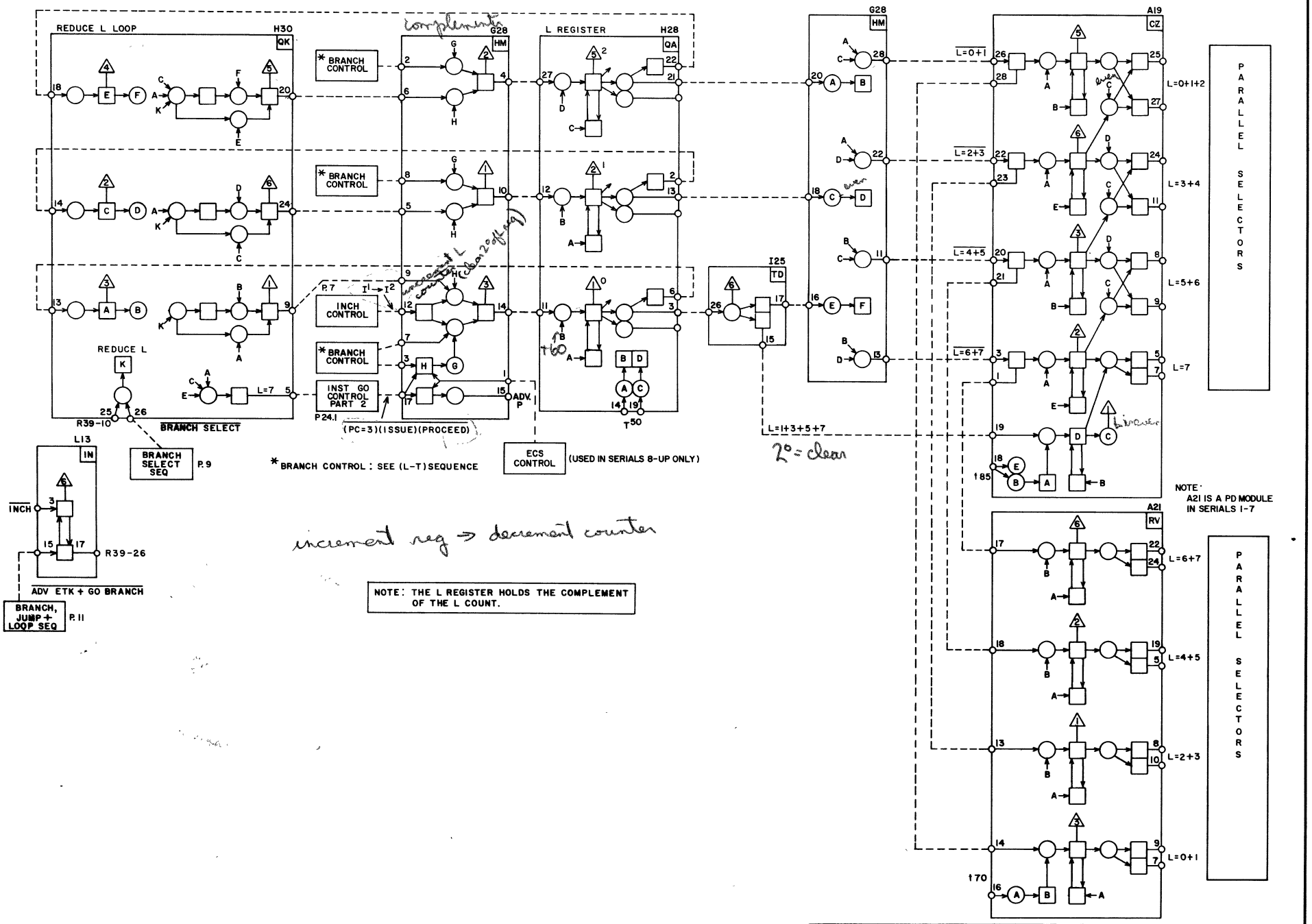
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR INSTRUCTION STACK ORGANIZATION	PRODUCT: 6601/04	
COMPUTER DIVISION		SIZE: C · DRAWING NO.: 60119300 · REV: K	
		SHEET: 48	5


BUFFER REG.
B41 59, 58, 57
A42 56 ← 51
A41 50 ← 45
C41 44 ← 39
B42 38 ← 33
B41 32, 31, 30
D42 29, 28, 27
D41 26 ← 21
C42 20 ← 15
E42 14 ← 9
E41 08, 07, 06, 05, 04, 03
D42 02, 01 00
25 input Buffer reg
INSTRUCTION STACK
D01
D02
D03
D04
D05
D06
D07
D08
D09
D10
D11
D12
D13
D14
D15
IA
I7
T640
I6
670
I5
740
I4
770
I3
840
I2
870
I1
940
I0
COUNT INCH
P. 24.1
INST. GO CONTROL PART 2
(L=0)(PC=0)(ISSUE)
Q26 TD
MC
G29 HN
INCH
INCH EVEN
INCH ODD
T560
T640
T910
INCH COUNTER
G32 HA
IC=0
IC=1
IC=2
IC=3
INCH
T20
T45
MC
mod 4 counter
(0,1,2,3,0,1,2,3,0 etc)
P. 187
INCR ADDER CONTROL 2
I6 → I7
I5 → I6
I4 → I5
I3 → I4
I2 → I3
I1 → I2
I0 → I1
E17 TC
E15 TD
E16 TC
E09 TD
E24 TC
E25 TD
E23 TC
E25 TD
T600
T900
T910
CLEAR INCH
CENTRAL MEMORY
BIT 59
P.
INST. GO CONTROL PART 1
BUFFER REG. → I0
24.01
B41 IT
BUFFER REGISTER
T20
C15 → C01
B15 → B01
A15 → A01
D40 → D26
C40 → C26
B40 → B26
A40 → A26
CONTROL DATA CORPORATION
COMPUTER DIVISION
TITLE
CENTRAL PROCESSOR
INSTRUCTION STACK
INCH COUNTER
PRODUCT 6601/04/13/14
SIZE C
DRAWING NO. 60119300
REV BT
SHEET 49
7

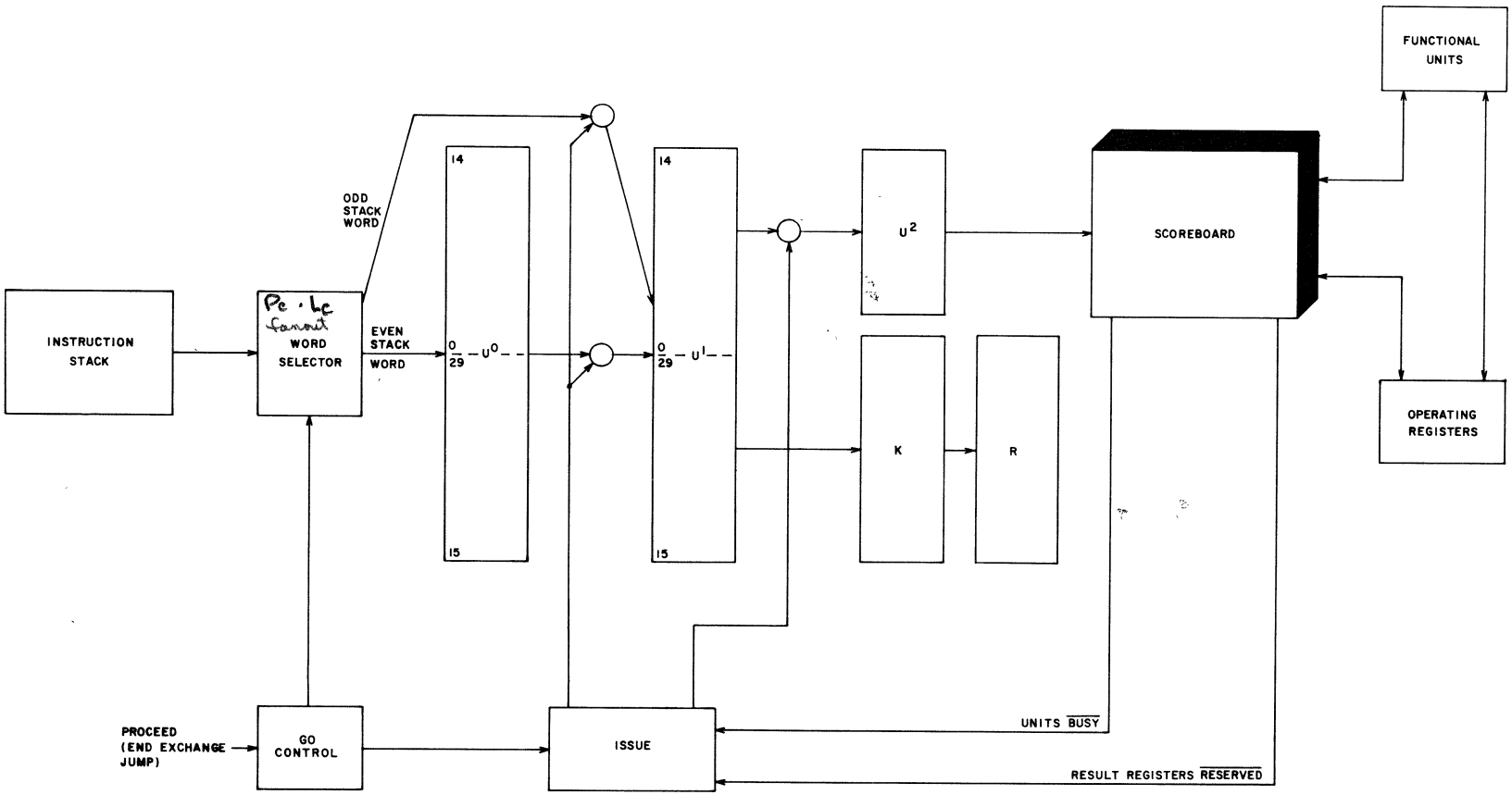

PARCEL COUNTER
G31
HA
PC=3
PC=2
PC=1
PC=0
ADV PC
CLEAR PC
INST GO CONTROL PART 1
P. 23
BRANCH UNIT LOOP SEQ RET. JUMP 2
P. 13, P. 14.7
BRANCH SEL SEQ
P. 9
(OX)(SCBD ISSUE)
ODD
A20
PD
EVEN
A22
PD
K20
I06
INST. GO CONTROL
INST. GO CONTROL
INST. GO CONTROL
INST. GO CONTROL
PARCEL 0
PARCEL 1
PARCEL 2
PARCEL 3
ODD STACK SELECTORS
EVEN STACK SELECTORS
C19
C16
B17
C21
C24
B21
C20
C17
B18
C22
C25
B22
B19
A16
A17
B20
B16
A18
B23
A25
A23
B24
B25
A24
CONTROL DATA CORPORATION
COMPUTER DIVISION
CENTRAL PROCESSOR
INSTRUCTION STACK
PARCEL COUNTER & SELECTORS
6601/04
C
60119300
BT
50
9



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR INSTRUCTION STACK PARCEL EXTRACTION	PRODUCT 6601/04/13/14	
		SIZE C	DRAWING NO. 60119300 REV BT
			SHEET 51 11



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR INSTRUCTION STACK L COUNTER	PRODUCT 6601/04/13/14	
		SIZE C	DRAWING NO. 60119300 REV 8T
		SHEET 52	13



CONTROL DATA CORPORATION	TITLE	PRODUCT	
COMPUTER DIVISION	CENTRAL PROCESSOR INSTRUCTION CONTROL BLOCK DIAGRAM	6601/04	
		SIZE C	DRAWING NO. 60119300 REV K
			SHEET 53 15

U_o

Module-TP

14	5E22	5	
13	21	5	
12	20	5	
11	19	5	
10	18	5	
9	5D25	5	
8	24	5	
7	23	5	
6	22	5	
5	21	5	
4	20	5	
3	19	5	
2	18	5	
1	17	5	
0	16	5	

U_1

Module-TP

5E22	6	14	
21	6	13	
20	6	12	
19	6	11	
18	6	10	
5D25	6	9	
24	6	8	
23	6	7	
22	6	6	
21	6	5	
20	6	4	
19	6	3	
18	6	2	
17	6	1	
16	6	0	

U_2

Module-TP

fm	2	5E32	5	
	1		4	
	0		6	
	2		2	
	1		1	
	0		3	
i	2	5E14	3	
	1		6	
	0		1	
j	2	5E13	6	
	1		4	
	0		3	
k	2	5E12		
	1			
	0			

Module-Tp

29	5E22	52	
28	21	52	
27	20	52	
26	19	52	
25	18	52	
24	5D25	52	
23	24	52	
22	23	52	
21	22	52	
20	21	52	
19	20	52	
18	19	52	
17	18	52	
16	17	52	
15	16	52	

U_o

Module-TP

5E22	61	29	
	61	28	
	61	27	
	61	26	
	61	25	
5D25	61	24	
	61	23	
	61	22	
	61	21	
	61	20	
	61	19	
	61	18	
	61	17	
	61	16	
	X1	15	

U_1

Module-TP

17	5M30	6	
16		5	
15		4	
14		3	
13		2	
12		1	
11	5M29	6	
10		5	
9		4	
8		3	
7		2	
6		1	
5	5M28	6	
4		5	
3		4	
2		3	
1		2	
0		1	

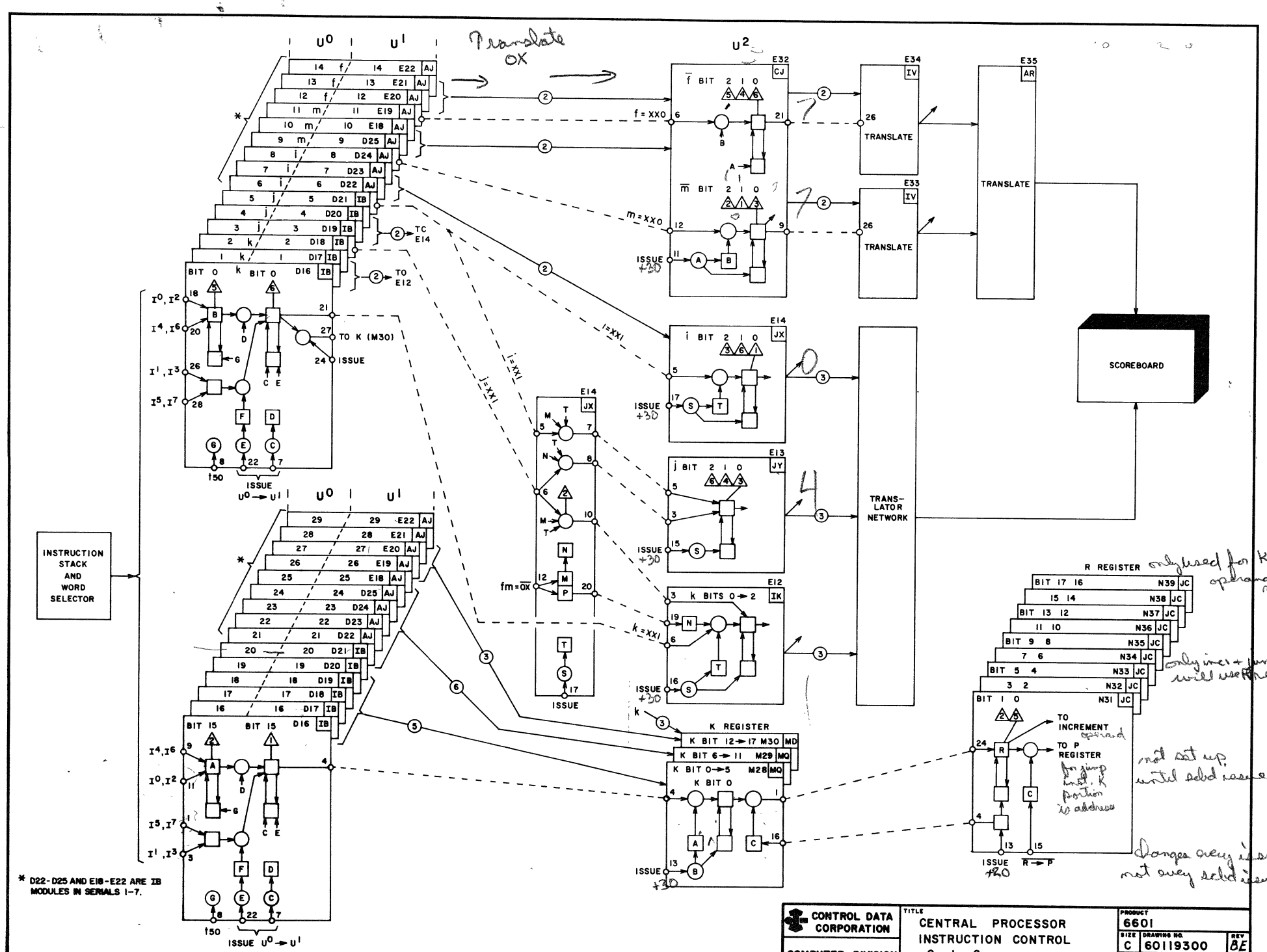
K

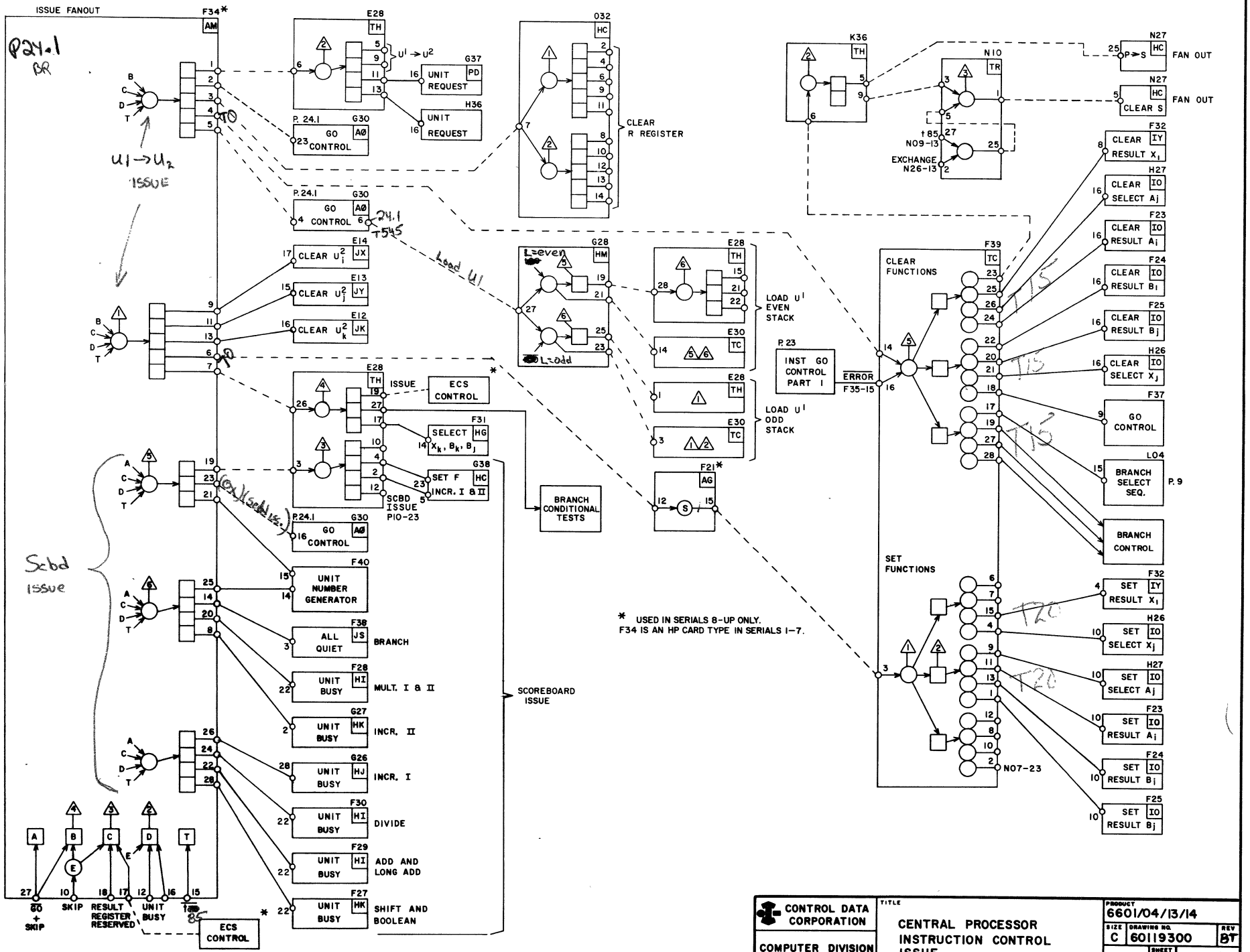
Module-TP

5N39	2	17	
	5	16	
38	2	15	
	5	14	
37	2	13	
	5	12	
36	2	11	
	5	10	
35	2	9	
	5	8	
34	2	7	
	5	6	
33	2	5	
	5	4	
32	2	3	
	5	2	
31	2	1	
	5	0	

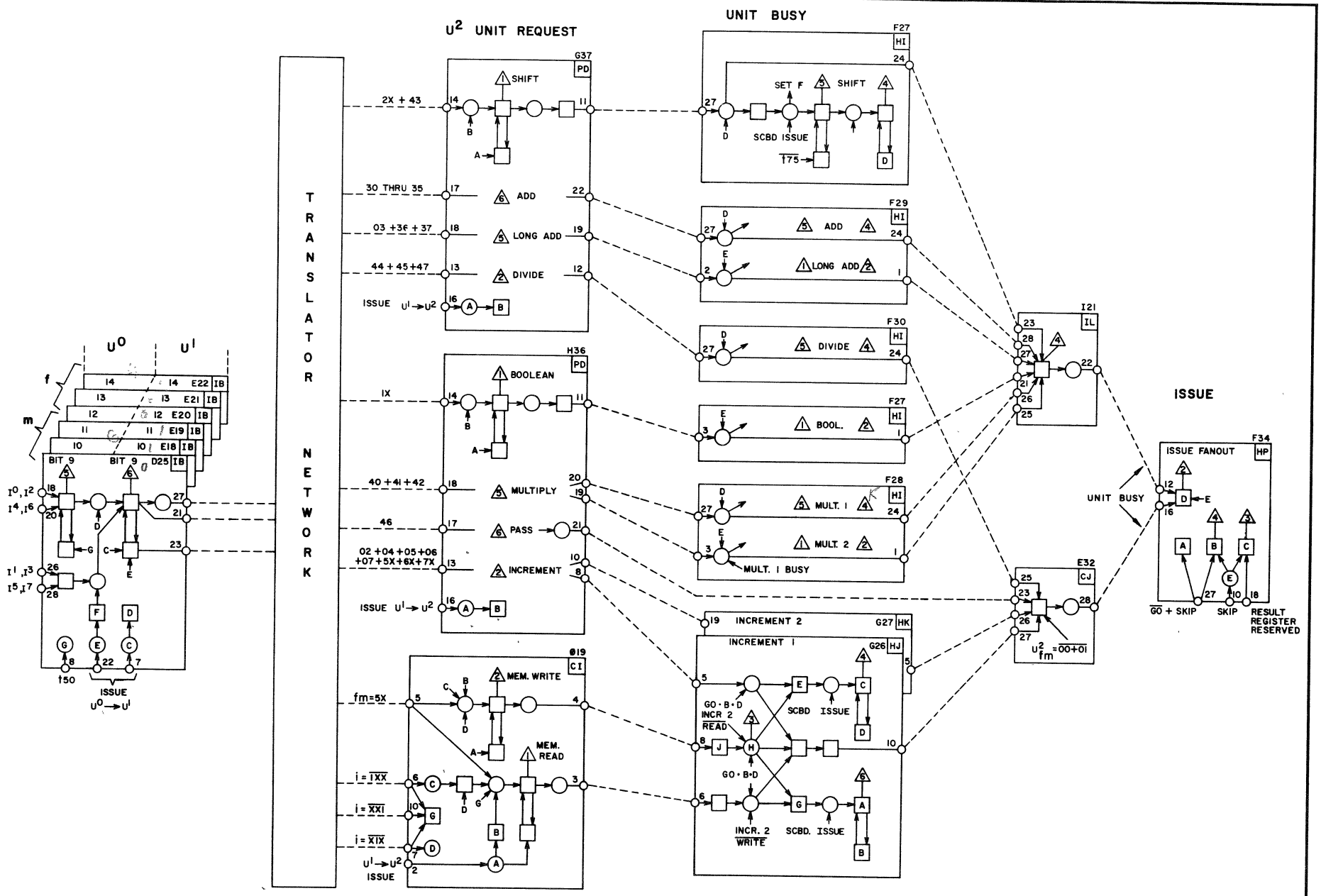
R

6601 Bit Locations & TP's.
U^o, U^1, U^2 Registers
K Register
R Register

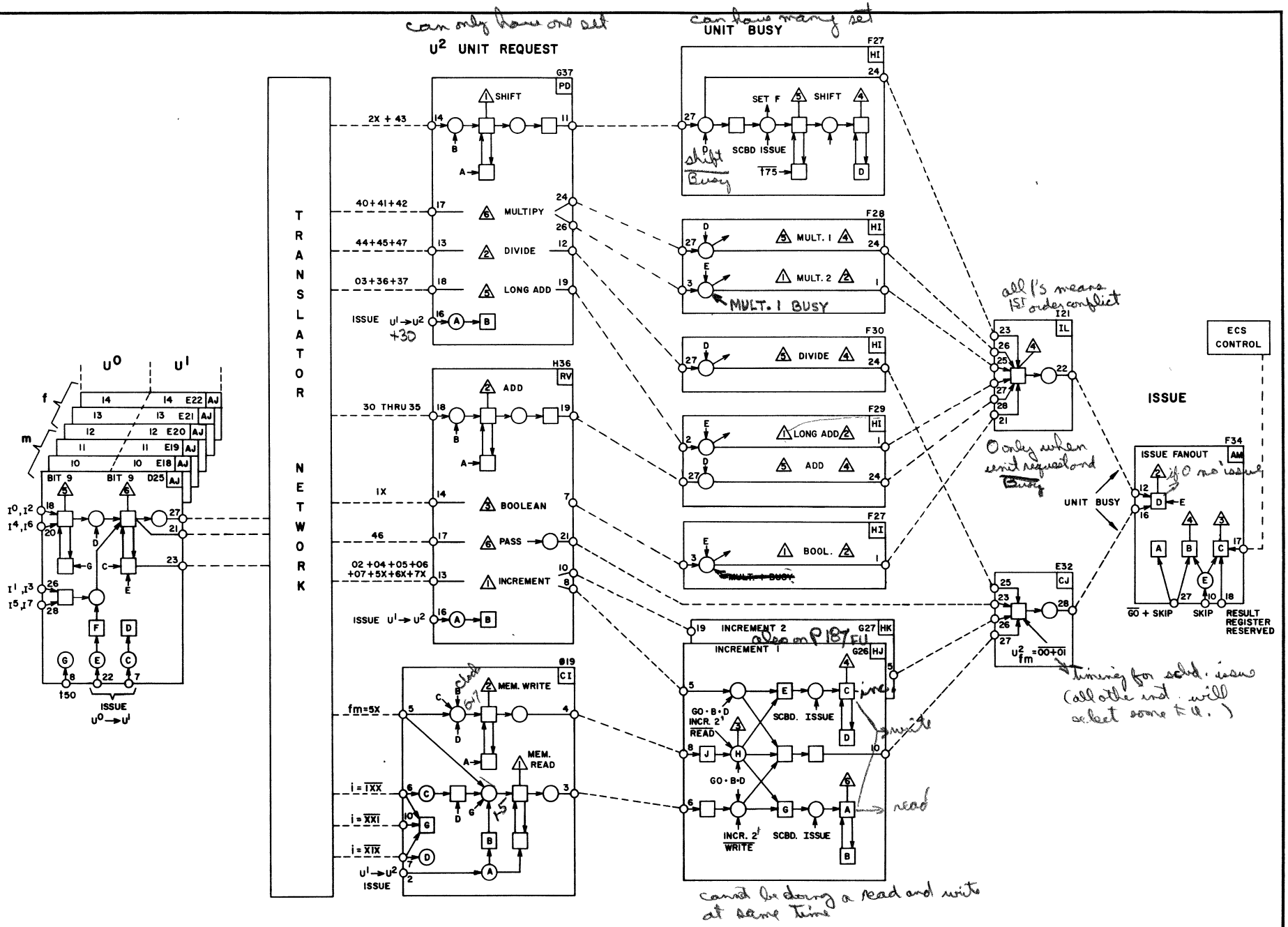

u0 u1
u2
Translate OX
INSTRUCTION STACK AND WORD SELECTOR
SCOREBOARD
TRANSLATE
TRANS-LATOR NETWORK
ISSUE
TO K (M30)
TO E12
TC E14
K REGISTER
R REGISTER
TO INCREMENT
TO P REGISTER
ISSUE U0 → U1
R → P
f = XX0
m = XX0
i = XX1
j = XX1
k = XX1
fm = OX
* D22-D25 AND E18-E22 ARE IB MODULES IN SERIALS 1-7.
CONTROL DATA CORPORATION
COMPUTER DIVISION
CENTRAL PROCESSOR
INSTRUCTION CONTROL
U0, U1, U2, K, R REGISTERS
6601
C
60119300
8E
54
17



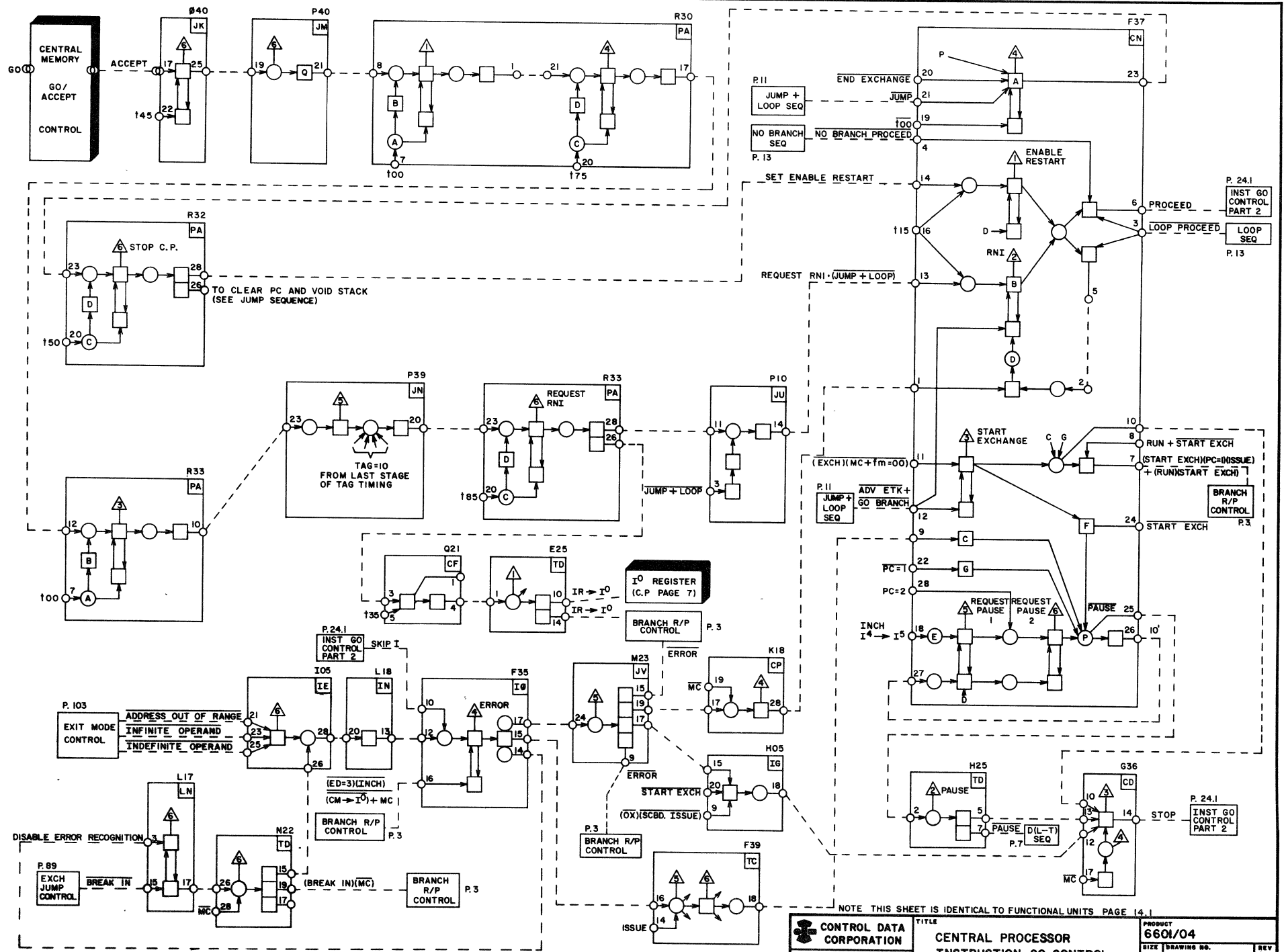
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR INSTRUCTION CONTROL ISSUE	PRODUCT: 6601/04/13/14	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: BT
			SHEET: 55 19

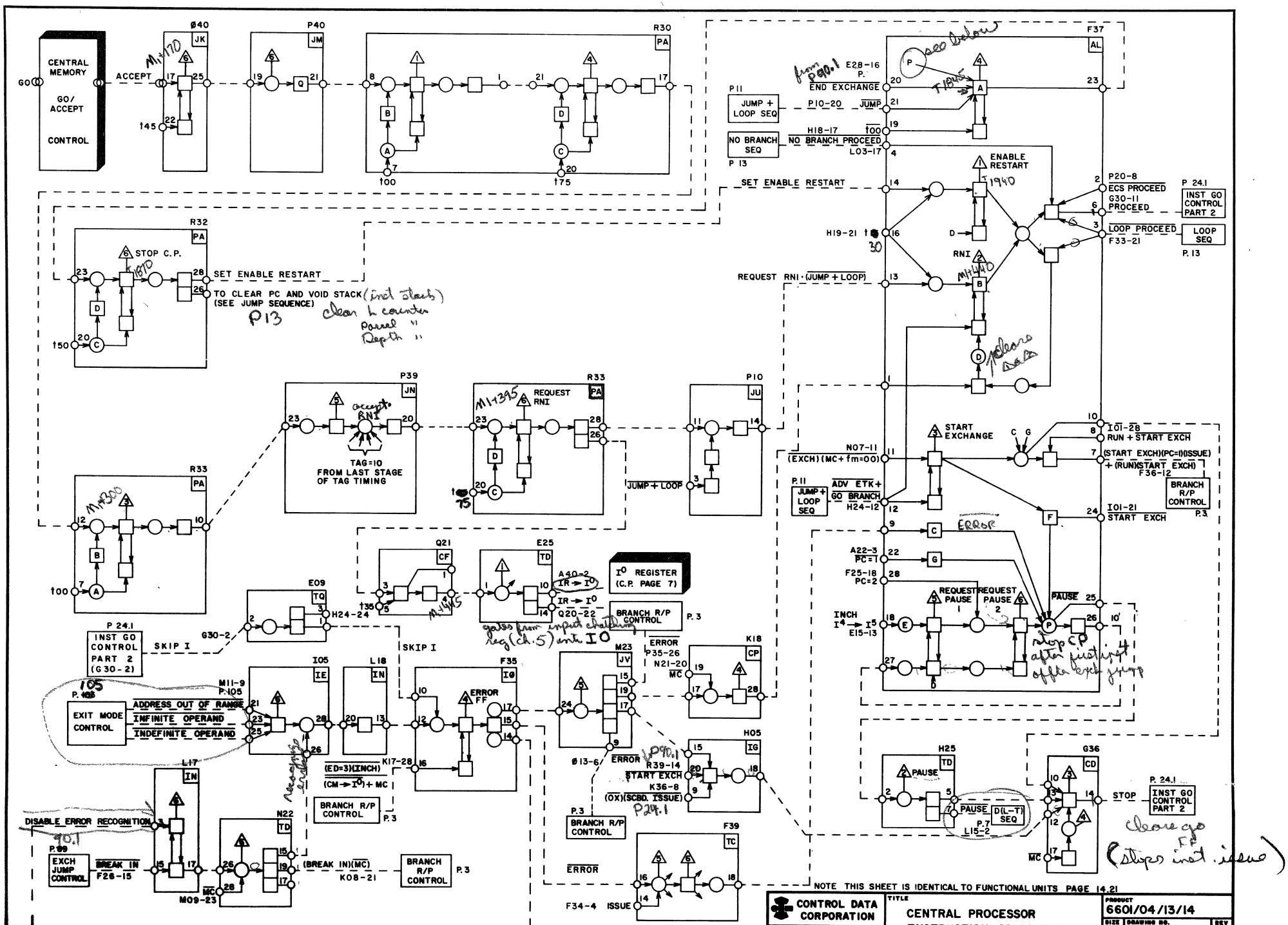


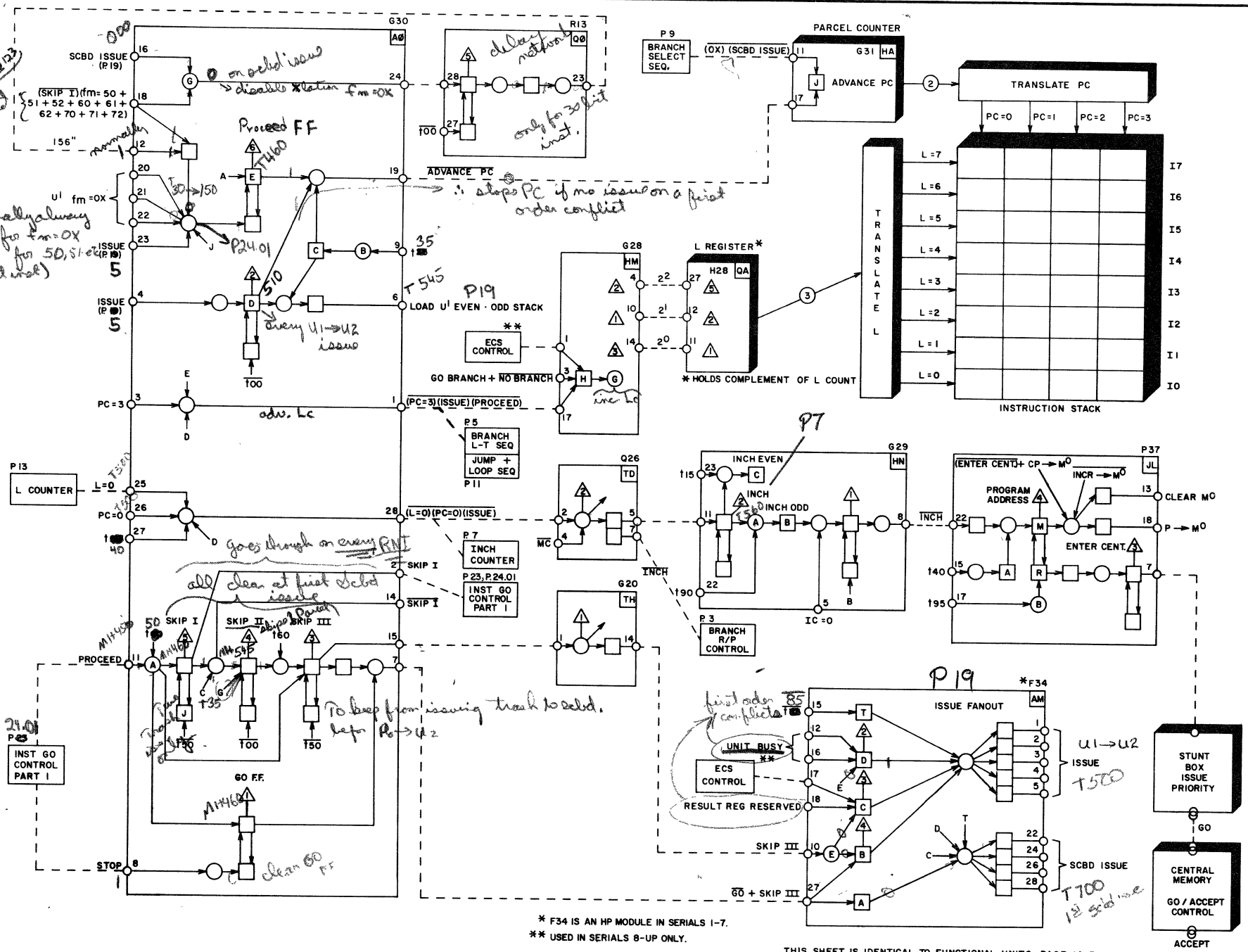
CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR INSTRUCTION CONTROL UNIT REQUEST, UNIT BUSY & ISSUE, SERIALS 1-7	PRODUCT 6601/04	
		SIZE C	DRAWING NO. 60119300 REV M
			SHEET 56 21



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR INSTRUCTION CONTROL UNIT REQUEST, UNIT BUSY & ISSUE. SERIALS 8-UP	PRODUCT 6601/04/13/14 SIZE C · DRAWING NO. 60119300 · REV 8T SHEET 251 · 22.1	


CENTRAL MEMORY
GO/ ACCEPT
CONTROL
GO
ACCEPT
Ø40
JK
P40
JM
R30
PA
t45
t00
t75
F37
CN
END EXCHANGE
JUMP
NO BRANCH PROCEED
P.11
JUMP + LOOP SEQ
NO BRANCH SEQ
P. 13
ENABLE RESTART
SET ENABLE RESTART
P. 24.1
INST GO CONTROL PART 2
PROCEED
LOOP PROCEED
LOOP SEQ
P. 13
R32
STOP C.P.
TO CLEAR PC AND VOID STACK
(SEE JUMP SEQUENCE)
t50
t15
RNI
REQUEST RNI · (JUMP + LOOP)
P39
JN
TAG=10
FROM LAST STAGE
OF TAG TIMING
R33
REQUEST RNI
P10
JU
JUMP + LOOP
t85
t00
START EXCHANGE
(EXCH)(MC+fm=00)
RUN + START EXCH
(START EXCH)(PC=1)(ISSUE) + (RUN)(START EXCH)
ADV ETK + GO BRANCH
BRANCH R/P CONTROL
P. 3
START EXCH
Q21
CF
E25
TD
IR → I⁰
I⁰ REGISTER
(C.P PAGE 7)
BRANCH R/P CONTROL
P. 3
t35
PC=1
PC=2
REQUEST PAUSE 1
REQUEST PAUSE 2
PAUSE
INCH
I⁴ → I⁵
P.24.1
INST GO CONTROL PART 2
SKIP I
I05
IE
L18
IN
F35
I@
M23
JV
K18
CP
ERROR
MC
P. 103
EXIT MODE CONTROL
ADDRESS OUT OF RANGE
INFINITE OPERAND
INDEFINITE OPERAND
(ED=3)(INCH)
(CM → I⁰) + MC
BRANCH R/P CONTROL
P. 3
H05
IG
START EXCH
(OX)(SCBD. ISSUE)
H25
TD
PAUSE
D(L-T) SEQ
P.7
G36
CD
STOP
P. 24.1
INST GO CONTROL PART 2
L17
LN
DISABLE ERROR RECOGNITION
P. 89
EXCH JUMP CONTROL
BREAK IN
N22
TD
(BREAK IN)(MC)
BRANCH R/P CONTROL
P. 3
F39
TC
ISSUE
NOTE THIS SHEET IS IDENTICAL TO FUNCTIONAL UNITS PAGE 14.1
CONTROL DATA CORPORATION
COMPUTER DIVISION
TITLE
CENTRAL PROCESSOR
INSTRUCTION GO CONTROL
PART 1, SERIALS 1-7
PRODUCT
6601/04
SIZE
C
DRAWING NO.
60119300
REV
AB
SHEET
57
PAGE
23


CENTRAL MEMORY
GO/ ACCEPT CONTROL
GO
ACCEPT
Ø40
JK
P40
JM
R30
PA
F37
AL
END EXCHANGE
JUMP + LOOP SEQ
P11
P10-20 JUMP
NO BRANCH SEQ
P 13
H18-17 NO BRANCH PROCEED
L03-17
ENABLE RESTART
SET ENABLE RESTART
H19-21
P20-8 ECS PROCEED
G30-11 PROCEED
LOOP PROCEED F33-21
P 24.1 INST GO CONTROL PART 2
LOOP SEQ
P. 13
REQUEST RNI · (JUMP + LOOP)
R32
STOP C.P.
SET ENABLE RESTART
TO CLEAR PC AND VOID STACK (SEE JUMP SEQUENCE)
P13
clear L counter
Parcel 11
Depth 11
P39
JN
TAG=10 FROM LAST STAGE OF TAG TIMING
R33
REQUEST RNI
P10
JU
JUMP + LOOP
START EXCHANGE
NO7-11 (EXCH)(MC+fm=00)
ADV ETK+ GO BRANCH H24-12
I01-28 RUN + START EXCH
(START EXCH)(PC=I)(ISSUE) + (RUN)(START EXCH) F36-12
BRANCH R/P CONTROL P.3
I01-21 START EXCH
ERROR
A22-3 PC=1
F25-18 PC=2
REQUEST PAUSE 1
REQUEST PAUSE 2
PAUSE
INCH I4 → I5 E15-13
stop CP after just put off exch jump
Q21
CF
E25
TD
A40-2 IR → I0
Q20-22 IR → I0
I0 REGISTER (C.P. PAGE 7)
BRANCH R/P CONTROL P. 3
E09
TQ
H24-24
G30-2
P 24.1 INST GO CONTROL PART 2 (G30-2)
SKIP I
gates from input chaining leg (ch. 5) into I0
I05
IE
L18
IN
F35
I0
M23
JV
K18
CP
ERROR P35-26 N21-20 MC
EXIT MODE CONTROL
ADDRESS OUT OF RANGE
INFINITE OPERAND
INDEFINITE OPERAND
M11-9 P.105
(ED=3)(INCH)
(CM → I0) + MC
K17-28
BRANCH R/P CONTROL P. 3
ERROR FF
Ø13-6
ERROR
R39-14 START EXCH
K36-8 (OX)(SCBD. ISSUE)
H05
IG
P29.1
P.3 BRANCH R/P CONTROL
F39
TC
F34-4 ISSUE
L17
IN
DISABLE ERROR RECOGNITION
EXCH JUMP CONTROL P.99
BREAK IN F26-15
MC M09-23
N22
TD
(BREAK IN)(MC)
K08-21
BRANCH R/P CONTROL P. 3
H25
TD
PAUSE
D(L-T) SEQ
P.7 L15-2
G36
CD
STOP
P. 24.1 INST GO CONTROL PART 2
clear go FF
(stops inst. issue)
NOTE THIS SHEET IS IDENTICAL TO FUNCTIONAL UNITS PAGE 14.21
CONTROL DATA CORPORATION
COMPUTER DIVISION
TITLE
CENTRAL PROCESSOR INSTRUCTION GO CONTROL PART 1, SERIALS 8-UP
PRODUCT 6601/04/13/14
SIZE C
DRAWING NO. 60119300
REV BT
SHEET 252
PAGE 24.01

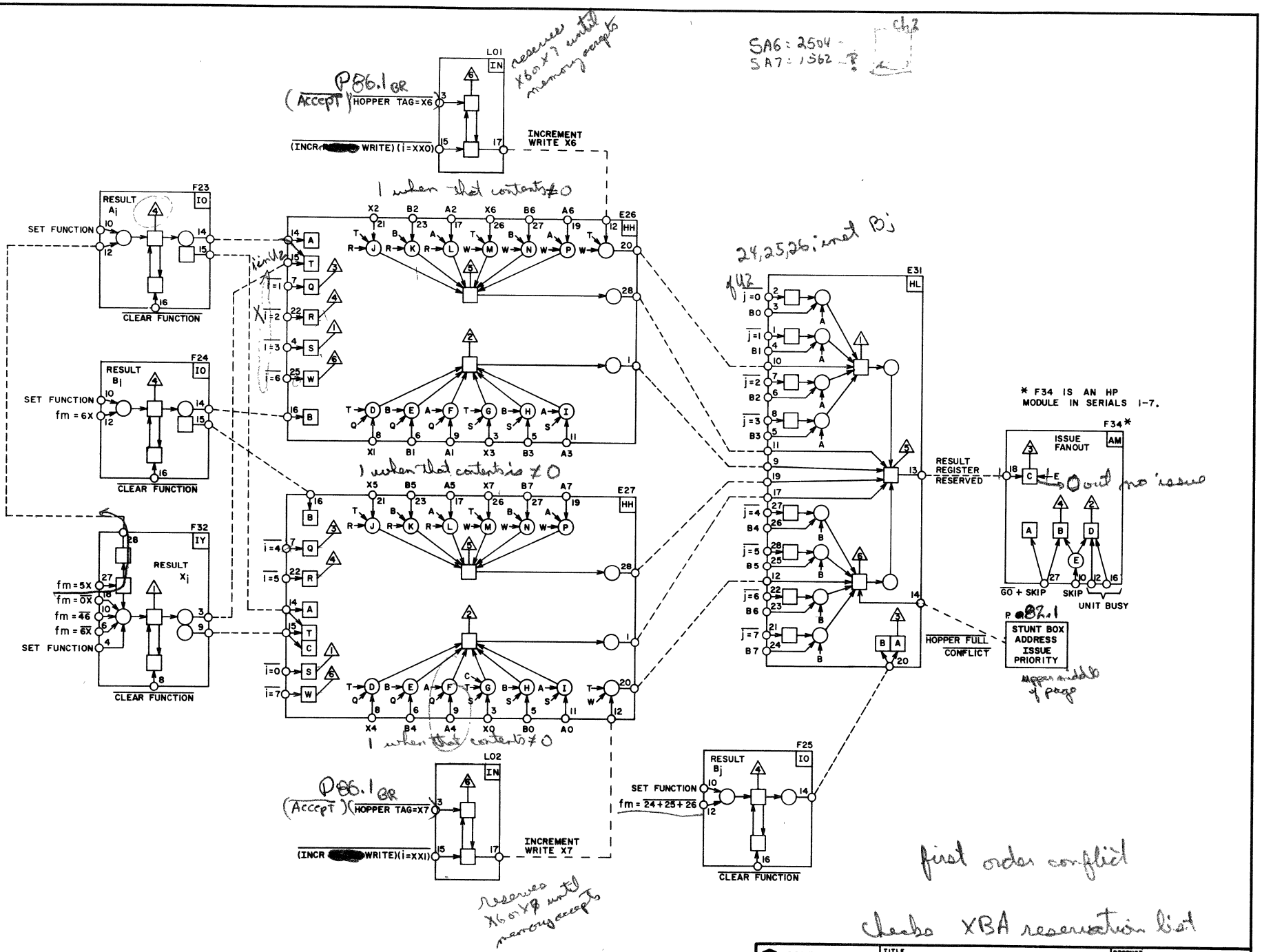


* F34 IS AN HP MODULE IN SERIALS 1-7.

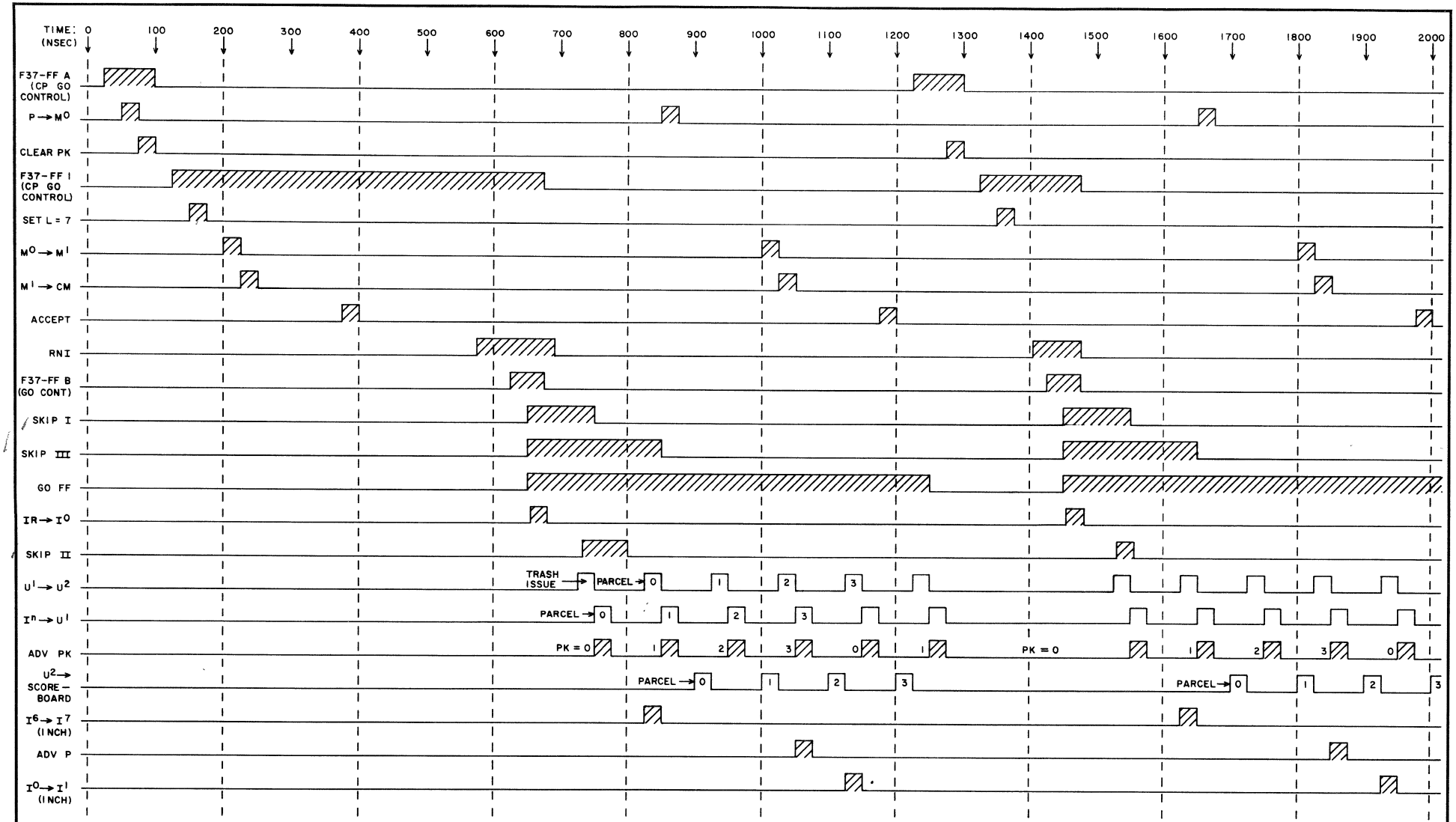
** USED IN SERIALS 8-UP ONLY.

THIS SHEET IS IDENTICAL TO FUNCTIONAL UNITS PAGE 14.3

CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR INSTRUCTION GO CONTROL PART 2	PRODUCT 6601/04/13/14	
		SIZE C	DRAWING NO. 60119300 REV 4G
		SHEET 219	PAGE 24.1



CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR INSTRUCTION CONTROL RESULT REGISTER RESERVED & ISSUE.	PRODUCT: 6601/04	
COMPUTER DIVISION		SIZE: C DRAWING NO.: 60119300	REV: L
		SHEET: 58	25

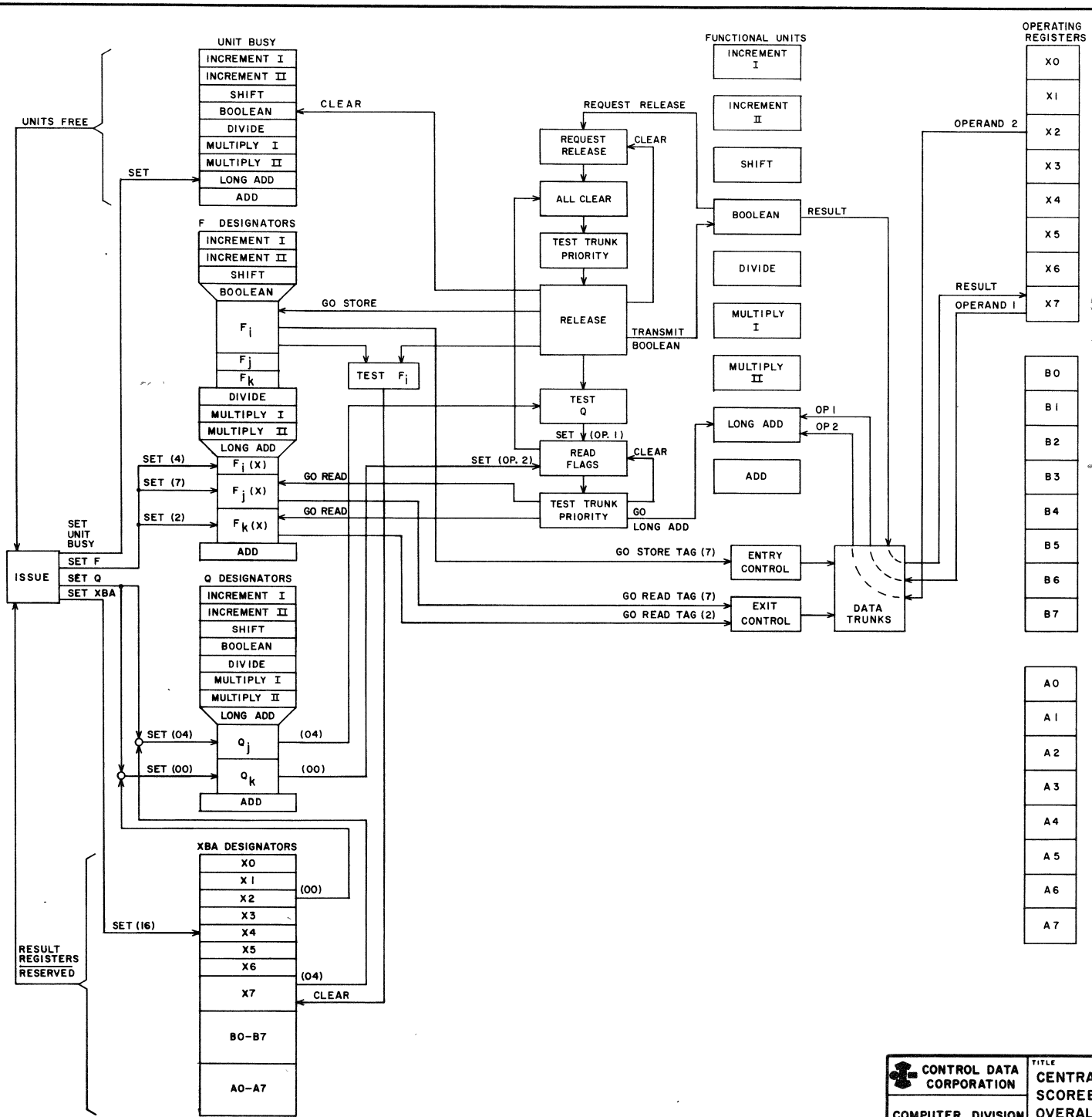


NOTE:
THIS TIMING DIAGRAM ASSUMES NO MEMORY AND/OR FUNCTIONAL UNIT CONFLICTS. NOTE THAT THE GO FF REMAINS SET UNTIL ALL PARCELS OF AN INSTRUCTION WORD HAVE BEEN ISSUED TO THE SCOREBOARD. EFFECTIVELY, THEN, A MEMORY AND/OR F.U. CONFLICT MERELY "LENGTHENS" THE TIMING CHART.

CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR INSTRUCTION CONTROL INSTRUCTION ISSUE TIMING	PRODUCT 6601	
		SIZE C	DRAWING NO. 60119300
		REV C	
		SHEET 59	27

SCOREBOARD

Concurrent operations in the Central Processor are controlled by a generalized queue and reservation scheme called the scoreboard. The scoreboard maintains a running status file of each operating register, functional unit and data trunk in the Central Processor. Typically, this file is made up of two, three and four-bit quantities identifying the nature of the unit and register usage. These status conditions (or designators) are examined as each new instruction is brought up. If no waiting is required, the execution of the instruction is begun immediately under control of the unit itself. If waiting is required (for example, an input operand may not yet be available) the scoreboard controls the delay and, when released, allows the unit to begin its execution of the instruction. The fact that one instruction is waiting does not necessarily prevent later instructions from being brought up and issued. The two restrictions on issuing are that the unit requested must be free, and that the result register must not be reserved for the result of a previously issued instruction. In the absence of these two restraints, instructions may be issued every minor cycle. Thus, several instructions may be executed concurrently, not necessarily in the same order as that of the original program.



1. Instruction Sequence:

   Issue: Boolean (Logical sum of X_3 and X_5 to X_7)
   Issue: Long Add (Integer sum of X_7 and X_2 to X_4)

2. Assumptions:

   a. Boolean is already issued and in progress.
      Unit Busy Set — Q_i = 00
      $F_i(x)$ = 111 — Q_k = 00
      $F_j(x)$ = 011 — X_7 = 0100
      $F_k(x)$ = 101

   b. Long Add is being issued.

3. Set Long Add Unit Busy FF.

4. Set Long Add F Designators.

   a. 100 → $F_i(x)$
   b. 111 → $F_j(x)$
   c. 010 → $F_k(x)$

5. Set Long Add Q Designators from XBA

   a. X7 = 0100 (Boolean) → Q_j
   b. X2 = 0000 (Clear) → Q_k
      Set Long Add Read Flag Operand 2.

6. Set XBA Designator

   1110 (Long Add) → X4

7. Boolean sends Request Release near the end of its computation.
8. All clear gate is made if no Read Flags are set for the result register.
9. Boolean Trunk Priority is tested.
10. Boolean is released by scoreboard.
    a. Unit Busy and Request Release FFs are cleared.
    b. Boolean F_i is tested to clear XBA (X7)
    c. Transmit signal is sent to Boolean to store its result.
    d. Go Store signal probes Boolean F_i and a tag is sent to the Operating Register Entry Control to define the result register.
11. Boolean Result is transmitted to Operating Register (X7)
12. Scoreboard Tests Q to determine any units that are waiting for the result being placed in X7. (Long Add is waiting.)
13. Long Add Read Flag (Operand 1) is set. (Op. 2 Read Flag previously set when Q_k = 0)
14. Long Add Trunk Priority is tested.
    a. Go signal is sent to Long Add.
    b. Go Read signal probes Long Add F_i and F_k and tags are sent to the Operating Register Exit Control to define the operand registers.
    c. Read Flags are cleared.
15. Long Add Operands are transmitted from Operating Registers (X7 and X2 respectively).
16. Long Add begins computation.

CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE: CENTRAL PROCESSOR SCOREBOARD OVERALL BLOCK DIAGRAM

PRODUCT: 6601/04
SIZE: C | DRAWING NO.: 60119300 | REV: K
SHEET: 60 | 29

SCOREBOARD DESIGNATORS

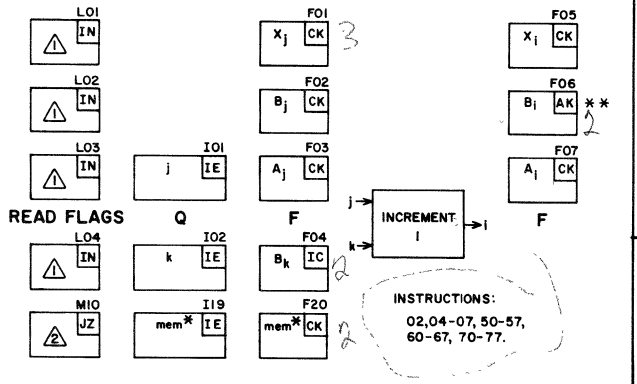
1. F Designators: These three bit designators assign specific operating registers to the selected functional unit. In the general case, F_i designates the result register whereas F_j and F_k designate the entry operand registers. The shift and increment units have multiple F designators for assigning either X, B or A registers. These designators are used to gate the entry and exit control networks which direct the data flow on trunks between the operating registers and the functional units.

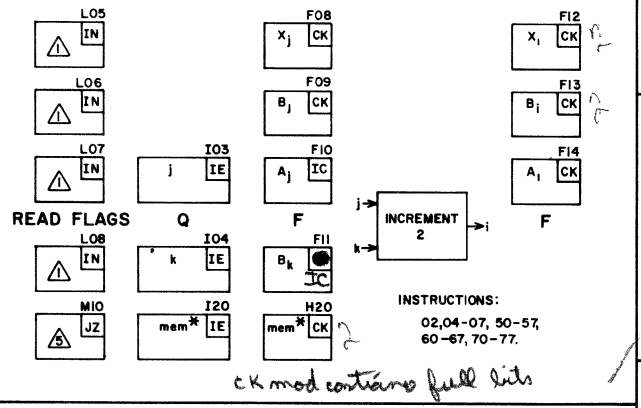
2. Q Designators: If an entry operand register has been previously reserved for the result of another functional unit already in operation, the four-bit Q designator specifies the reserving unit. These designators are used to set read flags for the entry operands.

3. XBA Designators: The result register reservation list is held in the twenty-four XBA designators. Each designator contains the four-bit* binary code of the current reserving unit. If there is no reserving unit, the designator is cleared. These designators are used both to prohibit the issuance of subsequent instructions which call for a result register previously reserved, and to set Q designators which indicate any entry operand conflicts that may exist.

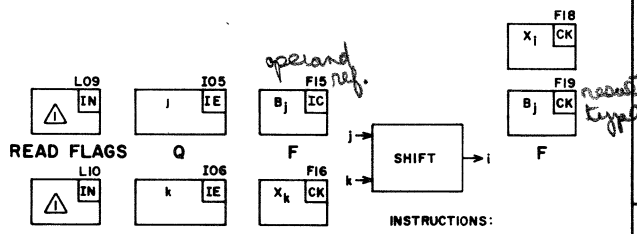
---

* For all B and A designators, the upper two bits are always zero.



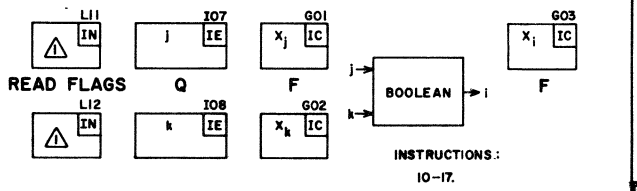


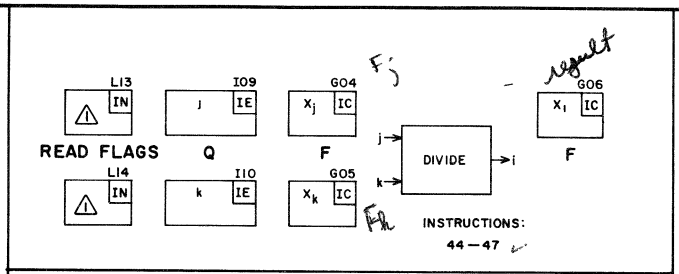
CK mod contains full bits

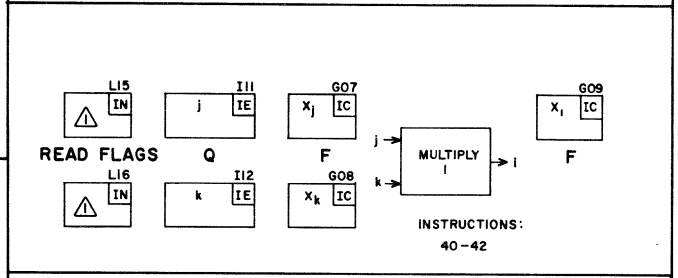


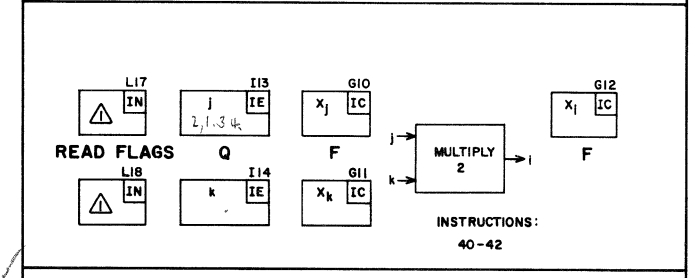
operand ref.

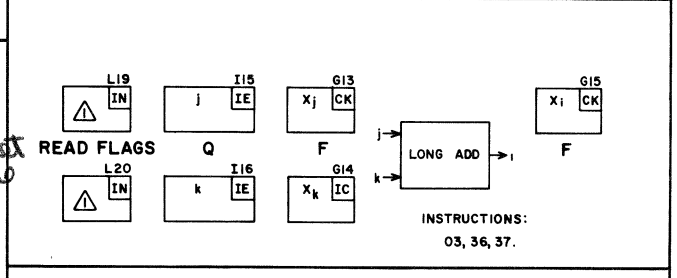
result type

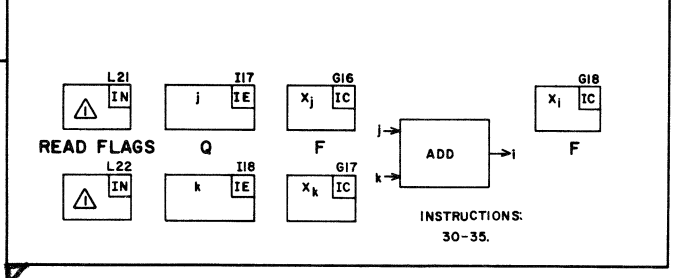












FUNCTIONAL UNIT	Q (OCTAL)
INCREMENT 1	01
INCREMENT 2	02
SHIFT	03
BOOLEAN	04
DIVIDE	05
MULTIPLY 1	06
MULTIPLY 2	07
READ MEMORY, CHAN. 1	11
READ MEMORY, CHAN. 2	12
READ MEMORY, CHAN 3	13
READ MEMORY, CHAN. 4	14
READ MEMORY, CHAN. 5	15
LONG ADD	16
ADD	17

load X1

	X	B	A
0	H01	H09	H09
1	H02	H10	H10
2	H03	H11	H11
3	H04	H12	H12
4	H05	N11	N11
5	H06	N12	N12
6	H07	N13	N13
7	H08	N14	N14

DESIGNATORS

map

	2^3	2^2	2^1	2^0
X	1	2	6	5
B	–	–	4	2
A	–	–	6	1

TP locaters

	2^3	2^2	2^1	2^0
F DESIGNATORS	–	5	6	1
Q DESIGNATORS	4	3	1	2

* USED IN STORE INSTRUCTIONS (5X) WHERE I = 6 + 7.

** F06 IS A CK MODULE IN SERIALS 1-7.

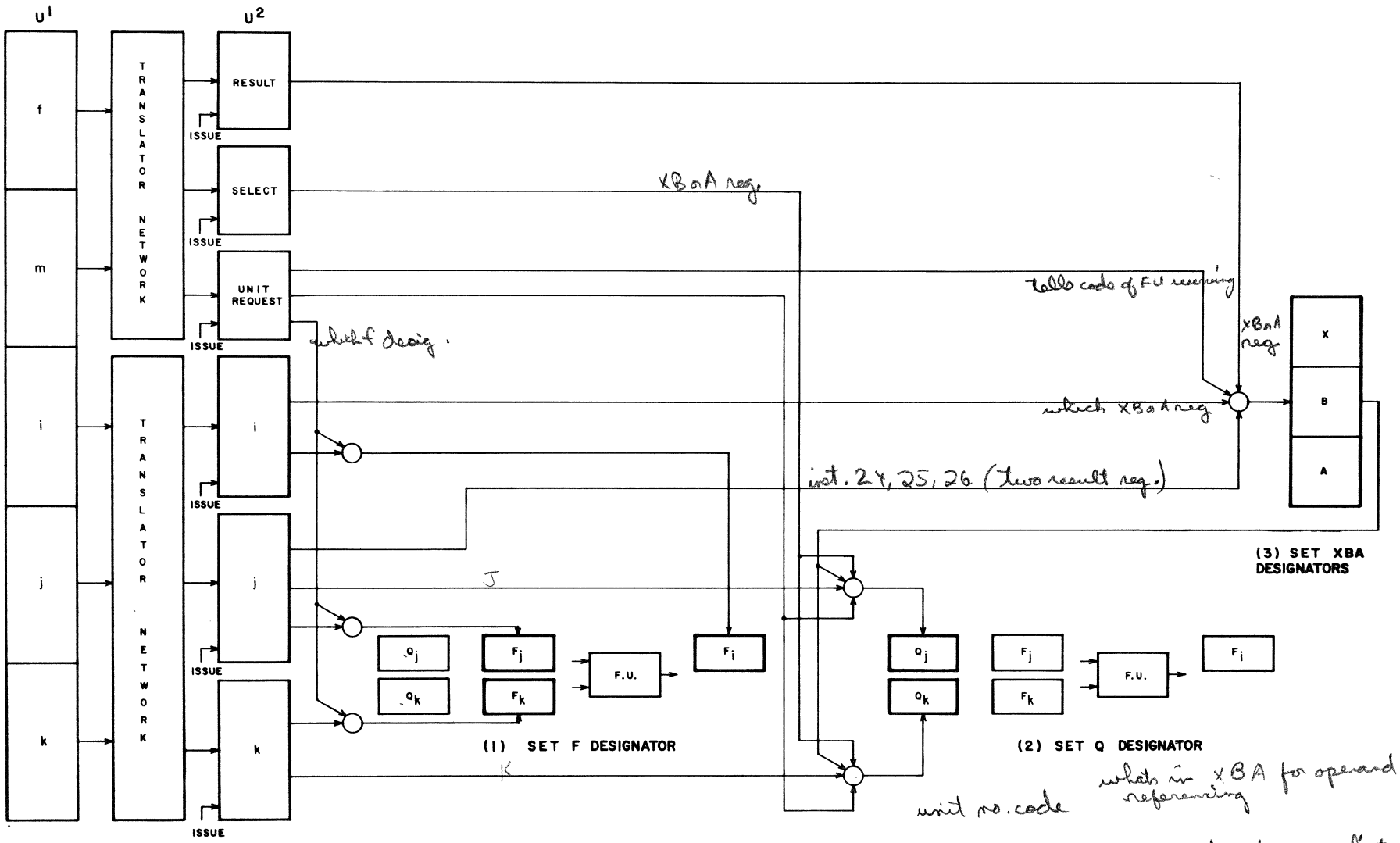
CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE
CENTRAL PROCESSOR
SCOREBOARD
DESIGNATORS

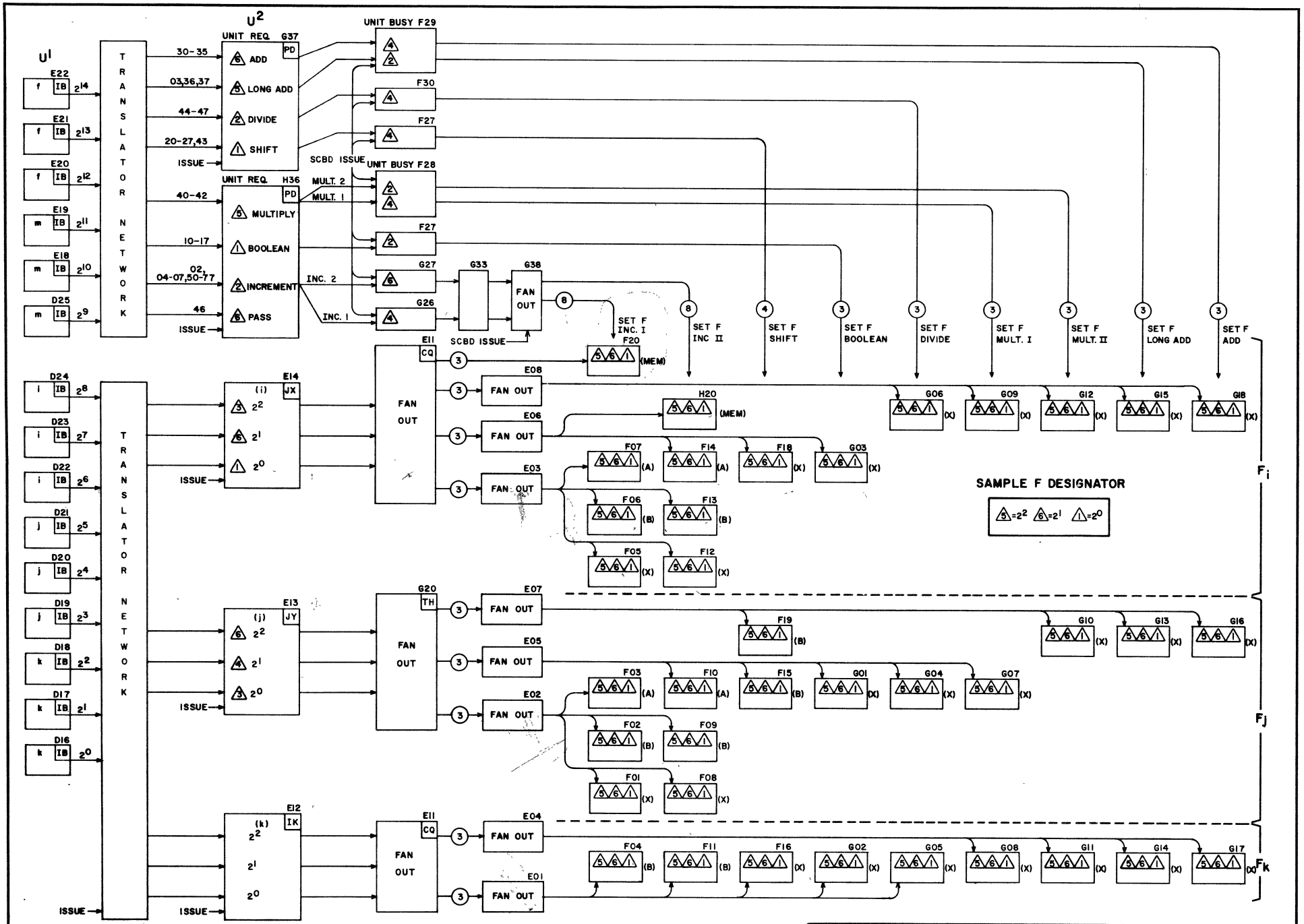
PRODUCT 6601/04
SIZE C | DRAWING NO. 60119300 | REV
SHEET 61 | 31

PLACING SCOREBOARD RESERVATIONS

When an instruction is issued to the scoreboard ($U^1 \rightarrow U^2$) the fm portion, or operation code, is translated to set various reservation control flip-flops. These include Unit Request (to specify the functional unit), Select (to specify entry operand register groups), and Result flip-flops (to specify result register groups). The particular registers within the specified register groups are identified from translations of the i, j and k portions of U^2. Signals emanating from U^2 set the appropriate Unit Busy flip-flops and scoreboard designators F, Q, and XBA.



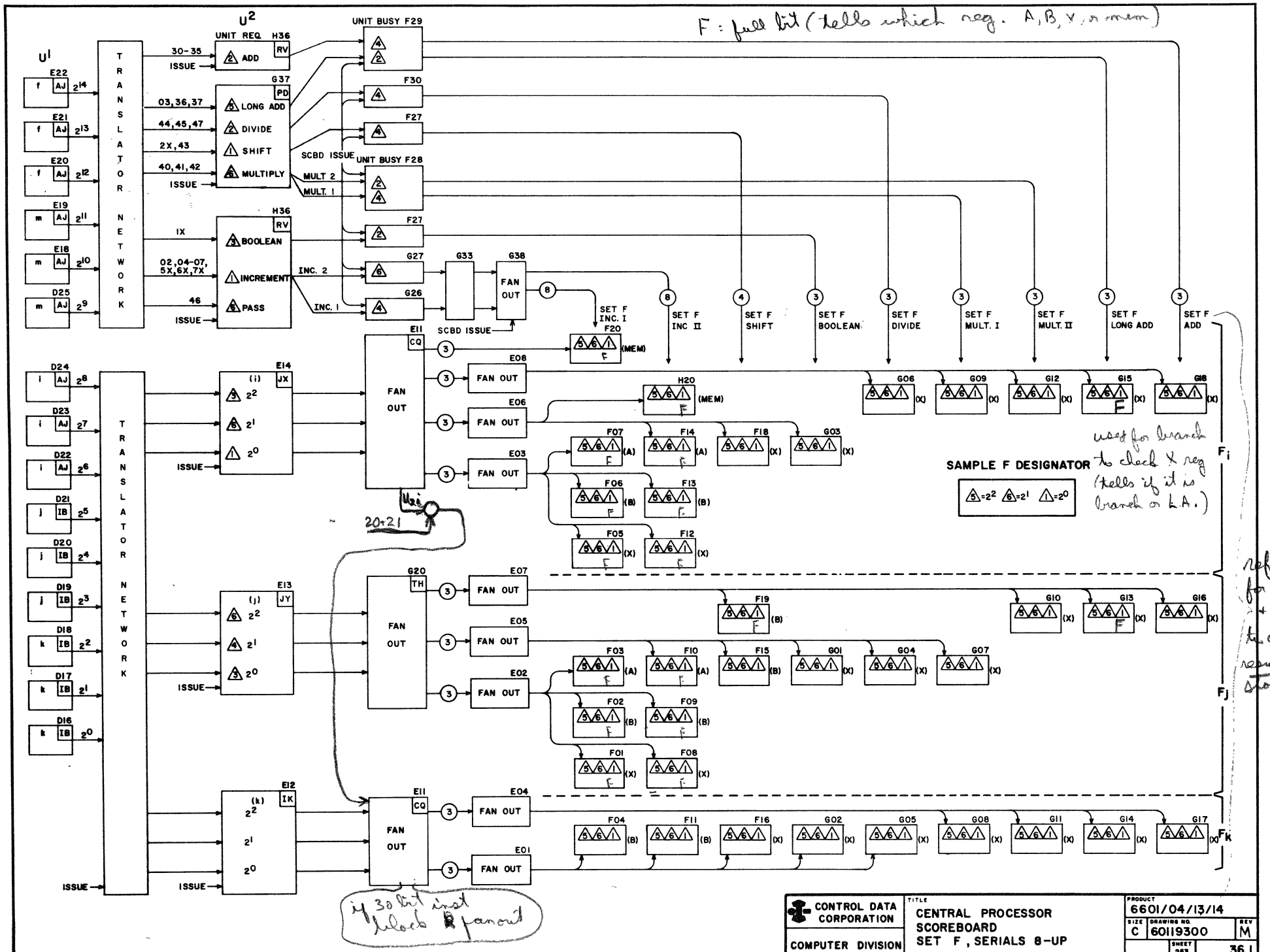
CONTROL DATA CORPORATION	TITLE CENTRAL PROCESSOR SCOREBOARD BLOCK DIAGRAM SET F, Q, XBA	PRODUCT 6601/04	
COMPUTER DIVISION		SIZE C	DRAWING NO. 60119300 REV K
			SHEET 62 33

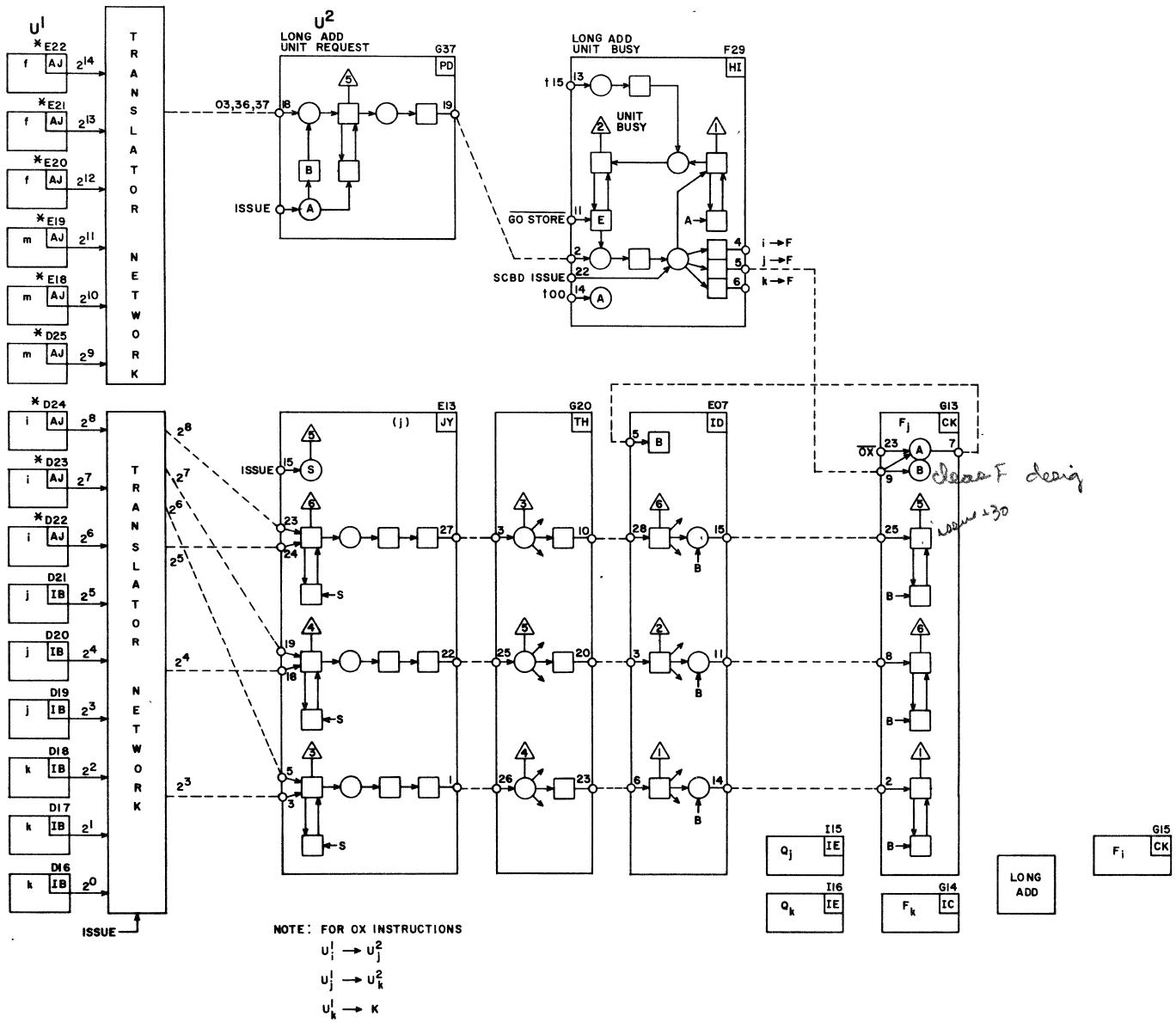


CONTROL DATA CORPORATION
COMPUTER DIVISION

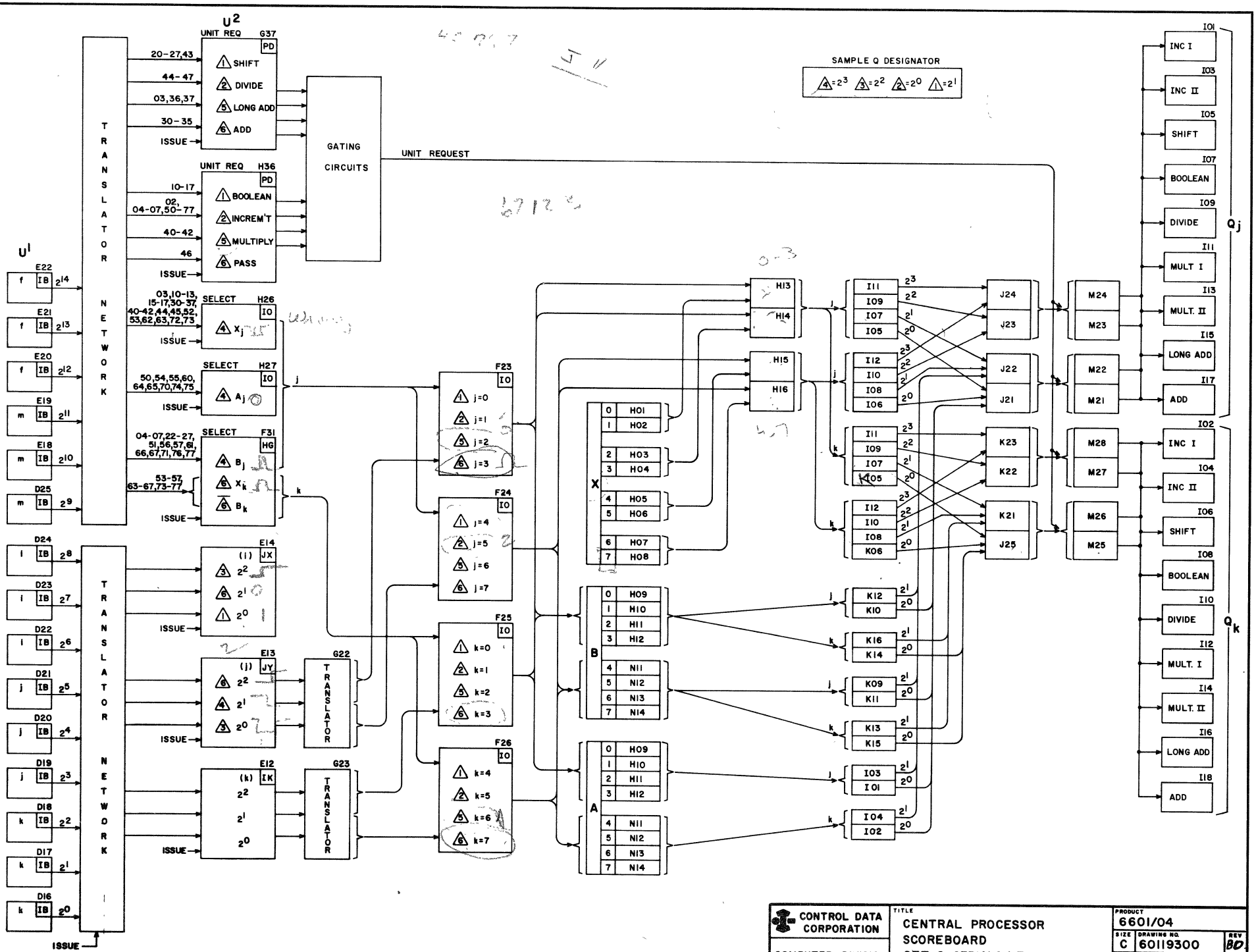
TITLE
CENTRAL PROCESSOR
SCOREBOARD
SET F, SERIALS 1-7

PRODUCT 6601/04
SIZE C | DRAWING NO. 60119300 | REV M
SHEET 63
35


F: full list (tells which reg. A, B, X, or mem)
U1
U2
TRANSLATOR NETWORK
UNIT REQ H36
ADD
LONG ADD
DIVIDE
SHIFT
MULTIPLY
BOOLEAN
INCREMENT
PASS
ISSUE
30-35
03,36,37
44,45,47
2X,43
40,41,42
IX
02,04-07, 5X,6X,7X
46
UNIT BUSY F29
UNIT BUSY F28
F30
F27
SCBD ISSUE
MULT 2
MULT 1
INC. 2
INC. 1
G37
G33
G38
G27
G26
FAN OUT
SET F INC. I
SET F INC II
SET F SHIFT
SET F BOOLEAN
SET F DIVIDE
SET F MULT. I
SET F MULT. II
SET F LONG ADD
SET F ADD
F20 (MEM)
H20 (MEM)
SAMPLE F DESIGNATOR
used for branch to check X reg (tells if it is branch or L.A.)
20+21
if 30 bit inst blocks B fanout
ref. for entry + exit to get results stored
Fi
Fj
Fk
CONTROL DATA CORPORATION
COMPUTER DIVISION
CENTRAL PROCESSOR SCOREBOARD SET F, SERIALS 8-UP
6601/04/13/14
C
60119300
M
253
36.1



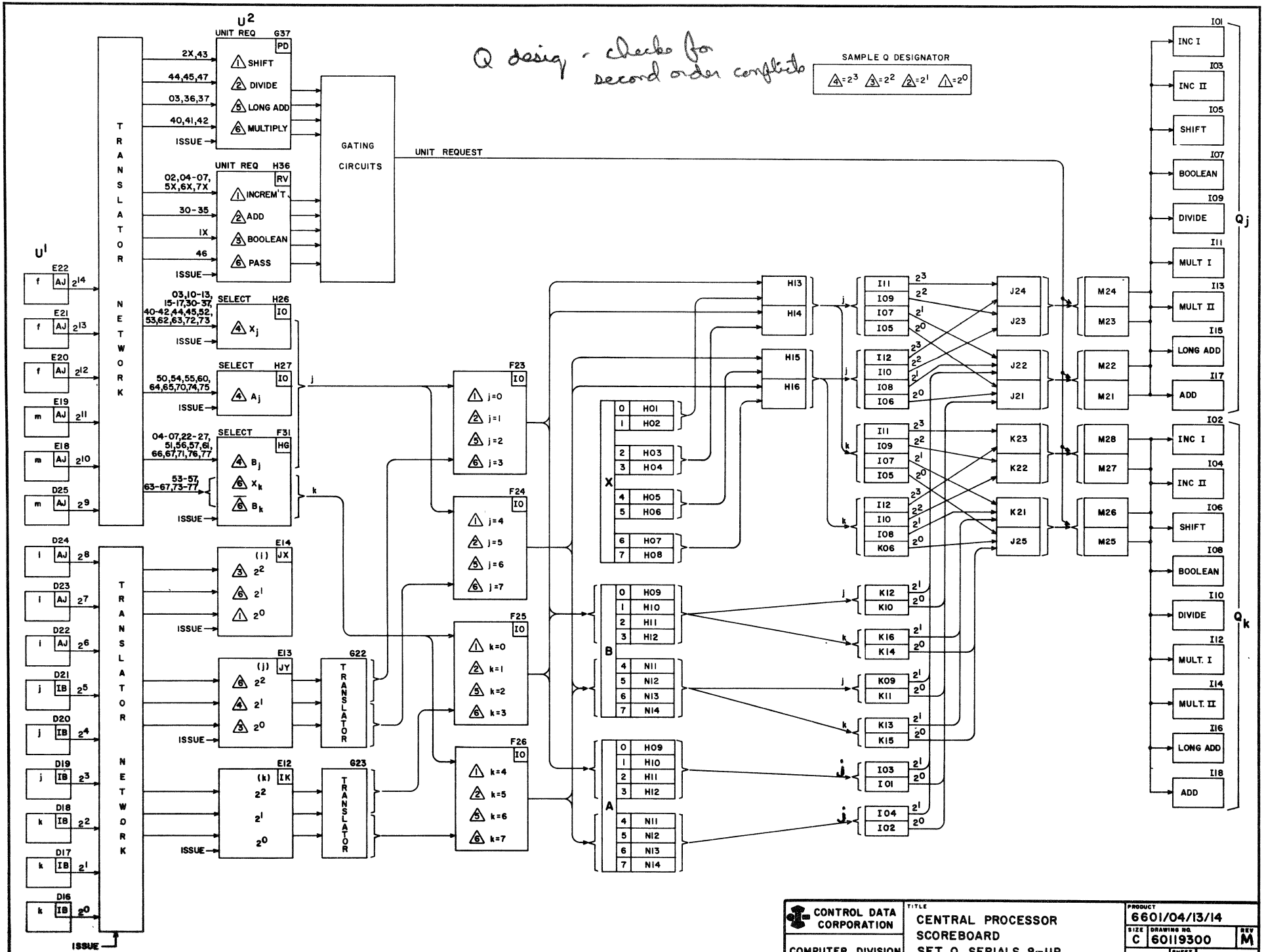
CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR SCOREBOARD SET F (LONG ADD, Fj)	PRODUCT 6601/04/13/14	
		SIZE C	DRAWING NO. 60119300 REV L
			SHEET 64 37



CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE
CENTRAL PROCESSOR
SCOREBOARD
SET Q, SERIALS 1-7

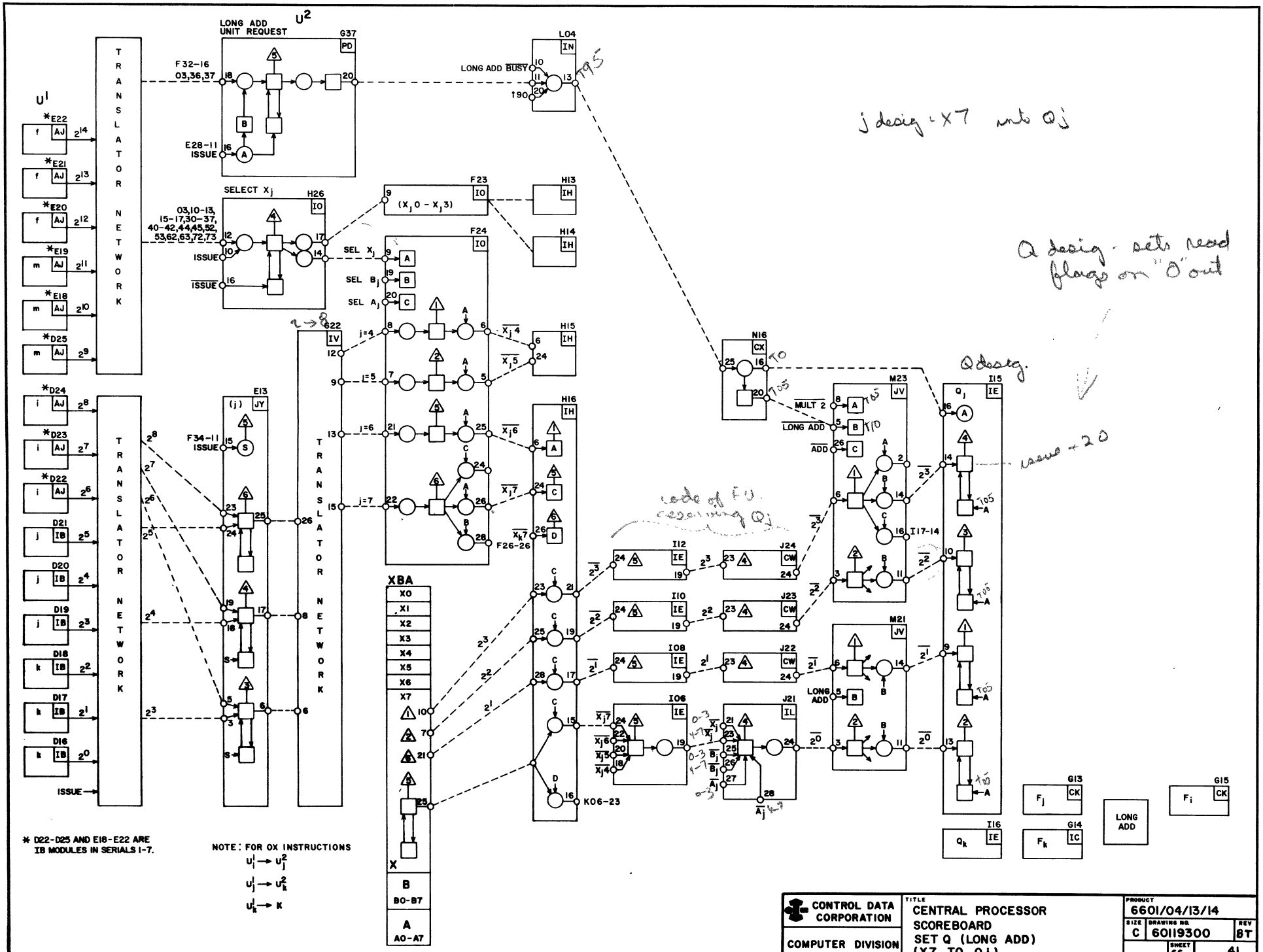
PRODUCT 6601/04
SIZE C
DRAWING NO. 60119300
REV BD
SHEET 65
39



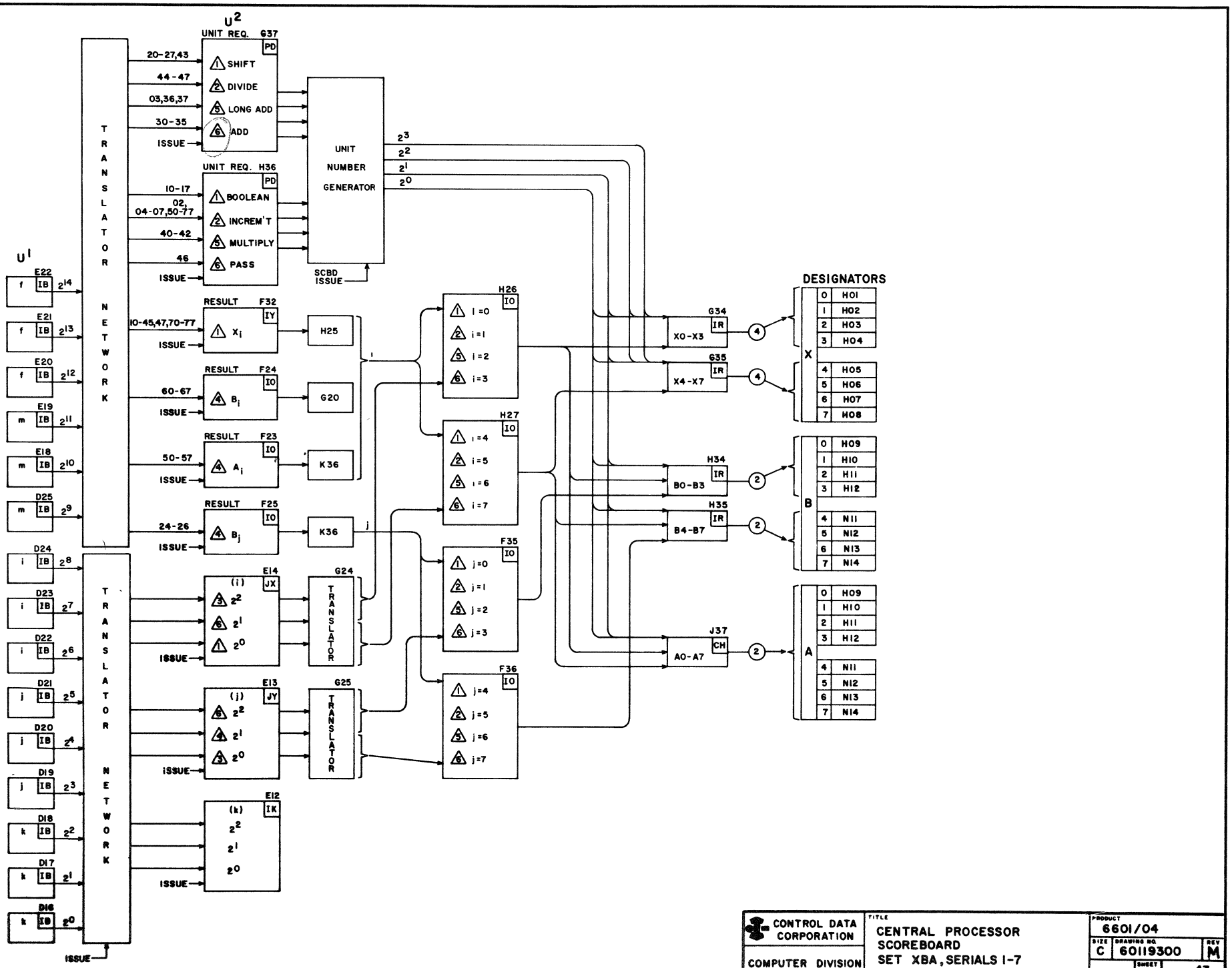
CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE
CENTRAL PROCESSOR
SCOREBOARD
SET Q, SERIALS 8-UP

PRODUCT 6601/04/13/14
SIZE C DRAWING NO. 60119300 REV M
SHEET 254 40.1



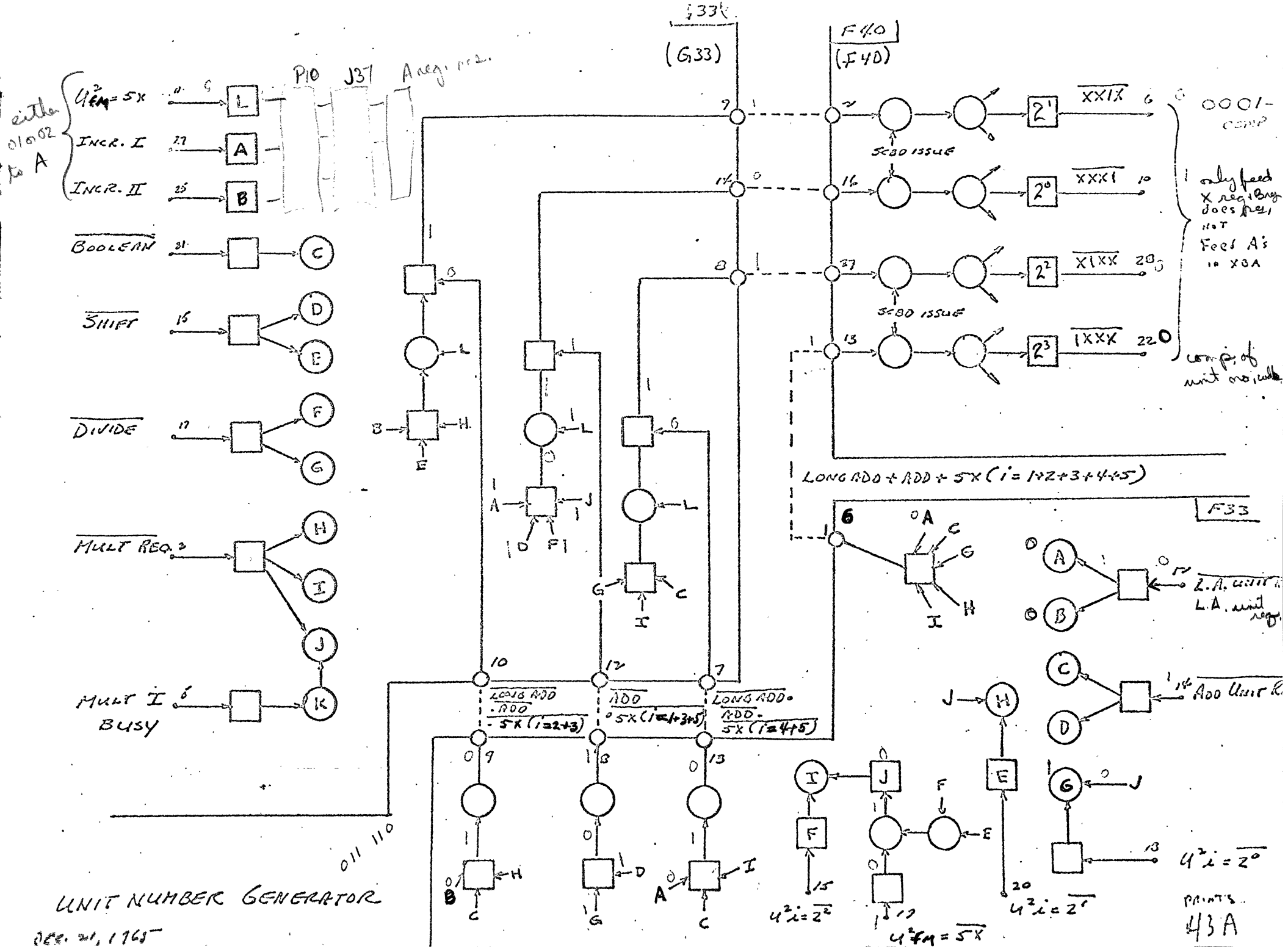
CONTROL DATA CORPORATION	TITLE	PRODUCT	
COMPUTER DIVISION	CENTRAL PROCESSOR SCOREBOARD SET Q (LONG ADD) (X7 TO Qj)	6601/04/13/14	
		SIZE C	DRAWING NO. 60119300 REV 8T
			SHEET 66 41

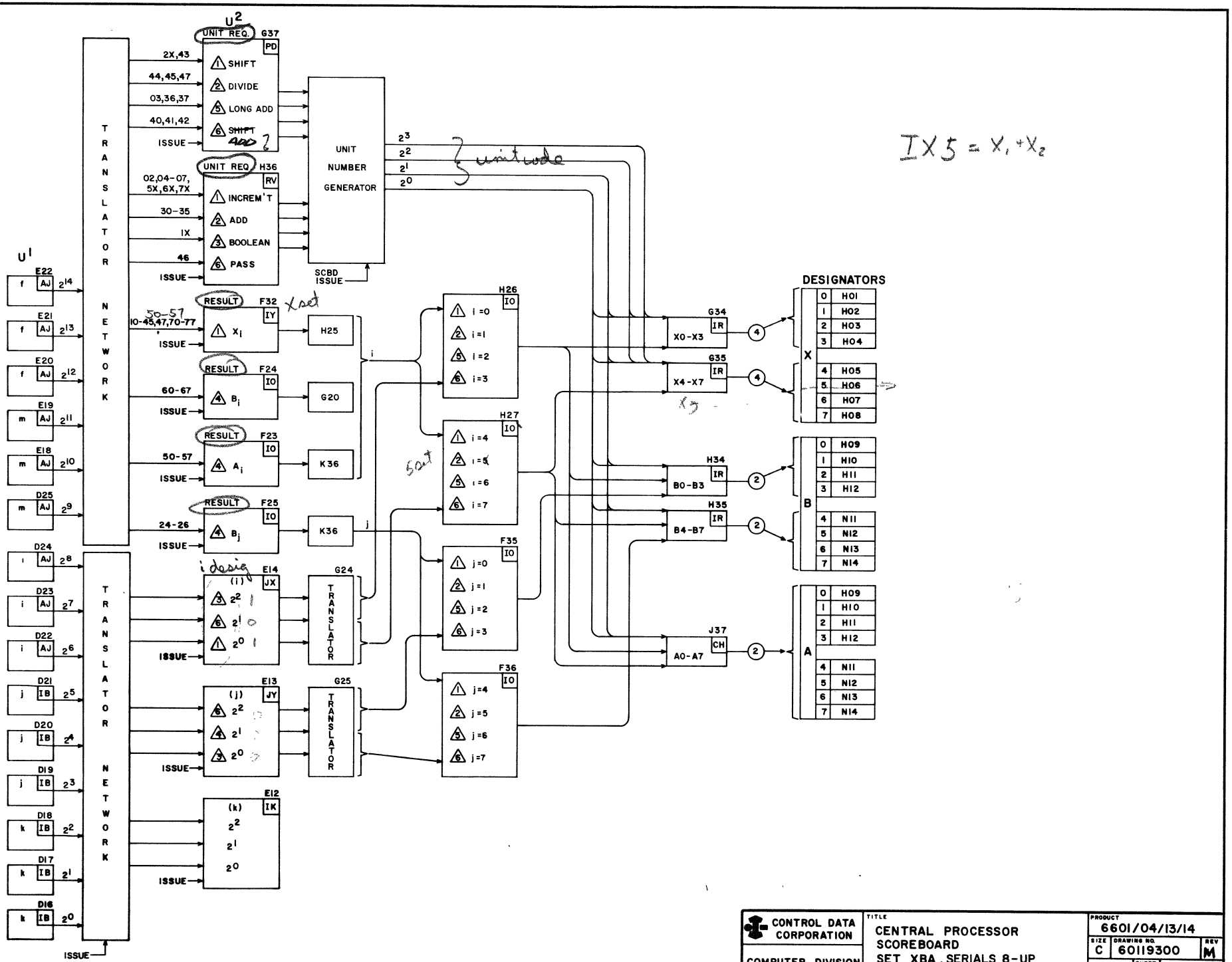


CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE
CENTRAL PROCESSOR
SCOREBOARD
SET XBA, SERIALS 1-7

PRODUCT 6601/04
SIZE C | DRAWING NO. 60119300 | REV M
SHEET 67
43

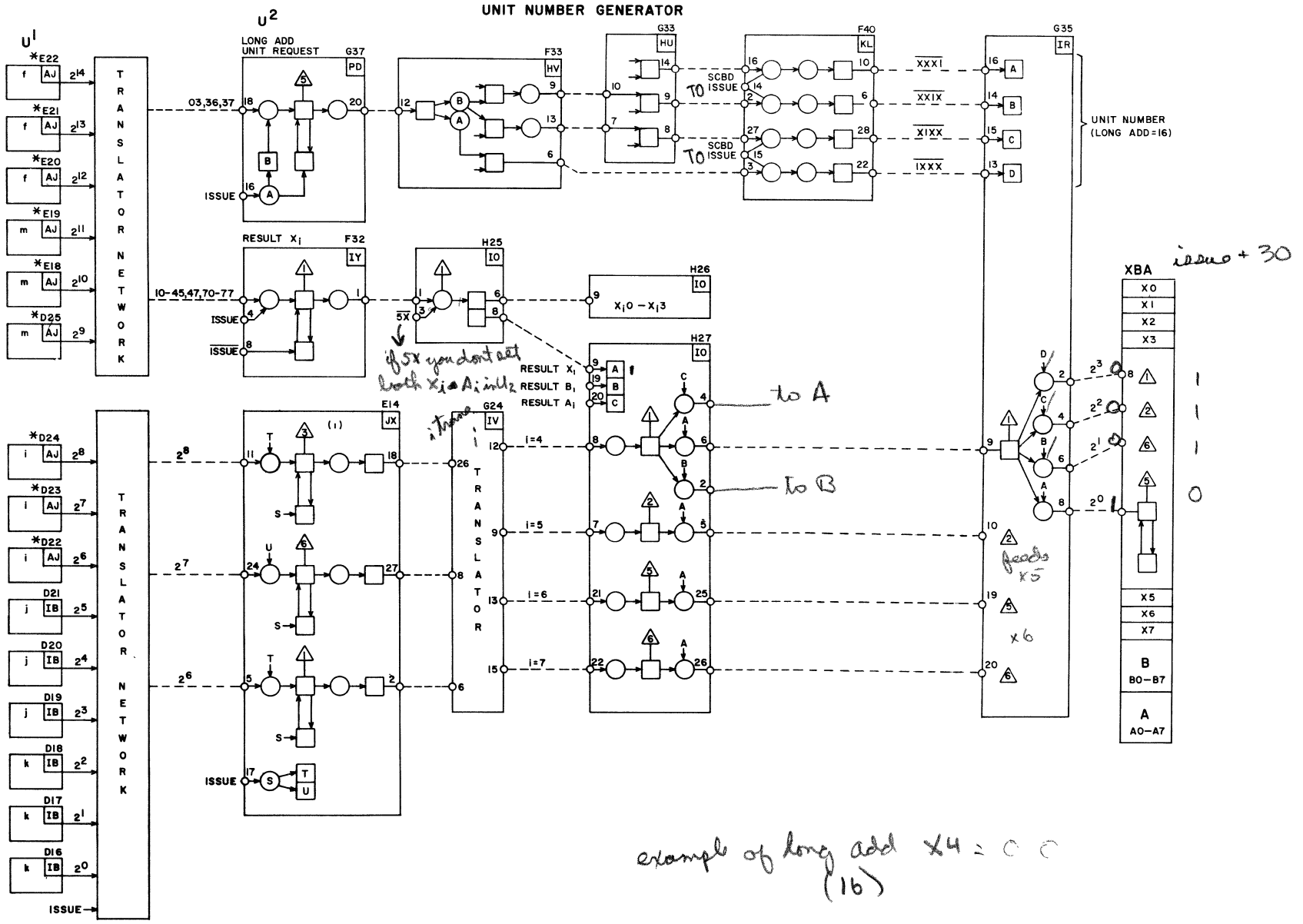

F33
(G33)
F40
(F40)
either 01 or 02 to A
INCR. I
INCR. II
BOOLEAN
SHIFT
DIVIDE
MULT REQ.
MULT I BUSY
P10
J37
A reg. no.
L
A
B
C
D
E
F
G
H
I
J
K
SC80 ISSUE
XXIX
XXXI
XIXX
IXXX
0001-
only feed X reg. Bus does req. NOT Feed A's to XOA
comp. of unit no. in
LONG ADD + ADD + 5X (i = 1+2+3+4+5)
LONG ADD · ADD · 5X (i=2+3)
ADD · 5X (i=1+3+5)
LONG ADD · ADD · 5X (i=4+5)
011 110
L.A. unit req.
ADD Unit Req.
UNIT NUMBER GENERATOR
DEC. 21, 1965
PRINTS 43A



CONTROL DATA CORPORATION
COMPUTER DIVISION

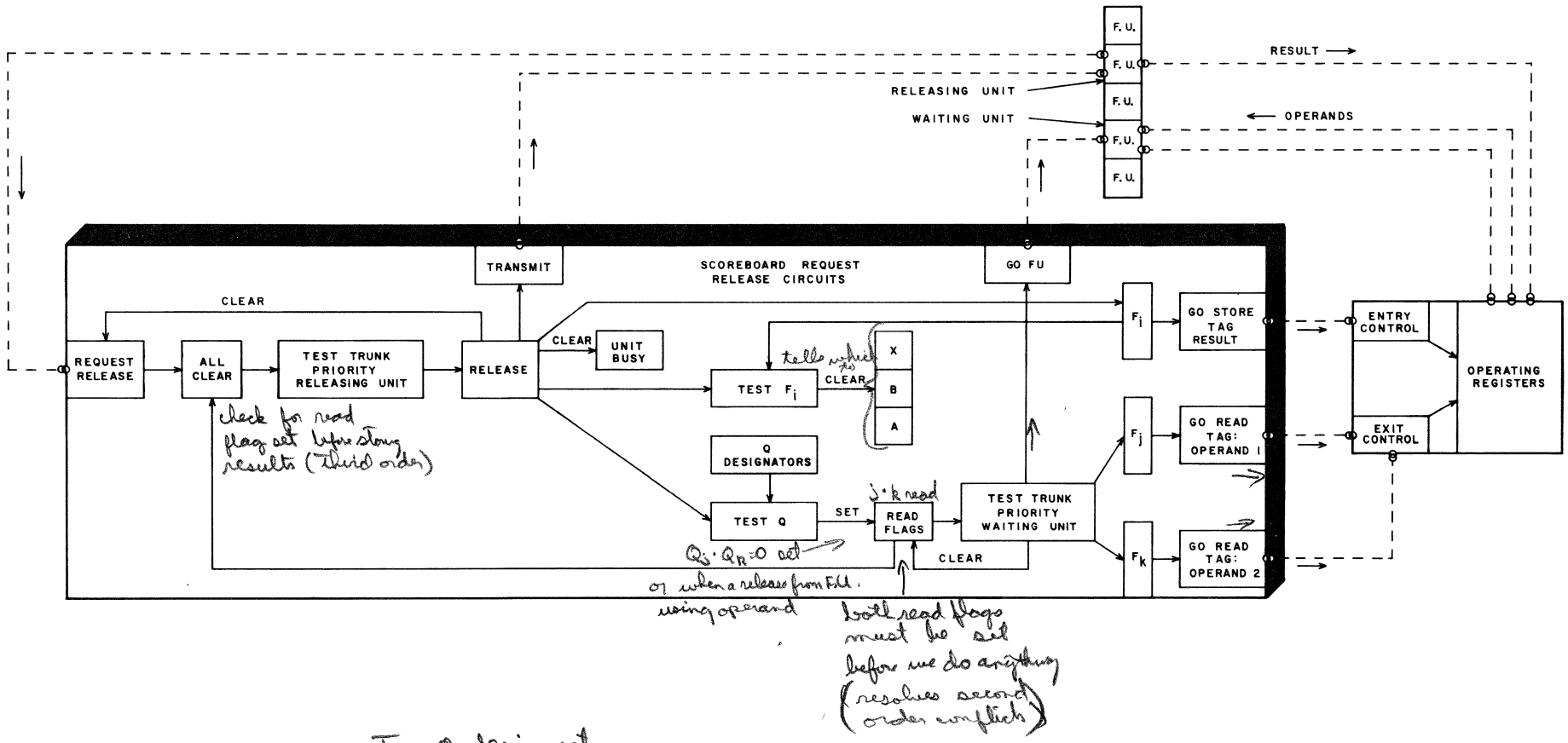
TITLE
CENTRAL PROCESSOR
SCOREBOARD
SET XBA, SERIALS 8-UP

PRODUCT 6601/04/13/14
SIZE C | DRAWING NO. 60119300 | REV M
SHEET 255 | 44.1

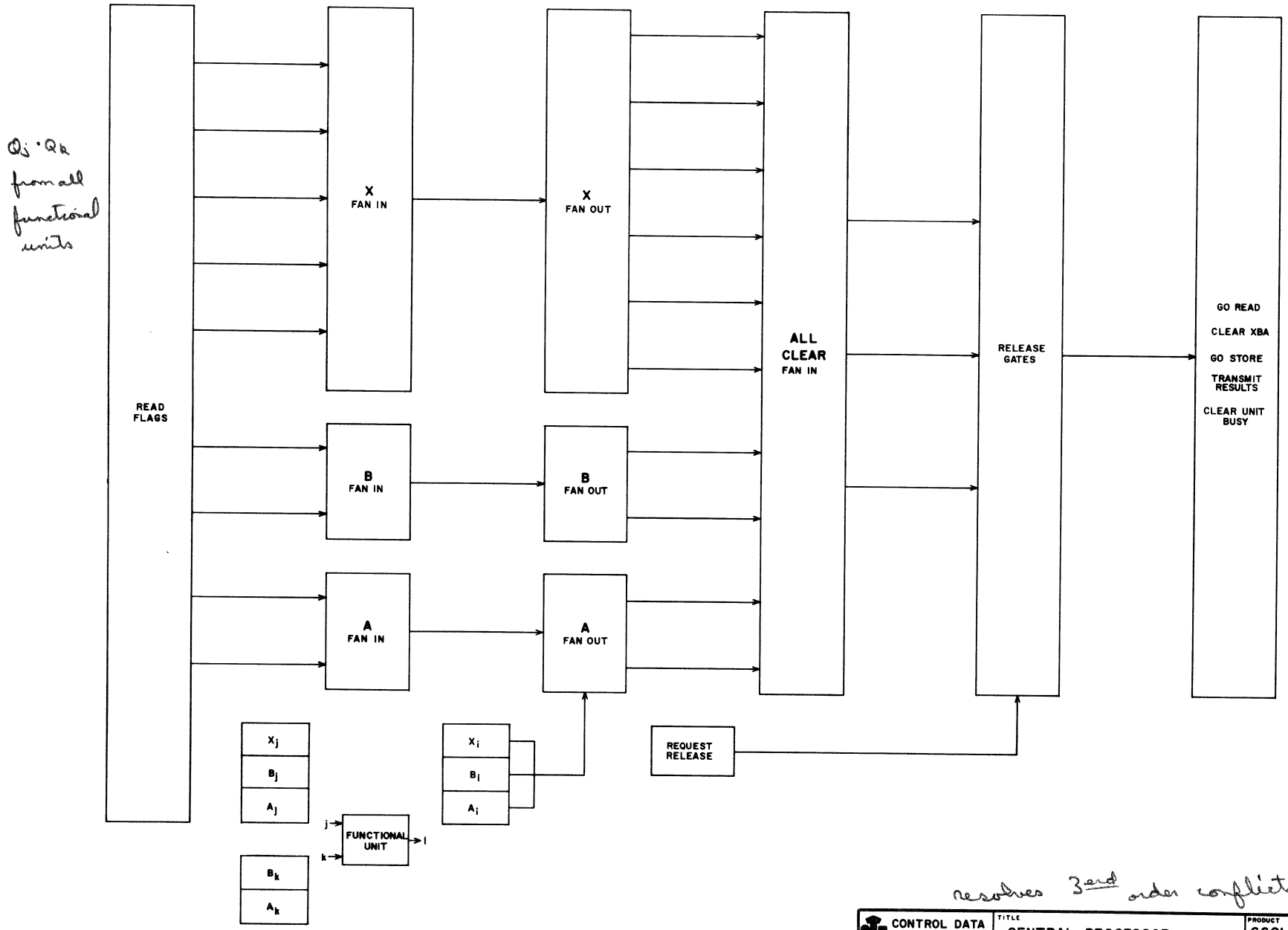


*D22–D25 AND E18–E22 ARE IB MODULES IN SERIALS 1-7.

CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR SCOREBOARD SET XBA LONG ADD (16) TO X4	PRODUCT 6601/04/13/14 SIZE C · DRAWING NO. 60119300 · REV L SHEET 68	45

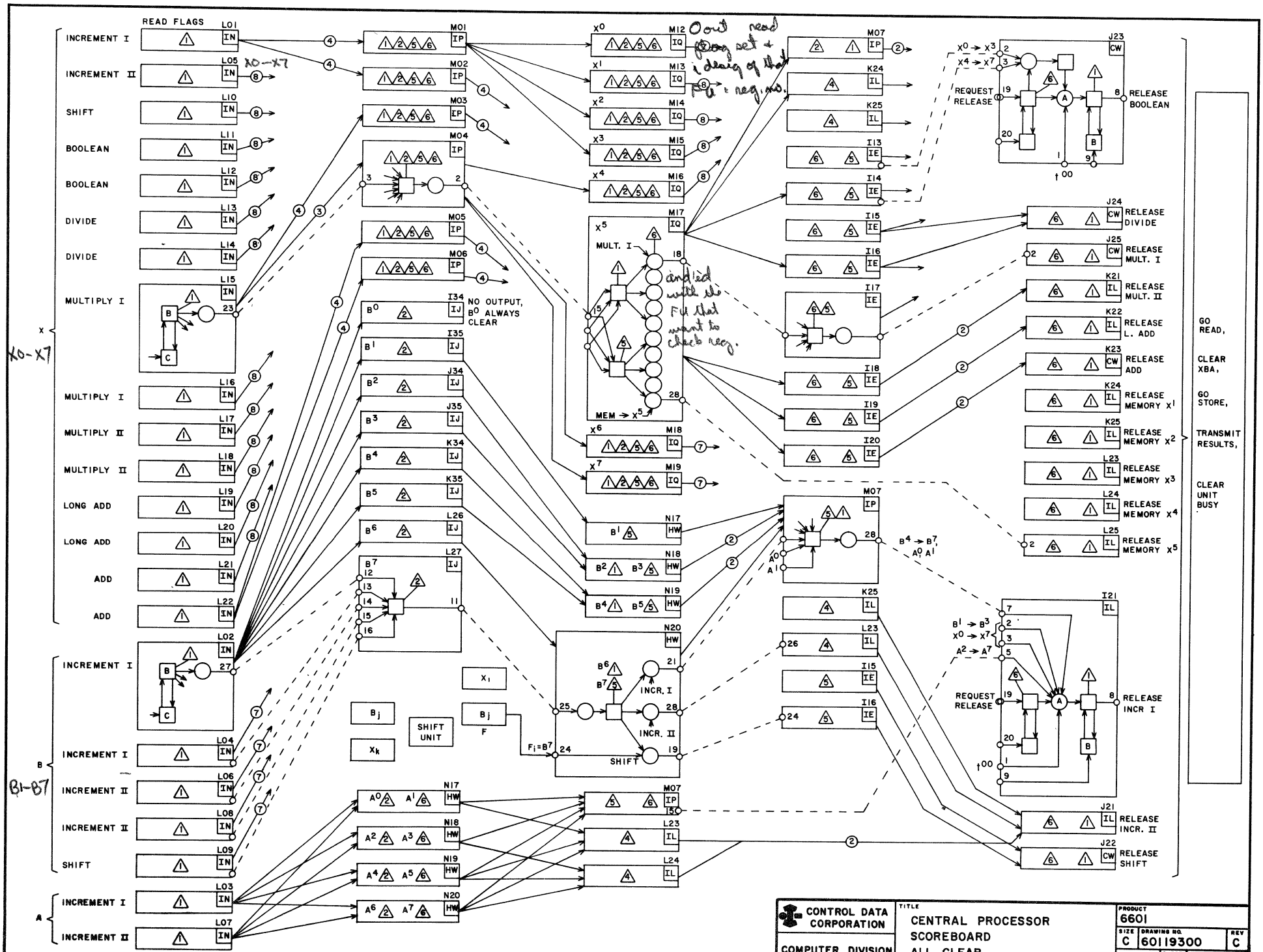


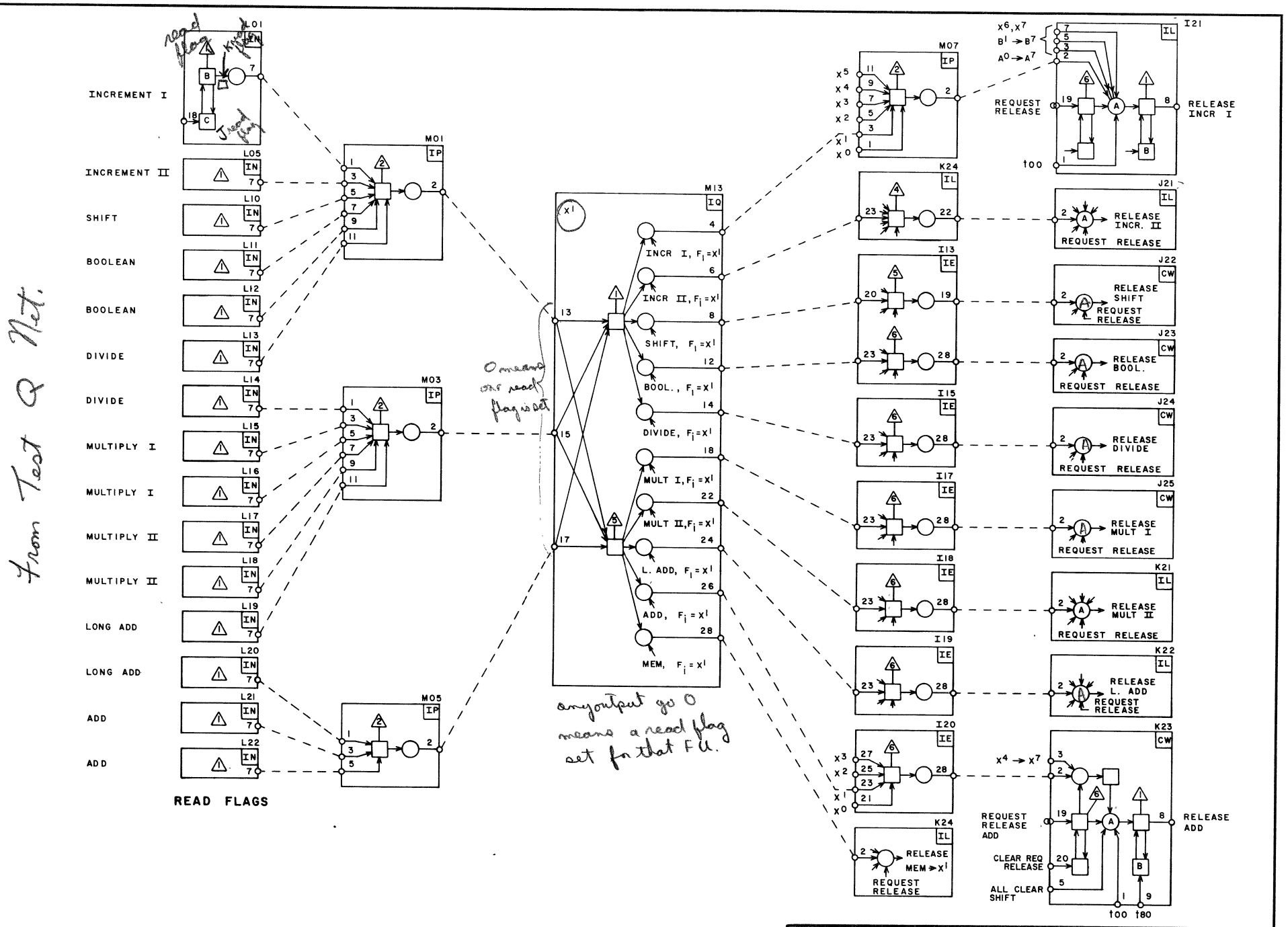
CONTROL DATA CORPORATION	TITLE CENTRAL PROCESSOR SCOREBOARD BLOCK DIAGRAM REQUEST RELEASE CIRCUITS	PRODUCT 6601	
COMPUTER DIVISION		SIZE C	DRAWING NO. 60119300 REV C
		SHEET 69	47



resolves 3rd order conflicts

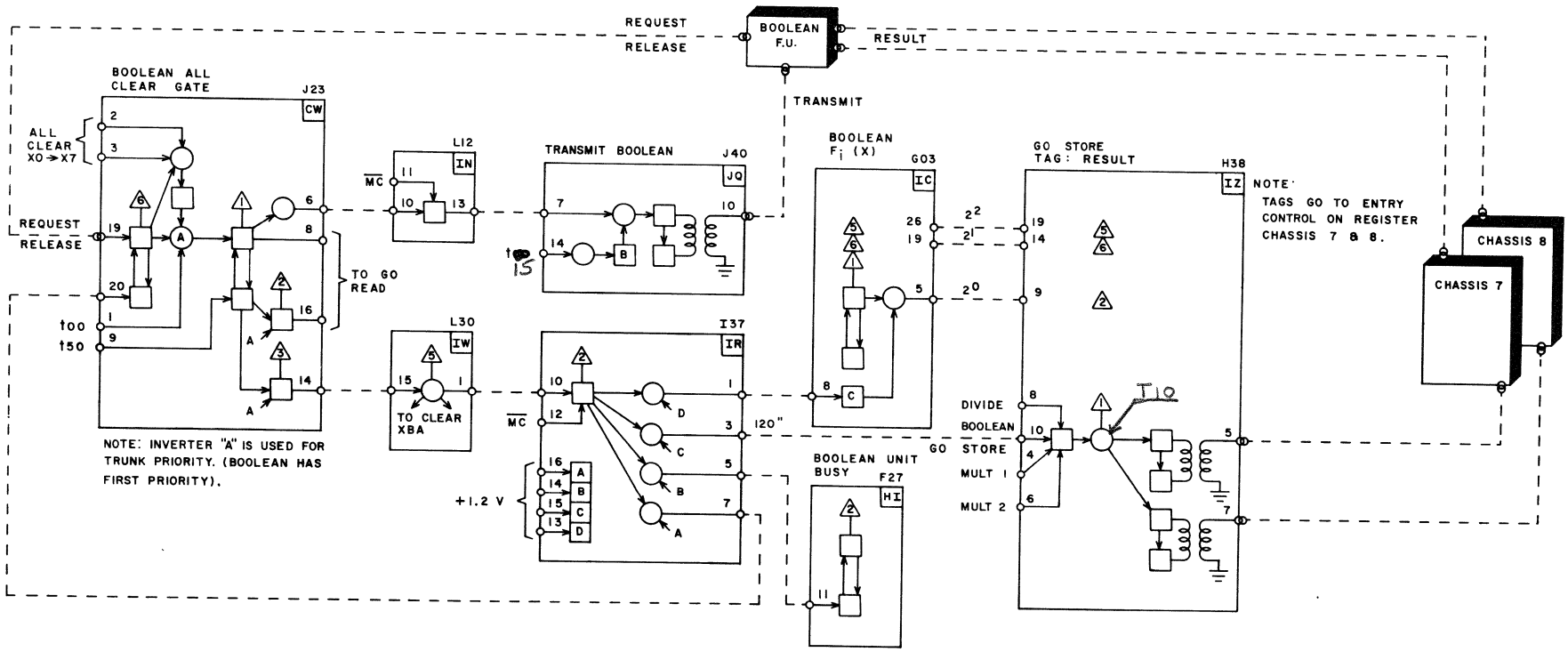
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR SCOREBOARD ALL CLEAR BLOCK DIAGRAM	PRODUCT: 6601	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: C
			SHEET: 70 49


READ FLAGS
INCREMENT I
INCREMENT II
SHIFT
BOOLEAN
BOOLEAN
DIVIDE
DIVIDE
MULTIPLY I
MULTIPLY I
MULTIPLY II
MULTIPLY II
LONG ADD
LONG ADD
ADD
ADD
X
X0-X7
INCREMENT I
INCREMENT I
INCREMENT II
INCREMENT II
SHIFT
B
B1-B7
INCREMENT I
INCREMENT II
A
NO OUTPUT, B0 ALWAYS CLEAR
MULT. I
MEM → X5
INCR. I
INCR. II
SHIFT
SHIFT UNIT
REQUEST RELEASE
RELEASE BOOLEAN
RELEASE DIVIDE
RELEASE MULT. I
RELEASE MULT. II
RELEASE L. ADD
RELEASE ADD
RELEASE MEMORY X1
RELEASE MEMORY X2
RELEASE MEMORY X3
RELEASE MEMORY X4
RELEASE MEMORY X5
RELEASE INCR I
RELEASE INCR II
RELEASE SHIFT
GO READ,
CLEAR XBA,
GO STORE,
TRANSMIT RESULTS,
CLEAR UNIT BUSY
CONTROL DATA CORPORATION
COMPUTER DIVISION
CENTRAL PROCESSOR SCOREBOARD ALL CLEAR
PRODUCT 6601
SIZE C
DRAWING NO. 60119300
REV C
SHEET 71
51

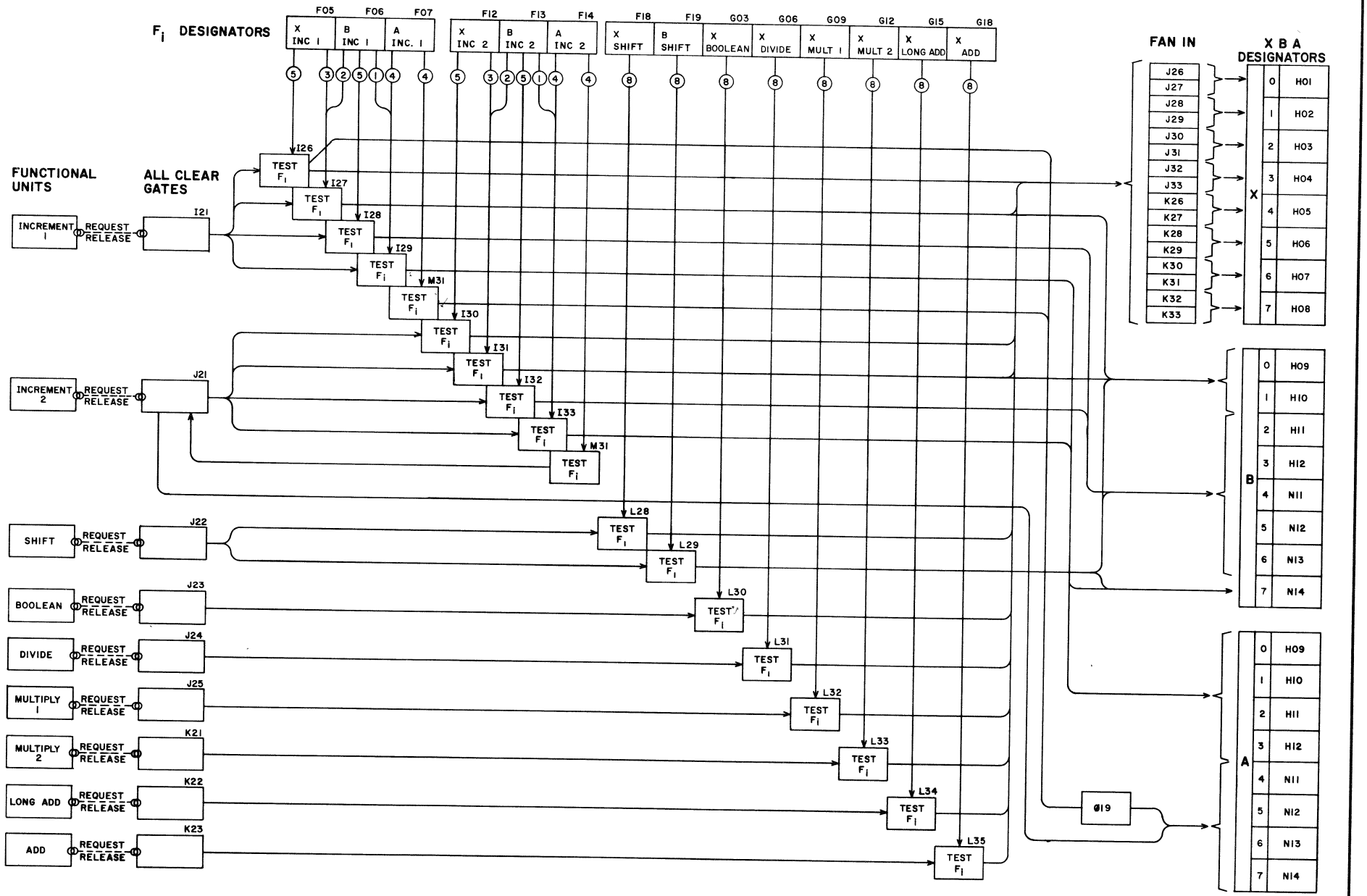


from Test & Net.

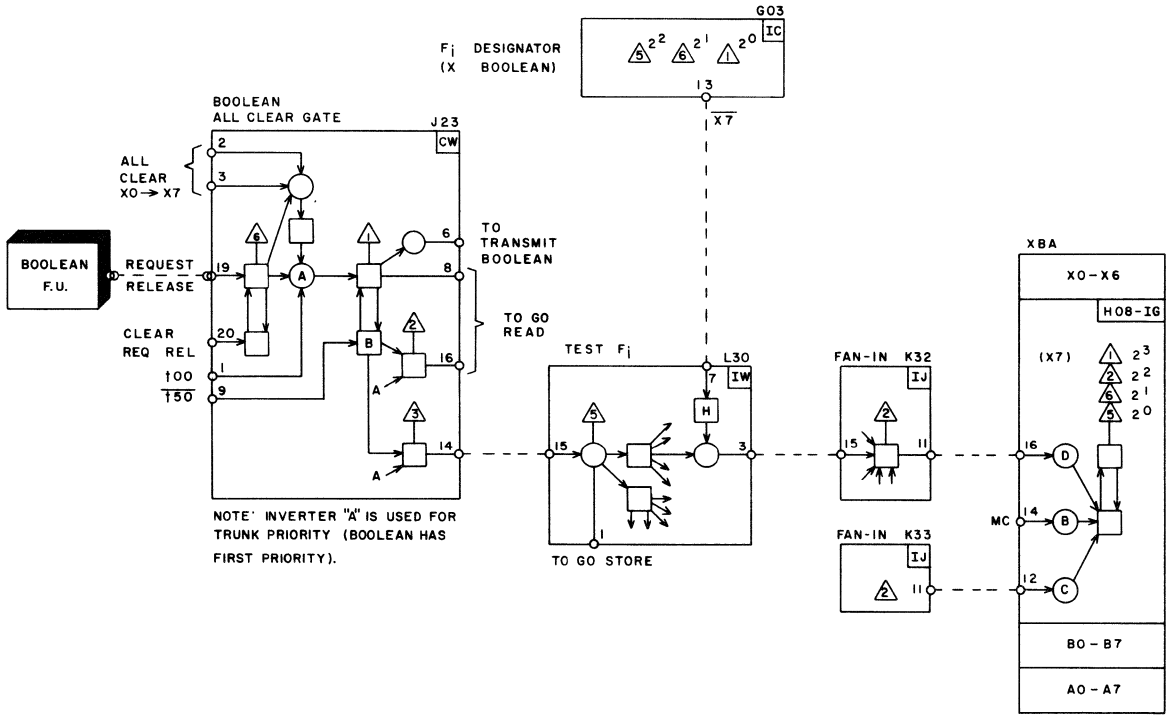
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR SCOREBOARD ALL CLEAR X1	PRODUCT: 6601/04	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: K
			SHEET: 72 53



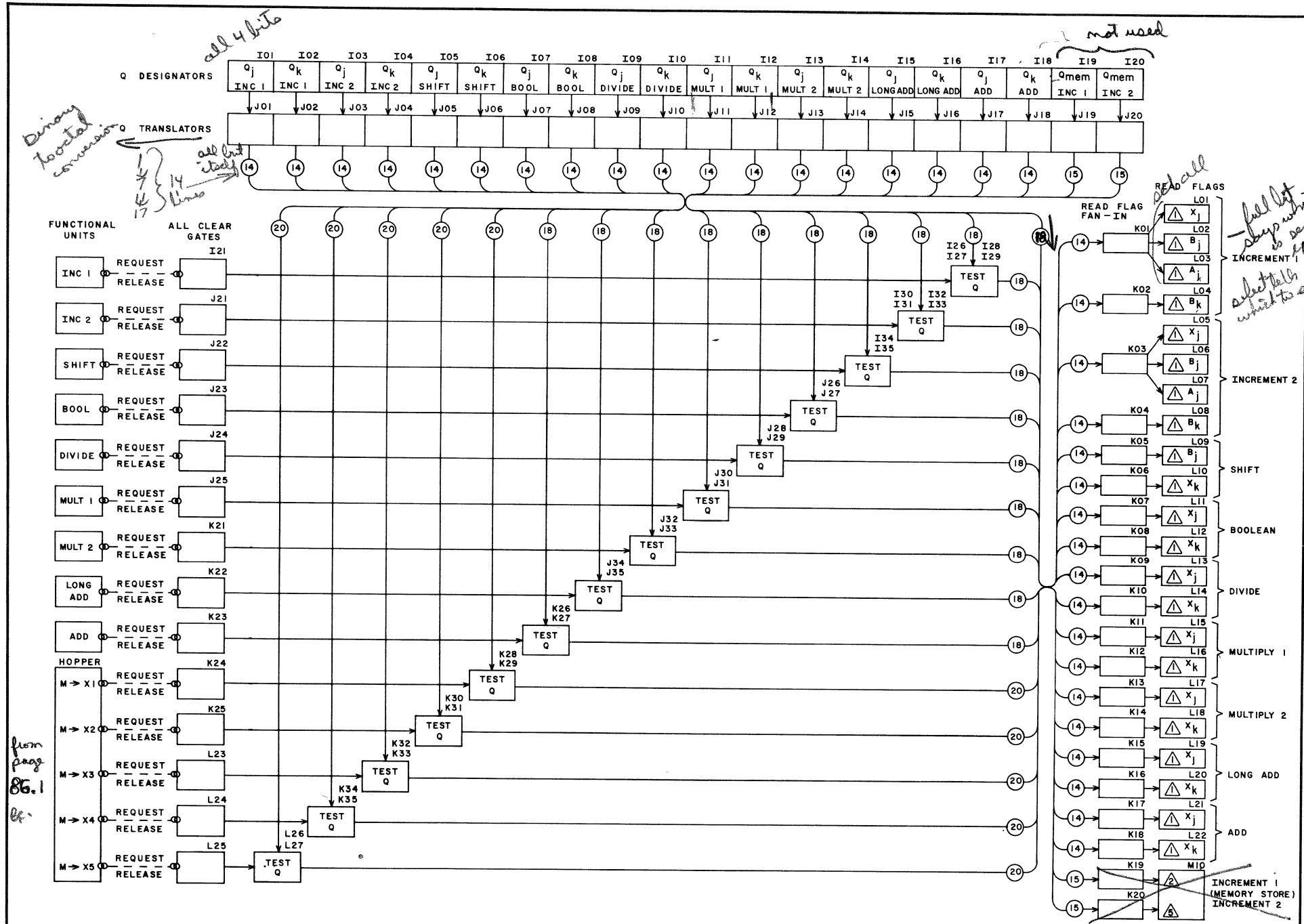
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR SCOREBOARD REQUEST RELEASE (BOOLEAN) TO GO STORE (BOOLEAN)	PRODUCT: 6601	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: AG
			SHEET: 73 55



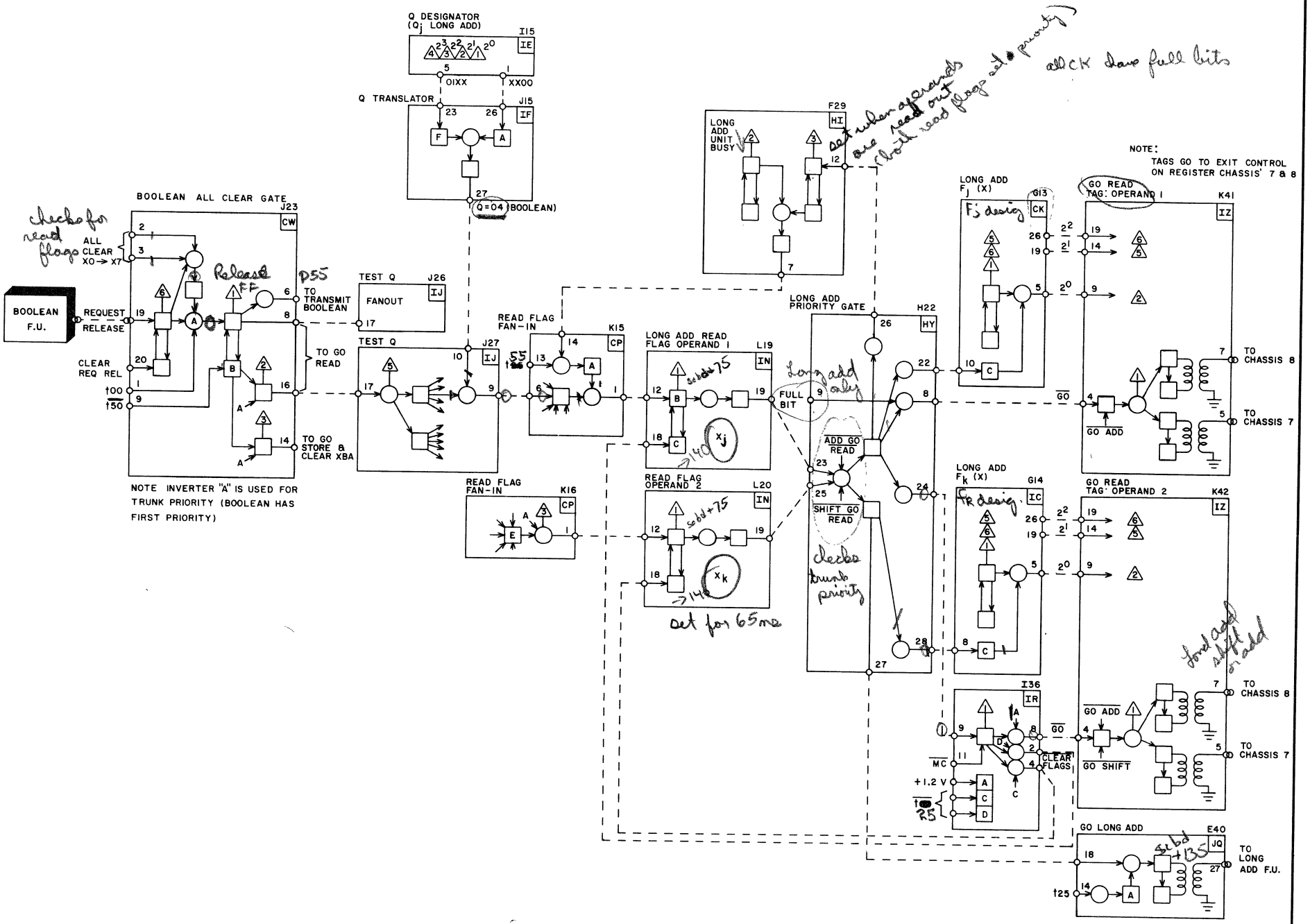
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR SCOREBOARD REQUEST RELEASE TO CLEAR XBA	PRODUCT: 6601/04	
COMPUTER DIVISION		SIZE: C · DRAWING NO.: 60119300	REV: K
		SHEET: 74	57



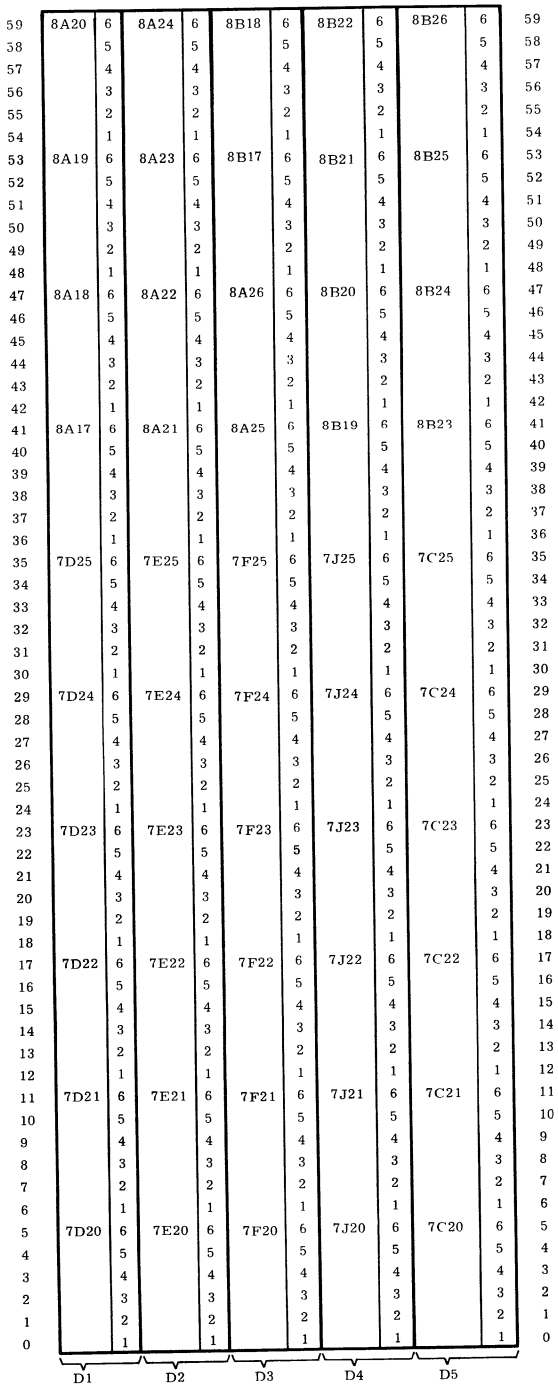
CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR SCOREBOARD REQUEST RELEASE (BOOLEAN) TO CLEAR XBA (X7)	PRODUCT 6601	
		SIZE C	DRAWING NO. 60119300 REV D
		SHEET 75	59



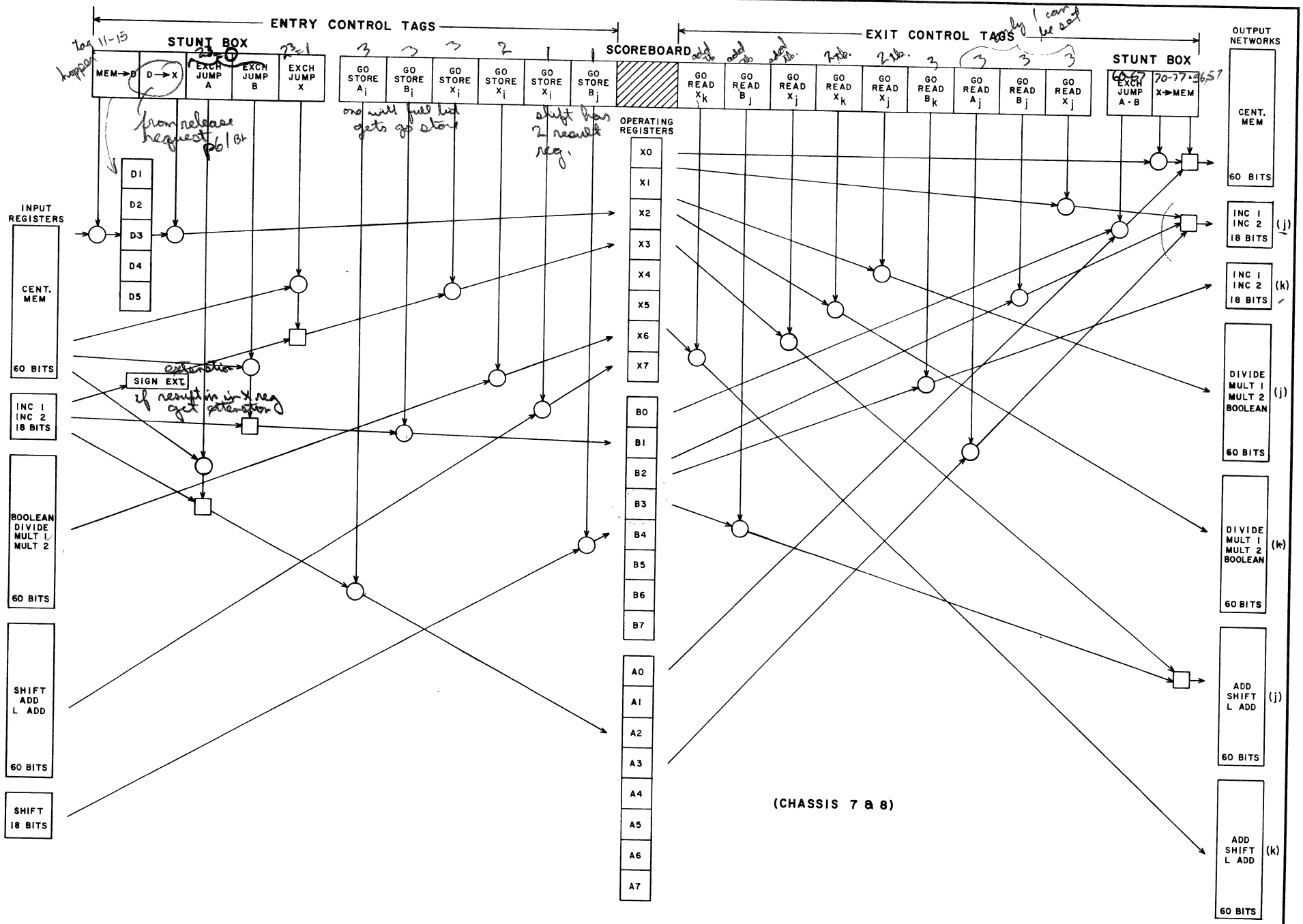
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR SCOREBOARD REQUEST RELEASE TO SET READ FLAGS	PRODUCT: 6601/04	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: K
		SHEET: 76	61



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR SCOREBOARD REQUEST RELEASE (BOOLEAN) TO GO READ (LONG ADD)	PRODUCT 6601/04/13/14	
		SIZE C	DRAWING NO. 60119300 REV M
			SHEET 77 63



6601/04 CENTRAL PROCESSOR
D REGISTERS
BIT LOCATIONS & TEST POINTS
PUB. NO. 60119300 REV. K 64



CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR OPERATING REGISTERS, ENTRY AND EXIT CONTROL BLOCK DIAGRAM	PRODUCT: 6601/04	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: K
			SHEET: 81 65

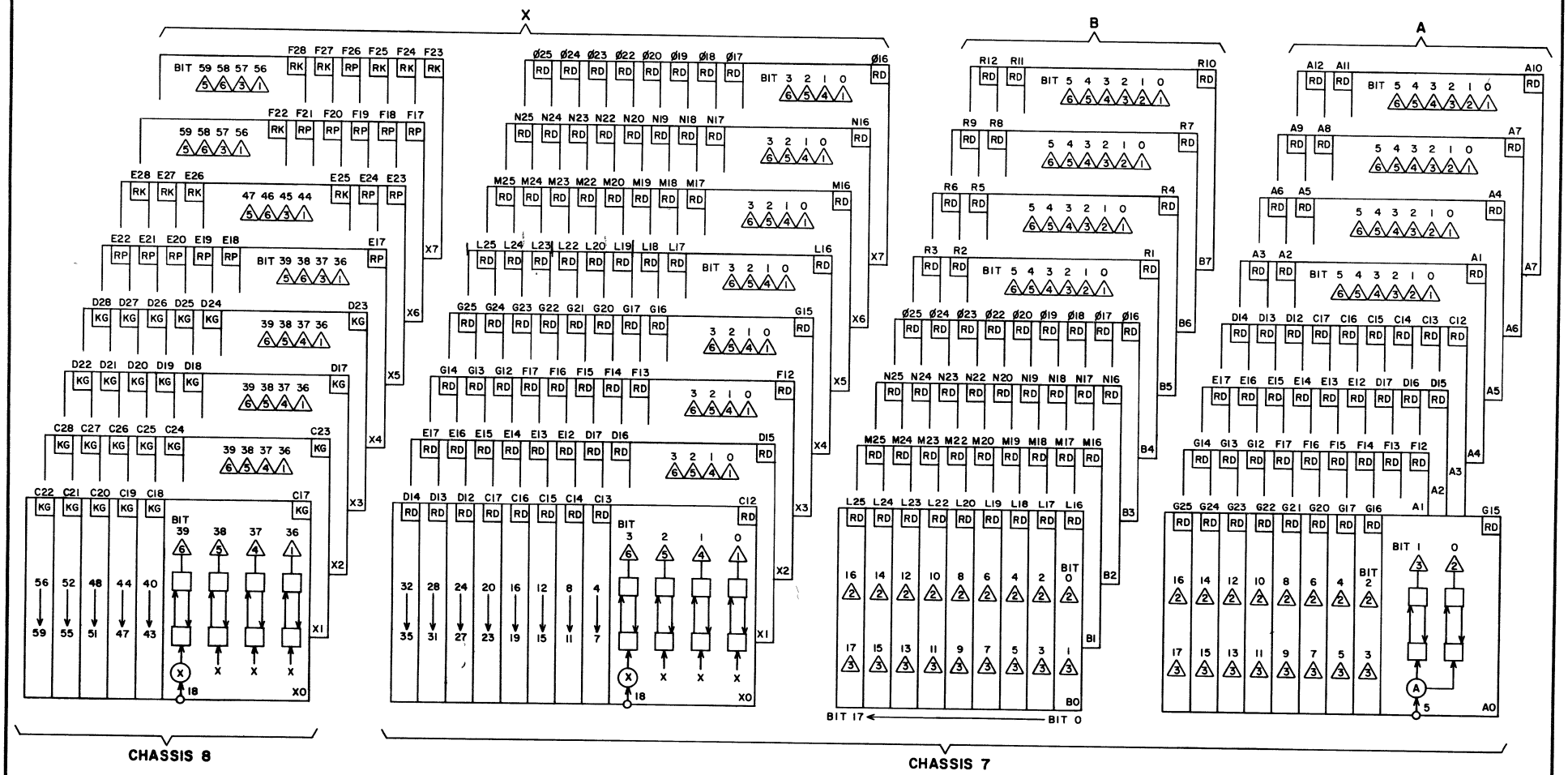
Bit	Module-TP		Module-TP		Module-TP		Module-TP		Bit	Module-TP		Module-TP		Module-TP		Module-TP		Bit
59	8C22	6	8C28	6	8D22	6	8D28	6	59	8E22	5	8E28	5	8F22	5	8F28	5	59
58		5		5		5		5	58		6		6		6		6	58
57		4		4		4		4	57		3		3		3		3	57
56		1		1		1		1	56		1		1		1		1	56
55	8C21	6	8C27	6	8D21	6	8D27	6	55	8E21	5	8E27	5	8F21	5	8F27	5	55
54		5		5		5		5	54		6		6		6		6	54
53		4		4		4		4	53		3		3		3		3	53
52		1		1		1		1	52		1		1		1		1	52
51	8C20	6	8C26	6	8D20	6	8D26	6	51	8E20	5	8E26	5	8F20	5	8F26	5	51
50		5		5		5		5	50		6		6		6		6	50
49		4		4		4		4	49		3		3		3		3	49
48		1		1		1		1	48		1		1		1		1	48
47	8C19	6	8C25	6	8D19	6	8D25	6	47	8E19	5	8E25	5	8F19	5	8F25	5	47
46		5		5		5		5	46		6		6		6		6	46
45		4		4		4		4	45		3		3		3		3	45
44		1		1		1		1	44		1		1		1		1	44
43	8C18	6	8C24	6	8D18	6	8D24	6	43	8E18	5	8E24	5	8F18	5	8F24	5	43
42		5		5		5		5	42		6		6		6		6	42
41		4		4		4		4	41		3		3		3		3	41
40		1		1		1		1	40		1		1		1		1	40
39	8C17	6	8C23	6	8D17	6	8D23	6	39	8E17	5	8E23	5	8F17	5	8F23	5	39
38		5		5		5		5	38		6		6		6		6	38
37		4		4		4		4	37		3		3		3		3	37
36		1		1		1		1	36		1		1		1		1	36
35	7D14	6	7E17	6	7G14	6	7G25	6	35	7L25	6	7M25	6	7N25	6	7O25	6	35
34		5		5		5		5	34		5		5		5		5	34
33		4		4		4		4	33		4		4		4		4	33
32		1		1		1		1	32		1		1		1		1	32
31	7D13	6	7E16	6	7G13	6	7G24	6	31	7L24	6	7M24	6	7N24	6	7O24	6	31
30		5		5		5		5	30		5		5		5		5	30
29		4		4		4		4	29		4		4		4		4	29
28		1		1		1		1	28		1		1		1		1	28
27	7D12	6	7E15	6	7G12	6	7G23	6	27	7L23	6	7M23	6	7N23	6	7O23	6	27
26		5		5		5		5	26		5		5		5		5	26
25		4		4		4		4	25		4		4		4		4	25
24		1		1		1		1	24		1		1		1		1	24
23	7C17	6	7E14	6	7F17	6	7G22	6	23	7L22	6	7M22	6	7N22	6	7O22	6	23
22		5		5		5		5	22		5		5		5		5	22
21		4		4		4		4	21		4		4		4		4	21
20		1		1		1		1	20		1		1		1		1	20
19	7C16	6	7E13	6	7F16	6	7G21	6	19	7L20	6	7M20	6	7N20	6	7O20	6	19
18		5		5		5		5	18		5		5		5		5	18
17		4		4		4		4	17		4		4		4		4	17
16		1		1		1		1	16		1		1		1		1	16
15	7C15	6	7E12	6	7F15	6	7G20	6	15	7L19	6	7M19	6	7N19	6	7O19	6	15
14		5		5		5		5	14		5		5		5		5	14
13		4		4		4		4	13		4		4		4		4	13
12		1		1		1		1	12		1		1		1		1	12
11	7C14	6	7D17	6	7F14	6	7G17	6	11	7L18	6	7M18	6	7N18	6	7O18	6	11
10		5		5		5		5	10		5		5		5		5	10
9		4		4		4		4	9		4		4		4		4	9
8		1		1		1		1	8		1		1		1		1	8
7	7C13	6	7D16	6	7F13	6	7G16	6	7	7L17	6	7M17	6	7N17	6	7O17	6	7
6		5		5		5		5	6		5		5		5		5	6
5		4		4		4		4	5		4		4		4		4	5
4		1		1		1		1	4		1		1		1		1	4
3	7C12	6	7D15	6	7F12	6	7G15	6	3	7L16	6	7M16	6	7N16	6	7O16	6	3
2		5		5		5		5	2		5		5		5		5	2
1		4		4		4		4	1		4		4		4		4	1
0		1		1		1		1	0		1		1		1		1	0
	X0		X1		X2		X3			X4		X5		X6		X7		

Bit	Module-TP		Module-TP		Module-TP		Module-TP		Bit	Module-TP		Module-TP		Module-TP		Module-TP		Bit
17	7G25	3	7G14	3	7E17	3	7D14	3	17	7A03	6	7A06	6	7A09	6	7A12	6	17
16		2		2		2		2	16		5		5		5		5	16
15	7G24	3	7G13	3	7E16	3	7D13	3	15		4		4		4		4	15
14		2		2		2		2	14		3		3		3		3	14
13	7G23	3	7G12	3	7E15	3	7D12	3	13		2		2		2		2	13
12		2		2		2		2	12		1		1		1		1	12
11	7G22	3	7F17	3	7E14	3	7C17	3	11	7A02	6	7A05	6	7A08	6	7A11	6	11
10		2		2		2		2	10		5		5		5		5	10
9	7G21	3	7F16	3	7E13	3	7C16	3	9		4		4		4		4	9
8		2		2		2		2	8		3		3		3		3	8
7	7G20	3	7F15	3	7E12	3	7C15	3	7		2		2		2		2	7
6		2		2		2		2	6		1		1		1		1	6
5	7G17	3	7F14	3	7D17	3	7C14	3	5	7A01	6	7A04	6	7A07	6	7A10	6	5
4		2		2		2		2	4		5		5		5		5	4
3	7G16	3	7F13	3	7D16	3	7C13	3	3		4		4		4		4	3
2		2		2		2		2	2		3		3		3		3	2
1	7G15	3	7F12	3	7D15	3	7C12	3	1		2		2		2		2	1
0		2		2		2		2	0		1		1		1		1	0
	A0		A1		A2		A3			A4		A5		A6		A7		

Bit Locations & Test Points: $A_0 \rightarrow A_7$ Registers (18 Bit)
$B_0 \rightarrow B_7$ Registers (18 Bit)
$X_0 \rightarrow X_7$ Registers (60 Bit)

Bit	Module-TP		Module-TP		Module-TP		Module-TP		Bit	Module-TP		Module-TP		Module-TP		Module-TP		Bit
17	7L25	3	7M25	3	7N25	3	7O25	3	17	7R03	6	7R06	6	7R09	6	7R12	6	17
16		2		2		2		2	16		5		5		5		5	16
15	7L24	3	7M24	3	7N24	3	7O24	3	15		4		4		4		4	15
14		2		2		2		2	14		3		3		3		3	14
13	7L23	3	7M23	3	7N23	3	7O23	3	13		2		2		2		2	13
12		2		2		2		2	12		1		1		1		1	12
11	7L22	3	7M22	3	7N22	3	7O22	3	11	7R02	6	7R05	6	7R08	6	7R11	6	11
10		2		2		2		2	10		5		5		5		5	10
9	7L20	3	7M20	3	7N20	3	7O20	3	9		4		4		4		4	9
8		2		2		2		2	8		3		3		3		3	8
7	7L19	3	7M19	3	7N19	3	7O19	3	7		2		2		2		2	7
6		2		2		2		2	6		1		1		1		1	6
5	7L18	3	7M18	3	7N18	3	7O18	3	5	7R01	6	7R04	6	7R07	6	7R10	6	5
4		2		2		2		2	4		5		5		5		5	4
3	7L17	3	7M17	3	7N17	3	7O17	3	3		4		4		4		4	3
2		2		2		2		2	2		3		3		3		3	2
1	7L16	3	7M16	3	7N16	3	7O16	3	1		2		2		2		2	1
0		2		2		2		2	0		1		1		1		1	0
	B0		B1		B2		B3			B4		B5		B6		B7		

6601/04 CENTRAL PROCESSOR
X,B AND A OPERATING REGISTERS
BIT LOCATIONS AND TEST POINTS
PUB. NO. 60119300 REV. K 66



CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR OPERATING REGISTERS X, B, & A BIT LOCATION	PRODUCT: 6601/04	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: BT
			SHEET: 78 67

Go Store X_i Mult → X		Module	Pin	TP	Module	Pin	TP
	Go	5H38	5	1	7G27	17	4
	0		1	2		16	5
	1		26	5		4	1
	2		22	6		5	2
	Go	5H38	7	1	8G06	17	4
	0		3	2		5	2
	1		28	5		4	1
	2		24	6		16	5

Go Store X_i Add → X		Module	Pin	TP	Module	Pin	TP
	Go	5H39	5	1	7I28	17	4
	0		1	2		16	5
	1		26	5		4	1
	2		22	6		5	2
	Go	5H39	7	1	8G07	17	4
	0		3	2		5	2
	1		28	5		4	1
	2		24	6		16	5

Go Store X_i Inc → X		Module	Pin	TP	Module	Pin	TP
	Go	5H40	5	1	7H28	17	4
	0		1	2		16	5
	1		26	5		4	1
	2		22	6		5	2
	Go	5H40	7	1	8G08	17	4
	0		3	2		5	2
	1		28	5		4	1
	2		24	6		16	5

Go Store B_i Inc → B		Module	Pin	TP	Module	Pin	TP
	Go	5H41	5	1	7K33	17	4
	0		1	2		16	5
	1		26	5		4	1
	2		22	6		5	2

Go Store B_j Shift → B		Module	Pin	TP	Module	Pin	TP
	Go	5I41	5	1	7K34	17	4
	0		1	2		16	5
	1		26	5		4	1
	2		22	6		5	2

Go Store A_i Inc → A		Module	Pin	TP	Module	Pin	TP
	Go	5H42	5	1	7H27	17	4
	0		1	2		16	5
	1		26	5		4	1
	2		22	6		5	2

Score-Board → Entry Control

Go Read X_j X_j → Inc		Module	Pin	TP	Module	Pin	TP
	Go	5K40	5	1	7J27	21	4
	0		1	2		24	5
	1		26	5	7J28	25	1
	2		22	6		10	3

Go Read X_j X_j → Add		Module	Pin	TP	Module	Pin	TP
	Go	5K41	5	1	7J28	7	
	0		1	2		26	
	1		26	5		21	
	2		22	6		24	
	Go	5K41	7	1	8E01	21	4
	0		3	2		5	1
	1		28	5		7	2
	2		24	6		10	3

Go Read X_j X_j → Mult		Module	Pin	TP	Module	Pin	TP
	Go	5J41	5	1	7I26	7	
	0		1	2		26	
	1		26	5		21	
	2		22	6		24	
	Go	5J41	7	1	8F01	21	4
	0		3	2		5	1
	1		28	5		7	2
	2		24	6		10	3

Go Read X_k X_k → Add		Module	Pin	TP	Module	Pin	TP
	Go	5K42	5	1	7I26	5	
	0		1	2		10	
	1		26	5		7	
	2		22	6		26	
	Go	5K42	7	1	8F01	21	4
	0		3	2		5	1
	1		28	5		7	2
	2		24	6		10	3

Go Read X_k X_k → Mult		Module	Pin	TP	Module	Pin	TP
	Go	5J42	5	1	7I26	21	
	0		1	2		24	
	1		26	5	7I27	5	
	2		22	6		10	
	Go	5J42	7	1	8F02	21	4
	0		3	2		5	1
	1		28	5		7	2
	2		24	6		10	3

Score-Board → Exit Control

Go Read B_j B_j → Inc		Module	Pin	TP	Module	Pin	TP
	Go	5L41	5	1	7L34	5	
	0		1	2		10	
	1		26	5		7	
	2		22	6		26	

Go Read B_j B_j → Add		Module	Pin	TP	Module	Pin	TP
	Go	5M41	5	1	7M34	7	
	0		1	2		26	
	1		26	5		21	
	2		22	6		24	

Go Read B_k B_k → Inc		Module	Pin	TP	Module	Pin	TP
	Go	5L42	5	1	7L34	21	
	0		1	2		24	
	1		26	5	7M34	1	
	2		22	6		10	

Go Read A_j A_j → Inc		Module	Pin	TP	Module	Pin	TP
	Go	5M42	5	1	7G26	5	
	0		1	2		10	
	1		26	5		7	
	2		22	6		26	

Score-Board → Exit Control

Memory to D		Module	Pin	TP	Module	Pin	TP
	Go	5I42	5	1	7G28	17	4
	0		1	2		4	1
	1		26	5		5	2
	2		22	6		16	5
	Go	5I42	7	1	8G05	17	4
	0		3	2		5	2
	1		28	5		4	1
	2		24	6		16	5

X to Memory		Module	Pin	TP	Module	Pin	TP
	Go	5O31	5	1	7H26	4	1
	0		1	2		6	2
	1		26	5		7	3
	2		22	6		27	6
	Go	5O31	7	1	8G02	21	4
	0		3	2		5	1
	1		28	5		7	2
	2		24	6		10	3
Exchange		5O33	5	1	7G37	5	1
			7	1	8E01	24	5

Stunt Box → Entry Exit Control

D → X

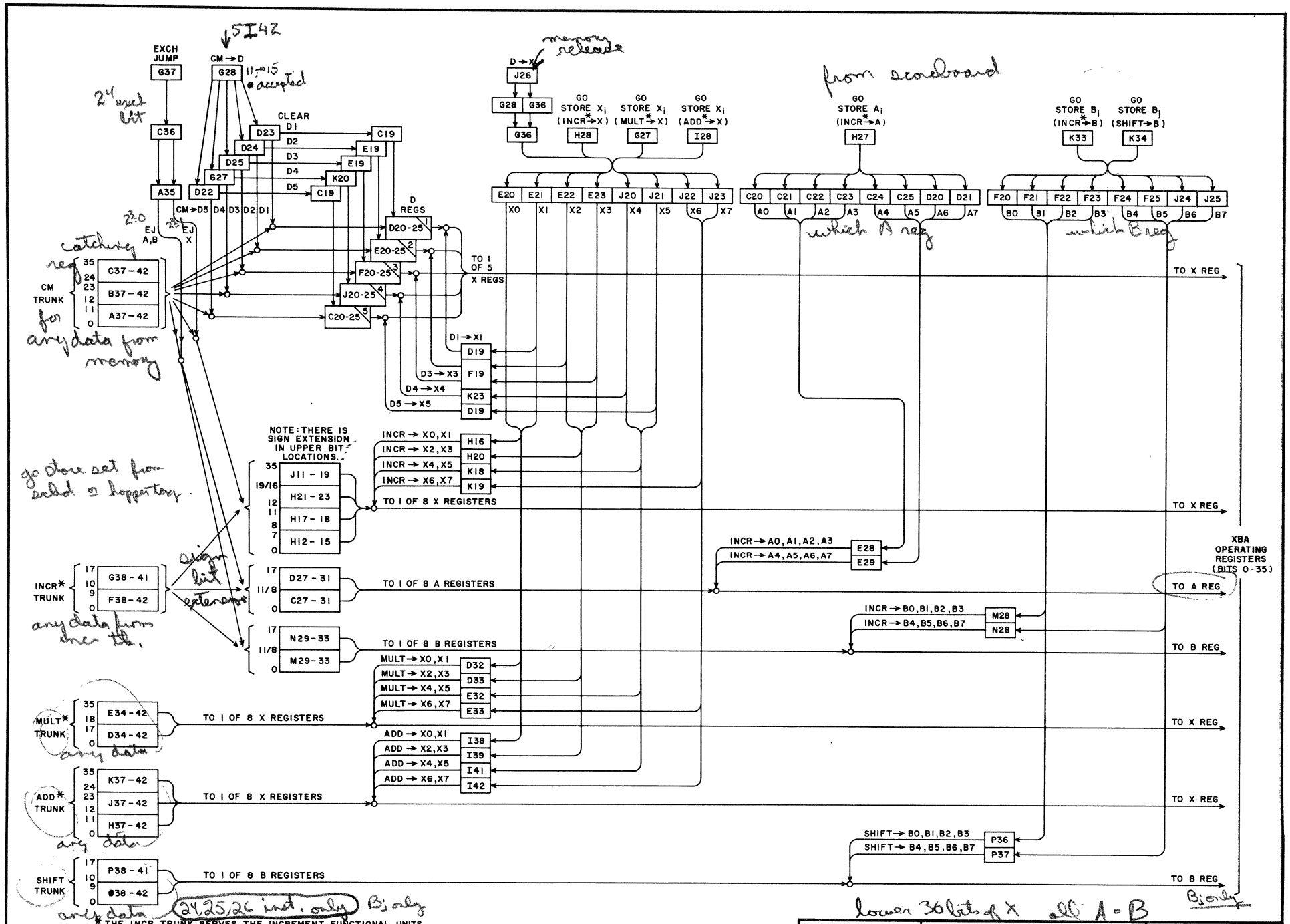
	Module	Pin	TP	Module	Pin	TP	Module	Pin	TP	Module	Pin	TP	Module	Pin	TP	Module	Pin	TP	
D1 → X1	5O34	12	1	7J26	5	1	7J26	9		7L26	9		7L26	8	1	8G11	5	1	D1 → X1
D2 → X2		10			7	2		1			11			6	2		7	2	
D3 → X3		8			10	3		13			13			4	3		10	3	
D4 → X4		6			21	4		18			18			27	4		21	4	
D5 → X5		4			24	5		28			20			25	5		24	5	

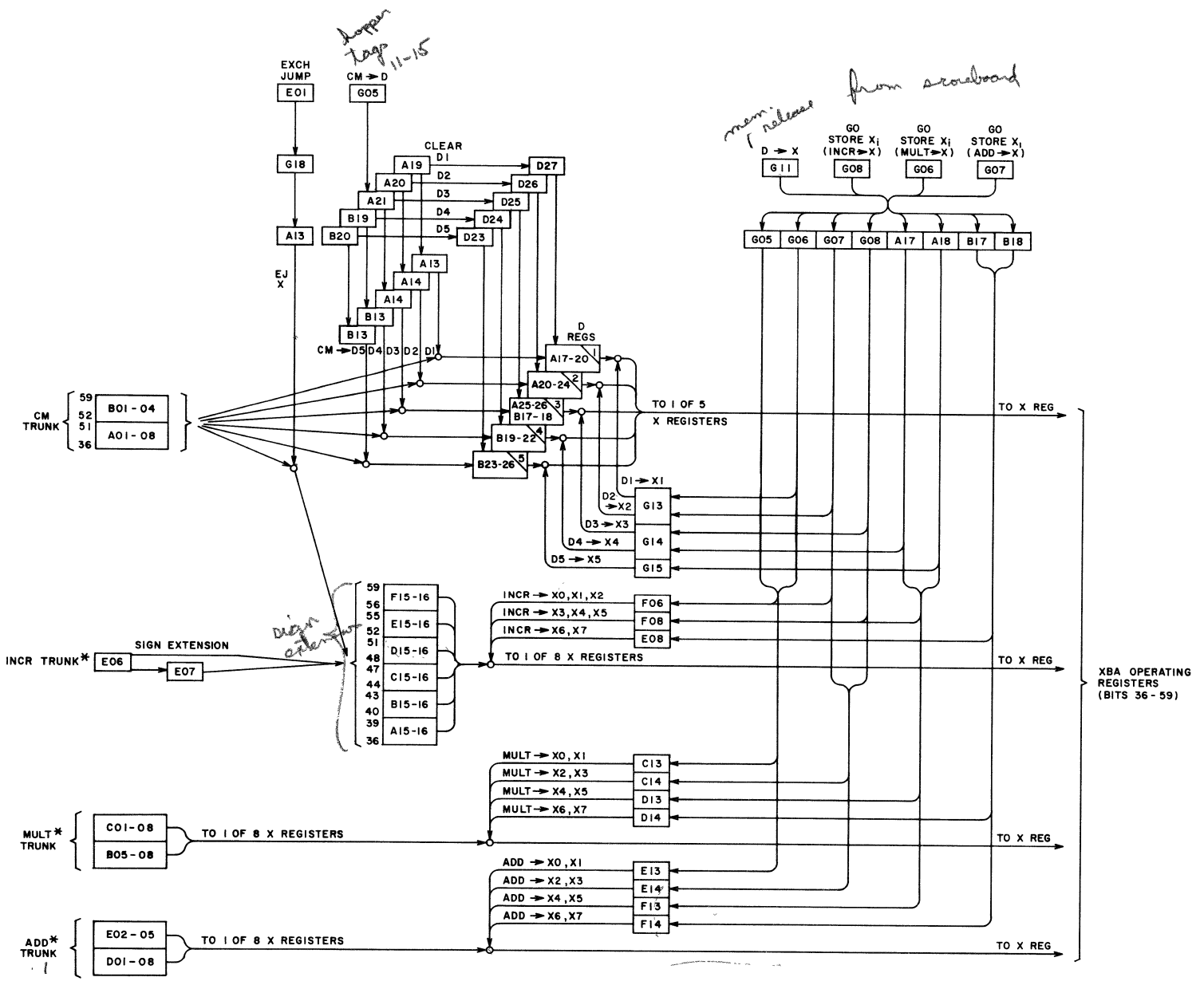
Stunt Box; D → X Signal on CH 7; D → X Signal CH 7 CH 8; D → X Signal on CH 8

6601/04 Central Processor

Go Read, Go Store
MEM → D, D → X, X → MEM.
Exchange.

REV. K 68.0


EXCH JUMP
G37
CM→D
G28
C36
A35
D22
D23
D24
D25
G27
CLEAR
D1
D2
D3
D4
D5
C19
E19
E19
K20
C19
CM→D5 D4 D3 D2 D1
EJ A,B
EJ X
D REGS
D20-25
E20-25
F20-25
J20-25
C20-25
TO 1 OF 5 X REGS
CM TRUNK
C37-42
B37-42
A37-42
D1→X1
D3→X3
D4→X4
D5→X5
D19
F19
K23
D19
NOTE: THERE IS SIGN EXTENSION IN UPPER BIT LOCATIONS.
J11-19
H21-23
H17-18
H12-15
INCR→X0,X1
INCR→X2,X3
INCR→X4,X5
INCR→X6,X7
H16
H20
K18
K19
TO 1 OF 8 X REGISTERS
INCR* TRUNK
G38-41
F38-42
D27-31
C27-31
TO 1 OF 8 A REGISTERS
N29-33
M29-33
TO 1 OF 8 B REGISTERS
MULT* TRUNK
E34-42
D34-42
MULT→X0,X1
MULT→X2,X3
MULT→X4,X5
MULT→X6,X7
D32
D33
E32
E33
ADD* TRUNK
K37-42
J37-42
H37-42
ADD→X0,X1
ADD→X2,X3
ADD→X4,X5
ADD→X6,X7
I38
I39
I41
I42
SHIFT TRUNK
P38-41
#38-42
D→X
J26
G28
G36
GO STORE Xi (INCR*→X)
GO STORE Xi (MULT*→X)
GO STORE Xi (ADD*→X)
H28
G27
I28
E20 E21 E22 E23 J20 J21 J22 J23
X0 X1 X2 X3 X4 X5 X6 X7
GO STORE Ai (INCR*→A)
H27
C20 C21 C22 C23 C24 C25 D20 D21
A0 A1 A2 A3 A4 A5 A6 A7
GO STORE Bj (INCR*→B)
GO STORE Bj (SHIFT→B)
K33
K34
F20 F21 F22 F23 F24 F25 J24 J25
B0 B1 B2 B3 B4 B5 B6 B7
INCR→A0,A1,A2,A3
INCR→A4,A5,A6,A7
E28
E29
INCR→B0,B1,B2,B3
INCR→B4,B5,B6,B7
M28
N28
SHIFT→B0,B1,B2,B3
SHIFT→B4,B5,B6,B7
P36
P37
XBA OPERATING REGISTERS (BITS 0-35)
TO X REG
TO A REG
TO B REG
TO X-REG
THE INCR TRUNK SERVES THE INCREMENT FUNCTIONAL UNITS.
THE MULT TRUNK SERVES THE MULTIPLY, DIVIDE, AND BOOLEAN FUNCTIONAL UNITS.
THE ADD TRUNK SERVES THE ADD, LONG ADD, AND SHIFT FUNCTIONAL UNITS.
CONTROL DATA CORPORATION
COMPUTER DIVISION
TITLE
CENTRAL PROCESSOR
ENTRY CONTROL
CHASSIS 7
BLOCK DIAGRAM
PRODUCT
6601/04
SIZE C
DRAWING NO. 60119300
REV K
SHEET 225
PAGE 68.1



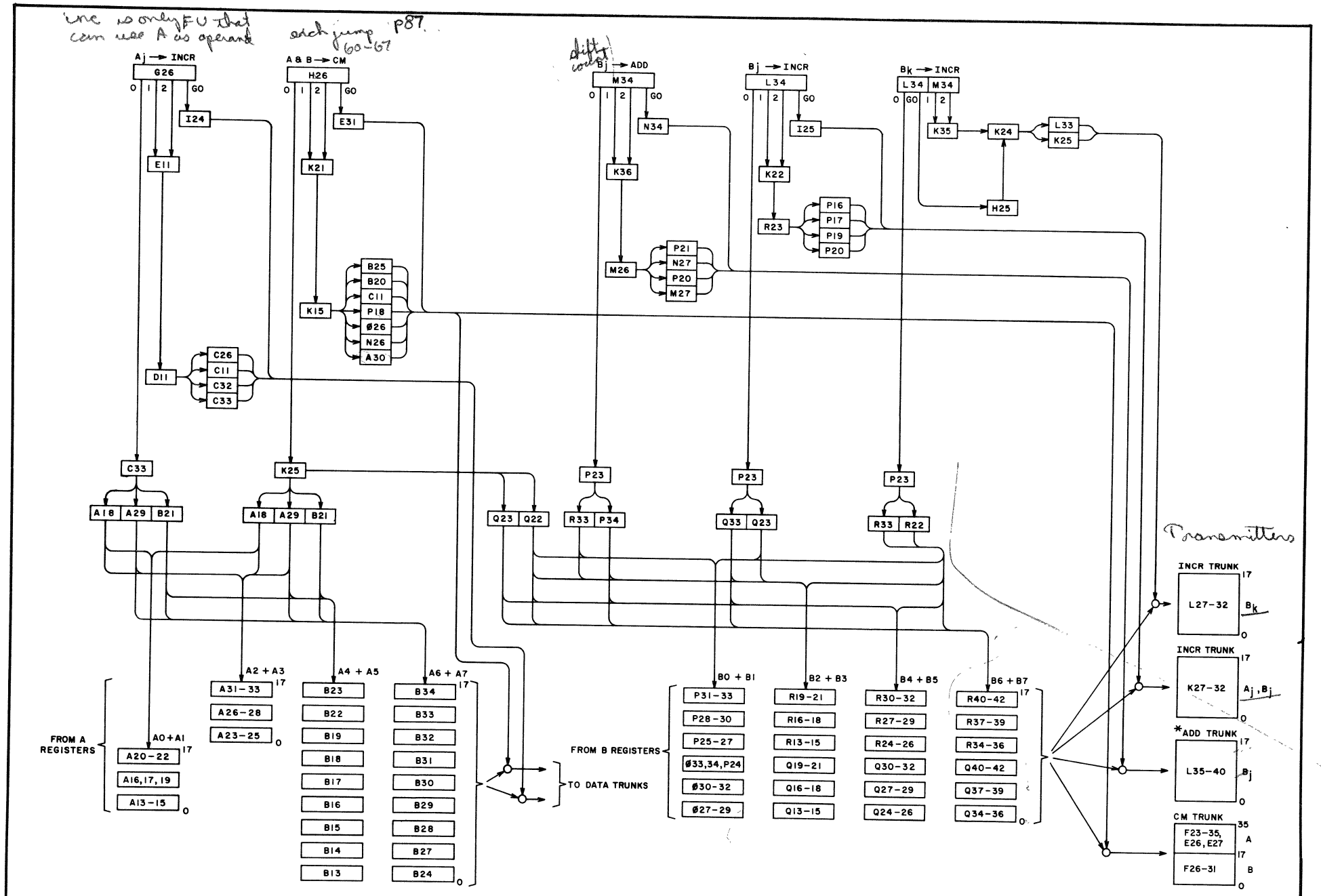
* THE INCR TRUNK SERVES THE INCREMENT FUNCTIONAL UNITS.
THE MULT TRUNK SERVES THE MULTIPLY, DIVIDE, AND BOOLEAN FUNCTIONAL UNITS.
THE ADD TRUNK SERVES THE ADD, LONG ADD, AND SHIFT FUNCTIONAL UNITS.

upper 24 bits of X reg

CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE: CENTRAL PROCESSOR ENTRY CONTROL CHASSIS 8 BLOCK DIAGRAM

PRODUCT: 6601/04
SIZE: C | DRAWING NO.: 60119300 | REV: K
SHEET: 226 | PAGE: 68.3

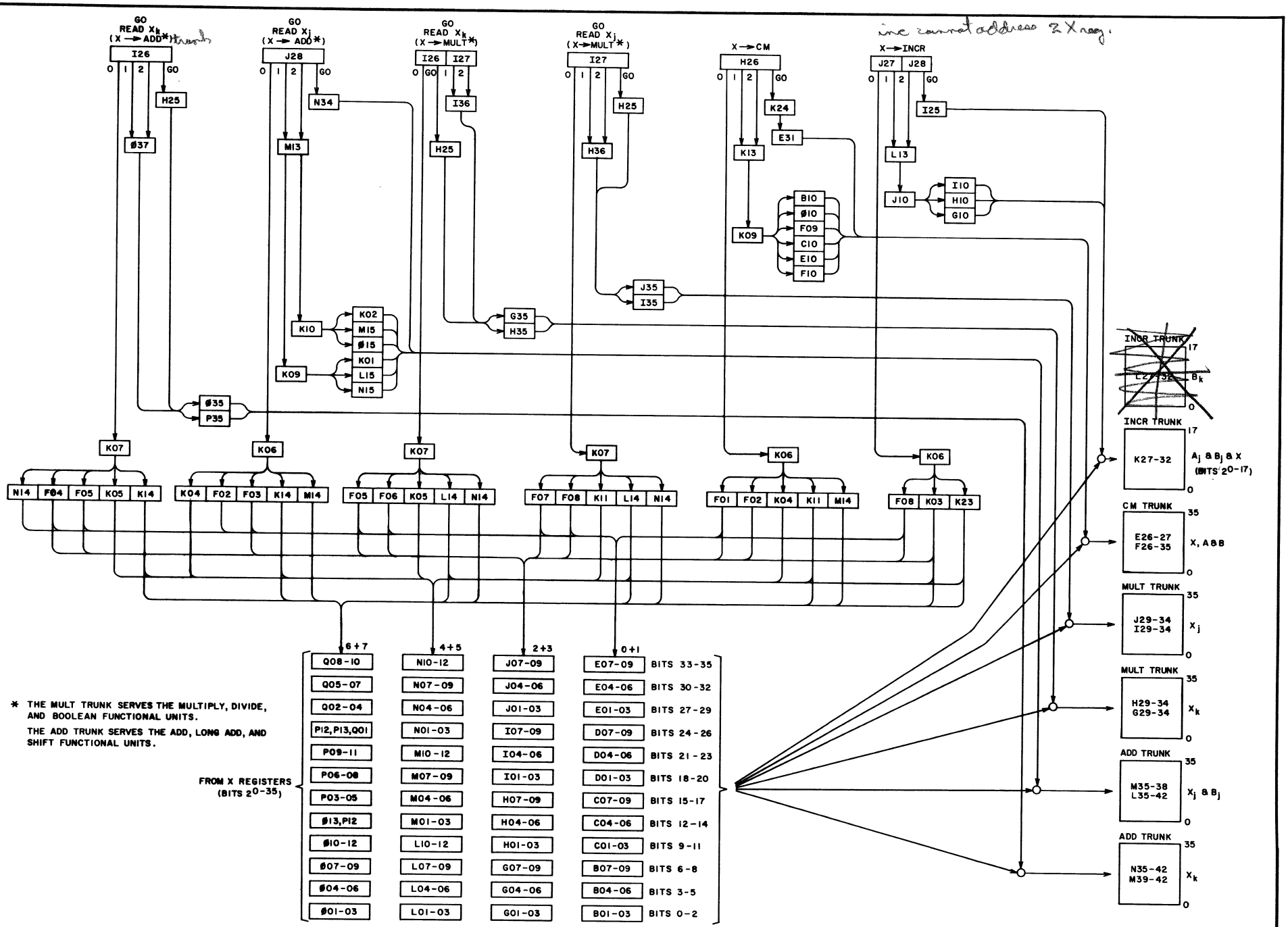


* THE ADD TRUNK SERVES THE ADD, LONG ADD, AND SHIFT FUNCTIONAL UNITS.

CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE: CENTRAL PROCESSOR
EXIT CONTROL, A&B REGISTERS
CHASSIS 7
BLOCK DIAGRAM

PRODUCT: 6601/04
SIZE: C | DRAWING NO.: 60119300 | REV: K
SHEET: 227 | PAGE: 68.5



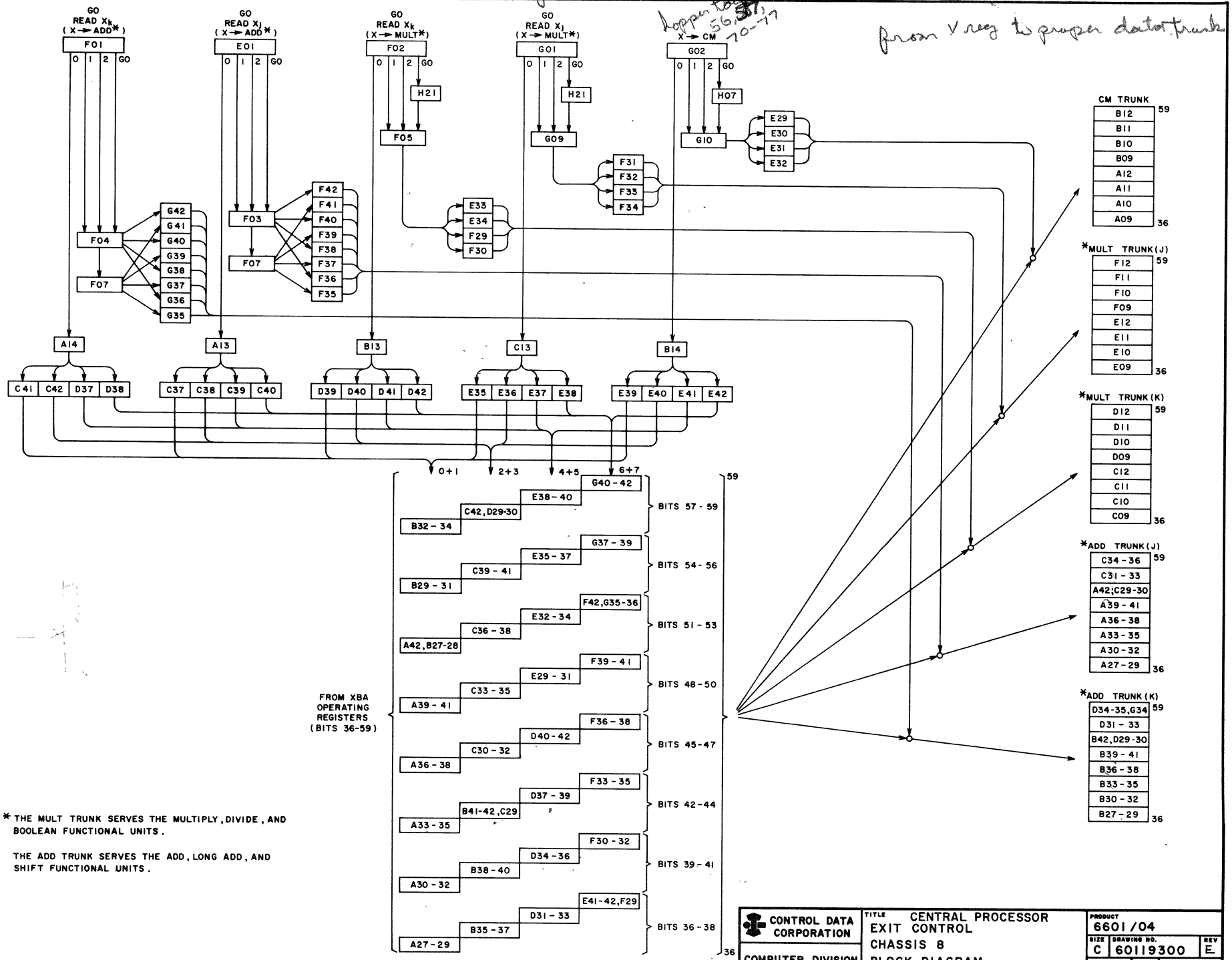
inc cannot address 2 X regs.

CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE: CENTRAL PROCESSOR
EXIT CONTROL, X REGISTERS
CHASSIS 7
BLOCK DIAGRAM

PRODUCT: 6601/04
SIZE: C | DRAWING NO.: 60119300 | REV: BT
SHEET: 229 | PAGE: 68.7

ch 5 fi fr reg.



hopper to 56, 57, 70-77

from V reg to proper dator trunk

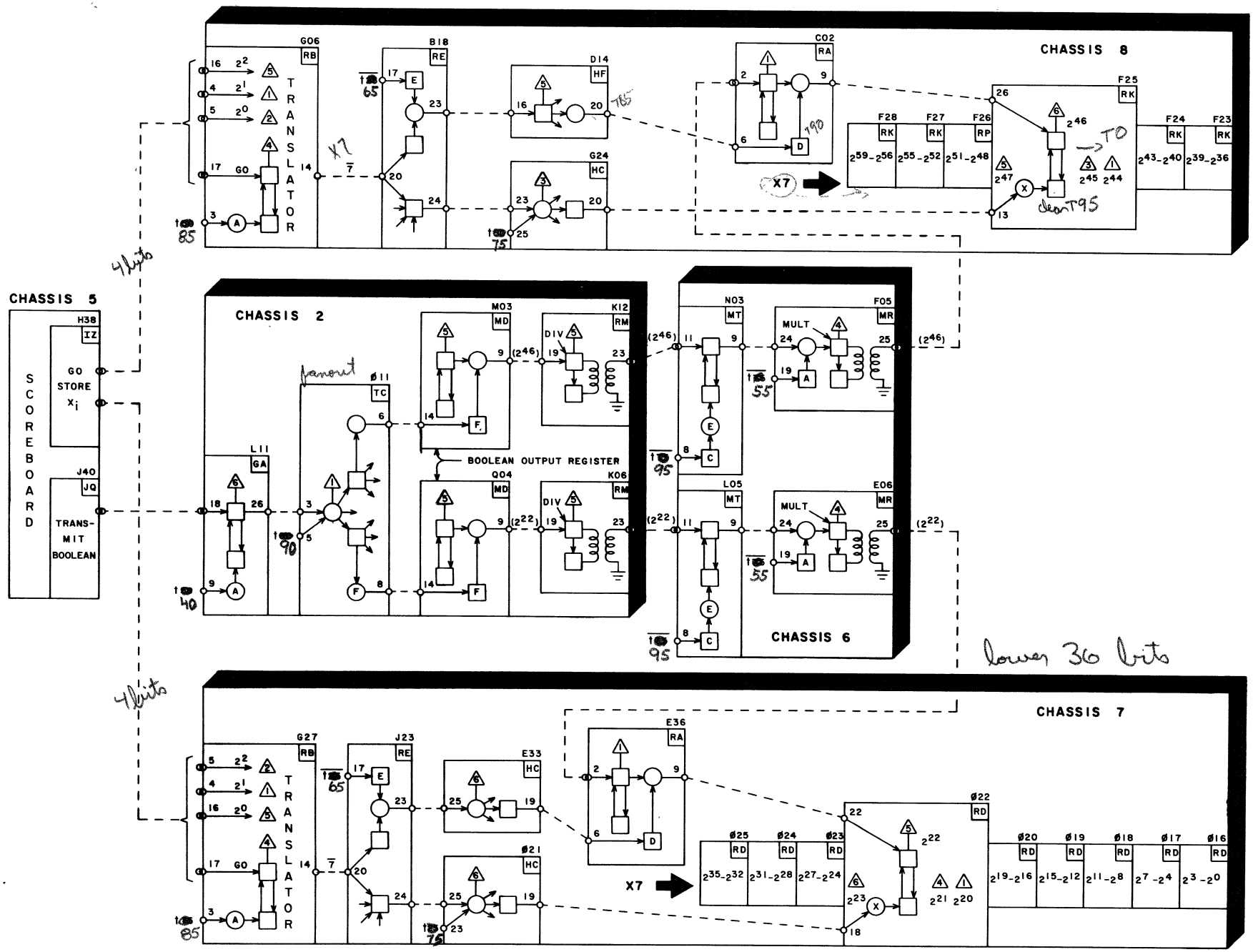
* THE MULT TRUNK SERVES THE MULTIPLY, DIVIDE, AND BOOLEAN FUNCTIONAL UNITS.

THE ADD TRUNK SERVES THE ADD, LONG ADD, AND SHIFT FUNCTIONAL UNITS.

CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE CENTRAL PROCESSOR
EXIT CONTROL
CHASSIS 8
BLOCK DIAGRAM

PRODUCT 6601/04
SIZE C DRAWING NO. 60119300 REV E
SHEET 228 PAGE 68.9



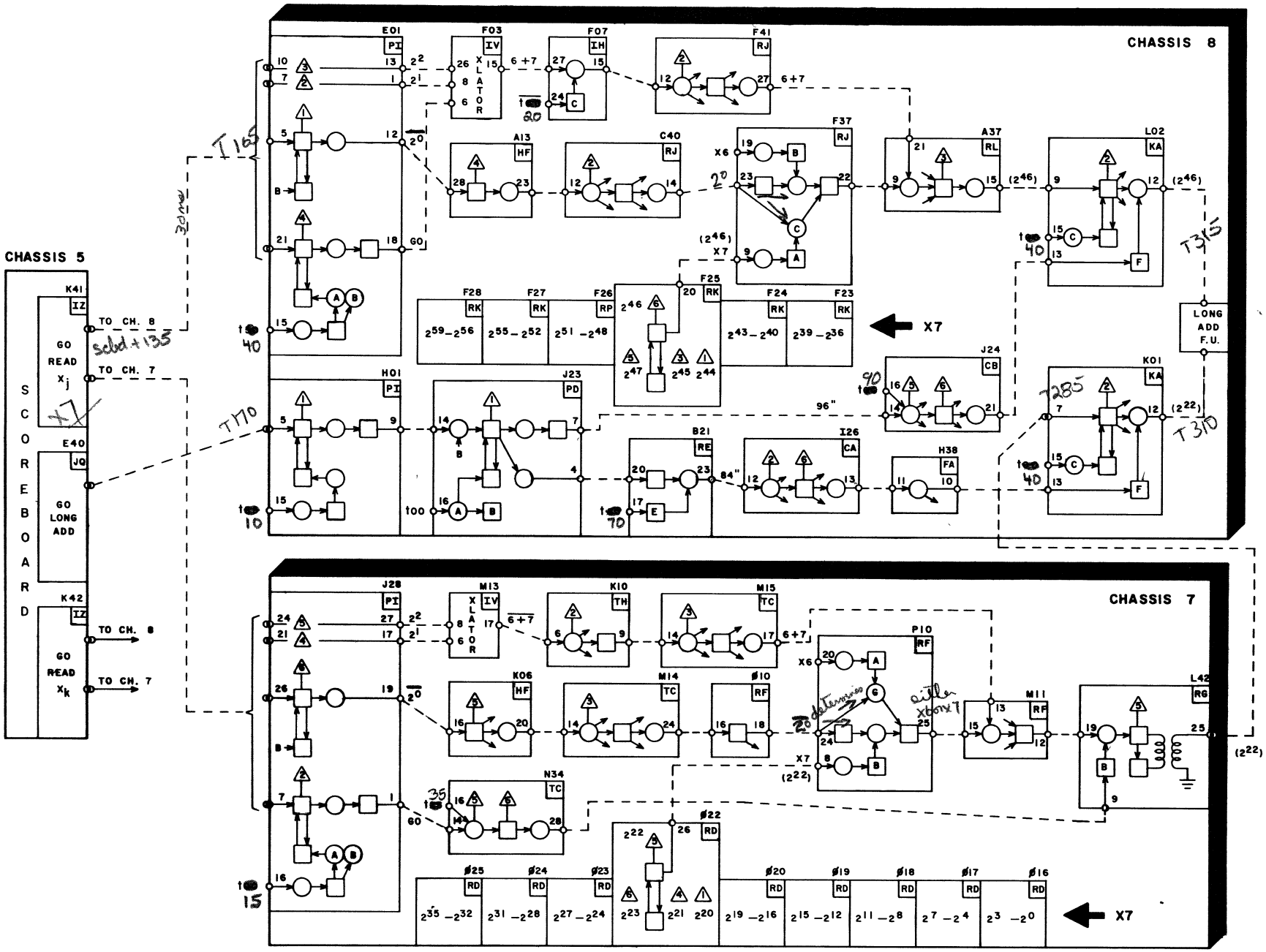
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR REGISTER ENTRY CONTROL TRANSMIT BOOLEAN TO X7	PRODUCT: 6601/04	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: BT
		SHEET: 79	69

			Source Module		TP	Cable Tab	Color	Cable Tab	Destin. Module		TP
Add	Go Add		5E40	25		5W12	91	8W12	8H01	10	3
	Transmit Add		5J40	19	6		903		8G02	24	5
	Request Rel. Add	*	8H06	6		8W22	98	5W26	5K23	19	6
Long Add	Go Long Add		5E40	27		5W12	90	8W12	8H01	5	1
	Transmit Long Add		5J40	21			901		8F01	24	5
	Request Rel. Long Add	*	8H06	8		5W12	906	5W12	5K22	19	6
Boolean	Go Boolean		5E40	21		5W13	902	2W21	2L11	16	4
	Transmit Boolean		5J40	10			99		2L11	18	6
	Request Rel. Boolean	*	2E01	6	2	2W21	906	5W13	5J23	19	6
Divide	Go Divide		5J40	8		5W13	903	2W21	2L11	12	1
	Transmit Divide		5J40	27			900		2L11	14	3
	Request Rel. Divide	*	2E01	8	1	2W21	905	5W13	5J24	19	6
Increment 1	Go Inc. 1		5M08	10		→			5M33	25	
	Transmit Inc. 1		5R32	17/19		→			5F41/42	14	
	Request Rel. Inc. 1	*	5E38	25		→			5I21	19	6
Increment 2	Go Inc. 2		5M08	17		→			5M33	4	
	Transmit Inc. 2		5R32	17/19		→			5F41/42	14	
	Request Rel. Inc. 2	*	5E38	27		→			5J21	19	6
Multiply 1	Go Mult. 1		5J40	4		5W14	90	6W16	6I09	23	6
	Transmit Mult. 1			23			93		6L08	26	6
	Request Rel. Mult. 1	*	6H09	1	2	6W16	95	5W14	5J25	19	6
Multiply 1 Exp	Go Mult. 1 Exp. to Ch. 2		6H10	1	2	6W22	905	2W26	2M11	14	2
	Transmit Mult. 1 Exp.		5O25	10		5W13	907	2W21	2O09	7	2
Multiply 2	Go Mult. 2		5J40	6		5W14	900	6W16	6J09	23	6
	Transmit Mult. 2			25			94		6L08	2	1
	Request Rel. Mult. 2	*	6K09	1	2	6W16	905	5W14	5K21	19	6
Multiply 2 Exp	Go Mult. 2 Exp. to Ch. 2		6K10	1	2	6W22	906	2W26	2M11	16	5
	Transmit Mult. 2 Exp.		5O35	12	1	5W13	908	2W21	2O09	21	5
Shift	Go shift		5E40	23		5W12	907	8W12	8H01	26	6
	Transmit Shift		5J40	12	1		902		8G01	24	5
	Request Rel. Shift	*	8H06	4		8W12	904	5W12	5J22	19	6

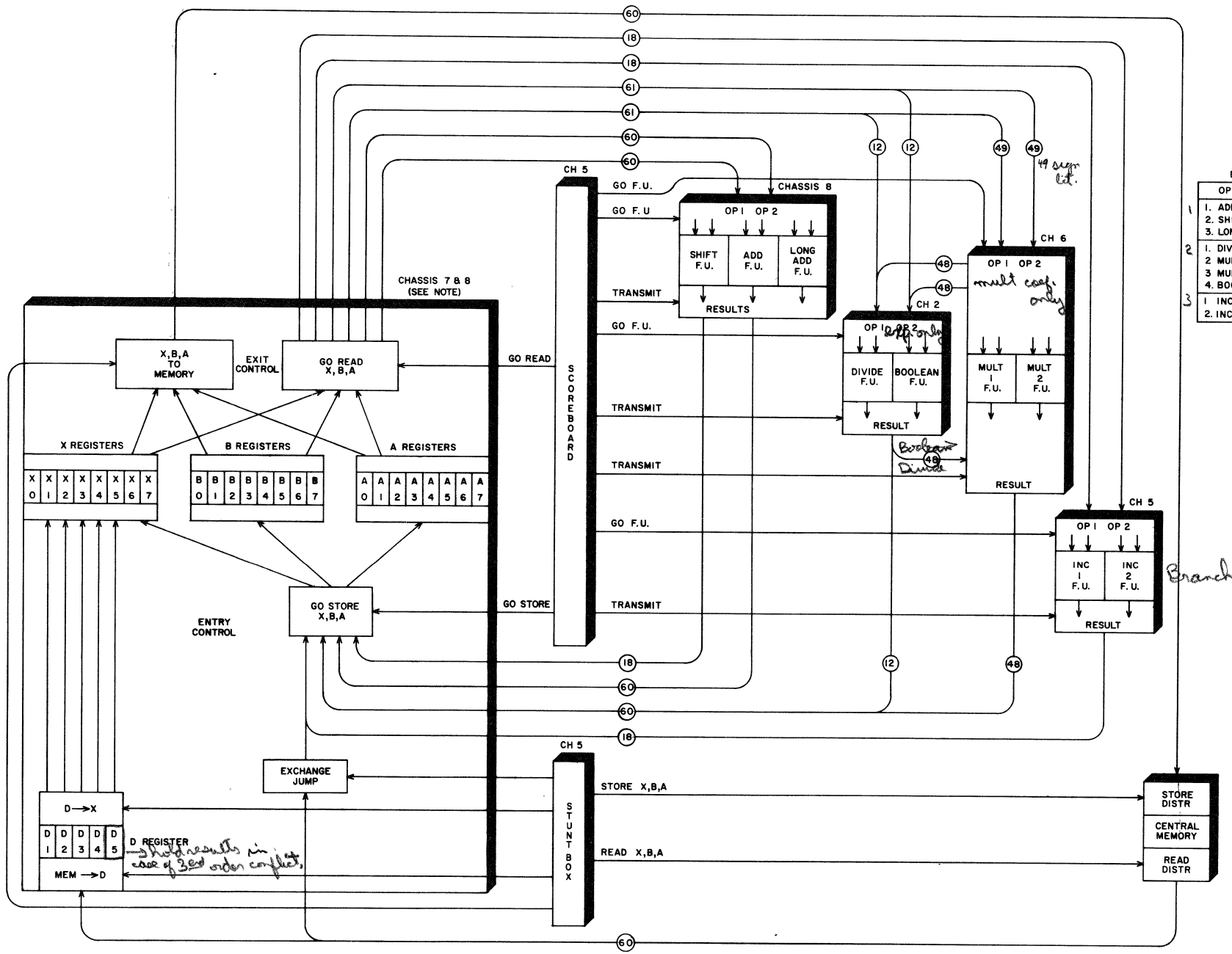
Scoreboard → Functional Unit

* Request release from Function Unit to Scoreboard. Module in Source column is F. U. Module in Destination column is Scoreboard.

6601/04 Central Processor
Go, Transmit, and Req. Rel. for Functional Units.
Pub. No. 60119300
Rev. K 70



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR REGISTER EXIT CONTROL GO READ (X7) TO LONG ADD	PRODUCT 6601/04/13/14	
		SIZE C	DRAWING NO. 60119300 REV BT
			SHEET 80 71



DATA TRUNK PRIORITIES

OPERANDS	RESULTS
1. ADD 2. SHIFT 3. LONG ADD	1. SHIFT 2 ADD 3. LONG ADD
1. DIVIDE 2 MULT 1 3 MULT 2 4. BOOLEAN	1. BOOLEAN 2. DIVIDE 3. MULT 1 4. MULT 2
1 INC 1 2. INC 2	1. INC 1 2. INC 2

NOTE:
THE B AND A REGISTERS ARE LOCATED ON CHASSIS 7 ALONG WITH BITS 0-35 OF THE X REGISTERS.
BITS 36-59 OF THE X REGISTERS ARE LOCATED ON CHASSIS 8.

CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE
CENTRAL PROCESSOR DATA TRUNKS

PRODUCT 6601/04
SIZE C | DRAWING NO. 60119300 | REV K
SHEET 82 | 73

CENTRAL MEMORY DATA TRUNKS

Bit	Memory → X			X → Memory			Bit
	Module-TP			Module-TP		Module-TP	
59	10I18	8B04	5	8B12	5	2A17 6	59
58	10I17		1		6	5	58
57	10I16	8B03	5		1	4	57
56	10I15		1	8B11	5	3	56
55	10I14	8B02	5		6	2	55
54	10I12		1		1	1	54
53	10I11	8B01	5	8B10	5	2A16 6	53
52	10I10		1		6	5	52
51	10I09	8A08	5		1	4	51
50	10I08		1	8B09	5	3	50
49	10I05	8A07	5		6	2	49
48	10I04		1		1	1	48
47	10I03	8A06	5	8A12	5	2A15 6	47
46	10I02		1		6	5	46
45	10I01	8A05	5		1	4	45
44	9I42		1	8A11	5	3	44
43	9I41	8A04	5		6	2	43
42	9I40		1		1	1	42
41	9I39	8A03	5	8A10	5	2A14 6	41
40	9I38		1		6	5	40
39	9I36	8A02	5		1	4	39
38	9I35		1	8A09	5	3	38
37	9I34	8A01	5		6	2	37
36	9I33		1		1	1	36
35	9I32	7C42	5	7E27	6	2A13 6	35
34	9I29		1		5	5	34
33	9I28	7C41	5		1	4	33
32	9I27		1	7E26	6	3	32
31	9I26	7C40	5		5	2	31
30	9I25		1		1	1	30
29	4I18	7C39	5	7F35	6	2A12 6	29
28	4I17		1		5	5	28
27	4I16	7C38	5		1	4	27
26	4I15		1	7F34	6	3	26
25	4I14	7C37	5		5	2	25
24	4I12		1		1	1	24
23	4I11	7B42	5	7F33	6	2A11 6	23
22	4I10		1		5	5	22
21	4I09	7B41	5		1	4	21
20	4I08		1	7F32	6	3	20
19	4I05	7B40	5		5	2	19
18	4I04		1		1	1	18
17	4I03	7B39	5	7F31	6	2A10 6	17
16	4I02		1		5	5	16
15	4I01	7B38	5		1	4	15
14	3I42		1	7F30	6	3	14
13	3I41	7B37	5		5	2	13
12	3I40		1		1	1	12
11	3I39	7A42	5	7F29	6	2A09 6	11
10	3I38		1		5	5	10
9	3I36	7A41	5		1	4	9
8	3I35		1	7F28	6	3	8
7	3I34	7A40	5		5	2	7
6	3I33		1		1	1	6
5	3I32	7A39	5	7F27	6	2A08 6	5
4	3I29		1		5	5	4
3	3I28	7A38	5		1	4	3
2	3I27		1	7F26	6	3	2
1	3I26	7A37	5		5	2	1
0	3I25		1		1	1	0

ADD, SHIFT, AND LONG ADD DATA TRUNKS

Bit	"j" Operand *		"k" Operand *		"i" Result		Bit
	Module-TP	Module-TP	Module-TP	Module-TP	Module-TP	Module-TP	
59	8C36 5	8L08 5	8G34 4	8L08 6	8H16 6	8E05 5	59
58	8C35 5	2	8D35 5	1	5	1	58
57	8C34 5	8L07 5	8D34 5	8L07 6	2	8E04 5	57
56	8C33 5	2	8D33 5	1	1	1	56
55	8C32 5	8L06 5	8D32 5	8L06 6	8H15 6	8E03 5	55
54	8C31 5	2	8D31 5	1	5	1	54
53	8C30 5	8L05 5	8D30 5	8L05 6	2	8E02 5	53
52	8C29 5	2	8D29 5	1	1	1	52
51	8A42 5	8L04 5	8B42 5	8L04 6	8H14 6	8D08 5	51
50	8A41 5	2	8B41 5	1	5	1	50
49	8A40 5	8L03 5	8B40 5	8L03 6	2	8D07 5	49
48	8A39 5	2	8B39 5	1	1	1	48
47	8A38 5	8L02 5	8B38 5	8L02 6	8H13 6	8D06 5	47
46	8A37 5	2	8B37 5	1	5	1	46
45	8A36 5	8L01 5	8B36 5	8L01 6	2	8D05 5	45
44	8A35 5	2	8B35 5	1	1	1	44
43	8A34 5	8K11 5	8B34 5	8K11 6	8H12 6	8D04 5	43
42	8A33 5	2	8B33 5	1	5	1	42
41	8A32 5	8K10 5	8B32 5	8K10 6	2	8D03 5	41
40	8A31 5	2	8B31 5	1	1	1	40
39	8A30 5	8K09 5	8B30 5	8K09 6	8H11 6	8D02 5	39
38	8A29 5	2	8B29 5	1	5	1	38
37	8A28 5	8K08 5	8B28 5	8K08 6	2	8D01 5	37
36	8A27 5	2	8B27 5	1	1	1	36
35	7M38 6	8K07 5	7N42 6	8K07 6	8N04 6	7K42 5	35
34	5	2	5	1	5	1	34
33	1	8K06 5	1	8K06 6	2	7K41 5	33
32	7M37 6	2	7N41 6	1	1	1	32
31	5	8K05 5	5	8K05 6	8N03 6	7K40 5	31
30	1	2	1	1	5	1	30
29	7M36 6	8K04 5	7N40 6	8K04 6	2	7K39 5	29
28	5	2	5	1	1	1	28
27	1	8K03 5	1	8K03 6	8N02 6	7K38 5	27
26	7M35 6	2	7N39 6	1	5	1	26
25	5	8K02 5	5	8K02 6	2	7K37 5	25
24	1	2	1	1	1	1	24
23	7L42 6	8K01 5	7N38 6	8K01 6	8N01 6	7J42 5	23
22	5	2	5	1	5	1	22
21	1	8J11 5	1	8J11 6	2	7J41 5	21
20	7L41 6	2	7N37 6	1	1	1	20
19	5	8J10 5	5	8J10 6	8M05 6	7J40 5	19
18	1	2	1	1	5	1	18
17	7L40 6	8I17 6	7N36 6	8J09 6	2	7J39 5	17
16	5	8J09 2	5	1	1	1	16
15	1	8J08 5	1	8J08 6	8M04 6	7J38 5	15
14	7L39 6	2	7N35 6	1	5	1	14
13	5	8J07 5	5	8J07 6	2	7J37 5	13
12	1	2	1	1	1	1	12
11	7L38 6	8J06 5	7M42 6	8J06 6	8M03 6	7H42 5	11
10	5	2	5	1	5	1	10
9	1	8J05 5	1	8J05 6	2	7H41 5	9
8	7L37 6	2	7M41 6	1	1	1	8
7	5	8J04 5	5	8J04 6	8M02 6	7H40 5	7
6	1	2	1	1	5	1	6
5	7L36 6	8L11	7M40 6	8J03 4	2	7H39 5	5
4	5		5	3	1	1	4
3	1	8L10	1	8J02 4	8M01 6	7H38 5	3
2	7L35 6		7M39 6	3	5	1	2
1	5	8L09	5	8J01 4	2	7H37 5	1
0	1		1	3	1	1	0

Bit	"j" Result				Bit
	Module-TP		Module-TP		
17	8H18 6	7P41 5			17
16	5	1			16
15	2	7P40 5			15
14	1	1			14
13	8H17 6	7P39 5			13
12	5	1			12
11	2	7P38 5			11
10	1	1			10
9	8H05 2	7O42 5			9
8	1	1			8
7	8H04 6	7O41 5			7
6	5	1			6
5	2	7O40 5			5
4	1	1			4
3	8H03 6	7O39 5			3
2	5	1			2
1	2	7O38 5			1
0	1	1			0

INCREMENT DATA TRUNKS

Bit	"j" Operand		"k" Operand		Bit
	Module-TP	Module-TP	Module-TP	Module-TP	
17	7K32 6	5O30 2	7L32 6	5O30 1	17
16	5	5	5	6	16
15	1	5O29 2	1	5O29 1	15
14	7K31 6	5	5	7L31 6	14
13	5	5O28 2	5	5O28 1	13
12	1	5	1	6	12
11	7K30 6	5O27 2	7L30 6	5O27 1	11
10	5	5	5	6	10
9	1	5O26 2	1	5O26 1	9
8	7K29 6	5	7L29 6	6	8
7	5	5O25 2	5	5O25 1	7
6	1	5	1	6	6
5	7K28 6	5O24 2	7L28 6	5O24 1	5
4	5	5	5	6	4
3	1	5O23 2	1	5O23 1	3
2	7K27 6	5	7L27 6	6	2
1	5	5O22 2	5	5O22 1	1
0	1	5	1	6	0

Bit	"i" Result			Bit
	Module-TP		Module-TP	
17	5F42		7G41 5	17
16			1	16
15		6	7G40 5	15
14		1	1	14
13			7G39 5	13
12			1	12
11			7G38 5	11
10			1	10
9	5F41		7F42 5	9
8			1	8
7			7F41 5	7
6			1	6
5		6	7F40 5	5
4		1	1	4
3			7F39 5	3
2			1	2
1			7F38 5	1
0			1	0

* The Exponent Bits (48-59) for both floating add operands do not use the data trunks as defined herein

6601/04 CENTRAL PROCESSOR
DATA TRUNK TABLE
PART I
PUB. NO. 60119300
REV. K 74

MULTIPLY, DIVIDE, AND BOOLEAN DATA TRUNKS

	Bit	"j" Operand (A) Module-TP		Module-TP		"j" Operand (B) Module-TP		Module-TP		Bit	"k" Operand (A) Module-TP		Module-TP		"k" Operand (B) Module-TP		Module-TP		Bit	"i" Result (A) Module-TP		Module-TP		"i" Result (B) Module-TP		Module-TP		Bit
EXPONENT	59	8F12	5	6F06	4	8F12	5	2L10	6	59	8D12	5	6F07	4	−8D12	5	2M10	6	59	2K15	6	→		→		8C08	5	59
	58		6	→		→			5	58		6			→			5	58		5			→			1	58
	57		1						2	57		1						2	57		2					8C07	5	57
	56	8F11	5	→		→			1	56	8D11	5	→		→			1	56		1	→		→			1	56
	55		6	→		→		2L09	6	55		6	→		→		2M09	6	55	2K14	6	→		→		8C06	5	55
	54		1						5	54		1						5	54		5						1	54
	53	8F10	5	→		→			2	53	8D10	5	→		→			2	53		2	→		→		8C05	5	53
	52		6						1	52		6						1	52		1						1	52
	51		1	→		→		2L08	6	51		1	→		→		2M08	6	51	2K13	6	→		→		8C04	5	51
	50	8F09	5	→		→			5	50	8D09	5	→		→			5	50		5	→		→			1	50
	49		6	→		→			2	49		6			→			2	49		2			→		8C03	5	49
	48		1						1	48		1						1	48		1						1	48
COEFFICIENT	47	8E12	5	6J06	6	6N07		2L07	6	47	8C12	5	6H06	6	6M02	1	2M07	6	47	2K12	6	6N03		6F05	6	8C02	5	47
	46		6		5				5	46		6		5		2		5	46		5			5		1	46
	45		1		2				2	45		1		2		3		2	45		2	6N02		2	8C01	5	45
	44	8E11	5		1				1	44	8C11	5		1	6M01	1		1	44		1			1		1	44
	43		6	6J05	6			2L06	6	43		6	6H05	6		2	2M06	6	43	2K11	6	6N01		6F04	6	8B08	5	43
	42		1		5				5	42		1		5		3		5	42		5			5		1	42
	41	8E10	5		2	6N06	6		2	41	8C10	5		2	6L07	1		2	41		2	6M07		2	8B07	5	41
	40		6		1		5		1	40		6		1		2		1	40		1			1		1	40
	39		1	6J04	6		4	2N10	6	39		1	6H04	6		3	2P10	6	39	2K10	6	6M06		6F03	6	8B06	5	39
	38	8E09	5		5		3		5	38	8C09	5		5	6L06	1		5	38		5			5		1	38
	37		6		2		2		2	37		6		2		2		2	37		2	6M05		2	8B05	5	37
	36		1		1		1		1	36		1		1		3		1	36		1			1		1	36
	35	7J34	6	6J03	6	6N05	6	2N09	6	35	7H34	6	6H03	6	6L05	1	2P09	6	35	2K09	6	6M04		6F02	6	7E42	5	35
	34		5		5		5		5	34		5		5		2		5	34		5			5		1	34
	33		1		2		4		2	33		1		2		3		2	33		2	6M03		2	7E41	5	33
	32	7J33	6		1		3		1	32	7H33	6		1	6L04	1		1	32		1			1		1	32
	31		5	6J02	6		2	2N08	6	31		5	6H02	6		2	2P98	6	31	2K08	6	6M02		6F01	6	7E40	5	31
	30		1		5		1		5	30		1		5		3		5	30		5			5		1	30
	29	7J32	6		2	6N04	6		2	29	7H32	6		2	6L03	1		2	29		2	6M01			2	7E39	5	29
	28		5		1		5		1	28		5		1		2		1	28		1				1		1	28
	27		1	6J01	6		4	2N07	6	27		1	6H01	6		3	2P07	6	27	2K07	6	6L07		6E07	6	7E38	5	27
	26	7J31	6		5		3		5	26	7H31	6		5	6L02	1		5	26		5				5		1	26
	25		5		2		2		2	25		5		2		2		2	25		2	6L06			2	7E37	5	25
	24		1		1		1		1	24		1		1		3		1	24		1				1		1	24
	23	7J30	6	6I06	6	6N03	1	2N06	6	23	7H30	6	6G06	6	6L01	1	2P06	6	23	2K06	6	6L05		6E06	6	7E36	5	23
	22		5		5		2		5	22		5		5		2		5	22		5				5		1	22
	21		1		2		3		2	21		1		2		3		2	21		2	6L04			2	7E35	5	21
	20	7J29	6		1	6N02	1		1	20	7H29	6		1	6K07	1		1	20		1				1		1	20
	19		5	6I05	6		2	2Q10	6	19		5	6G05	6		2	2R10	6	19	2K05	6	6L03		6E05	6	7E34	5	19
	18		1		5		3		5	18		1		5		3		5	18		5				5		1	18
	17	7I34	6		2	6N01	1		2	17	7G34	6		2	6K06	1		2	17		2	6L02			2	7D42	5	17
	16		5		1		2		1	16		5		1		2		1	16		1				1		1	16
	15		1	6I04	6		3	2Q09	6	15		1	6G04	6		3	2R09	6	15	2K04	6	6L01		6E04	6	7D41	5	15
	14	7I33	6		5	6M07	1		5	14	7G33	6		5	6K05	1		5	14		5				5		1	14
	13		5		2		2		2	13		5		2		2		2	13		2	6K07			2	7D40	5	13
	12		1		1		3		1	12		1		1		3		1	12		1				1		1	12
	11	7I32	6	6I03	6	6M06	1	2Q08	6	11	7G32	6	6G03	6	6K04	1	2R08	6	11	2K03	6	6K06		6E03	6	7D39	5	11
	10		5		5		2		5	10		5		5		2		5	10		5				5		1	10
	9		1		2		3		2	9		1		2		3		2	9		2	6K05			2	7D38	5	9
	8	7I31	6		1	6M05	1		1	8	7G31	6		1	6K03	1		1	8		1				1		1	8
	7		5	6I02	6		2	2Q07	6	7		5	6G02	6		2	2R07	6	7	2K02	6	6K04		6E02	6	7D37	5	7
	6		1		5		3		5	6		1		5		3		5	6		5				5		1	6
	5	7I30	6		2	6M04	1		2	5	7G30	6		2	6K02	1		2	5		2	6K03			2	7D36	5	5
	4		5		1		2		1	4		5		1		2		1	4		1				1		1	4
	3		1	6I01	6		3	2Q06	6	3		1	6G01	6		3	2R06	6	3	2K01	6	6K02		6E01	6	7D35	5	3
	2	7I29	6		5	6M03	1		5	2	7G29	6		5	6K01	1		5	2		5				5		1	2
	1		5		2		2		2	1		5		2		2		2	1		2	6K01			2	7D34	5	1
	0		1		1		3		1	0		1		1		3		1	0		1				1		1	0

2052

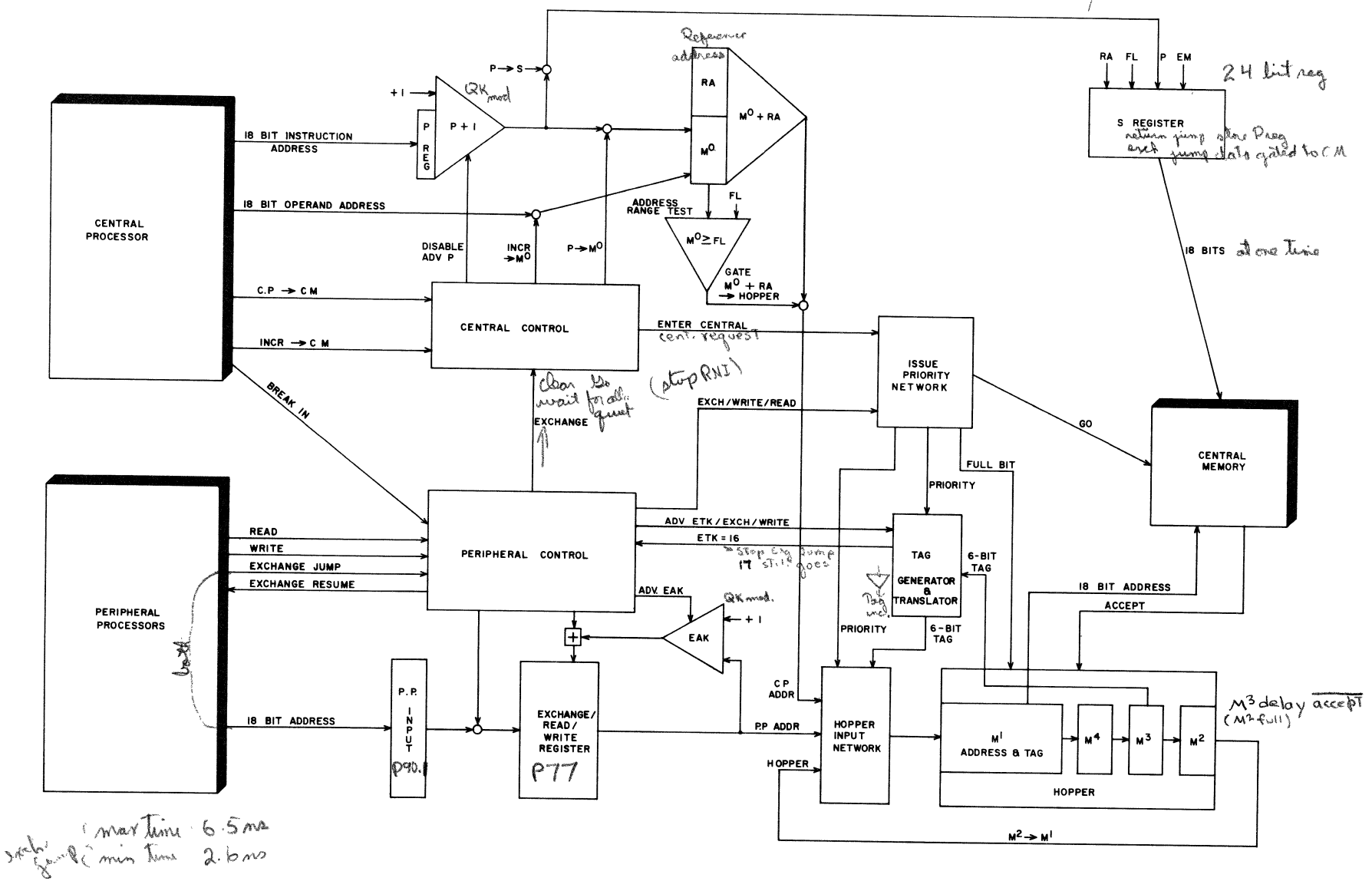
DATA TRUNK TABLE
PART 2
PUB. NO. 60119300
REV. K 74.1

STUNT BOX

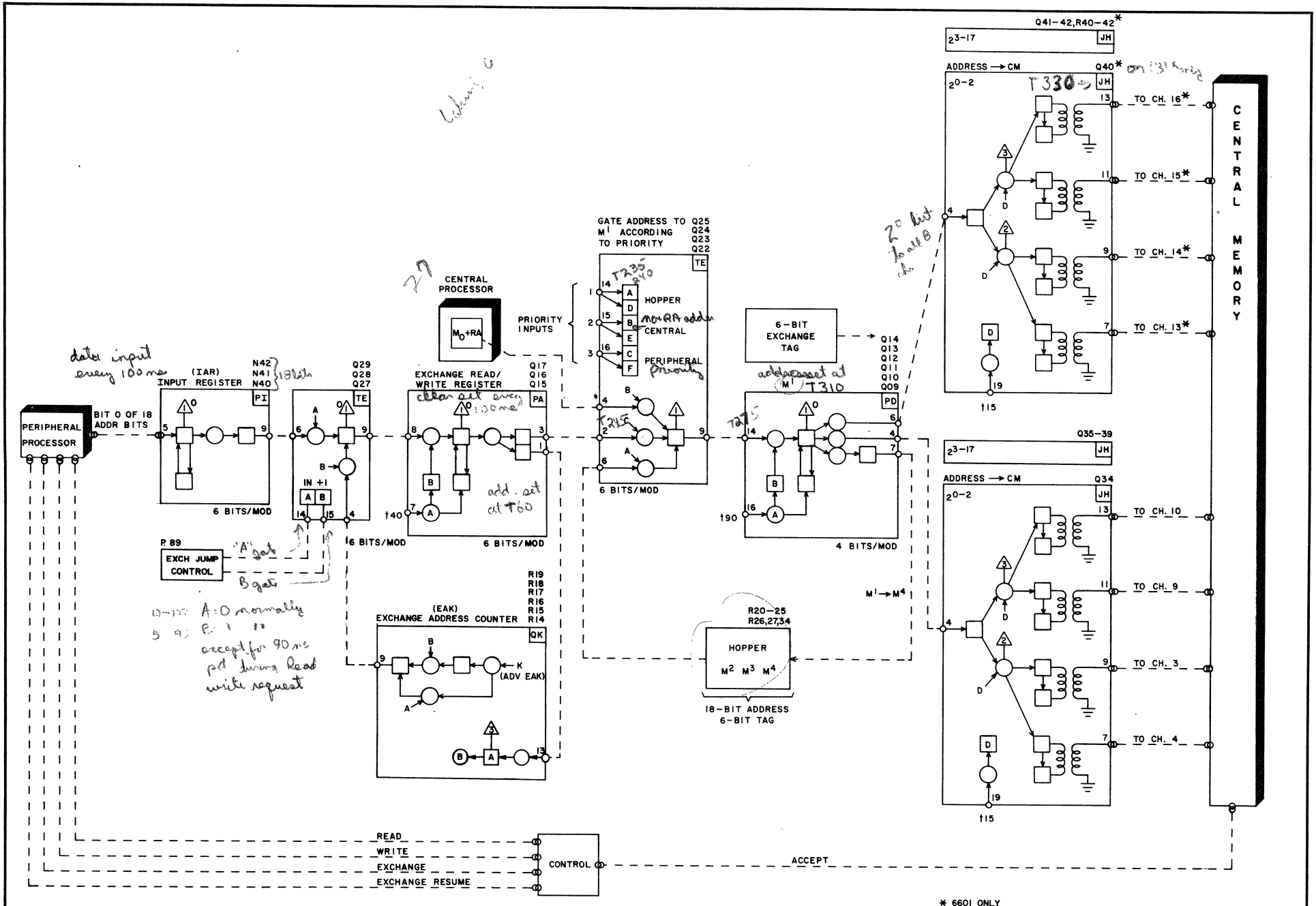
The Stunt Box is an address controlling center in the 6601 and 6604 Central Processors and is located on chassis 5. Its primary function is to gather addresses from the Peripheral and Control Processors and the Central Processor according to a priority scheme, place them into a hopper, and issue them to Central Memory until accepted by the desired bank. An address enters the hopper in the following order of priority:

1) an address coming out of the hopper which was not accepted by Central Memory (bank was busy).
2) an address from the Central Processor.
3) an address from a Peripheral or Control Processor.

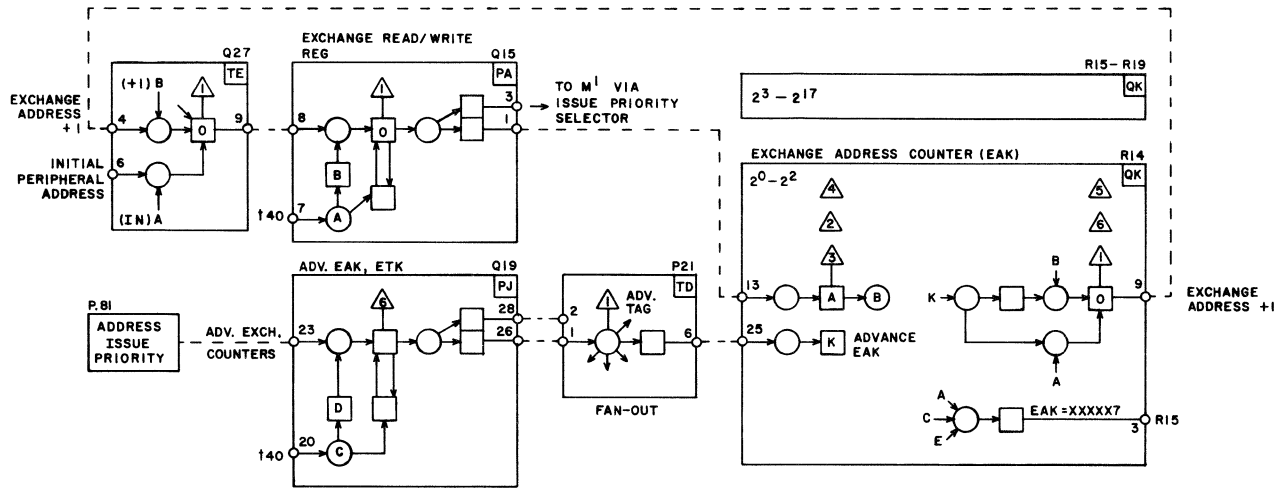
When not accepted by Central Memory, an address is sent back into the hopper where it is circulated and reissued to Central Memory every 0.3 microseconds (3 minor cycles) until the desired bank is free to accept it.



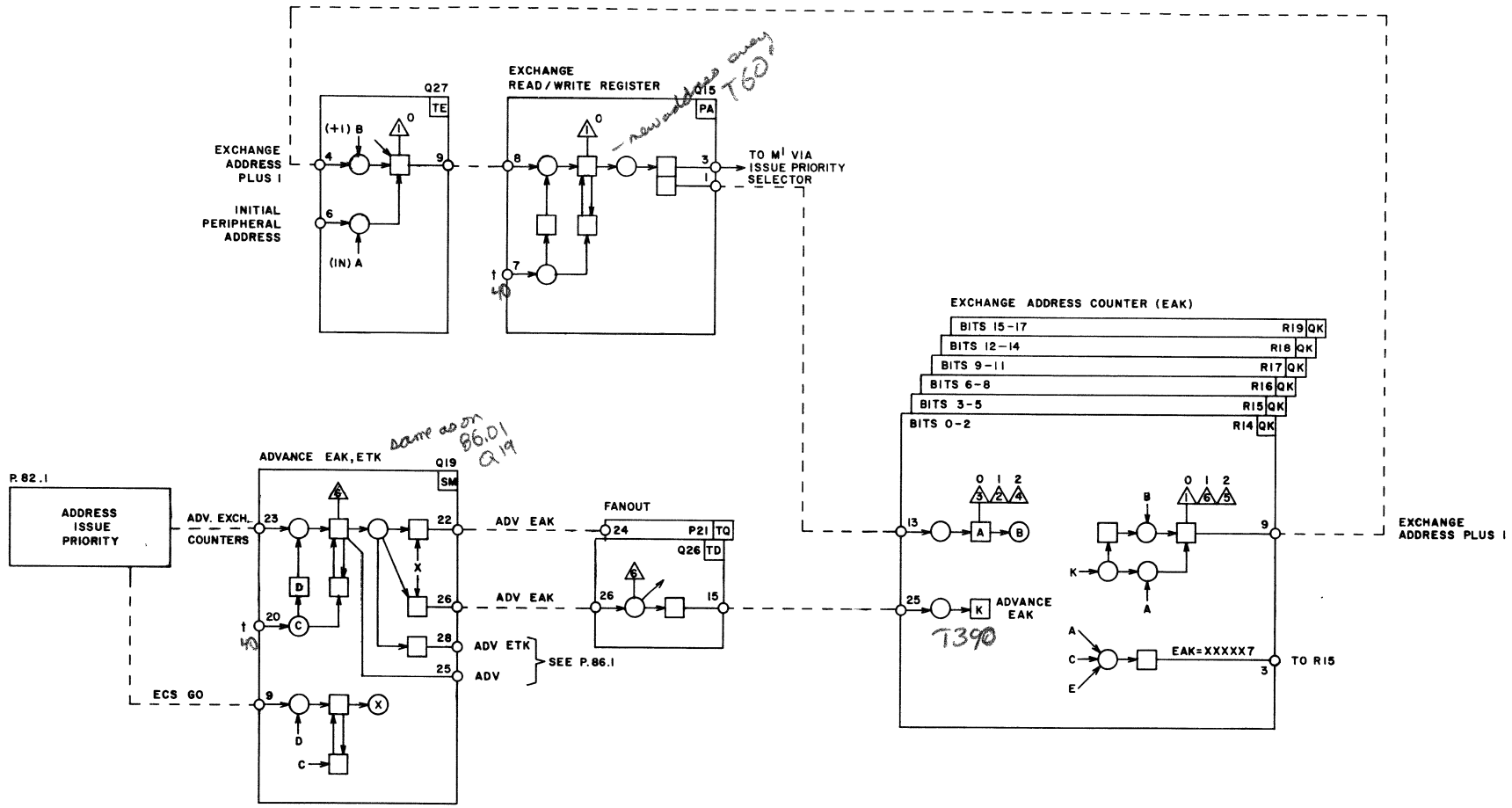
CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR STUNT BOX BLOCK DIAGRAM	PRODUCT 6601/04	
		SIZE C	DRAWING NO. 60119300 REV K
		SHEET 201	PAGE 7.5



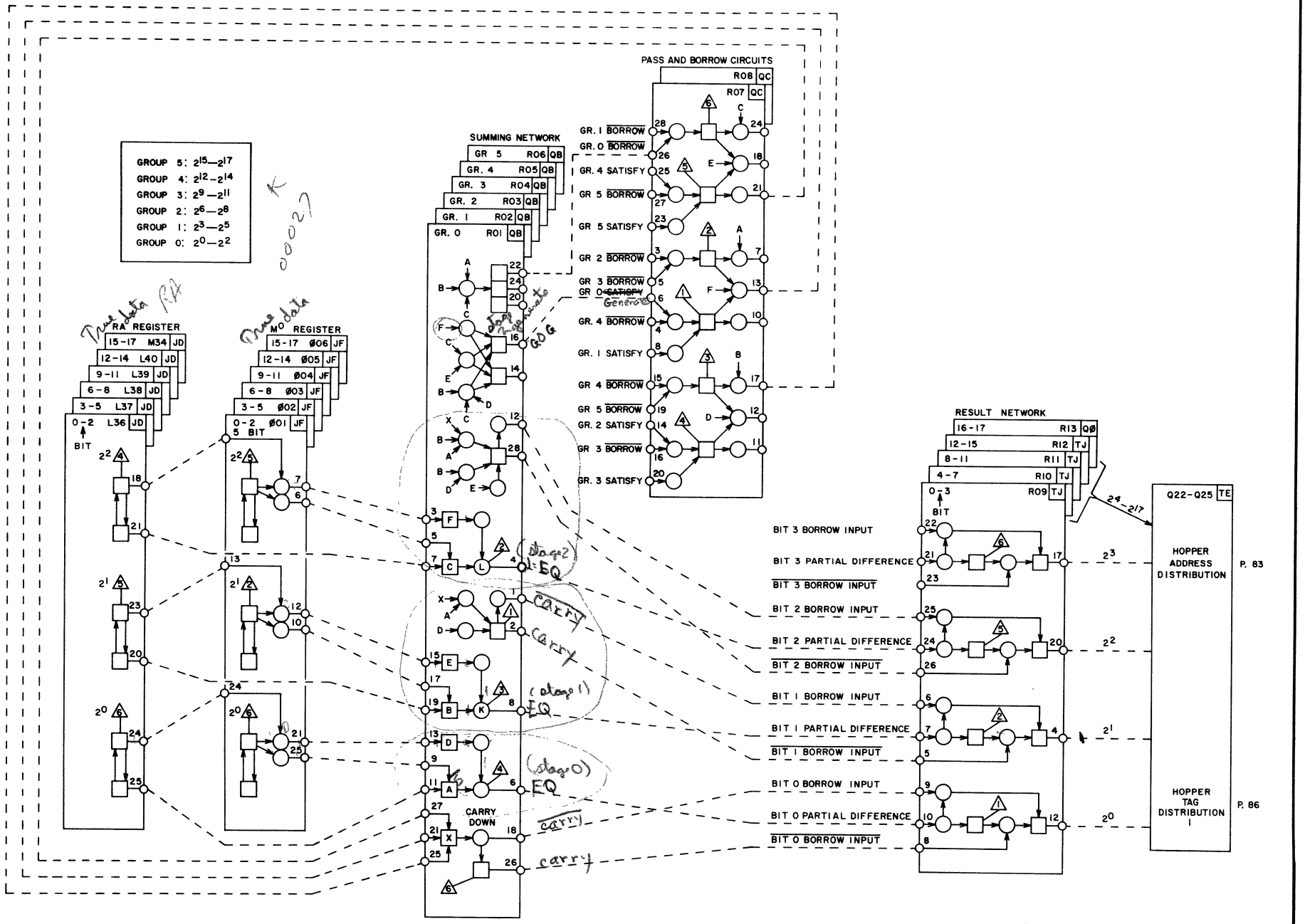
CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR STUNT BOX, EXCHANGE/READ/WRITE ADDR→CM	PRODUCT 6601/04	
		SIZE C	DRAWING NO. 60119300 REV K
			SHEET 83 77



CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR STUNT BOX EXCHANGE ADDRESS COUNTER SERIALS 1-7	PRODUCT: 6601/04	
COMPUTER DIVISION		SIZE: C · DRAWING NO.: 60119300 · REV: M	
		SHEET: 85	79

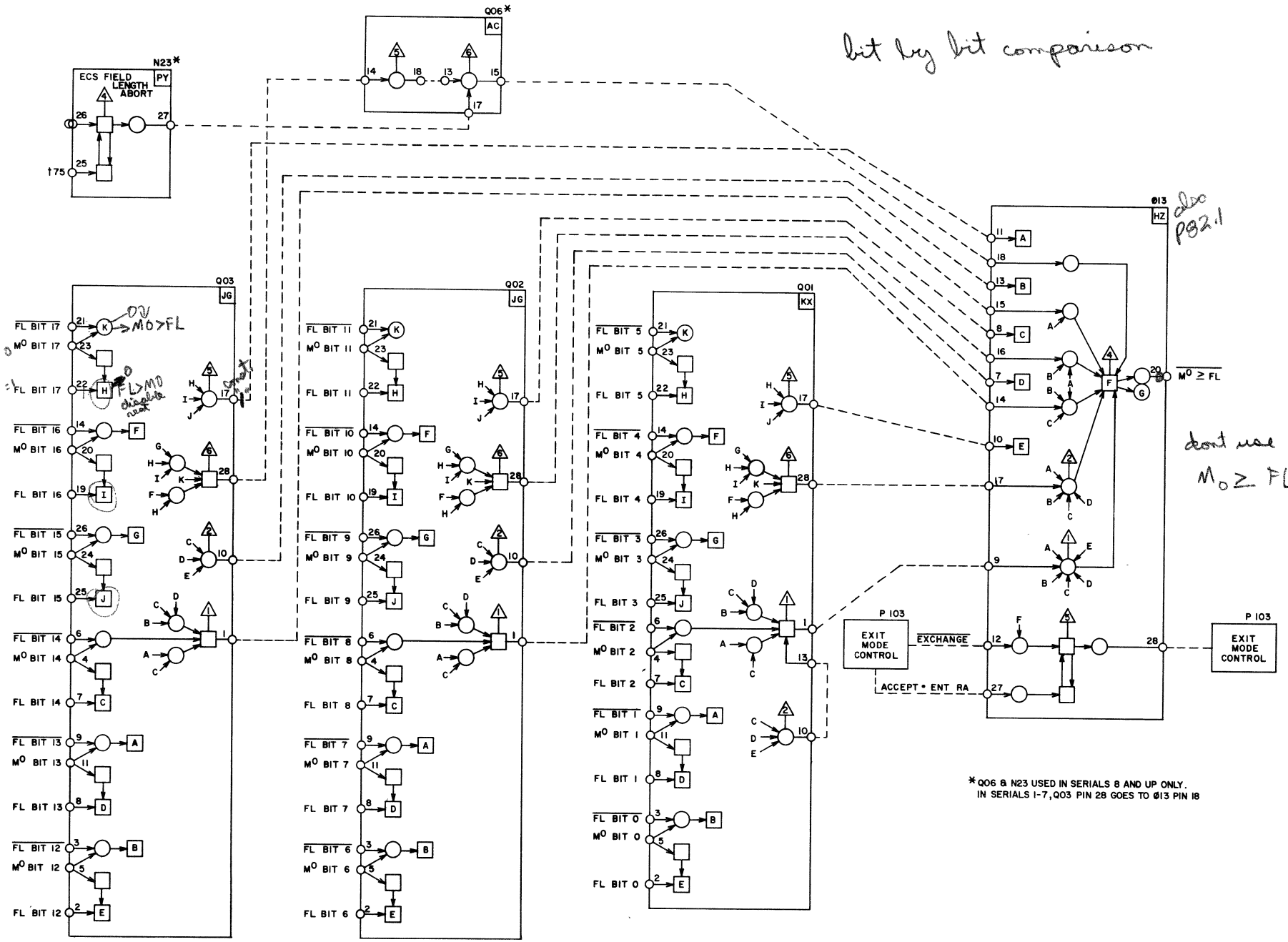


CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR STUNT BOX EXCHANGE ADDRESS COUNTER SERIALS 8-UP	PRODUCT 6601/04/13/14	
		SIZE C DRAWING NO. 60119300	REV M
		SHEET 256	80.01

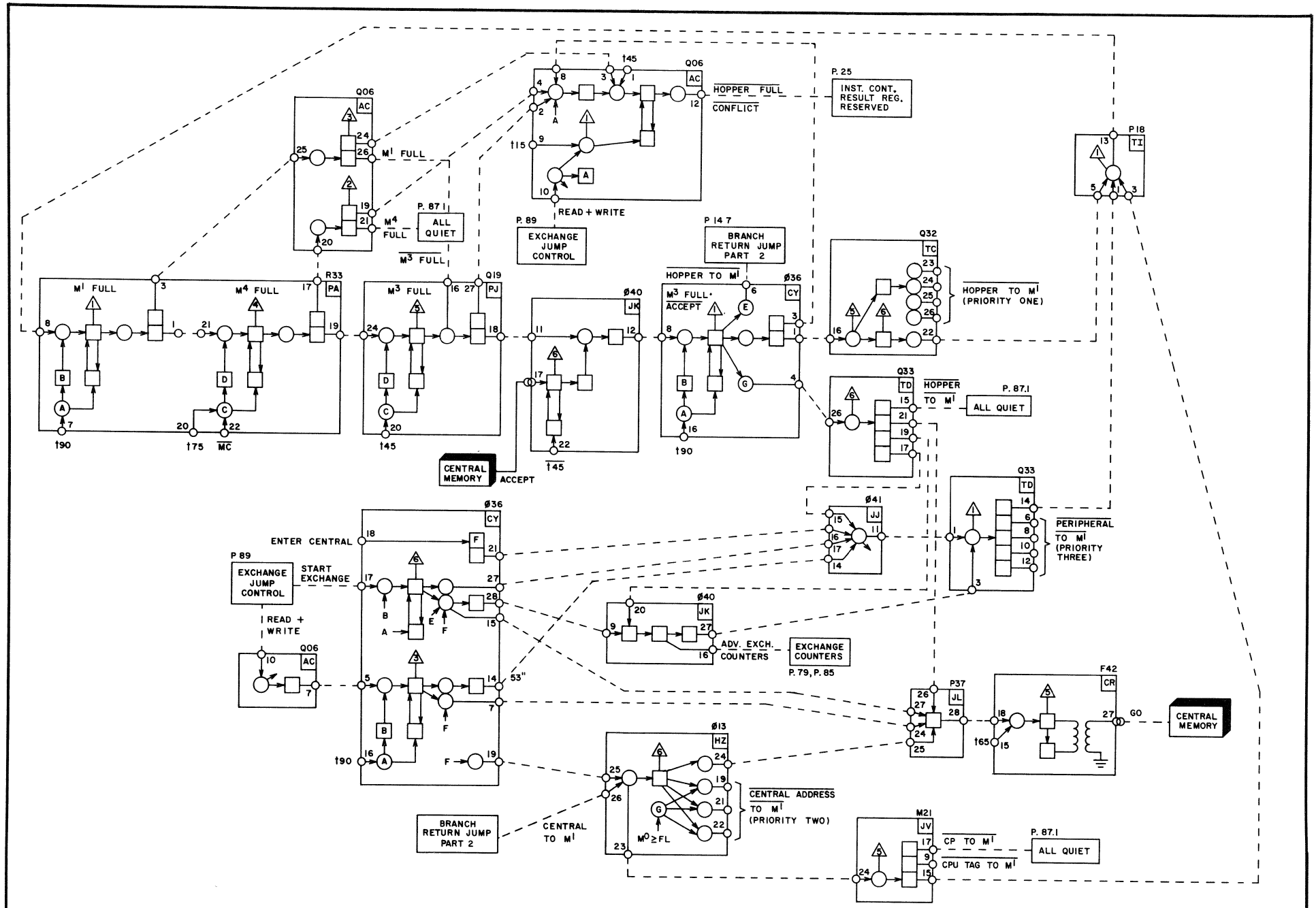


True adder

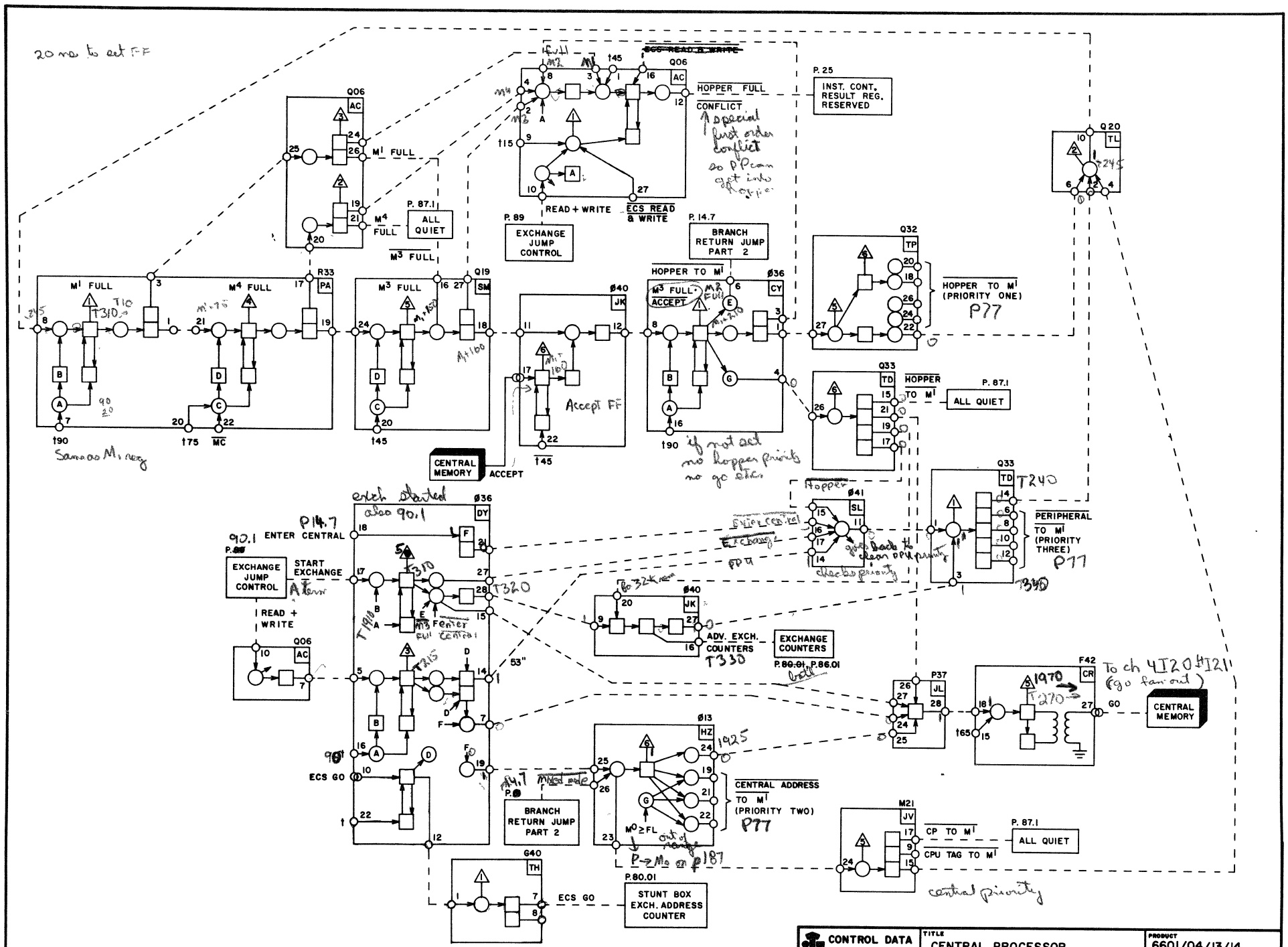
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR STUNT BOX	PRODUCT: 6601/04	
COMPUTER DIVISION	M^0 + RA ADDER	SIZE: C · DRAWING NO.: 60119300 · REV: K	
		SHEET: 205 · PAGE: 80.1	



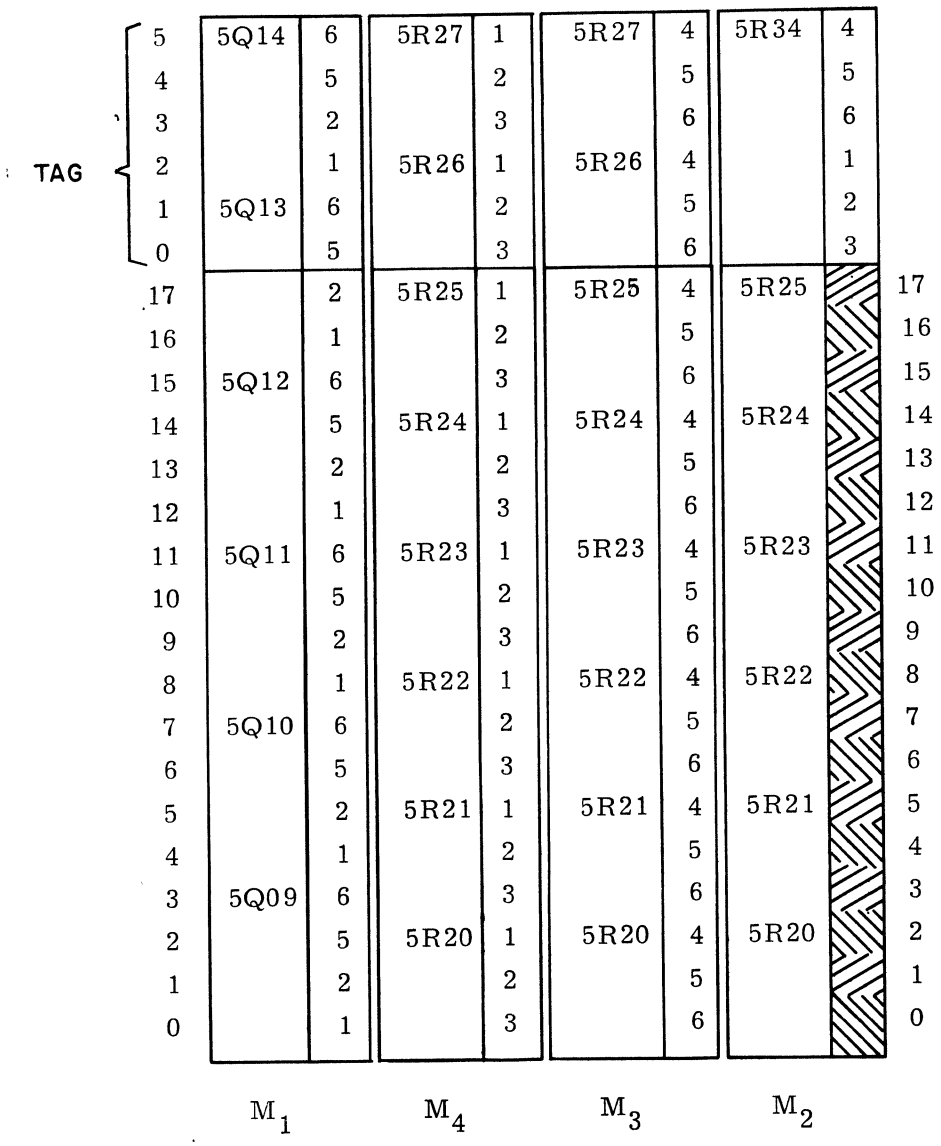
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR STUNT BOX $M^0 \geq FL$ ADDRESS RANGE TEST	PRODUCT: 6601/04/13/14	
COMPUTER DIVISION		SIZE: C · DRAWING NO.: 60119300	REV: M
		SHEET: 204 · PAGE: 80.3	

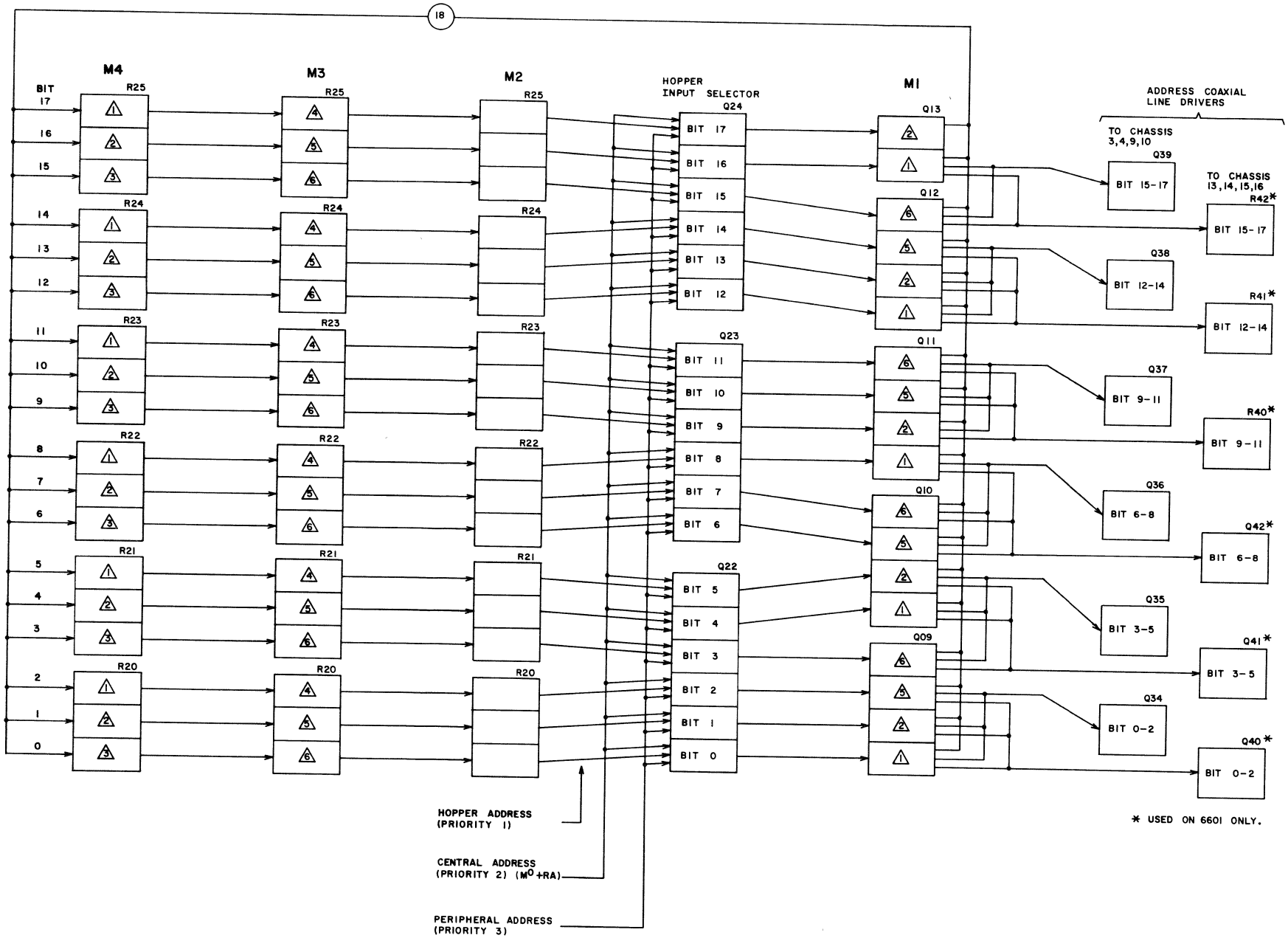


CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR STUNT BOX ADDRESS ISSUE PRIORITY SERIALS 1-7	PRODUCT 6601/04	
COMPUTER DIVISION		SIZE C · DRAWING NO. 60119300 · REV AR	
		SHEET 90 · PAGE 81	

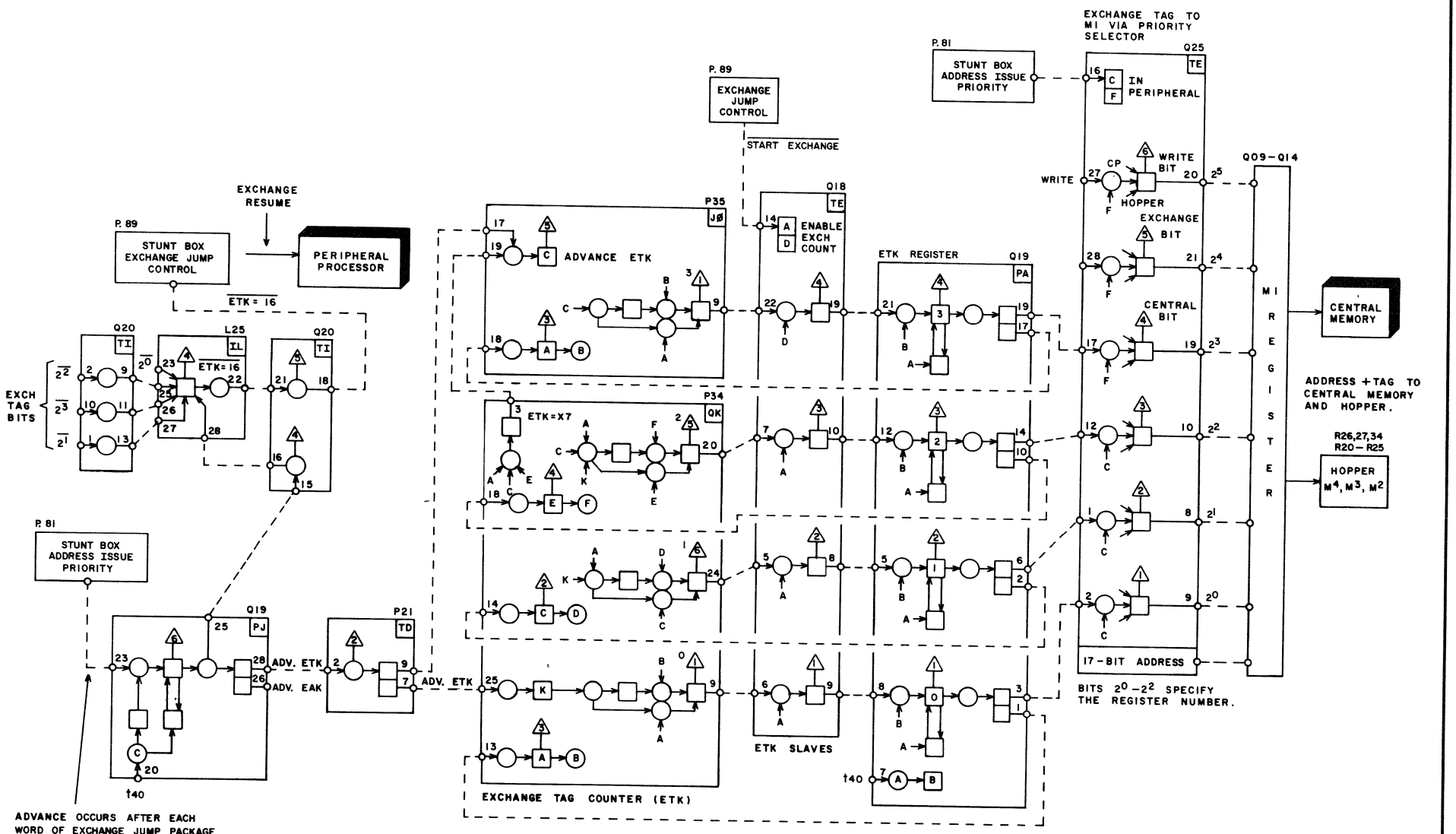


CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR STUNT BOX ADDRESS ISSUE PRIORITY SERIALS 8-UP	PRODUCT 6601/04/13/14	
		SIZE C	DRAWING NO. 60119300
		REV AR	
		SHEET 257	PAGE 82.1

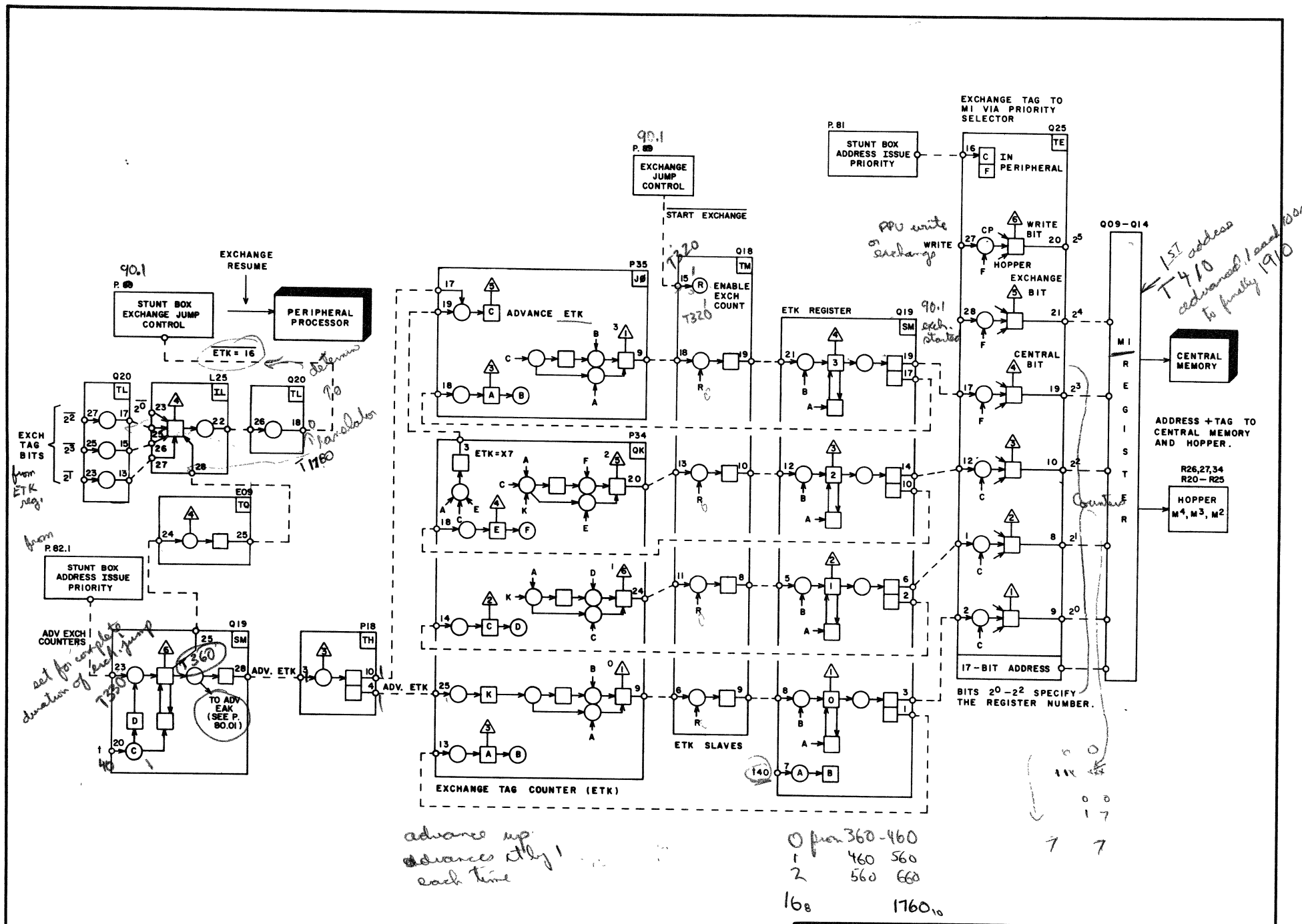

TAG
5 4 3 2 1 0
5Q14 6 5 2 1 5Q13 6 5
5R27 1 2 3 5R26 1 2 3
5R27 4 5 6 5R26 4 5 6
5R34 4 5 6 1 2 3
17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
2 1 5Q12 6 5 2 1 5Q11 6 5 2 1 5Q10 6 5 2 1 5Q09 6 5 2 1
5R25 1 2 3 5R24 1 2 3 5R23 1 2 3 5R22 1 2 3 5R21 1 2 3 5R20 1 2 3
5R25 4 5 6 5R24 4 5 6 5R23 4 5 6 5R22 4 5 6 5R21 4 5 6 5R20 4 5 6
5R25 5R24 5R23 5R22 5R21 5R20
17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
M1
M4
M3
M2



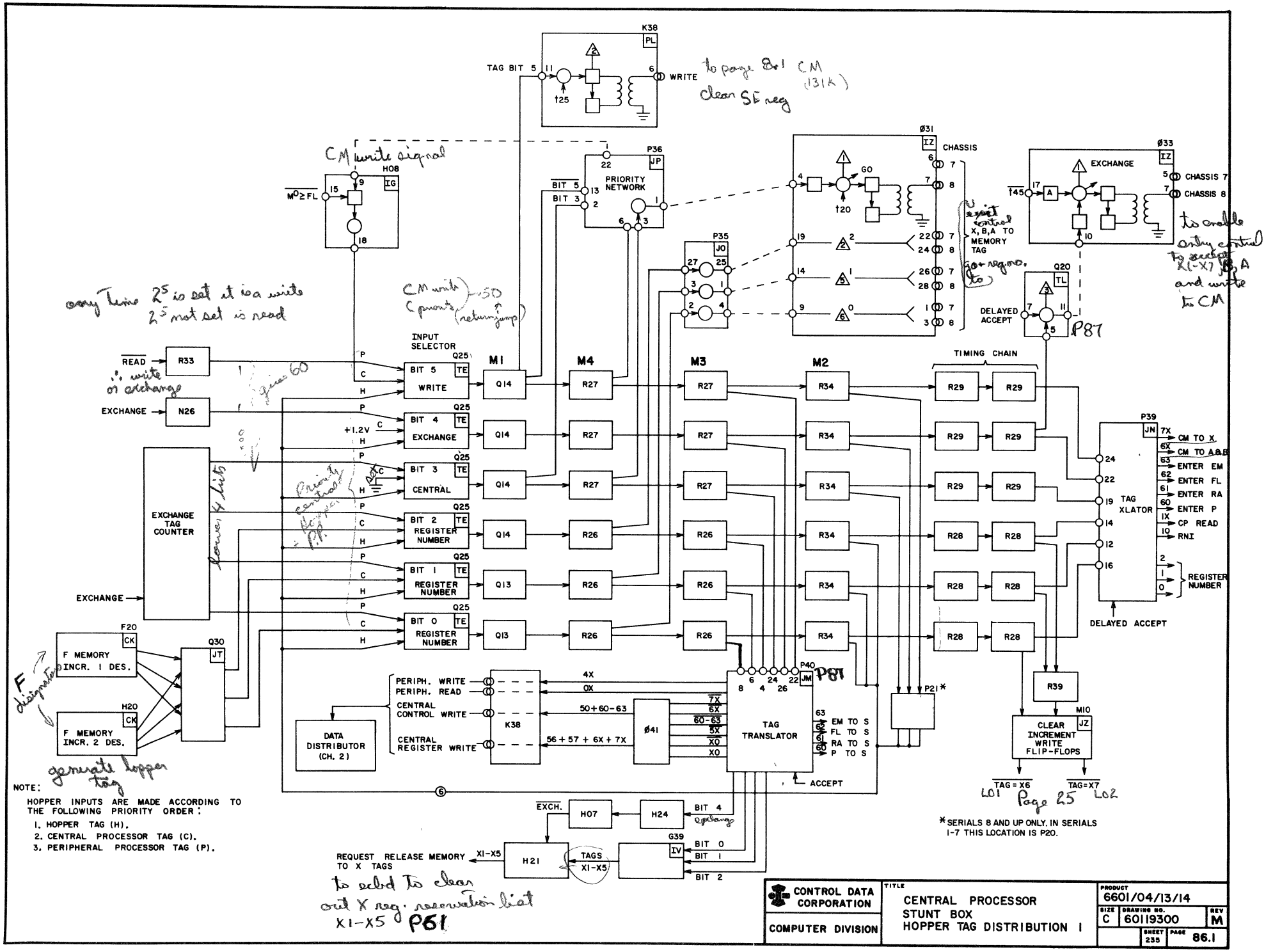
CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR STUNT BOX HOPPER ADDRESS DISTRIBUTION	PRODUCT 6601/6604	
		SIZE C DRAWING NO. 60119300	REV K
		SHEET 239 PAGE 83	



CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR STUNT BOX EXCHANGE TAG COUNTER (ETK) SERIALS 1-7	PRODUCT: 6601/04	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: M
			SHEET: 84 85



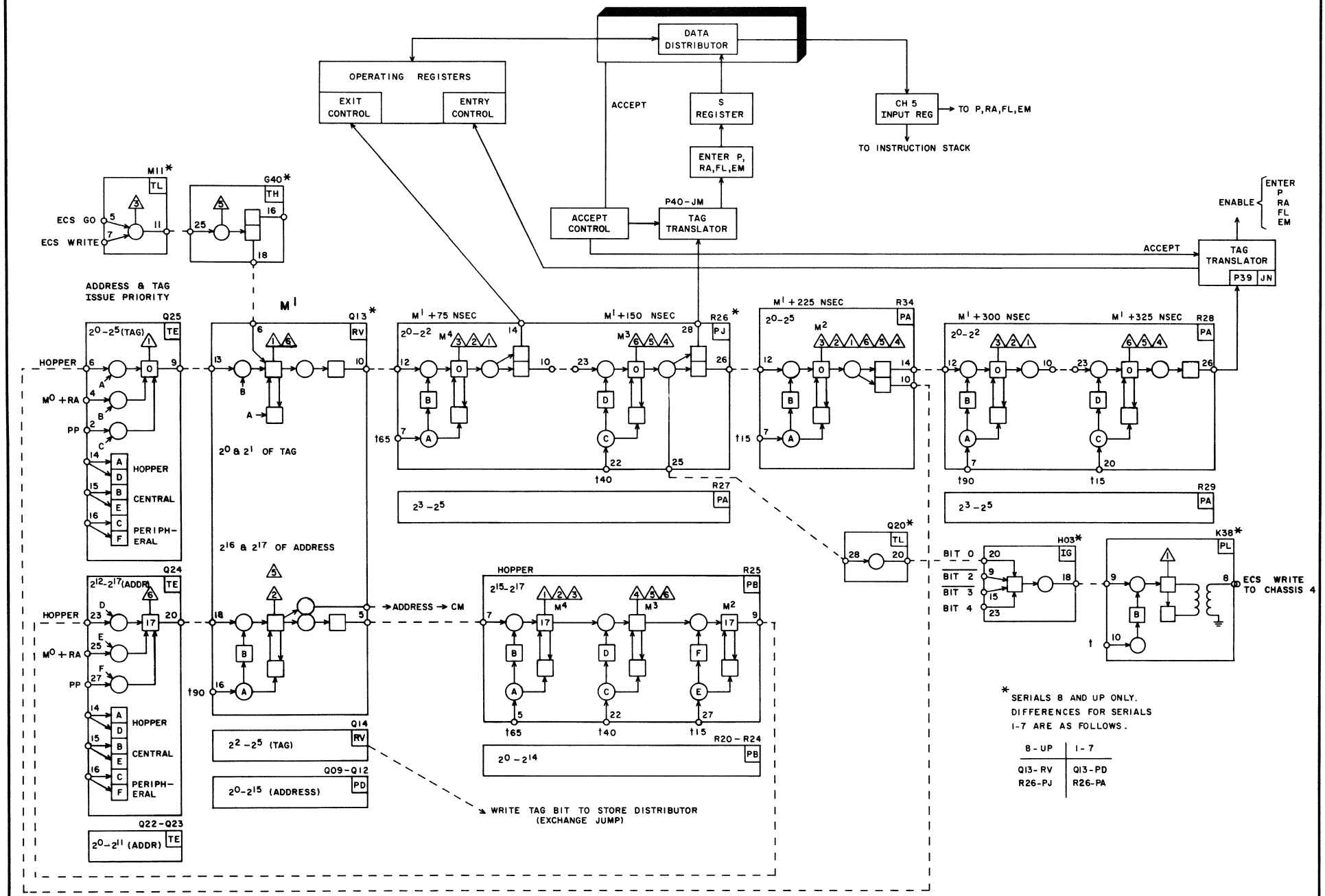
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR STUNT BOX EXCHANGE TAG COUNTER (ETK) SERIALS 8-UP	PRODUCT: 6601/04/13/14	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: AW
		SHEET: 258	86.01



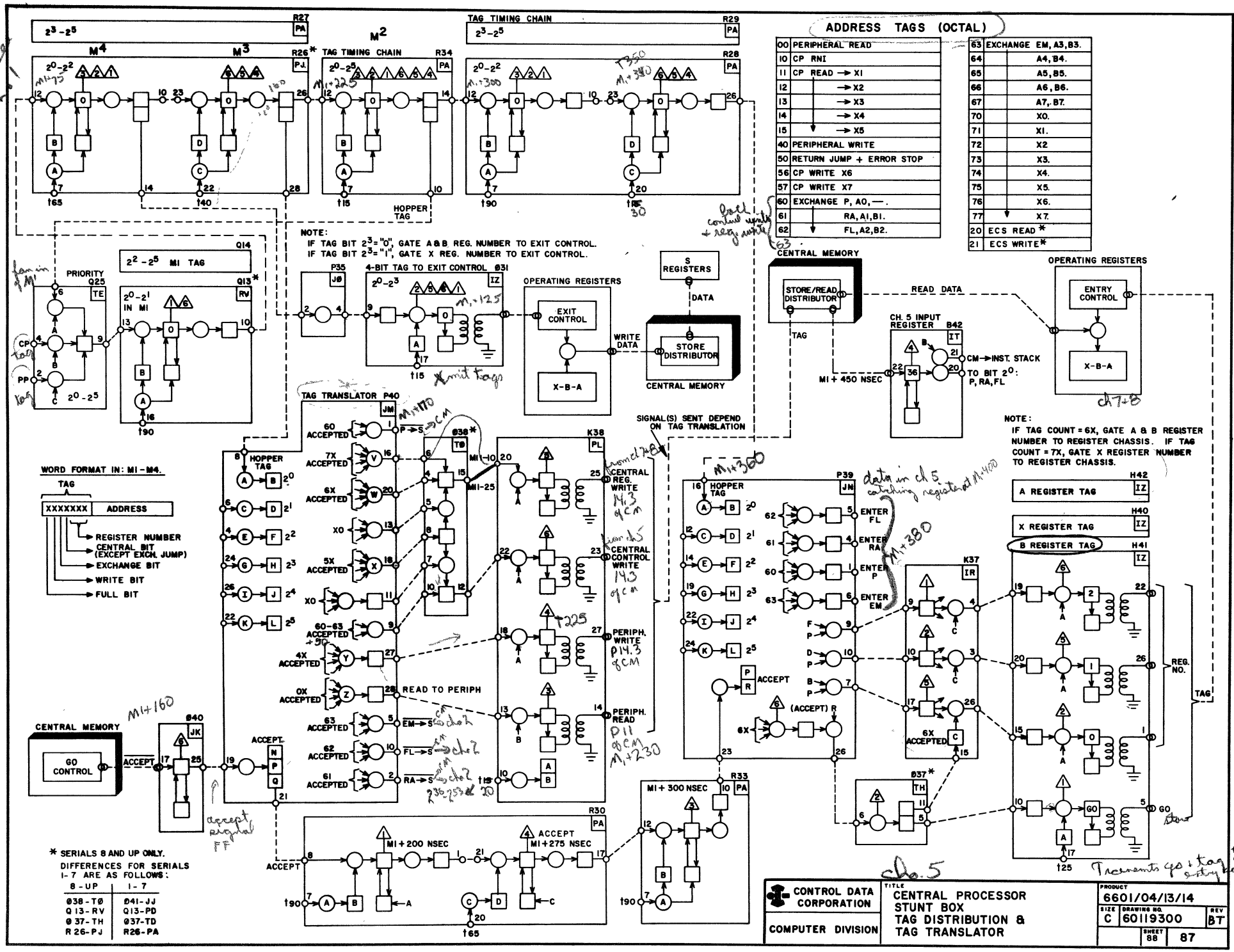
NOTE:
HOPPER INPUTS ARE MADE ACCORDING TO THE FOLLOWING PRIORITY ORDER:
1. HOPPER TAG (H).
2. CENTRAL PROCESSOR TAG (C).
3. PERIPHERAL PROCESSOR TAG (P).

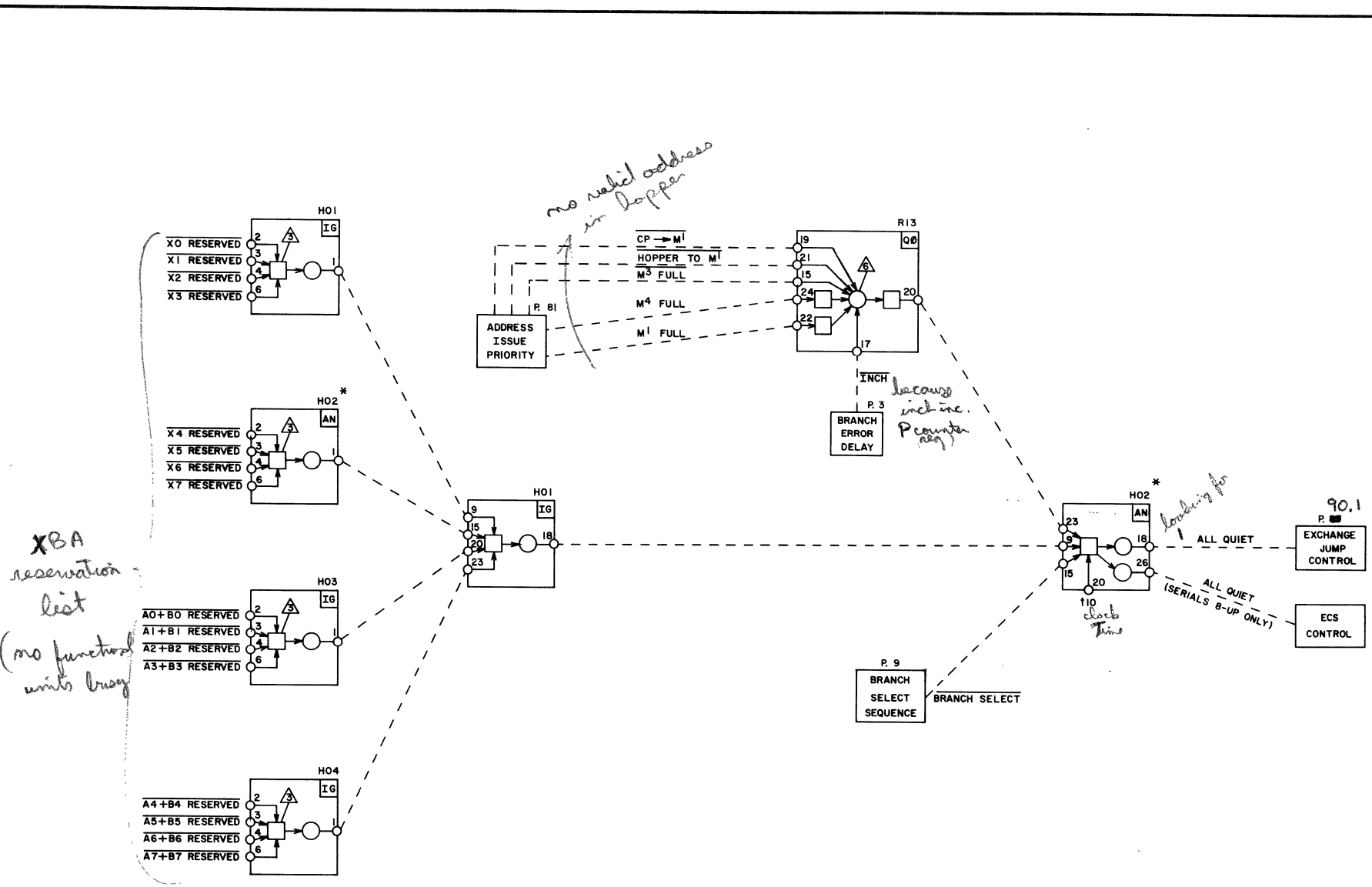
*SERIALS 8 AND UP ONLY. IN SERIALS 1-7 THIS LOCATION IS P20.

CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR STUNT BOX HOPPER TAG DISTRIBUTION 1	PRODUCT 6601/04/13/14	
		SIZE C	DRAWING NO. 60119300 REV M
		SHEET 235	PAGE 86.1



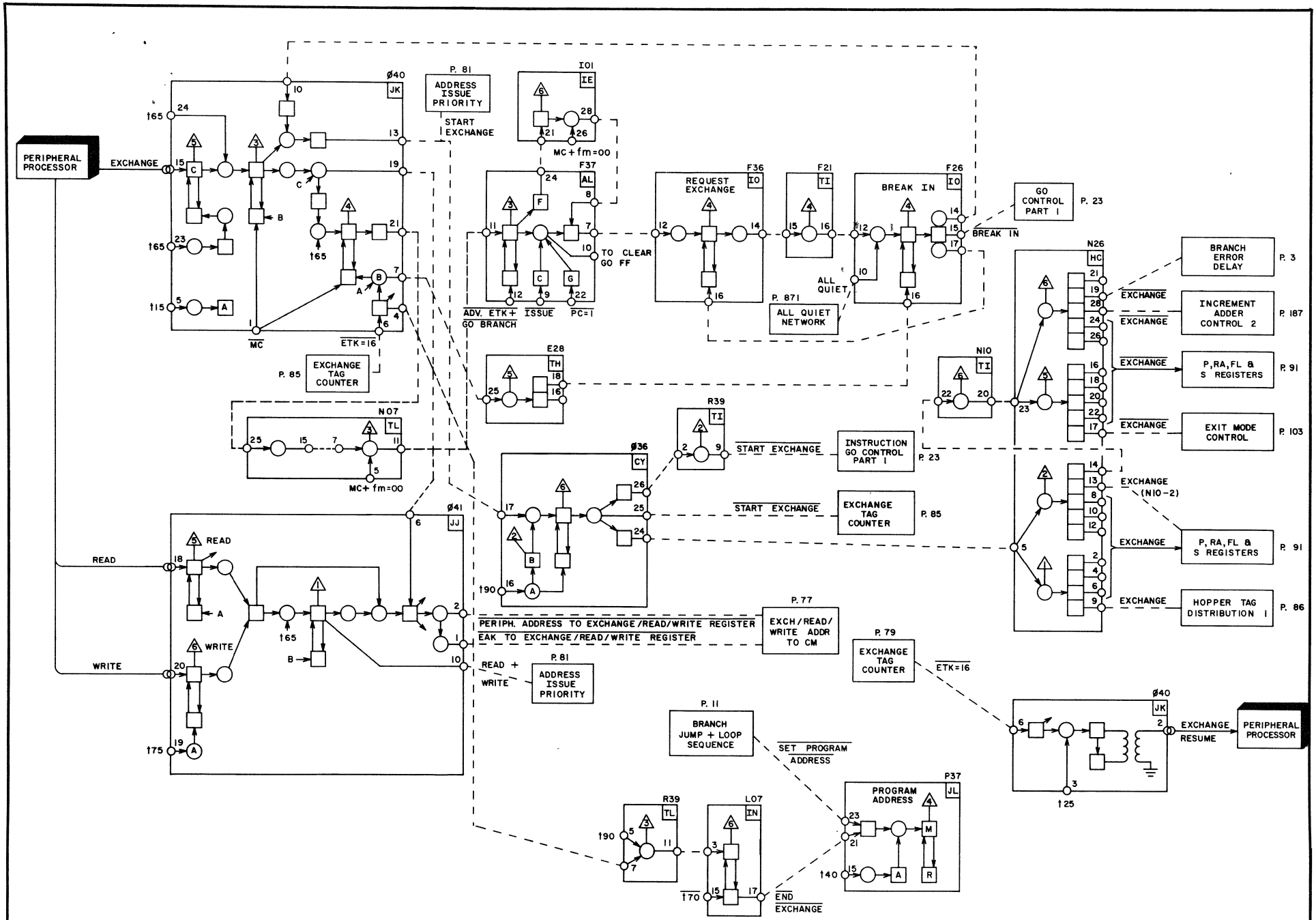
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR STUNT BOX HOPPER TAG DISTRIBUTION 2	PRODUCT: 6601/04/13/14	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: AM
		SHEET: 87	86.3


ADDRESS TAGS (OCTAL)
00 PERIPHERAL READ
10 CP RNI
11 CP READ → X1
12 → X2
13 → X3
14 → X4
15 → X5
40 PERIPHERAL WRITE
50 RETURN JUMP + ERROR STOP
56 CP WRITE X6
57 CP WRITE X7
60 EXCHANGE P, A0, —.
61 RA, A1, B1.
62 FL, A2, B2.
63 EXCHANGE EM, A3, B3.
64 A4, B4.
65 A5, B5.
66 A6, B6.
67 A7, B7.
70 X0.
71 X1.
72 X2
73 X3.
74 X4.
75 X5.
76 X6.
77 X7.
20 ECS READ *
21 ECS WRITE *
TAG TIMING CHAIN
NOTE:
IF TAG BIT 2^3="0", GATE A&B REG. NUMBER TO EXIT CONTROL.
IF TAG BIT 2^3="1", GATE X REG. NUMBER TO EXIT CONTROL.
4-BIT TAG TO EXIT CONTROL Ø31
OPERATING REGISTERS
EXIT CONTROL
X-B-A
S REGISTERS
DATA
STORE DISTRIBUTOR
CENTRAL MEMORY
WRITE DATA
STORE/READ DISTRIBUTOR
READ DATA
TAG
CH. 5 INPUT REGISTER B42
CM→INST. STACK
TO BIT 2^0: P, RA, FL
MI + 450 NSEC
ENTRY CONTROL
PRIORITY Q25
2^2-2^5 MI TAG
2^0-2^1 IN MI
TAG TRANSLATOR P40
HOPPER TAG
60 ACCEPTED
7X ACCEPTED
6X ACCEPTED
XO
5X ACCEPTED
60-63 ACCEPTED +50
4X ACCEPTED
0X ACCEPTED
63 ACCEPTED
62 ACCEPTED
61 ACCEPTED
READ TO PERIPH
CENTRAL REG. WRITE
CENTRAL CONTROL WRITE
PERIPH. WRITE
PERIPH. READ
SIGNAL(S) SENT DEPEND ON TAG TRANSLATION
WORD FORMAT IN: MI-M4.
TAG
XXXXXXX ADDRESS
REGISTER NUMBER
CENTRAL BIT (EXCEPT EXCH. JUMP)
EXCHANGE BIT
WRITE BIT
FULL BIT
CENTRAL MEMORY
GO CONTROL
ACCEPT
MI + 200 NSEC
MI + 275 NSEC
MI + 300 NSEC
ENTER FL
ENTER RA
ENTER P
ENTER EM
(ACCEPT) R
A REGISTER TAG
X REGISTER TAG
B REGISTER TAG
REG. NO.
TAG
GO
* SERIALS 8 AND UP ONLY.
DIFFERENCES FOR SERIALS 1-7 ARE AS FOLLOWS:
8-UP | 1-7
Ø38-TØ | Ø41-JJ
Q13-RV | Q13-PD
Ø37-TH | Ø37-TD
R26-PJ | R26-PA
CONTROL DATA CORPORATION
COMPUTER DIVISION
TITLE
CENTRAL PROCESSOR STUNT BOX TAG DISTRIBUTION & TAG TRANSLATOR
PRODUCT 6601/04/13/14
SIZE C
DRAWING NO. 60119300
REV BT
SHEET 88
87

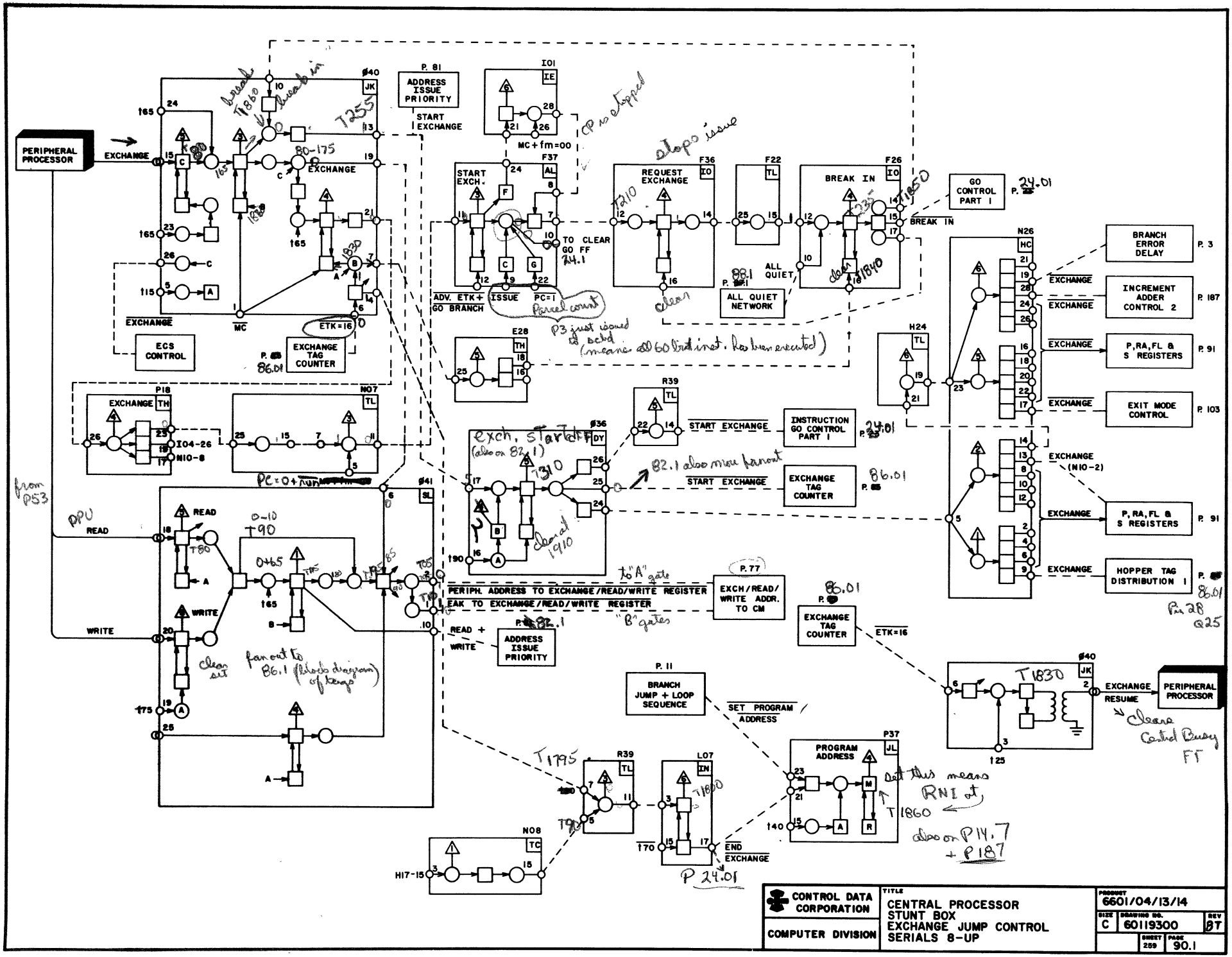


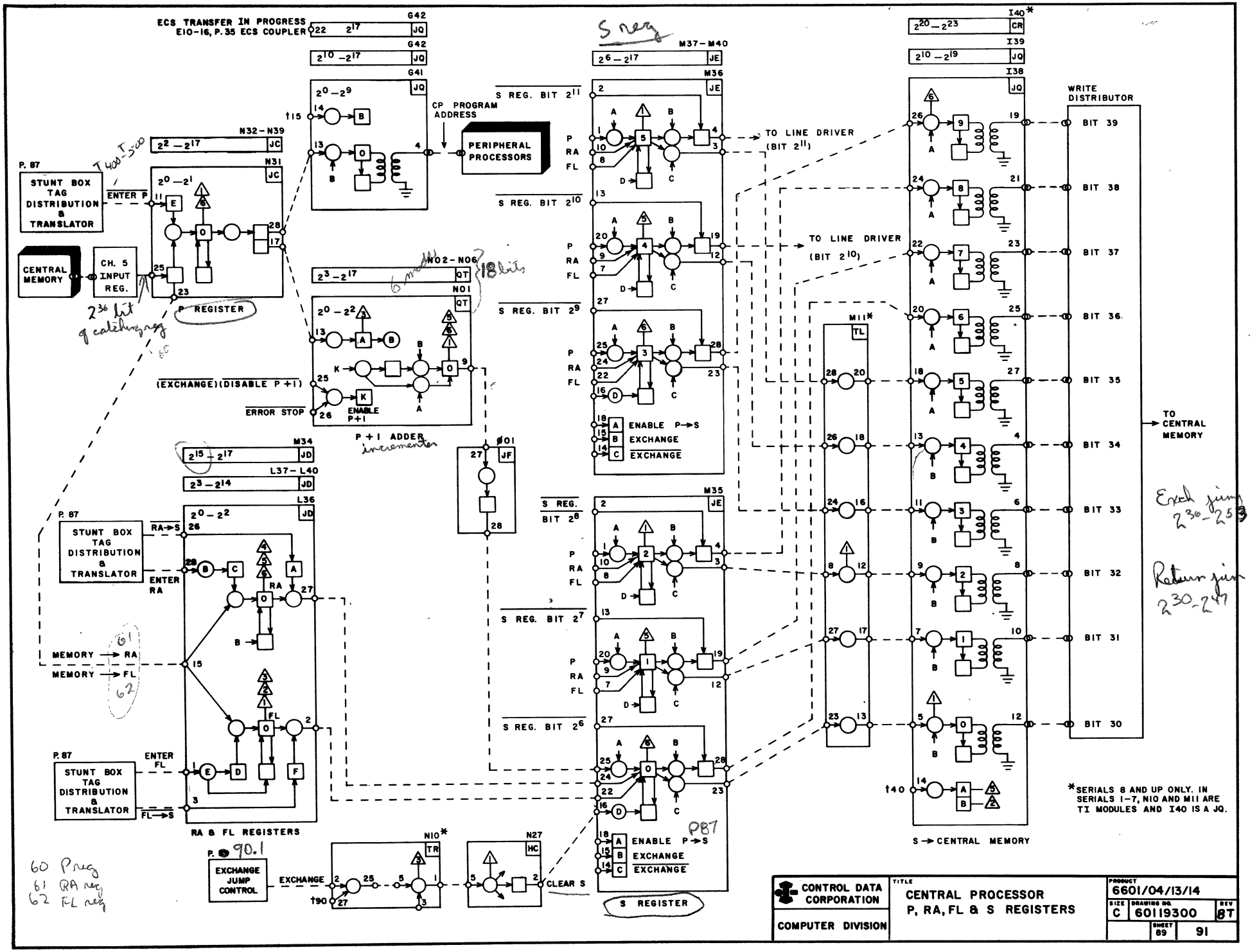
NOTE
* H02 IS AN IG MODULE IN SERIALS 1-7

CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR STUNT BOX ALL QUIET NETWORK	PRODUCT 6601/04 SIZE C · DRAWING NO. 60119300 · REV M SHEET 237 · PAGE 88.1	

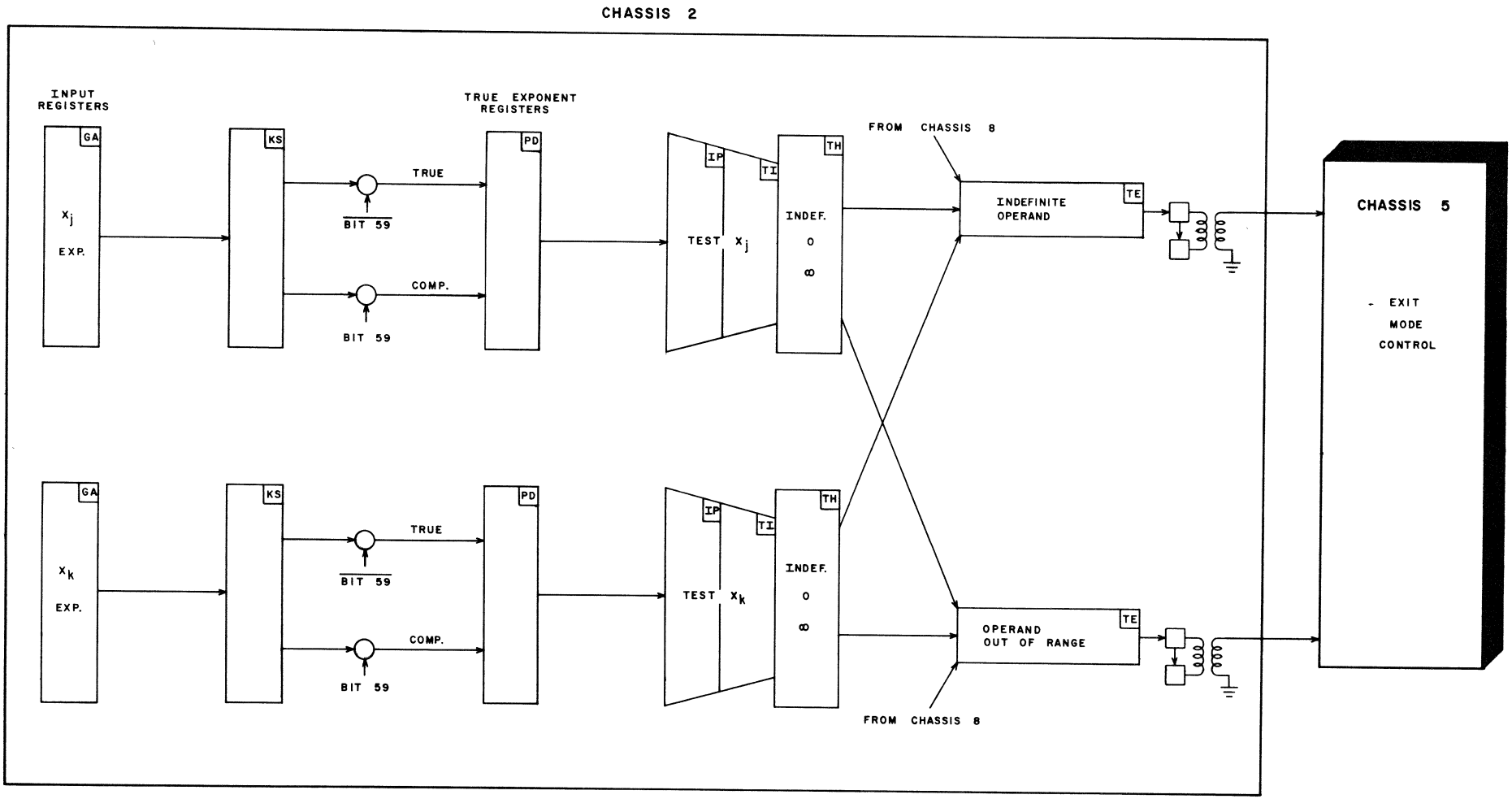


CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR STUNT BOX EXCHANGE JUMP CONTROL SERIALS 1–7	PRODUCT 6601/04	
		SIZE C	DRAWING NO. 60119300 REV AY
		SHEET 238	PAGE 89


PERIPHERAL PROCESSOR
EXCHANGE
ECS CONTROL
EXCHANGE TAG COUNTER
ADDRESS ISSUE PRIORITY
START EXCHANGE
MC + fm = 00
START EXCH.
ADV. ETK + GO BRANCH
ISSUE
PC=1
TO CLEAR GO FF
REQUEST EXCHANGE
ALL QUIET NETWORK
BREAK IN
GO CONTROL PART 1
BRANCH ERROR DELAY
INCREMENT ADDER CONTROL 2
P, RA, FL & S REGISTERS
EXIT MODE CONTROL
HOPPER TAG DISTRIBUTION
INSTRUCTION GO CONTROL PART 1
EXCHANGE TAG COUNTER
PERIPH. ADDRESS TO EXCHANGE/READ/WRITE REGISTER
LEAK TO EXCHANGE/READ/WRITE REGISTER
EXCH/READ/WRITE ADDR. TO CM
BRANCH JUMP + LOOP SEQUENCE
SET PROGRAM ADDRESS
PROGRAM ADDRESS
END EXCHANGE
EXCHANGE RESUME
PERIPHERAL PROCESSOR
READ
WRITE
ETK=16
MC
CONTROL DATA CORPORATION
COMPUTER DIVISION
CENTRAL PROCESSOR
STUNT BOX
EXCHANGE JUMP CONTROL
SERIALS 8-UP
6601/04/13/14
C
60119300
BT
90.1

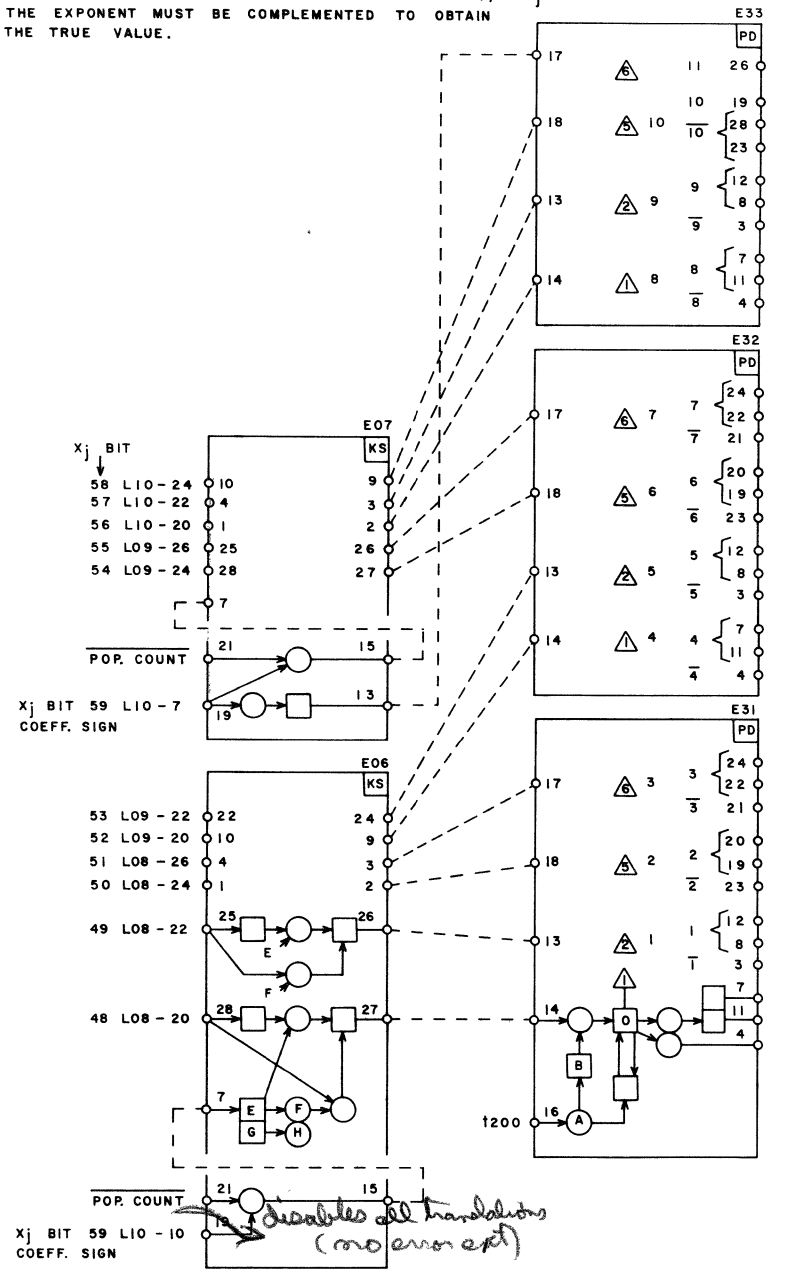


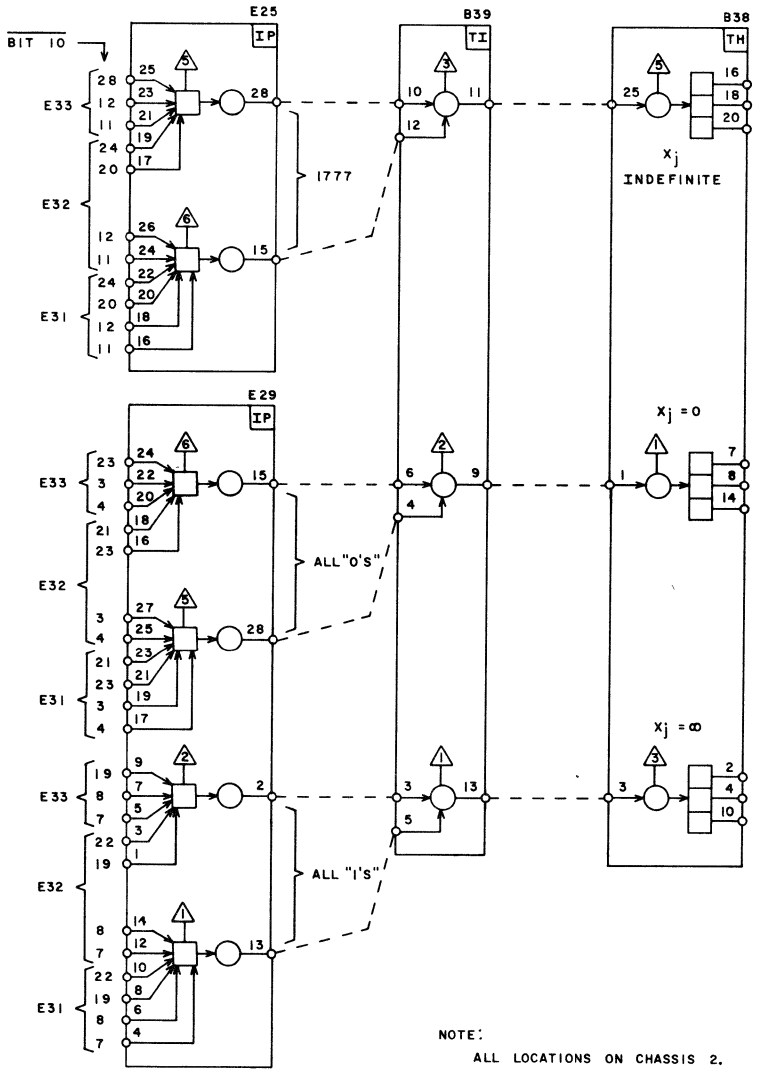
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR P, RA, FL & S REGISTERS	PRODUCT: 6601/04/13/14	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: 8T
			SHEET: 89 91



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR EXIT MODE TESTS MULTIPLY, DIVIDE, BOOLEAN BLOCK DIAGRAM	PRODUCT 6601 SIZE C DRAWING NO. 60119300 REV D SHEET 92 93	

IF COEFFICIENT IS NEGATIVE (SIGN BIT 59 = "1"), X_j TRUE EXPONENT REGISTER THE EXPONENT MUST BE COMPLEMENTED TO OBTAIN THE TRUE VALUE.

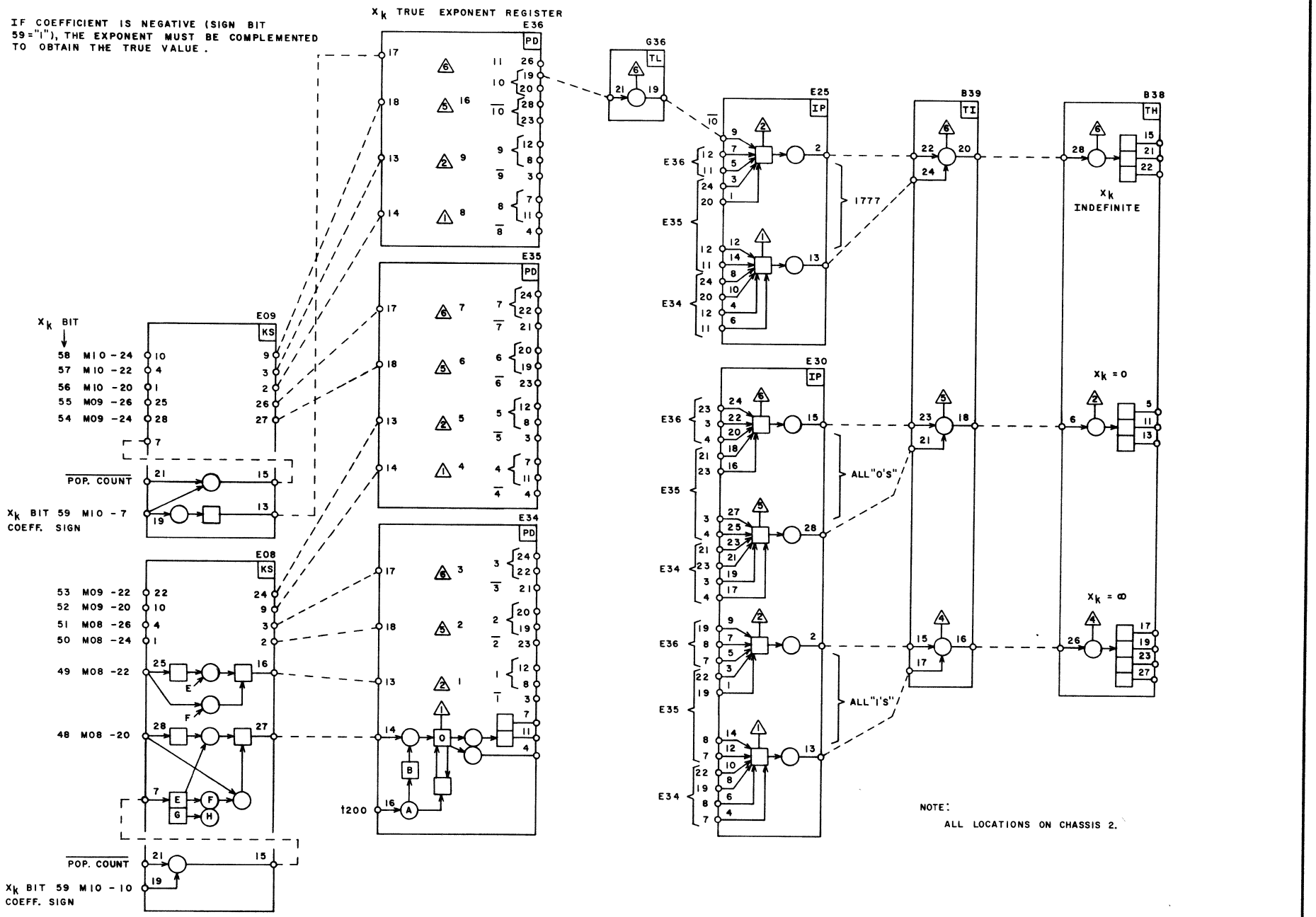




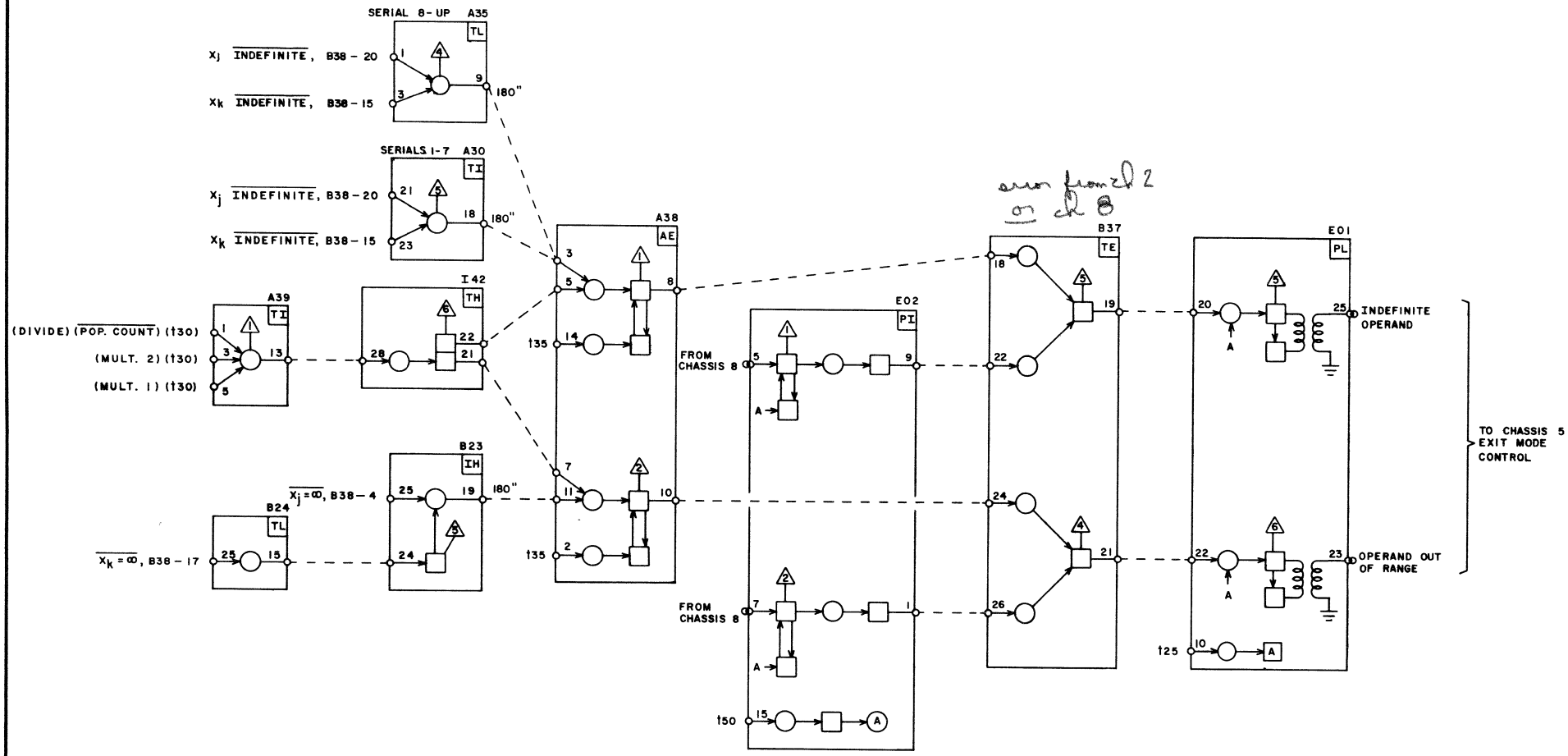
NOTE:
ALL LOCATIONS ON CHASSIS 2.

only tests exp.

CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR X_j EXIT MODE TEST MULTIPLY, DIVIDE, BOOLEAN	PRODUCT 6601 SIZE C · DRAWING NO. 60119300 · REV D SHEET 93	95

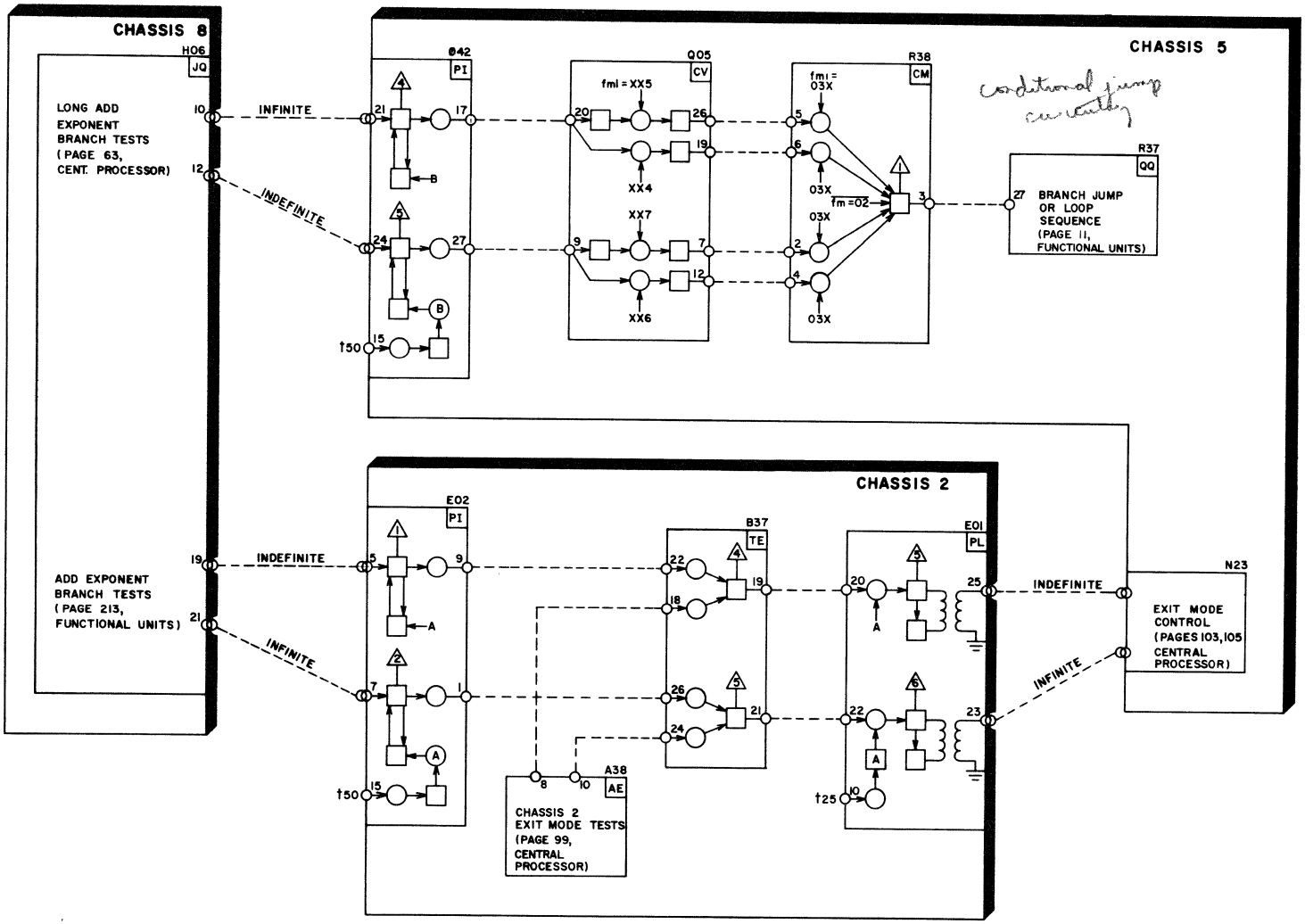


CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR X_k EXIT MODE TEST	PRODUCT: 6601	
COMPUTER DIVISION	MULTIPLY, DIVIDE, BOOLEAN	SIZE: C · DRAWING NO.: 60119300 · REV: D	
		SHEET: 94	97

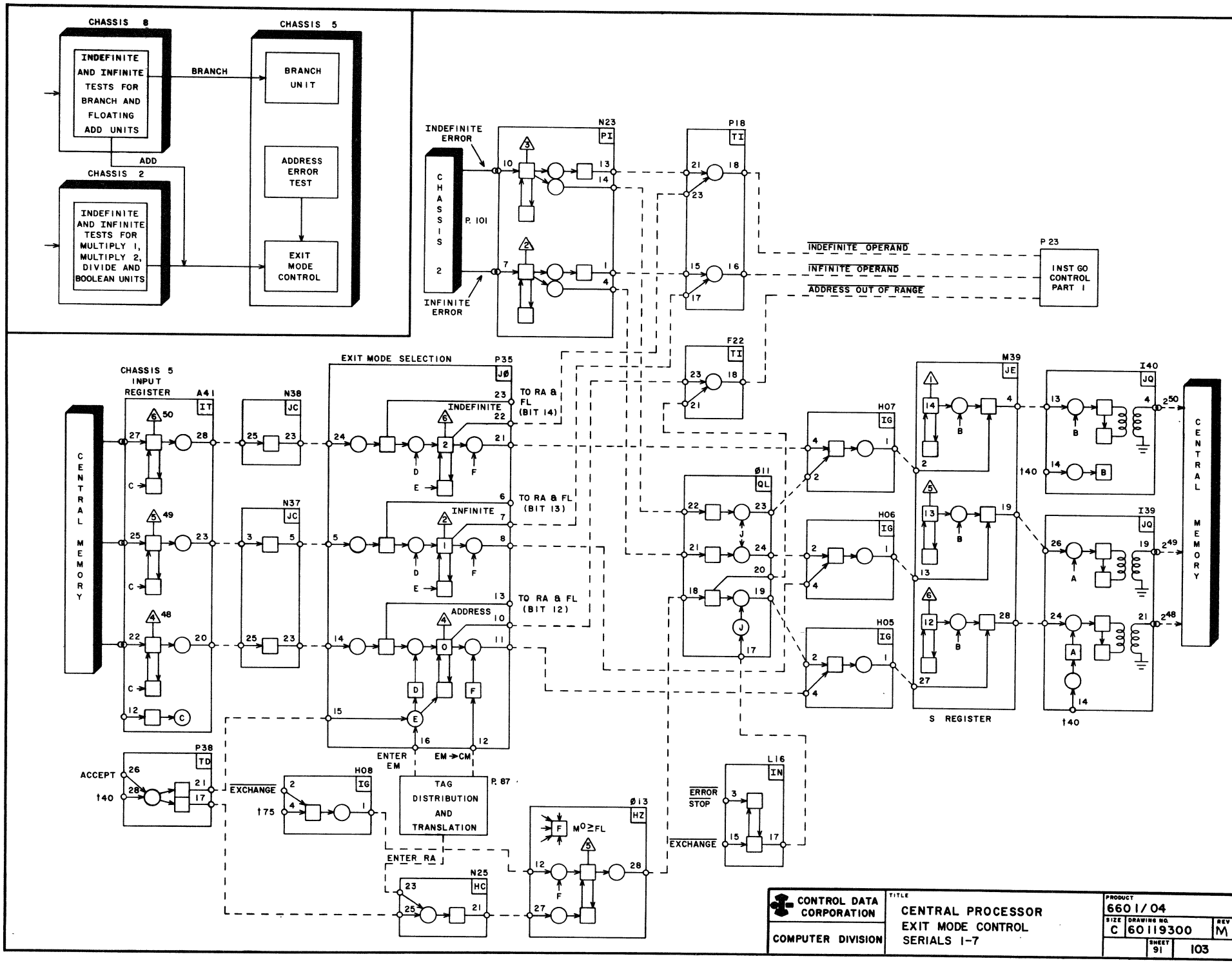


NOTE:
ALL LOCATIONS ON CHASSIS 2.

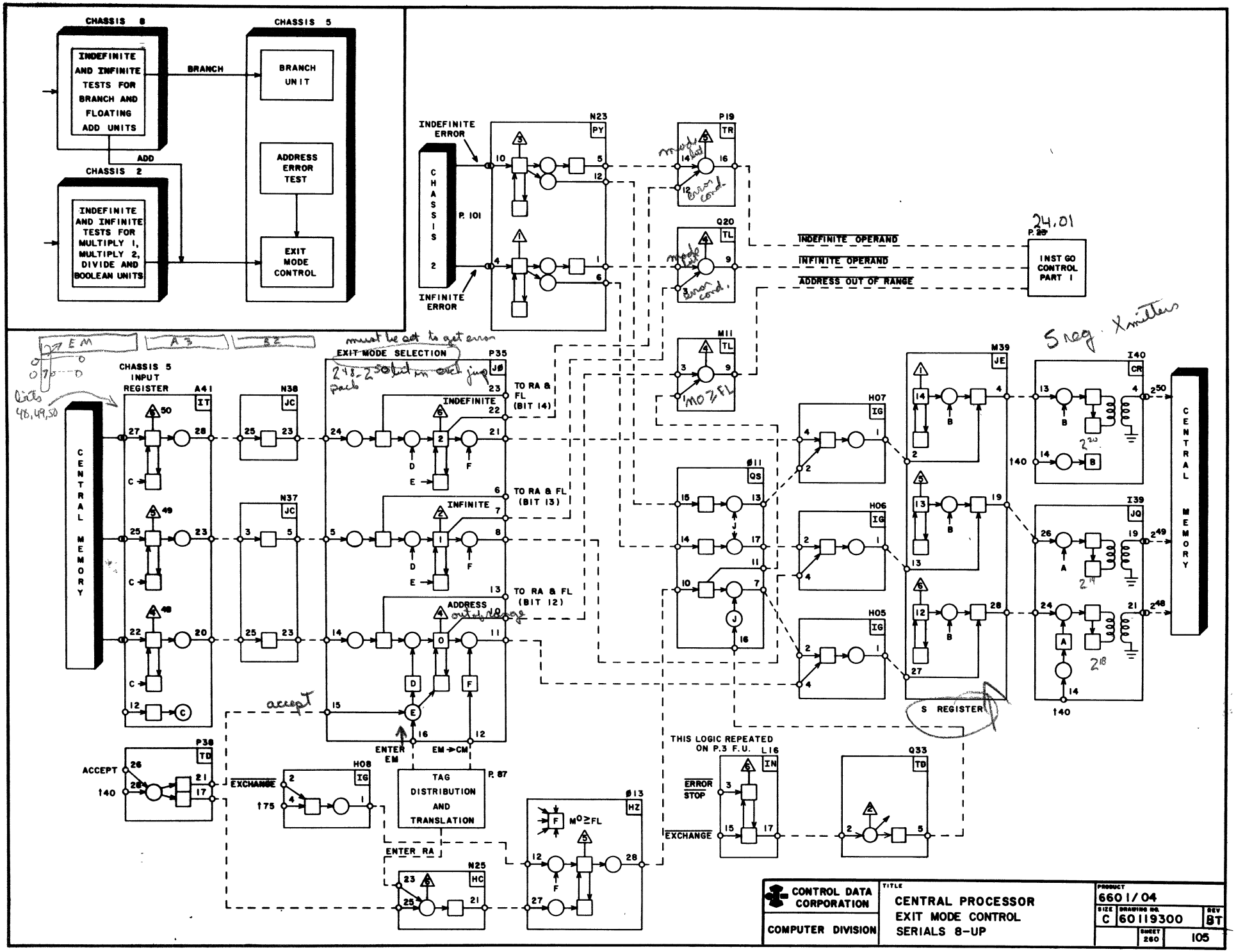
CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR EXIT MODE TESTS TRANSMIT TO CHASSIS 5	PRODUCT 6601/04 SIZE C · DRAWING NO. 60119300 · REV AW SHEET 95	99



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR EXIT MODE FAN-IN, BRANCH RANGE & INDEFINITE TESTS	PRODUCT 6601/04/13/14	
		SIZE C	DRAWING NO. 60119300 REV M
			SHEET 241 PAGE 101



CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR EXIT MODE CONTROL SERIALS 1-7	PRODUCT: 6601/04	
COMPUTER DIVISION		SIZE: C	DRAWING NO: 60119300 REV: M
			SHEET: 91 103



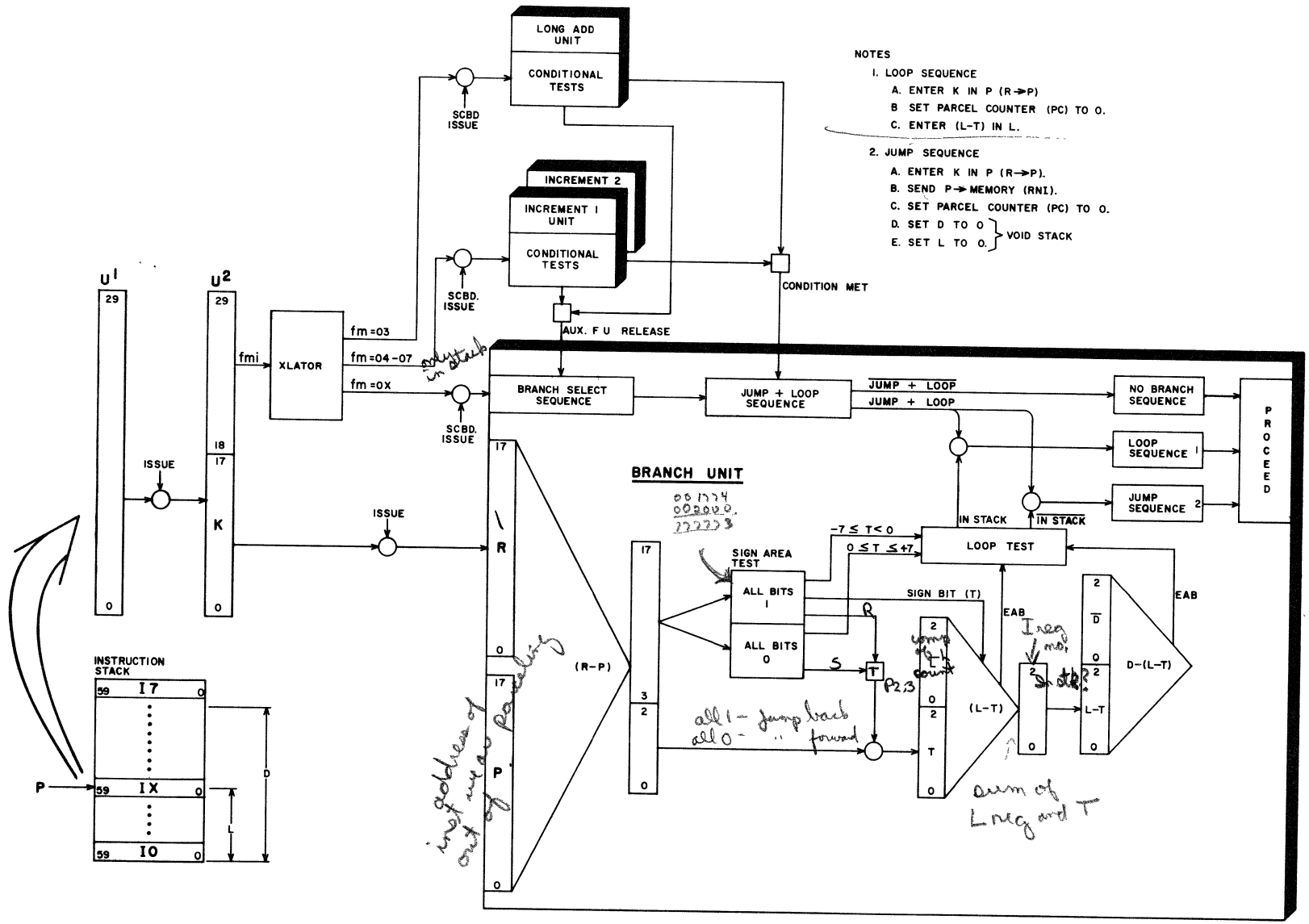
CONTROL DATA CORPORATION	TITLE	PRODUCT	6601/04
COMPUTER DIVISION	CENTRAL PROCESSOR EXIT MODE CONTROL SERIALS 8-UP	SIZE C	DRAWING NO. 60119300 REV BT
		SHEET 260	105

FUNCTIONAL UNITS
CONTENTS

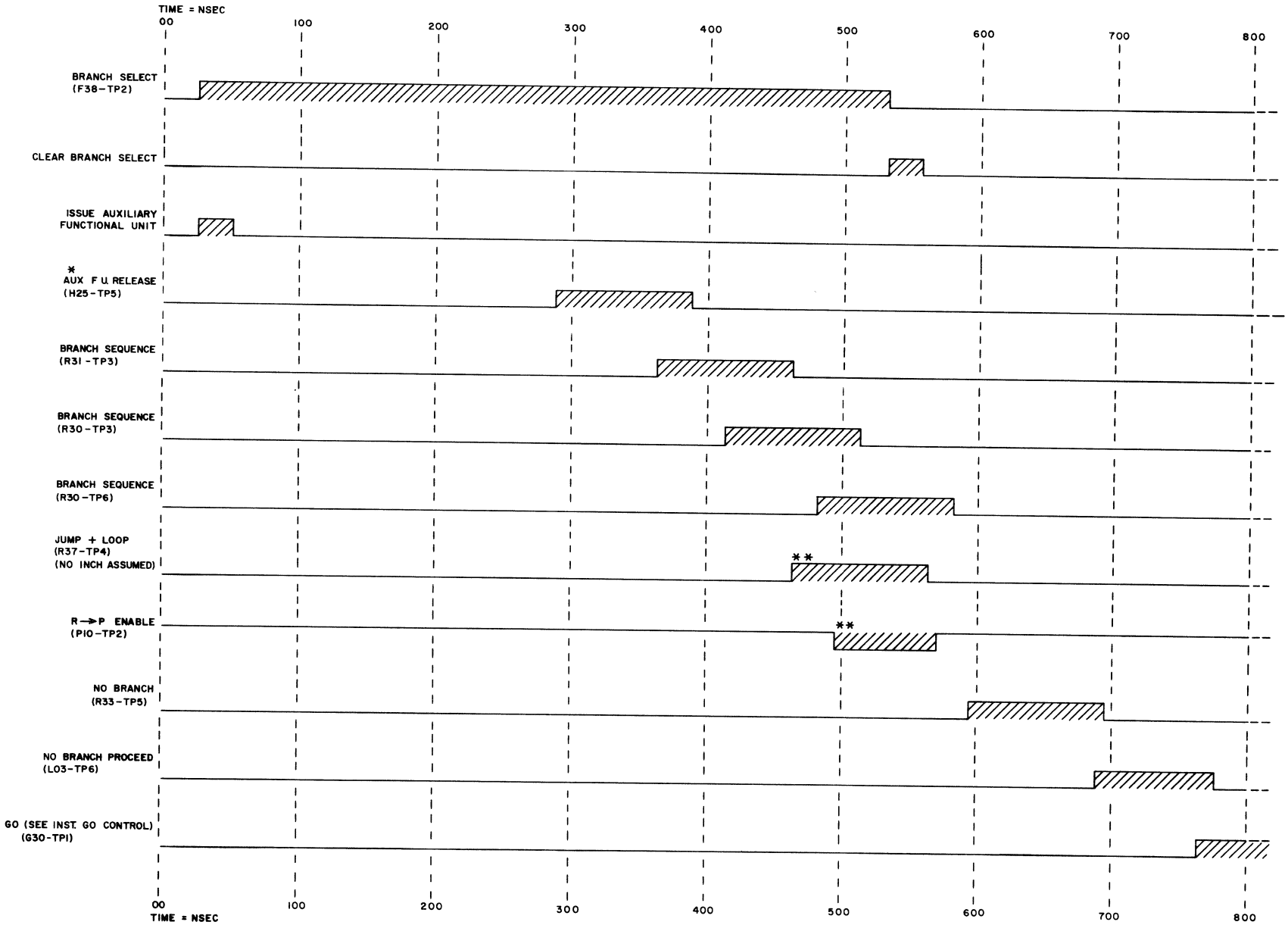
Page	Title
	Branch
1	Block Diagram Conditional Branch (03-07)
2.1	Timing Chart
2.3	(R-P) Sequence
3	R/P Control, Error Delay
5	(L-T) Sequence
6	Loop Test
7	D - (L-T) Sequence
9	Branch Select Sequence
11	Jump + Loop Sequence
13	No Branch Sequence, Loop Sequence, Jump Sequence
14.1	Instruction Go Control Part 1, Serials 1-7
14.21	Instruction Go Control Part 1, Serials 8 and up
14.3	Instruction Go Control Part 2
14.4	Return Jump Sequence
14.5	Return Jump Part 1
14.7	Return Jump Part 2
14.8	Boolean
15	Block Diagram & Data Flow
16	Card Placement
17	Timing Chart
19	Control
20	Shift
21	Block Diagram
23	Block Diagram
25	Block Diagram Data Flow
26	Card Placement
27	Block Diagram, Timing Chart
29	Mode Bits, jk Constant
31	Timing Chain, Transmit Shift
33	B_j, jk & SK Registers, Direct, Control, Bj Ones Test
35	Shifting Network
37	Feeder Regs., Bj & Xi Drivers, Pack & Unpack
39	Normalize Network, All Zero's Test
41	Exponent Adder
42	Long Add
43	Block Diagram
44	Card Placement
45	Timing Chart
47	Block Diagram, Data Flow
48	Adder
49	Data Path (chB catch. req)
51	Adder, Section 0
53	Adder, Section 1
55	Adder, Section 2
57	Adder, Section 3
59	Adder, Section 4
60	Branch Tests
61	Sign/Zero Tests
62	Branch Tests (continued)
63	Range/Indefinite Tests
64	Multiply
65	Block Diagram
66.1	Timing Chart
66.2	Multiply, Coefficient
67	Coefficient, Block Diagram, 4 Iterations
68	Merge
69	Coefficient, Block Diagram, Merge
70	Multiply, Exponent
71	Exponent, Block Diagram
73	Timing Chain, Chassis 6
75	Timing Chain, Chassis 6
76	Chassis 6 Input Registers
77	Multiplier X_j, Input Register, Chassis 6
79	Multiplicand X_k, Input Register, Chassis 6
80	Sign Record
81	Coefficient Sign Record

FUNCTIONAL UNITS
CONTENTS (CONTINUED)

Page	Title
82	Multiplier X_j
83	Coefficient Multiplier X_j Reg. (Upper)
85	Coefficient Multiplier X_j Reg. (Lower)
86	Multiplicand X_k
87	Coefficient 1, 2 & 3 Times Multiplicand X_k
88	Three-Level Adders, Upper & Lower
89	Coefficient, Upper Adder, 3-Level
91	Coefficient, Lower Adder, 3-Level
92	Partial Sum and Partial Carry Registers
93	Coefficient, Partial Sum & Carry Register Controls
95	Coefficient, Upper Full Adder, 6 Bits
97	Coefficient, Lower Full Adder, 6 Bits
99	Coefficient, Upper 15 Bits of Final Product
101	Coefficient, 96-Bit Product, Double Precision
102	Normalize; Complement
103	Coefficient, Output Network, Normalize, Complement
105	Exponent, Timing Chain, Chassis 2
107	Exponent, Timing Chain, Chassis 2
108	Chassis 2 Input Register
109	X_j Exponent, Chassis 2 Input Register
111	X_k Exponent, Chassis 2 Input Register
112	Exponent Addition
113	Exponent Add, $X_j + X_k$
115	Exponent Add 60_8 (48_{10}), $X_j + X_k + 60_8$
117	Exponent, Single or Double Precision
118	Normalize, Complement
119	Exponent Normalize, Complement
121	Exponent Select Complement, True Value
122	Test Result
123	Exponent, Test Result
125	Exponent, Output Network
126	**Divide**
126.1	Block Diagram
126.2	Card Placement Chassis 2 (A)
127	Card Placement Chassis 2 (B)
128.1	Timing Chart
128.2	Divide, Coefficient
129	Coefficient, Block Diagram
130	Divide, Exponent
131	Exponent, Block Diagram
132	Divide Example
133	Timing Chain 1
135	Timing Chain 2
136	Chassis 2 Input Registers
137	Divide X_j, Chassis 2 Input Register
139	Divide X_k, Chassis 2 Input Register
140	Dividend X_j; Round
141	Coefficient X_j (Dividend) Registers
142	Divisor X_k
143	Coefficient X_k (Divisor) Registers; Times 1, 2 and 3
145	Coefficient Carries on 3 Times X_k, 1
147	Coefficient Carries on 3 Times X_k, 2
148	Subtraction Networks
149	Coefficient Subtract X_k From X_j
151	Coefficient Borrows on Subtract X_j From X_k, 1
153	Coefficient Borrows on Subtract X_j From X_k, 2
155	Coefficient Borrows on Subtract X_j From X_k, 3
156	2-Bit Quotient; Quotient Left - Shift Register
157	Coefficient, Form 2-Bit Quotient
159	Coefficient, Quotient Left-Shift Register
160	Normalize
161	Coefficient, Quotient Output Network
162	Gate Quotient Output
163	Coefficient, Gate Quotient Output
165	X_j Exponent, Chassis 2 Input Register
167	X_k Exponent, Chassis 2 Input Register



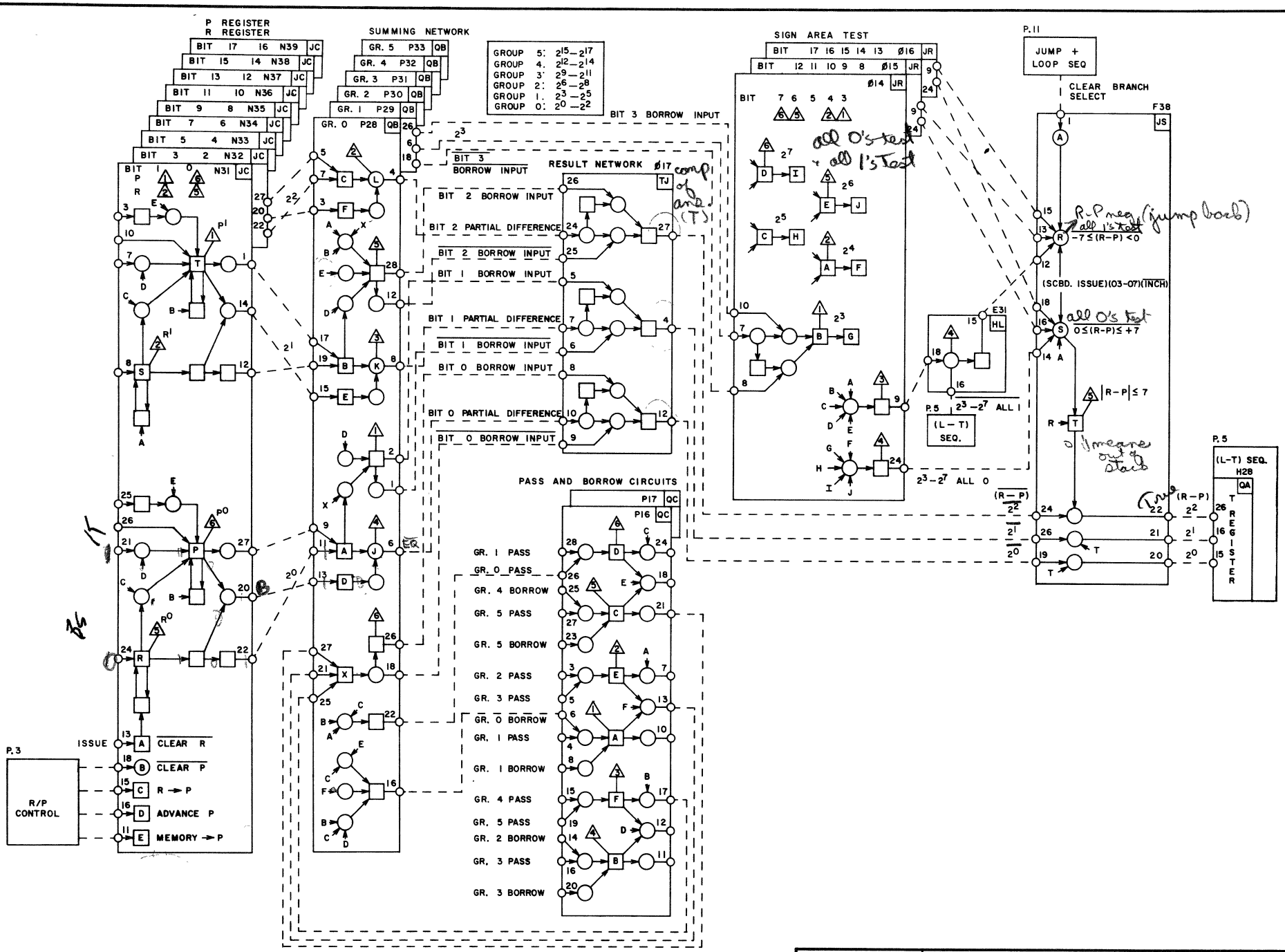
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR BRANCH UNIT BLOCK DIAGRAM CONDITIONAL BRANCH (03-07)	PRODUCT: 6601	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: D
		SHEET: 203	PAGE: 1



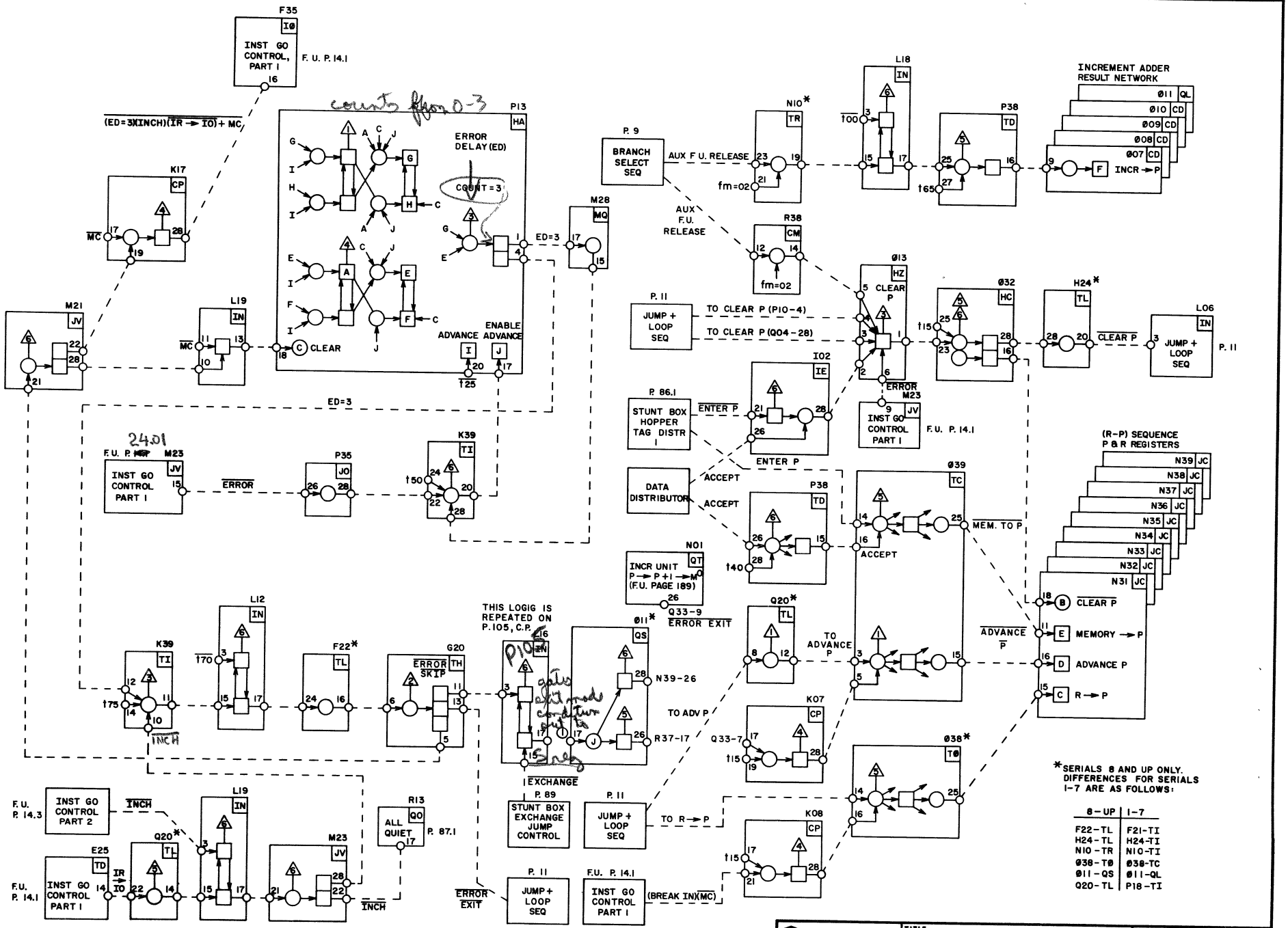
* THE AUXILIARY FUNCTIONAL UNIT FOR THIS CHART IS INCREMENT 1.

** ASSUME THE CONDITIONAL TEST IS MET IN THE AUXILIARY FUNCTIONAL UNIT (INCR 1).

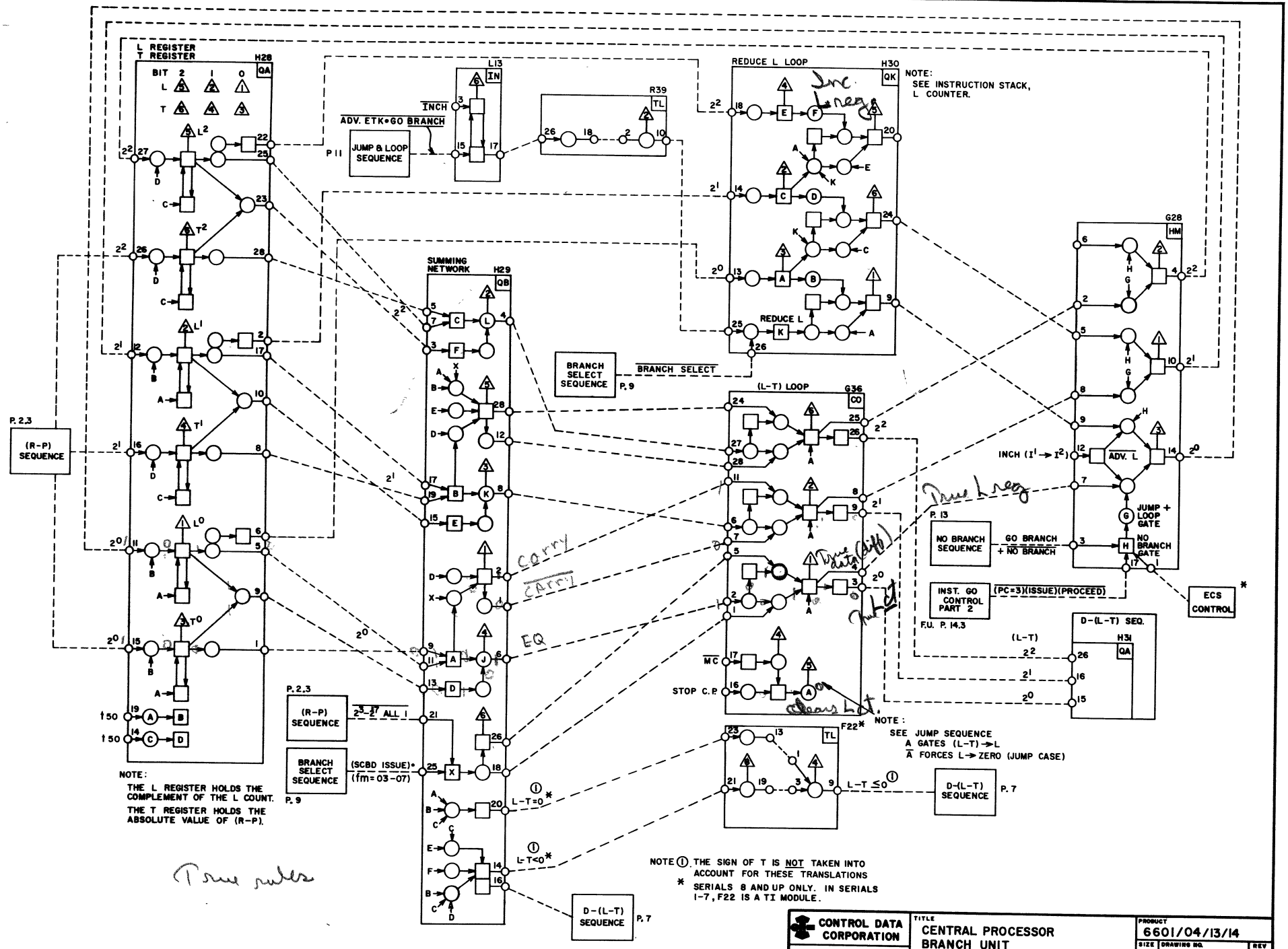
CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR BRANCH UNIT TIMING CHART	PRODUCT 6601 SIZE C | DRAWING NO. 60119300 | REV D SHEET 202 | PAGE 2.1	



CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR BRANCH UNIT (R-P) SEQUENCE	PRODUCT: 6601/04	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: K
			SHEET: 96 PAGE: 2.3



CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR BRANCH UNIT R/P CONTROL, ERROR DELAY	PRODUCT: 6601/04/13/14	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: BT
			SHEET: 97 PAGE: 3

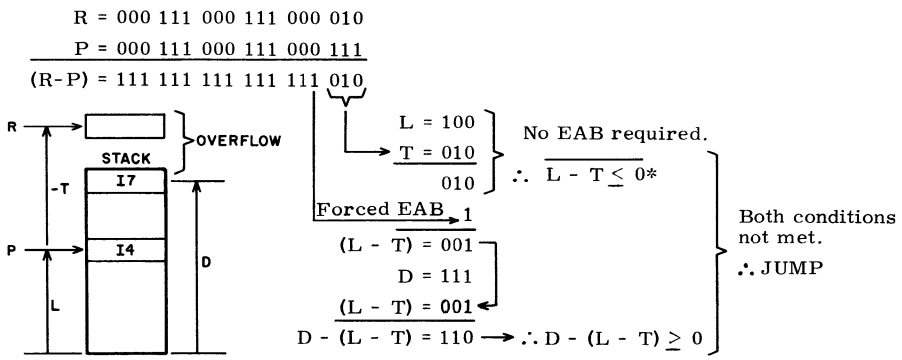


CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR BRANCH UNIT (L-T) SEQUENCE	PRODUCT 6601/04/13/14	
		SIZE C	DRAWING NO. 60119300 REV 8T
			SHEET 98 5

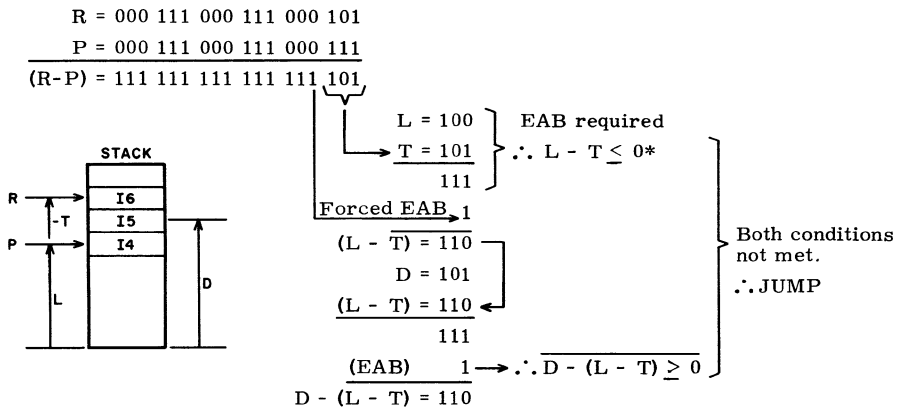
To generate an In Stack signal at test point six of F38, either gate X or gate Y must be made. Gate Y can be made only when $T \geq 0$, indicating that the branch is down the stack or forward in the program. For this case, $L - T \geq 0$ is a sufficient condition for loop. The D - (L - T) difference is taken, but not used.

Gate X can be made only when $T < 0$, indicating that the branch is up the stack or backward in the program. If $L - T \leq 7$, then $D - (L - T) \geq 0$ is a sufficient condition for loop. Due to hardware limitations of the three bit summing network, however, overflow is not recognized ($L - T > 7$). Thus a second term must be ANDed with the EAB term on I03 to cover all cases. In each of the examples below, one of the two terms is zero, thereby prohibiting a loop.

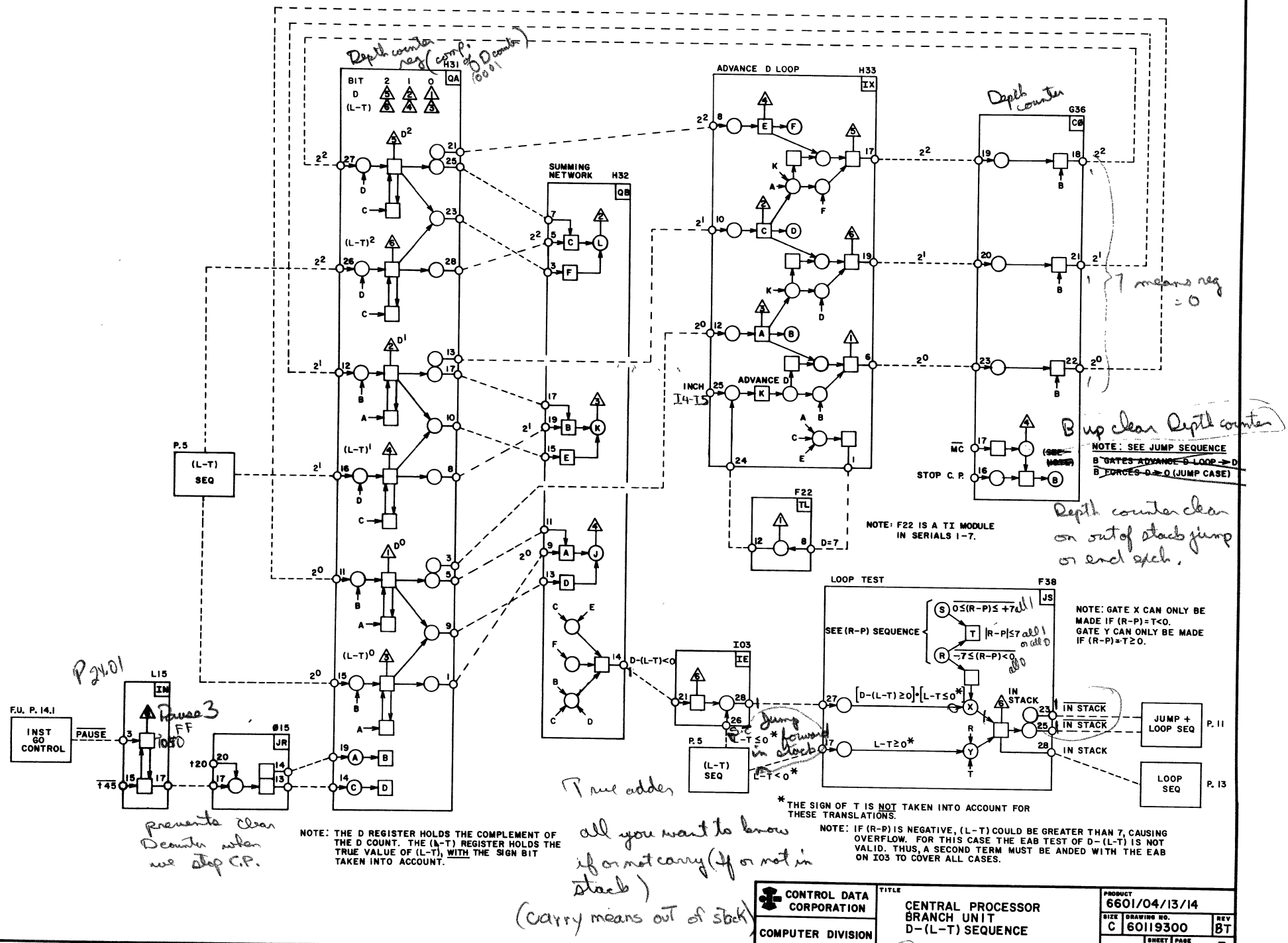
Example 1. Overflow



Example 2. No Overflow, but D is too small



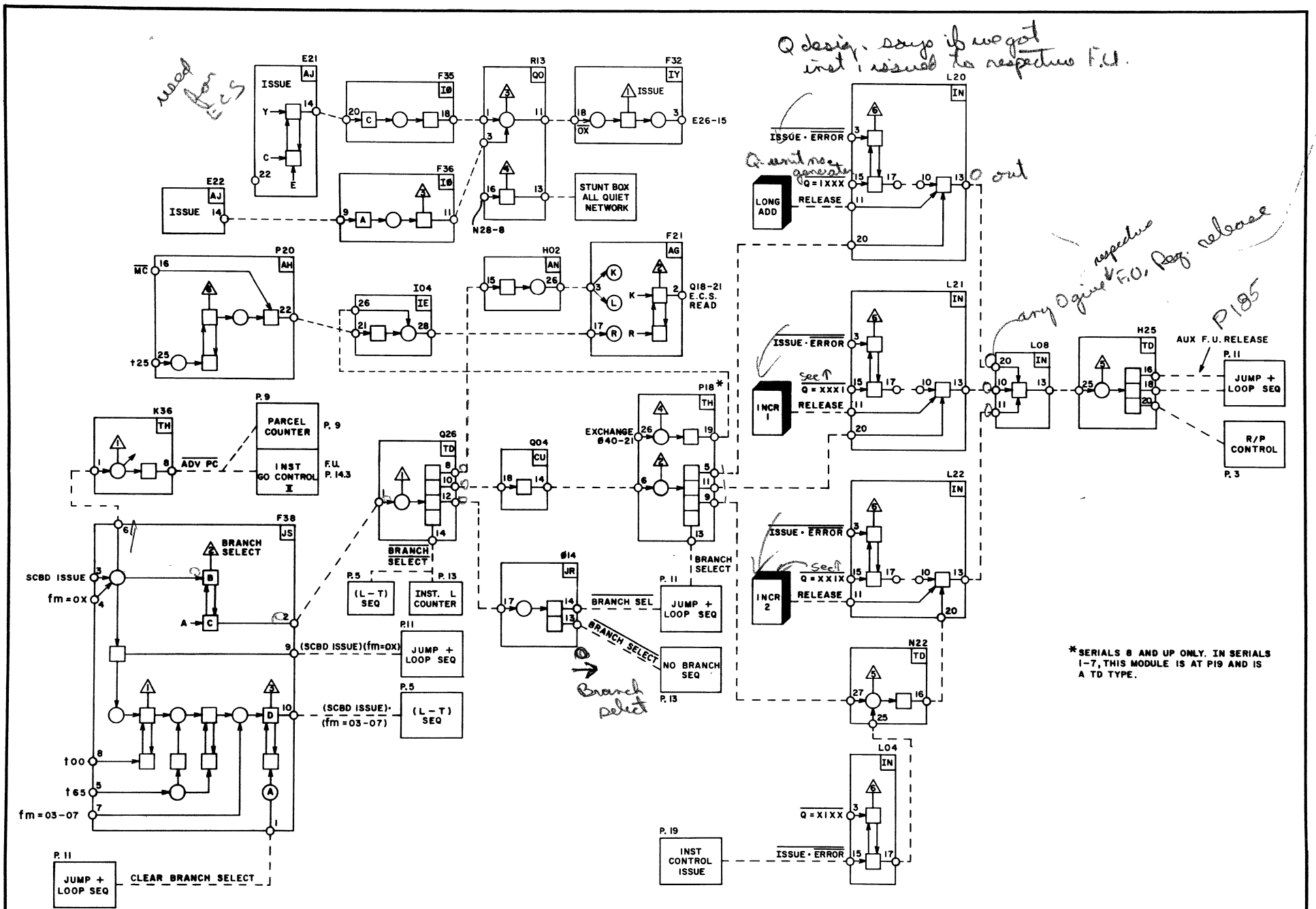
*The sign of T is not taken into account for these translations.



CONTROL DATA CORPORATION
COMPUTER DIVISION

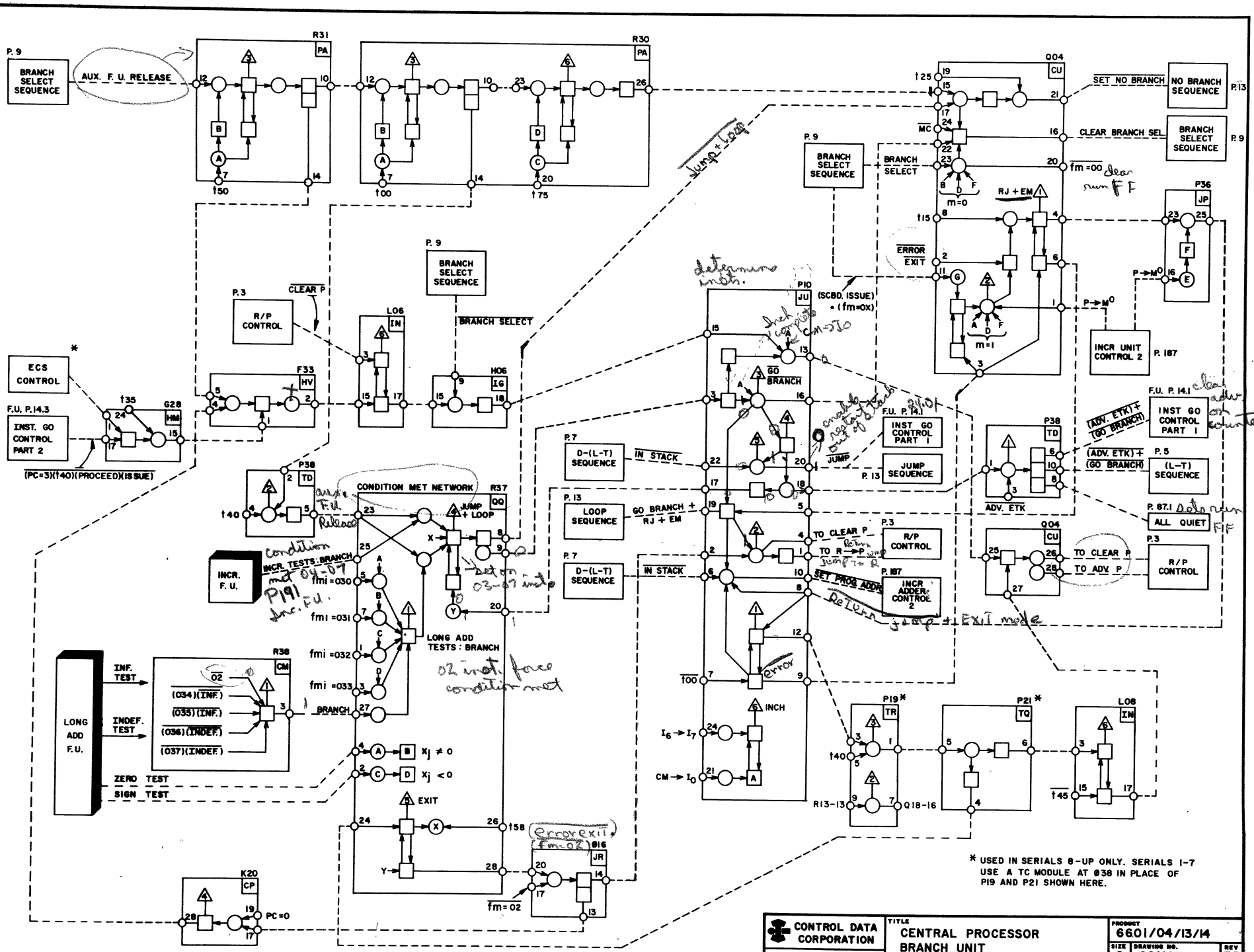
TITLE
CENTRAL PROCESSOR
BRANCH UNIT
D-(L-T) SEQUENCE

PRODUCT 6601/04/13/14
SIZE C | DRAWING NO. 60119300 | REV BT
SHEET 99 | PAGE 7

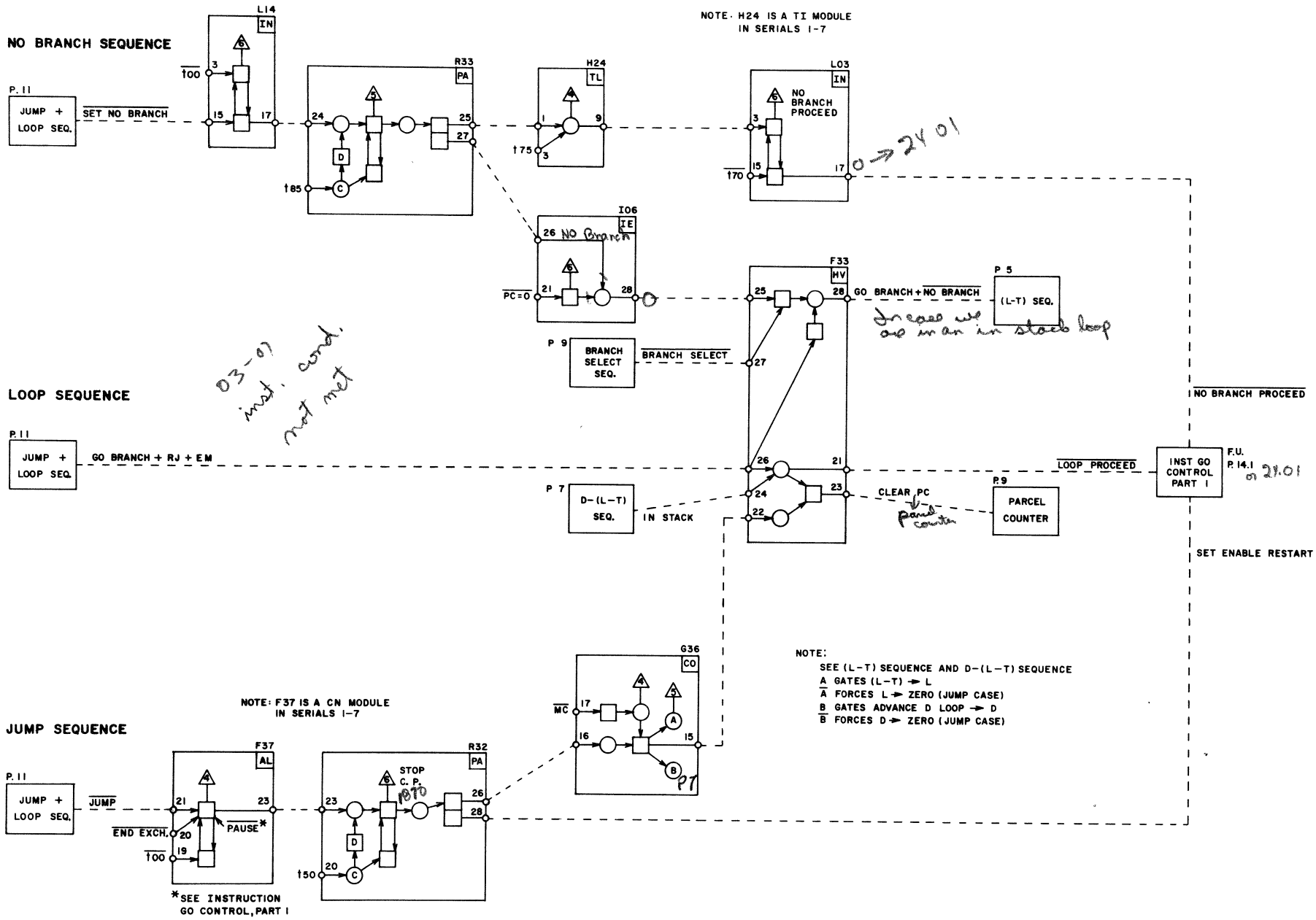


*SERIALS 8 AND UP ONLY. IN SERIALS 1-7, THIS MODULE IS AT P19 AND IS A TD TYPE.

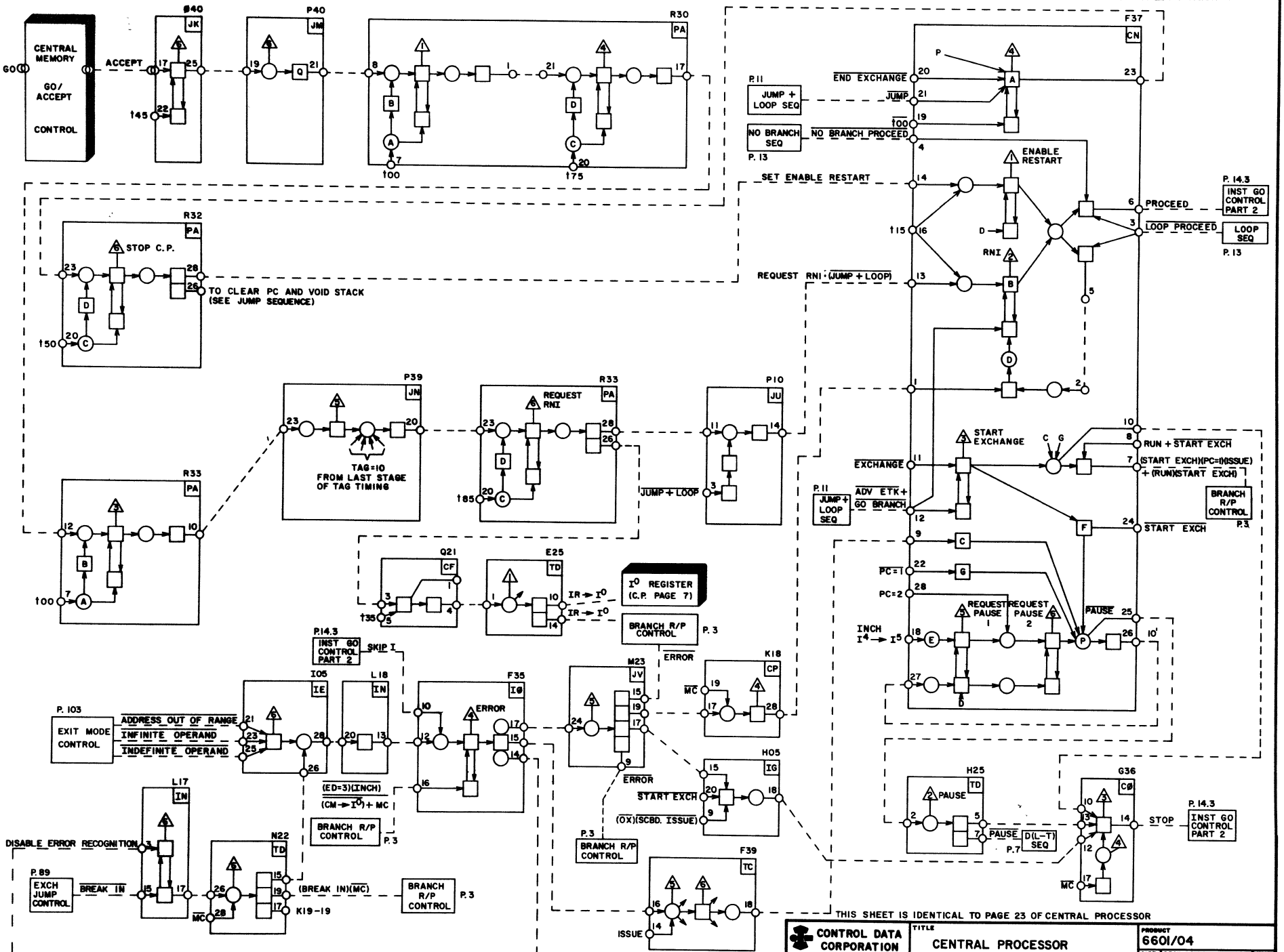
CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR BRANCH UNIT BRANCH SELECT SEQUENCE	PRODUCT 6601/04/13/14	
		SIZE C	DRAWING NO. 60119300
		REV BT	
		SHEET 100	PAGE 9



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR BRANCH UNIT JUMP + LOOP SEQUENCE	PRODUCT 6601/04/13/14 SIZE C · DRAWING NO. 60119300 · REV BT SHEET 101 · PAGE 11



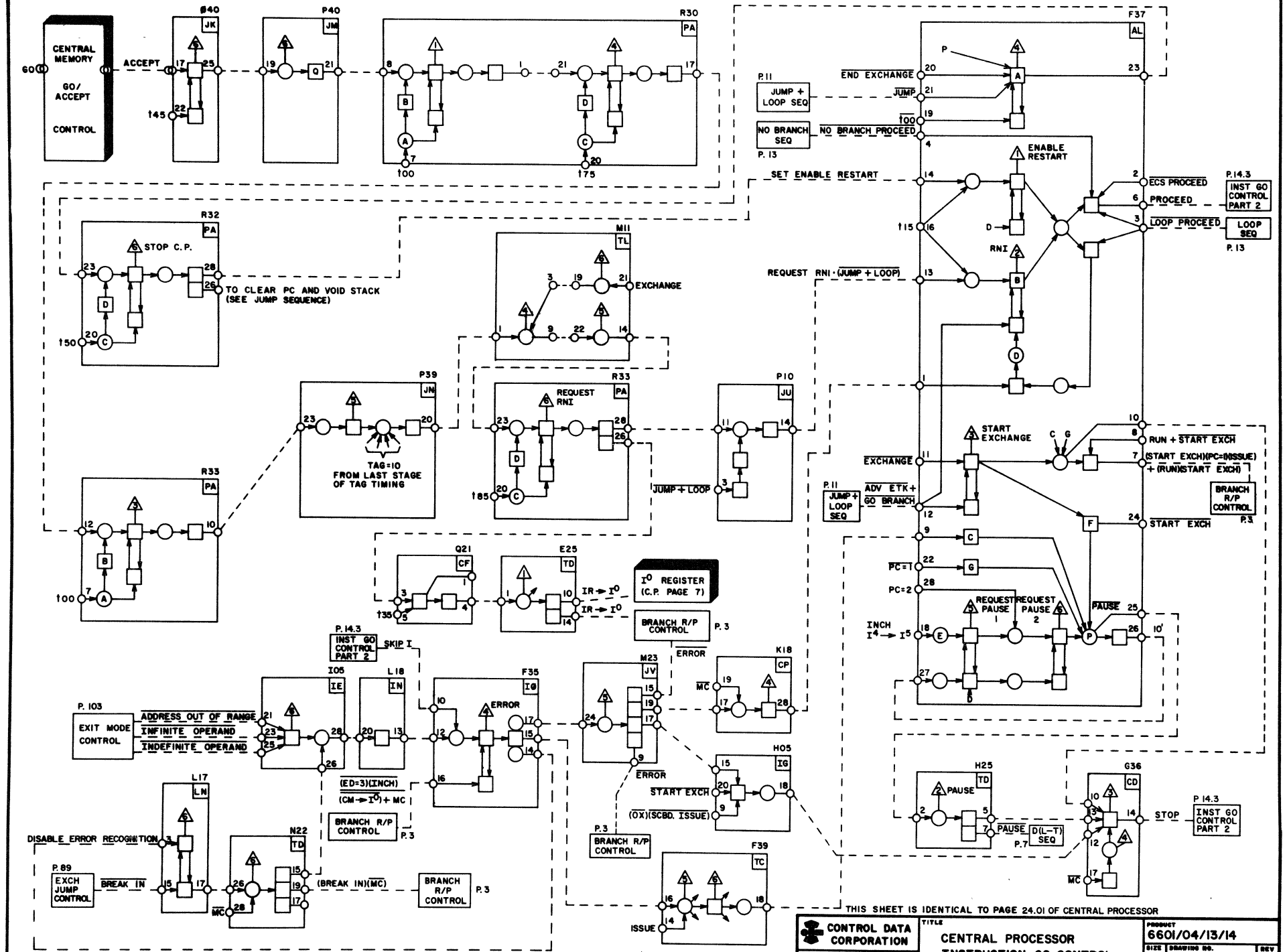
CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR BRANCH UNIT NO BRANCH SEQUENCE, LOOP SEQ., JUMP SEQ.	PRODUCT 6601/04/13/14	
		SIZE C	DRAWING NO. 60119300 REV M
		SHEET 102	PAGE 13

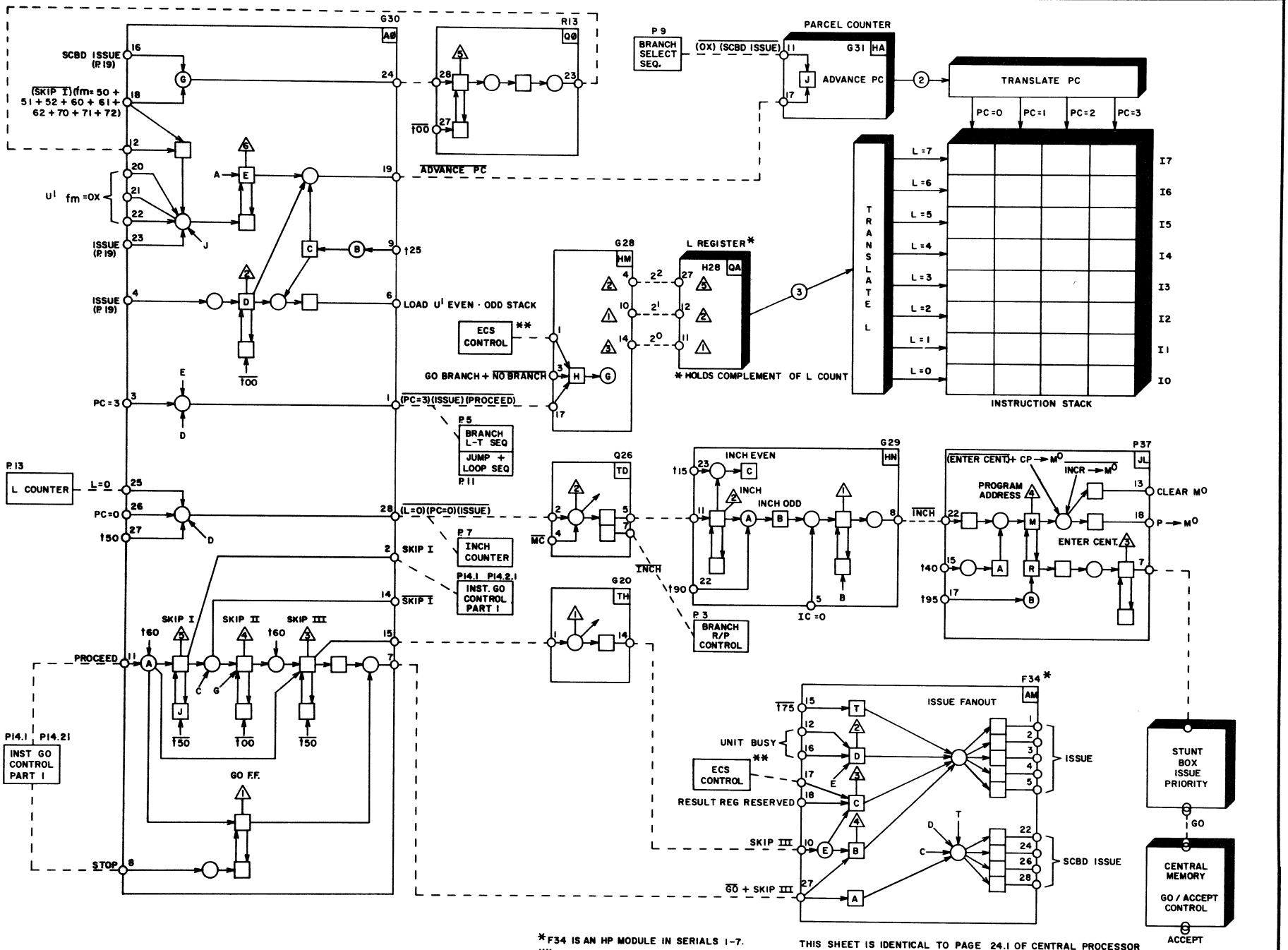


CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE: CENTRAL PROCESSOR INSTRUCTION GO CONTROL PART 1, SERIALS 1-7

PRODUCT: 6601/04
SIZE: C | DRAWING NO.: 60119300 | REV: BT
SHEET: 261 | PAGE: 14.1


CENTRAL MEMORY
GO/ ACCEPT CONTROL
ACCEPT
P40
R30
F37
P.11
JUMP + LOOP SEQ
NO BRANCH SEQ
P. 13
END EXCHANGE
JUMP
NO BRANCH PROCEED
ENABLE RESTART
SET ENABLE RESTART
ECS PROCEED
PROCEED
LOOP PROCEED
P.14.3
INST GO CONTROL PART 2
LOOP SEQ
P. 13
R32
STOP C.P.
TO CLEAR PC AND VOID STACK (SEE JUMP SEQUENCE)
M11
EXCHANGE
REQUEST RNI · (JUMP + LOOP)
RNI
P39
TAG=10 FROM LAST STAGE OF TAG TIMING
R33
REQUEST RNI
P10
JUMP + LOOP
START EXCHANGE
EXCHANGE
ADV ETK + GO BRANCH
RUN + START EXCH
(START EXCH)(PC=MISSUE) + (RUN)(START EXCH)
BRANCH R/P CONTROL
P.3
START EXCH
Q21
E25
I0 REGISTER (C.P. PAGE 7)
IR → I0
BRANCH R/P CONTROL
P. 3
PC=1
PC=2
REQUEST PAUSE 1
REQUEST PAUSE 2
PAUSE
INCH
I4 → I5
I05
L18
F35
K18
M23
ERROR
SKIP I
P. 103
EXIT MODE CONTROL
ADDRESS OUT OF RANGE
INFINITE OPERAND
INDEFINITE OPERAND
(ED=3)(INCH)
(CM → I0) + MC
H05
START EXCH
(OX)(SCBD. ISSUE)
H25
PAUSE
D(L-T) SEQ
P.7
G36
STOP
L17
N22
DISABLE ERROR RECOGNITION
P. 89
EXCH JUMP CONTROL
BREAK IN
(BREAK IN)(MC)
F39
ISSUE
THIS SHEET IS IDENTICAL TO PAGE 24.01 OF CENTRAL PROCESSOR
CONTROL DATA CORPORATION
COMPUTER DIVISION
CENTRAL PROCESSOR INSTRUCTION GO CONTROL PART 1, SERIALS 8-UP
6601/04/13/14
C
60119300
M
262
14.21



*F34 IS AN HP MODULE IN SERIALS 1-7.
**USED IN SERIALS 8-UP ONLY.

THIS SHEET IS IDENTICAL TO PAGE 24.1 OF CENTRAL PROCESSOR

CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE: CENTRAL PROCESSOR INSTRUCTION GO CONTROL PART 2

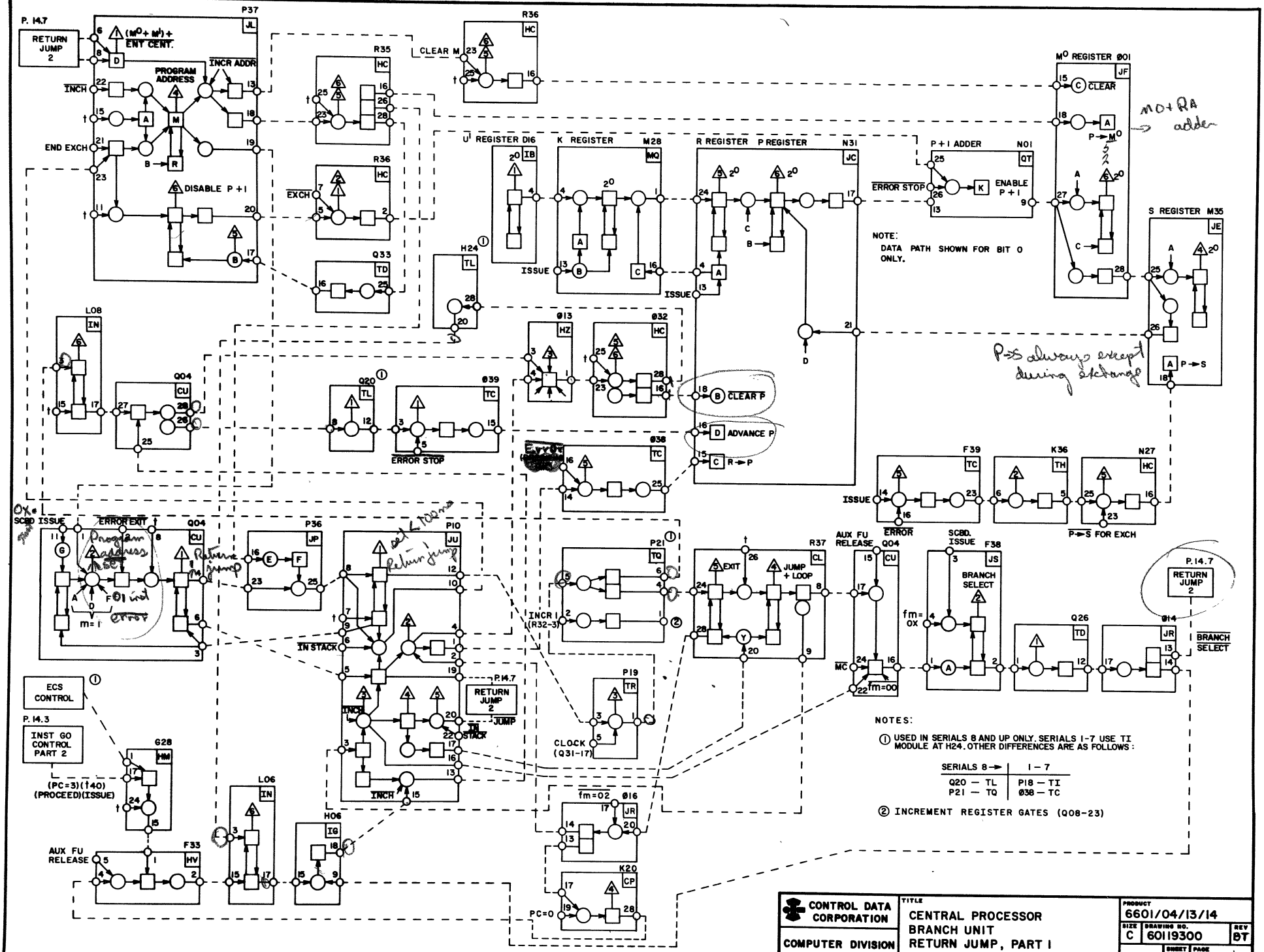
PRODUCT: 6601/04/13/14
SIZE: C | DRAWING NO.: 60119300 | REV: BT
SHEET 219 | PAGE 14.3

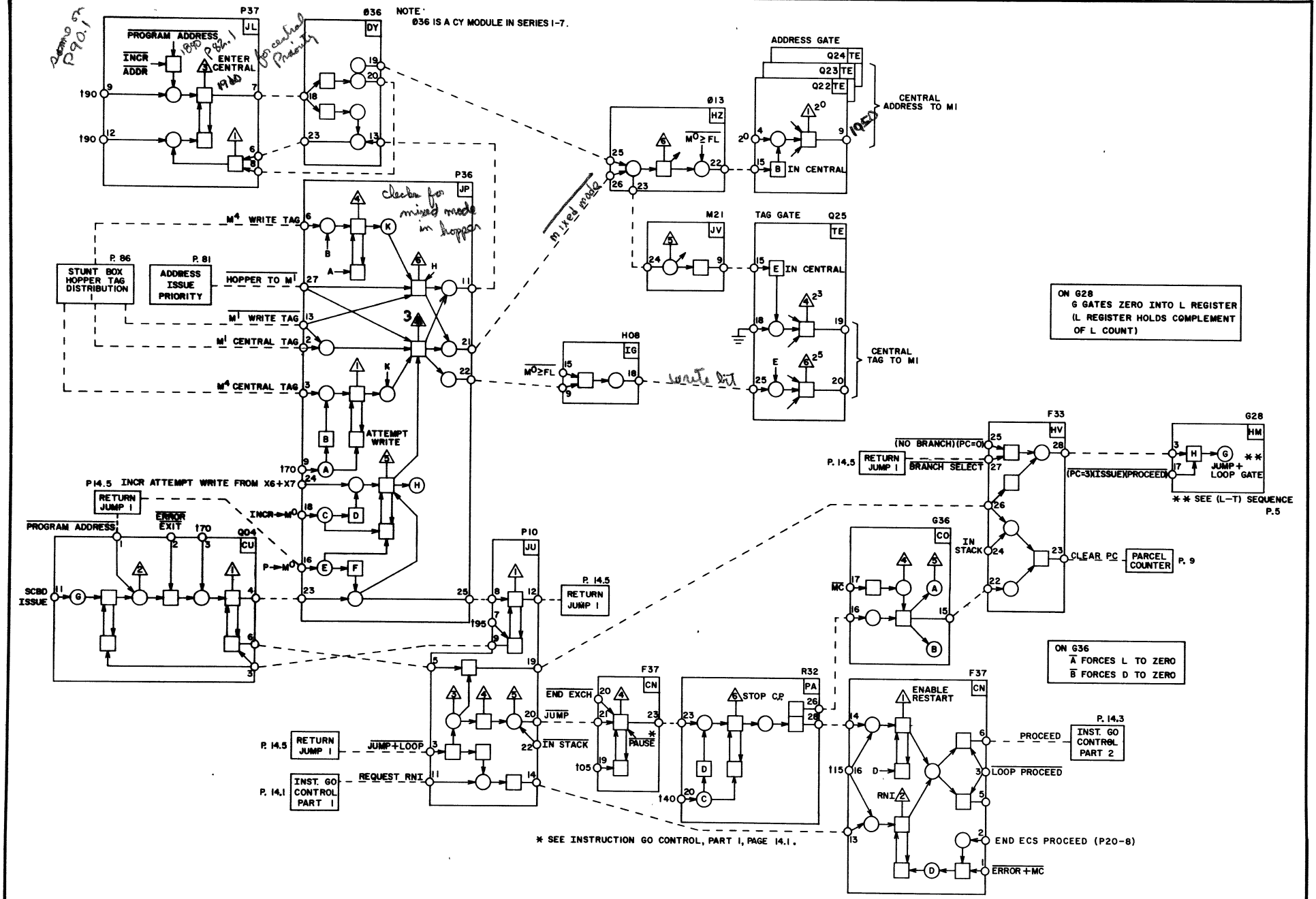
RETURN JUMP SEQUENCE

$U^1 \rightarrow U^2$	01 instruction issued to U^2 K field (jump address) gated to K register
Issue 01 to scoreboard	
K → R	Jump address gated to R register
P+1 → S	Program address plus one (return address) gated to S register
S → Memory	To form 0400(P+1)0---0 at jump address
(Disable P+1)	
R → P	Jump address gated to P register
$P \rightarrow M^O$	Jump address gated to M^O
(Enable P+1)	
$M^O + RA \rightarrow M^1$	Absolute jump address and tag (50) gated to stunt box
Advance P	Jump address plus one gated to P register
(Disable P+1)	
Void instruction stack	Set PC = 0 Set L = 0* Set D = 0**
$P \rightarrow M^O$	Jump address plus one gated to M^O
$M^O + RA \rightarrow M^1$	Absolute jump address plus one and tag (10) gated to stunt box RNI initiated

*The L register holds the complement of the L count.

**The D register holds the complement of the D count.

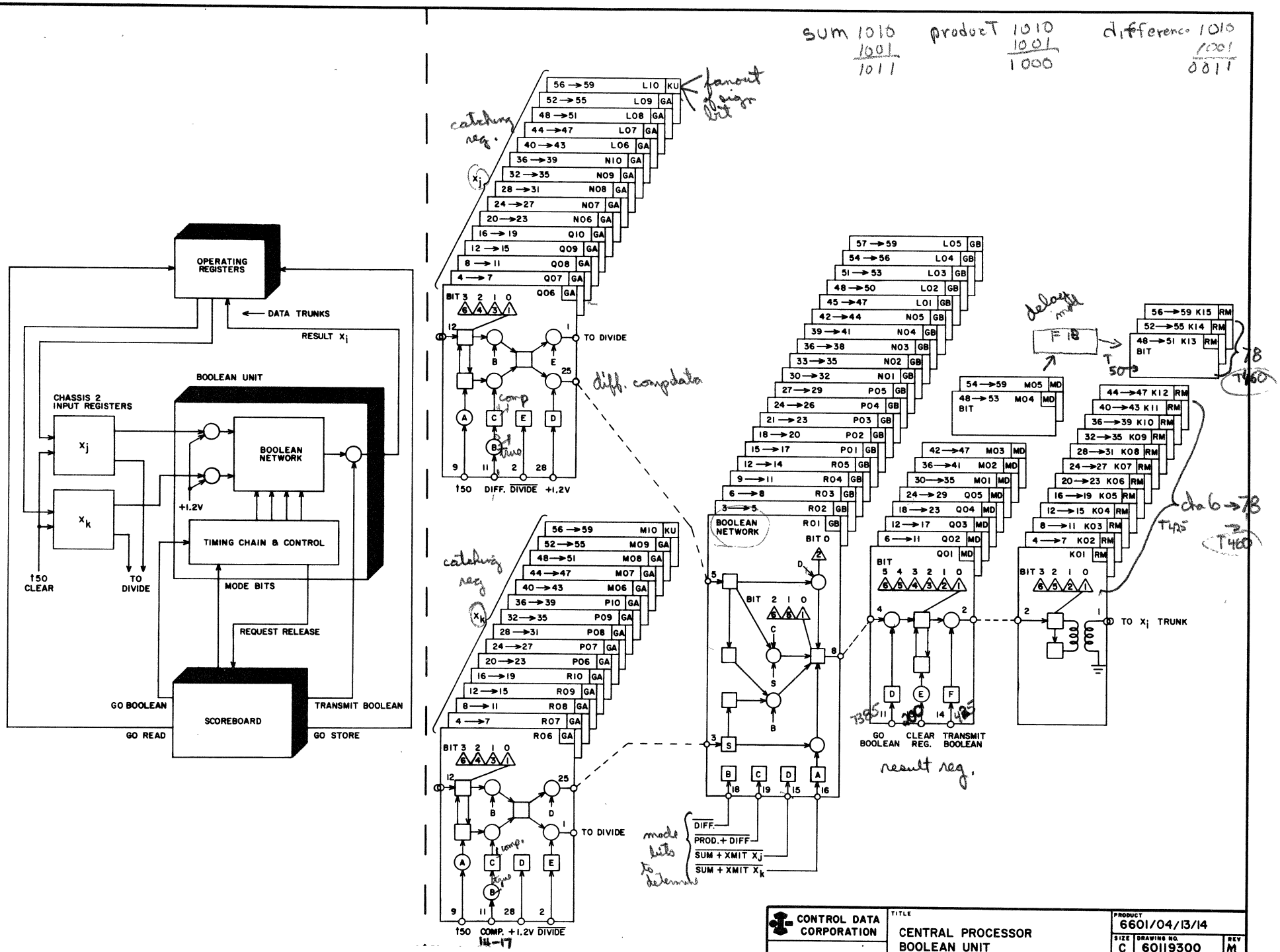

RETURN JUMP 2
P. 14.7
P37
JL
(M0 + M1) + ENT CENT.
INCR ADDR
PROGRAM ADDRESS
INCH
END EXCH
DISABLE P + 1
R35
HC
R36
HC
EXCH
Q33
TD
CLEAR M
R36
HC
U REGISTER DI6
IB
K REGISTER
M28
MQ
R REGISTER
P REGISTER
N31
JC
P + 1 ADDER
N01
QT
ERROR STOP
ENABLE P + 1
ISSUE
H24
TL
M0 REGISTER Ø01
JF
CLEAR
P → M0
M0 + RA adder
S REGISTER M35
JE
P → S
P → S always except during skchange
NOTE:
DATA PATH SHOWN FOR BIT 0 ONLY.
L08
IN
Q04
CU
Q20
TL
Ø39
TC
ERROR STOP
Ø13
HZ
Ø32
HC
B CLEAR P
D ADVANCE P
C R → P
Ø38
TC
F39
TC
K36
TH
N27
HC
ISSUE
ERROR
P → S FOR EXCH
SCBD ISSUE
ERROR EXIT
Q04
CU
P36
JP
P10
JU
P21
TQ
R37
CL
EXIT
JUMP + LOOP
AUX FU RELEASE
Q04
CU
SCBD. ISSUE
F38
JS
BRANCH SELECT
P. 14.7
RETURN JUMP 2
BRANCH
SELECT
Q26
TD
Ø14
JR
INCR I (R32-3)
IN STACK
INCH
P19
TR
CLOCK (Q31-17)
ECS CONTROL
P. 14.3
INST GO CONTROL PART 2
(PC=3)(f40) (PROCEED)(ISSUE)
G28
HM
AUX FU RELEASE
F33
HV
L06
IN
H06
IG
fm=02
Ø16
JR
K20
CP
PC=0
MC
fm=00
fm=0
NOTES:
① USED IN SERIALS 8 AND UP ONLY. SERIALS 1-7 USE TI MODULE AT H24. OTHER DIFFERENCES ARE AS FOLLOWS:
SERIALS 8 → | 1-7
Q20 — TL | P18 — TI
P21 — TQ | Ø38 — TC
② INCREMENT REGISTER GATES (Q08-23)
CONTROL DATA CORPORATION
COMPUTER DIVISION
TITLE
CENTRAL PROCESSOR
BRANCH UNIT
RETURN JUMP, PART 1
PRODUCT
6601/04/13/14
SIZE C
DRAWING NO. 60119300
REV 8T
SHEET 233
PAGE 14.5

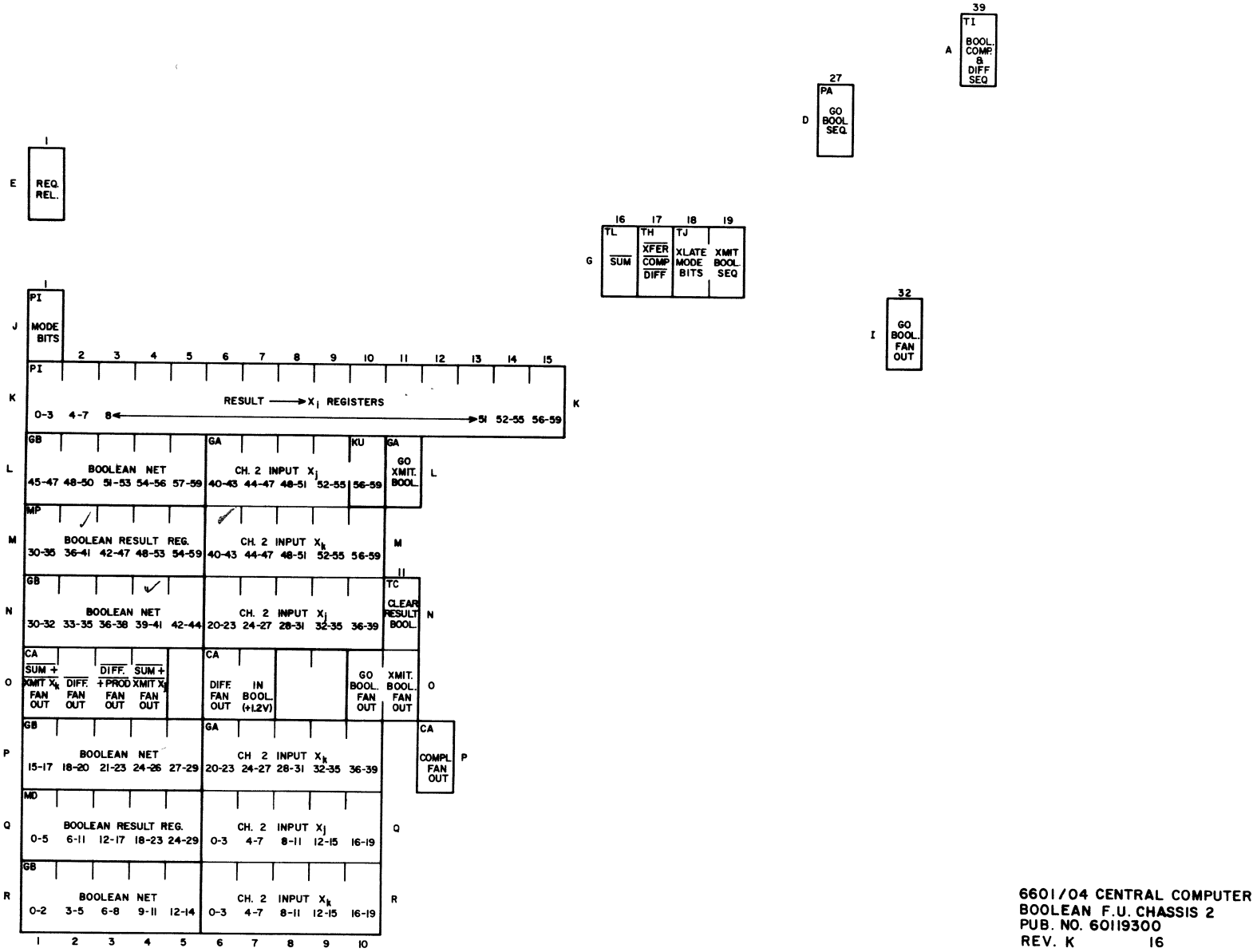


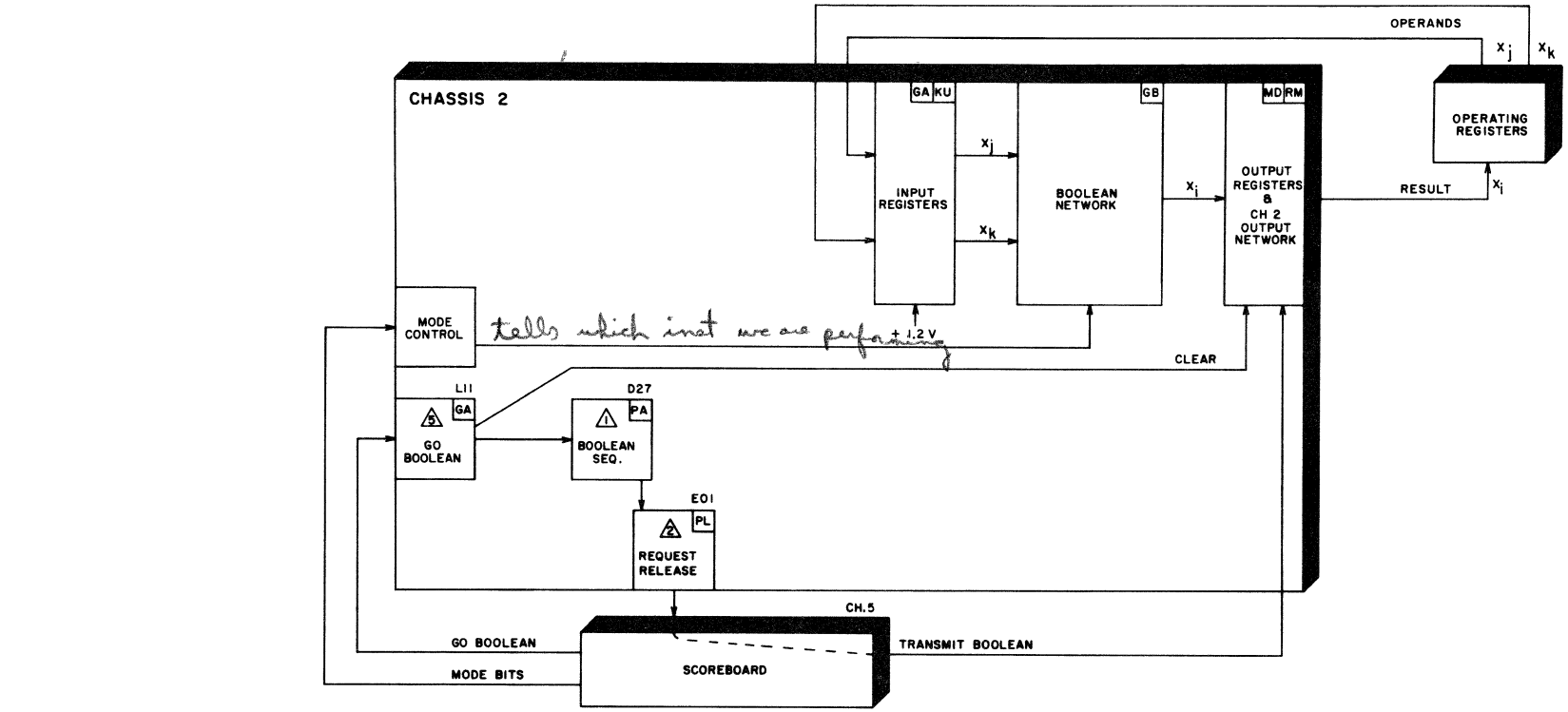
CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR BRANCH UNIT RETURN JUMP, PART 2	PRODUCT 6601/04/13/14	
		SIZE C	DRAWING NO. 60119300 REV 8T
		SHEET 234	PAGE 14.7

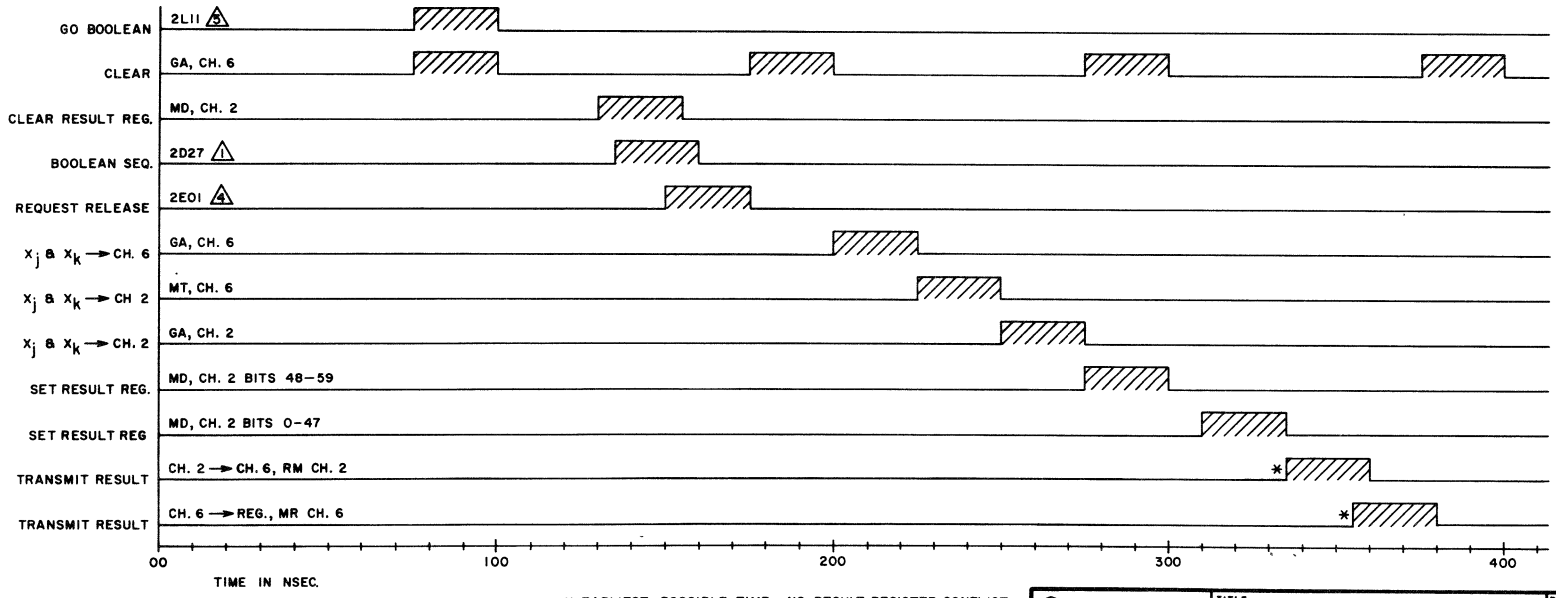
BOOLEAN

Logical operations in the CONTROL DATA 6601 and 6604 Central Processor are performed by the Boolean Unit located on chassis 2. This unit contains all the hardware necessary to perform a 60-bit transmit, logical product, sum or difference of one or two 60-bit operands. The time required for a single operation is 0.3 microseconds (3 minor cycles).


sum 1010 + 1001 = 1011
product 1010 × 1001 = 1000
difference 1010 − 1001 = 0011
OPERATING REGISTERS
DATA TRUNKS
RESULT X_j
BOOLEAN UNIT
CHASSIS 2 INPUT REGISTERS
X_j
X_k
BOOLEAN NETWORK
+1.2V
TIMING CHAIN & CONTROL
T50 CLEAR
TO DIVIDE
MODE BITS
REQUEST RELEASE
GO BOOLEAN
GO READ
SCOREBOARD
TRANSMIT BOOLEAN
GO STORE
catching reg. X_j
fanout of sign bit
56 → 59 L10 KU
52 → 55 L09 GA
48 → 51 L08 GA
44 → 47 L07 GA
40 → 43 L06 GA
36 → 39 N10 GA
32 → 35 N09 GA
28 → 31 N08 GA
24 → 27 N07 GA
20 → 23 N06 GA
16 → 19 Q10 GA
12 → 15 Q09 GA
8 → 11 Q08 GA
4 → 7 Q07 GA
Q06 GA
BIT 3 2 1 0
TO DIVIDE
diff. comp data
comp
true
T50 DIFF. DIVIDE +1.2V
catching reg X_k
56 → 59 M10 KU
52 → 55 M09 GA
48 → 51 M08 GA
44 → 47 M07 GA
40 → 43 M06 GA
36 → 39 P10 GA
32 → 35 P09 GA
28 → 31 P08 GA
24 → 27 P07 GA
20 → 23 P06 GA
16 → 19 R10 GA
12 → 15 R09 GA
8 → 11 R08 GA
4 → 7 R07 GA
R06 GA
BIT 3 2 1 0
TO DIVIDE
comp.
true
T50 COMP. +1.2V DIVIDE
14-17
BOOLEAN NETWORK
57 → 59 L05 GB
54 → 56 L04 GB
51 → 53 L03 GB
48 → 50 L02 GB
45 → 47 L01 GB
42 → 44 N05 GB
39 → 41 N04 GB
36 → 38 N03 GB
33 → 35 N02 GB
30 → 32 N01 GB
27 → 29 P05 GB
24 → 26 P04 GB
21 → 23 P03 GB
18 → 20 P02 GB
15 → 17 P01 GB
12 → 14 R05 GB
9 → 11 R04 GB
6 → 8 R03 GB
3 → 5 R02 GB
R01 GB
BIT 0
BIT 2 1 0
DIFF.
PROD. + DIFF
SUM + XMIT X_j
SUM + XMIT X_k
mode bits to determine
delay unit
F18
T50
54 → 59 M05 MD
48 → 53 M04 MD
BIT
42 → 47 M03 MD
36 → 41 M02 MD
30 → 35 M01 MD
24 → 29 Q05 MD
18 → 23 Q04 MD
12 → 17 Q03 MD
6 → 11 Q02 MD
Q01 MD
BIT 5 4 3 2 1 0
GO BOOLEAN
CLEAR REG.
TRANSMIT BOOLEAN
result reg.
56 → 59 K15 RM
52 → 55 K14 RM
48 → 51 K13 RM
BIT
T78
T460
44 → 47 K12 RM
40 → 43 K11 RM
36 → 39 K10 RM
32 → 35 K09 RM
28 → 31 K08 RM
24 → 27 K07 RM
20 → 23 K06 RM
16 → 19 K05 RM
12 → 15 K04 RM
8 → 11 K03 RM
4 → 7 K02 RM
K01 RM
BIT 3 2 1 0
da 6 → 78
T125
T460
TO X_j TRUNK
CONTROL DATA CORPORATION
COMPUTER DIVISION
TITLE
CENTRAL PROCESSOR
BOOLEAN UNIT
BLOCK DIAGRAM & DATA FLOW
PRODUCT 6601/04/13/14
SIZE C
DRAWING NO. 60119300
REV M
SHEET 103
15


39
TI
BOOL. COMP. & DIFF SEQ
A
27
PA
GO BOOL SEQ.
D
1
REQ. REL.
E
16
TL
SUM
17
TH
XFER
COMP
DIFF
18
TJ
XLATE MODE BITS
19
XMIT BOOL SEQ
G
32
GO BOOL. FAN OUT
I
1
PI
MODE BITS
J
2 3 4 5 6 7 8 9 10 11 12 13 14 15
PI
RESULT → X_j REGISTERS
0-3 4-7 8 51 52-55 56-59
K
GB
BOOLEAN NET
45-47 48-50 51-53 54-56 57-59
GA
CH. 2 INPUT X_j
40-43 44-47 48-51 52-55
KU
56-59
GA
GO XMIT. BOOL.
L
MP
BOOLEAN RESULT REG.
30-35 36-41 42-47 48-53 54-59
CH. 2 INPUT X_k
40-43 44-47 48-51 52-55 56-59
M
GB
BOOLEAN NET
30-32 33-35 36-38 39-41 42-44
CH. 2 INPUT X_j
20-23 24-27 28-31 32-35 36-39
11
TC
CLEAR RESULT BOOL.
N
CA
SUM + XMIT X_k FAN OUT
DIFF. FAN OUT
DIFF. + PROD FAN OUT
SUM + XMIT X_j FAN OUT
CA
DIFF. FAN OUT
IN BOOL. (+1.2V)
GO BOOL. FAN OUT
XMIT. BOOL. FAN OUT
O
GB
BOOLEAN NET
15-17 18-20 21-23 24-26 27-29
GA
CH. 2 INPUT X_k
20-23 24-27 28-31 32-35 36-39
CA
COMPL. FAN OUT
P
MD
BOOLEAN RESULT REG.
0-5 6-11 12-17 18-23 24-29
CH. 2 INPUT X_j
0-3 4-7 8-11 12-15 16-19
Q
GB
BOOLEAN NET
0-2 3-5 6-8 9-11 12-14
CH. 2 INPUT X_k
0-3 4-7 8-11 12-15 16-19
R
1 2 3 4 5 6 7 8 9 10



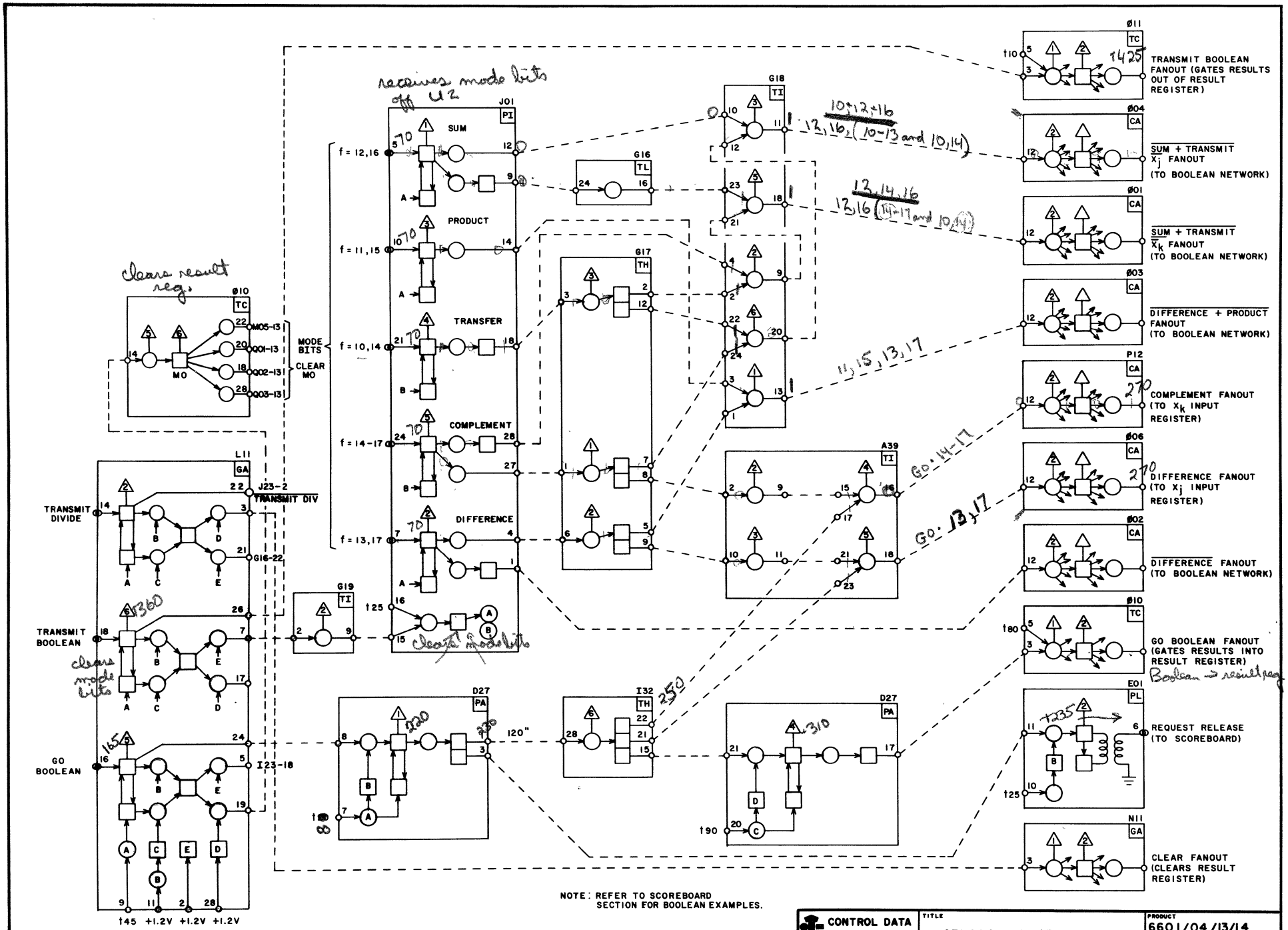


* EARLIEST POSSIBLE TIME – NO RESULT REGISTER CONFLICT.

CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE
CENTRAL PROCESSOR
BOOLEAN UNIT
TIMING CHART

PRODUCT 6601/04
SIZE C | DRAWING NO. 60119300 | REV K
SHEET 104
17

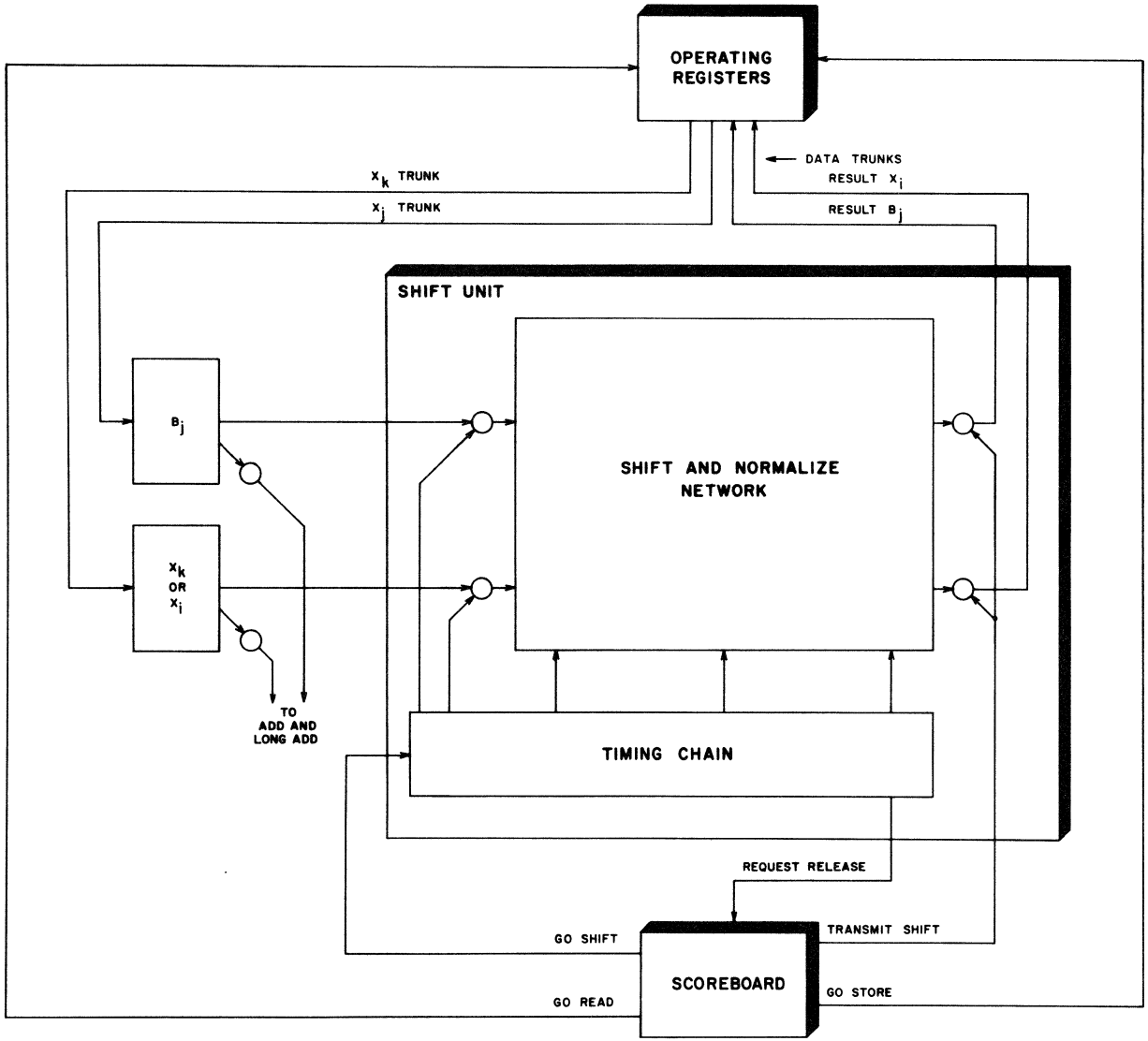


NOTE: REFER TO SCOREBOARD SECTION FOR BOOLEAN EXAMPLES.

CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR BOOLEAN UNIT CONTROL	PRODUCT: 6601/04/13/14	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: 87
			SHEET: 105 19

Shift operations in the CONTROL DATA 6601 and 6604 Central Processor are performed in the Shift Unit located on chassis 8. This unit contains all the hardware necessary to perform a 60-bit right or left shift, normalize, unpack, pack and mask of one 60-bit operand. The time required for all but normalize operations is 0.3 microseconds (3 minor cycles). Normalize operations require 0.4 microseconds (4 minor cycles).

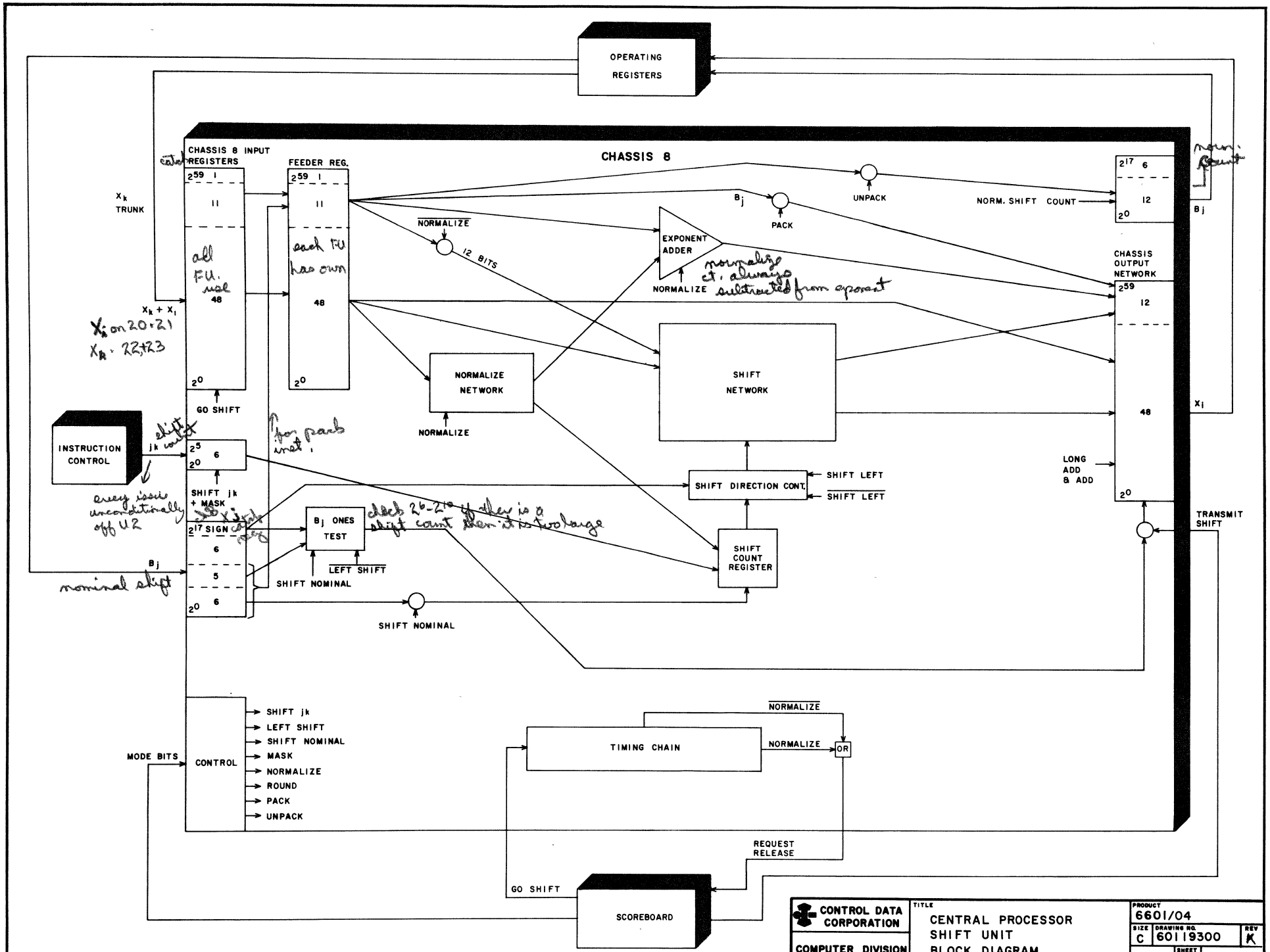
The shift instruction directs the unit to right or left shift the operand either normally or nominally. It may also direct the unit to normalize a floating point quantity from X_k and place the result in X_i and B_j, or unpack X_k into X_i and B_j, or pack X_i and B_j into X_i. The operand may also be rounded when performing a normalize instruction. The shift count is obtained from either the jk portion of the instruction or operating register B_j for right and left shift and mask instructions. The shift count is generated within the unit for normalize operations.

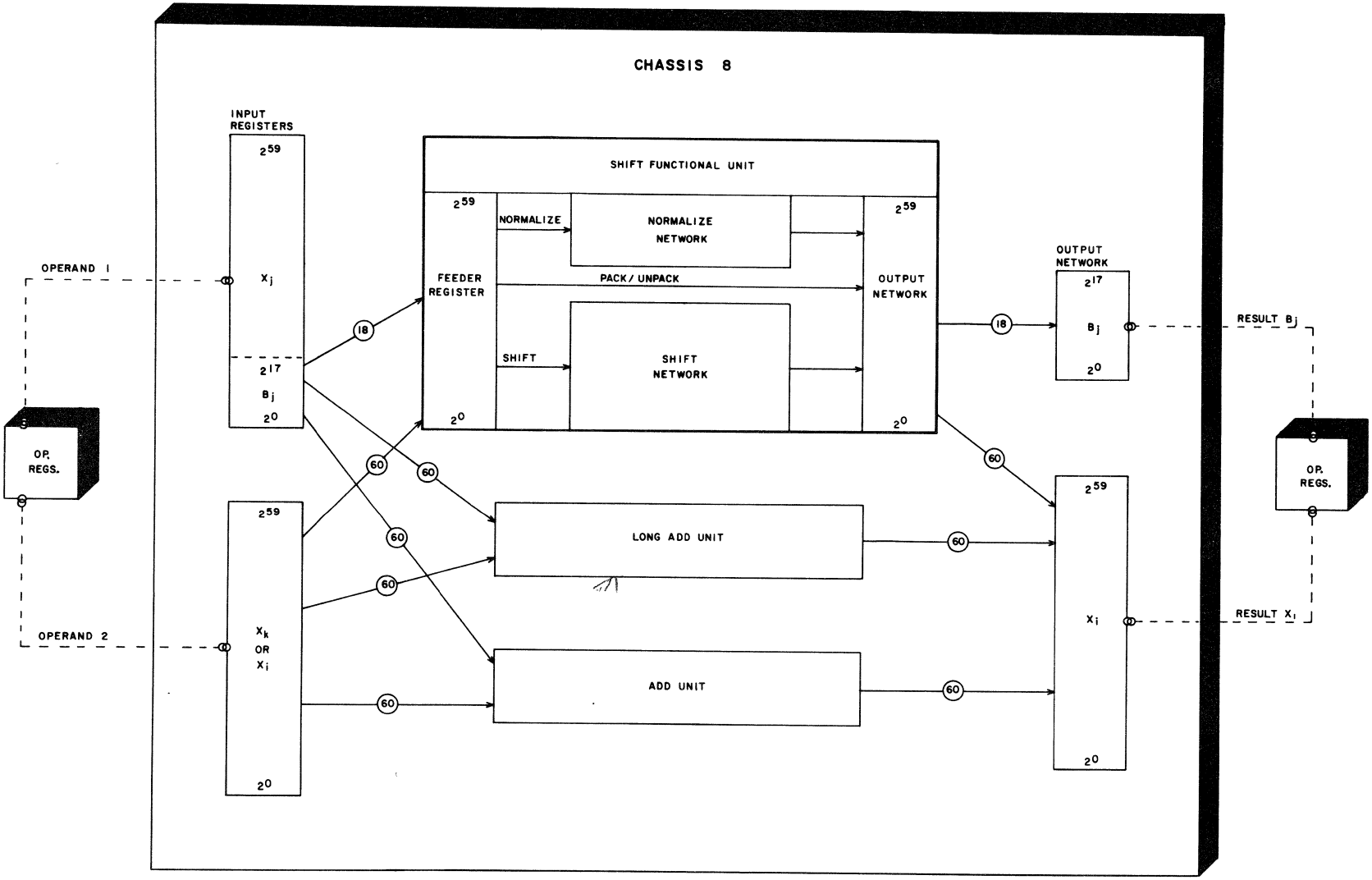


CONTROL DATA CORPORATION
COMPUTER DIVISION

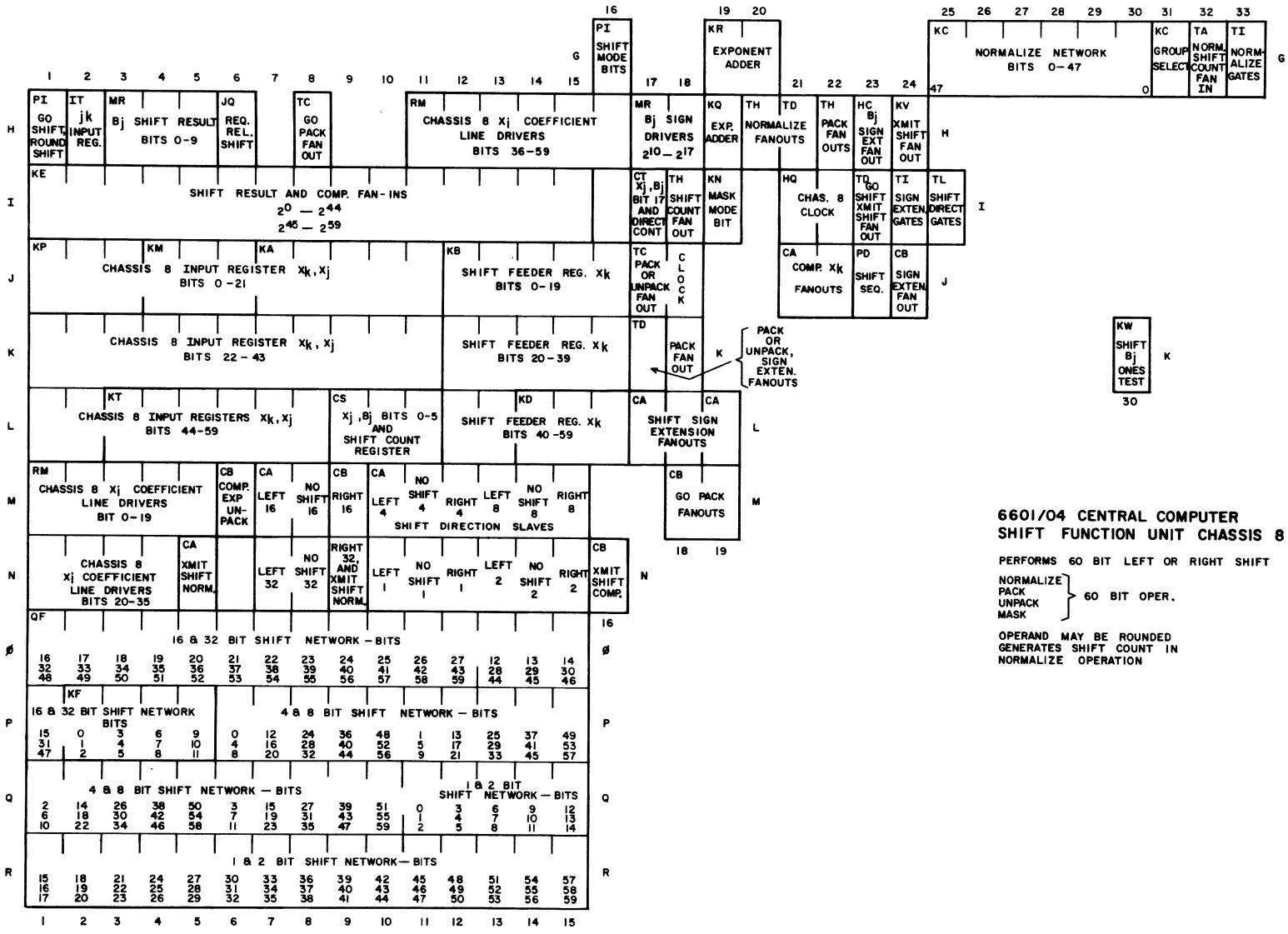
TITLE
CENTRAL PROCESSOR
SHIFT UNIT
BLOCK DIAGRAM

PRODUCT 6601
SIZE C | DRAWING NO. 60119300 | REV C
SHEET 106
21


OPERATING REGISTERS
CHASSIS 8
CHASSIS 8 INPUT REGISTERS
FEEDER REG.
X_k TRUNK
$X_k + X_i$
GO SHIFT
INSTRUCTION CONTROL
SHIFT jk + MASK
B_j
NORMALIZE
12 BITS
EXPONENT ADDER
NORMALIZE
UNPACK
B_j
PACK
NORM. SHIFT COUNT
CHASSIS OUTPUT NETWORK
NORMALIZE NETWORK
NORMALIZE
SHIFT NETWORK
SHIFT DIRECTION CONT.
SHIFT LEFT
$\overline{\text{SHIFT LEFT}}$
LONG ADD
B ADD
X_i
TRANSMIT SHIFT
B_j ONES TEST
$\overline{\text{LEFT SHIFT}}$
SHIFT NOMINAL
SHIFT COUNT REGISTER
SHIFT NOMINAL
MODE BITS
CONTROL
SHIFT jk
LEFT SHIFT
SHIFT NOMINAL
MASK
NORMALIZE
ROUND
PACK
UNPACK
$\overline{\text{NORMALIZE}}$
TIMING CHAIN
NORMALIZE
OR
REQUEST RELEASE
GO SHIFT
SCOREBOARD
CONTROL DATA CORPORATION
COMPUTER DIVISION
TITLE
CENTRAL PROCESSOR
SHIFT UNIT
BLOCK DIAGRAM
PRODUCT
6601/04
SIZE C
DRAWING NO. 60119300
REV K
SHEET 107
23



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR SHIFT UNIT BLOCK DIAGRAM DATA FLOW	PRODUCT 6601/04	
		SIZE C	DRAWING NO. 60119300 REV K
			SHEET 108 25

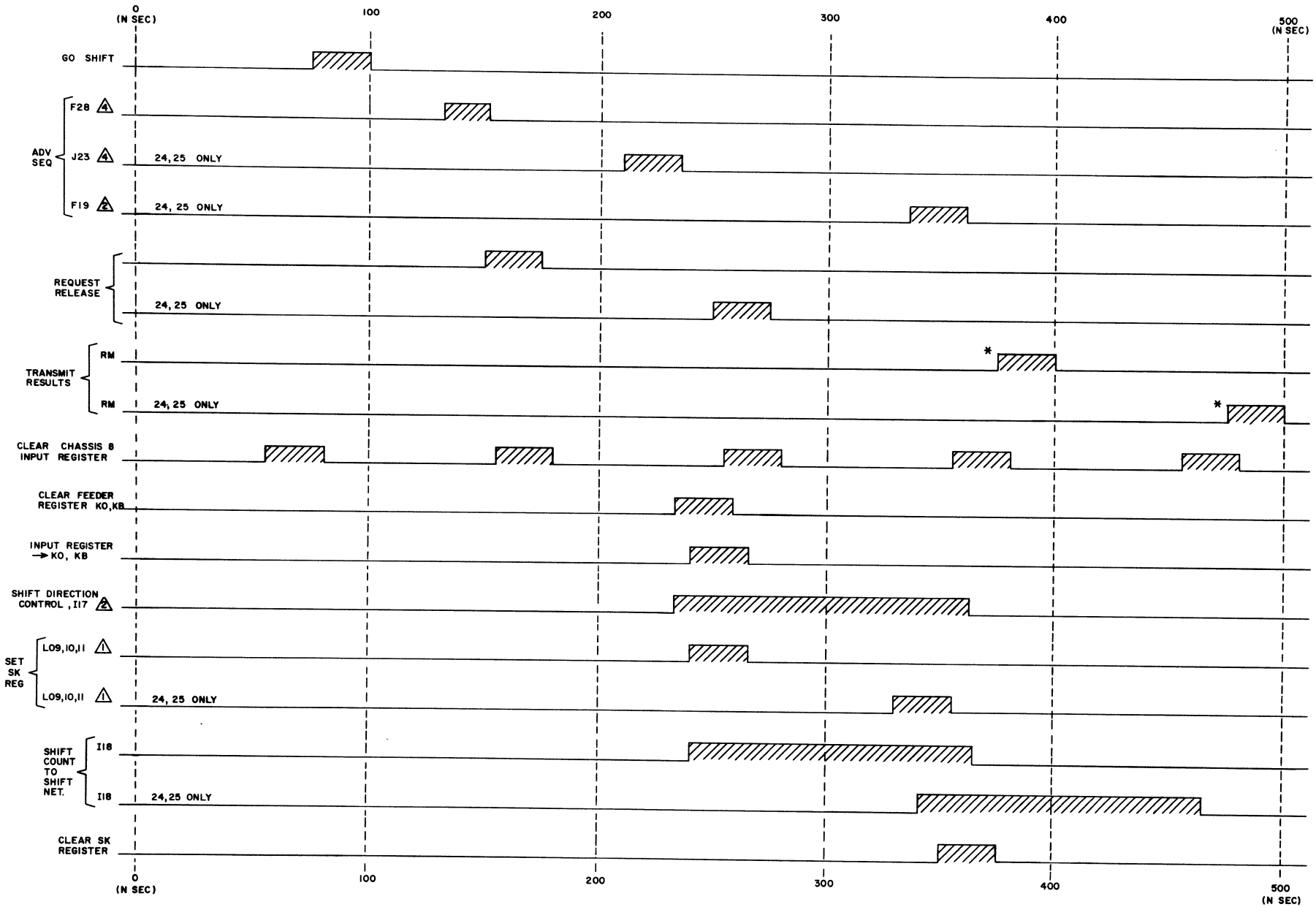


6601/04 CENTRAL COMPUTER SHIFT FUNCTION UNIT CHASSIS 8

PERFORMS 60 BIT LEFT OR RIGHT SHIFT

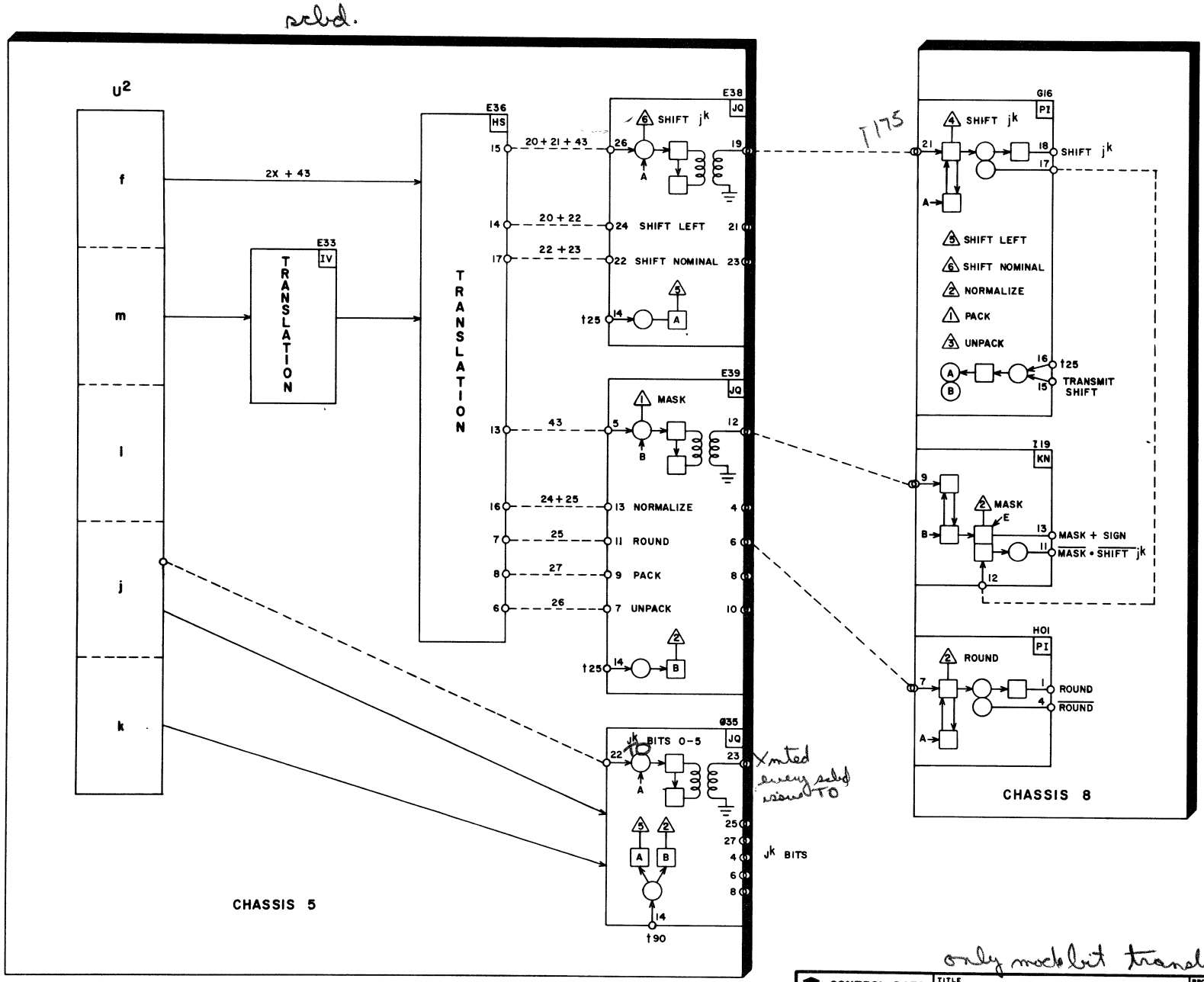
NORMALIZE
PACK
UNPACK
MASK
} 60 BIT OPER.

OPERAND MAY BE ROUNDED
GENERATES SHIFT COUNT IN NORMALIZE OPERATION

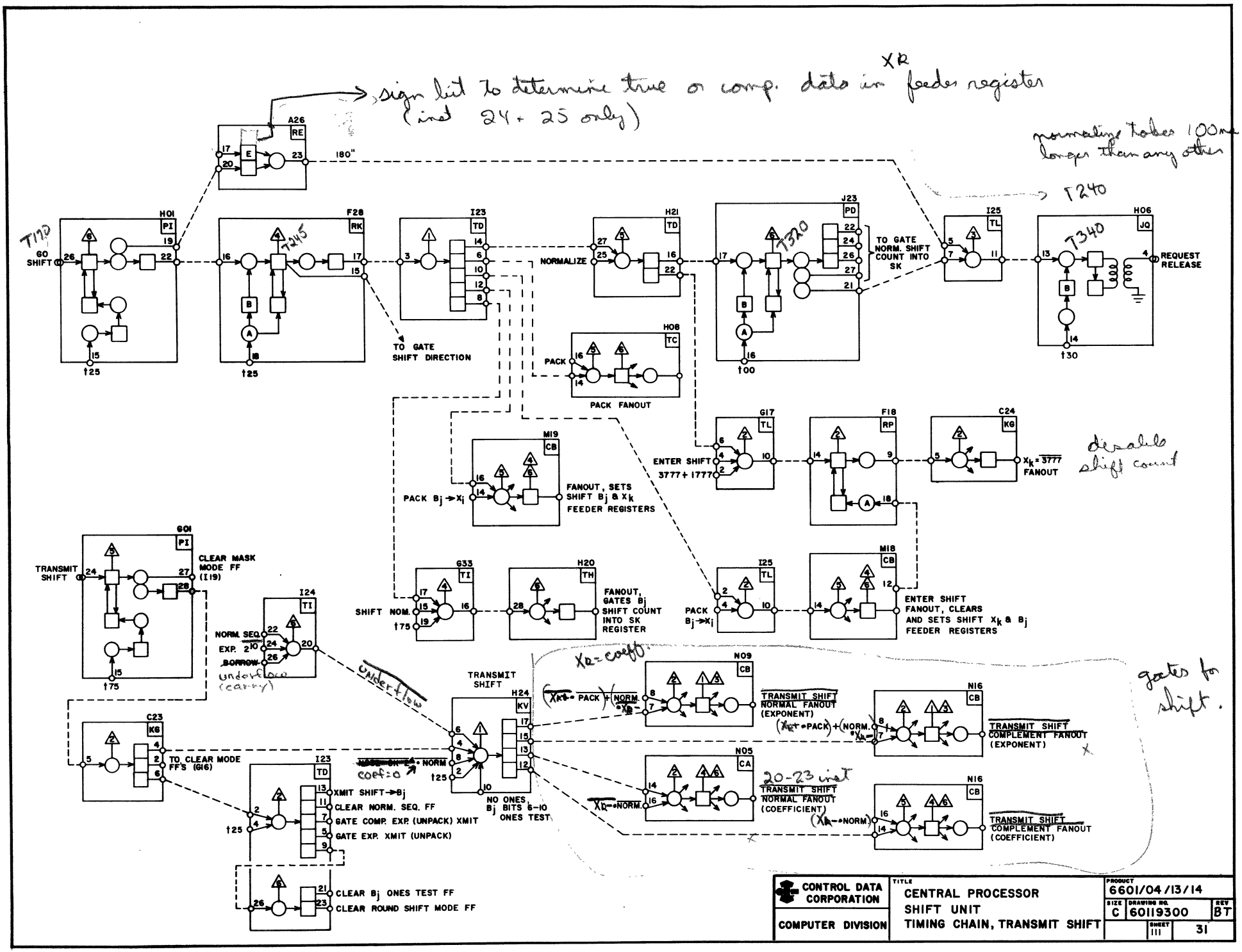


* NOTE
EARLIEST POSSIBLE TIME-
NO RESULT REGISTER CONFLICT

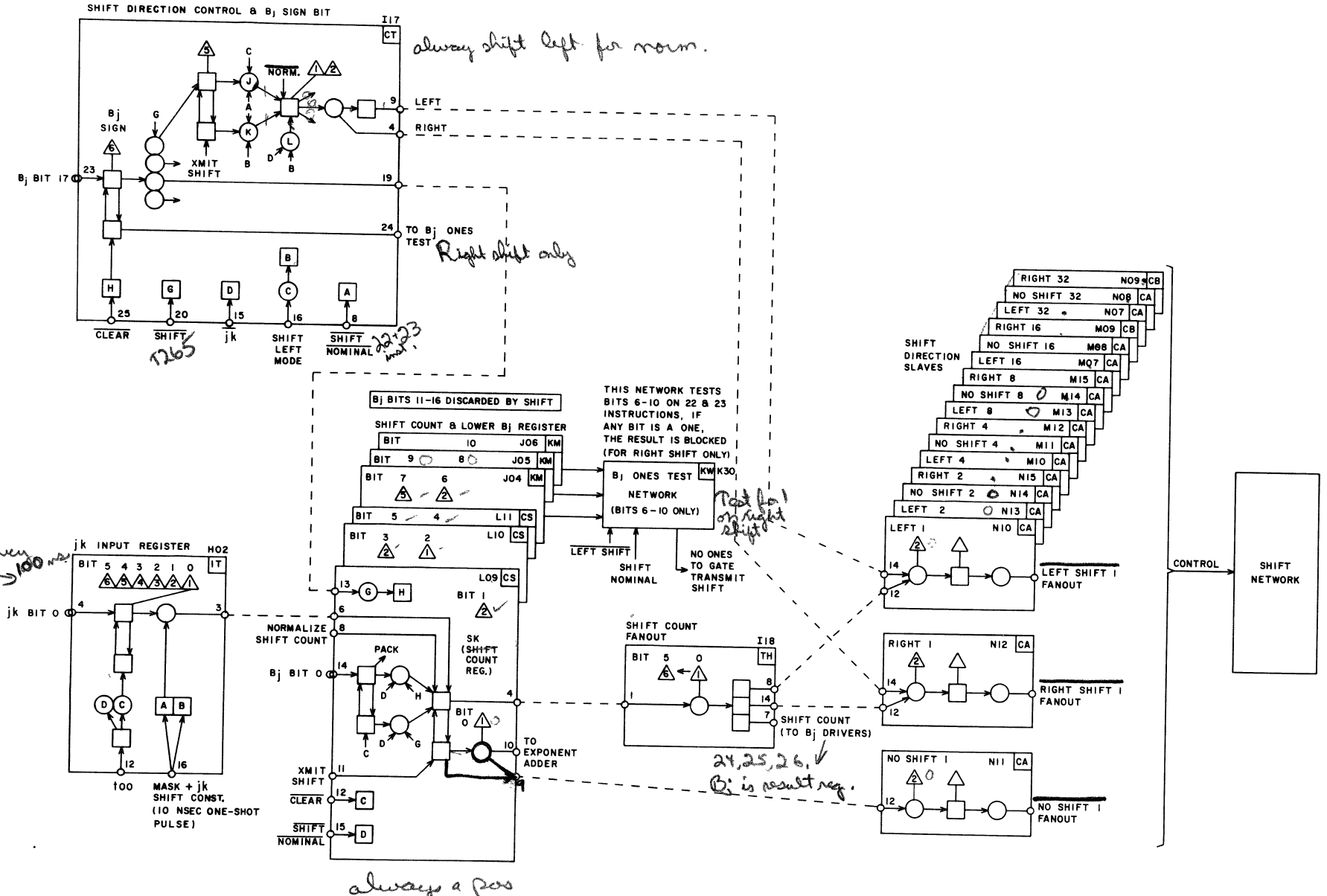
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR SHIFT UNIT TIMING CHART	PRODUCT: 6601	
COMPUTER DIVISION		SIZE: C · DRAWING NO.: 60119300	REV: C
		SHEET: 109	27



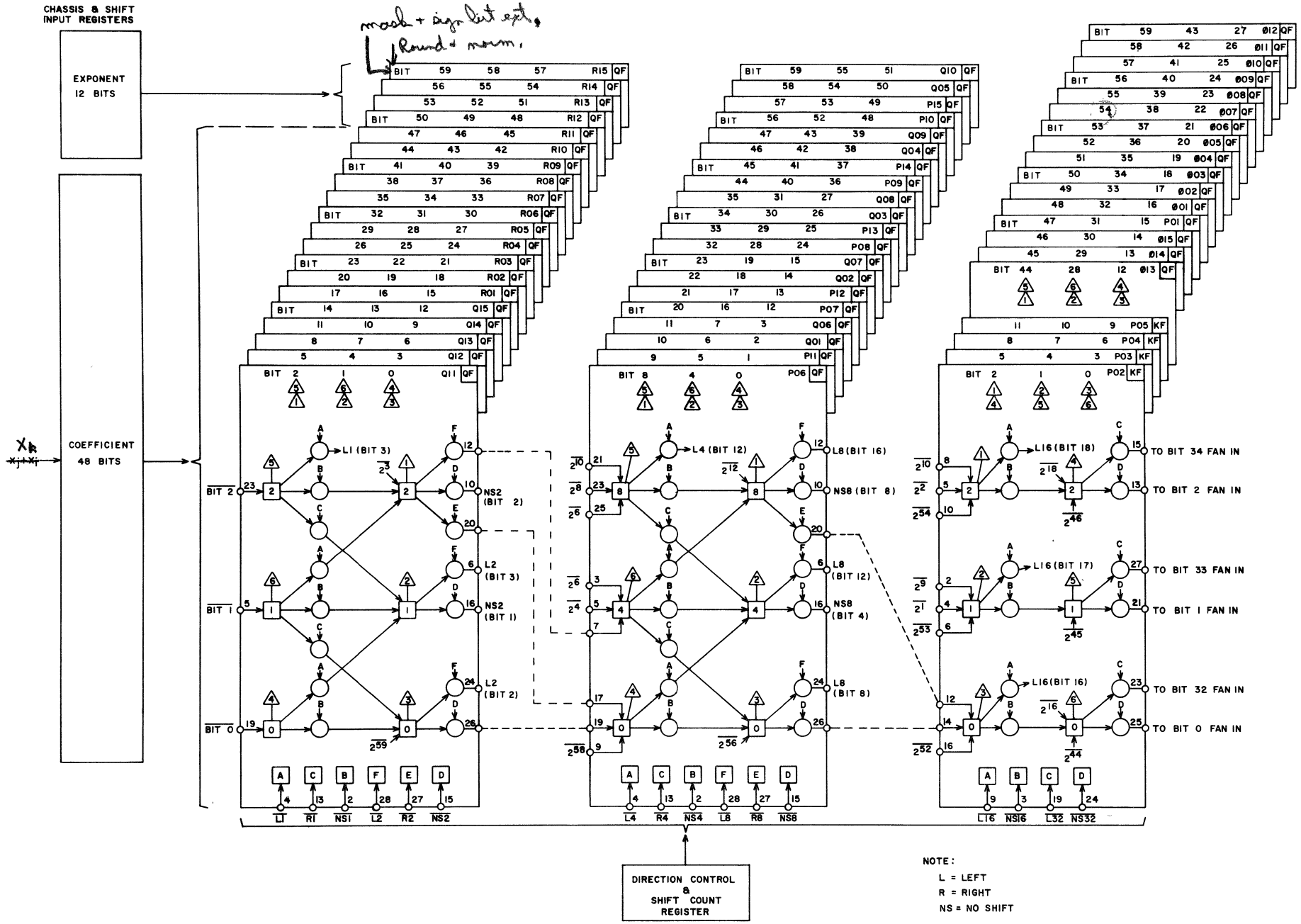
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR SHIFT UNIT MODE BITS, jk CONSTANT	PRODUCT: 6601/04	
COMPUTER DIVISION		SIZE: C DRAWING NO.: 60119300	REV: K
		SHEET: 110	29



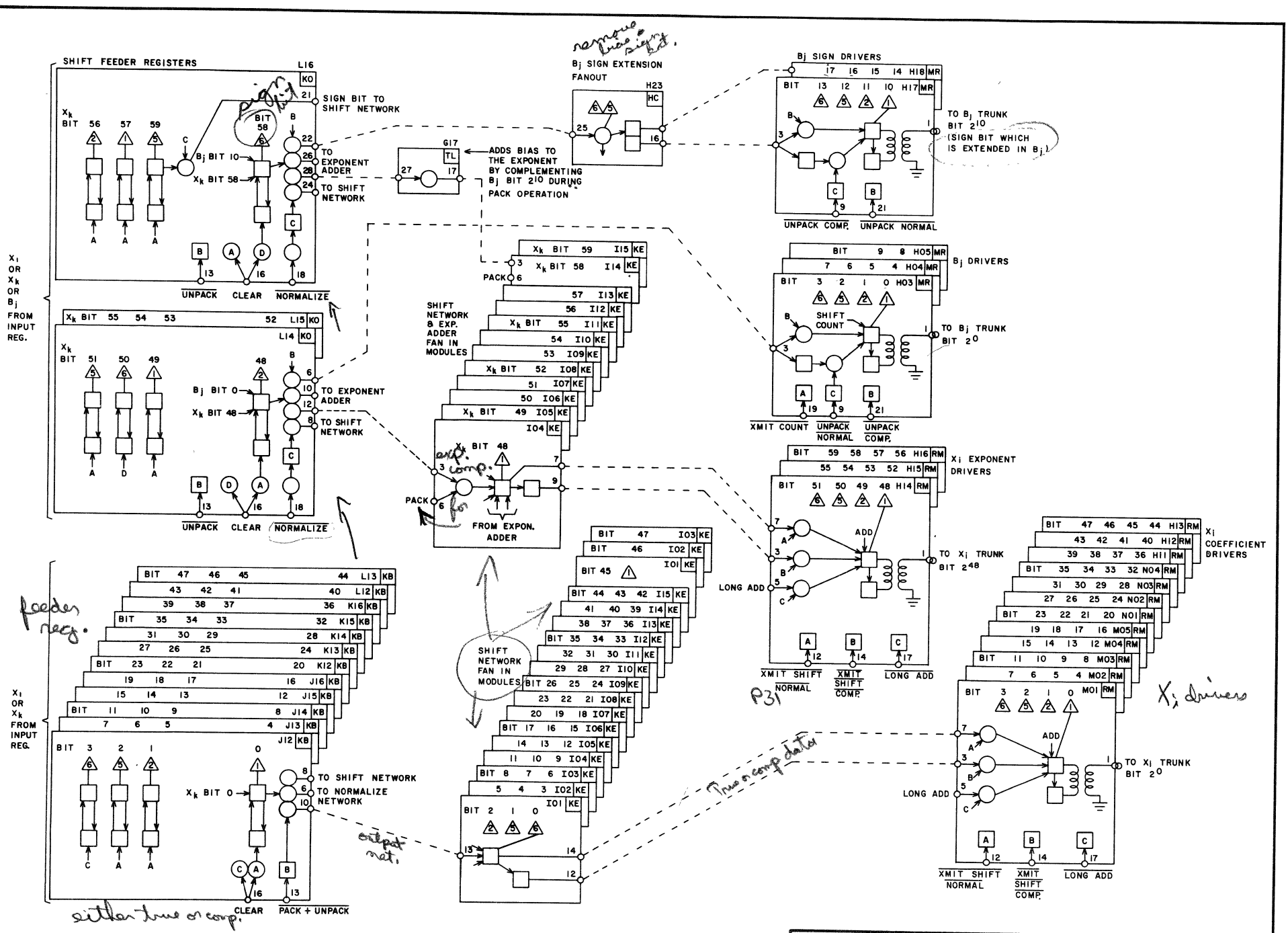
CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR SHIFT UNIT TIMING CHAIN, TRANSMIT SHIFT	PRODUCT 6601/04/13/14	
		SIZE C	DRAWING NO. 60119300 REV BT
			SHEET 111 31



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR SHIFT UNIT Bj, jk & SK REGISTERS DIRECT. CONTROL, Bj ONES TEST	PRODUCT 6601/04 SIZE C DRAWING NO. 60119300 REV K SHEET 112 33	



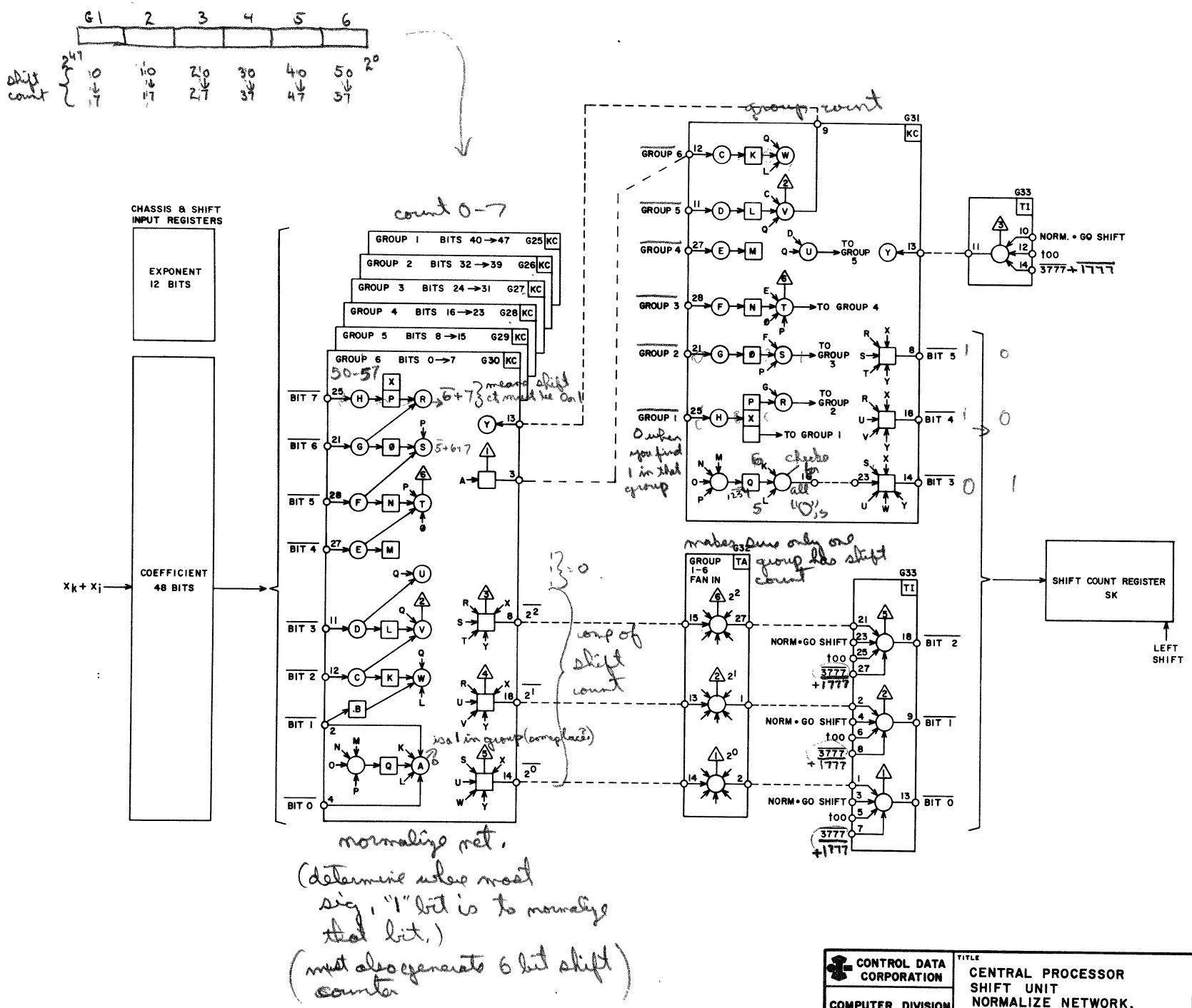
CONTROL DATA CORPORATION	TITLE	PRODUCT	
COMPUTER DIVISION	CENTRAL PROCESSOR SHIFT UNIT SHIFTING NETWORK	6601/04	
		SIZE C · DRAWING NO. 60119300 · REV K	
		SHEET 113	35

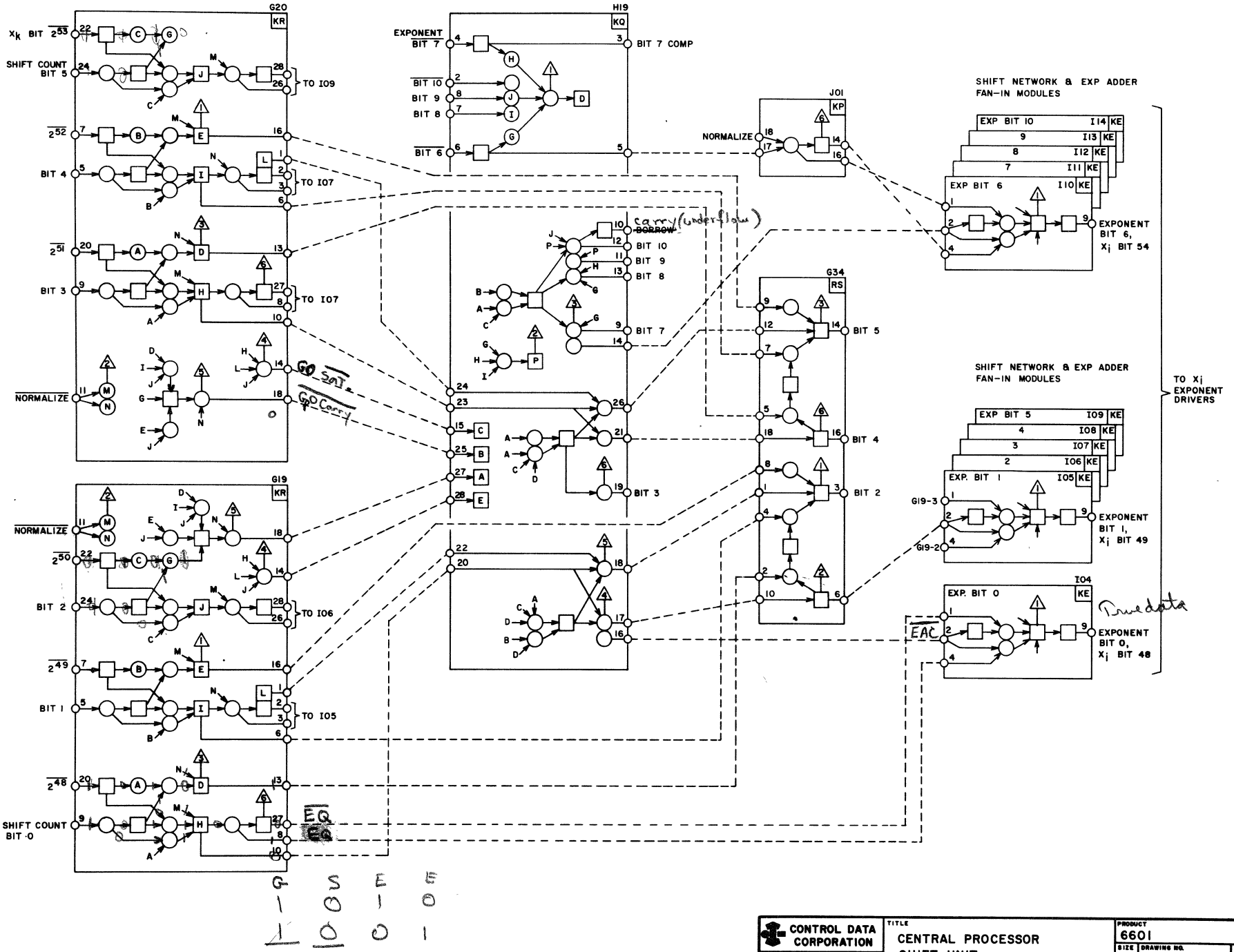


NOTE:

REFER TO LONG ADD DATA PATH DWG., SHEET NO. 129 FOR CHASSIS 8 INPUT REGISTER.

CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR SHIFT UNIT FEEDER REGS., Bj & Xi DRIVERS PACK & UNPACK	PRODUCT 6601/04 SIZE C · DRAWING NO. 60119300 · REV K SHEET 114 · 37	


G1 2 3 4 5 6
shift count
CHASSIS & SHIFT INPUT REGISTERS
EXPONENT 12 BITS
COEFFICIENT 48 BITS
count 0-7
group count
GROUP 1 BITS 40→47 G25 KC
GROUP 2 BITS 32→39 G26 KC
GROUP 3 BITS 24→31 G27 KC
GROUP 4 BITS 16→23 G28 KC
GROUP 5 BITS 8→15 G29 KC
GROUP 6 BITS 0→7 G30 KC
G31 KC
G32 TA
G33 TI
BIT 7
BIT 6
BIT 5
BIT 4
BIT 3
BIT 2
BIT 1
BIT 0
TO GROUP 5
TO GROUP 4
TO GROUP 3
TO GROUP 2
TO GROUP 1
NORM. • GO SHIFT
too
3777+1777
GROUP 1-6 FAN IN
SHIFT COUNT REGISTER SK
LEFT SHIFT
normalize net.
(determine where most sig. "1" bit is to normalize that bit.)
(must also generate 6 bit shift counter

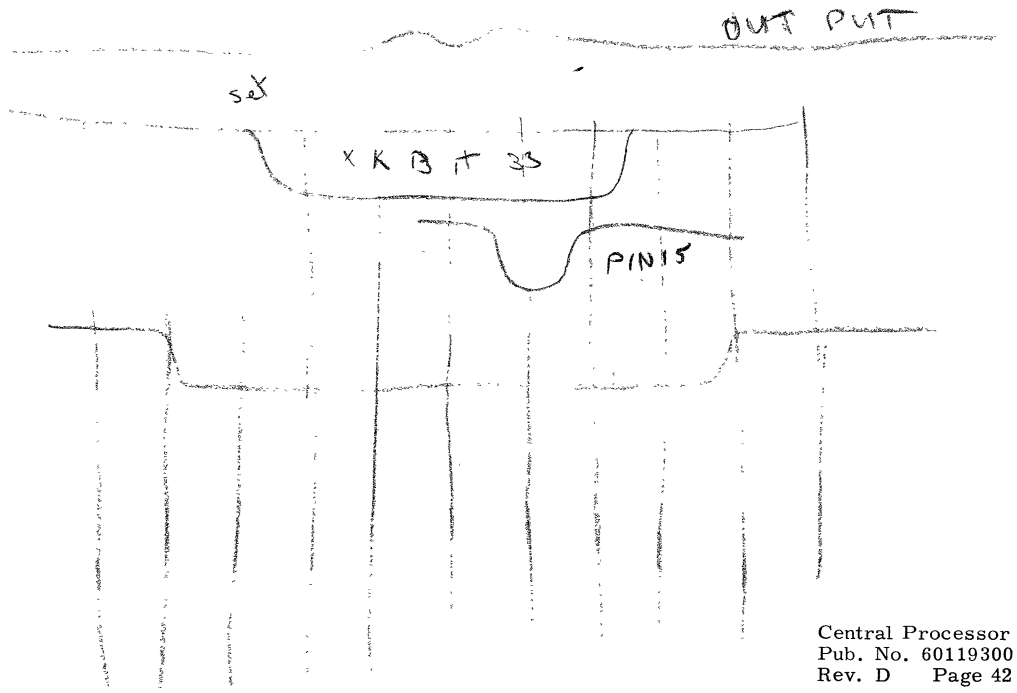


CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR SHIFT UNIT EXPONENT ADDER	PRODUCT 6601	
		SIZE C	DRAWING NO. 60119300 REV C
			SHEET 125 41

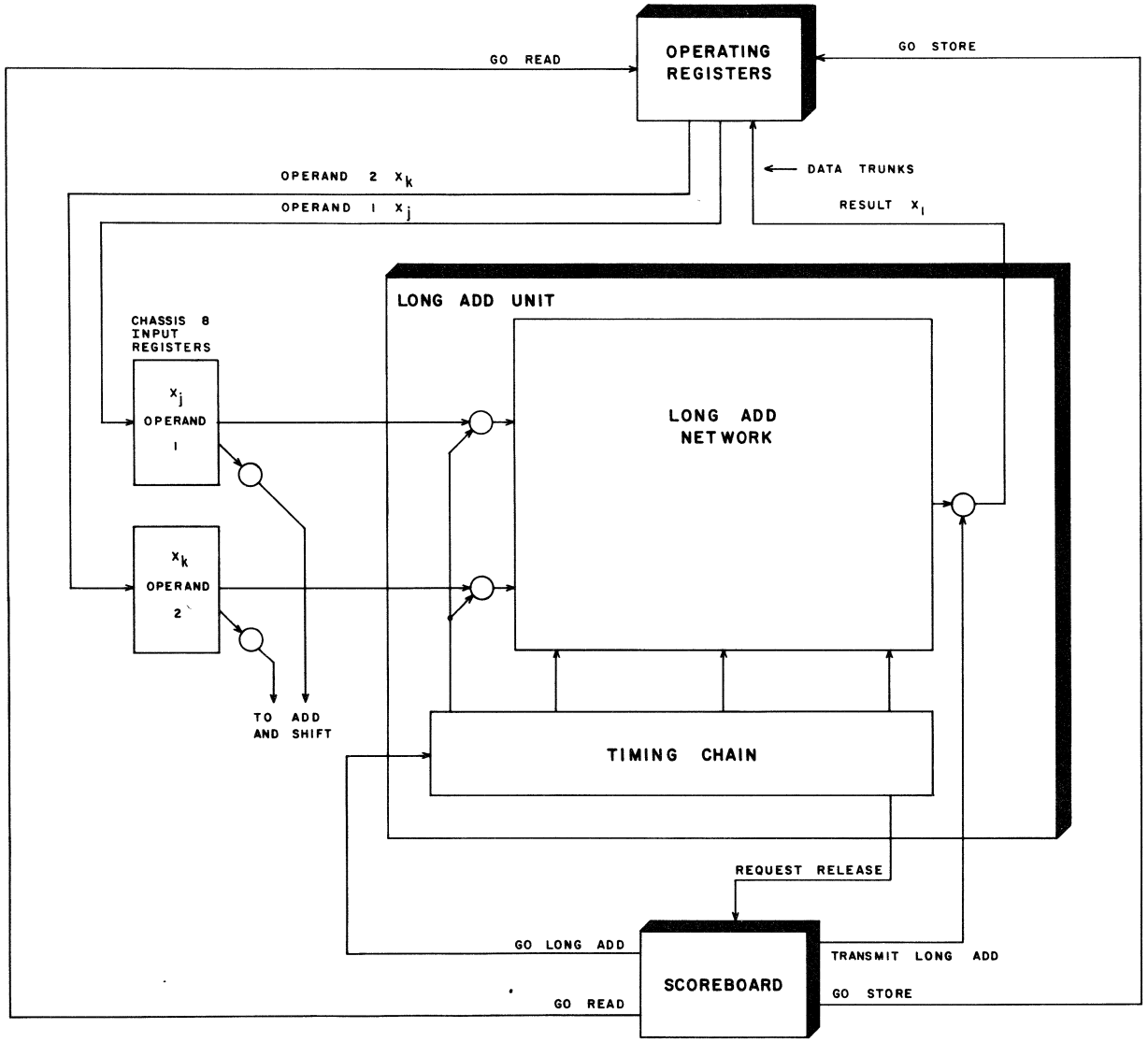
LONG ADD

Fixed point addition and subtraction in the CONTROL DATA 6601 and 6604 Central Computers are performed by the Long Add Unit on chassis 8. This unit contains all necessary registers and logic elements to form the 60-bit, fixed point sum (or difference) of two 60-bit, fixed point operands. The time required for an operation is 0.3 microseconds (3 minor cycles).

A Long Add instruction directs the unit either to add the addend X_k to the augend X_j and send the sum to X_i, or subtract the subtrahend X_k from the minuend X_j and send the difference to X_i.

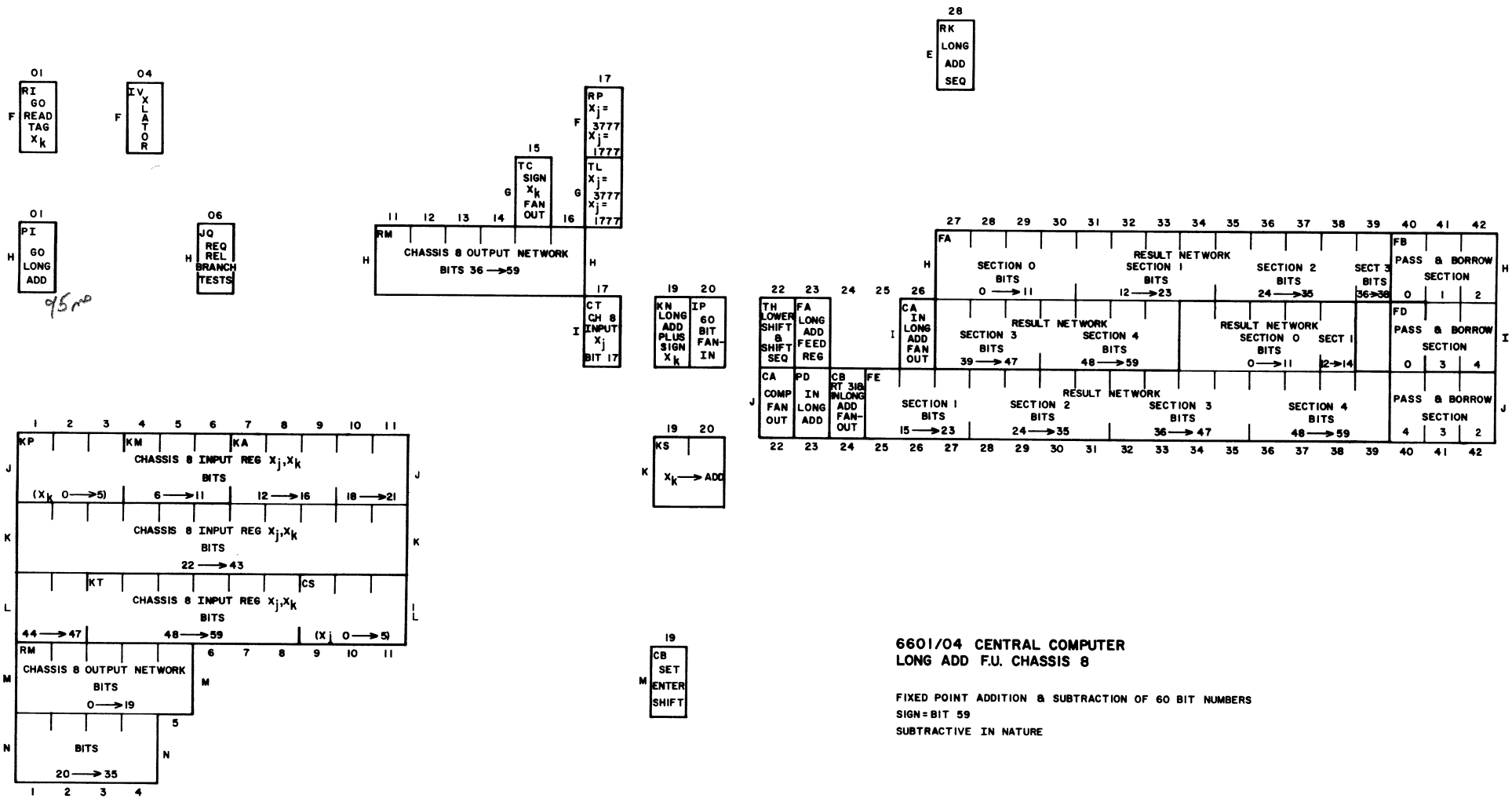


Integer add, Integer subtract
+ check conditions for jump



Chas. 8

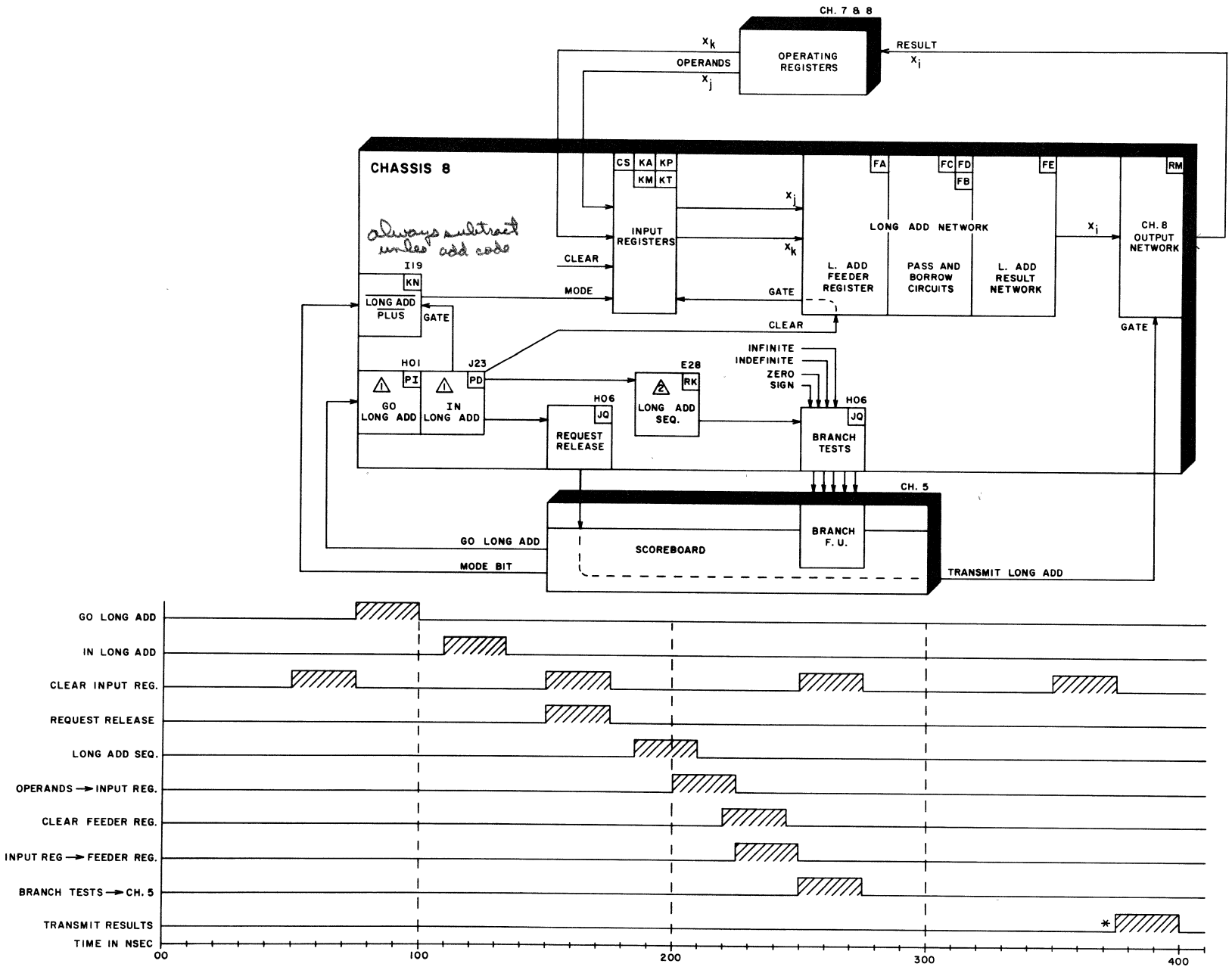
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR LONG ADD UNIT BLOCK DIAGRAM	PRODUCT: 6601/04	
COMPUTER DIVISION		SIZE: C · DRAWING NO.: 60119300 · REV: K	
		SHEET: 126	43



6601/04 CENTRAL COMPUTER
LONG ADD F.U. CHASSIS 8

FIXED POINT ADDITION & SUBTRACTION OF 60 BIT NUMBERS
SIGN = BIT 59
SUBTRACTIVE IN NATURE

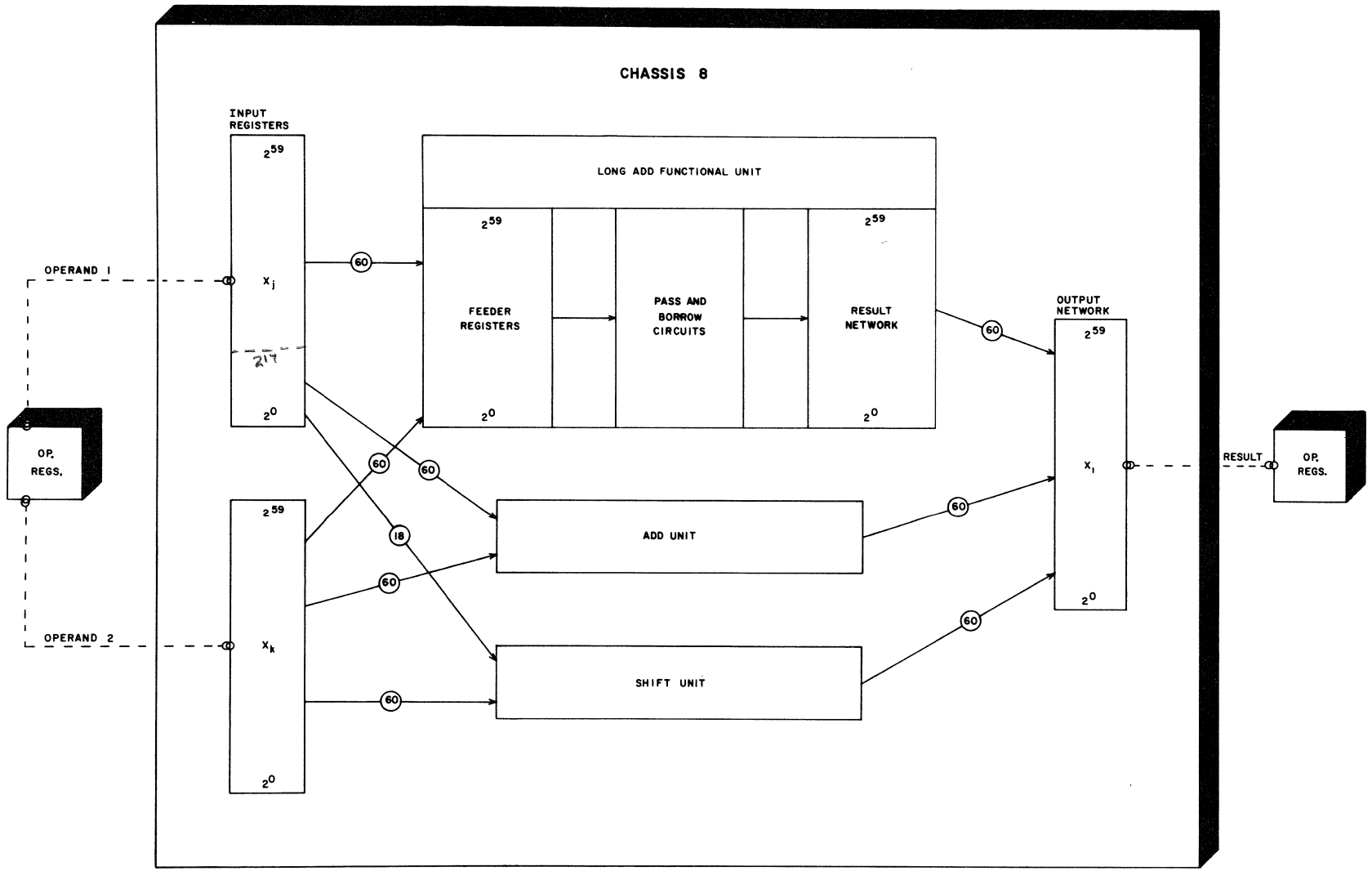
PUB. NO. 60119300
REV. K 44



CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE
CENTRAL PROCESSOR
LONG ADD UNIT
TIMING CHART

PRODUCT 6601
SIZE C | DRAWING NO. 60119300 | REV D
SHEET 127
45



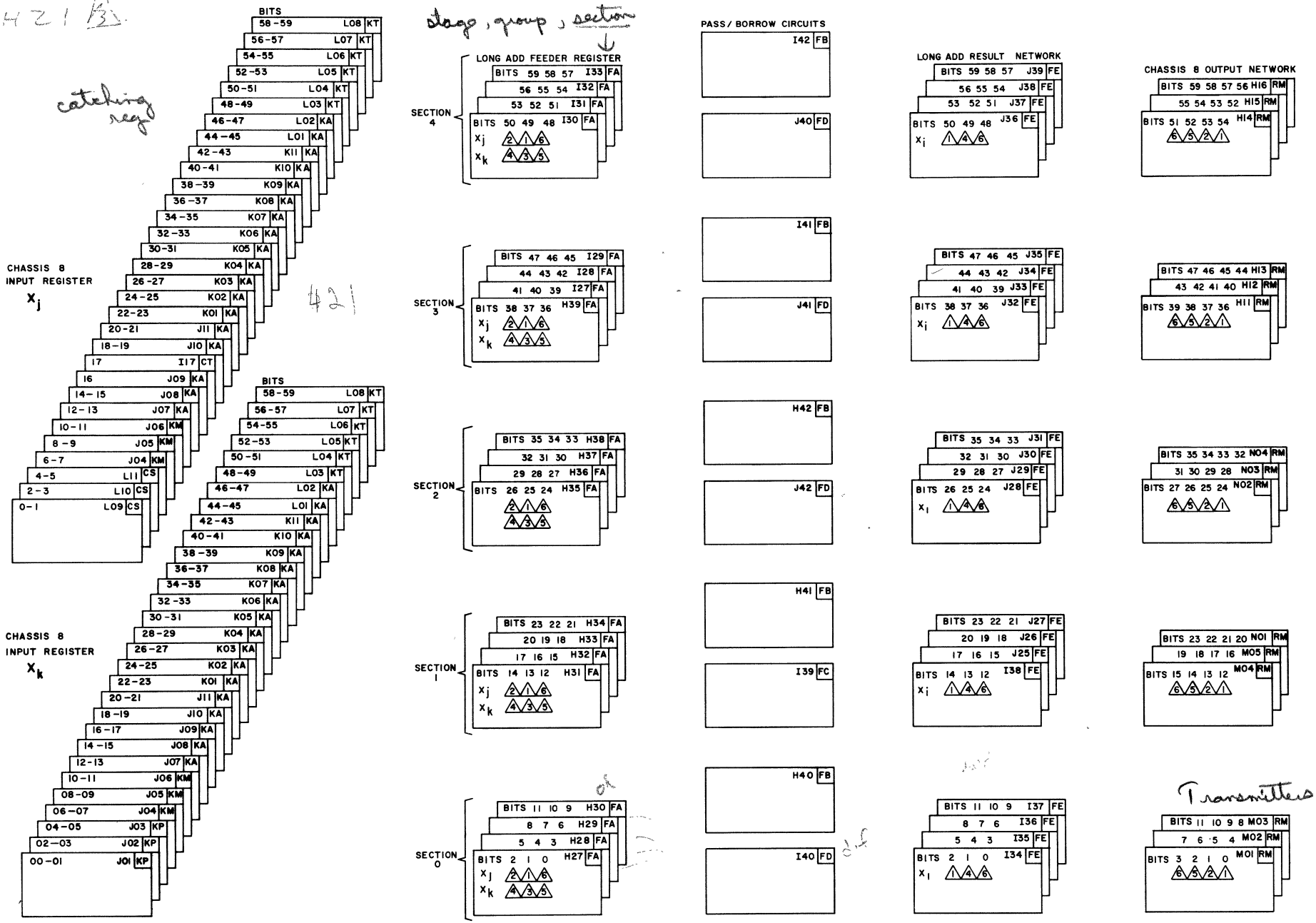
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR LONG ADD UNIT BLOCK DIAGRAM DATA FLOW	PRODUCT 6601/04	
COMPUTER DIVISION		SIZE C · DRAWING NO. 60119300 · REV K	
		SHEET 128	47

ADDER

The Long Add Unit performs fixed point addition and subtraction of 60-bit numbers. Negative numbers are represented in one's complement notation, and overflows are ignored. The sign bit is in the high order bit position (bit 59), and the binary point is at the right of the lowest order bit position (bit 0).

The adder is divided into three levels of operation, with three bits making up a group, four groups making up a section, and five sections making up the entire adder.

Even though the adder is subtractive in nature, the Long Add Plus mode bit gates the true value of X_k from the input register to the Feeder register. The one's complement value of X_k is gated to the Feeder register for Subtract instructions.



catching reg

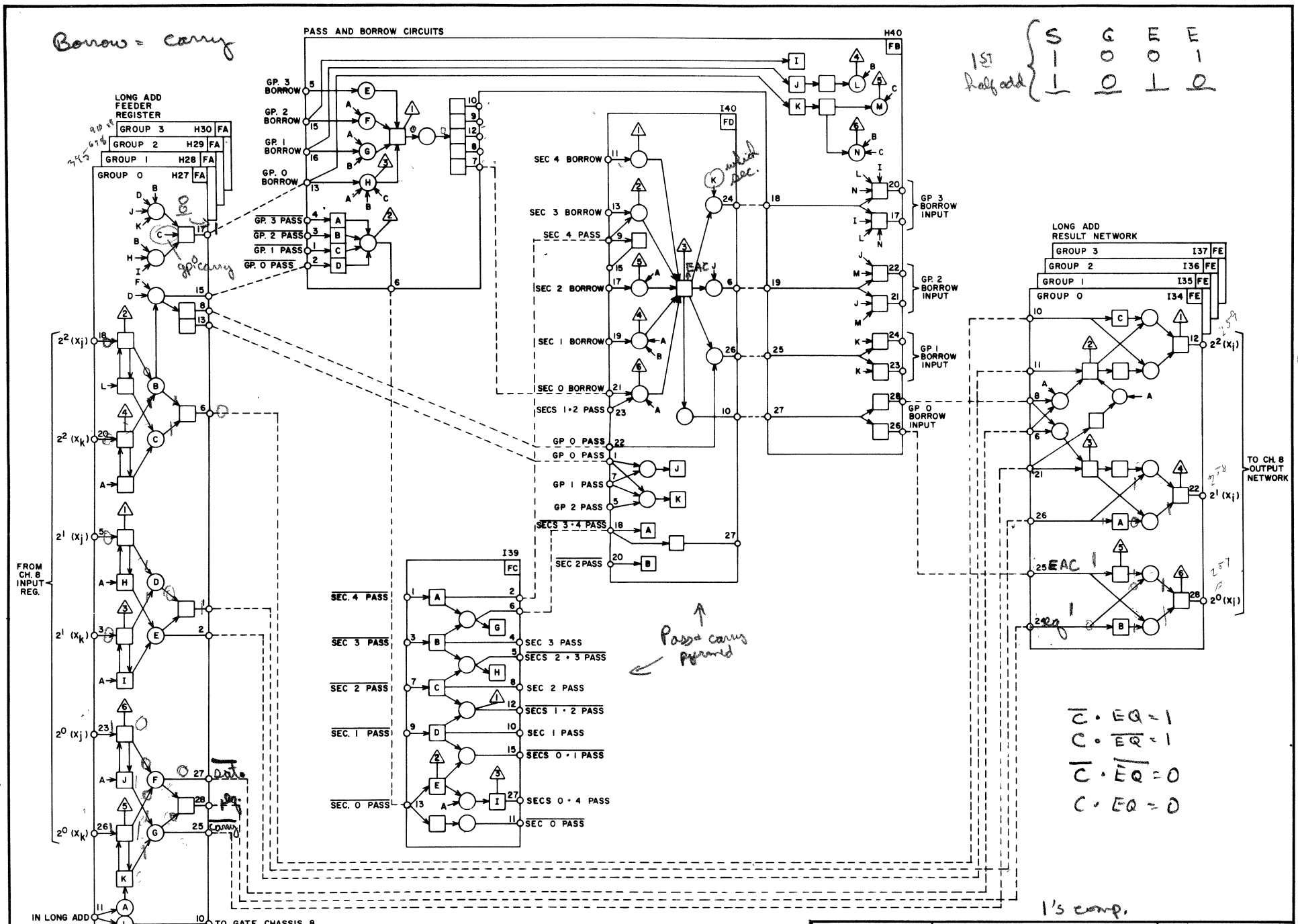
421

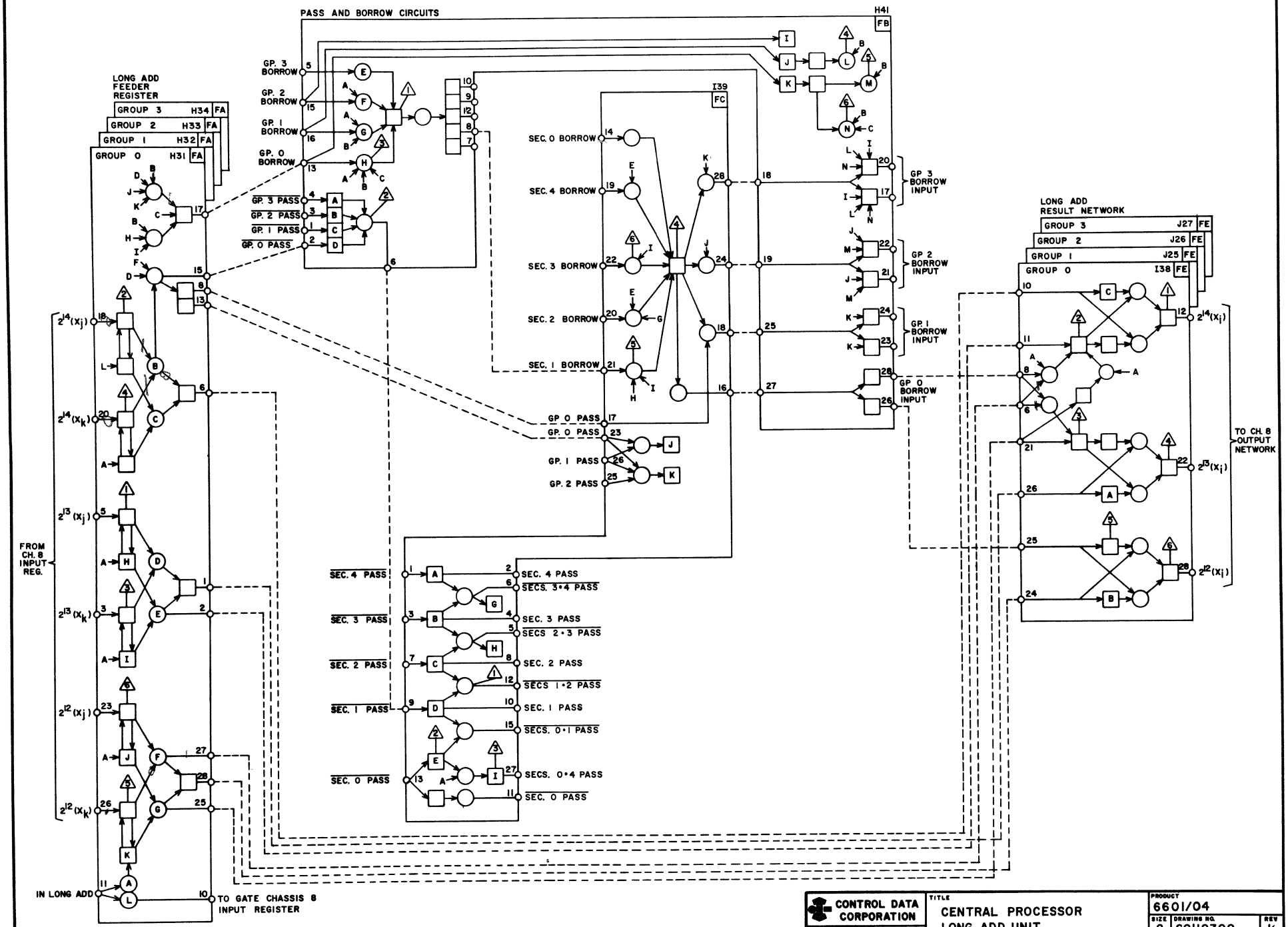
dogg, group, section

Transmitters

4 group/sec.
3 bits/group

CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR LONG ADD UNIT DATA PATH (AB catch. reg.)	PRODUCT: 6601/04	
COMPUTER DIVISION		SIZE: C · DRAWING NO.: 60119300 · REV: K	
		SHEET: 129	49

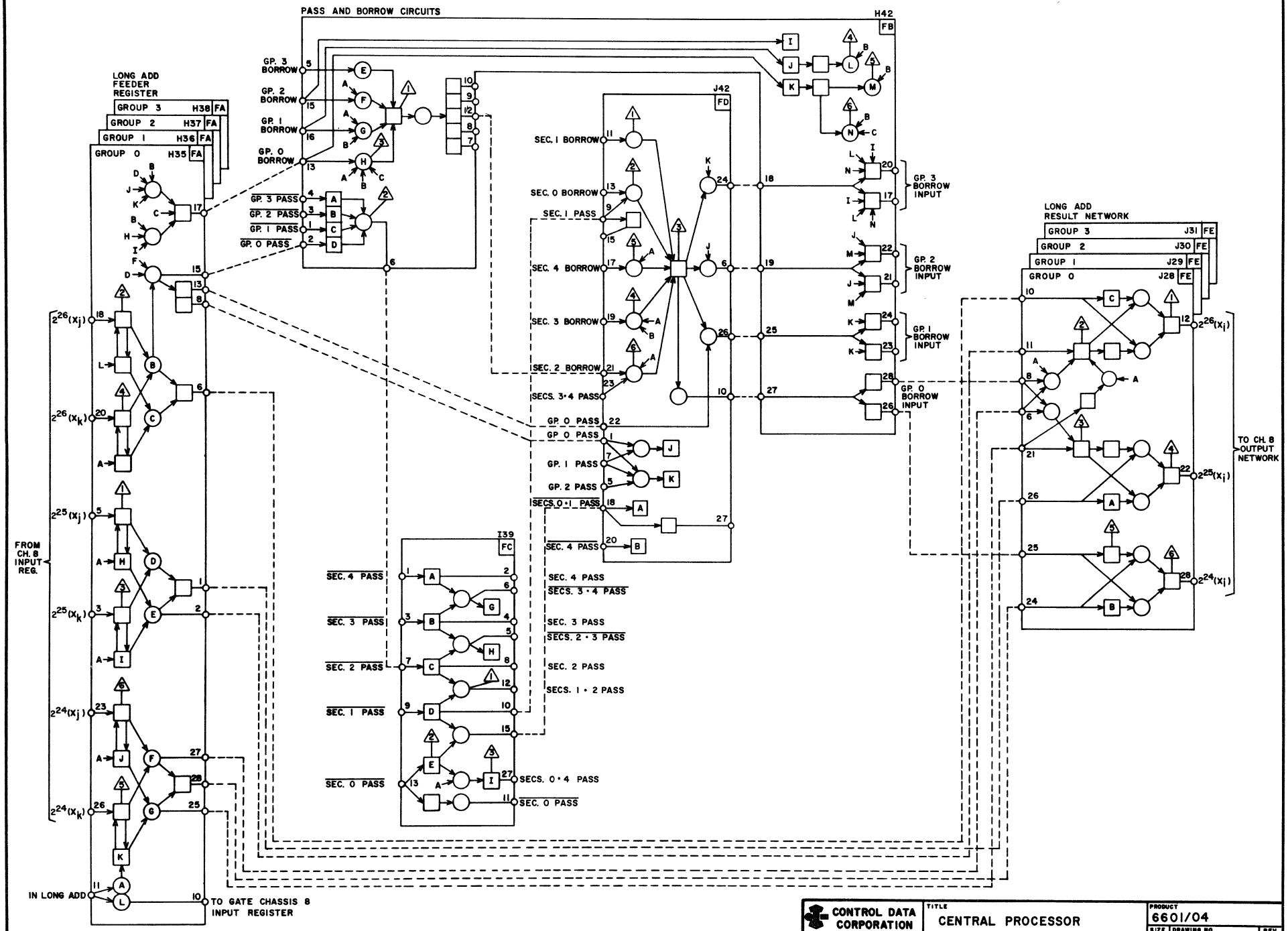

Borrow = carry
PASS AND BORROW CIRCUITS
H40
FB
LONG ADD FEEDER REGISTER
GROUP 3 H30 FA
GROUP 2 H29 FA
GROUP 1 H28 FA
GROUP 0 H27 FA
GP. 3 BORROW
GP. 2 BORROW
GP. 1 BORROW
GP. 0 BORROW
GP. 3 PASS
GP. 2 PASS
GP. 1 PASS
GP. 0 PASS
I40
FD
SEC 4 BORROW
SEC 3 BORROW
SEC 4 PASS
SEC 2 BORROW
SEC 1 BORROW
SEC 0 BORROW
SECS 1•2 PASS
GP 0 PASS
GP 1 PASS
GP 2 PASS
SECS 3•4 PASS
SEC 2 PASS
GP. 3 BORROW INPUT
GP. 2 BORROW INPUT
GP. 1 BORROW INPUT
GP. 0 BORROW INPUT
LONG ADD RESULT NETWORK
GROUP 3 I37 FE
GROUP 2 I36 FE
GROUP 1 I35 FE
GROUP 0 I34 FE
TO CH. 8 OUTPUT NETWORK
FROM CH. 8 INPUT REG.
I39
FC
SEC. 4 PASS
SEC 3 PASS
SEC 2 PASS
SEC. 1 PASS
SEC. 0 PASS
SEC 3 PASS
SECS 2 • 3 PASS
SEC 2 PASS
SECS 1 • 2 PASS
SEC 1 PASS
SECS 0 • 1 PASS
SECS 0 • 4 PASS
SEC 0 PASS
IN LONG ADD
TO GATE CHASSIS 8 INPUT REGISTER
1ST half add
S G E F
1 0 0 1
1 0 1 0
no result network (input held in reg. if no result always avail)
C • EQ = 1
C • EQ = 1
C • EQ = 0
C • EQ = 0
1's comp.
CONTROL DATA CORPORATION
COMPUTER DIVISION
TITLE
CENTRAL PROCESSOR
LONG ADD UNIT
ADDER, SECTION 0
PRODUCT 6601/04
SIZE C
DRAWING NO. 60119300
REV K
SHEET 130
51



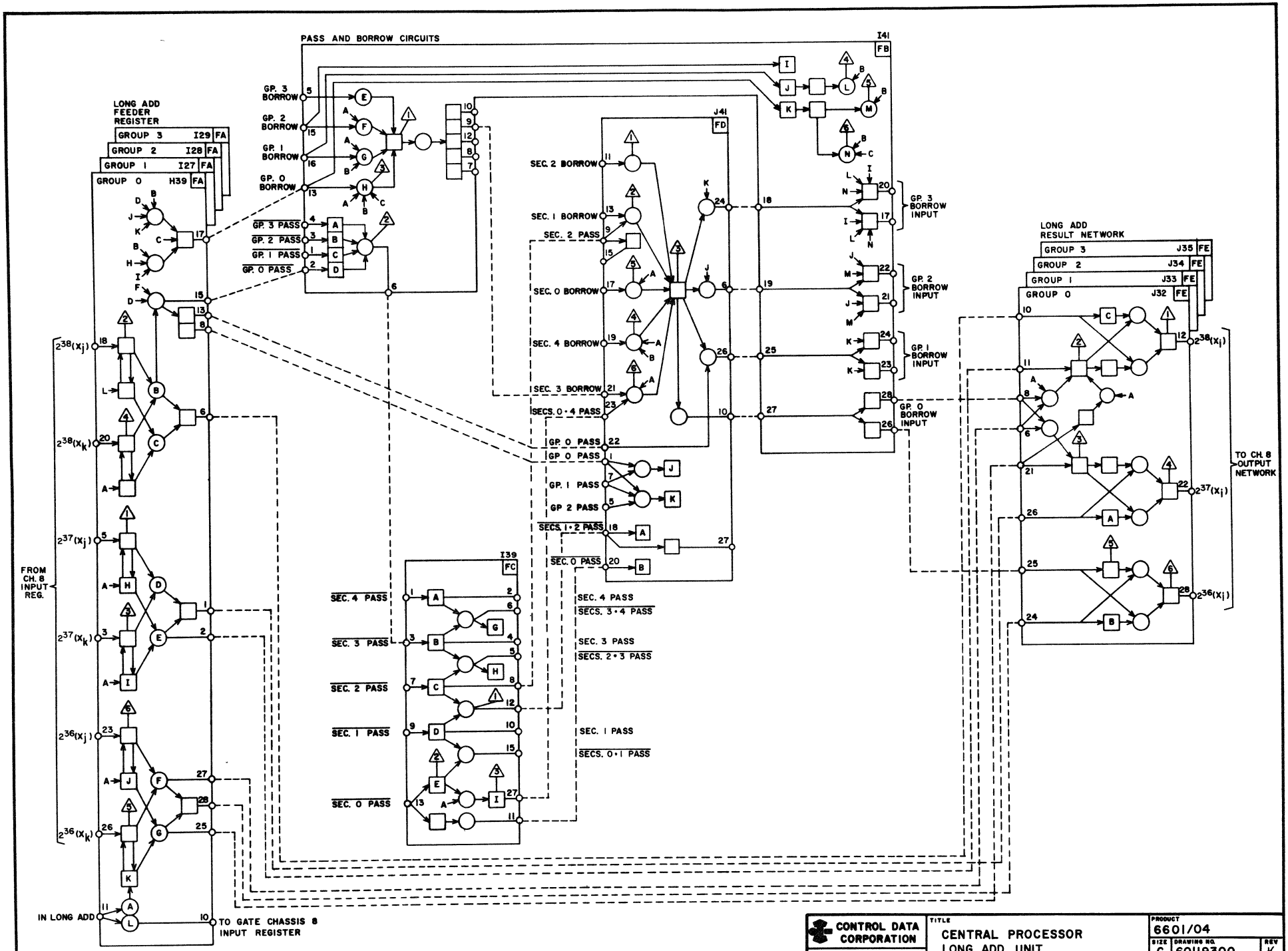
CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE
CENTRAL PROCESSOR
LONG ADD UNIT
ADDER, SECTION 1

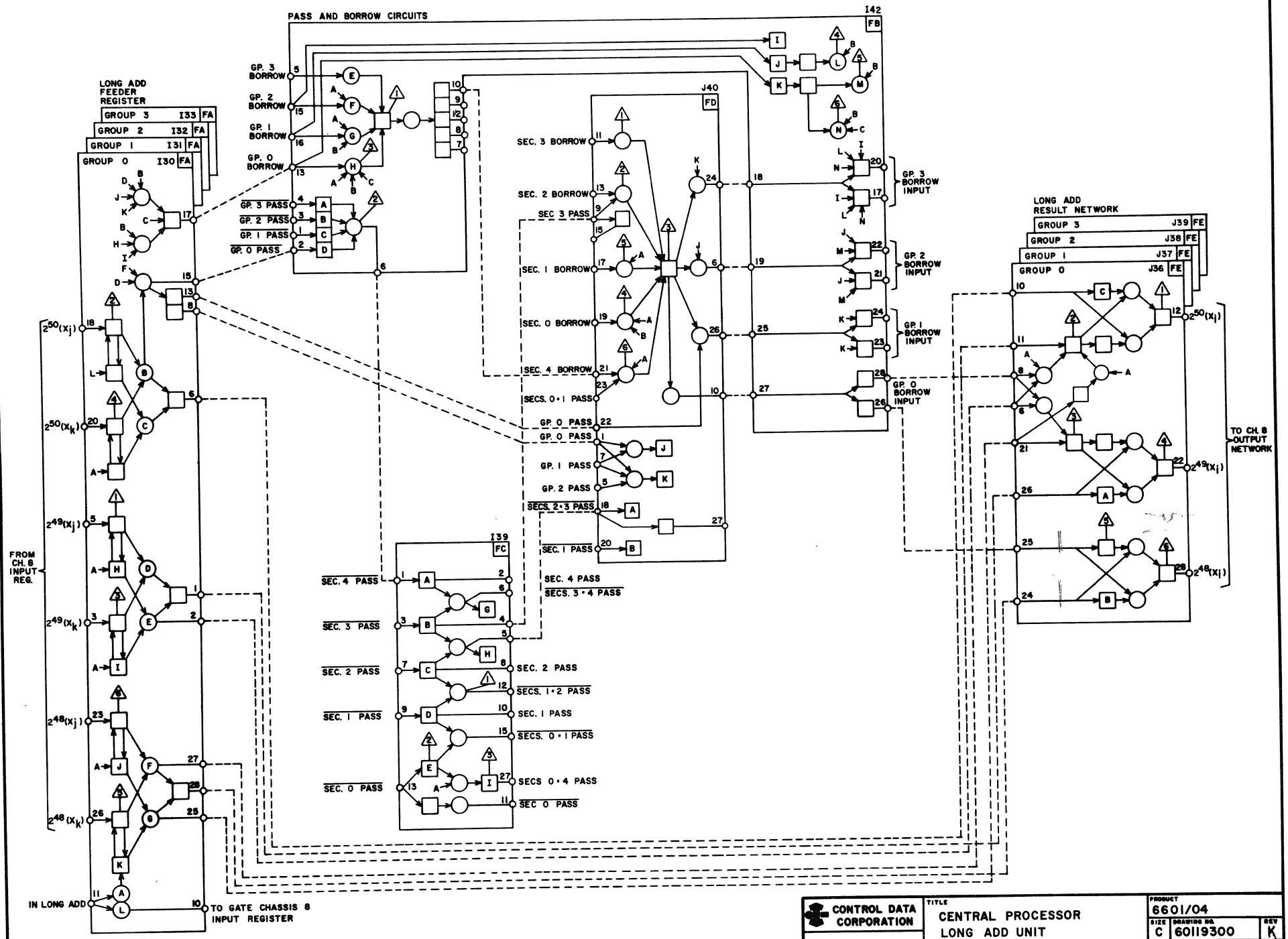
PRODUCT 6601/04
SIZE C | DRAWING NO. 60119300 | REV K
SHEET 131 | 53



CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR LONG ADD UNIT ADDER, SECTION 2	PRODUCT: 6601/04	
COMPUTER DIVISION		SIZE: C · DRAWING NO.: 60119300 · REV: K	
		SHEET: 132	55



CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR LONG ADD UNIT ADDER, SECTION 3	PRODUCT: 6601/04	
COMPUTER DIVISION		SIZE: C · DRAWING NO.: 60119300 · REV: K	
		SHEET: 133	57



CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR LONG ADD UNIT ADDER, SECTION 4	PRODUCT: 6601/04	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: K
		SHEET: 134	59

BRANCH TESTS

Besides being used as a fixed point adder, the Long Add Unit also serves to test certain conditions in branch instructions. For all these tests, the contents of operand register X_j are gated to the Long Add Unit (bits 2^{36} - 2^{59} are already on chassis 8). Here they are examined, and the resulting control bits are sent to the Branch Unit on chassis 5.

Op. Code	Go to K if:	Test Performed
030	$X_j = 0$	Zero
031	$X_j = 0$	Zero
032	$X_j > 0$	Sign
033	$X_j < 0$	Sign
034	X_j is in range	Range or Infinite
035	X_j is out of range	Range or Infinite
036	X_j is definite	Indefinite
037	X_j is indefinite	Indefinite

Zero Test

1. Checks all 60-bits of X_j for 0, or all for 1.

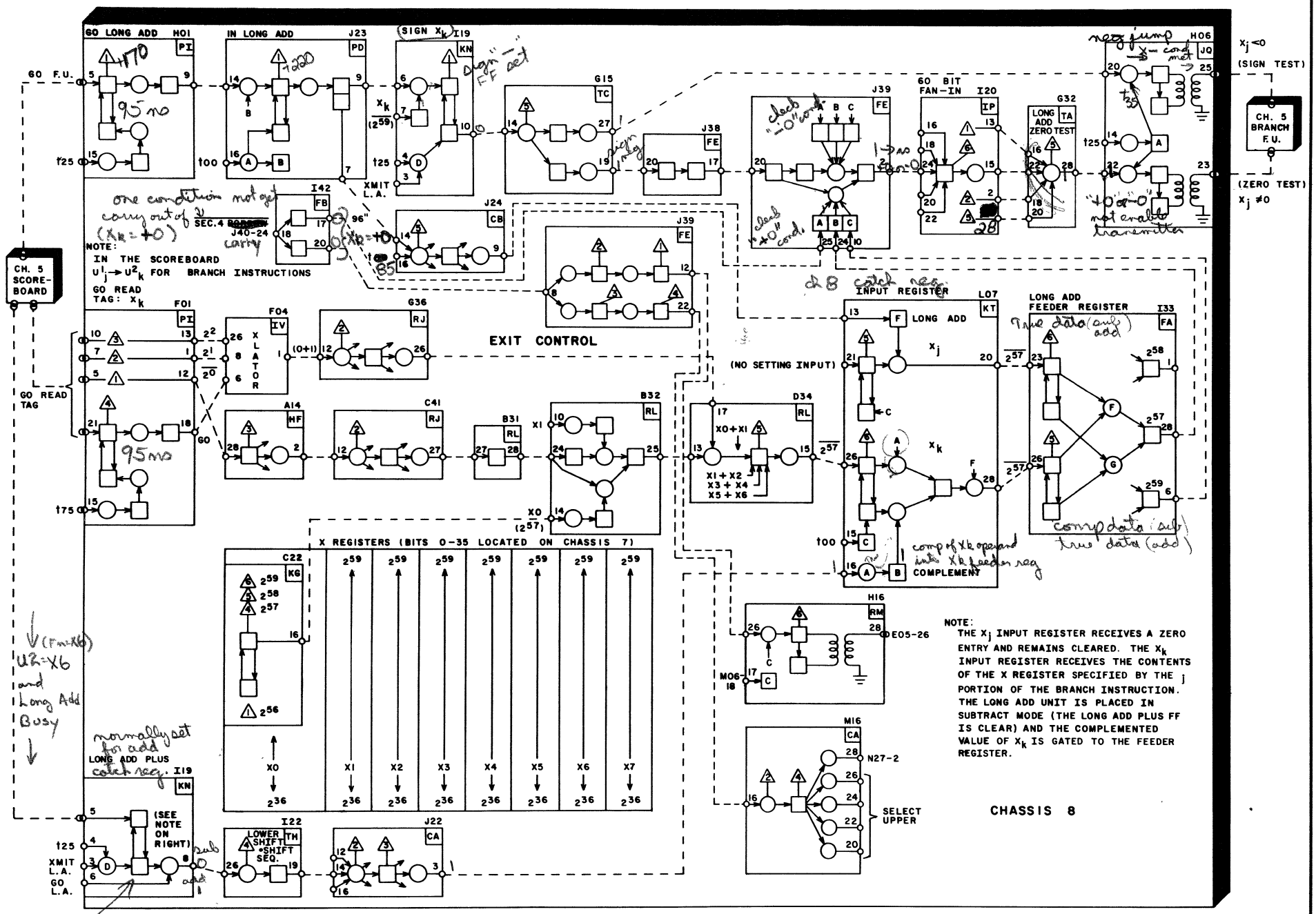
2. Valid for both fixed and floating point numbers.

   If underflow occurs in a Floating Point operation, the result is treated as absolute zero. A zero quantity is packed with a zero exponent and a zero coefficient.

Sign Test

1. Examines the sign bit (2^{59}) of X_j.

2. Valid for both fixed and floating point numbers.



NOTE:
THE X_j INPUT REGISTER RECEIVES A ZERO ENTRY AND REMAINS CLEARED. THE X_k INPUT REGISTER RECEIVES THE CONTENTS OF THE X REGISTER SPECIFIED BY THE j PORTION OF THE BRANCH INSTRUCTION. THE LONG ADD UNIT IS PLACED IN SUBTRACT MODE (THE LONG ADD PLUS FF IS CLEAR) AND THE COMPLEMENTED VALUE OF X_k IS GATED TO THE FEEDER REGISTER.

CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR LONG ADD UNIT SIGN / ZERO TESTS	PRODUCT 6601/04/13/14	
		SIZE C	DRAWING NO. 60119300 REV BT
		SHEET 136	61

Range Test

1. Checks bits 2^{48} - 2^{58} of X_j against 3777, or 4000

2. Valid for both fixed and floating point numbers.

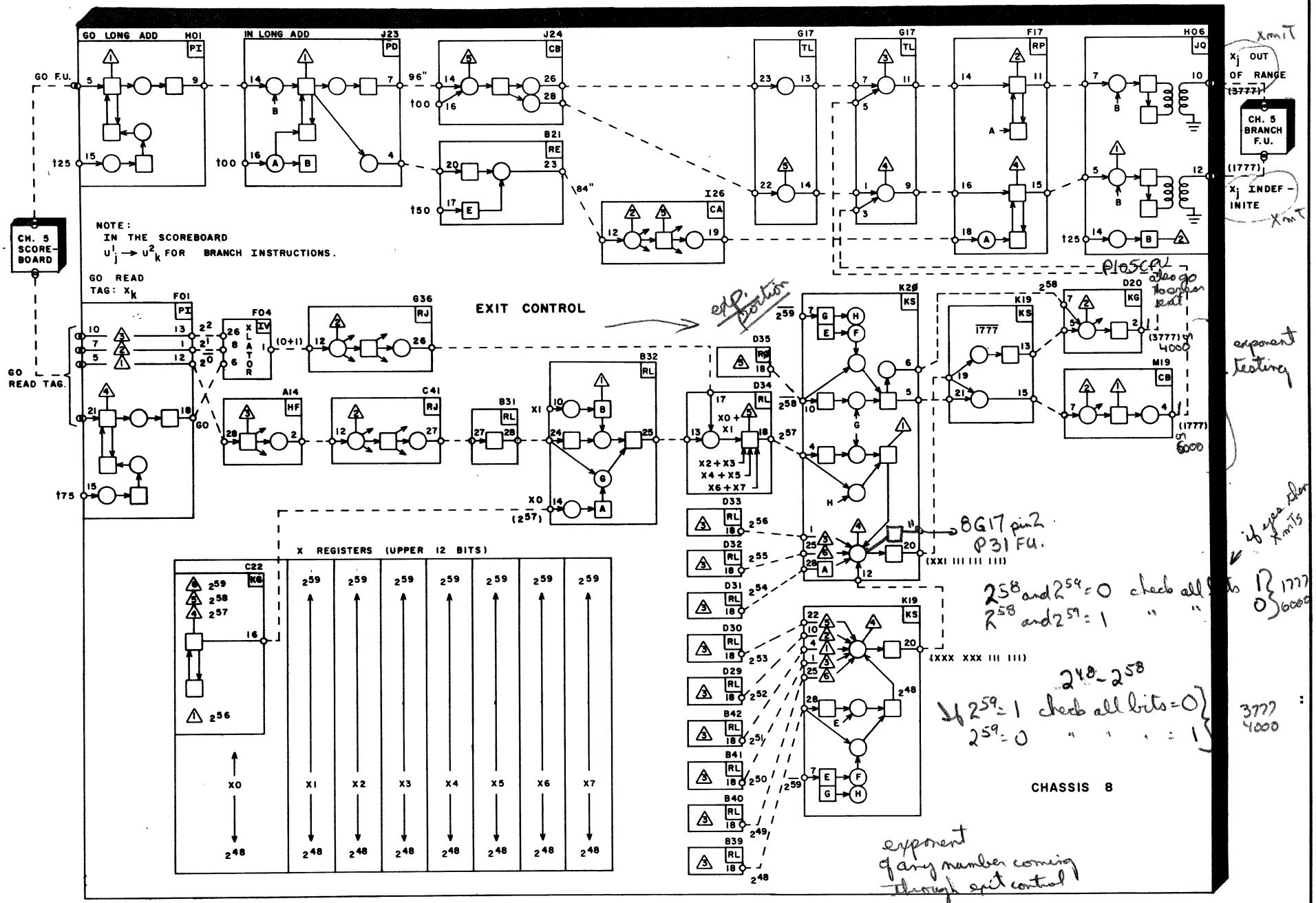
   If overflow occurs in a floating point operation, the result is treated as infinite. An infinite quantity is packed with an exponent of octal 3777 and a zero coefficient. Since the lower order bits are ignored for this test, near-overflow numbers (fixed and floating point) are also considered out of range if the upper 12-bits are 3777 (or 4000).

Indefinite Test

1. Checks bits 2^{48} - 2^{58} of X_j against 1777, or 6000

2. Valid for floating point numbers only.

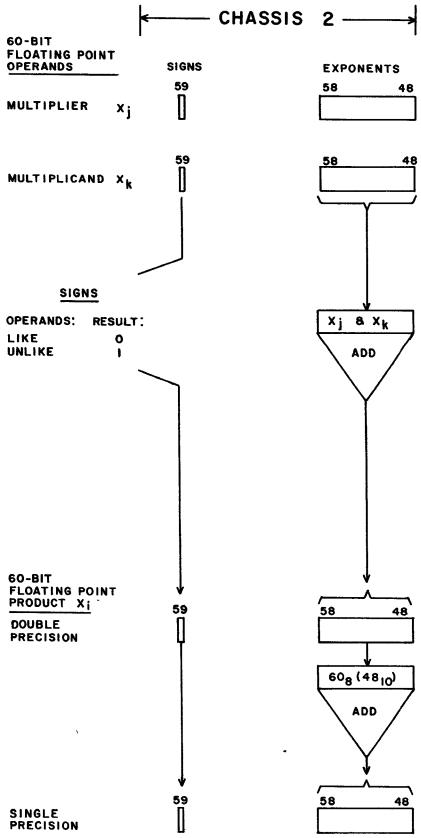
   The use of either infinity or zero as operands in a floating point operation may produce an indefinite result. An indefinite quantity is packed with an exponent of octal 1777 and a zero coefficient.

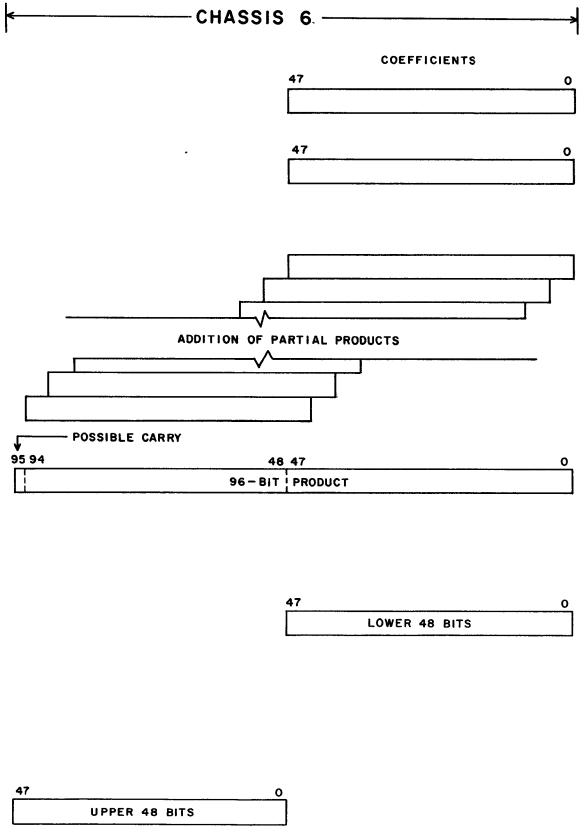


CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR LONG ADD UNIT RANGE / INDEFINITE TESTS	PRODUCT 6601/04/13/14 SIZE C · DRAWING NO. 60119300 · REV 8T SHEET 135	63

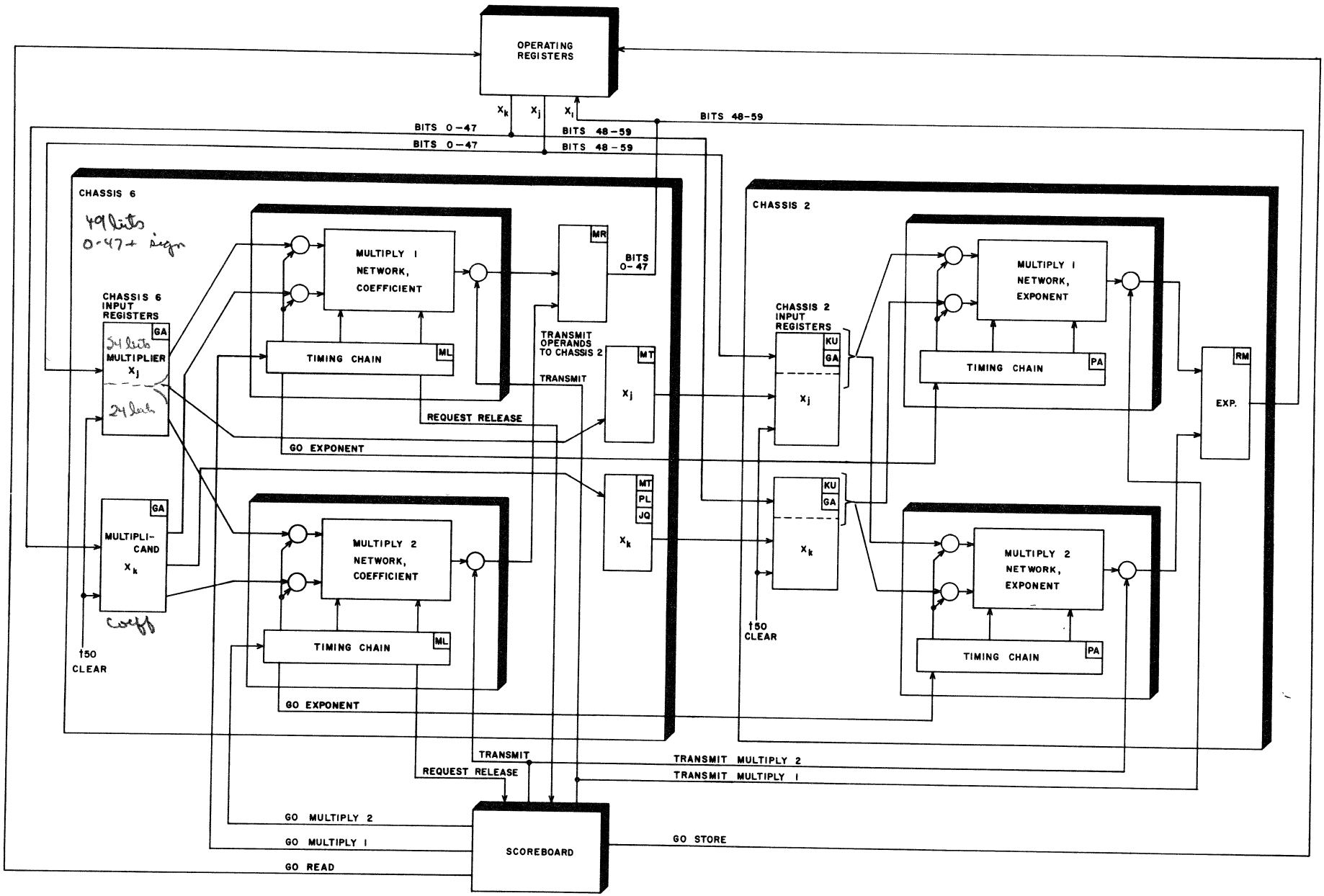
Multiplication in the CONTROL DATA 6601 Central Computer is performed by two identical multiply units. Each of these units contains all necessary registers, adders, and logic elements to form the 60-bit, floating-point product of two 60-bit, floating-point operands. A given multiply instruction may be performed by either unit if the other unit is busy. The time required for a multiplication is one microsecond (10 minor cycles).

A Multiply instruction directs the available unit to take the multiplier X_j times the multiplicand X_k and send the product to X_i. The coefficient portions of operands X_j and X_k are sent to chassis 6 where they are multiplied by a method using addition of partial products. The exponents are sent to chassis 2 to be added.

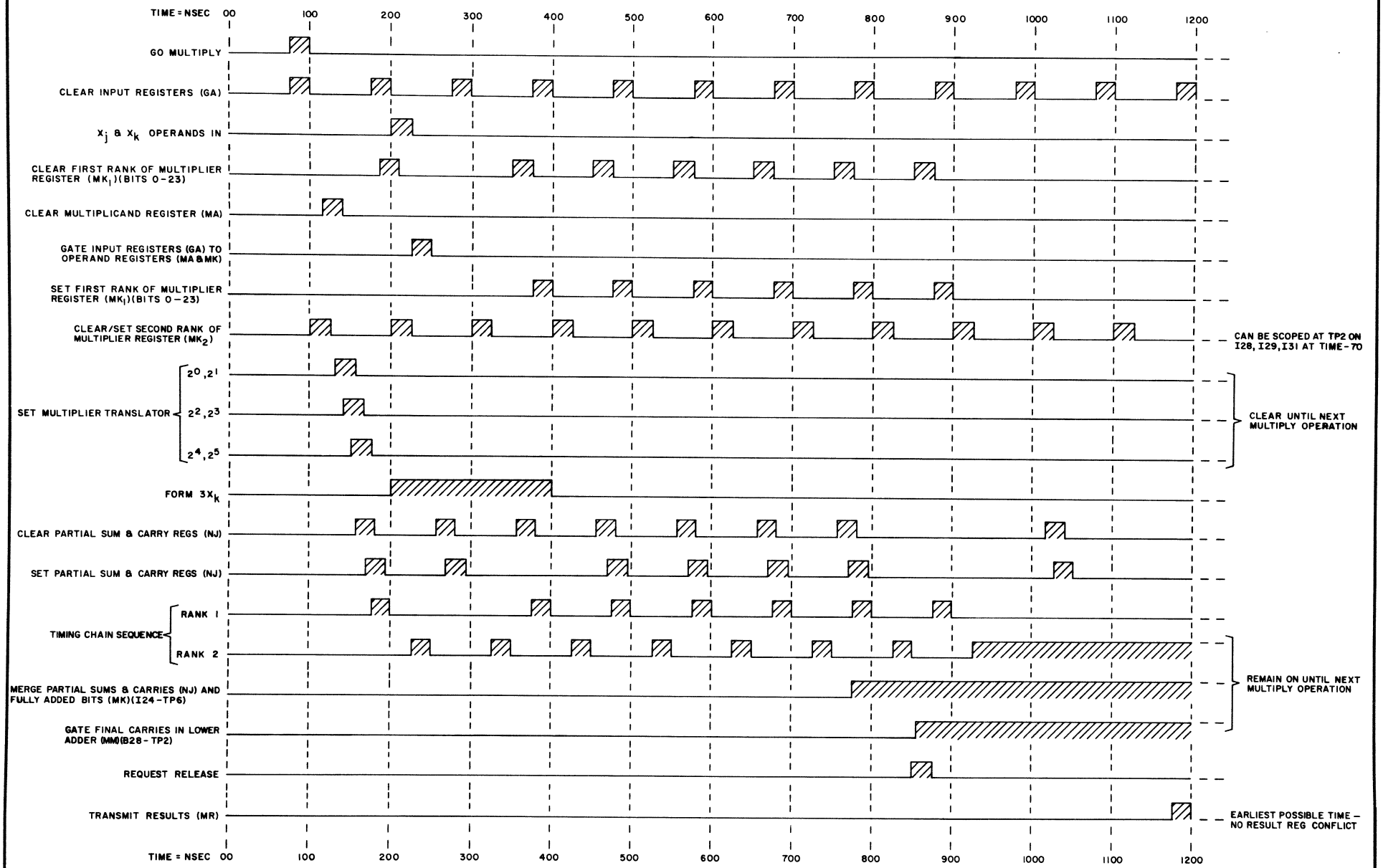




TO NORMALIZE, COEFFICIENT IS SHIFTED LEFT 1 PLACE; EXPONENT IS REDUCED BY 1.



CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR MULTIPLY BLOCK DIAGRAM	PRODUCT: 6601	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: C
		SHEET: 137	65



CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE
CENTRAL PROCESSOR
MULTIPLY UNIT
TIMING CHART

PRODUCT 6601
SIZE C | DRAWING NO. 60119300 | REV D
SHEET 216 | PAGE 66.1

MULTIPLY, COEFFICIENT

The Go signal to one of the multiply units directs that unit to take the two 48-bit quantities from the chassis 6 input registers and form their 96-bit product. The Go signal is timed so that the chassis 6 input registers are sampled while they are holding the multiply operands X_j and X_k.

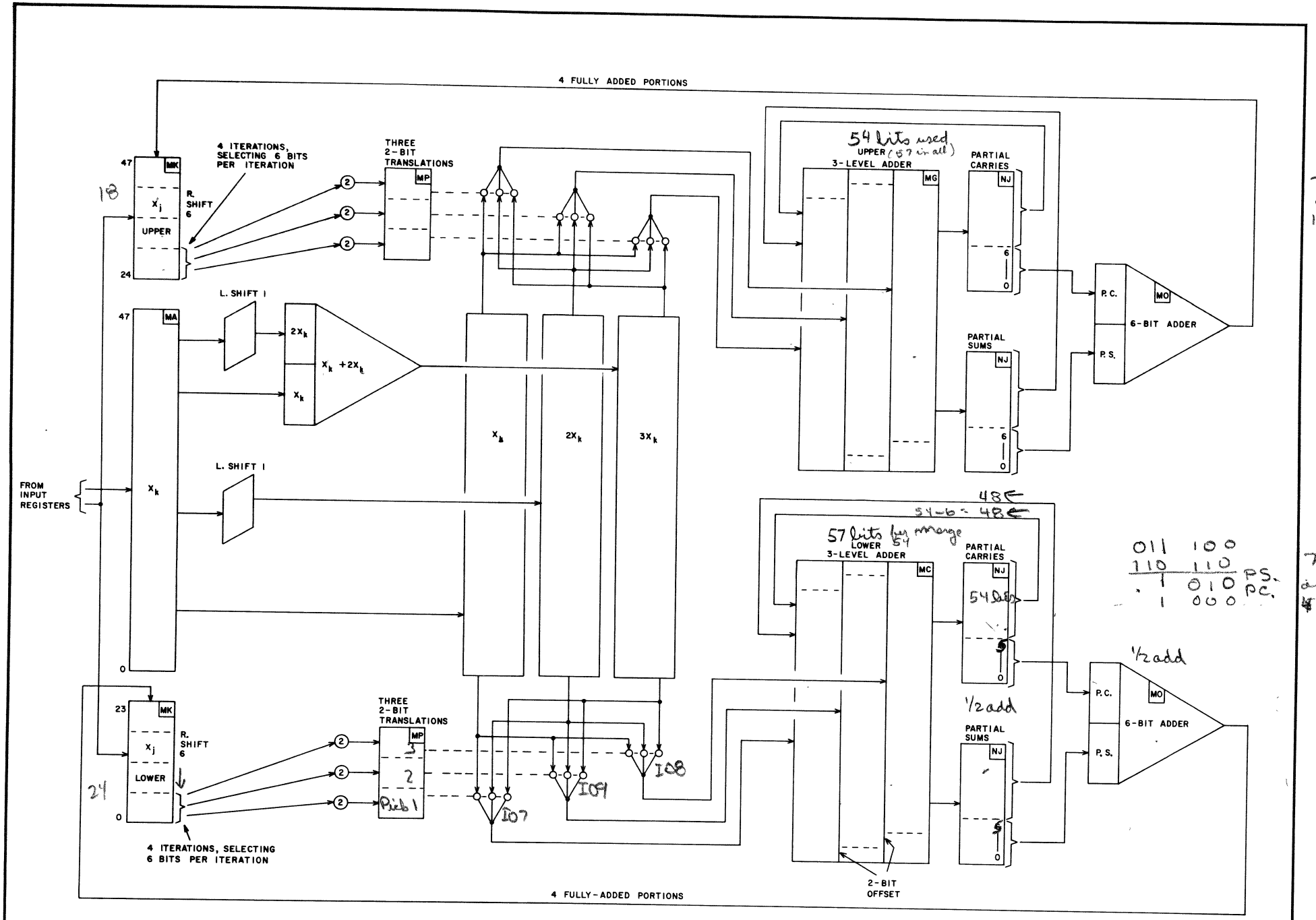
The multiplicand X_k is brought into a holding register where it remains during the entire operation. Three multiplied values of X_k are then formed: X_k times one, times two, and times three. These values are held available during the entire operation.

The multiplier X_j is divided into two 24-bit quantities, upper and lower. These two portions are set into registers which perform a 6-place right shift after each iteration.

Four iterations are required to multiply the two 48-bit operands. At the start of each iteration, three 2-bit translations are made from the lower six bits of the quantities held in the upper and lower multiplier registers. These translations select the multiplied values of X_k ($3X_k$, $2X_k$, X_k, or zero) which are to be added in that iteration. The quantities specified by the upper and lower multipliers are gated into the upper and lower three-level adders, respectively. The result of each of the three-level additions is a set of partial sums and partial carries, which are brought out separately and placed in separate registers. During the next iteration, these partial sums and carries (right-shifted 6 places) are fed back into the three-level adders to be added into the partial product; the final addition for the partial sums and carries resulting from the fourth iteration is performed during merge.

The lower six bits of both the upper and lower partial sums and carries are fully added in separate 6-bit adders during the latter part of each iteration. If either of these additions results in a carry to a seventh place, the carry is placed in stage 0 of the respective upper or lower partial carry register.

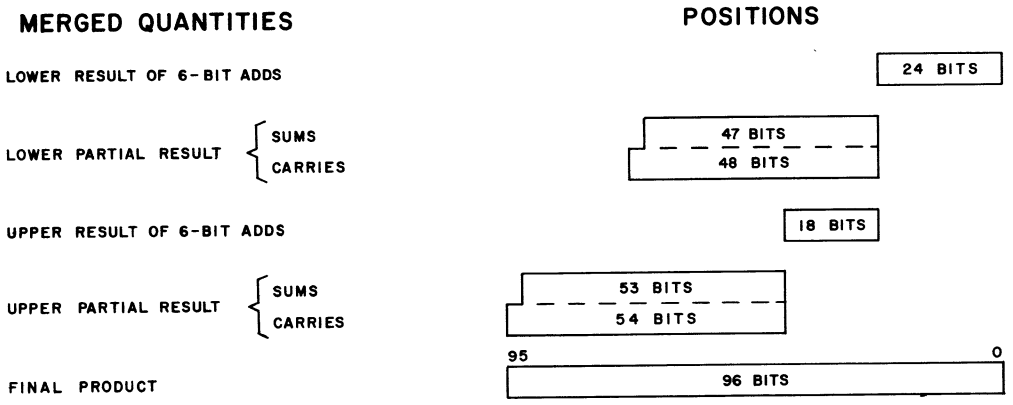
At the end of each iteration, a 6-place right shift is performed on both the upper and lower portions of the multiplier X_j. The multiplier bits used in the previous iteration are discarded. As the multipliers are shifted, the upper bits of the registers are refilled with the fully added output of the 6-bit partial sum and carry adders. At the end of four iterations, the lower multiplier register holds the fully added lower 24 bits of the final 96-bit product. The upper multiplier register holds only 18 bits because all of the upper partial sums and carries (except the upper 15) from the fourth iteration are sent directly to the lower adder for merge. The 18 bits in the upper multiplier register are added with lower partial sums and carries on merge to form bits 24 through 41 of the final 96-bit product.

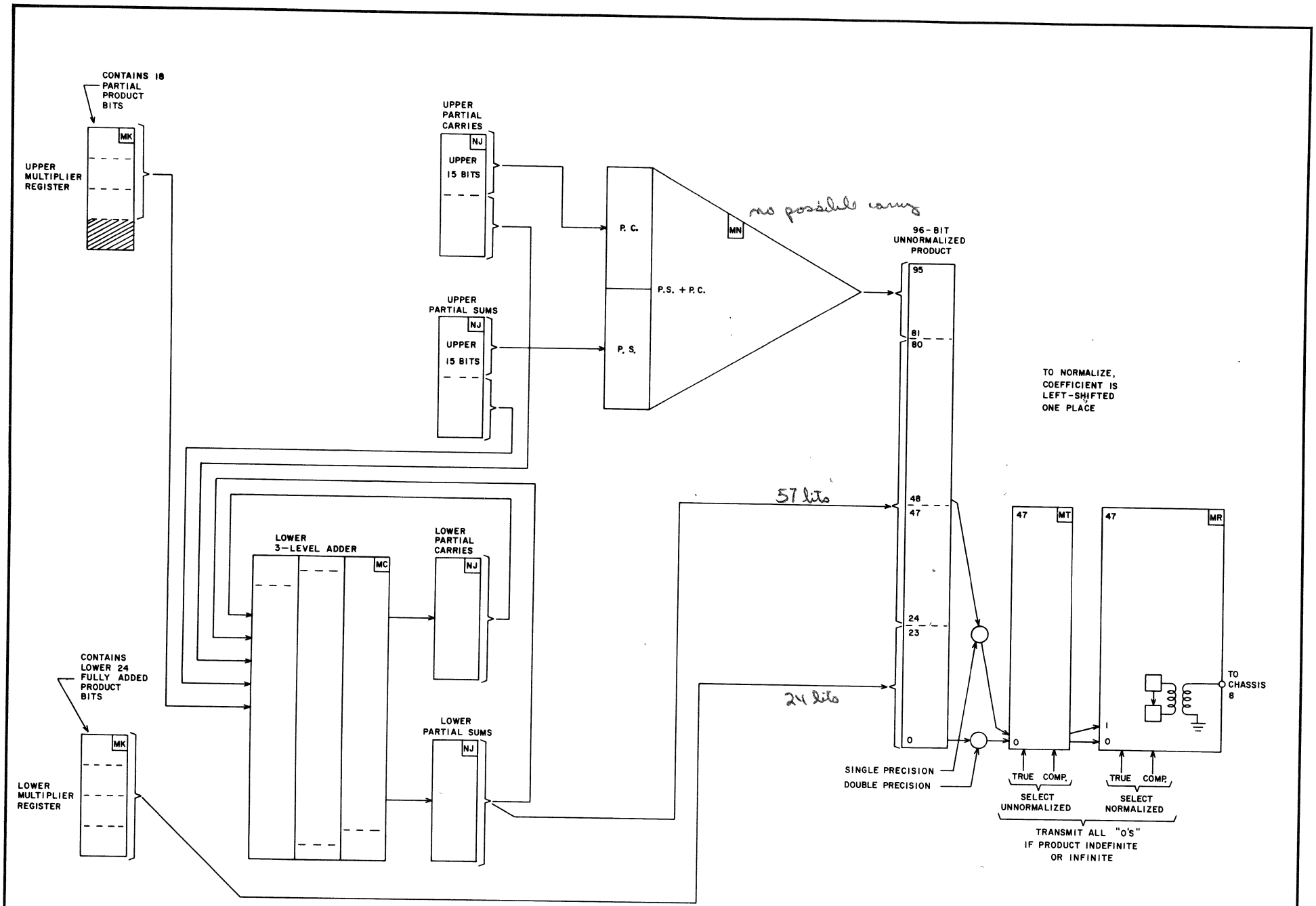


CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR MULTIPLY, COEFFICIENT BLOCK DIAGRAM, 4 ITERATIONS	PRODUCT: 6601/04	
COMPUTER DIVISION		SIZE: C DRAWING NO.: 60119300	REV: K
		SHEET: 138	67

The final 96-bit product is formed by merging the upper and lower partial sums and carries with the fully added contents of the upper and lower multiplier registers. The lower 24 bits of the final product are taken directly from the lower multiplier register, which contains the quantities produced by the lower 6-bit adder during each of the four iterations. Bits 24 through 80 of the final product are produced in the lower adder. Bits 24 through 41 are produced by merging lower partial sums and carries with the fully added 18-bit quantity in the upper multiplier register. The upper multiplier register receives only three 6-bit quantities; all partial sums and carries (except the upper 15) from the upper adder on the fourth iteration are sent directly to the lower adder for the merge operation. Final product bits 42 through 80 result from the 4-level add of upper and lower partial sums and carries. Bits 81 through 95 are not produced by the lower adder. These bits result from adding the upper partial sums and carries in a special 2-level adder used only during merge.

The result bits 24 through 80 from the lower adder are sent to the lower partial sum register. The final 96-bit product is therefore sent to the output network from three sources; the lower multiplier register, the lower partial sum register, and the upper 2-level merge adder.





CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR MULTIPLY, COEFFICIENT BLOCK DIAGRAM, MERGE	PRODUCT 6601 SIZE C · DRAWING NO. 60119300 · REV C SHEET 139	69

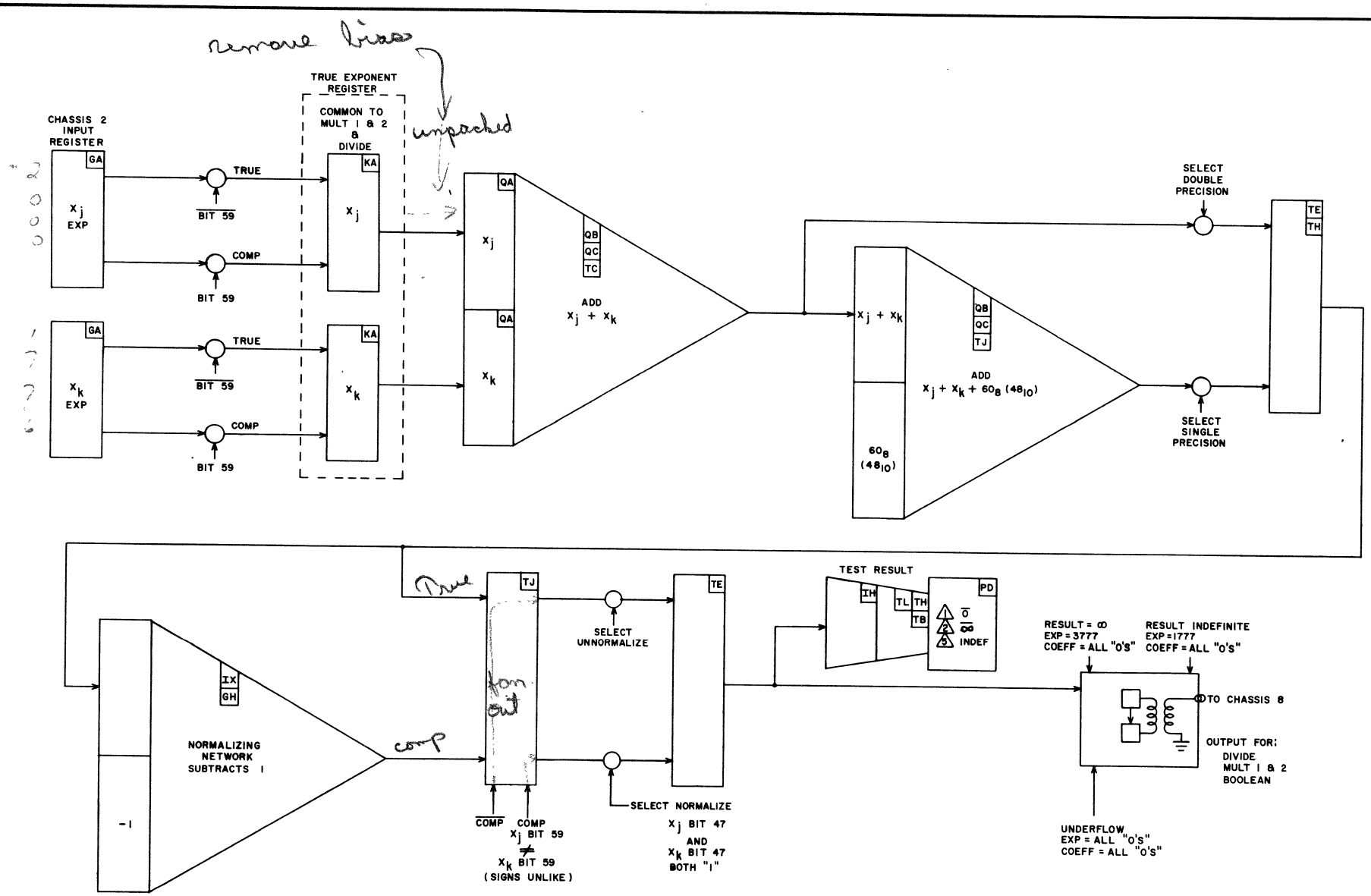
MULTIPLY, EXPONENT

All exponent arithmetic for the two multiply units is performed on chassis 2. The exponent portions of the two operands are sent directly to chassis 2 from the operating registers. The multiply unit receiving the Go signal on chassis 6 re-transmits the Go signal to the respective exponent arithmetic logic on chassis 2.

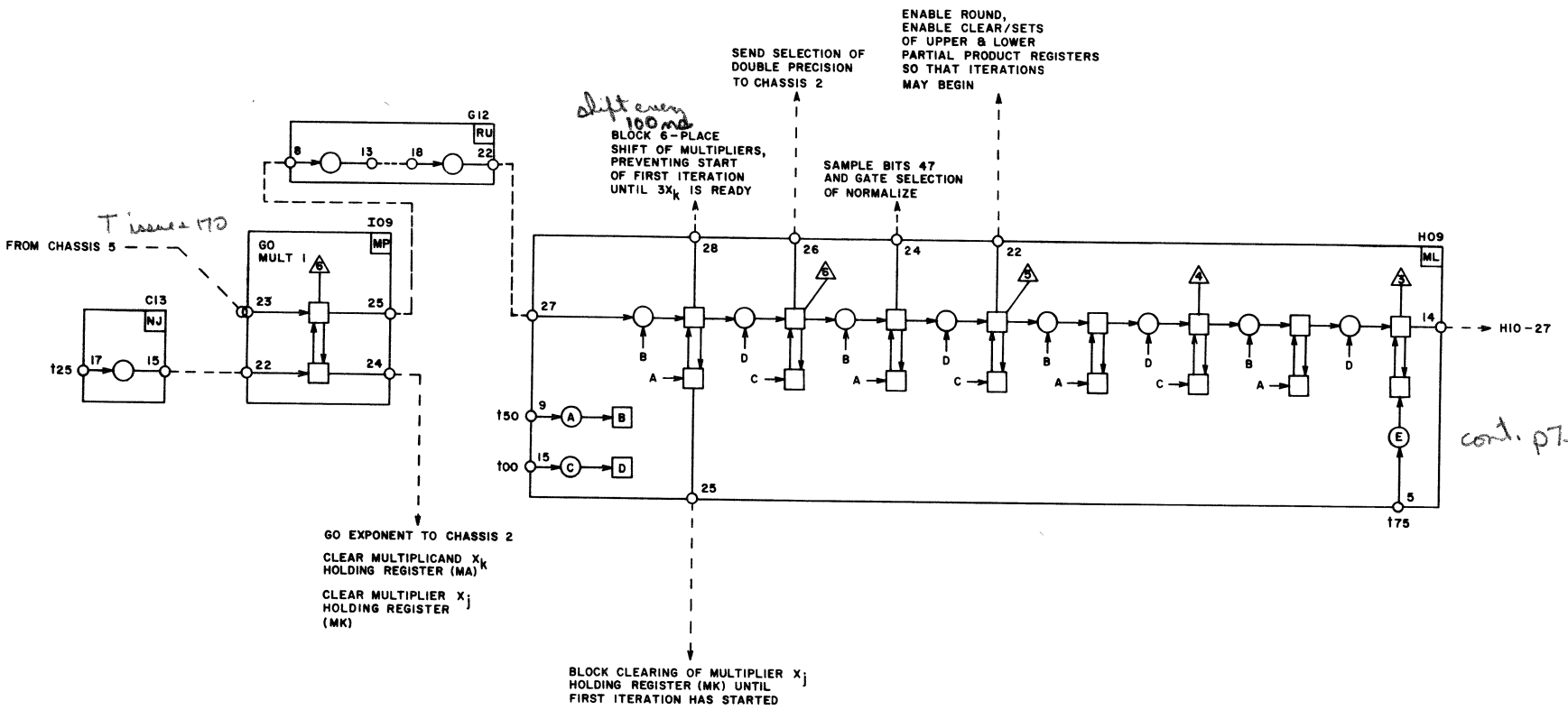
The final product exponent is found by adding the operand exponents according to the rules of exponential numbers. This exponent, produced by the direct addition, is the double-precision exponent; it is used if bits 0 through 47 of the 95 bit product are selected. The quantity 60_8 (48_{10}) must be added to the double-precision exponent to obtain the single-precision exponent; this is used if bits 48 through 95 of the 95-bit product are selected.

If it is necessary to normalize the product, the quantity 1 is subtracted from the result exponent. This is because in normalizing, the final product coefficient is left-shifted one place.

The final result exponent is tested to determine if the product is valid. If the product is infinite or indefinite, an Error signal is transmitted to chassis 6 and the final coefficient is held to all "0's".



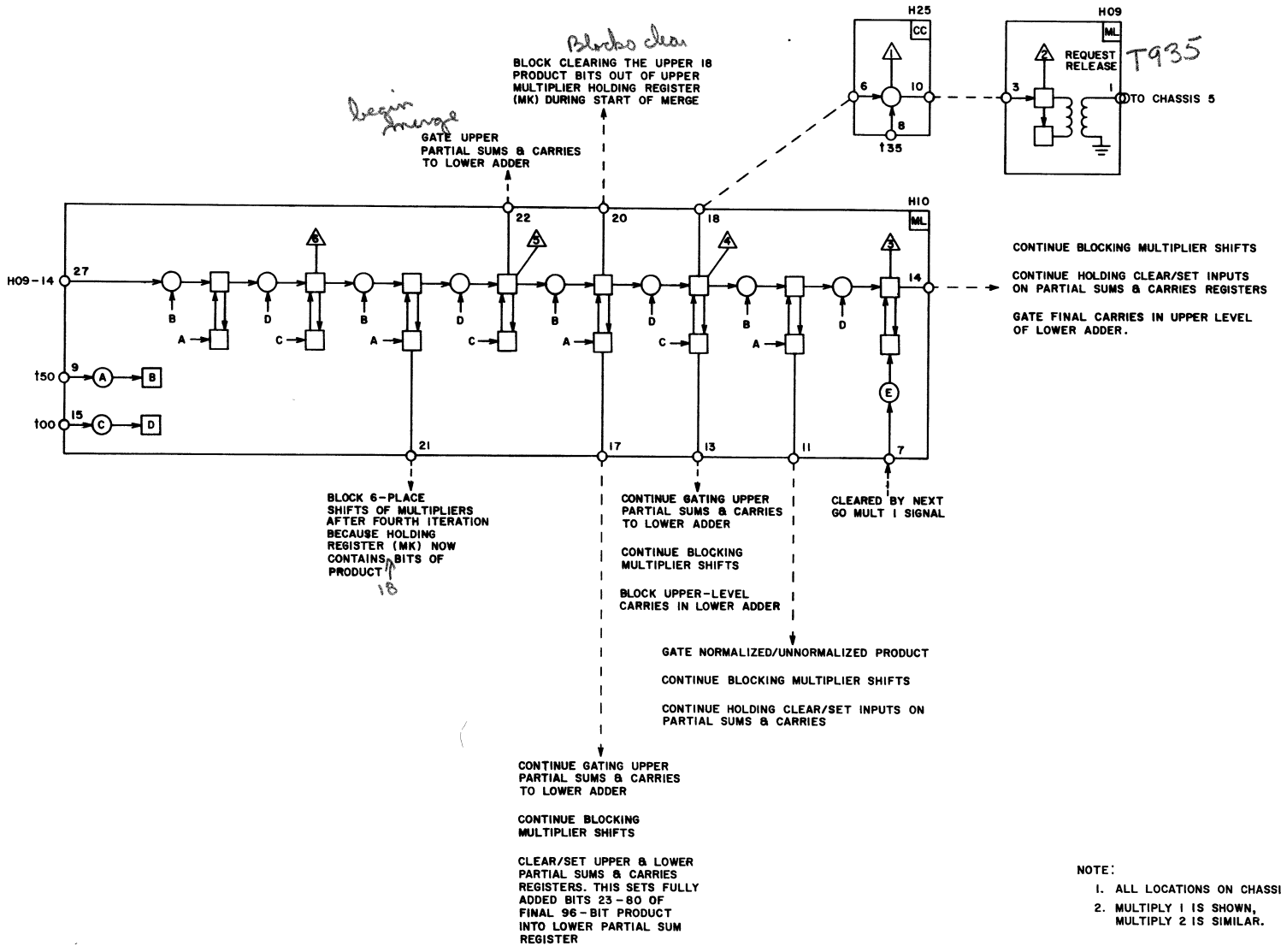
CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR MULTIPLY, EXPONENT BLOCK DIAGRAM	PRODUCT 6601	
		SIZE C	DRAWING NO. 60119300 REV C
		SHEET 140	71



NOTES:

1. ALL LOCATIONS ON CHASSIS 6.
2. MULTIPLY 1 IS SHOWN, MULTIPLY 2 IS SIMILAR.

CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR MULTIPLY TIMING CHAIN, CHASSIS 6	PRODUCT 6601/04/13/14	
		SIZE C	DRAWING NO. 60119300 REV AH
		SHEET 141	73



NOTE:

1. ALL LOCATIONS ON CHASSIS 6.
2. MULTIPLY I IS SHOWN, MULTIPLY 2 IS SIMILAR.

CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR MULTIPLY TIMING CHAIN, CHASSIS 6	PRODUCT: 6601	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: BT
		SHEET: 142	75

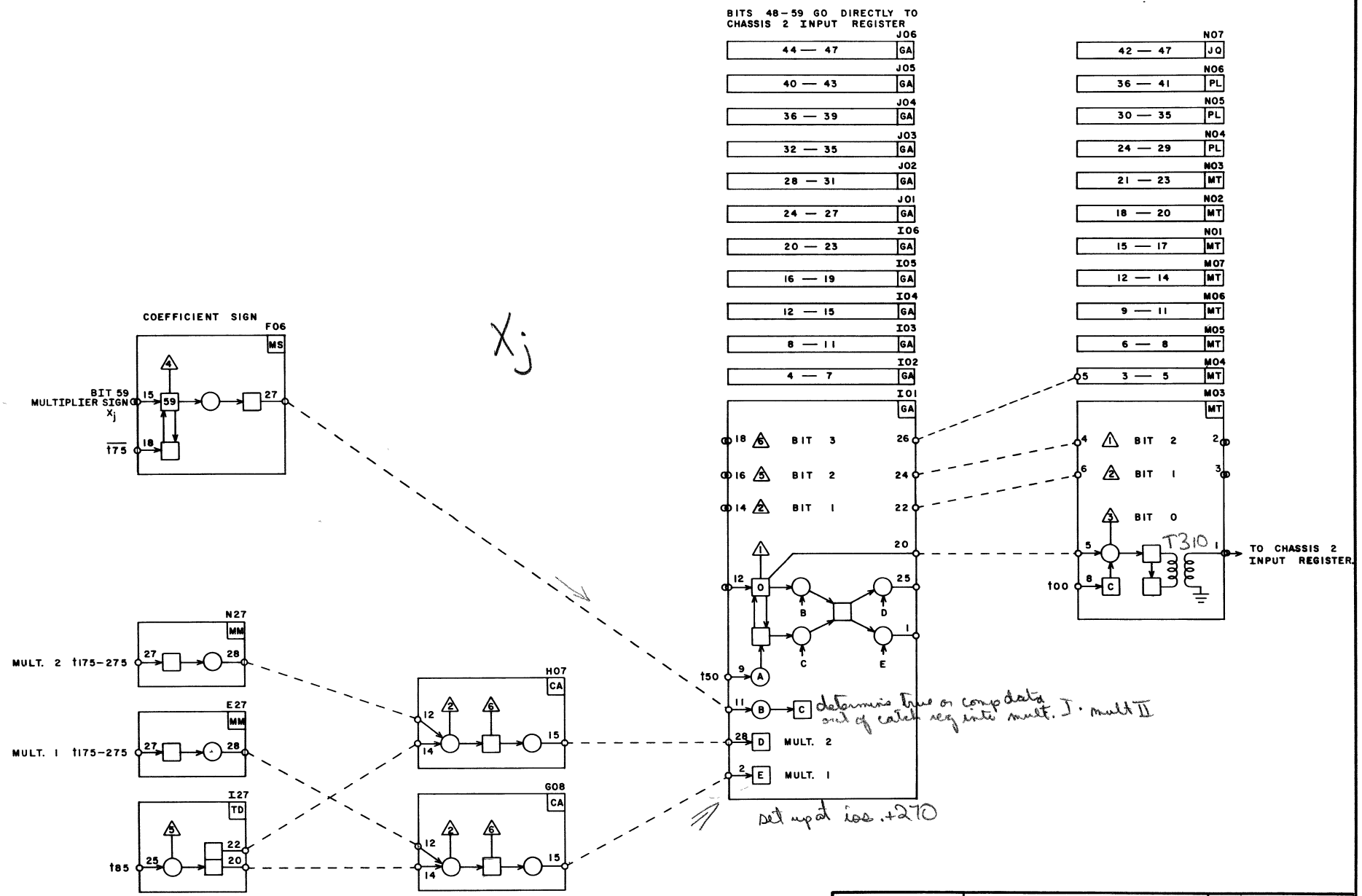
CHASSIS 6 INPUT REGISTERS

Chassis 6 contains the coefficient arithmetic logic of both Multiply 1 and Multiply 2 functional units. Since no other operations are performed on chassis 6, its input registers are only 48 bits in length. The exponents of the two operands are not sent to chassis 6 but instead go directly to the input registers on chassis 2.

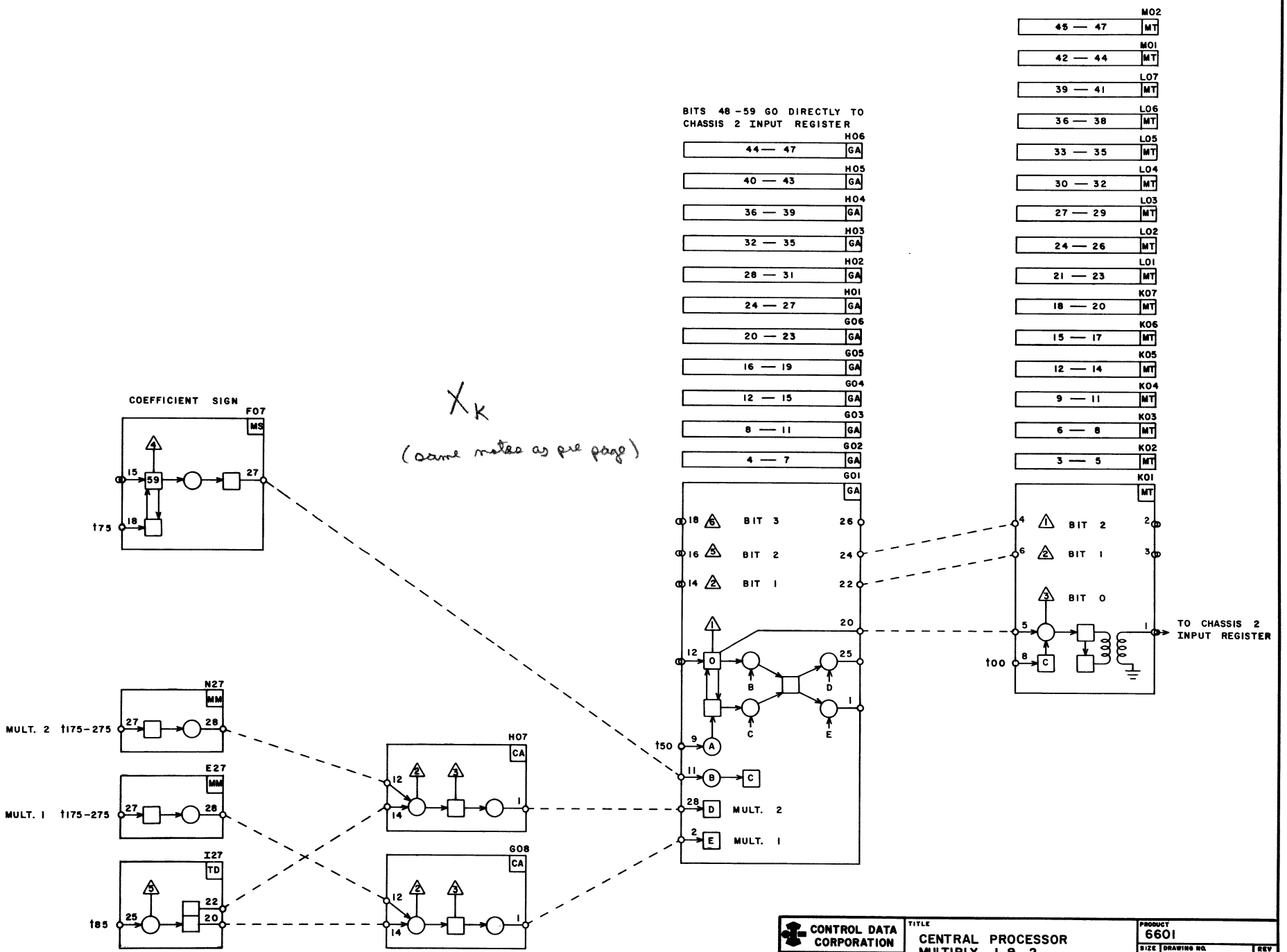
The input registers on chassis 6 are shared by both Multiply 1 and 2. The unit receiving the Go signal samples the input registers and performs the multiply operation.

The two operands are held in the chassis 6 input registers for a time of approximately 100 nanoseconds. During this time, the operands are transmitted on to chassis 2, and the chassis 6 input registers are then cleared. If a multiplication is to be performed, the operation is timed so that one of the multiply units receives a Go signal and samples the operands while they are held in the chassis 6 input registers.

The coefficient sign (bit 59) of each operand is sent to both chassis 2 and chassis 6. A sign bit of "1" indicates a negative operand. The input registers always contain the true value of the operands, but since negative numbers are handled in one's complement form, a sign bit of "1" gates the complement of the quantity into the multiply unit.



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR MULTIPLY 1 & 2 MULTIPLIER X_j, INPUT REGISTER, CHASSIS 6	PRODUCT 6601 SIZE C · DRAWING NO. 60119300 · REV BT SHEET 143	77



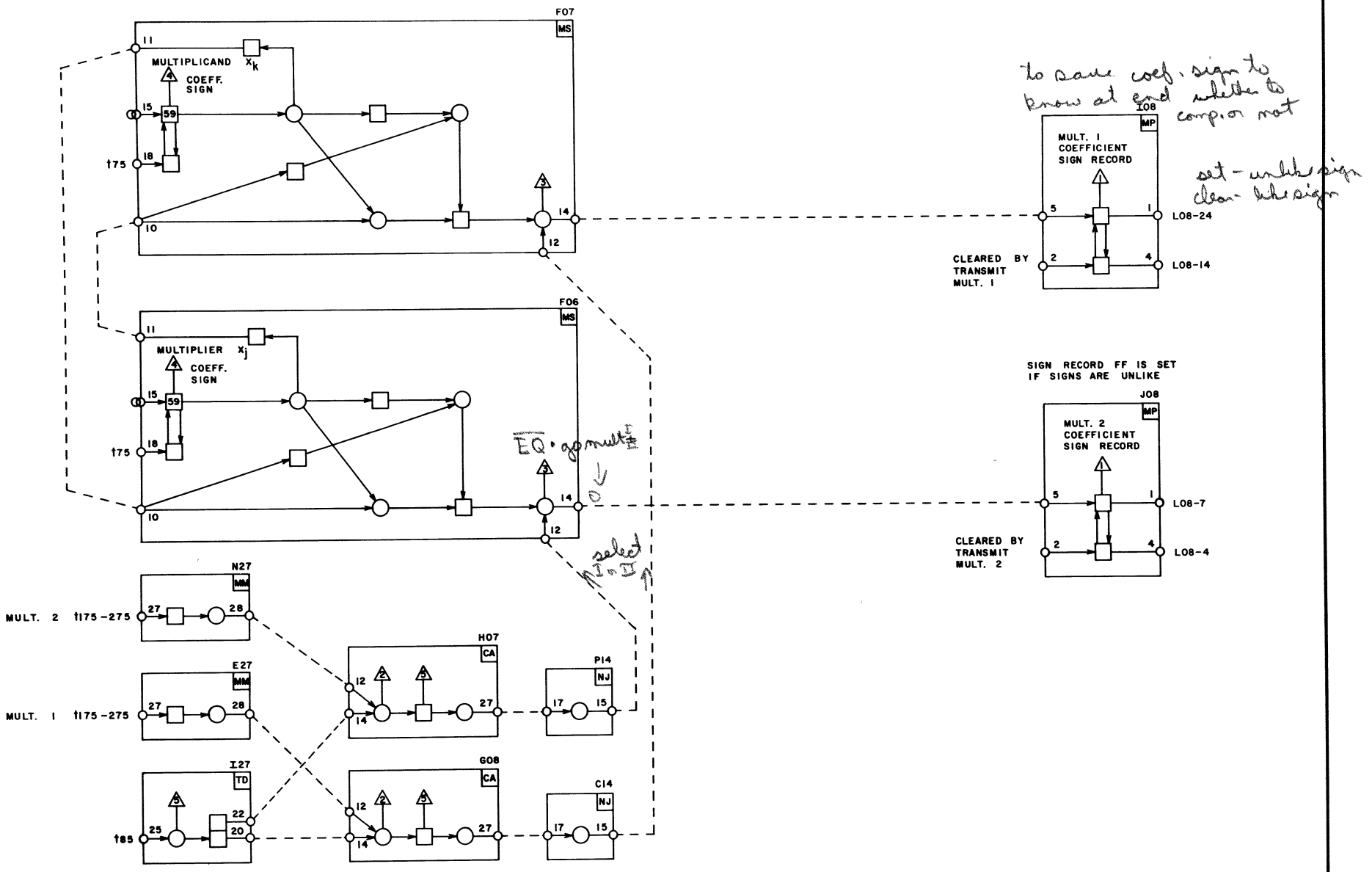
CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR MULTIPLY 1 & 2 MULTIPLICAND X_k, INPUT REGISTER, CHASSIS 6	PRODUCT 6601 SIZE C · DRAWING NO. 60119300 · REV BT SHEET 144	79

SIGN RECORD

The sign of the product is determined according to standard rules. If the operands are of the same sign, the product is positive. If the operand signs are unlike, the product is negative.

A comparison test is always performed on the signs of the two operands coming into chassis 6. If a multiplication is not performed, the result of the comparison is not used. However, if either multiply unit has received a Go signal, the result of the comparison is stored and held until the end of the operation.

The sign of the final product is determined by the state of the Sign Record flip-flop of the respective multiply unit. If the initial operand signs are the same, the Sign Record flip-flop remains clear. This indicates a positive final product positive with a sign bit of "0". If the initial operand signs are unlike, the Sign Record flip-flop is set, indicating a negative product with a sign bit of "1".



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE: CENTRAL PROCESSOR MULTIPLY 1 & 2 COEFFICIENT SIGN RECORD	PRODUCT 6601 SIZE C · DRAWING NO. 60119300 · REV BT SHEET 145	81

MULTIPLIER X_j

The 48-bit multiplier X_j is divided into upper and lower halves of 24 bits each. These two quantities are placed in registers which perform a 6-place right shift at the end of each iteration.

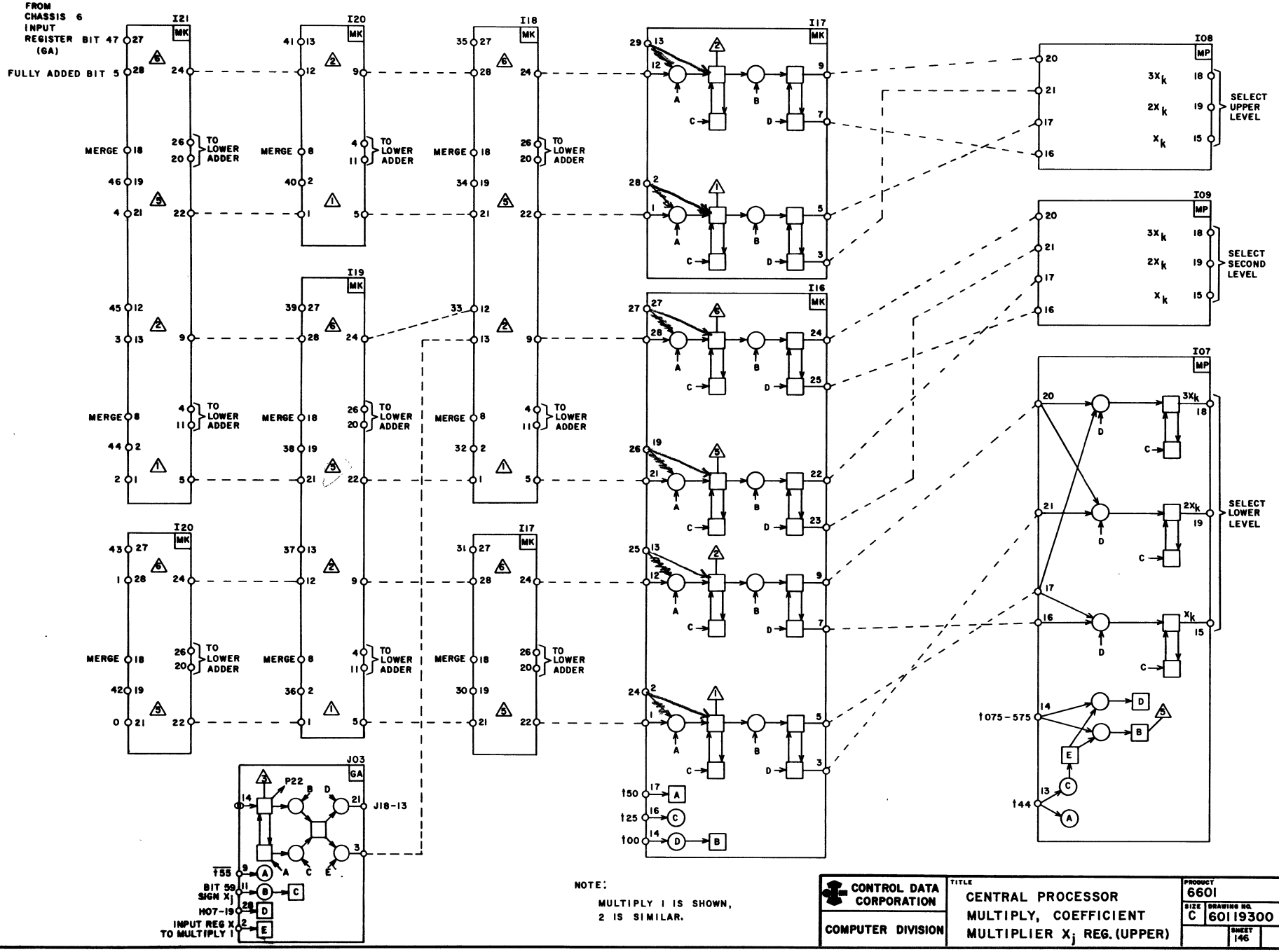
The lower 6 bits of each of the multiplier sections select quantities to be added to the partial product on each iteration. The multiplier is thereby processed 12 bits at a time, enabling a 48-bit multiplication to be performed in 4 iterations.

At the beginning of each iteration, the lower 6 bits of each multiplier are examined to select 3 values of the multiplicand X_k. The 3 quantities are then gated into the 3-level adder. Since the multiplier is examined in 2-bit groups, the selected values of the multiplicand may be either $3X_k$, $2X_k$, X_k, or zero. These functions of the multiplicand X_k are held constantly available during the operation.

The two multipliers are shifted 6 places to the right at the end of each iteration. The previously used 6-bit portions are discarded. The upper 6 stages of the two multiplier registers are then used as a place to hold the result of the addition of partial sum and carry bits 0 through 5 (both upper and lower). After the fourth iteration, nothing remains of the original multiplier X_j. The lower multiplier register holds the fully added lower 24 bits of the final 96-bit product and the upper multiplier register holds 18 bits which must be added with lower partial sums and carries on merge. Note that the upper multiplier receives only 3 fully added 6-bit quantities, since those upper partial sums and carries from the fourth iteration are sent directly to the lower adder for merge.

24-BIT, 6-PLACE SHIFT REGISTER

AS MULTIPLIER X_j IS SHIFTED, UPPER 18 BITS OF REGISTER
ARE REFILLED WITH FULLY ADDED 6-BIT QUANTITIES FROM MØ MODULES.

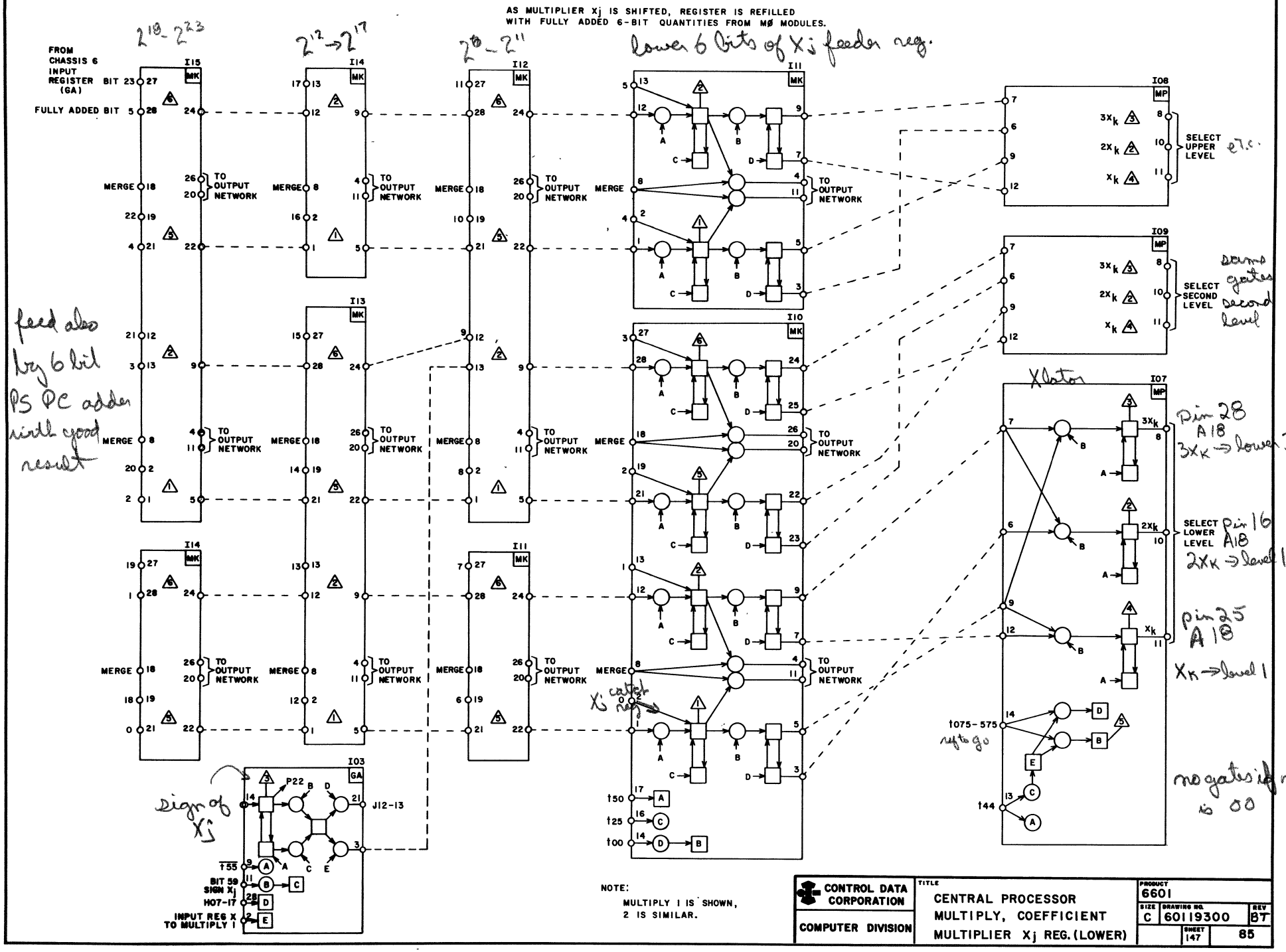


NOTE:
MULTIPLY I IS SHOWN,
2 IS SIMILAR.

CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR MULTIPLY, COEFFICIENT MULTIPLIER X_j REG. (UPPER)	PRODUCT 6601	
COMPUTER DIVISION		SIZE C · DRAWING NO. 60119300 · REV BT	
		SHEET 146	83

24-BIT, 6-PLACE SHIFT REGISTER

AS MULTIPLIER Xj IS SHIFTED, REGISTER IS REFILLED
WITH FULLY ADDED 6-BIT QUANTITIES FROM MØ MODULES.



NOTE:
MULTIPLY 1 IS SHOWN,
2 IS SIMILAR.

CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE
CENTRAL PROCESSOR
MULTIPLY, COEFFICIENT
MULTIPLIER Xj REG. (LOWER)

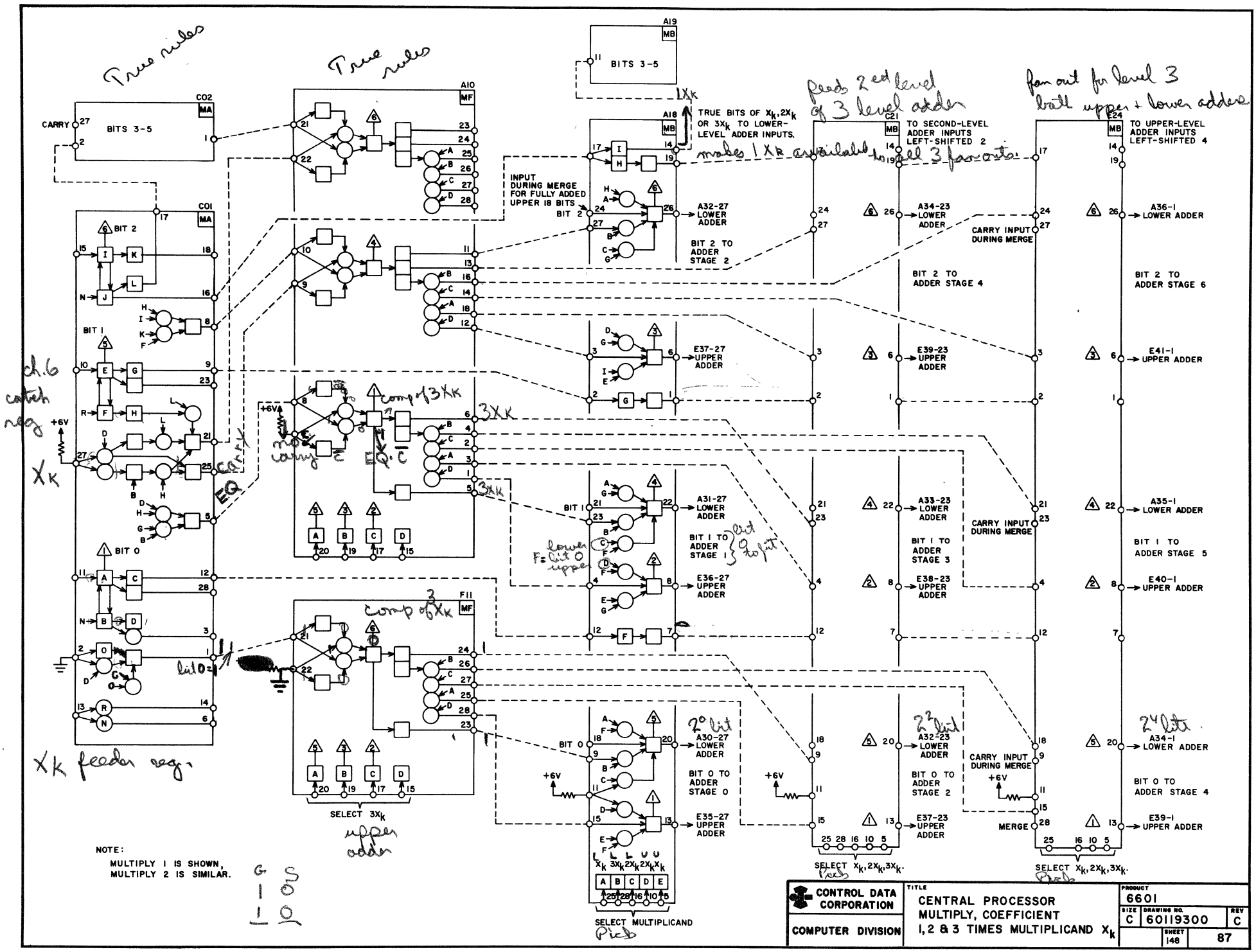
PRODUCT 6601
SIZE C | DRAWING NO. 60119300 | REV BT
SHEET 147 | 85

MULTIPLICAND X_k

At the beginning of the multiply operation, the multiplicand X_k is brought into its register where it is held during the remainder of the operation. Three values are formed; the multiplicand X_k, X_k times 2, and X_k times 3. These three values are made available at each of the inputs to the 3-level adders, with 2-place shifts performed between levels. The quantity gated in at each level depends upon the translation of those 2 bits of the multiplier X_j; if the two multiplier bits are "0", nothing is gated in.

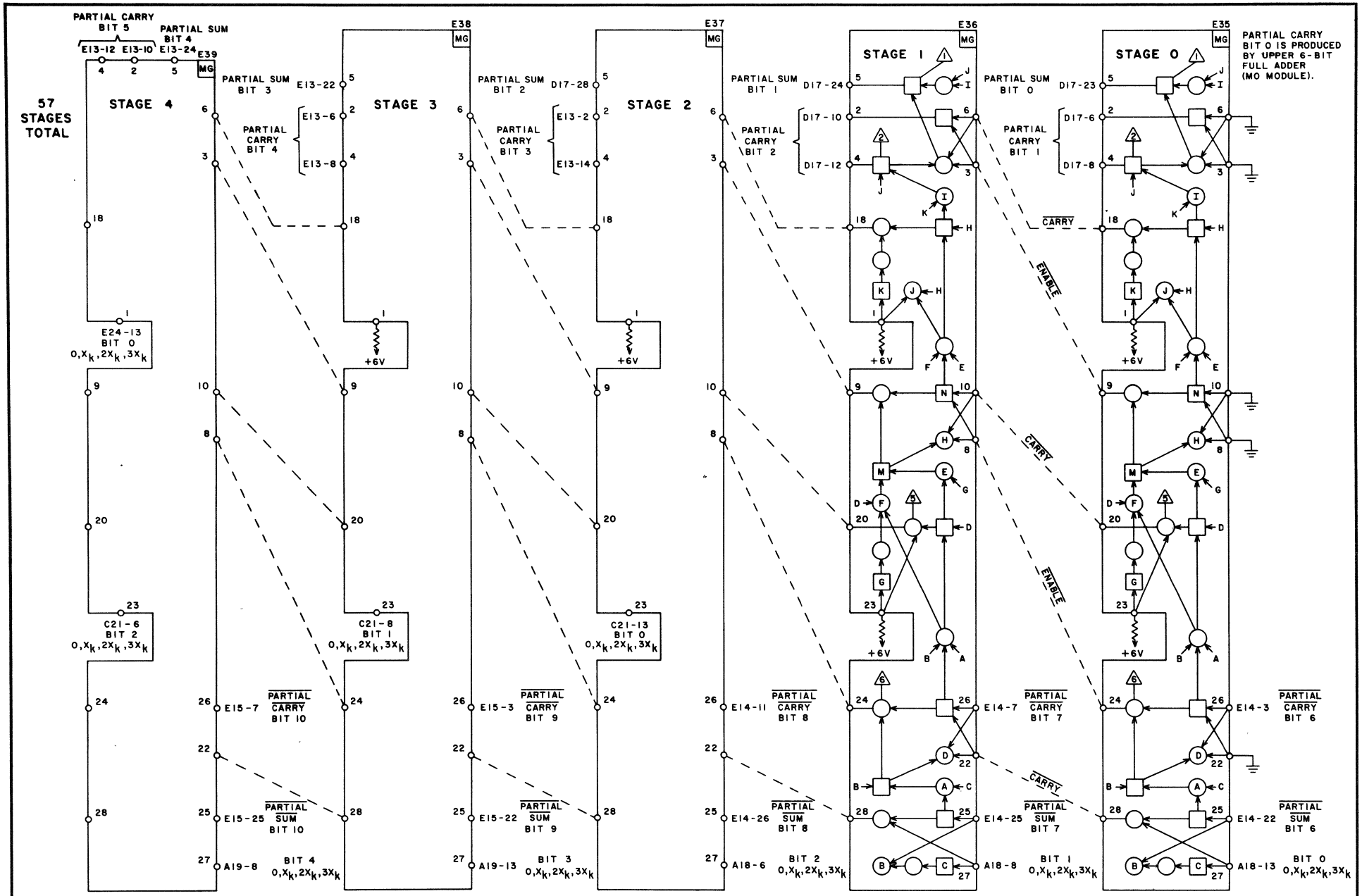
The quantity $2X_k$ is formed by shifting X_k one place to the left. This is equivalent to multiplying X_k by 010_2. The low-order bit of $2X_k$ is always "0".

The quantity $3X_k$ is formed by shifting X_k one place to the left, then adding this shifted quantity to the original X_k. This is equivalent to multiplying X_k by 011_2. Carries are generated in this process, and all carries are handled in parallel by a separate sensing network. Because of possible carries at the high-order bit positions, the quantity $3X_k$ may be 50 bits in length.


True rules
True rules
CO2
MA
CARRY
BITS 3-5
CO1
BIT 2
BIT 1
BIT 0
A10
MF
A19
MB
BITS 3-5
1Xk
A18
TRUE BITS OF Xk, 2Xk OR 3Xk TO LOWER-LEVEL ADDER INPUTS.
feeds 2nd level of 3 level adder
makes 1Xk available to all 3 fan outs.
fan out for level 3 both upper + lower adders
C21
TO SECOND-LEVEL ADDER INPUTS LEFT-SHIFTED 2
C24
TO UPPER-LEVEL ADDER INPUTS LEFT-SHIFTED 4
INPUT DURING MERGE FOR FULLY ADDED UPPER 18 BITS
BIT 2
A32-27 LOWER ADDER
BIT 2 TO ADDER STAGE 2
A34-23 LOWER ADDER
CARRY INPUT DURING MERGE
BIT 2 TO ADDER STAGE 4
A36-1 LOWER ADDER
BIT 2 TO ADDER STAGE 6
E37-27 UPPER ADDER
E39-23 UPPER ADDER
E41-1 UPPER ADDER
ch.6 catch reg
+6V
Xk
comp 3Xk
3Xk
mod carry
EQ
A31-27 LOWER ADDER
BIT 1 TO ADDER STAGE 1
lower F=1 if 0 to 1 upper
A33-23 LOWER ADDER
BIT 1 TO ADDER STAGE 3
A35-1 LOWER ADDER
BIT 1 TO ADDER STAGE 5
E36-27 UPPER ADDER
E38-23 UPPER ADDER
E40-1 UPPER ADDER
F11
comp of Xk
lsb 0=1
Xk feeder reg.
2° bit
A30-27 LOWER ADDER
BIT 0 TO ADDER STAGE 0
2² bit
A32-23 LOWER ADDER
BIT 0 TO ADDER STAGE 2
2⁴ bit
A34-1 LOWER ADDER
BIT 0 TO ADDER STAGE 4
E35-27 UPPER ADDER
E37-23 UPPER ADDER
E39-1 UPPER ADDER
SELECT 3Xk
upper adder
SELECT MULTIPLICAND
Xk 3Xk 2Xk 2Xk Xk
SELECT Xk, 2Xk, 3Xk.
SELECT Xk, 2Xk, 3Xk.
NOTE:
MULTIPLY 1 IS SHOWN,
MULTIPLY 2 IS SIMILAR.
G 1 0
1 0
CONTROL DATA CORPORATION
COMPUTER DIVISION
TITLE
CENTRAL PROCESSOR
MULTIPLY, COEFFICIENT
1, 2 & 3 TIMES MULTIPLICAND Xk
PRODUCT 6601
SIZE C
DRAWING NO. 60119300
REV C
SHEET 148
87

THREE-LEVEL ADDERS, UPPER AND LOWER

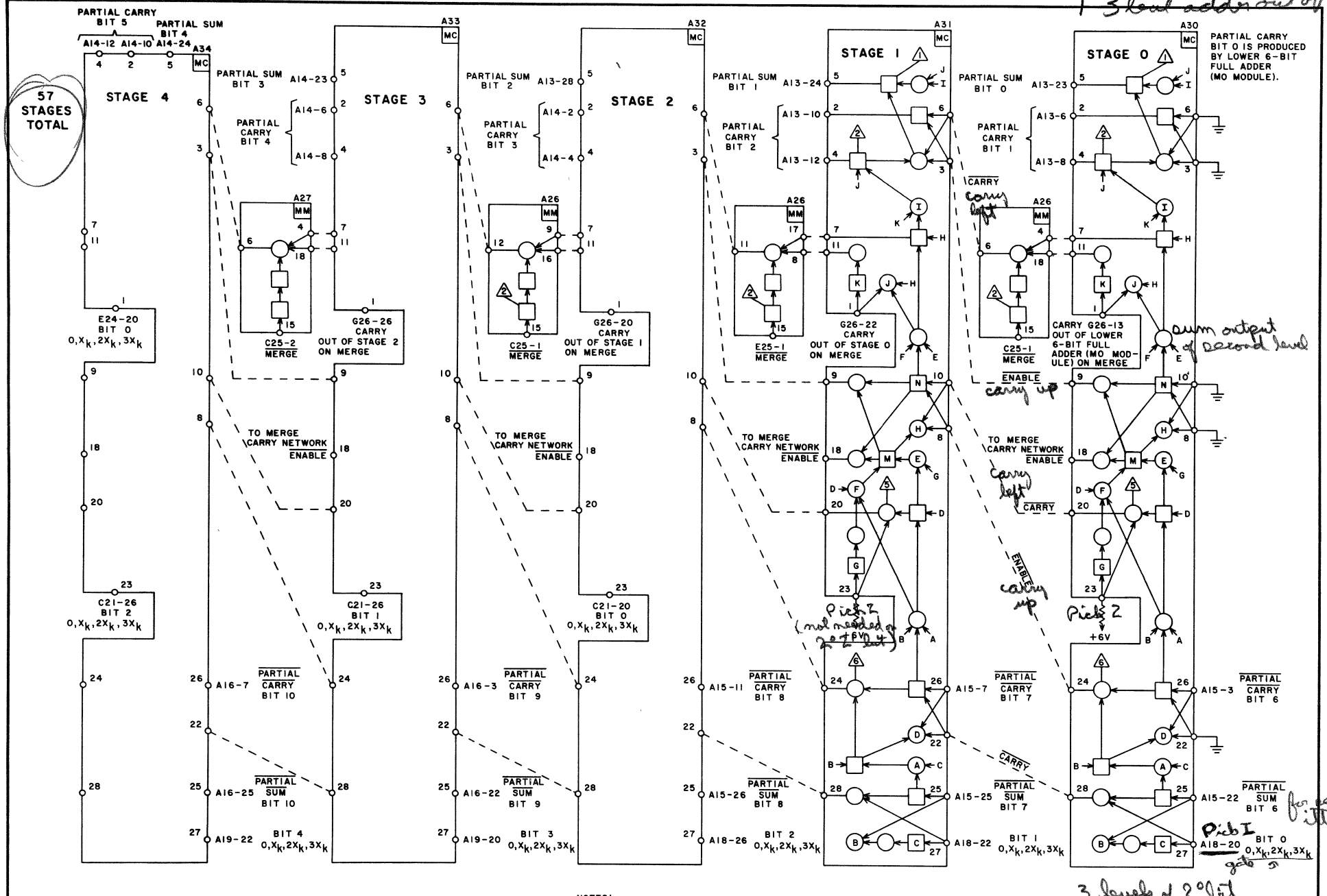
On each iteration, the 3-level adders perform the addition of the 3 selected values of the multiplicand X_k together with the partial sums and carries from the previous iteration. To maintain positional accuracy in the partial product, the bit positions of the multiplicand values entering upper levels of the adder are left-shifted 2 places with respect to the next lower level, and the partial sums and carries from the previous iteration are fed back into the lower level shifted 6 places to the right. In effect, the partial product sums and carries are being shifted as the multiplier is shifted, but the multiplicand values entering the adder are not; bit 0 of the lower-level multiplicand value always enters adder stage 0 on every iteration.



NOTES:

1. ALL LOCATIONS ON CHASSIS 6.
2. MULTIPLY 1 SHOWN, 2 IS SIMILAR.

CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR MULTIPLY, COEFFICIENT UPPER ADDER, 3-LEVEL	PRODUCT: 6601	
COMPUTER DIVISION		SIZE: C — DRAWING NO.: 60119300	REV: C
		SHEET: 149	89



NOTES:

1. ALL LOCATIONS ON CHASSIS 6.
2. MULTIPLY 1 SHOWN, 2 IS SIMILAR.

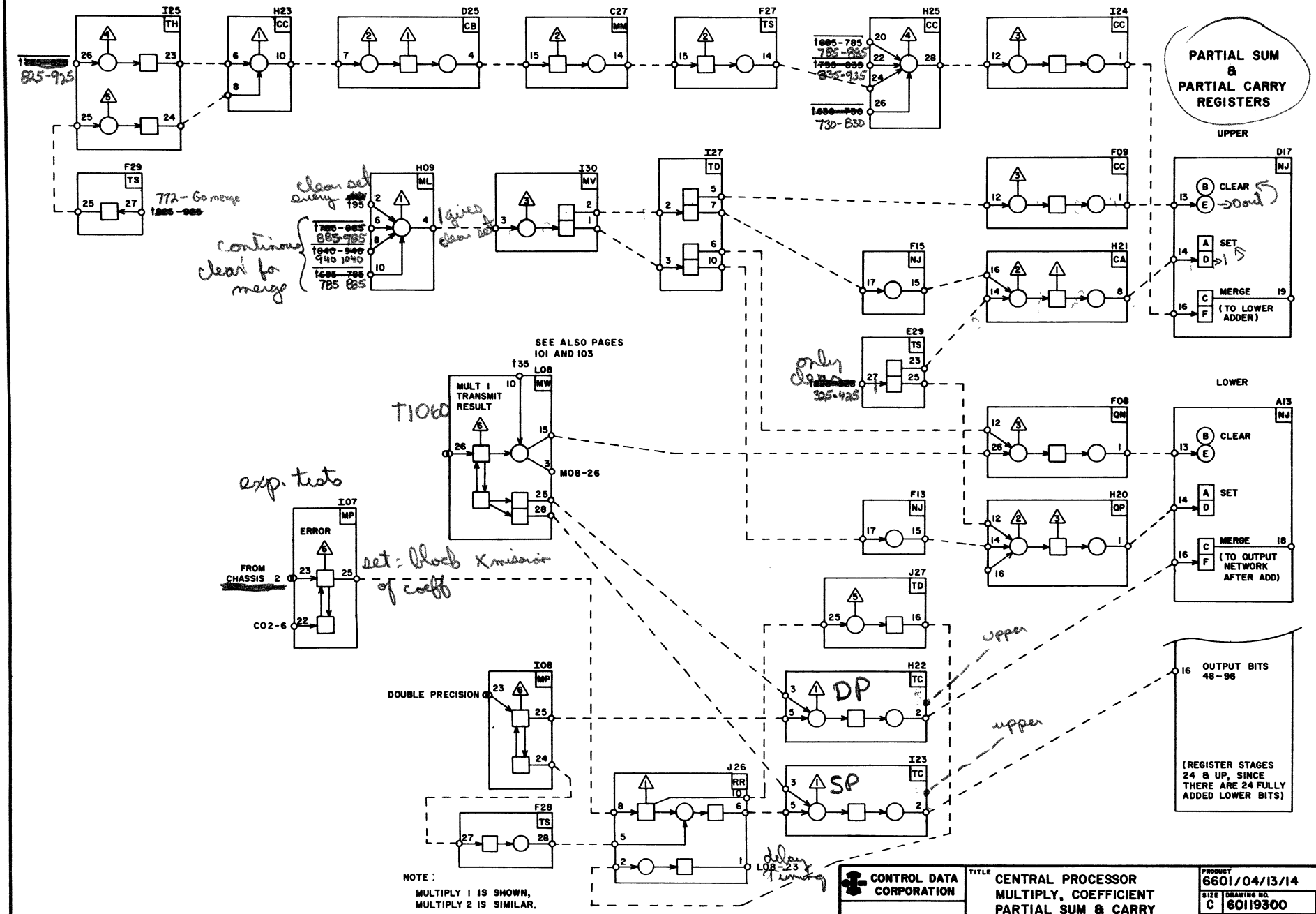
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR MULTIPLY, COEFFICIENT LOWER ADDER, 3-LEVEL	PRODUCT 6601	
COMPUTER DIVISION		SIZE C	DRAWING NO. 60119300 REV C
			SHEET 150 91

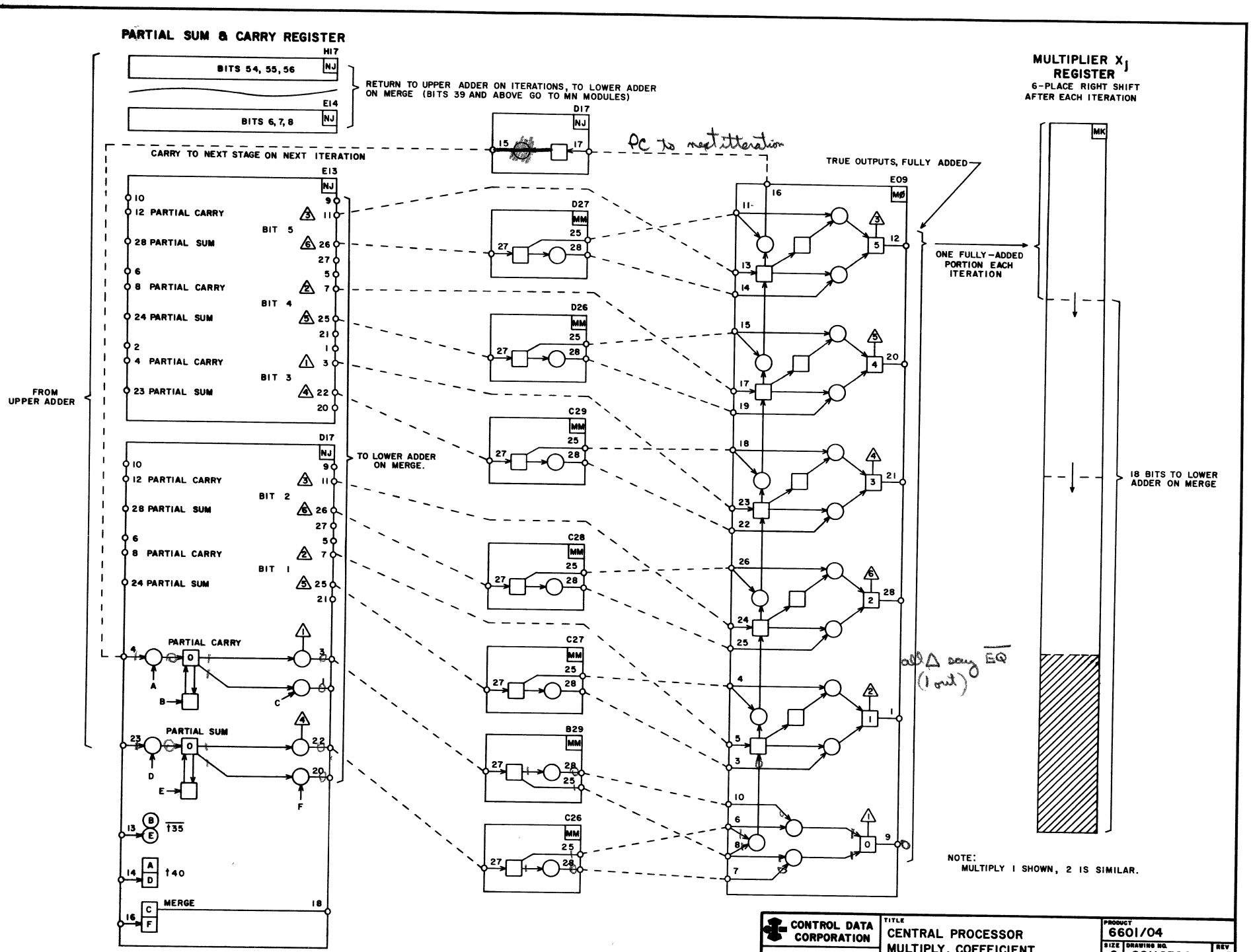
PARTIAL SUM AND PARTIAL CARRY REGISTERS

At the end of each iteration, the partial sums and carries resulting from the 3-level additions are gated into their respective registers. The lower six bits of both sets of partial sums and carries are then fully added, since the 6-place shift of the multiplier insures that there will be no more bits to be added to these. The remaining partial sums and carries are then returned to the 3-level adders to be added into the partial product on the next iteration. It is therefore necessary to clear and reset the partial sum and partial carry registers at the end of each iteration. As shown on the accompanying diagram, the AND gates which do this are enabled at each t95 after t325.

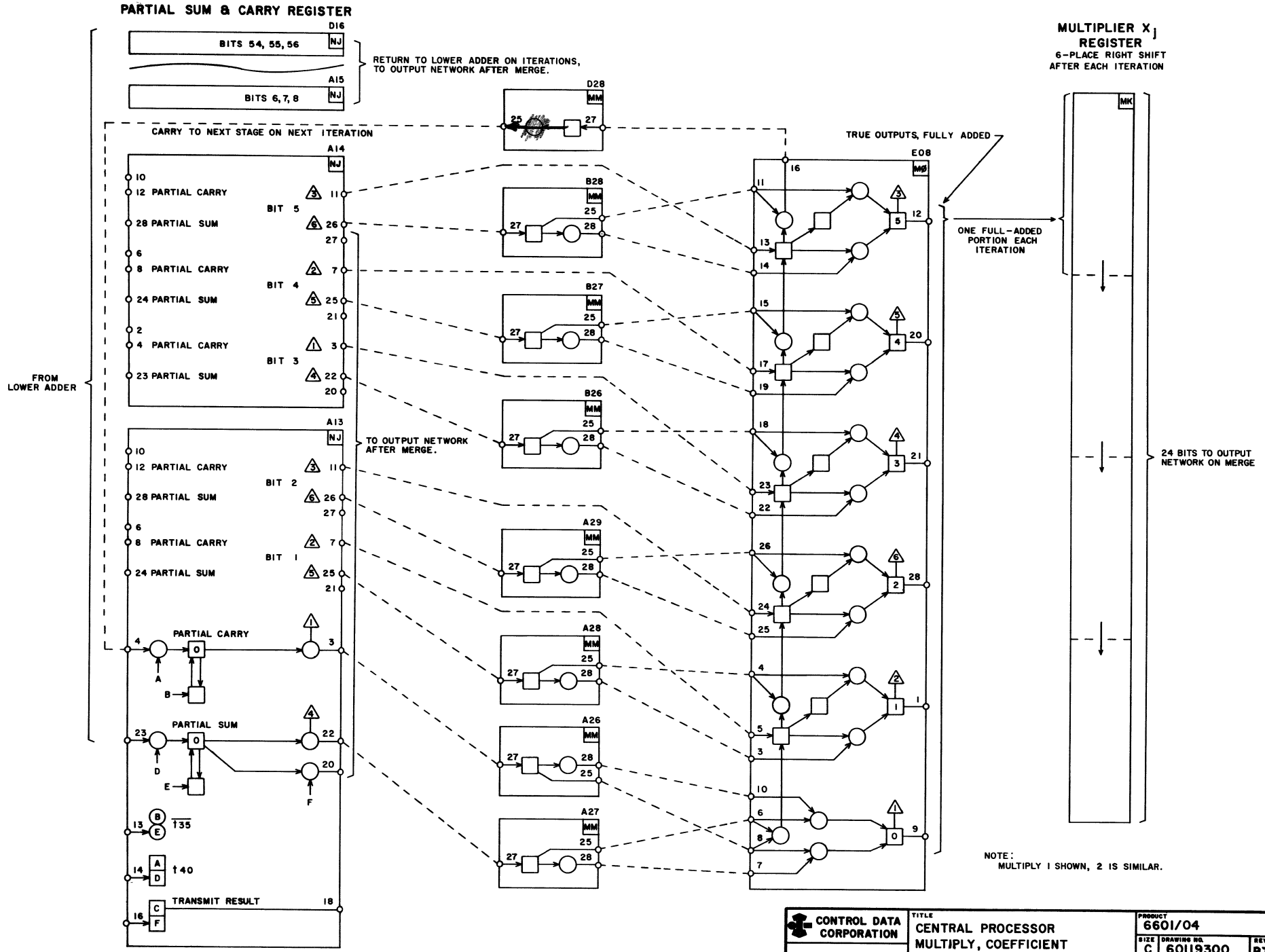
Following the fourth iteration, all of the partial results are merged into the final 96-bit product. For the merge operation, all partial sums and carries are sent to the lower adder for final addition (except the upper 15 bits of the upper partial sums and carries which are added in a special 2-level network). The final product is then transmitted through the lower partial sum register to the output network. The merge gate for the upper partial sums and carries is enabled at t630, and at t685, the registers are cleared. These gates are held enabled through the remainder of the operation.

If the Error signal is received from chassis 2, the final product from the output network is all "0's". This signal results from the exponent test and indicates that the product is either infinite or indefinite. Assuming no Error signal, the clear and set enables of the lower partial sum and carry registers are brought up at t685 and held until the end of the operation. As soon as the Transmit signal is received, the merged product is gated to the output network.

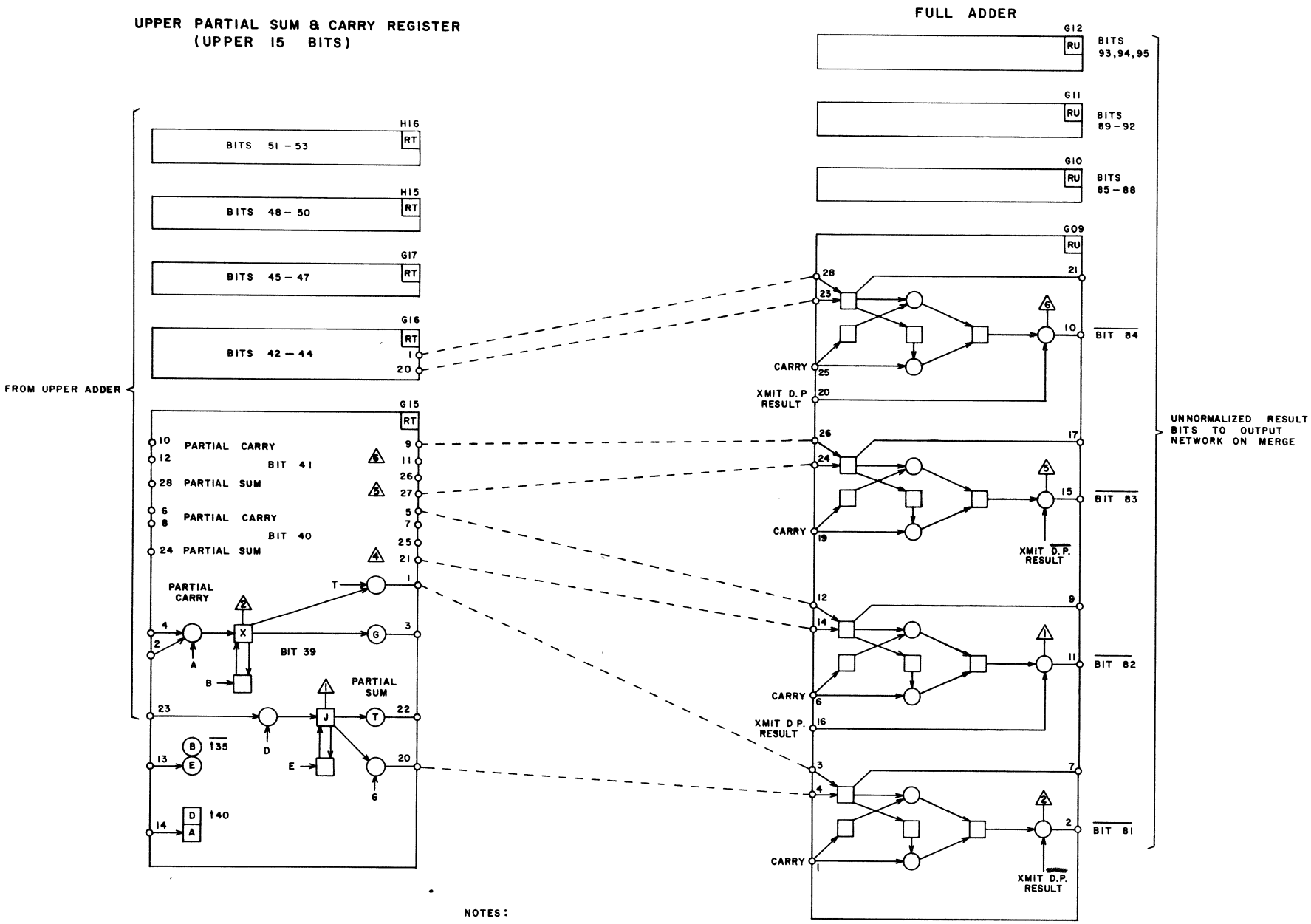

I25 TH
H23 CC
D25 CB
C27 MM
F27 TS
H25 CC
I24 CC
825-925
730-830
PARTIAL SUM & PARTIAL CARRY REGISTERS
UPPER
F29 TS
772 - Go merge
clear set every 195
H09 ML
I30 MV
I27 TD
F09 CC
D17 NJ
B CLEAR
E SET
A
D
C MERGE 19
F (TO LOWER ADDER)
continue clear for merge
1 gives clear set
F15 NJ
H21 CA
E29 TS
only 325-425
SEE ALSO PAGES 101 AND 103
T35 L08 MW
MULT I TRANSMIT RESULT
T1060
M08-26
LOWER
F08 QN
A13 NJ
exp. tests
I07 MP
ERROR
FROM CHASSIS 2
CO2-6
set: block x mission of coeff
F13 NJ
H20 QP
MERGE 18
(TO OUTPUT NETWORK AFTER ADD)
J27 TD
upper
I08 MP
DOUBLE PRECISION
H22 TC
DP
upper
16 OUTPUT BITS 48-96
J26 RR
I23 TC
SP
F28 TS
(REGISTER STAGES 24 & UP, SINCE THERE ARE 24 FULLY ADDED LOWER BITS)
delay L08-23 1 running
NOTE :
MULTIPLY 1 IS SHOWN,
MULTIPLY 2 IS SIMILAR.
CONTROL DATA CORPORATION
COMPUTER DIVISION
TITLE
CENTRAL PROCESSOR
MULTIPLY, COEFFICIENT
PARTIAL SUM & CARRY
REGISTER CONTROLS
PRODUCT 6601/04/13/14
SIZE C
DRAWING NO. 60119300
REV BT
SHEET 151
93

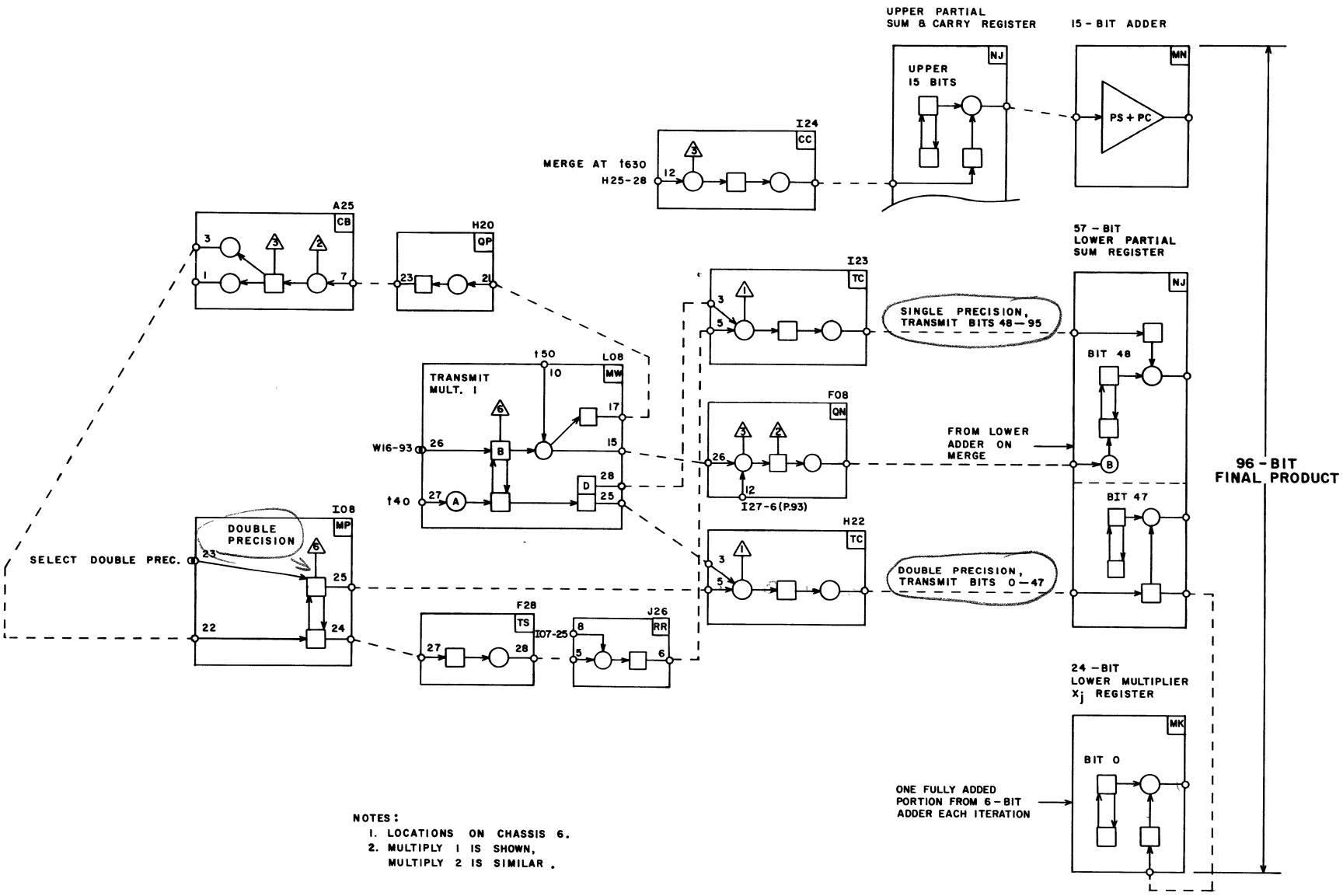


CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR MULTIPLY, COEFFICIENT UPPER FULL ADDER, 6 BITS	PRODUCT: 6601/04	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: BT
		SHEET: 152	95



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR MULTIPLY, COEFFICIENT LOWER FULL ADDER, 6 BITS	PRODUCT 6601/04 SIZE C · DRAWING NO. 60119300 · REV BT SHEET 153 · 97


UPPER PARTIAL SUM & CARRY REGISTER
(UPPER 15 BITS)
FULL ADDER
FROM UPPER ADDER
H16
BITS 51 – 53
RT
H15
BITS 48 – 50
RT
G17
BITS 45 – 47
RT
G16
BITS 42 – 44
RT
G15
RT
10 PARTIAL CARRY
12 BIT 41
28 PARTIAL SUM
6 PARTIAL CARRY
8 BIT 40
24 PARTIAL SUM
PARTIAL CARRY
BIT 39
PARTIAL SUM
G12
RU
BITS 93,94,95
G11
RU
BITS 89 – 92
G10
RU
BITS 85 – 88
G09
RU
CARRY
XMIT D.P RESULT
BIT 84
CARRY
XMIT D.P. RESULT
BIT 83
CARRY
XMIT D.P. RESULT
BIT 82
CARRY
XMIT D.P. RESULT
BIT 81
UNNORMALIZED RESULT BITS TO OUTPUT NETWORK ON MERGE
NOTES:
1. LOCATIONS ON CHASSIS 6.
2. MULTIPLY 1 IS SHOWN, MULTIPLY 2 IS SIMILAR.
only used during merge
CONTROL DATA CORPORATION
COMPUTER DIVISION
TITLE
CENTRAL PROCESSOR MULTIPLY, COEFFICIENT UPPER 15 BITS OF FINAL PRODUCT
PRODUCT 6601/04/13/14
SIZE C
DRAWING NO. 60119300
REV M
SHEET 154
99



NOTES:
1. LOCATIONS ON CHASSIS 6.
2. MULTIPLY 1 IS SHOWN, MULTIPLY 2 IS SIMILAR.

CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR MULTIPLY, COEFFICIENT 96 - BIT PRODUCT, DOUBLE PRECISION	PRODUCT: 6601/04/13/14	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: 07
			SHEET: 155 101

NORMALIZE

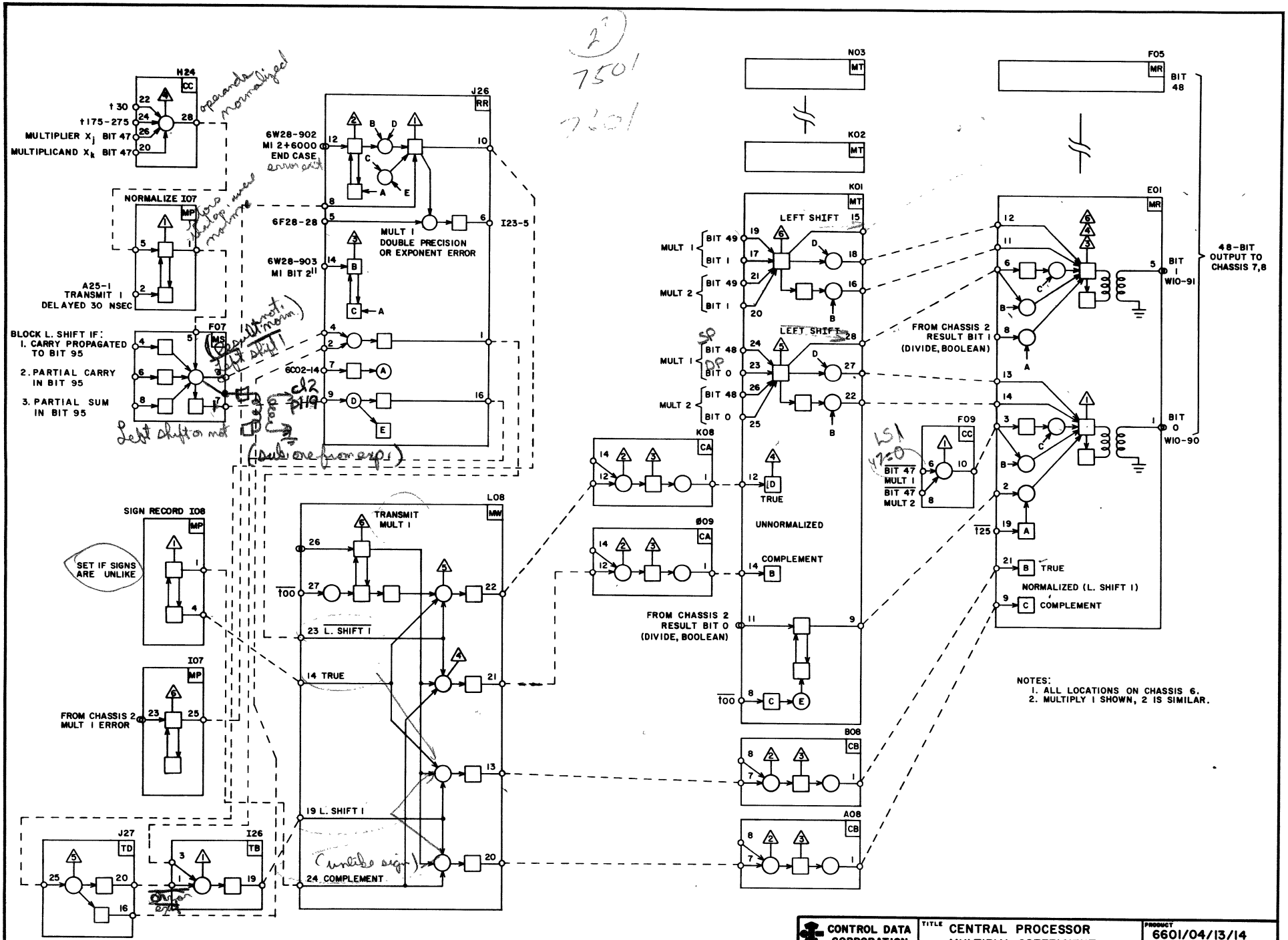
At the beginning of the multiply operation, the operands are examined to determine if both X_j and X_k have "1's" in bit 47. If both operands are normalized (bit 47 = "1"), the multiply unit will normalize the final product. The final product, if normalized, will have a "1" in bit 95; in an unnormalized product, bit 95 may be either "1" or "0".

The fact that both operands have "1's" in bit 47 automatically results in a "1" in either bit 94 or bit 95 of the product, depending upon the carries from lower positions. During merge, the product is examined to determine if a "1" has been propagated to bit 95. If not and normalize is indicated, the output network performs a 1-place left shift on the final product.

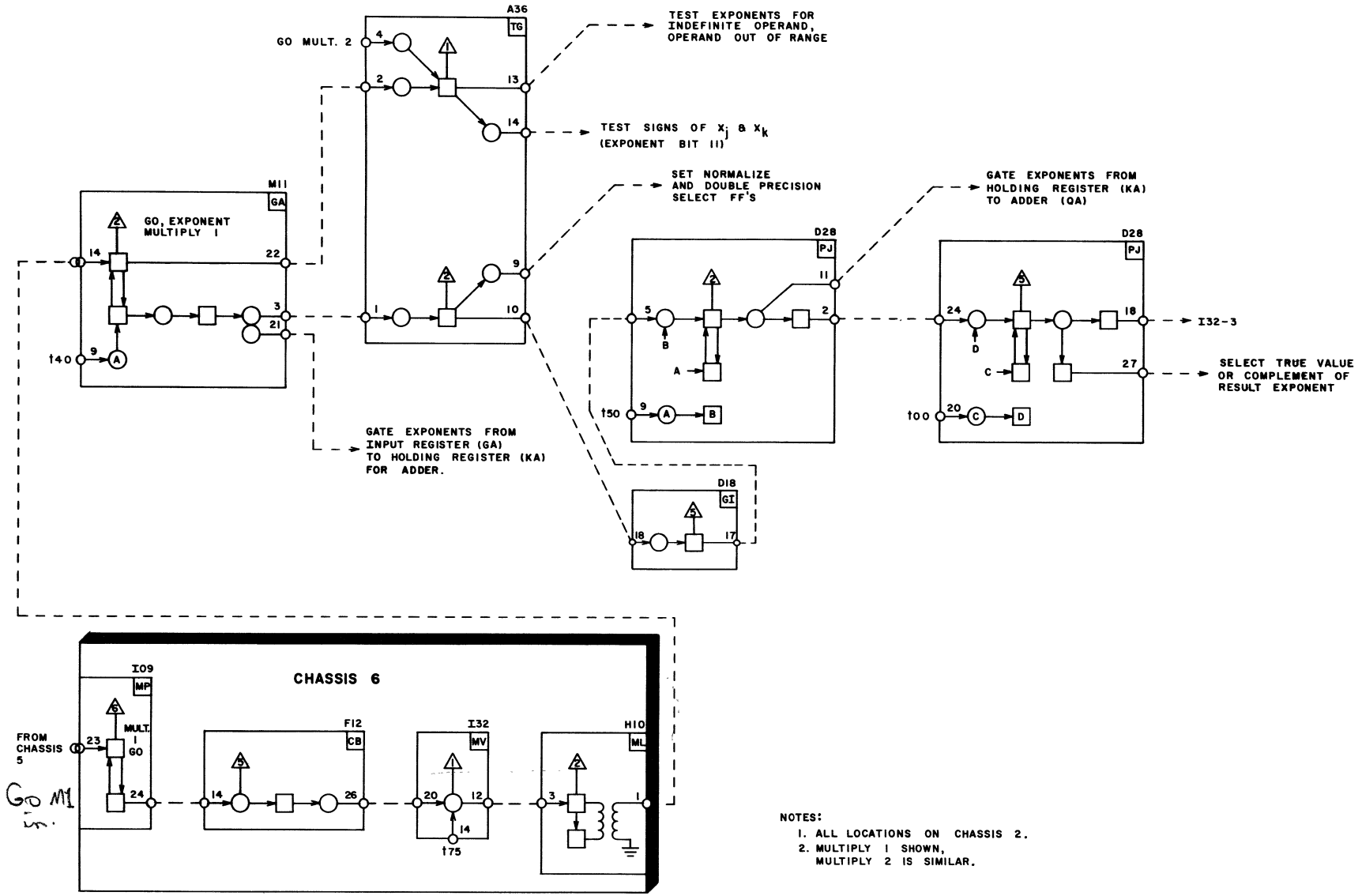
COMPLEMENT

When operands are received on chassis 6 and a multiply unit receives a Go signal, a comparison is performed on the coefficient signs (bits 59). This comparison determines the sign of the final product, by means of the Sign Record flip-flop. If the operand signs are the same (either both "1" or both "0"), the Sign Record flip-flop remains clear. This indicates a positive product with a sign bit of "0". If the operand signs are unlike, the Sign Record flip-flop becomes set, indicating a negative product with a sign bit of "1".

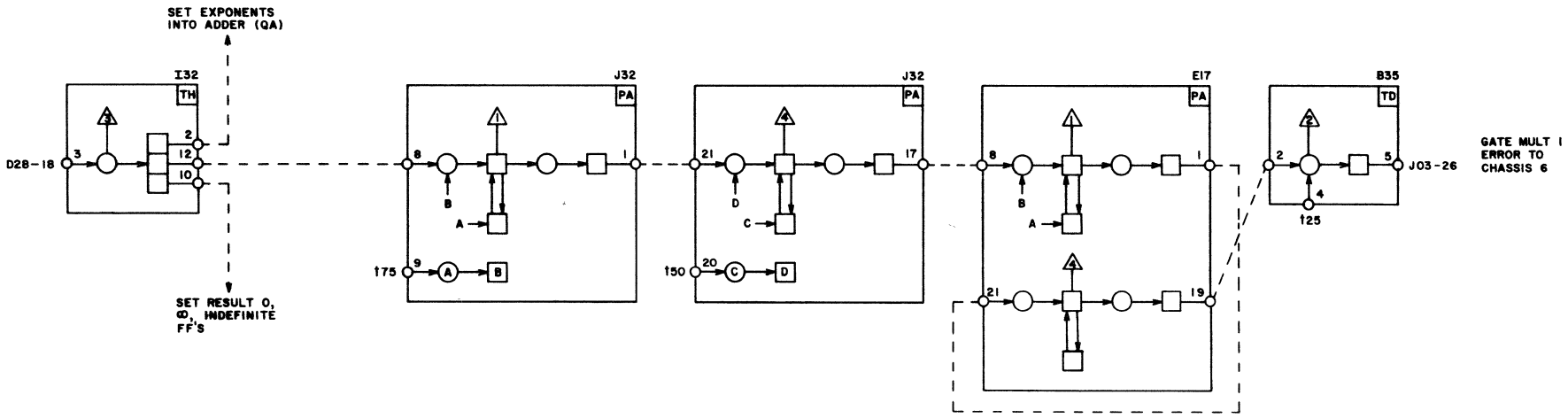
Negative numbers are represented in one's complement form. If the Sign Record flip-flop is set indicating a negative product, the output network transmits the complement of the actual result.



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE: CENTRAL PROCESSOR MULTIPLY, COEFFICIENT OUTPUT NETWORK NORMALIZE, COMPLEMENT	PRODUCT: 6601/04/13/14	
		SIZE: C	DRAWING NO.: 60119300 REV: BT
			SHEET: 156 103



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR MULTIPLY, EXPONENT TIMING CHAIN, CHASSIS 2	PRODUCT 6601	
		SIZE C	DRAWING NO. 60119300
		REV 8T	
		SHEET 157	105



NOTES:

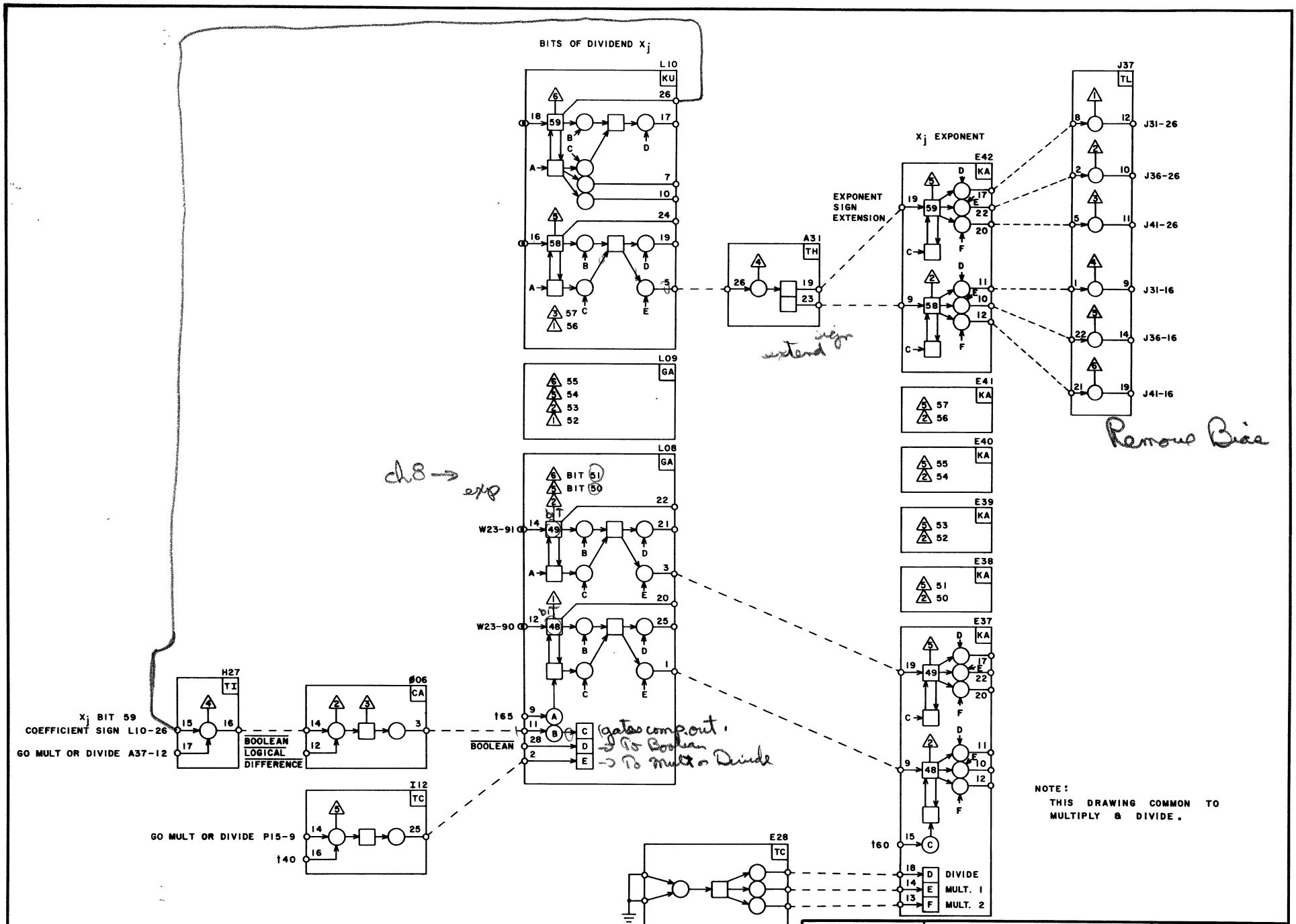
1. ALL LOCATIONS ON CHASSIS 2.
2. MULTIPLY 1 IS SHOWN, MULTIPLY 2 IS SIMILAR.

CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR MULTIPLY, EXPONENT TIMING CHAIN, CHASSIS 2	PRODUCT: 6601	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300
		REV: DH	
		SHEET: 158	107

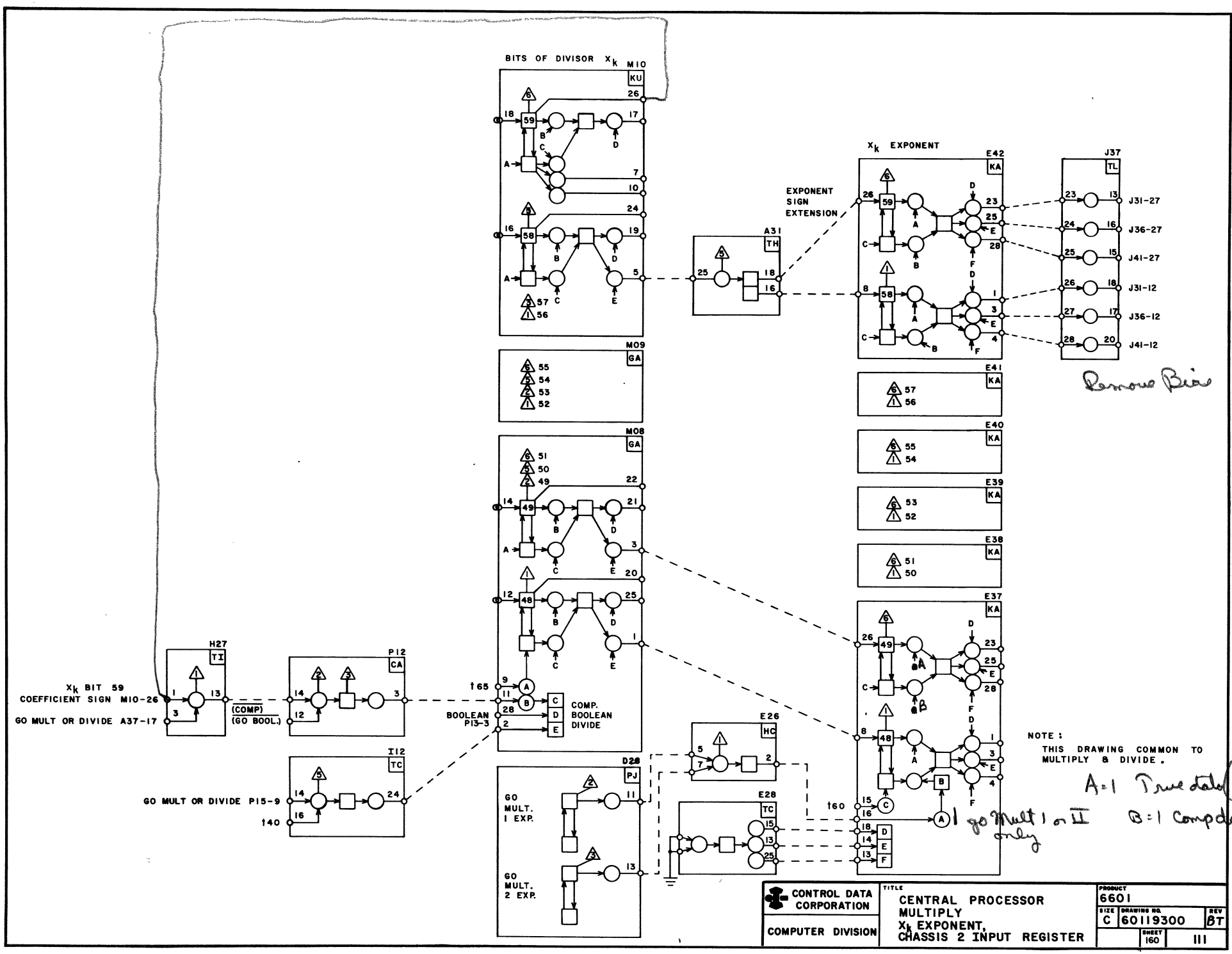
CHASSIS 2 INPUT REGISTERS

Chassis 2 contains the exponent arithmetic networks of Multiply 1 and Multiply 2 functional units, in addition to Divide and other logic. The input registers on chassis 2 are 60 bits in length. The exponent (bits 48 through 59) are sent to chassis 2 directly from the operating registers. The coefficient (bits 0 through 47) are sent first to chassis 6 and then re-transmitted to chassis 2.

The input registers on chassis 2 are shared by both Multiply 1 and 2, Divide, and Boolean. The unit receiving the Go signal samples the input registers and performs its respective operation. The two operands are held in the chassis 2 input registers for approximately 100 nanoseconds. If a multiplication is to be performed, the operation is timed so that the respective Multiply Exponent unit receives a Go signal and samples operand exponents while they are held in the chassis 2 input registers. The operand sign (bit 59) accompanies each operand to chassis 2. A sign bit of "1" indicates a negative operand. The input registers always contain the true values of the operands, but since negative numbers are handled in one's complement form, a sign bit of "1" gates the complement of the quantity into the exponent arithmetic unit.



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR MULTIPLY Xj EXPONENT, CHASSIS 2 INPUT REGISTER	PRODUCT 6601 SIZE C · DRAWING NO. 60119300 · REV BT SHEET 159 · 109	


BITS OF DIVISOR X_k M10
KU
EXPONENT SIGN EXTENSION
X_k EXPONENT
E42
KA
J37
TL
A31
TH
M09
GA
E41
E40
E39
E38
M08
E37
H27
TI
P12
CA
I12
TC
D28
PJ
E26
HC
E28
X_k BIT 59 COEFFICIENT SIGN M10-26
GO MULT OR DIVIDE A37-17
(COMP)
(GO BOOL.)
t65
BOOLEAN P13-3
COMP. BOOLEAN DIVIDE
GO MULT OR DIVIDE P15-9
t40
GO MULT. 1 EXP.
GO MULT. 2 EXP.
t60
J31-27
J36-27
J41-27
J31-12
J36-12
J41-12
NOTE :
THIS DRAWING COMMON TO MULTIPLY & DIVIDE.
A=1 True data (mult only)
B=1 Comp data
go Mult I or II only
CONTROL DATA CORPORATION
COMPUTER DIVISION
TITLE
CENTRAL PROCESSOR
MULTIPLY
X_k EXPONENT,
CHASSIS 2 INPUT REGISTER
PRODUCT 6601
SIZE C
DRAWING NO. 60119300
REV 8T
SHEET 160
111

EXPONENT ADDITION

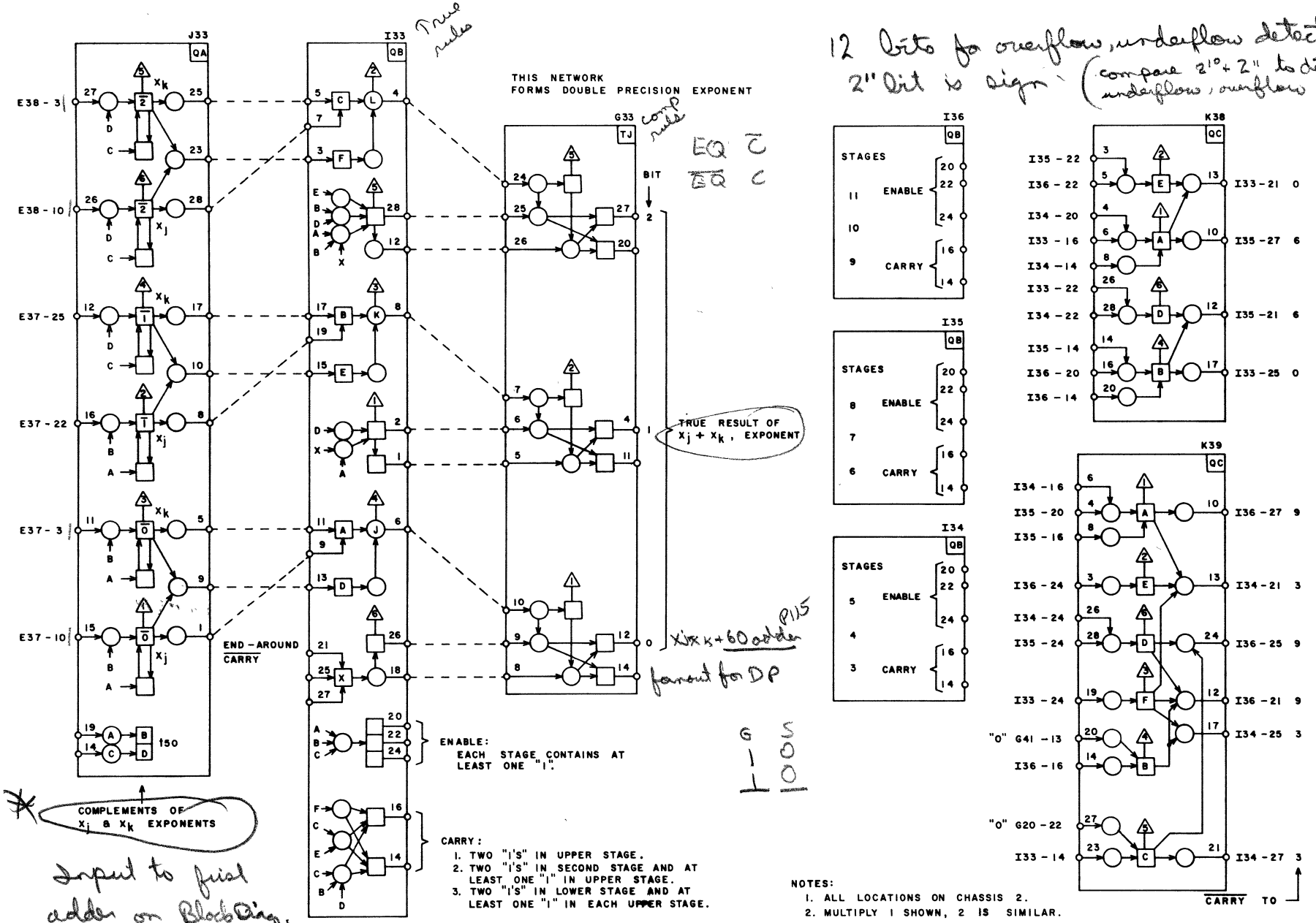
The exponent arithmetic logic automatically performs two additions at the beginning of each multiply operation. The two operand exponents are added to each other, and the result of this is added to 60_8 (48_{10}). These two results are the basic double-precision and single-precision product exponents, respectively. Later in the operation, the control logic selects one of these and determines whether further modification such as normalizing should be done. The adders which perform the exponent addition are one's complement; if complementary numbers are added, the result is all "0's" (positive zero).

DOUBLE-PRECISION EXPONENT

This exponent is used if the double-precision product is selected. The coefficient of this product is the lower 48 bits of the 96-bit product; the upper 48 bits are discarded. The double-precision exponent is produced by direct addition of the operand exponents.

SINGLE-PRECISION EXPONENT

This exponent is used with the single-precision product. The coefficient is the upper 48 bits of the 96-bit product; the lower 48 bits are discarded. The single-precision exponent is produced by adding 60_8 (48_{10}) to the double-precision exponent.



12 bits for overflow, underflow detection
2^11 bit is sign (compare 2^10 + 2^11 to determine underflow, overflow)

EQ C

EQ C

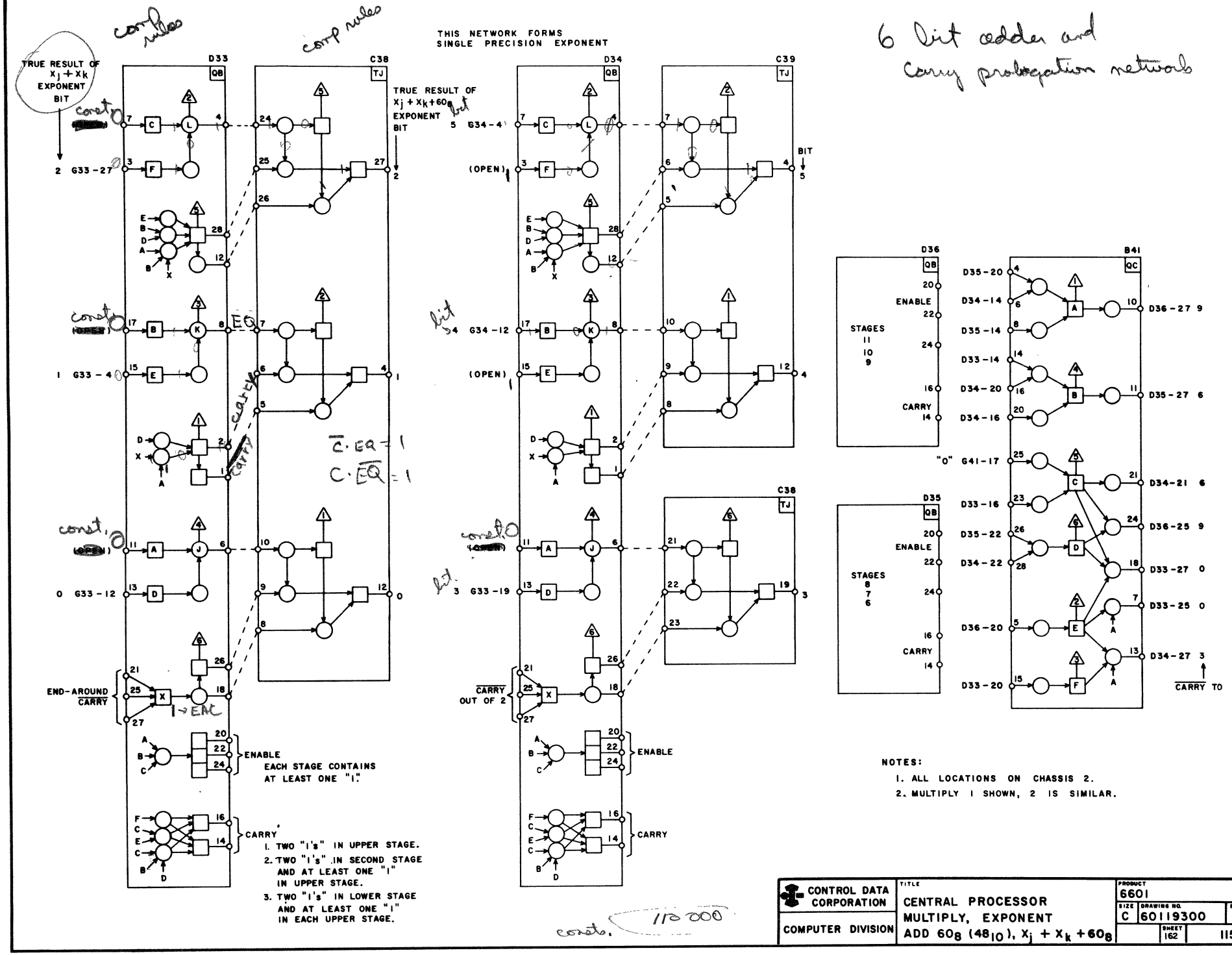
$X_j \times X_k$ + 60 adder PIDS

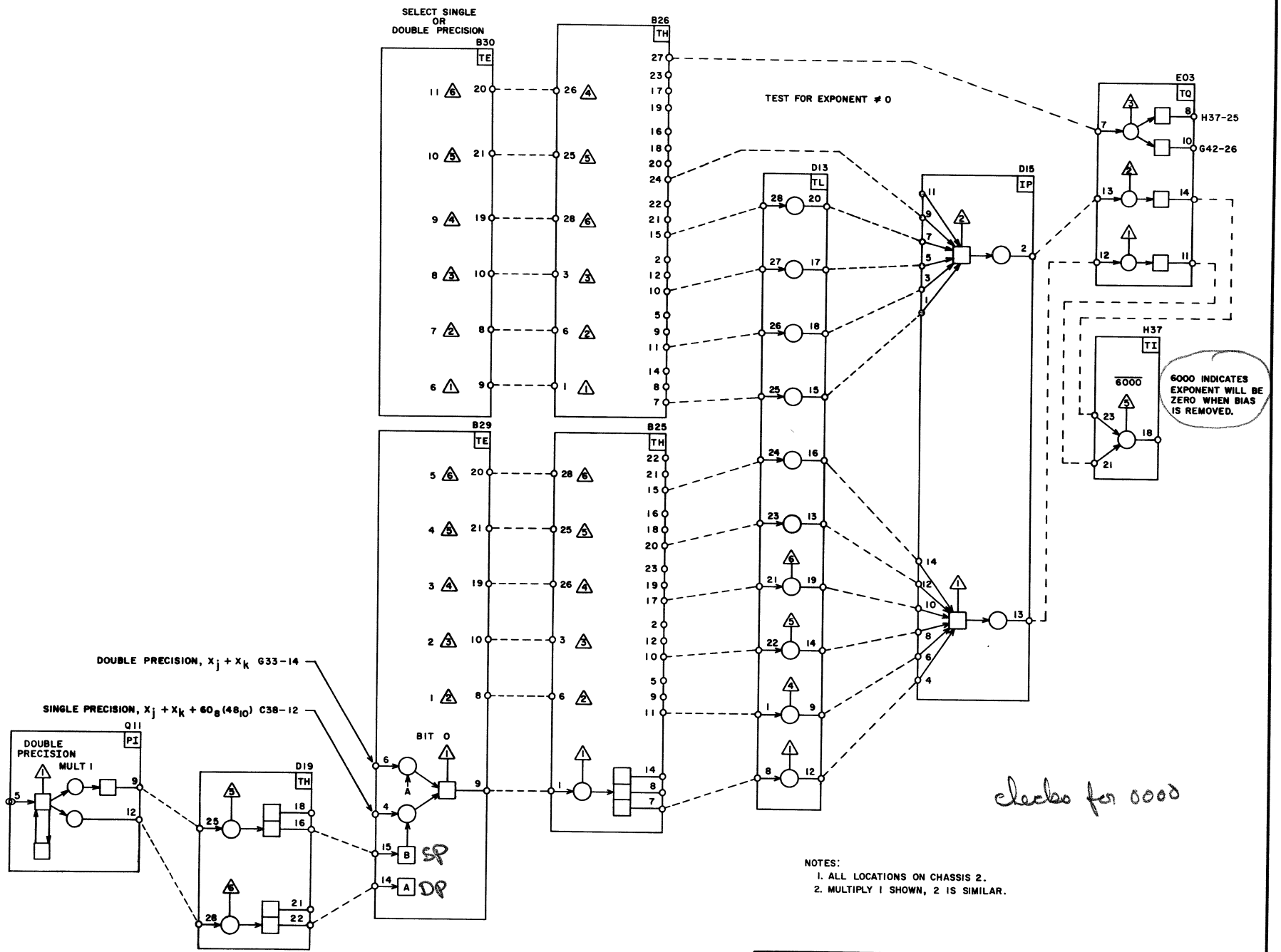
fanout for DP

Input to first adder on Block Diag.

NOTES:
1. ALL LOCATIONS ON CHASSIS 2.
2. MULTIPLY 1 SHOWN, 2 IS SIMILAR.

CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR MULTIPLY, EXPONENT ADD, $X_j + X_k$	PRODUCT: 6601	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: C
		SHEET: 161	113


6 bit adder and
carry probagation network
THIS NETWORK FORMS
SINGLE PRECISION EXPONENT
TRUE RESULT OF
$X_j + X_k$
EXPONENT
BIT
TRUE RESULT OF
$X_j + X_k + 60_8$
EXPONENT
BIT
D33
QB
C38
TJ
D34
QB
C39
TJ
D36
QB
D35
QB
B41
QC
2 G33-27
1 G33-4
0 G33-12
5 G34-4
4 G34-12
3 G33-19
(OPEN)
"0" G41-17
const 0
const 1
EQ
carry
$\overline{C} \cdot EQ = 1$
$C \cdot \overline{EQ} = 1$
END-AROUND
CARRY
1 → END
CARRY
OUT OF 2
ENABLE
EACH STAGE CONTAINS
AT LEAST ONE "1"
CARRY
1. TWO "1's" IN UPPER STAGE.
2. TWO "1's" IN SECOND STAGE
AND AT LEAST ONE "1"
IN UPPER STAGE.
3. TWO "1's" IN LOWER STAGE
AND AT LEAST ONE "1"
IN EACH UPPER STAGE.
STAGES
11
10
9
STAGES
8
7
6
ENABLE
CARRY
D35-20
D34-14
D35-14
D33-14
D34-20
D34-16
D33-16
D35-22
D34-22
D36-20
D33-20
D36-27 9
D35-27 6
D34-21 6
D36-25 9
D33-27 0
D33-25 0
D34-27 3
CARRY TO
NOTES:
1. ALL LOCATIONS ON CHASSIS 2.
2. MULTIPLY 1 SHOWN, 2 IS SIMILAR.
CONTROL DATA CORPORATION
COMPUTER DIVISION
TITLE
CENTRAL PROCESSOR
MULTIPLY, EXPONENT
ADD 60_8 (48_{10}), $X_j + X_k + 60_8$
PRODUCT
6601
SIZE C
DRAWING NO. 60119300
REV C
SHEET 162
115

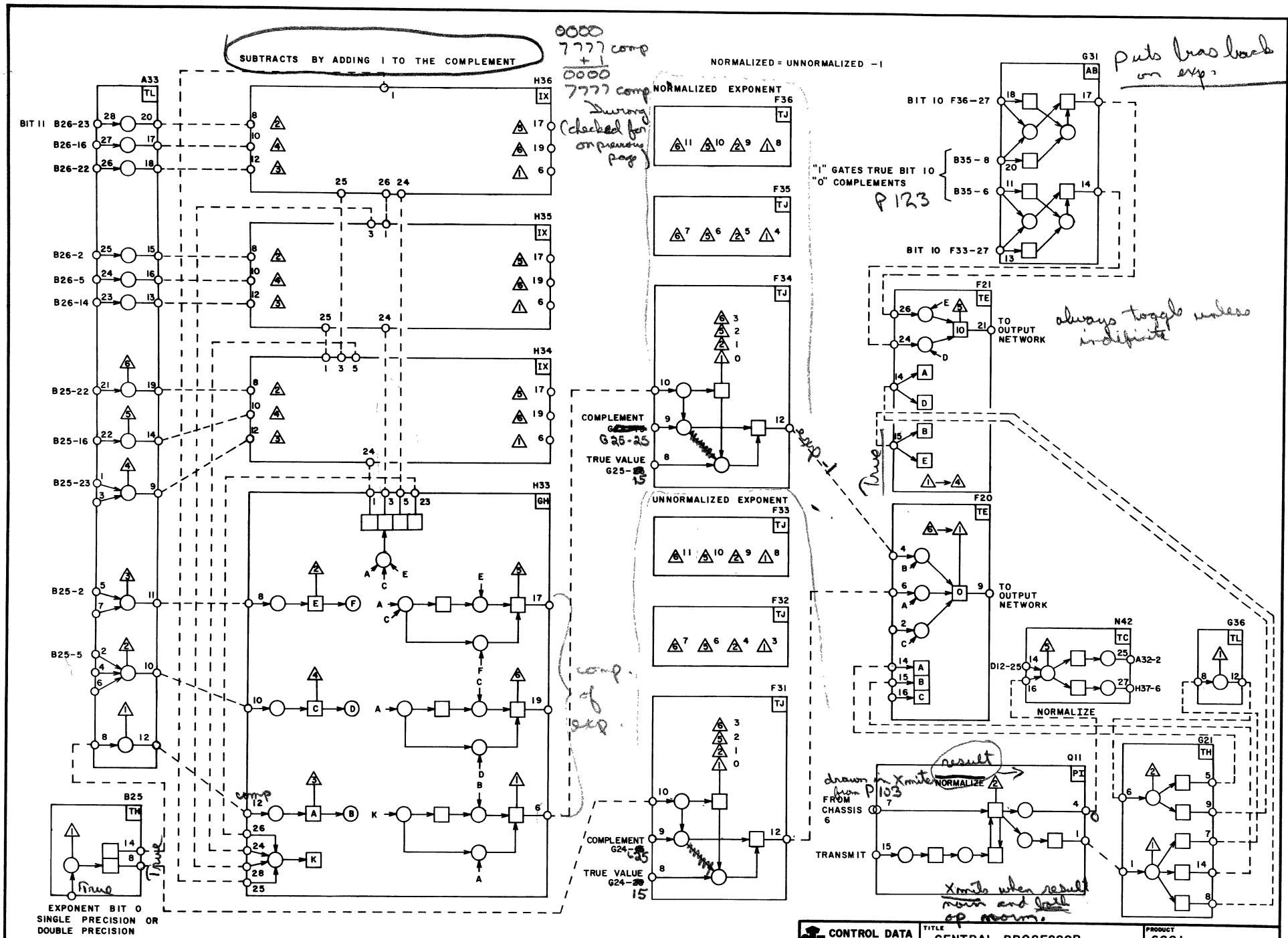


clocks for 0000

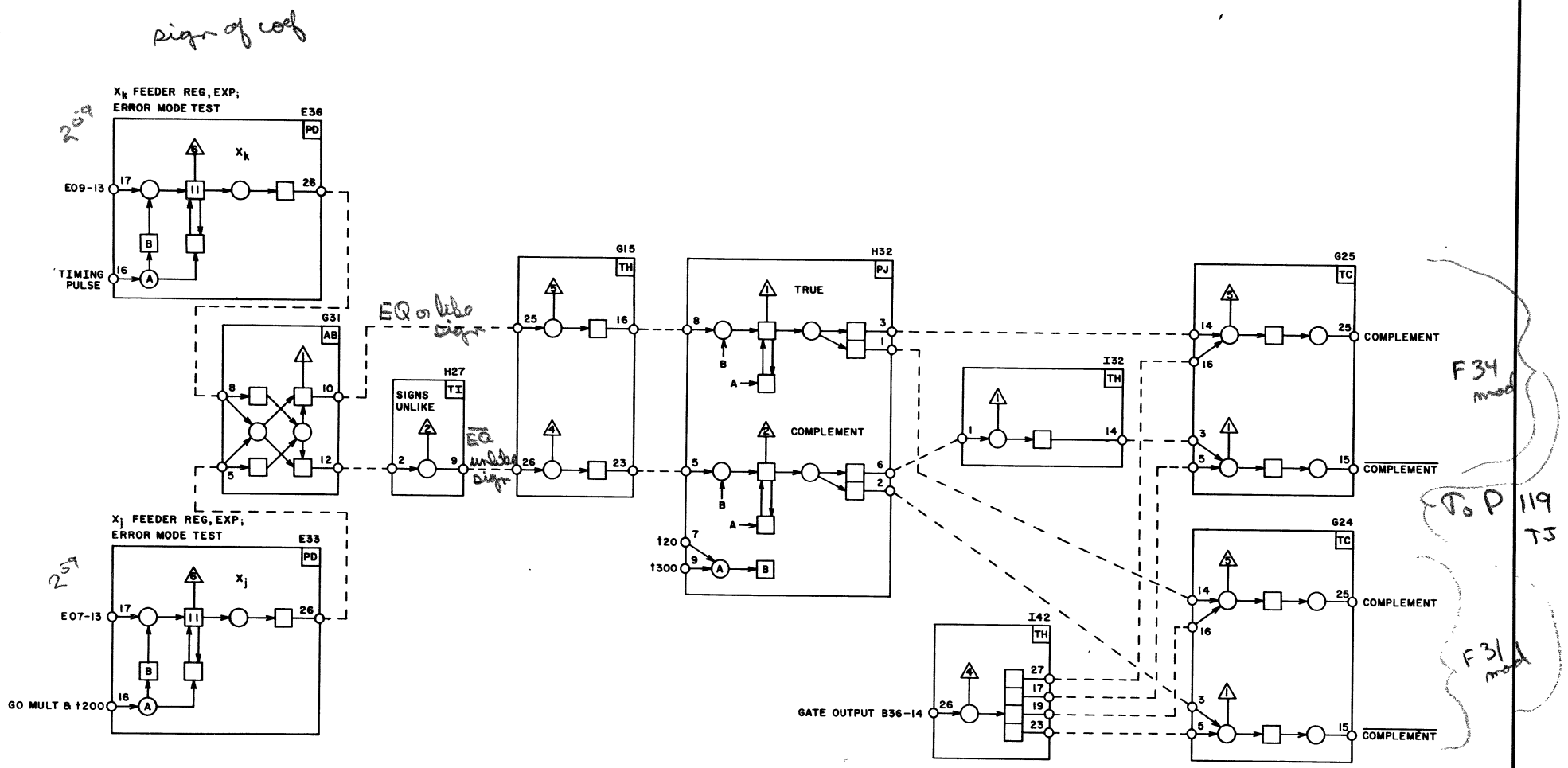
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR MULTIPLY, EXPONENT SINGLE OR DOUBLE PRECISION	PRODUCT: 6601	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: BT
		SHEET: 163	117

The selected exponent (single or double precision) is made available in both normalized and unnormalized form. In the process of normalizing, the coefficient is shifted one place to the left and the exponent is reduced by one. The network which reduces the exponent effectively does this by adding one to the complement.

Both the normalized and unnormalized values of the exponent are made available at the fan-in to the output network. The fan-in is gated by a signal from chassis 6 which sets the Normalize flip-flop. The decision of whether to take the true value or the complement of the exponent is based on the comparison of the signs (exponent bit 11, operand bit 59) of the original operands X_j and X_k. If the signs are the same, the true exponent is used; if the signs are different, the exponent is complemented.



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR MULTIPLY, EXPONENT NORMALIZE, COMPLEMENT	PRODUCT 6601	
		SIZE C	DRAWING NO. 60119300
		REV BT	
		SHEET 164	119



NOTES:

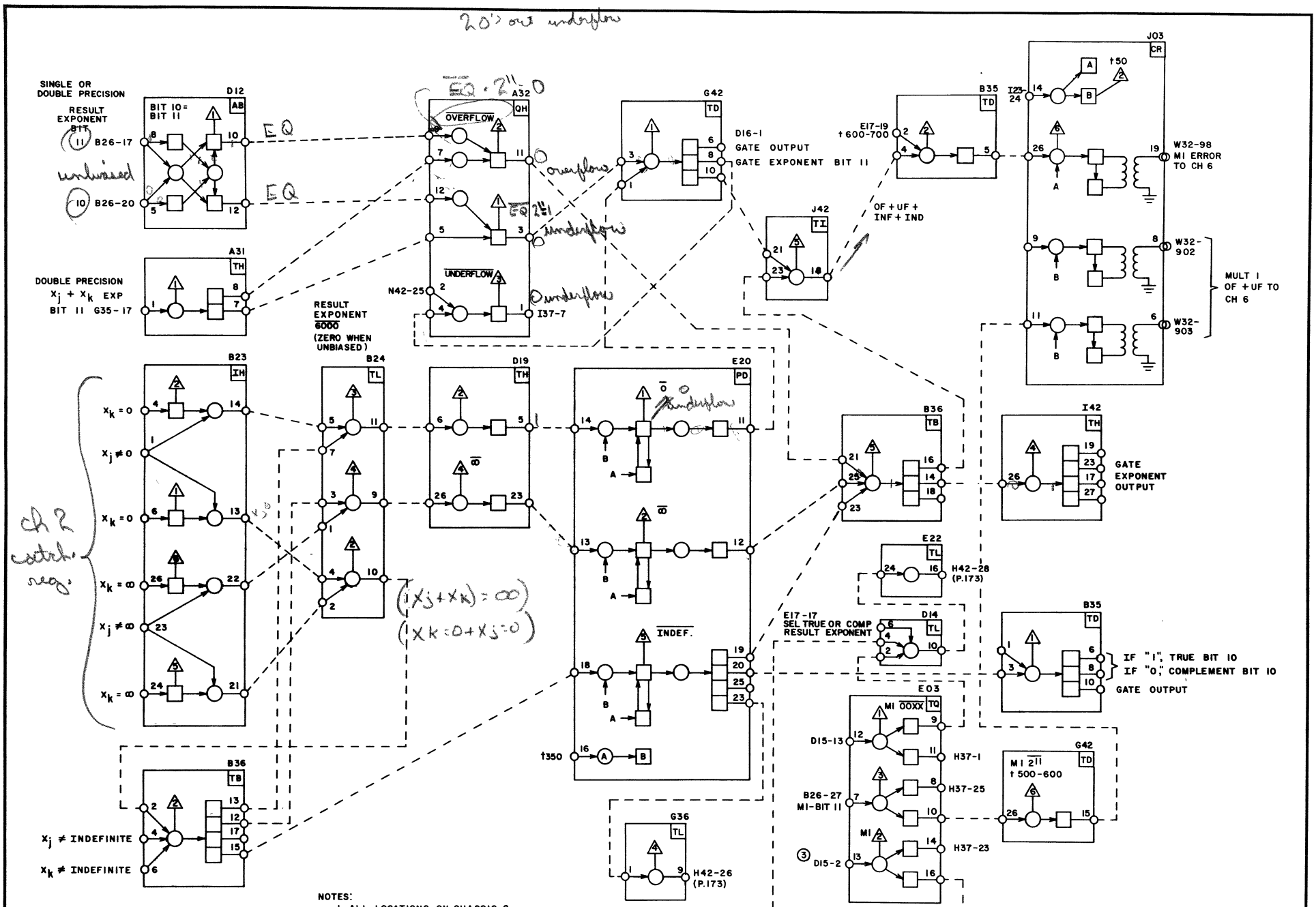
1. ALL LOCATIONS ON CHASSIS 2.
2. MULTIPLY 1 IS SHOWN, MULTIPLY 2 IS SIMILAR.

CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR MULTIPLY, EXPONENT SELECT COMPLEMENT, TRUE VALUE	PRODUCT: 6601	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: BT
			SHEET: 165 121

TEST RESULT

Several tests are performed in the exponent arithmetic to determine if the product is valid. The initial operands are tested to see if either X_j or X_k is indefinite, infinite, or zero. The final product exponent is tested to determine if it is equal to 2000; it would then become zero when unbiased.

The results of the tests are held until the end of the operation and are then used to condition the gating of the final product. An indefinite result is packed with an exponent of 1777_8 and a zero coefficient. A result of infinity is packed with an exponent of 3777_8 and a zero coefficient. A result of zero is packed with a zero exponent and a zero coefficient. All adjustments of the exponent are performed on chassis 2 and if the result is such that the coefficient should be held to zero, the Error signals are sent to chassis 6 to accomplish this.

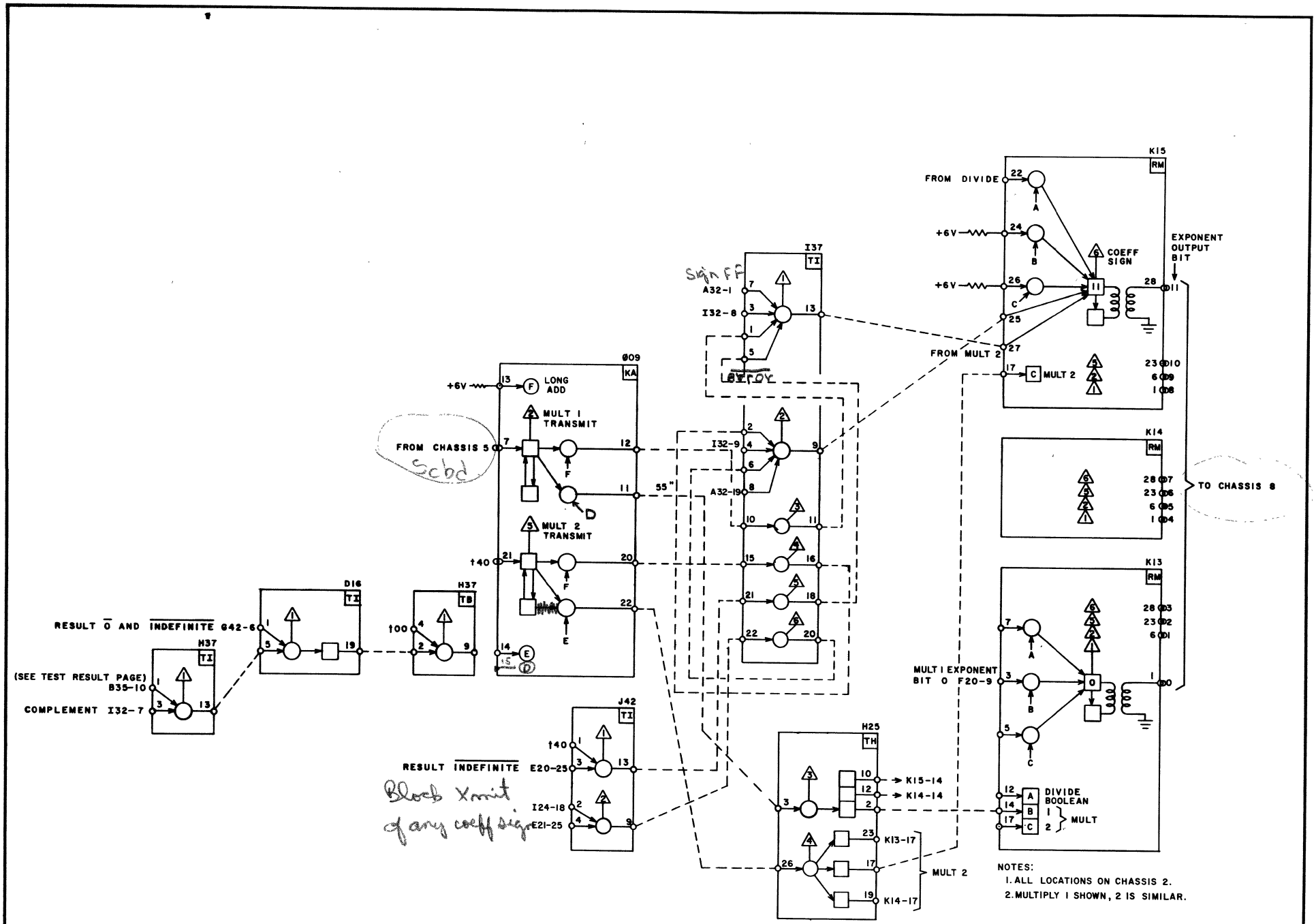


NOTES:
1. ALL LOCATIONS ON CHASSIS 2.
2. MULTIPLY 1 SHOWN, MULTIPLY 2 IS SIMILAR.
3. E03-13 EXPONENT CHECK = $\overline{2^6 \cdot 2^7 \cdot 2^8 \cdot 2^9 \cdot 2^{10}} \cdot (\overline{X_j + X_k = 0})$

CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE: CENTRAL PROCESSOR MULTIPLY, EXPONENT TEST RESULT

PRODUCT 6601
SIZE C | DRAWING NO. 60119300 | REV 8T
SHEET 166 | 123



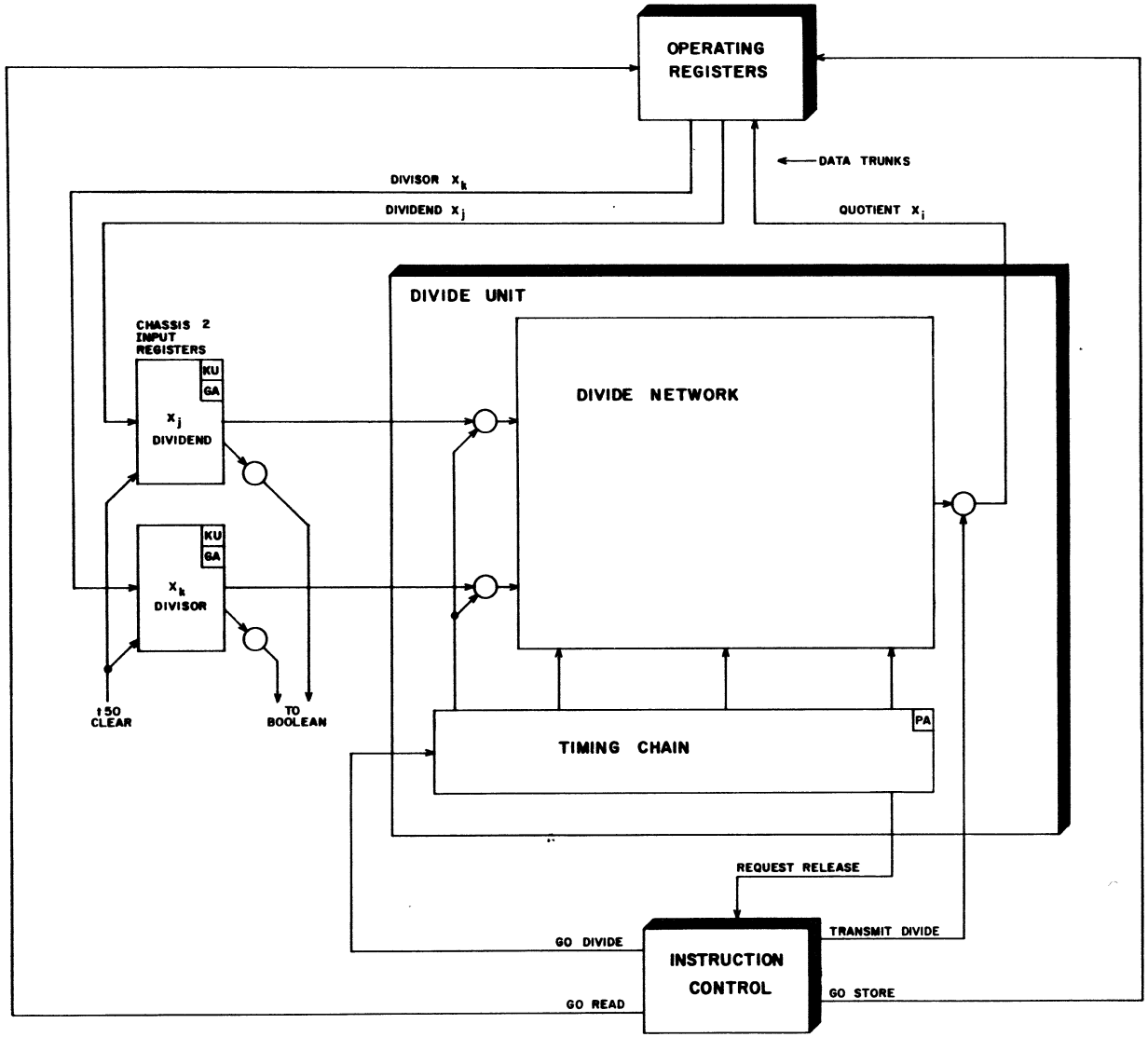
CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR MULTIPLY, EXPONENT OUTPUT NETWORK	PRODUCT 6601/04/13/14	
		SIZE C	DRAWING NO. 60119300 REV BT
		SHEET 167	125

Division in the CONTROL DATA* 6601 Central Computer is performed by the Divide Unit located on chassis 2. This unit contains all necessary registers, adders, and logic elements to form the 60-bit, floating-point quotient of two 60-bit, floating-point operands. The time required for a division is 2.9 microseconds (29 minor cycles).

A divide instruction directs the unit to divide the dividend X_j by the divisor X_k and send the quotient to X_i. The coefficients are divided by the method of repeated subtractions. The exponents are subtracted according to the rules of exponential arithmetic.

RESTRICTION:
The coefficient arithmetic portion of the divide unit is not capable of producing a quotient larger than $1.77\ --\ 7_8$. It is therefore a requirement in all cases that the coefficient of the dividend X_j must be less than two times the coefficient of the divisor X_k. The coefficient of the divisor X_k may exceed the coefficient of the dividend X_j without limit, so long as neither operand is infinite or zero.

It is important to note that if the divisor is normalized, the above restriction never applies. In addition, if both operands are normalized, the divide unit produces a normalized quotient.

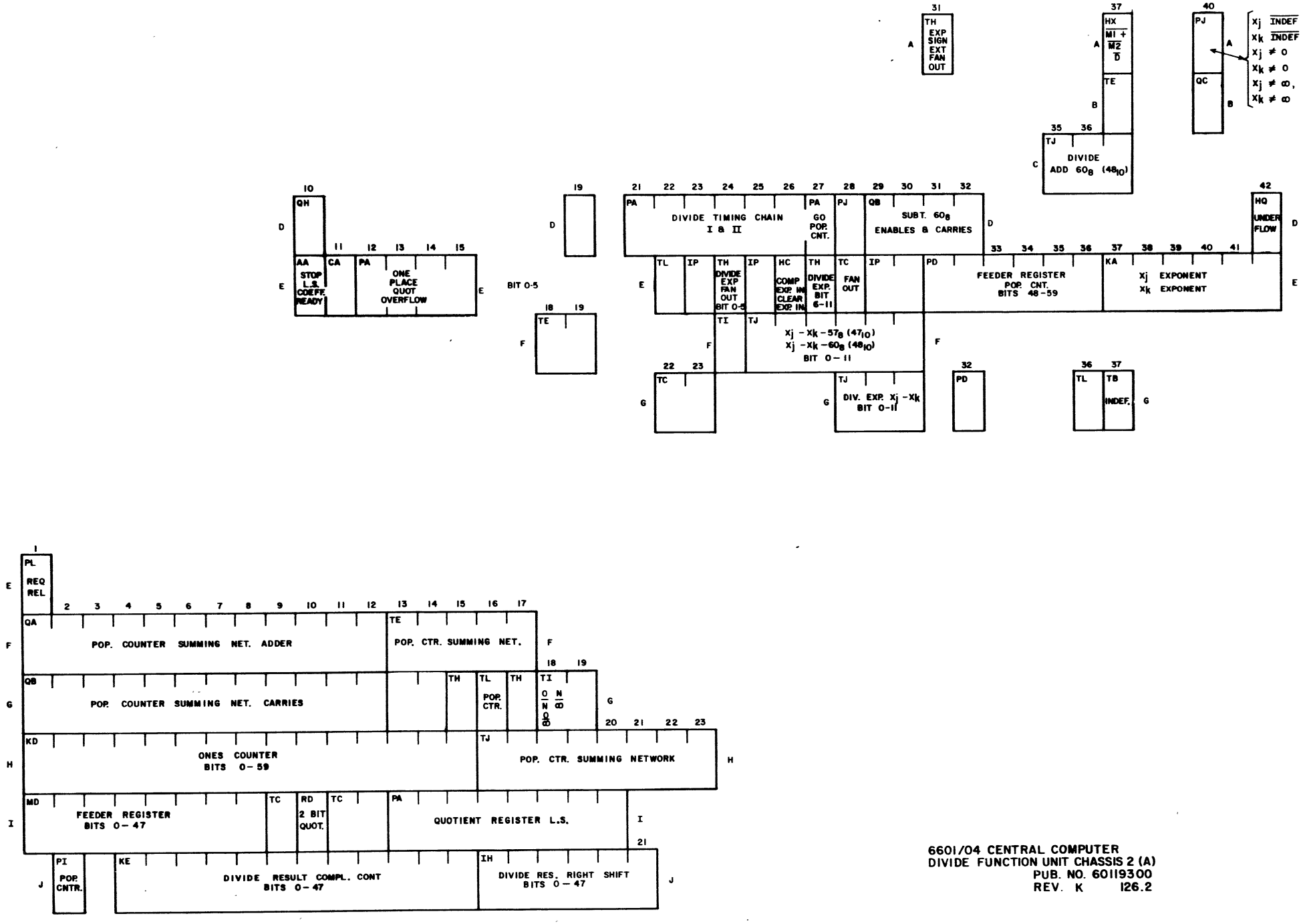


NOTE:

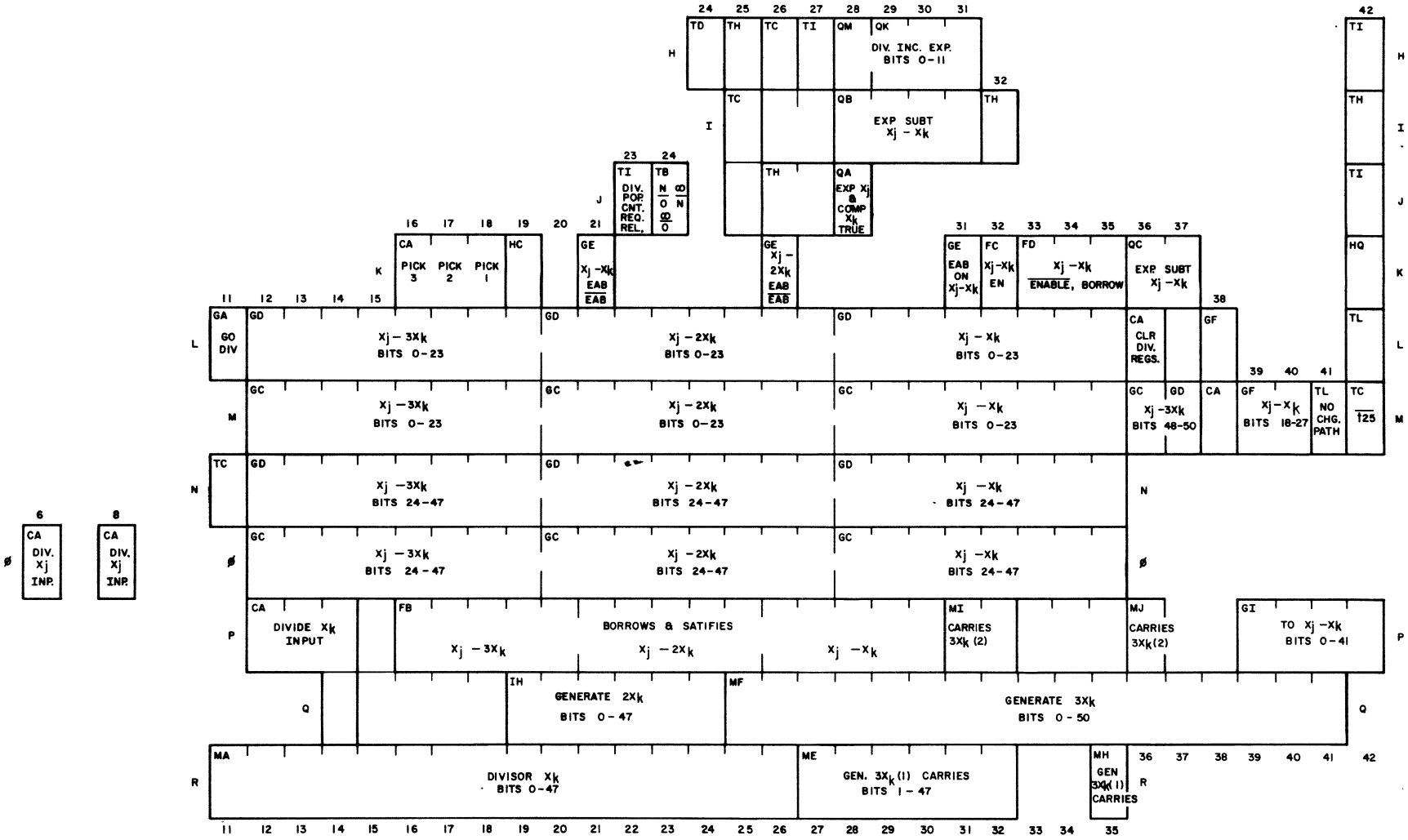
DIVIDE IS FRACTIONAL, ONLY.

IN ORDER TO WORK PROPERLY, $\frac{X_j \text{ COEFF.}}{X_k \text{ COEFF.}}$ MUST BE LESS THAN 2

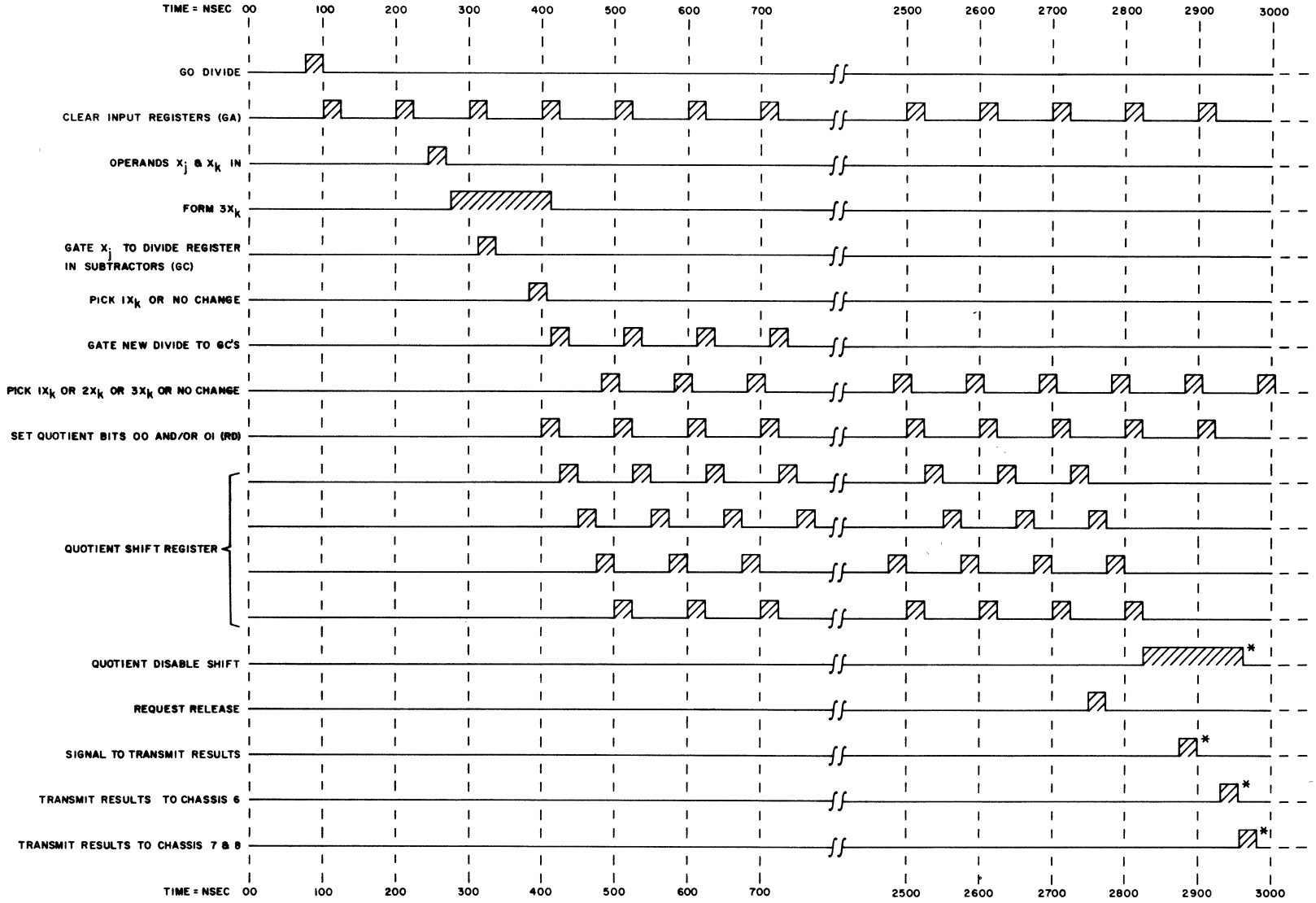
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR DIVIDE BLOCK DIAGRAM	PRODUCT: 6601/04	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: K
			SHEET: 168 126.1



6601/04 CENTRAL COMPUTER
DIVIDE FUNCTION UNIT CHASSIS 2 (A)
PUB. NO. 60119300
REV. K 126.2



6601/04 CENTRAL COMPUTER
DIVIDE FUNCTION UNIT
CHASSIS 2 (B)
PUB. NO. 60119300
REV. K 127



* EARLIEST POSSIBLE TIME — NO RESULT REGISTER CONFLICT

CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE
CENTRAL PROCESSOR
DIVIDE UNIT
TIMING CHART

PRODUCT 6601
SIZE C | DRAWING NO. 60119300 | REV D
SHEET 217 | PAGE 128.1

Coefficient division of the two operands is performed using three subtraction networks. The original dividend X_j is gated directly from the chassis 2 input register into each of the 3 subtractors. The divisor X_k is gated into a holding register from which three values are formed: X_k times one, times two, and times three. These quantities are gated into the 3 subtraction networks and are held during the remainder of the operation.

The quotient is formed 2 bits at a time by trial subtractions of the 3 multiples of X_k from the dividend. The largest multiple which subtracts without causing an end-around borrow determines the selection of the 2 bits. If end-around borrows occur in all three subtractions, the two bits are zero.

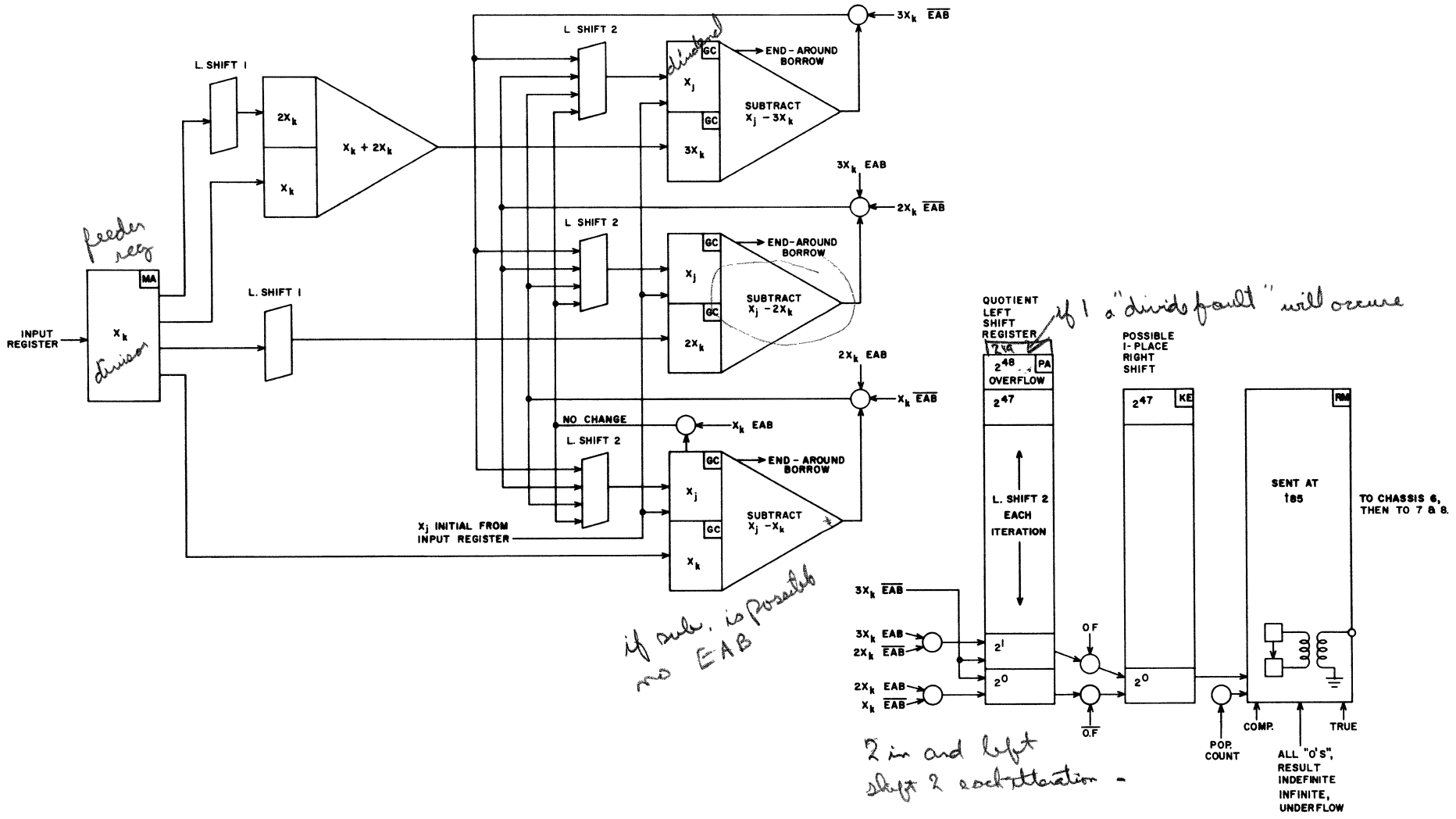
The largest usable multiple of X_k is subtracted from the dividend during each iteration. This new quantity is then shifted left 2 places and re-inserted into the dividend registers of the 3 subtraction networks. On the first iteration, the original dividend X_j is held in the subtraction networks.

The coefficient arithmetic network forms a 50-bit quotient. Twenty-four iterations are required to process the 48-bit dividend; however, 2-bit quotients are selected both before the first iteration and after the 24th iteration.

The high-order bit (bit 49) of the final quotient is always "0" and is never used. This is due to the fractional restriction of the coefficient arithmetic; the 2-bit quotient picked on the first iteration is always either 00 or 01. Quotient bit 48 is tested at the output network, and if a "1", a 1-place right shift is performed.

Xk into Xj

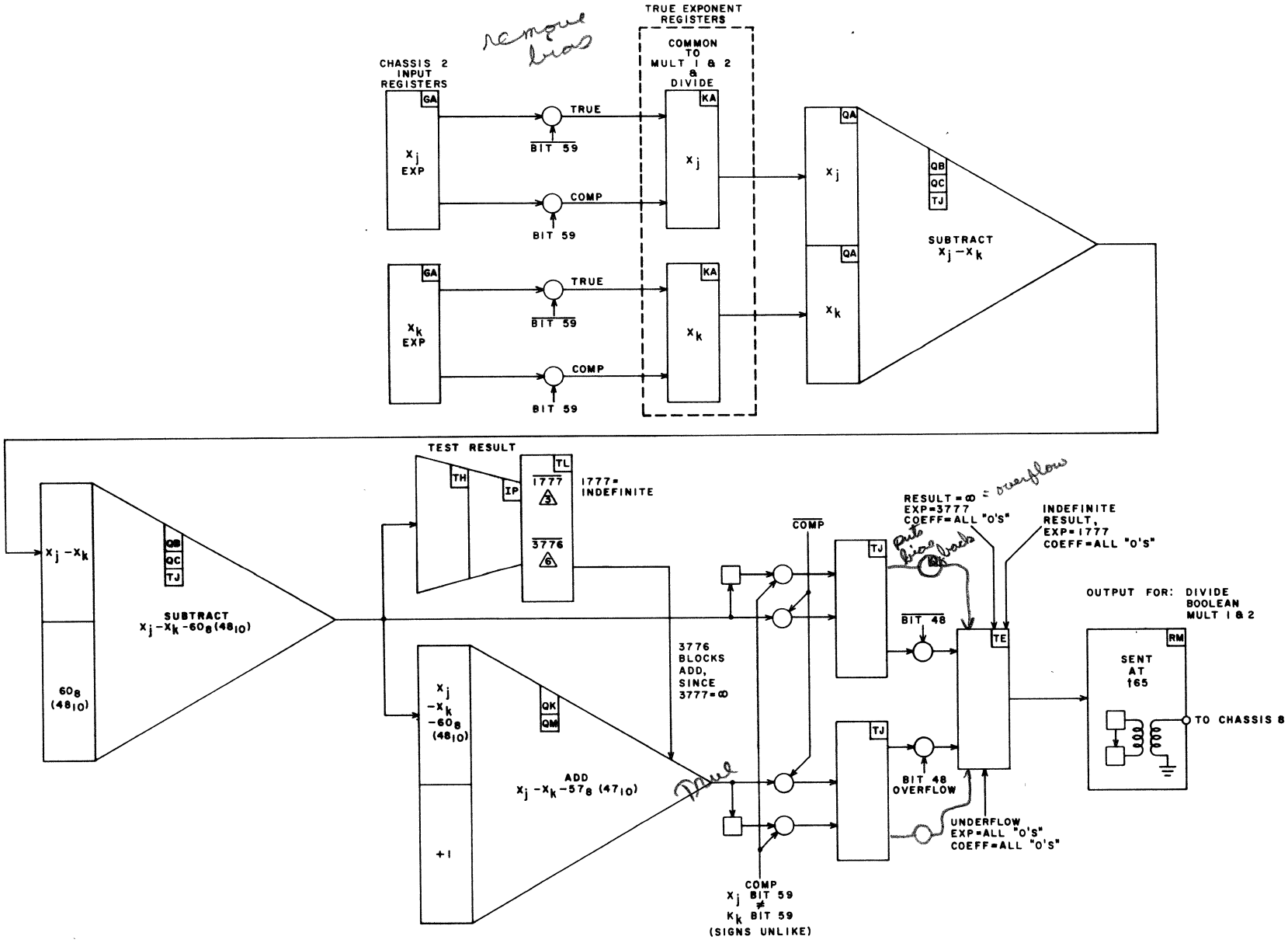
3Xj feeders with 50 bits each



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR DIVIDE, COEFFICIENT BLOCK DIAGRAM	PRODUCT 6601 SIZE C · DRAWING NO. 60119300 · REV C SHEET 169	129

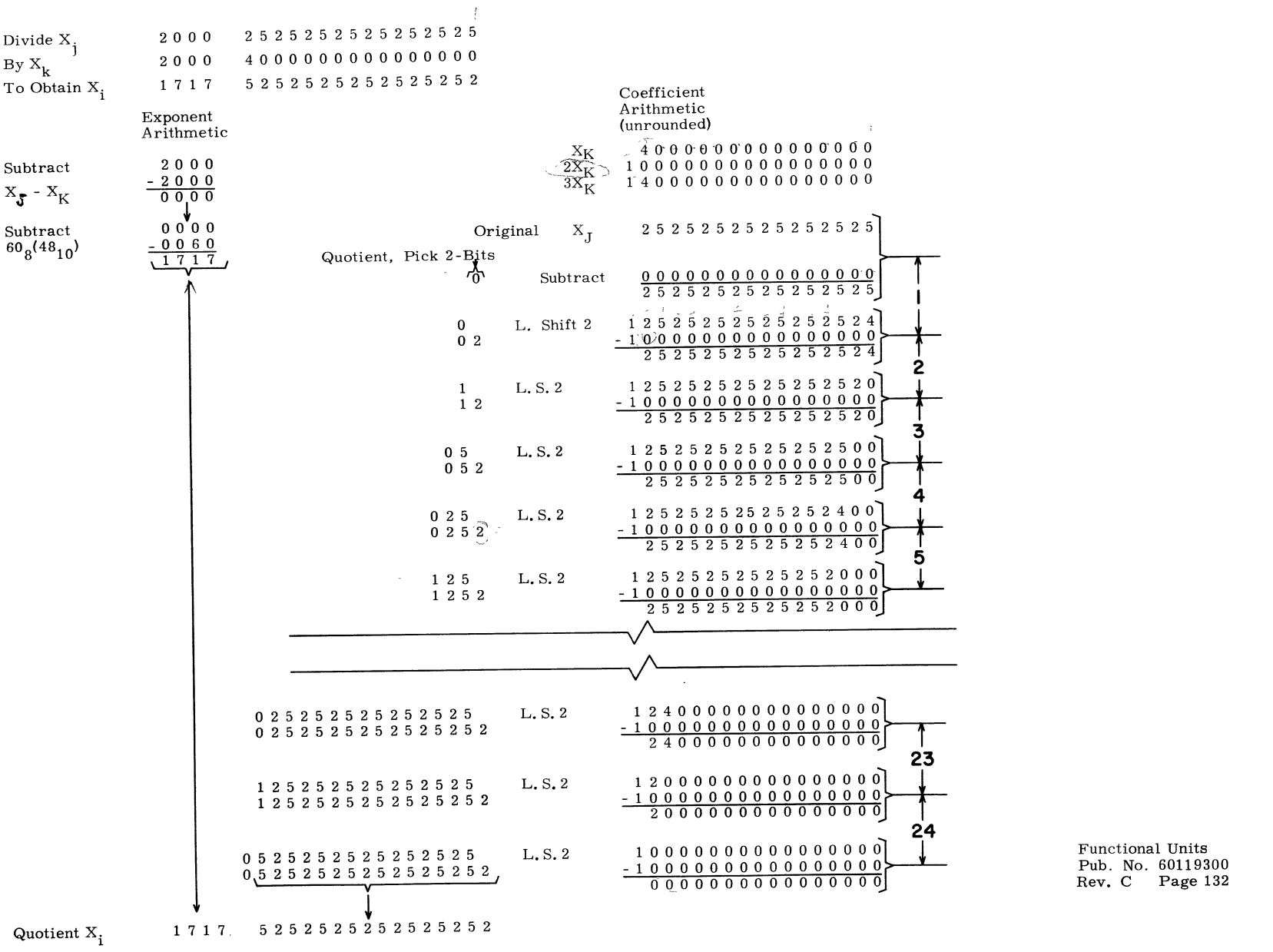
The exponents of the divide operands are handled according to standard rules of exponential numbers. For the basic result exponent, the exponent of the divisor X_k is subtracted from the exponent of the dividend X_j. Next, the quantity 60_8 (48_{10}) is subtracted from the result of X_j minus X_k. This has the effect of moving the binary point of the coefficient 48 places to the right. The quotient produced by the coefficient arithmetic is a fraction. It is therefore necessary to shift the binary point to the right in this manner to make the floating-point quotient of the divide unit correspond with the standard floating-point format of the other functional units.

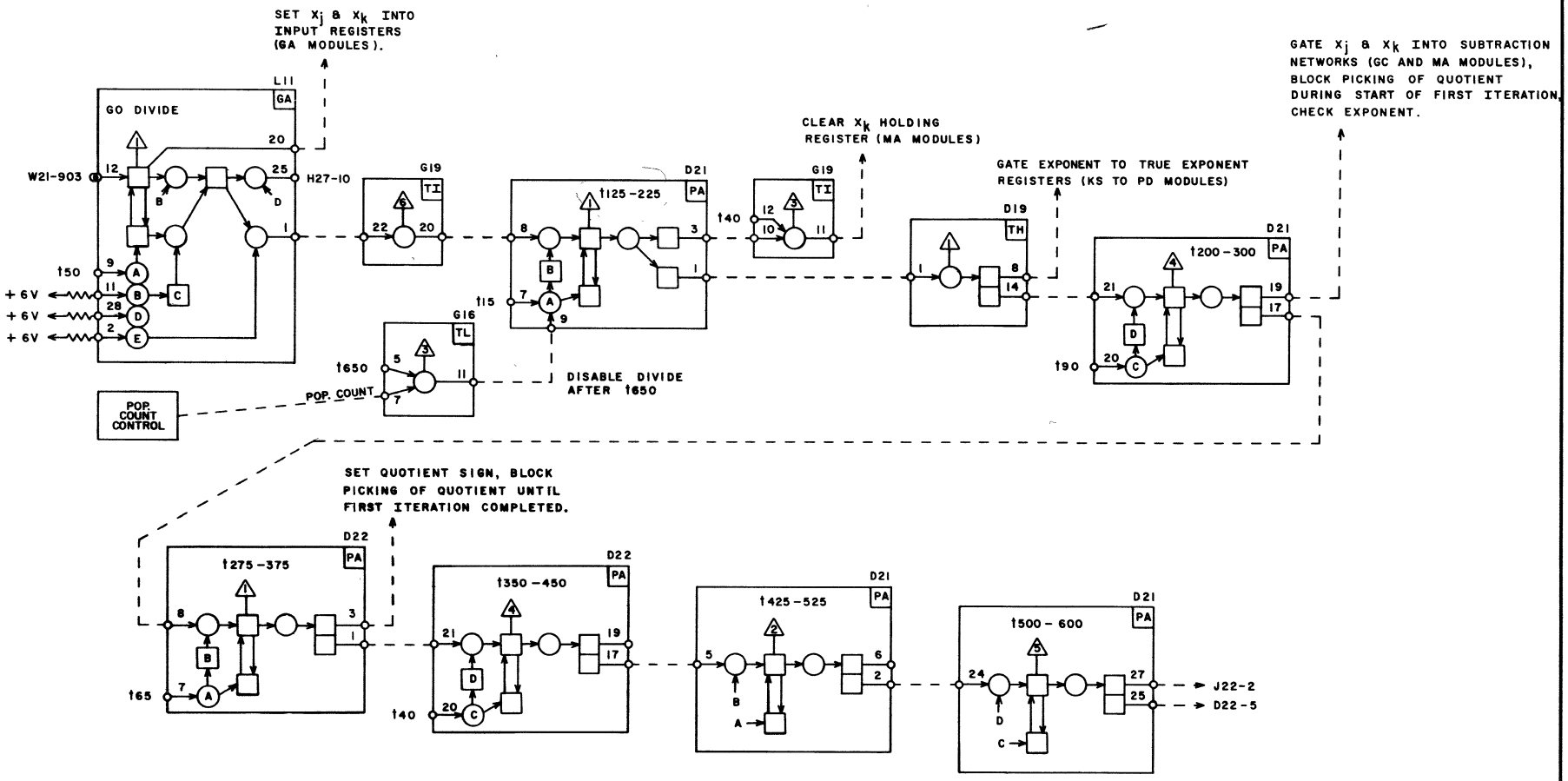
Bit 48 of the coefficient is examined by the quotient output network and if a "1", the coefficient is shifted one place to the right. This requires the addition of +1 to the exponent to maintain positional accuracy.



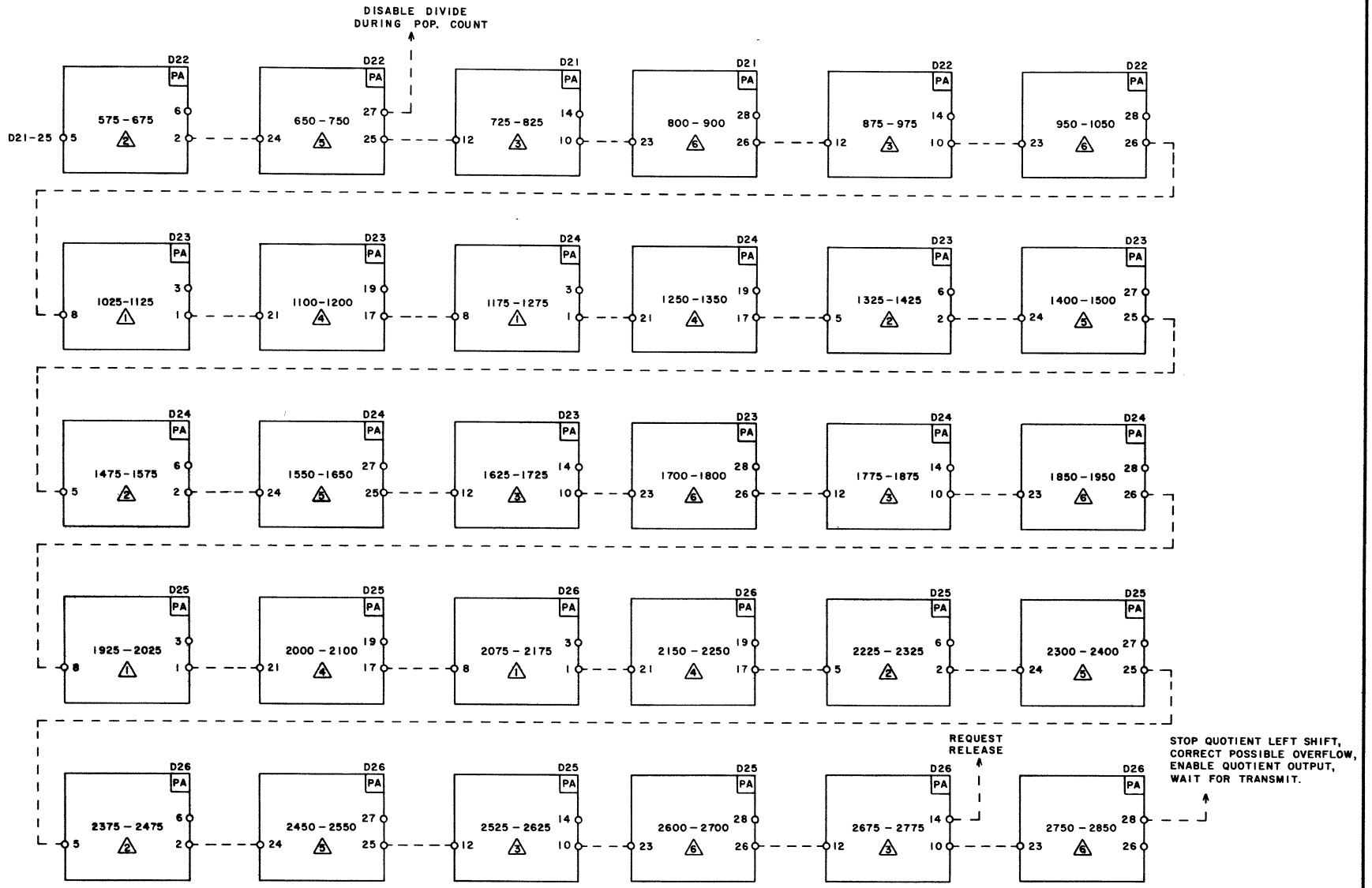
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR DIVIDE, EXPONENT BLOCK DIAGRAM	PRODUCT: 6601	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300
		REV: C	
		SHEET: 170	131

EXAMPLE: (using octal notation)





CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR DIVIDE TIMING CHAIN 1	PRODUCT 6601	
		SIZE C	DRAWING NO. 60119300 REV BT
			SHEET 171 133

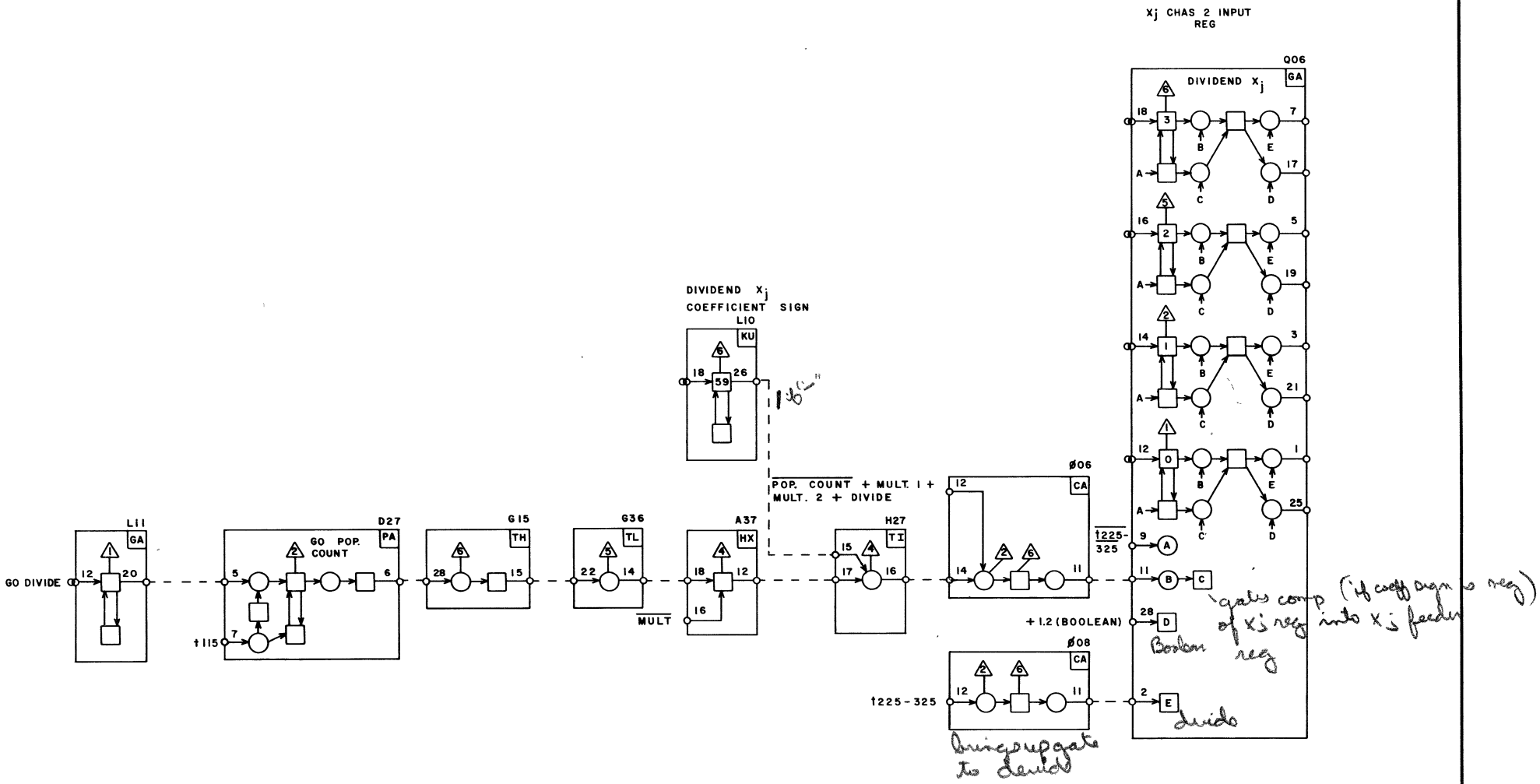


CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR DIVIDE TIMING CHAIN 2	PRODUCT: 6601	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: BT
			SHEET: 172 135

CHASSIS 2 INPUT REGISTERS

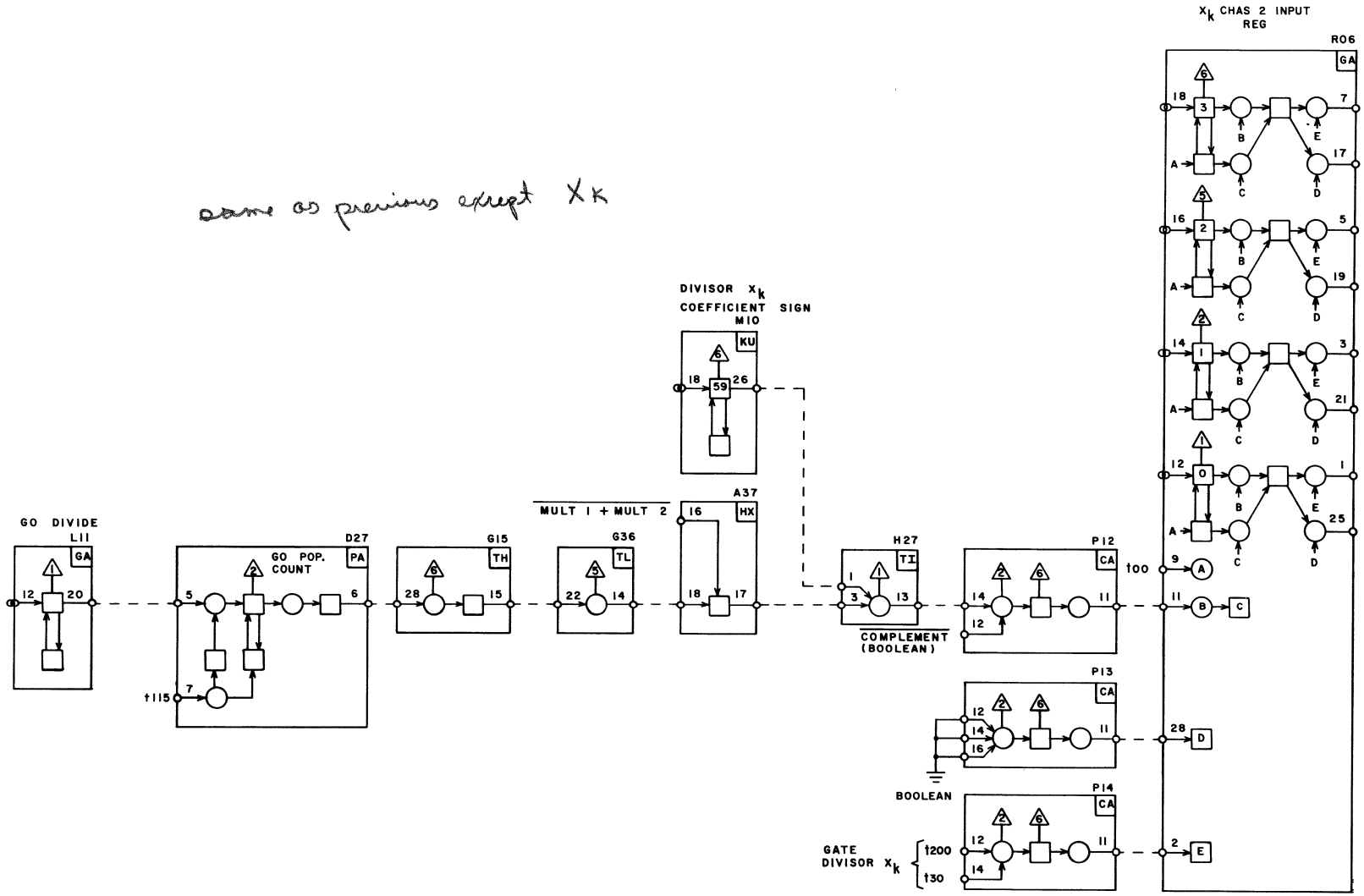
Chassis 2 contains the exponent arithmetic networks of Multiply 1 and Multiply 2 functional units, in addition to Divide and other logic. The input registers on chassis 2 are 60 bits in length. The exponent (bits 48 through 59) are sent to chassis 2 directly from the operating registers. The coefficient (bits 0 through 47) are sent first to chassis 6 and then re-transmitted to chassis 2.

The input registers on chassis 2 are shared by both Multiply 1 and 2, Divide and Boolean. The unit receiving the Go signal will sample the input registers and perform its respective operation. The two operands are held in the chassis 2 input registers for a time of approximately 100 nanoseconds. If a division is to be performed, the operation will be timed so that the Divide unit receives a Go signal and samples the operands while they are held in the chassis 2 input registers. The operand sign (bit 59) accompanies each operand to chassis 2. A sign bit of "1" indicates a negative operand. The input registers always contain the true values of the operands, but since negative numbers are handled in one's complement form, a sign bit of "1" gates the complement into the Divide unit.



CONTROL DATA CORPORATION	TITLE	PRODUCT	6601/04/13/14
COMPUTER DIVISION	CENTRAL PROCESSOR DIVIDE X_j, CHASSIS 2 INPUT REGISTER	SIZE C / DRAWING NO. 60119300 / REV 8T	
		SHEET 173	137

same as previous except X_k



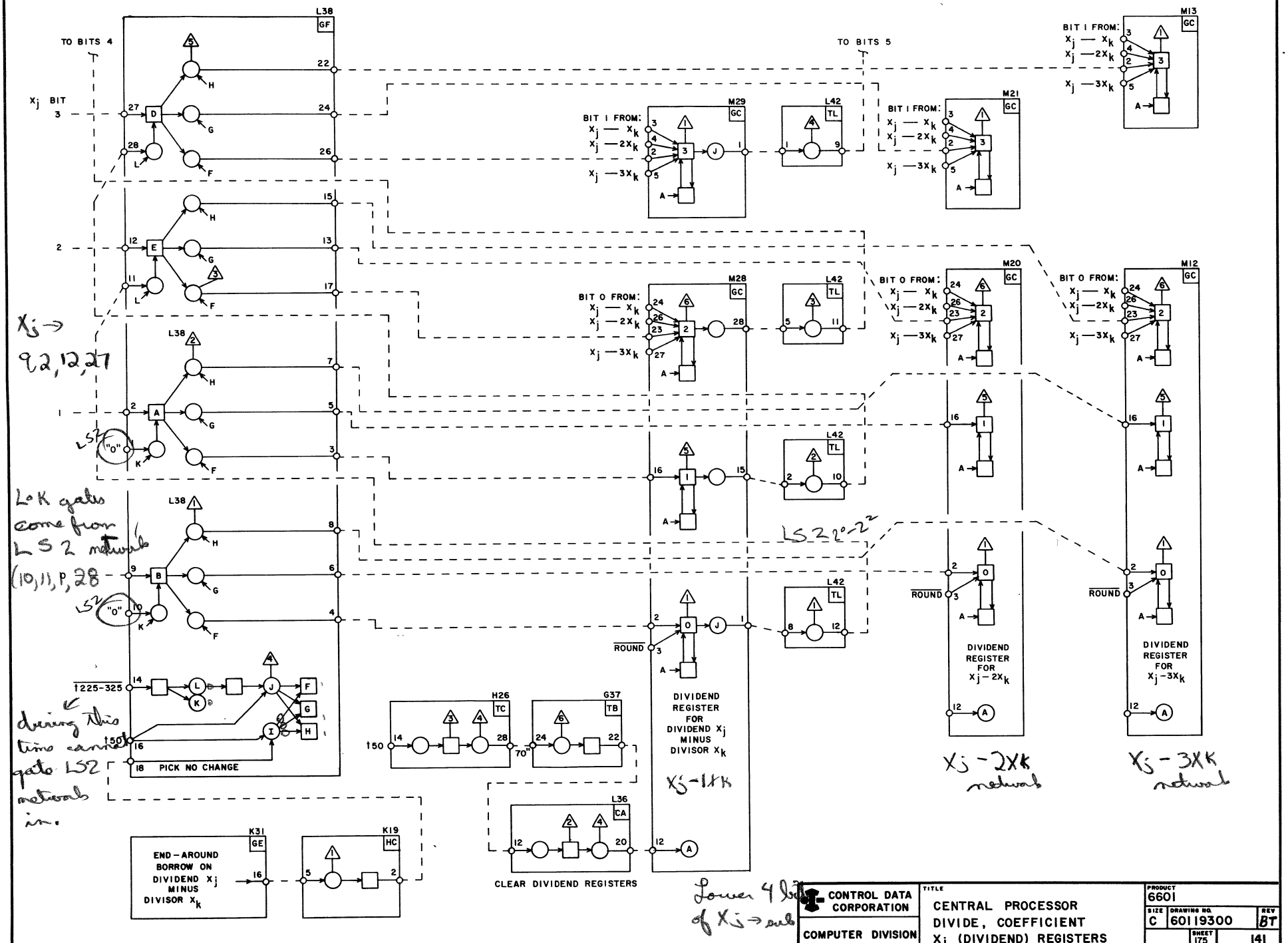
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR	PRODUCT: 6601	
COMPUTER DIVISION	DIVIDE X_k, CHASSIS 2 INPUT REGISTER	SIZE: C · DRAWING NO.: 60119300 · REV: 8T	
		SHEET: 174	139

DIVIDEND X_j

At the beginning of the first iteration, the dividend X_j is gated into holding registers in the 3 subtraction networks. Trial subtractions are then performed of the dividend X_j minus 1, 2, and 3 times the divisor X_k. The largest multiple of X_k which subtracts without causing an end-around borrow is taken as 2 bits of the quotient, and the result of that subtraction, left-shifted 2 places, is used as the dividend for the next iteration. The original dividend X_j is gone after the first iteration, and at the end of 24 iterations, the dividend is reduced to all "0's" (on an unrounded operation). If an end-around borrow is produced on the trial subtraction of X_j minus X_k, the two quotient bits will be "0's". The unsubtracted dividend quantity is then returned left-shifted 2 places to the 3 subtraction networks for the next iteration.

ROUND

Rounding in the Divide unit is performed by holding "1" inputs on bit 0 of the dividend registers in each of the three subtraction networks. The "1" inputs are held through all 24 iterations. Since the dividends are left-shifted 2 places on each iteration, the quantity remaining in the registers at the end of 24 iterations will be a succession of alternating "0's" and "1's". Rounding on a divide operation forces bit 0 of the original dividend X_j to a "1" and makes the portion of the dividend to the right of the binary point equal to 1/3 ($.2525\ --\ 25_8$).



CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE
CENTRAL PROCESSOR
DIVIDE, COEFFICIENT
X_j (DIVIDEND) REGISTERS

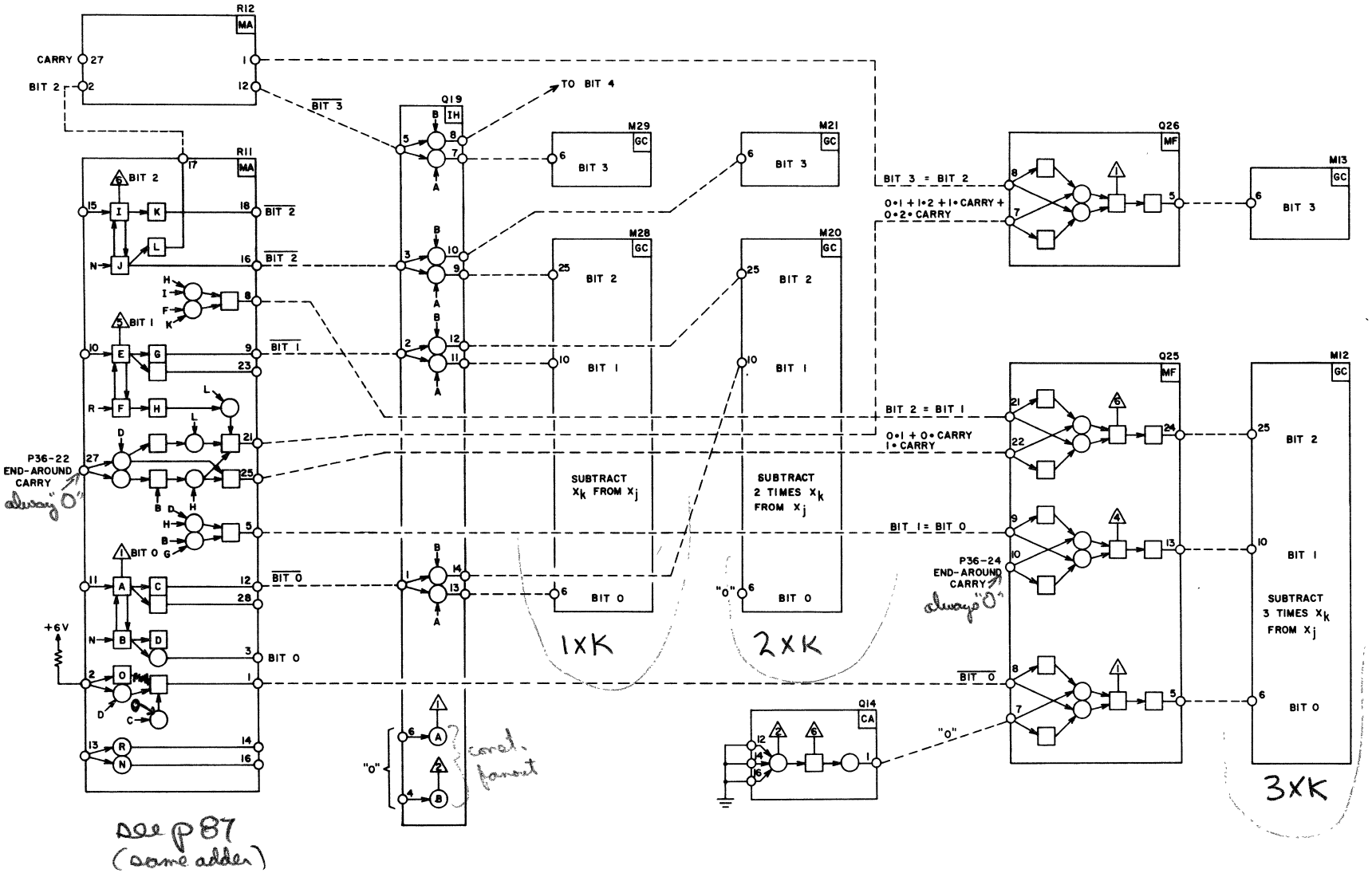
PRODUCT			
6601			
SIZE	DRAWING NO.		REV
C	60119300		BT
	SHEET 175		141

DIVISOR X_k

The original divisor X_k is brought into a register at the beginning of the operation and three values are formed; X_k times 1, times 2, and times 3. These three values are held during the entire operation and are used as inputs to the three subtraction networks.

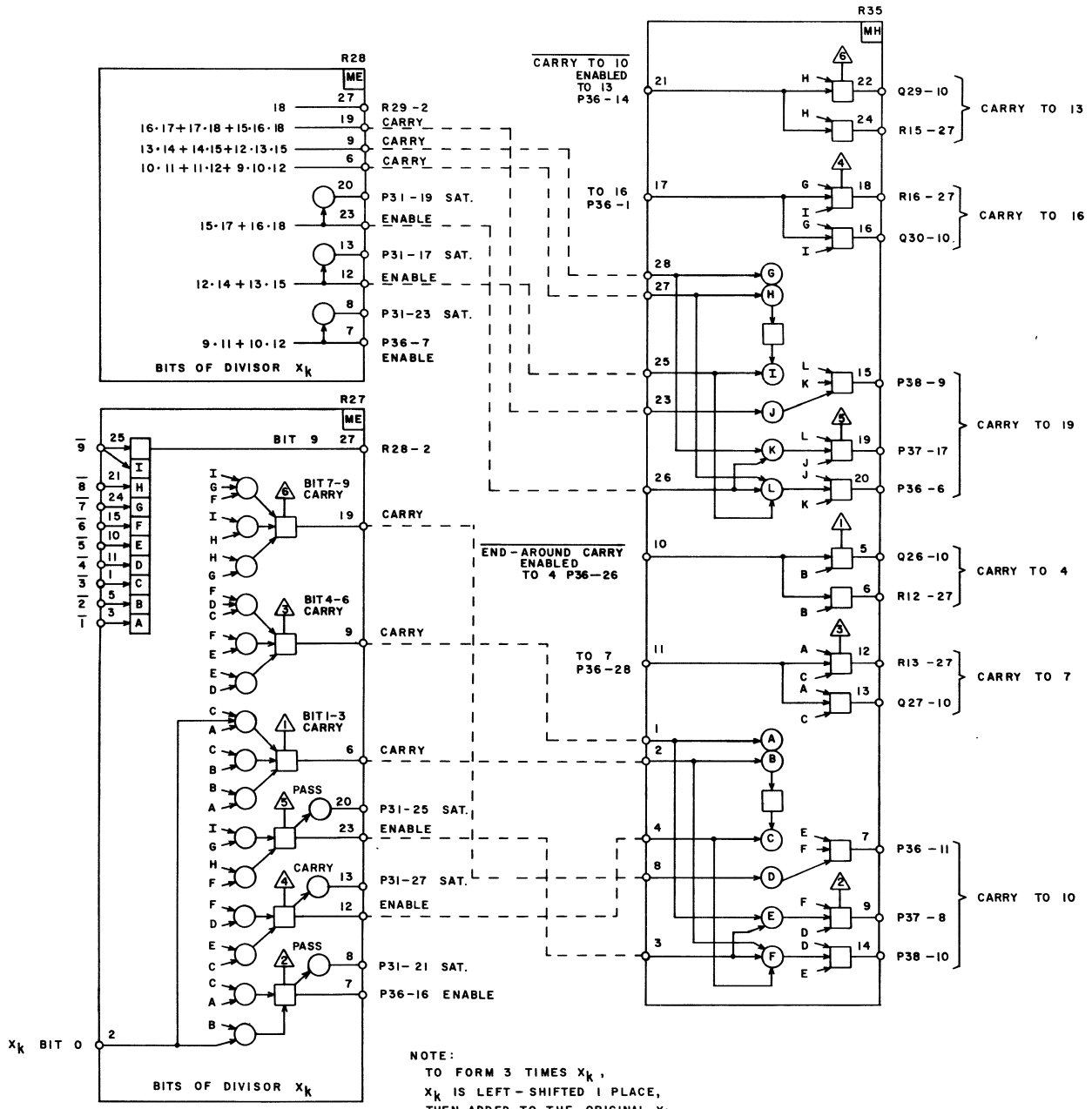
The quantity $2X_k$ is formed by shifting X_k one place to the left. This is equivalent to multiplying by 010_2.

The quantity $3X_k$ is formed by shifting X_k one place to the left and adding this quantity to the original X_k. This is equivalent to multiplying by 011_2. The carries produced by this addition are handled by a separate sensing network.



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR DIVIDE, COEFFICIENT X_k (DIVISOR) REGISTERS; TIMES 1, 2 AND 3	PRODUCT 6601 SIZE C · DRAWING NO. 60119300 · REV D SHEET 176 · PAGE 143	

no EAC



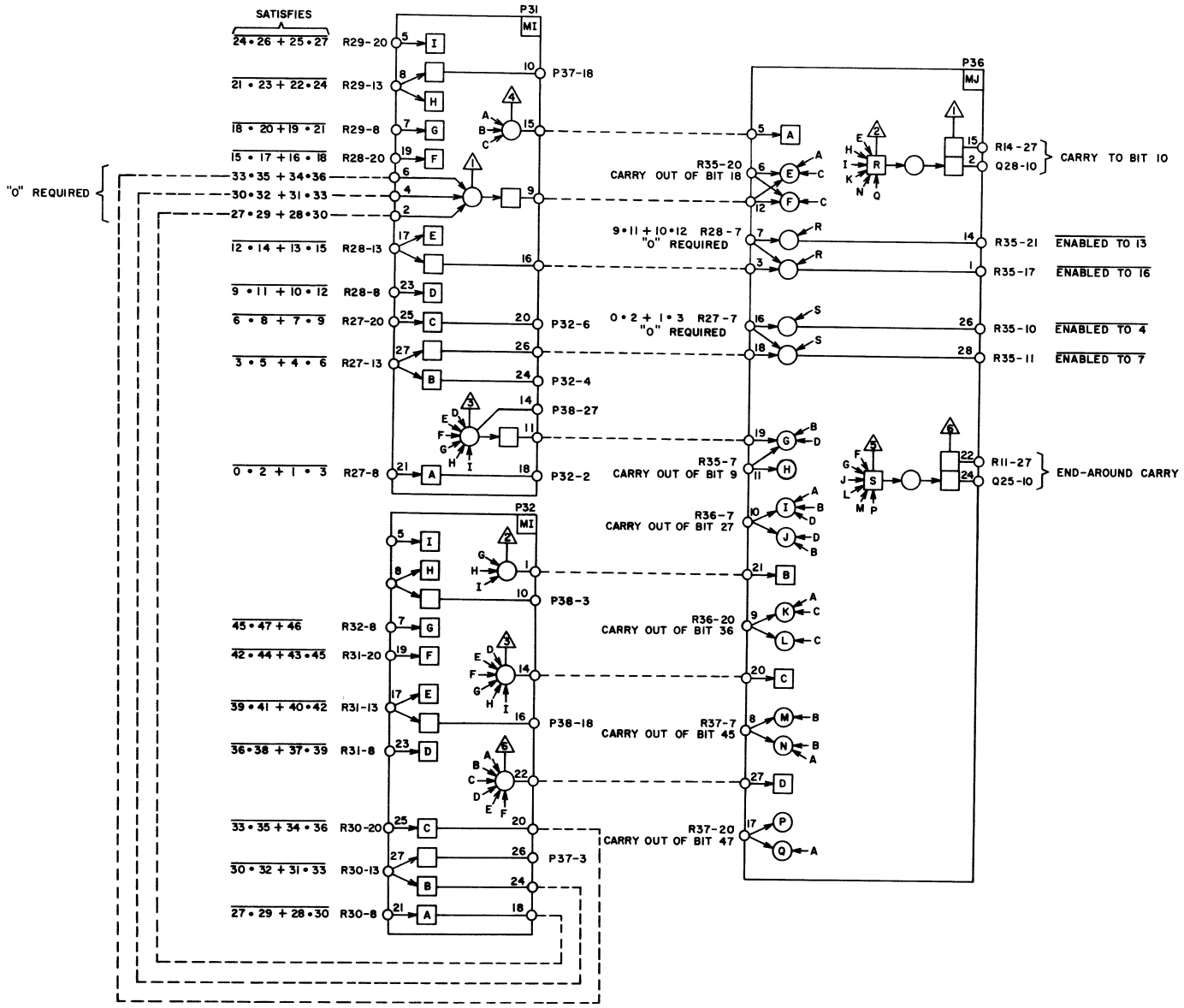
NOTE:
TO FORM 3 TIMES X_k,
X_k IS LEFT - SHIFTED 1 PLACE,
THEN ADDED TO THE ORIGINAL X_k

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

CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE
CENTRAL PROCESSOR
DIVIDE, COEFFICIENT
CARRIES ON 3 TIMES X_k, 1

PRODUCT 6601
SIZE C | DRAWING NO. 60119300 | REV BT
SHEET 177
145



CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR DIVIDE, COEFFICIENT CARRIES ON 3 TIMES X_k, 2	PRODUCT: 6601	
COMPUTER DIVISION		SIZE: C · DRAWING NO.: 60119300 · REV: BT	
		SHEET: 178 · PAGE: 147	

SUBTRACTION NETWORKS

The trial subtractions of the dividend minus X_k, $2X_k$, and $3X_k$ are performed by the 3 subtraction networks. Each of these networks contains an input register for the dividend. The multiple value of X_k is brought in at a separate input.

Each subtraction network has 3 gated outputs from each stage. These are fed back to the dividend input registers of the subtraction networks left-shifted 2 places. The output of each subtraction network is the complement of the true result, and is therefore able to directly set the flip-flops in the input registers.

Borrows, satisfies, and enables are handled by a separate sensing network. The subtraction networks are one's complement, and a borrow out of the upper stage is taken end-around. However, if an end-around borrow is produced by a subtraction network, that particular trial subtraction is rejected.

The existence of end-around borrows from the trial subtraction networks determines the selection of the two quotient bits for that iteration. The highest multiple of X_k which will subtract from the dividend without causing an end-around borrow is used. The fully subtracted output from that subtraction network is then gated back to the dividend input registers left-shifted 2 places to be used as the dividend on the next iteration.

E	E	G	B	S
1	0		0	1
1	0		1	0

$\overline{EQ} \cdot \overline{B} = 1$

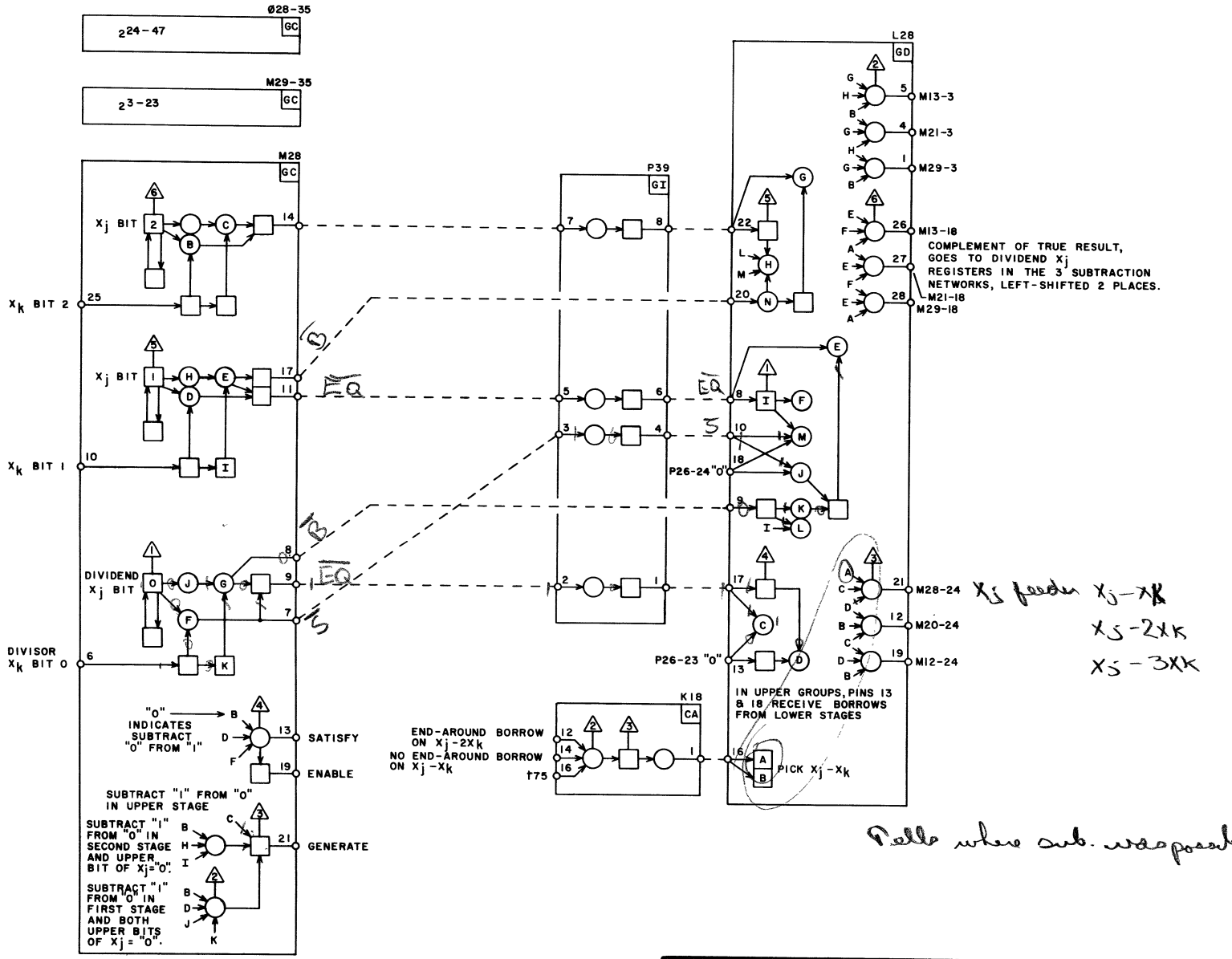
$EQ \cdot B = 1$

$\overline{EQ} \cdot B = 0$

$EQ \cdot \overline{B} = 0$

True subtracter

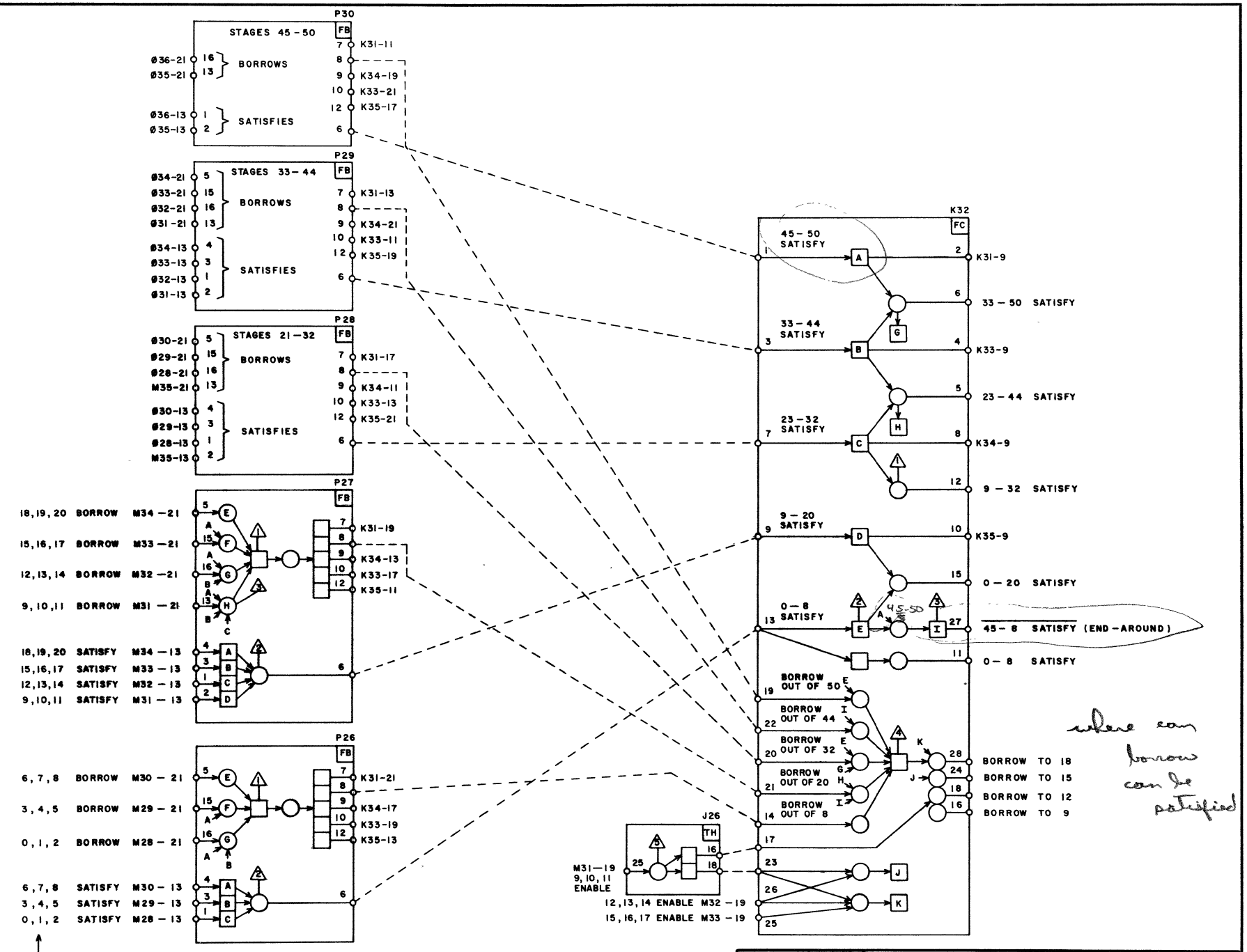
True data

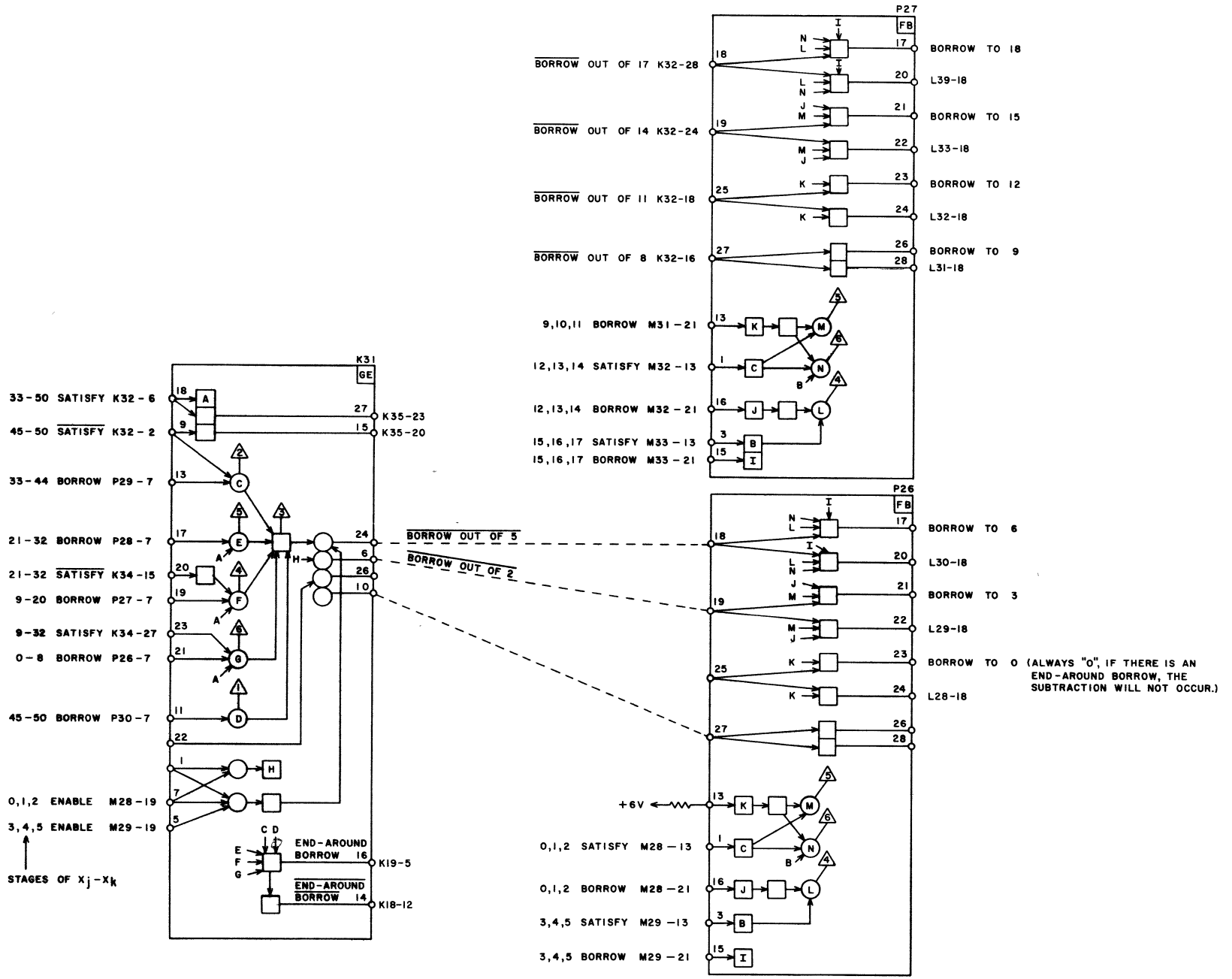


X_j feeds $X_j - X_k$
$X_j - 2X_k$
$X_j - 3X_k$

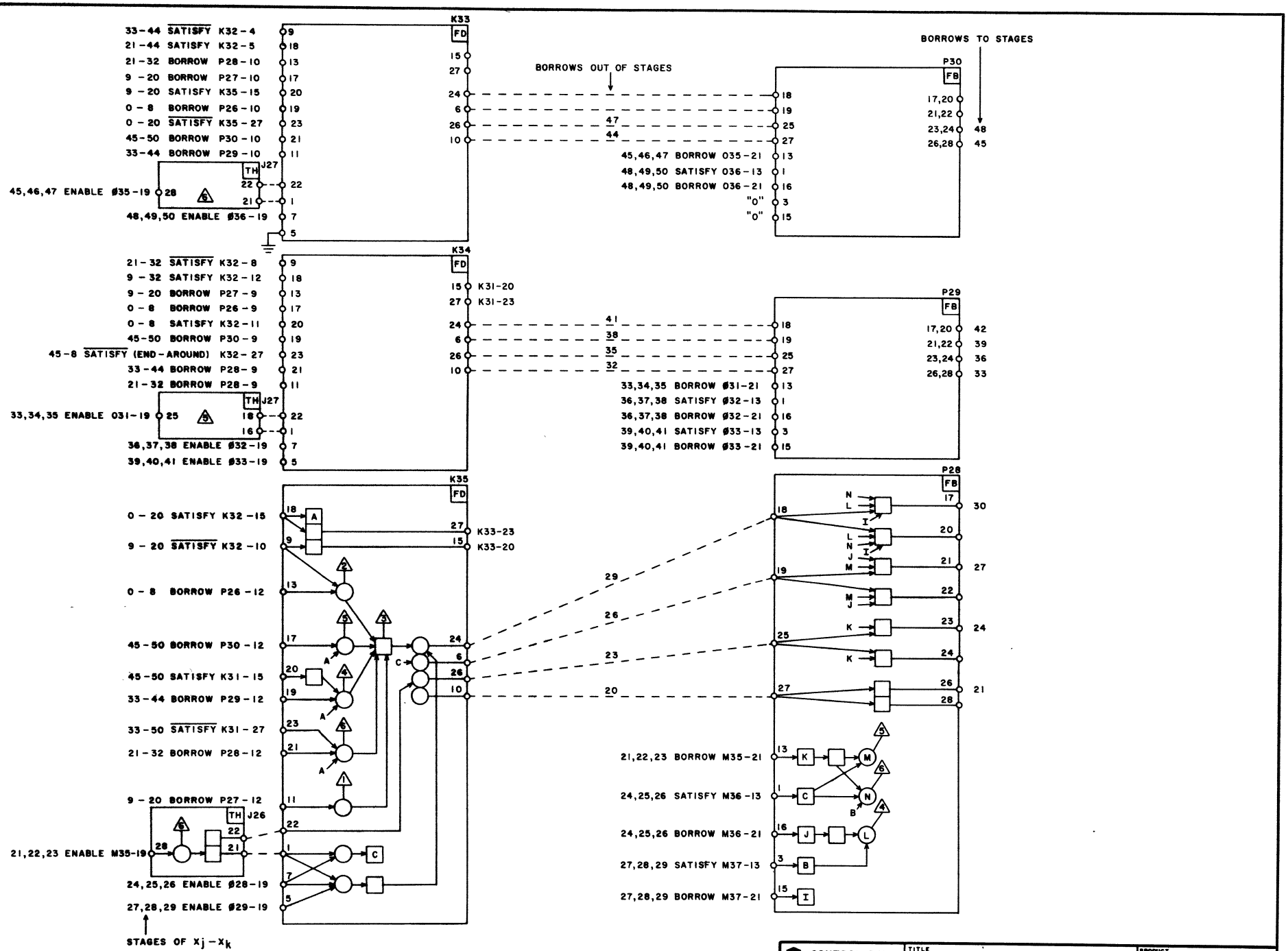
Table where sub. was possible

CONTROL DATA CORPORATION	TITLE	PRODUCT
COMPUTER DIVISION	CENTRAL PROCESSOR DIVIDE, COEFFICIENT SUBTRACT X_k FROM X_j	6601/04/13/14
	SIZE C · DRAWING NO. 60119300 · REV 8T	
	SHEET 179	149


P30
STAGES 45 – 50
FB
Ø36-21 16
Ø35-21 13
BORROWS
Ø36-13 1
Ø35-13 2
SATISFIES
7 K31-11
8
9 K34-19
10 K33-21
12 K35-17
6
P29
STAGES 33 – 44
FB
Ø34-21 5
Ø33-21 15
Ø32-21 16
Ø31-21 13
BORROWS
Ø34-13 4
Ø33-13 3
Ø32-13 1
Ø31-13 2
SATISFIES
7 K31-13
8
9 K34-21
10 K33-11
12 K35-19
6
P28
STAGES 21 – 32
FB
Ø30-21 5
Ø29-21 15
Ø28-21 16
M35-21 13
BORROWS
Ø30-13 4
Ø29-13 3
Ø28-13 1
M35-13 2
SATISFIES
7 K31-17
8
9 K34-11
10 K33-13
12 K35-21
6
P27
FB
18,19,20 BORROW M34 – 21
15,16,17 BORROW M33 – 21
12,13,14 BORROW M32 – 21
9,10,11 BORROW M31 – 21
18,19,20 SATISFY M34 – 13
15,16,17 SATISFY M33 – 13
12,13,14 SATISFY M32 – 13
9,10,11 SATISFY M31 – 13
7 K31-19
8
9 K34-13
10 K33-17
12 K35-11
6
P26
FB
6,7,8 BORROW M30 – 21
3,4,5 BORROW M29 – 21
0,1,2 BORROW M28 – 21
6,7,8 SATISFY M30 – 13
3,4,5 SATISFY M29 – 13
0,1,2 SATISFY M28 – 13
STAGES OF $X_j - X_k$
7 K31-21
8
9 K34-17
10 K33-19
12 K35-13
6
K32
FC
1 45 – 50 SATISFY
3 33 – 44 SATISFY
7 23 – 32 SATISFY
9 9 – 20 SATISFY
13 0 – 8 SATISFY
19 BORROW OUT OF 50
22 BORROW OUT OF 44
20 BORROW OUT OF 32
21 BORROW OUT OF 20
14 BORROW OUT OF 8
17
23
26
25
2 K31-9
6 33 – 50 SATISFY
4 K33-9
5 23 – 44 SATISFY
8 K34-9
12 9 – 32 SATISFY
10 K35-9
15 0 – 20 SATISFY
27 45 – 8 SATISFY (END-AROUND)
11 0 – 8 SATISFY
28 BORROW TO 18
24 BORROW TO 15
18 BORROW TO 12
16 BORROW TO 9
J26
TH
M31–19 9,10,11 ENABLE
12,13,14 ENABLE M32 – 19
15,16,17 ENABLE M33 – 19
where can borrow can be satisfied
CONTROL DATA CORPORATION
COMPUTER DIVISION
TITLE
CENTRAL PROCESSOR
DIVIDE, COEFFICIENT
BORROWS ON SUBTRACT
X_j FROM X_k, I
PRODUCT 6601
SIZE C
DRAWING NO. 60119300
REV BT
SHEET 180
151



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR DIVIDE, COEFFICIENT BORROWS ON SUBTRACT X_j FROM X_k, 2	PRODUCT 6601	
		SIZE C	DRAWING NO. 60119300 REV BT
			SHEET 181 153



CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE
CENTRAL PROCESSOR
DIVIDE, COEFFICIENT BORROWS
ON SUBTRACT Xj FROM Xk, 3

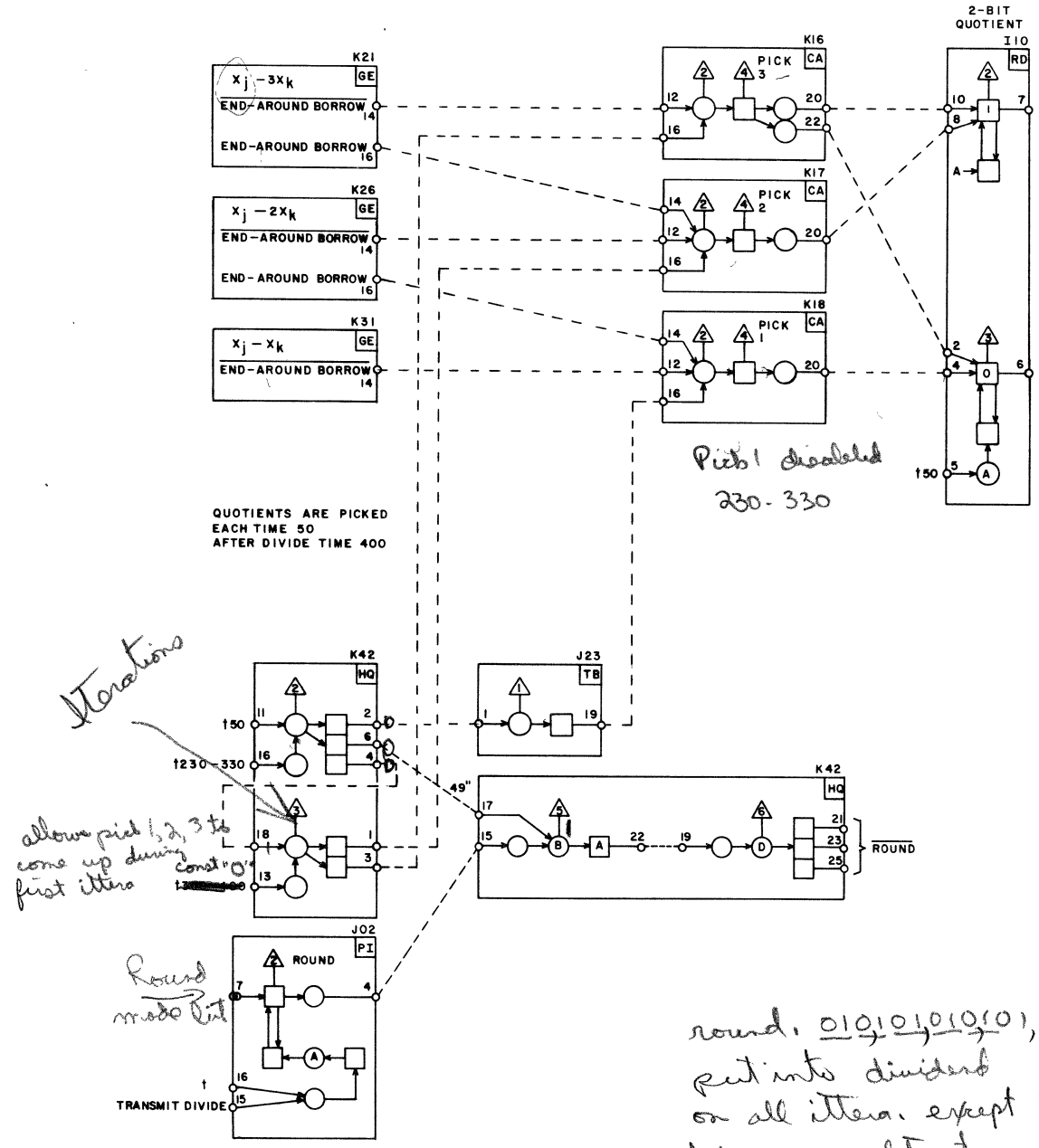
PRODUCT 6601
SIZE C DRAWING NO. 60119300 REV BT
SHEET 182
155

2-BIT QUOTIENTS

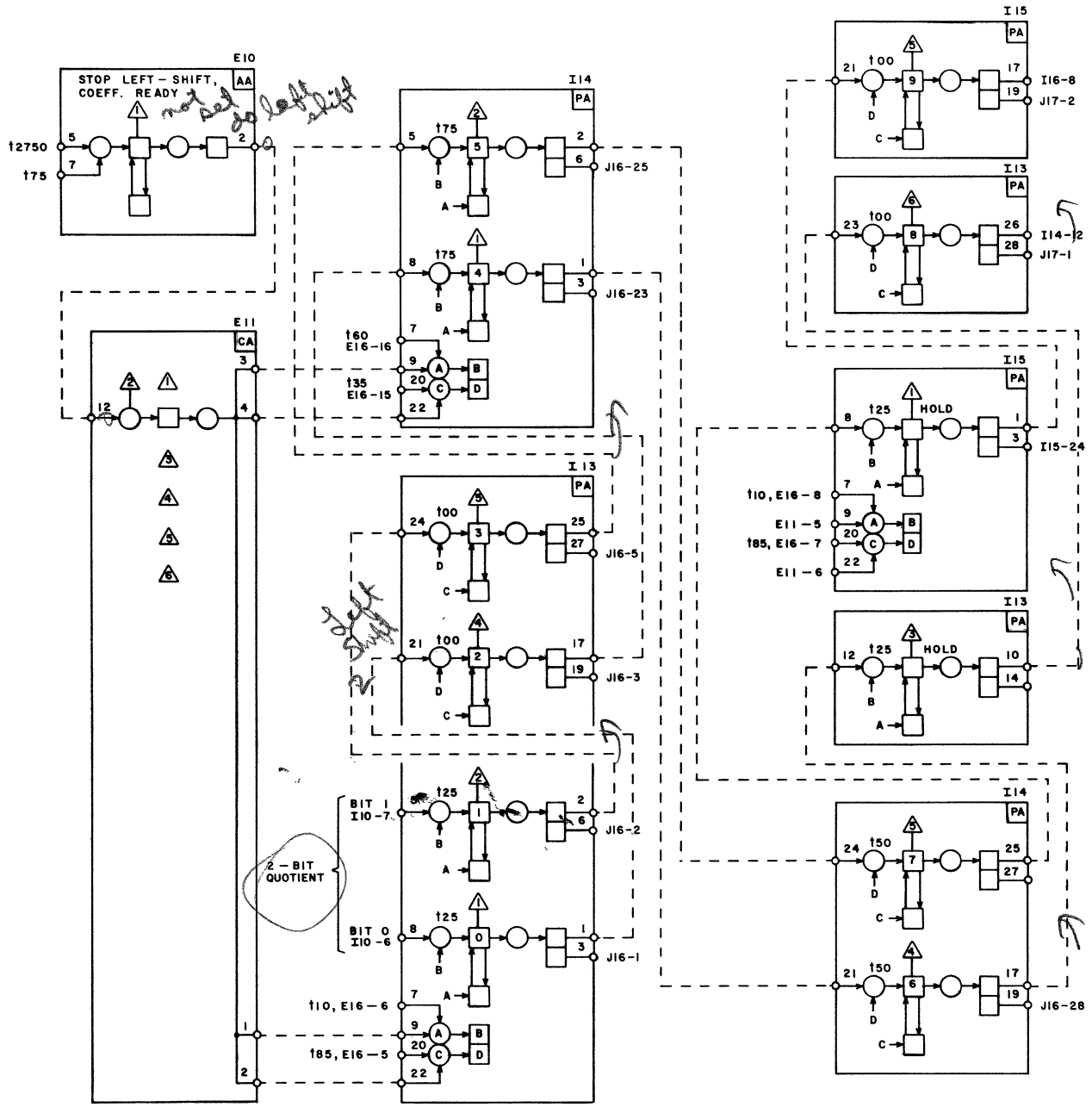
The coefficient arithmetic network performs division of the 48-bit operands in 24 iterations. The quotient is assembled 2 bits at a time, as determined by the three trial subtractions of X_k, $2X_k$, and $3X_k$ from the dividend. The resulting quotient is 50 bits in length, since an additional 2 bits are picked after the 24th subtraction.

QUOTIENT LEFT-SHIFT REGISTER

A 2-place left-shift register is used to assemble the 2-bit quotients into the final 50-bit quotient. The quotients are selected at 100-nanosecond intervals according to the trial subtractions, and are inserted into stages 0 and 1 of the left-shift register. This register performs a 2-place left shift at 75-nanosecond intervals. It is therefore necessary to provide 2 extra register stages after each 4 shifts to allow the picking of quotients to "catch up" with the shifting of the register.

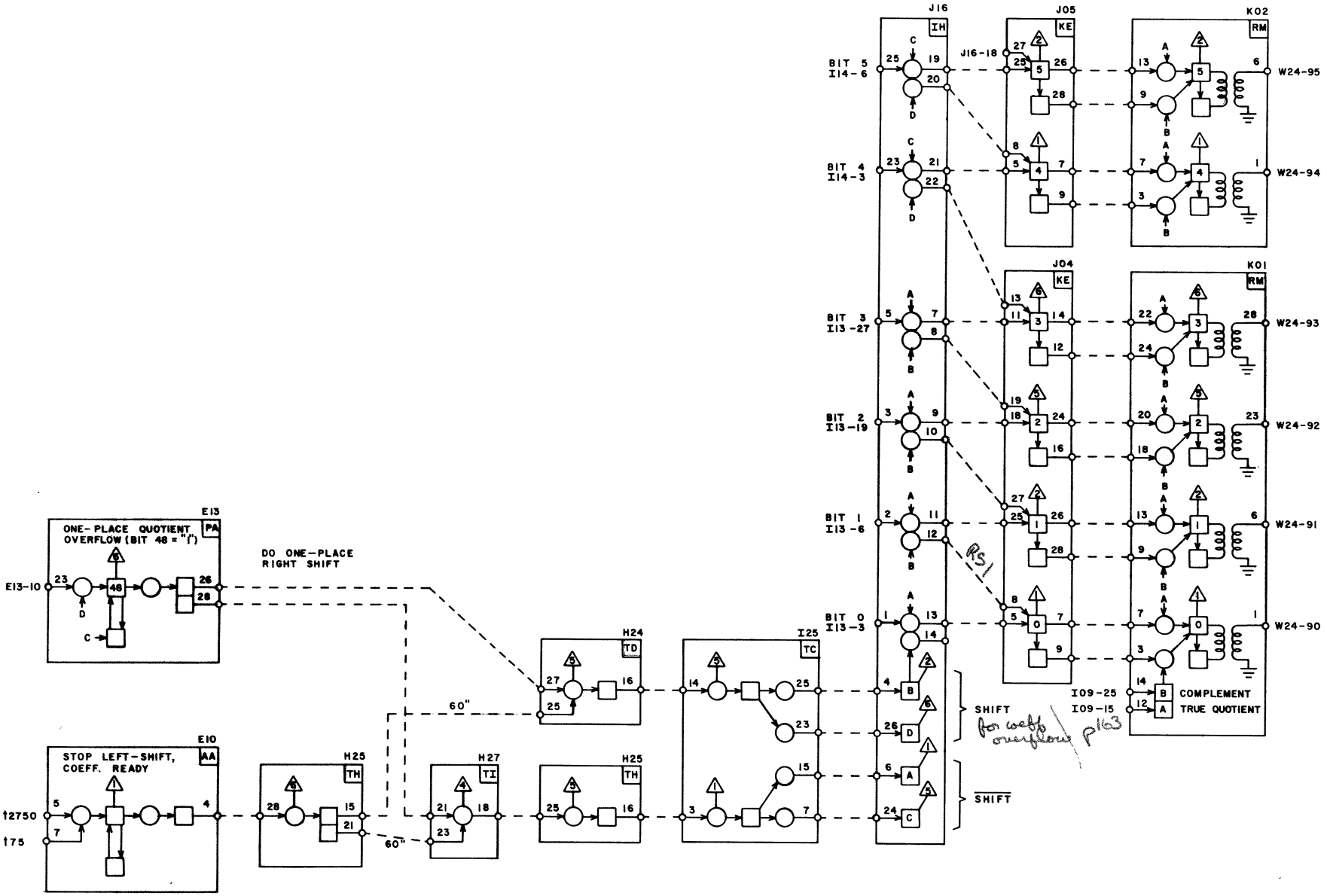


CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR	PRODUCT: 6601/04/13/14	
COMPUTER DIVISION	DIVIDE, COEFFICIENT FORM 2-BIT QUOTIENT	SIZE: C · DRAWING NO.: 60119300 · REV: N	
		SHEET: 183	157



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR DIVIDE, COEFFICIENT QUOTIENT LEFT-SHIFT REG.	PRODUCT 6601	
		SIZE C	DRAWING NO. 60119300 REV 8T
		SHEET 184	159

The divide unit produces a normalized quotient (a "1" in bit 47) if both of the original operands were normalized. With normalized operands, the first 2-bit quotient will be either 00 or 01. If the first 2-bit quotient is 00, the second 2-bit quotient will be either 10 or 11. Due to the left shifts, the first 2-bit quotient becomes bits 49 and 48 of the final 50-bit result, and the second 2-bit quotient becomes bits 47 and 46. The use of normalized operands therefore always results in a "1" in either (or both) bit 47 or 48, while bit 49 is always a "0". If bit 48 is a "1", the output network shifts the coefficient one place to the right and +1 is added to the exponent to maintain positional accuracy.



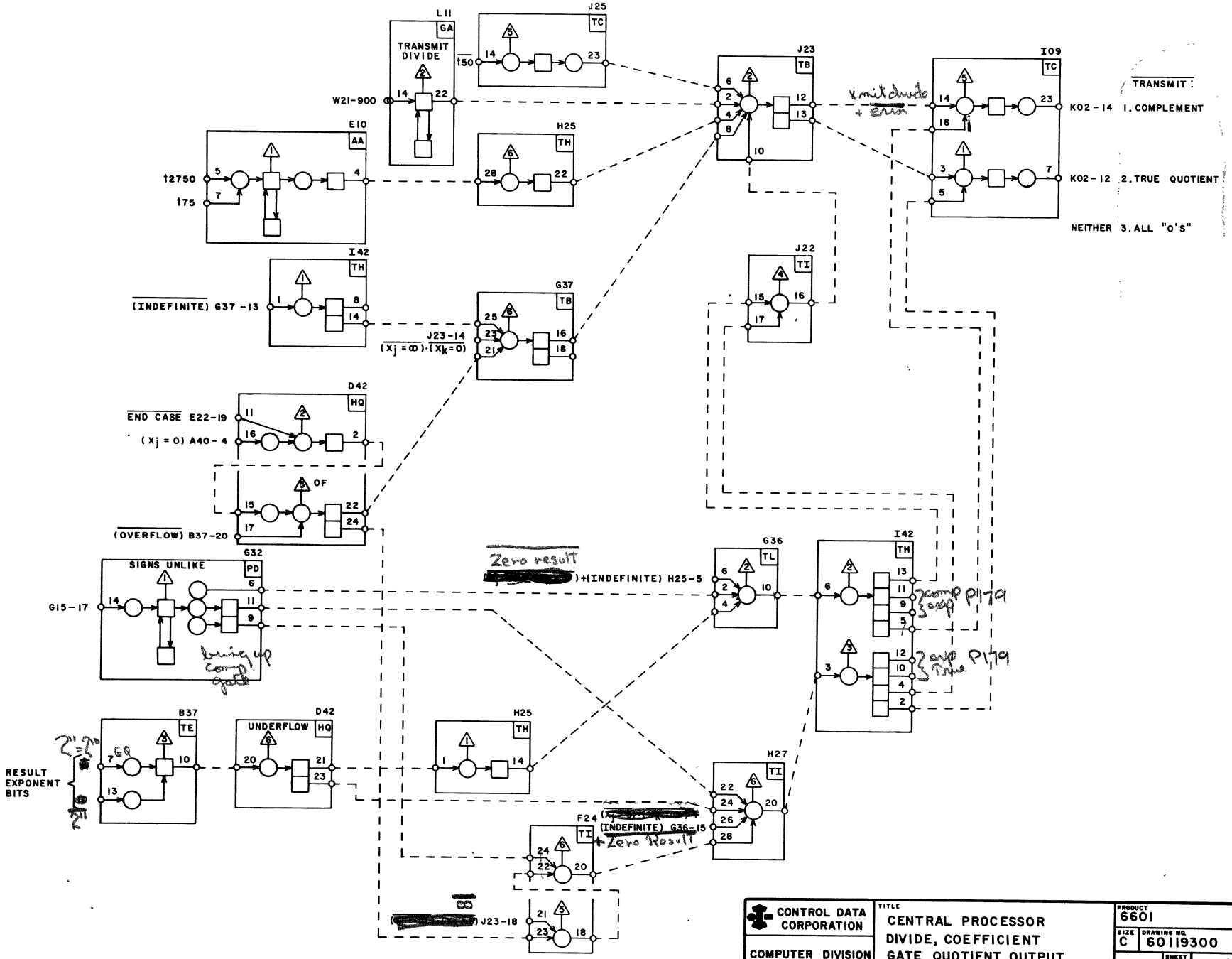
CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR DIVIDE, COEFFICIENT QUOTIENT OUTPUT NETWORK	PRODUCT 6601 SIZE C · DRAWING NO. 60119300 · REV BT SHEET 185	161

GATE QUOTIENT OUTPUT

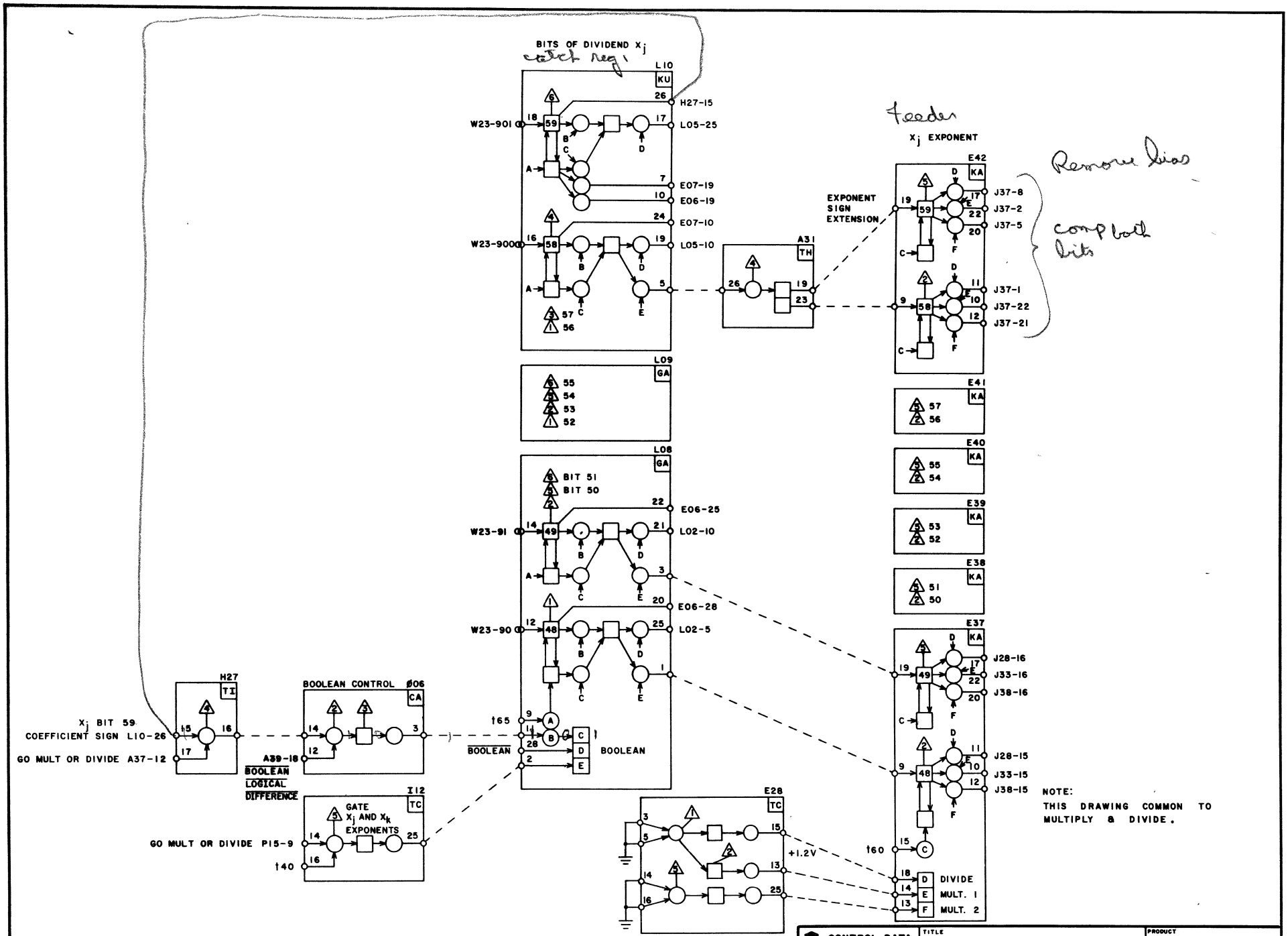
The final result quotient is gated out of the Divide unit by the Transmit signal from the Scoreboard. The final transmitted coefficient value is determined by a sensing network which selects either the true value, the complement or all "0's".

The signs of the original operands X_j and X_k determine the selection of either the true value or the complement of the result coefficient. If the signs were alike, either both positive or both negative, the quotient will be positive; if unlike, the quotient will be negative. Negative numbers are handled in one's complement notation. The sensing network therefore selects the complement of the coefficient if the quotient is negative.

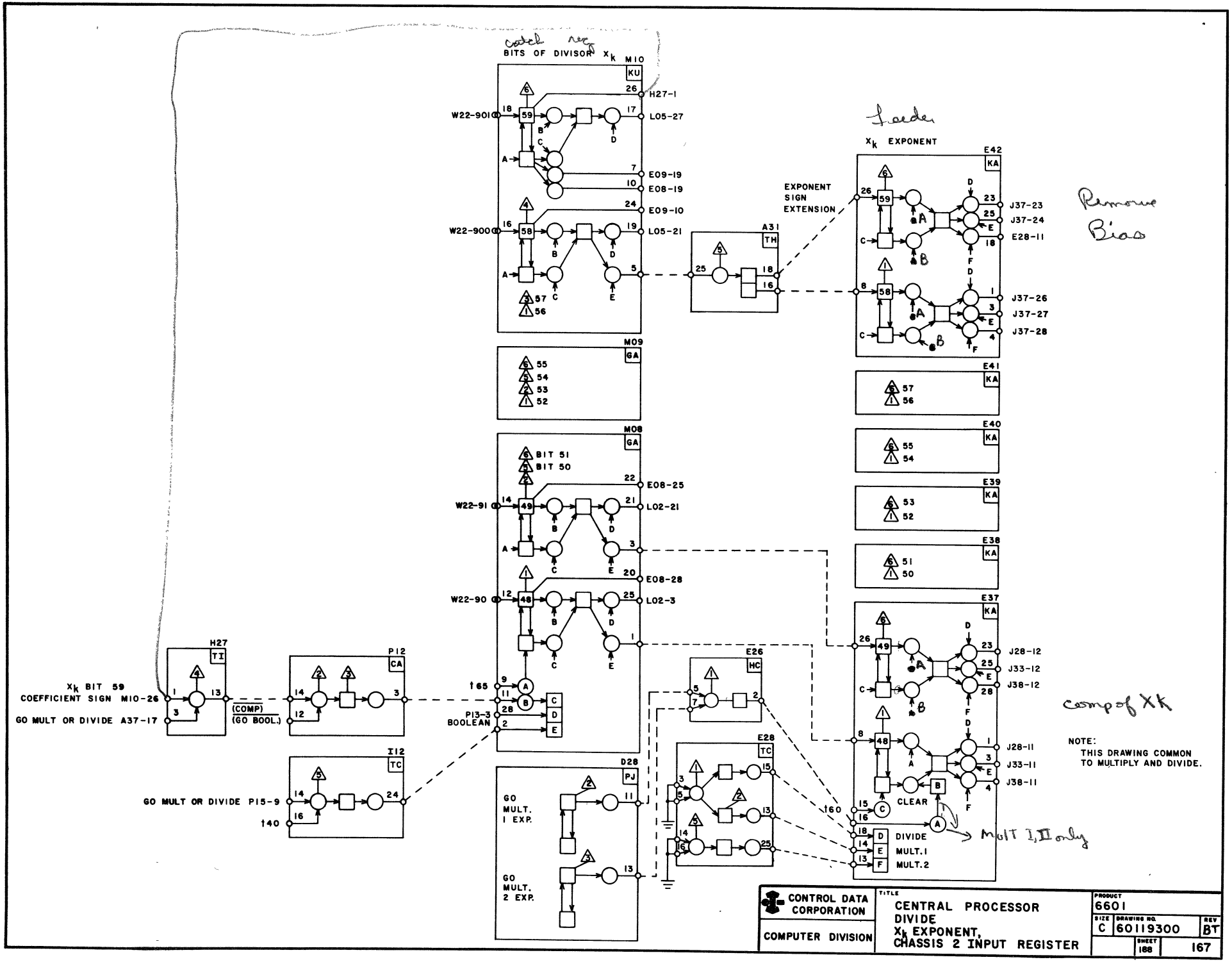
The output coefficient is held to all "0's" if the quotient is infinite, indefinite, or zero. The existence of these conditions is determined by examining the original operands and the exponent of the result.



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR DIVIDE, COEFFICIENT GATE QUOTIENT OUTPUT	PRODUCT 6601 SIZE C · DRAWING NO. 60119300 · REV BT SHEET 186	163

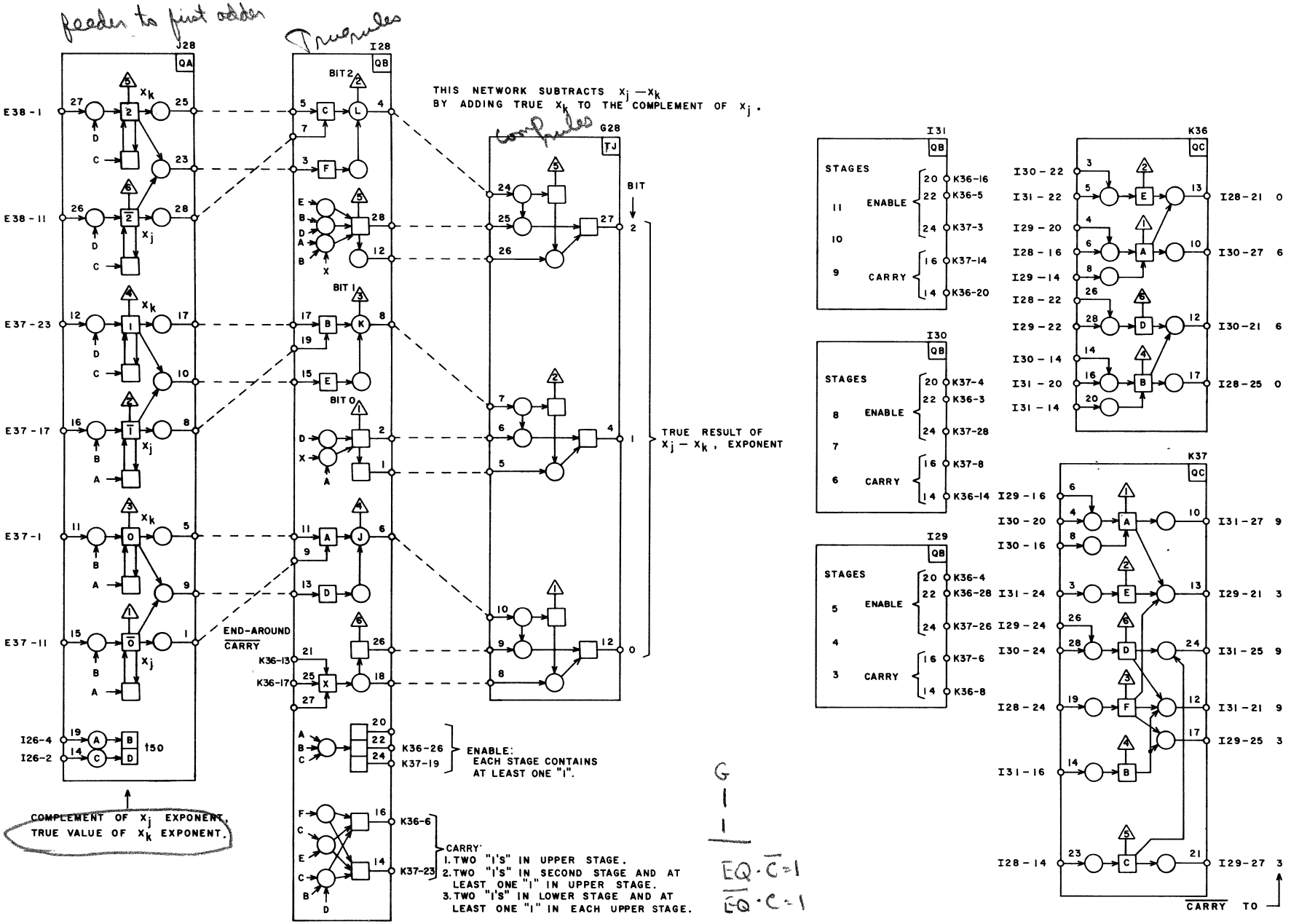


CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR DIVIDE X_j EXPONENT, CHASSIS 2 INPUT REGISTER	PRODUCT 6601 SIZE C · DRAWING NO. 60119300 · REV 8T SHEET 187 · 165	


BITS OF DIVISOR X_k M10
KU
H27-1
W22-901
L05-27
E09-19
E08-19
E09-10
W22-900
L05-21
M09
GA
M08
E08-25
W22-91
L02-21
E08-28
W22-90
L02-3
X_k BIT 59
COEFFICIENT SIGN M10-26
GO MULT OR DIVIDE A37-17
(COMP)
(GO BOOL.)
P13-3
BOOLEAN
GO MULT OR DIVIDE P15-9
GO MULT. 1 EXP.
GO MULT. 2 EXP.
A31
TH
EXPONENT SIGN EXTENSION
X_k EXPONENT
E42
KA
J37-23
J37-24
E28-11
J37-26
J37-27
J37-28
E41
E40
E39
E38
E37
J28-12
J33-12
J38-12
J28-11
J33-11
J38-11
E26
HC
E28
TC
CLEAR
DIVIDE
MULT.1
MULT.2
H27
TI
P12
CA
I12
D28
PJ
CONTROL DATA CORPORATION
COMPUTER DIVISION
CENTRAL PROCESSOR
DIVIDE
X_k EXPONENT,
CHASSIS 2 INPUT REGISTER
6601
C
60119300
BT
168
167

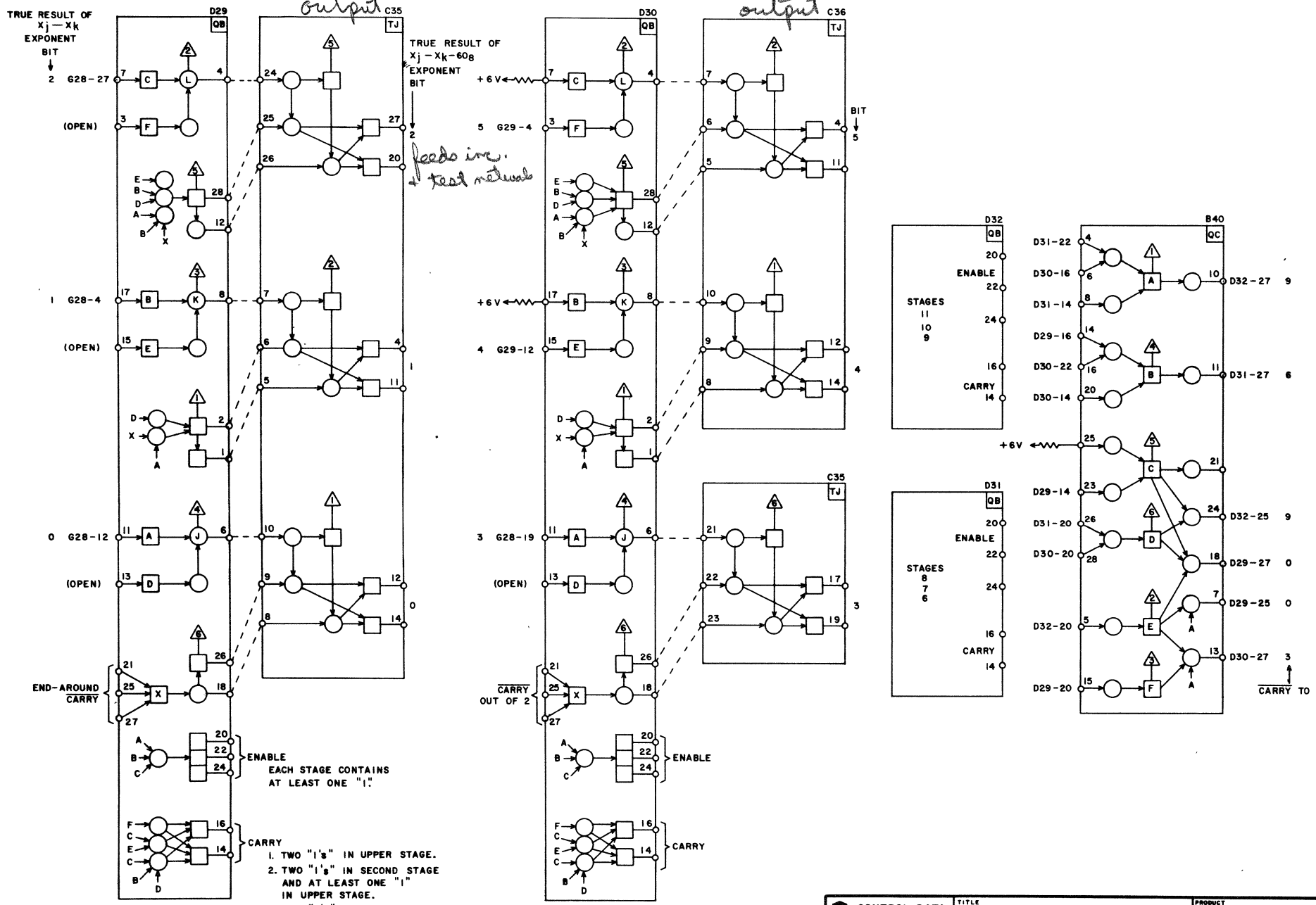
EXPONENT SUBTRACTION

The exponent arithmetic logic unconditionally performs two subtractions at the beginning of the operation. The first step is to subtract the exponent of the divisor X_k from the exponent of the dividend X_j. The second step is to subtract from this result the value 60_8 (48_{10}). This produces the basic quotient exponent. Later in the operation, the control logic will determine whether further modification such as normalizing should be done. The networks which perform the subtractions are adders which subtract a quantity by adding it to the complement of the first quantity. The adders are one's complement; if complementary numbers are added, the result is all "0's" (positive zero).



CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR DIVIDE, EXPONENT SUBTRACT, $X_j - X_k$	PRODUCT: 6601	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: BT
		SHEET: 169	169

SUBTRACTS 60_8 BY ADDING COMPLEMENT

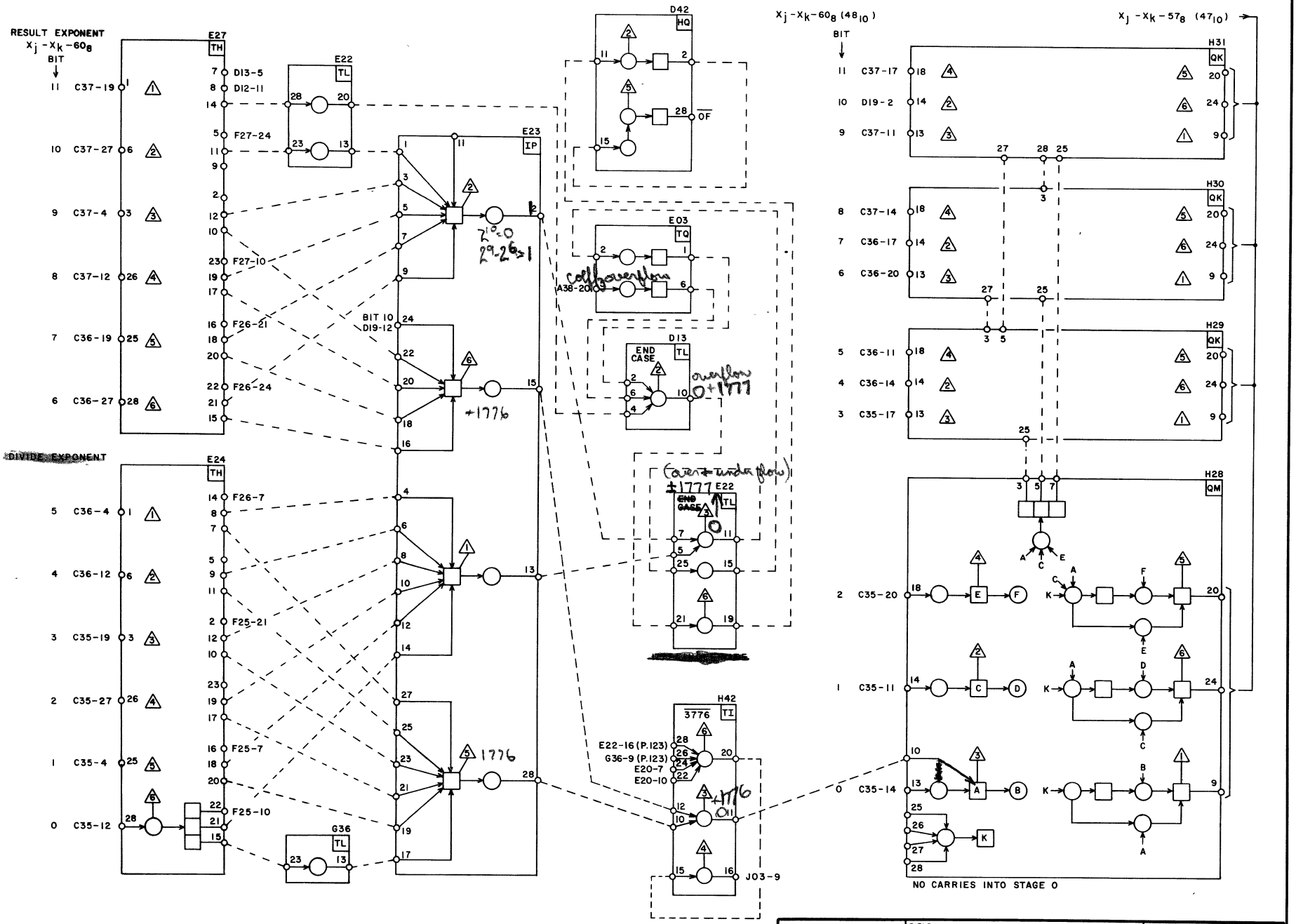


CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR DIVIDE, EXPONENT SUBTRACT $60_8 (48_{10})$, $X_j - X_k - 60_8$	PRODUCT 6601	
COMPUTER DIVISION		SIZE C	DRAWING NO. 60119300 REV C
		SHEET 190	171

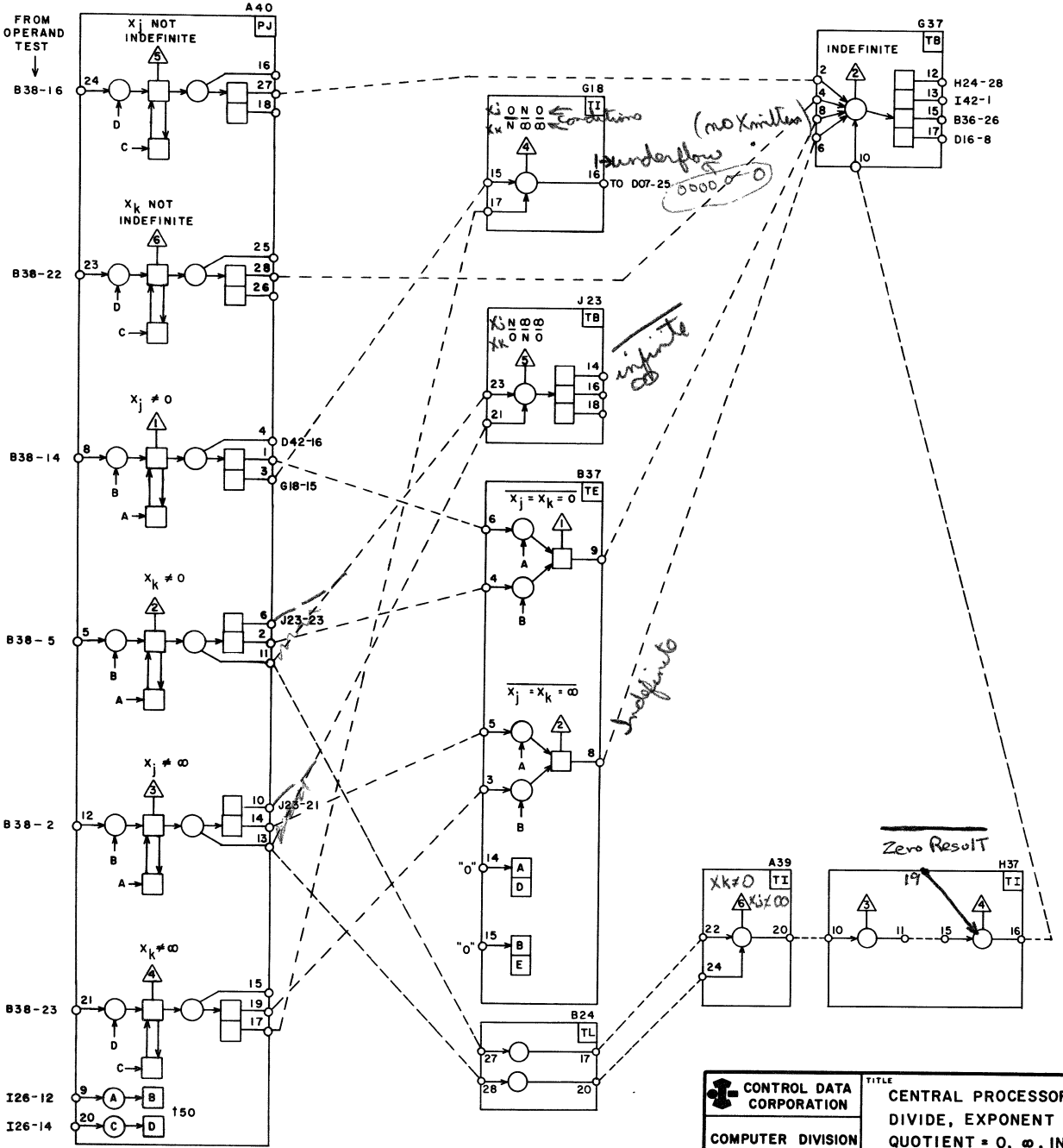
TEST RESULT

Several tests are performed in the exponent arithmetic to determine if the product is valid. The initial operands are tested to see if either X_j or X_k is indefinite, infinite, or zero.

The results of the tests are held until the end of the operation and are then used to condition the gating of the final product. An indefinite result is packed with an exponent of 1777_8 and a zero coefficient. A result of infinity is packed with an exponent of 3777_8 and a zero coefficient. A result of zero is packed with a zero exponent and a zero coefficient.



CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR	PRODUCT: 6601/04/13/14	
COMPUTER DIVISION	DIVIDE, EXPONENT TEST RESULT, INCREMENT	SIZE C · DRAWING NO. 60119300 · REV 8T	
		SHEET 191	173



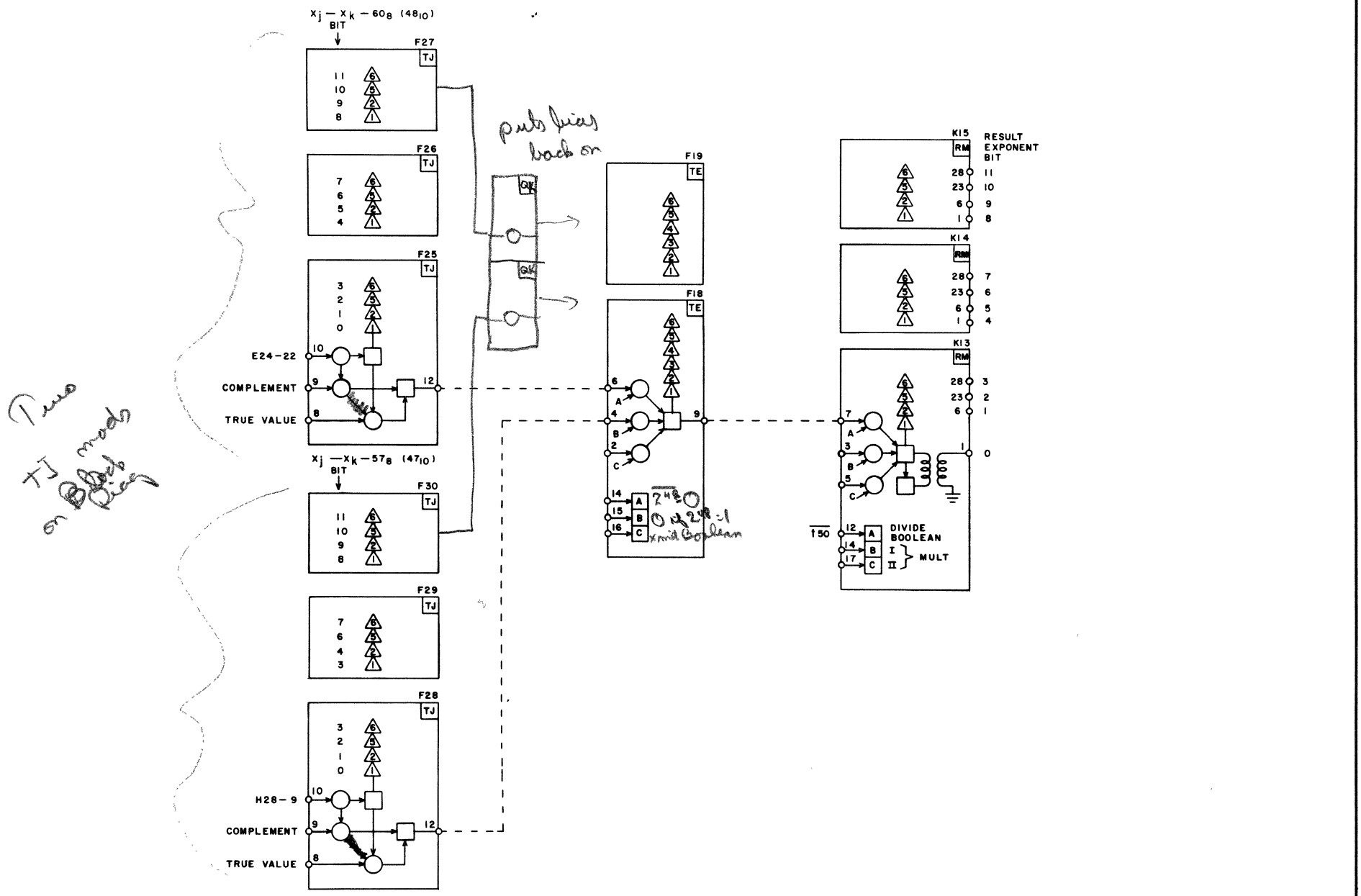
CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE
CENTRAL PROCESSOR
DIVIDE, EXPONENT
QUOTIENT = 0, ∞, INDEFINITE

PRODUCT 6601/04/13/14
SIZE C | DRAWING NO. 60119300 | REV BT
SHEET 192 | 175

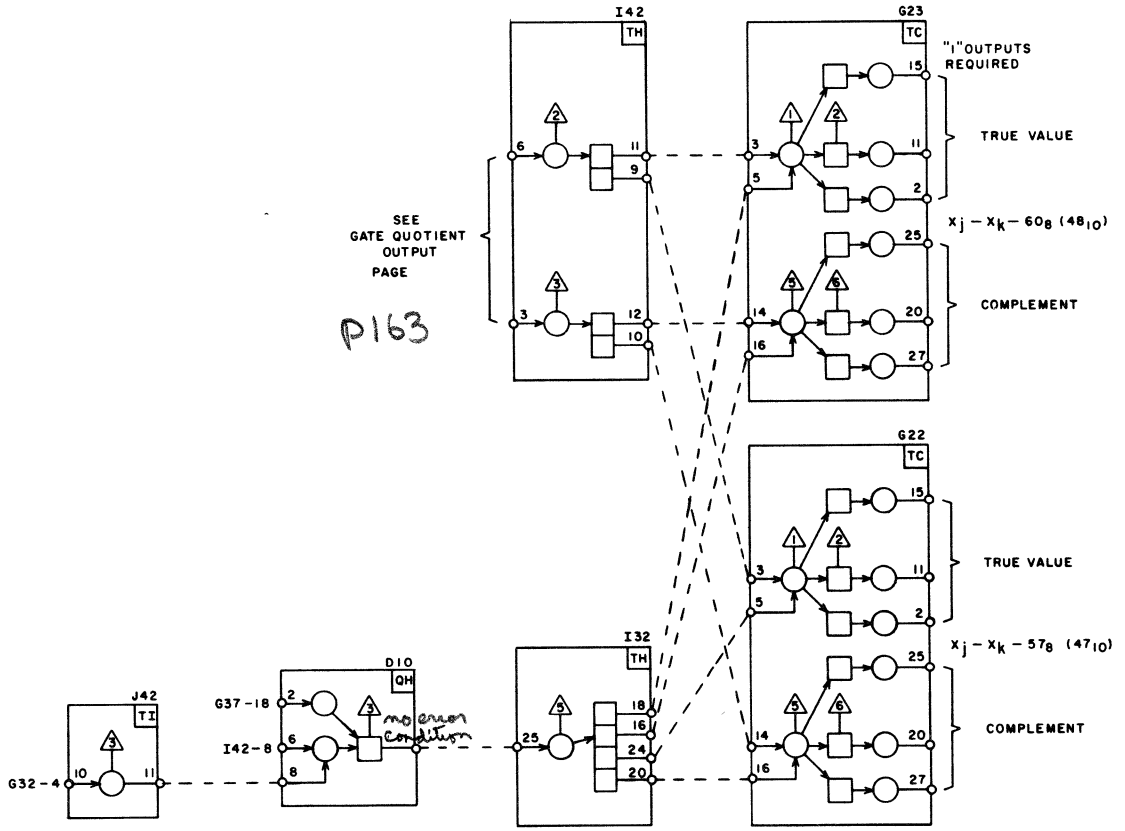
The use of normalized operands causes the divide unit to produce a normalized quotient (coefficient bit 47 equals "1"). Normalized operands automatically result in a "1" in either (or both) coefficient bits 47 and 48. If bit 48 is a "1", a 1-place right shift is performed and +1 is added to the exponent to maintain positional accuracy. Both values of the exponent, $X_j - X_k - 60_8 (48_{10})$ and $X_j - X_k - 57_8 (47_{10})$, are made available at the fan-in to the output network, and the gating selection is made according to coefficient bit 48.

The decision of whether to take the true value or the complement of the exponent is based on the comparison of the signs (exponent bit 11, operand bit 59) of the original operands X_j and X_k. If the signs are the same, the true exponent is used; if the signs are different, the exponent is complemented.



CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR DIVIDE, EXPONENT OUTPUT NETWORK	PRODUCT: 6601	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300
		REV: C	
		SHEET: 193	177

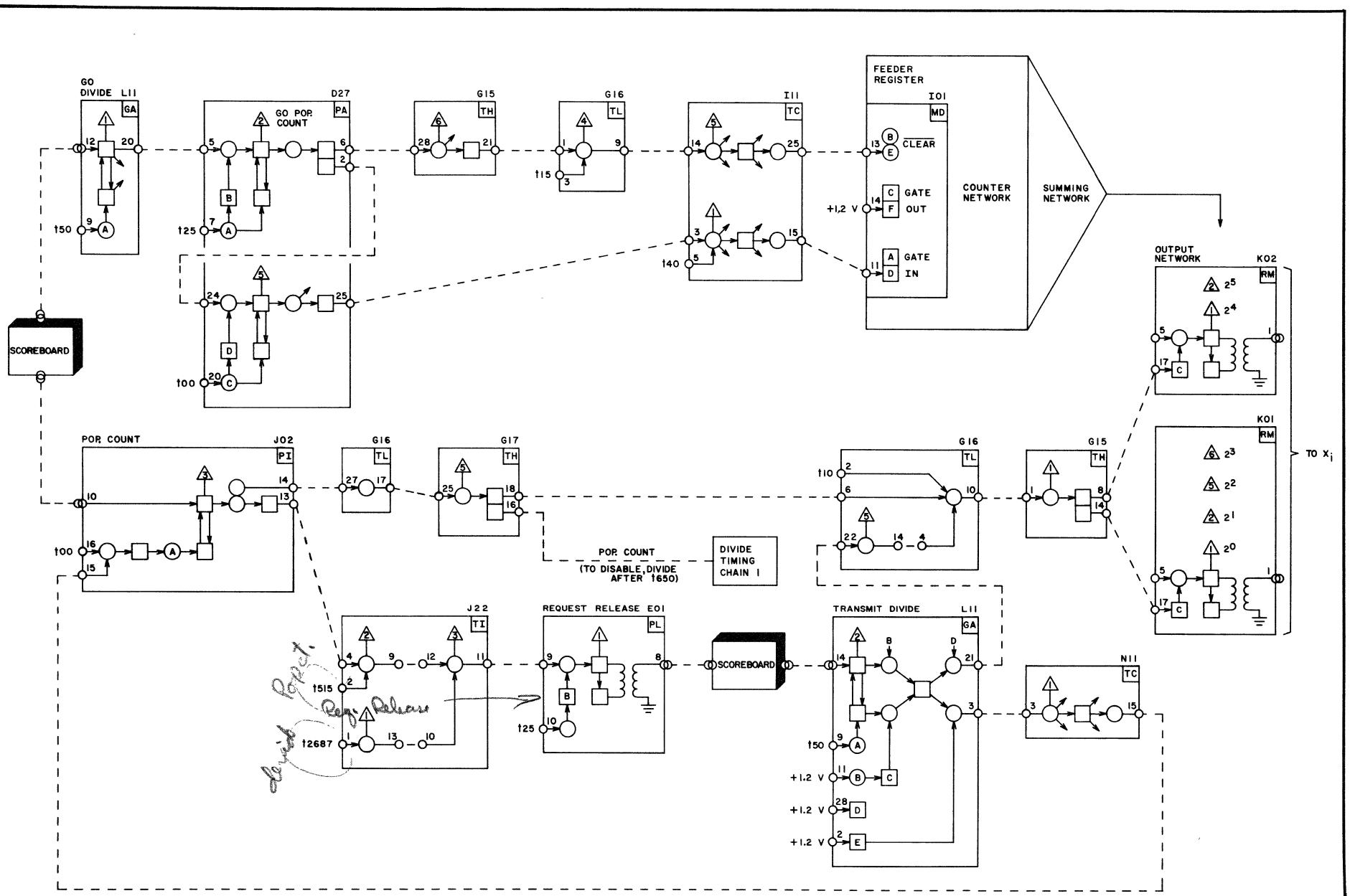
underflow -> no gates



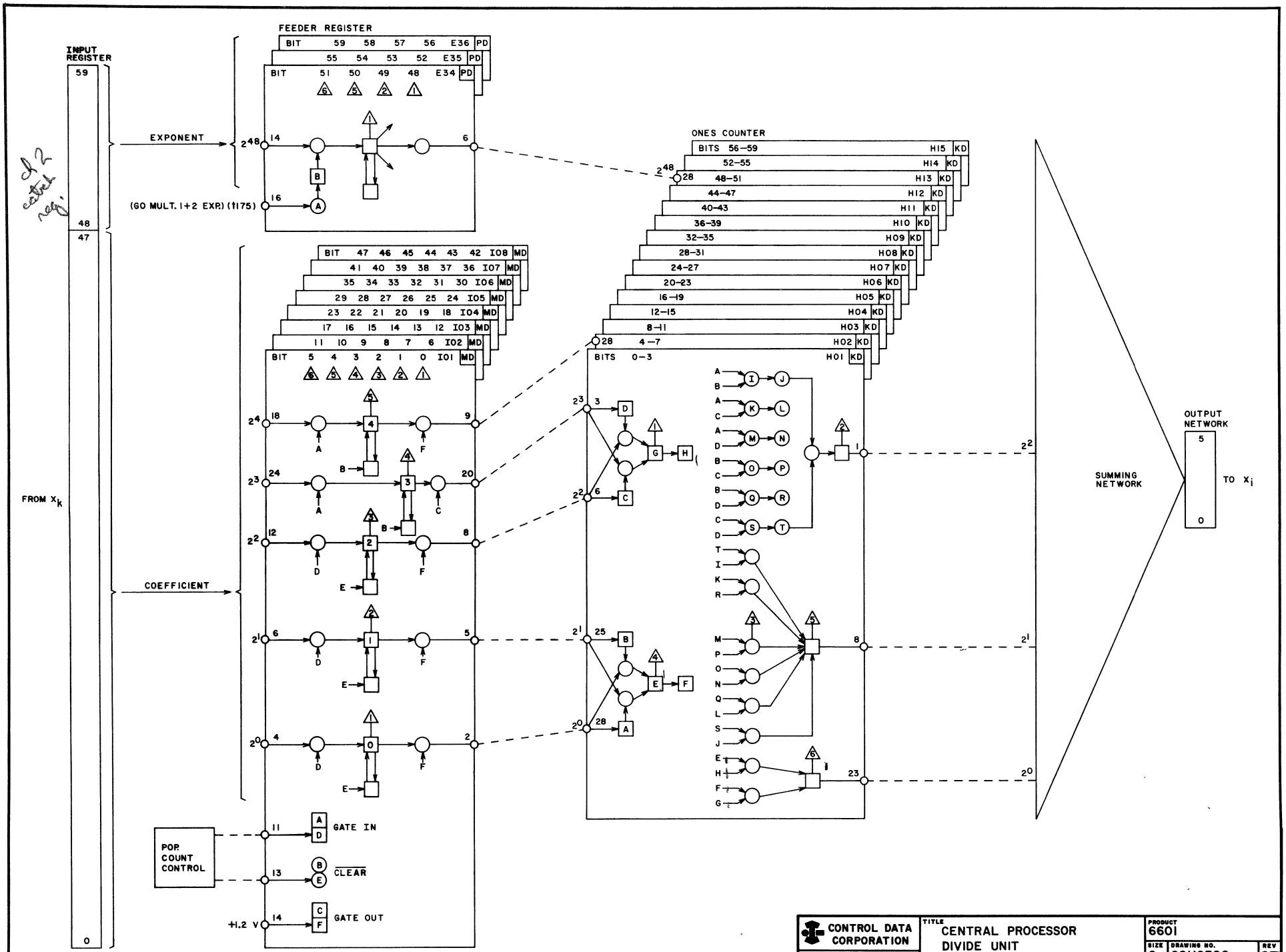
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR	PRODUCT: 6601	
COMPUTER DIVISION	DIVIDE, EXPONENT GATE RESULT OUTPUT	SIZE: C	DRAWING NO.: 60119300 REV: C
		SHEET: 194	179

POPULATION COUNT

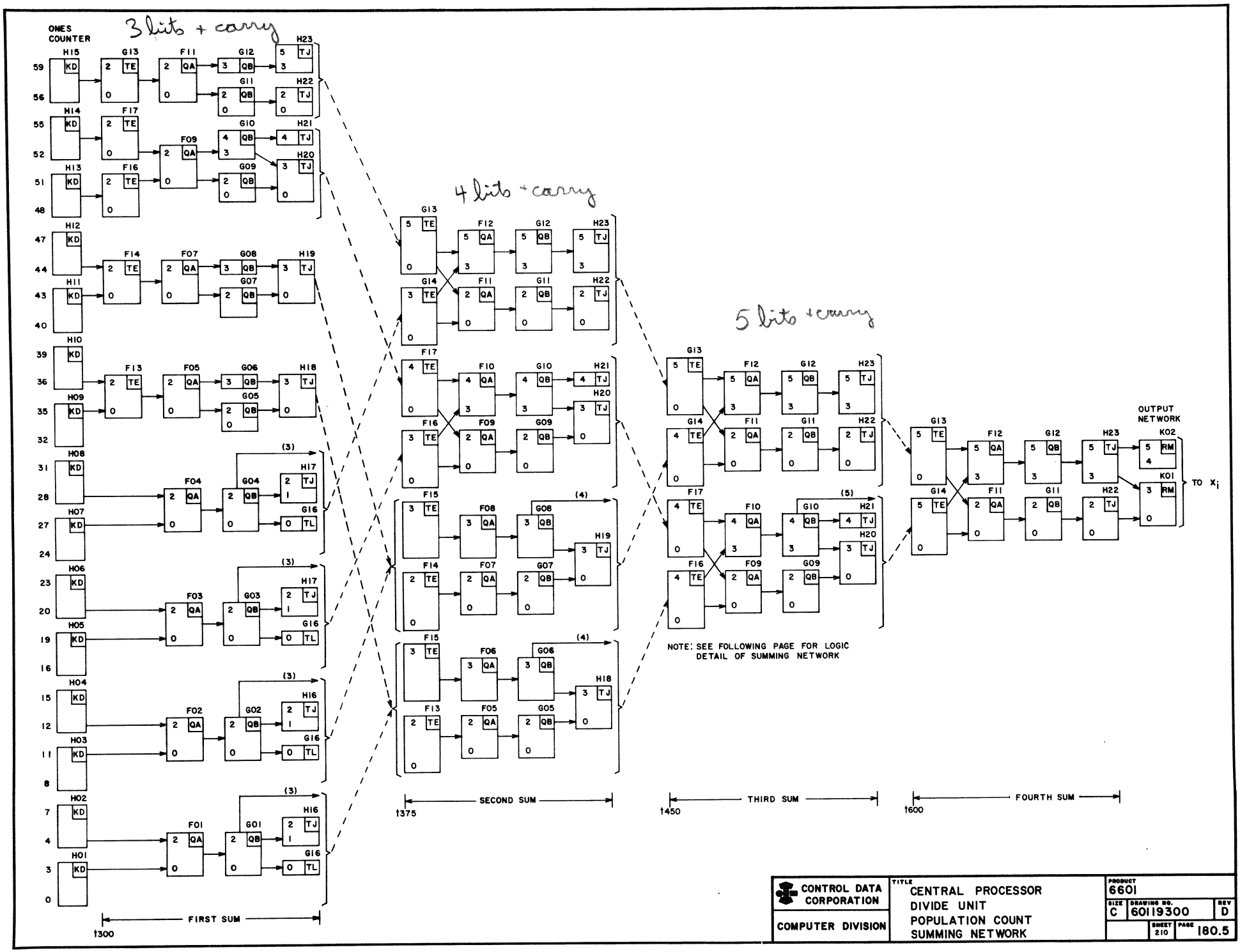
The Population Count instruction (47) counts the number of "1" bits in operand register X_k and stores the count in operand register X_i. The bits are taken four at a time in the counter network (KD) and the fifteen partial counts are added together in a four-stage summing network (QA, QB, TJ). Because $60_{10} = 74_8$, only the lower six bit positions of the chassis 2 output network are used to transmit the count to operand register X_i. The time required for a count is 0.8 microseconds (8 minor cycles).



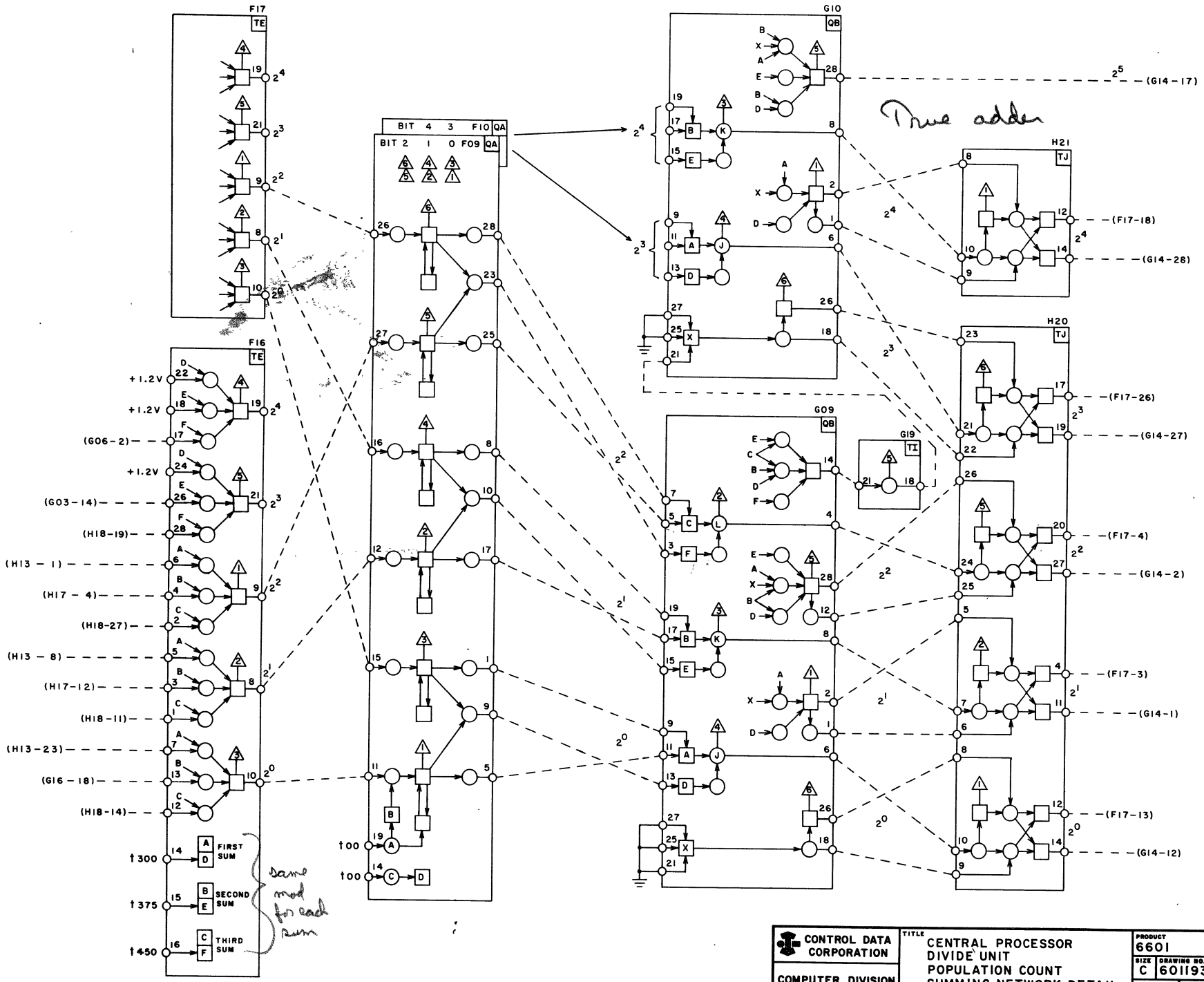
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR DIVIDE UNIT POPULATION COUNT CONTROL	PRODUCT: 6601	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: D
		SHEET: 208	PAGE: 180.1



CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR DIVIDE UNIT POPULATION COUNT COUNTER NETWORK	PRODUCT: 6601	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: BT
			SHEET: 209 PAGE: 180.3



CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR DIVIDE UNIT POPULATION COUNT SUMMING NETWORK	PRODUCT 6601			
COMPUTER DIVISION		SIZE C	DRAWING NO. 60119300	REV D	
			SHEET 210	PAGE 180.5	

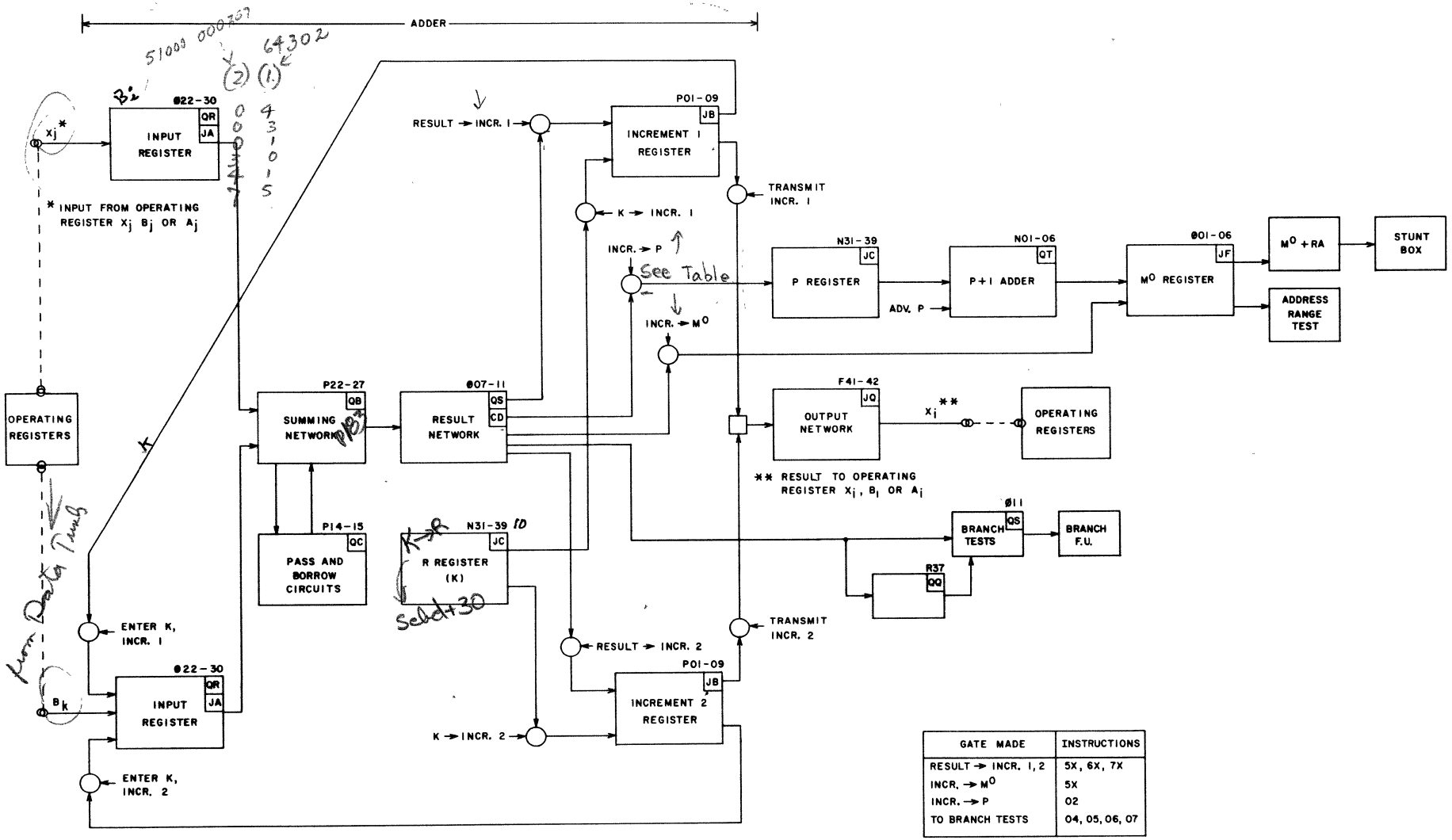


CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR DIVIDE UNIT POPULATION COUNT SUMMING NETWORK DETAIL	PRODUCT: 6601	
COMPUTER DIVISION		SIZE: C · DRAWING NO.: 60119300 · REV: BT	
		SHEET: 211 · PAGE: 180.7	

Incrementing and memory addressing in the CONTROL DATA 6601 and 6604 Central Computers are performed by two identical Increment Units on chassis 5. Each of these units contains all necessary registers and logic elements to form the 18-bit, fixed point sum (or difference) of two 18-bit, fixed point operands. The time required for an operation is 0.3 microseconds (3 minor cycles).

An Increment instruction directs the unit either to 1) add the addend B_k or K to the augend X_j, B_j, or A_j and send the sum to X_i, B_i, or A_i or 2) subtract the subtrahend B_k from the minuend B_j or A_j and send the difference to X_i, B_i, or A_i. Operands obtained from an X_j operand register are the truncated lower 18-bits of the 60-bit word. Conversely, an 18-bit result placed in an X_i operand register carries the sign bit extended to the remaining bits of the 60-bit word.

Note that the results of all 5X instructions are transmitted to one of seven address registers (A1 - A7). An immediate memory reference is caused by any change in the contents of an A register. When A1 - A5 is changed, the operand is read from the new address specified and sent to the corresponding operand register X1 - X5. When A6 or A7 is changed, the operand from the corresponding operand register X6 or X7 and stored at the new address specified.



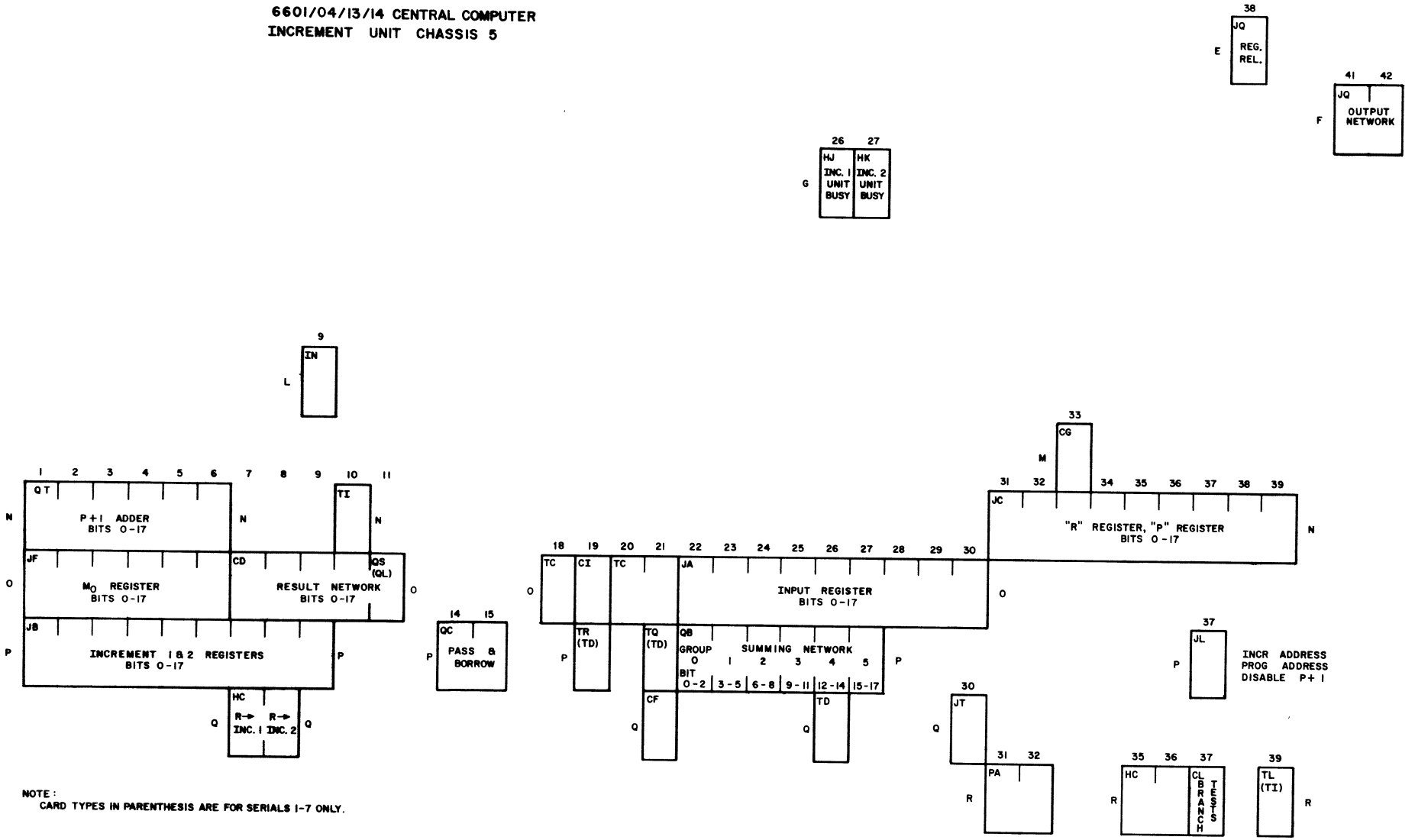
GATE MADE	INSTRUCTIONS
RESULT → INCR. 1, 2	5X, 6X, 7X
INCR. → M0	5X
INCR. → P	02
TO BRANCH TESTS	04, 05, 06, 07

BRANCH TEST INSTRUCTIONS:

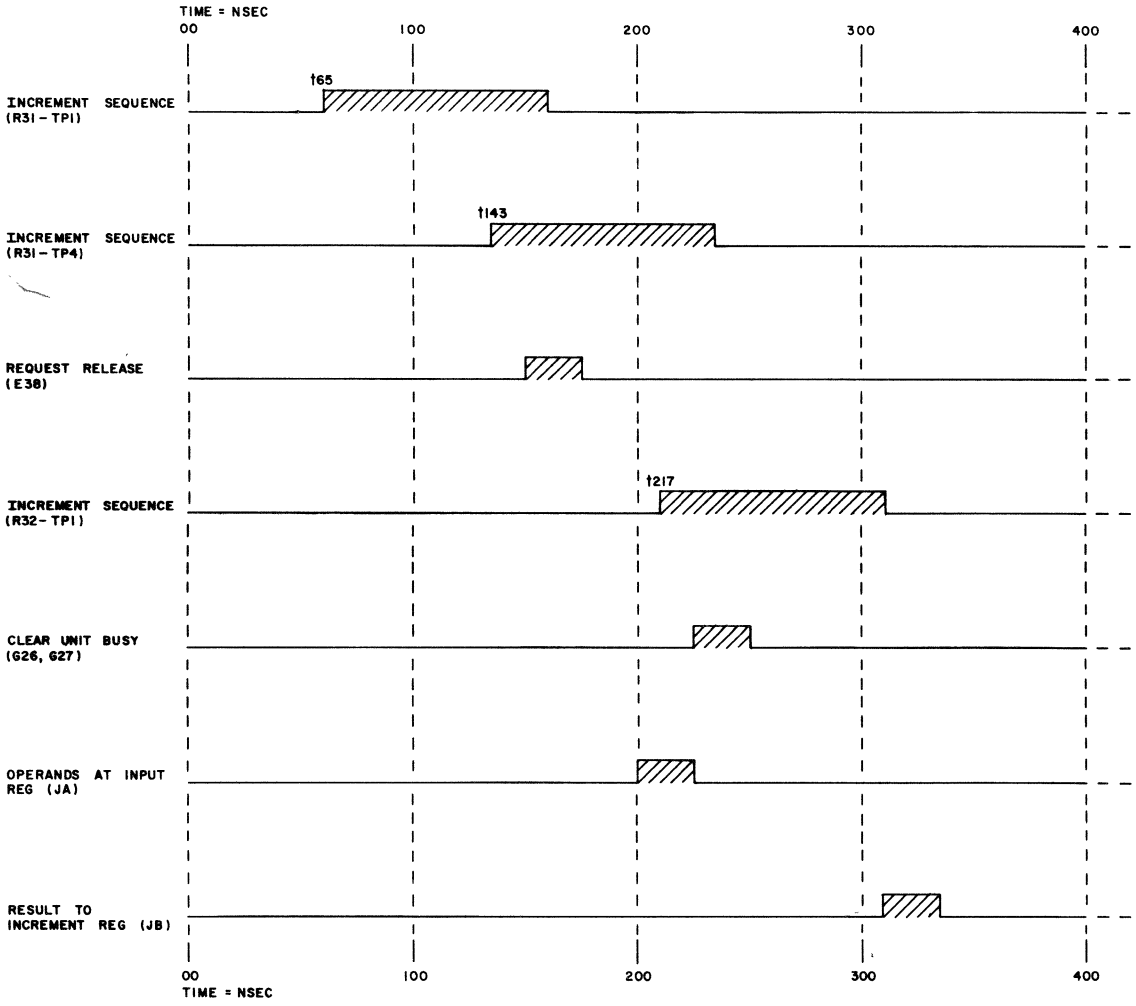
02	04 – 07
i OPERAND USES j TRUNK	
K FIELD ENTERS VIA INCR. REG.	j OPERAND USES k TRUNK

CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR INCREMENT UNIT BLOCK DIAGRAM	PRODUCT 6601/04/13/14	
		SIZE C	DRAWING NO. 60119300
		REV B	
		SHEET 195	181

6601/04/13/14 CENTRAL COMPUTER
INCREMENT UNIT CHASSIS 5



NOTE:
CARD TYPES IN PARENTHESIS ARE FOR SERIALS 1-7 ONLY.



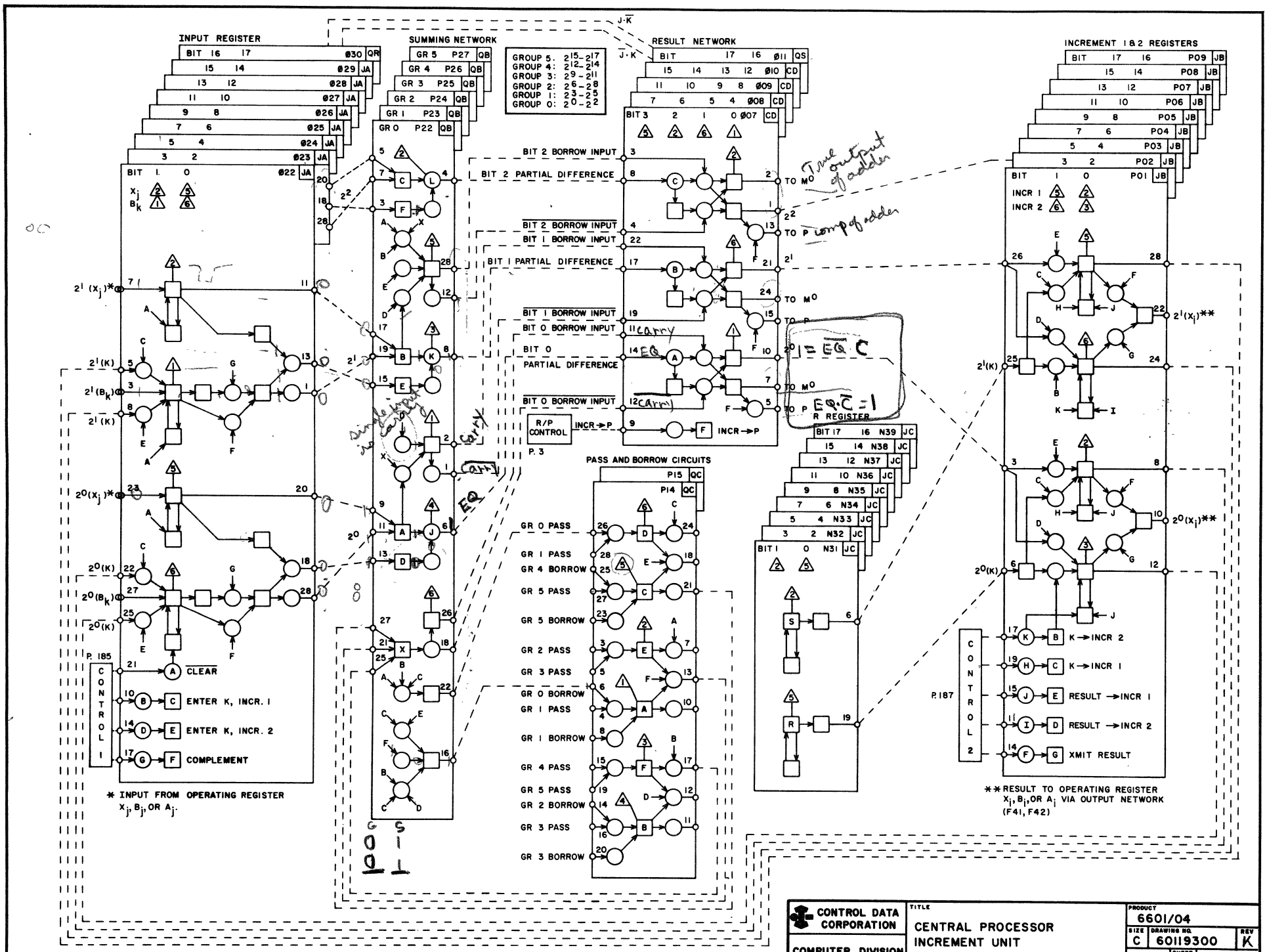
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR INCREMENT UNIT TIMING CHART	PRODUCT: 6601/04/13/14	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300
		REV: M	
		SHEET: 206	PAGE: 182.1

ADDER

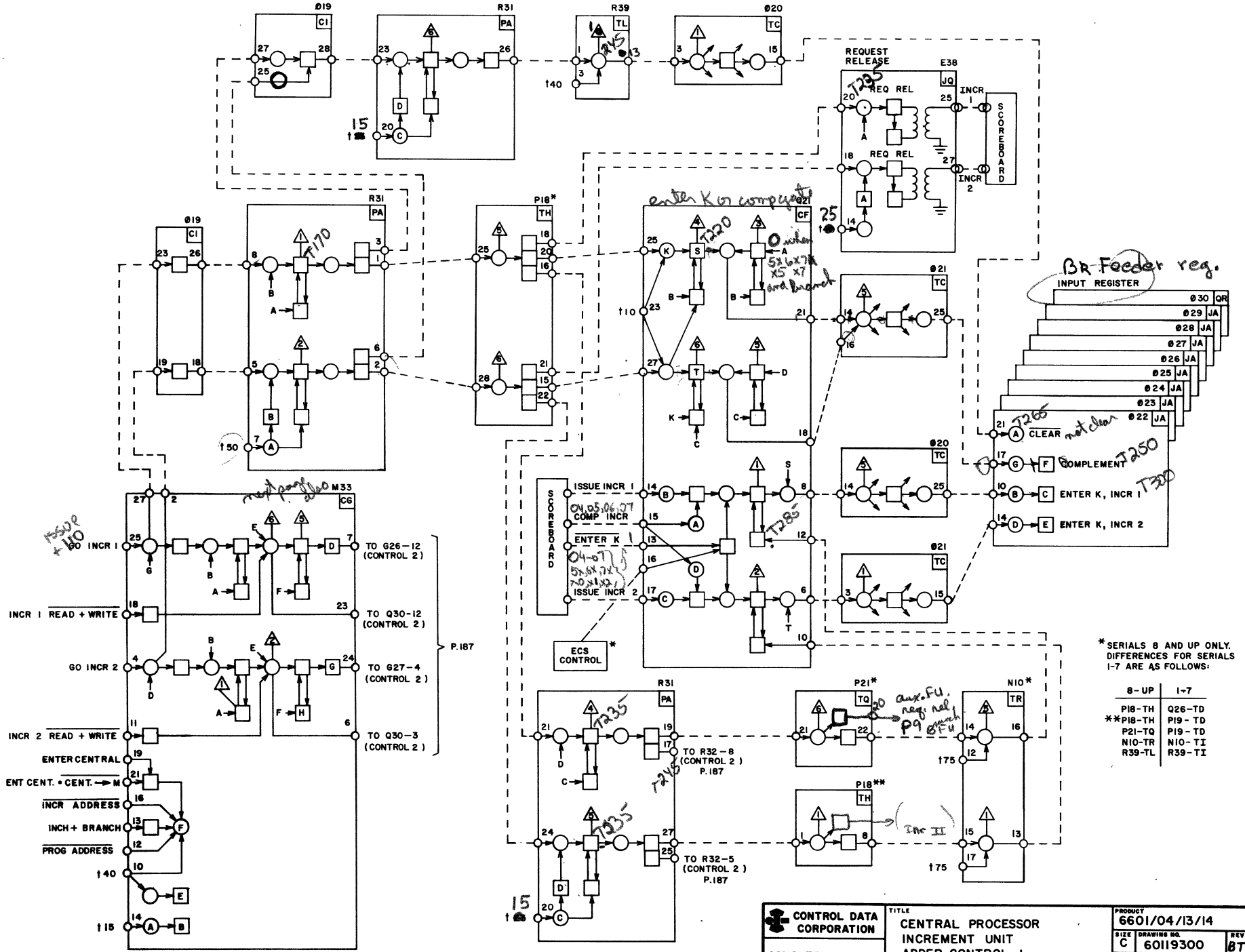
The Increment Unit performs fixed point addition and subtraction of 18-bit numbers. Negative numbers are represented in one's complement notation. The sign bit is in the high order bit position (bit 17), and the binary point is at the right of the lowest order bit position (bit 0).

The adder is divided into two levels of operation, with two bits making up a group and six groups making up the entire adder.

The adder is subtractive in nature, but because of the logic configuration of the input register (JA), the true value of B_k or K is gated for Add instructions and the one's complement value for Subtract instructions.



CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR INCREMENT UNIT ADDER	PRODUCT: 6601/04	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300 REV: K
		SHEET: 196	183



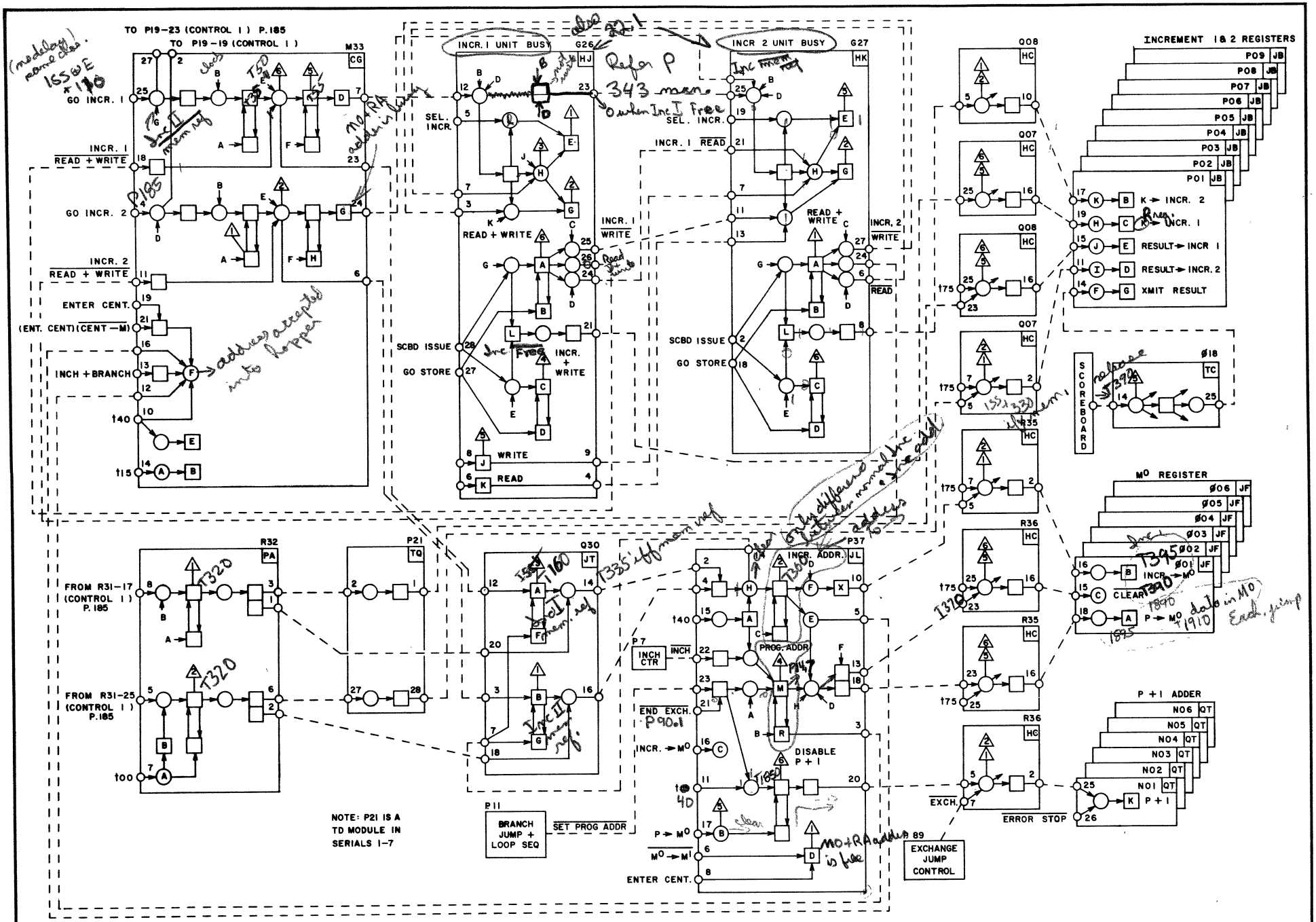
*SERIALS 8 AND UP ONLY. DIFFERENCES FOR SERIALS 1-7 ARE AS FOLLOWS:

8-UP	1-7
P18-TH	Q26-TD
**P18-TH	P19-TD
P21-TQ	P19-TD
N10-TR	N10-TI
R39-TL	R39-TI

CONTROL DATA CORPORATION
COMPUTER DIVISION

TITLE
CENTRAL PROCESSOR
INCREMENT UNIT
ADDER CONTROL 1

PRODUCT 6601/04/13/14
SIZE C | DRAWING NO. 60119300 | REV BT
SHEET 197 | 185

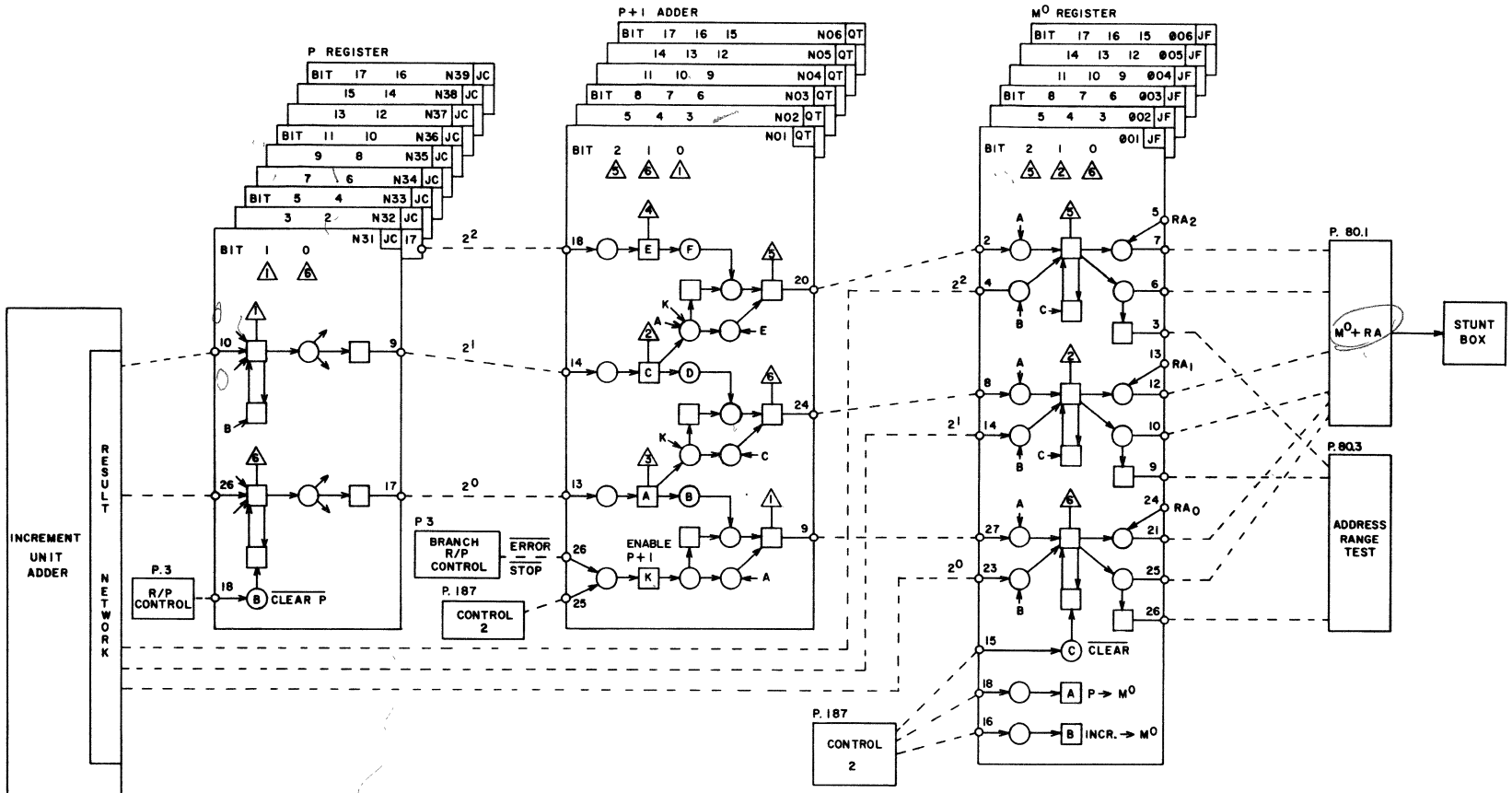


CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR INCREMENT UNIT ADDER CONTROL 2	PRODUCT 6601/04/13/14	
		SIZE C	DRAWING NO. 60119300 REV BJ
			SHEET 198 187.

The Increment Units are also used as auxiliaries to the Branch Unit, both for the Unconditional Jump instruction 02, and the Conditional Jump instructions 04-07.

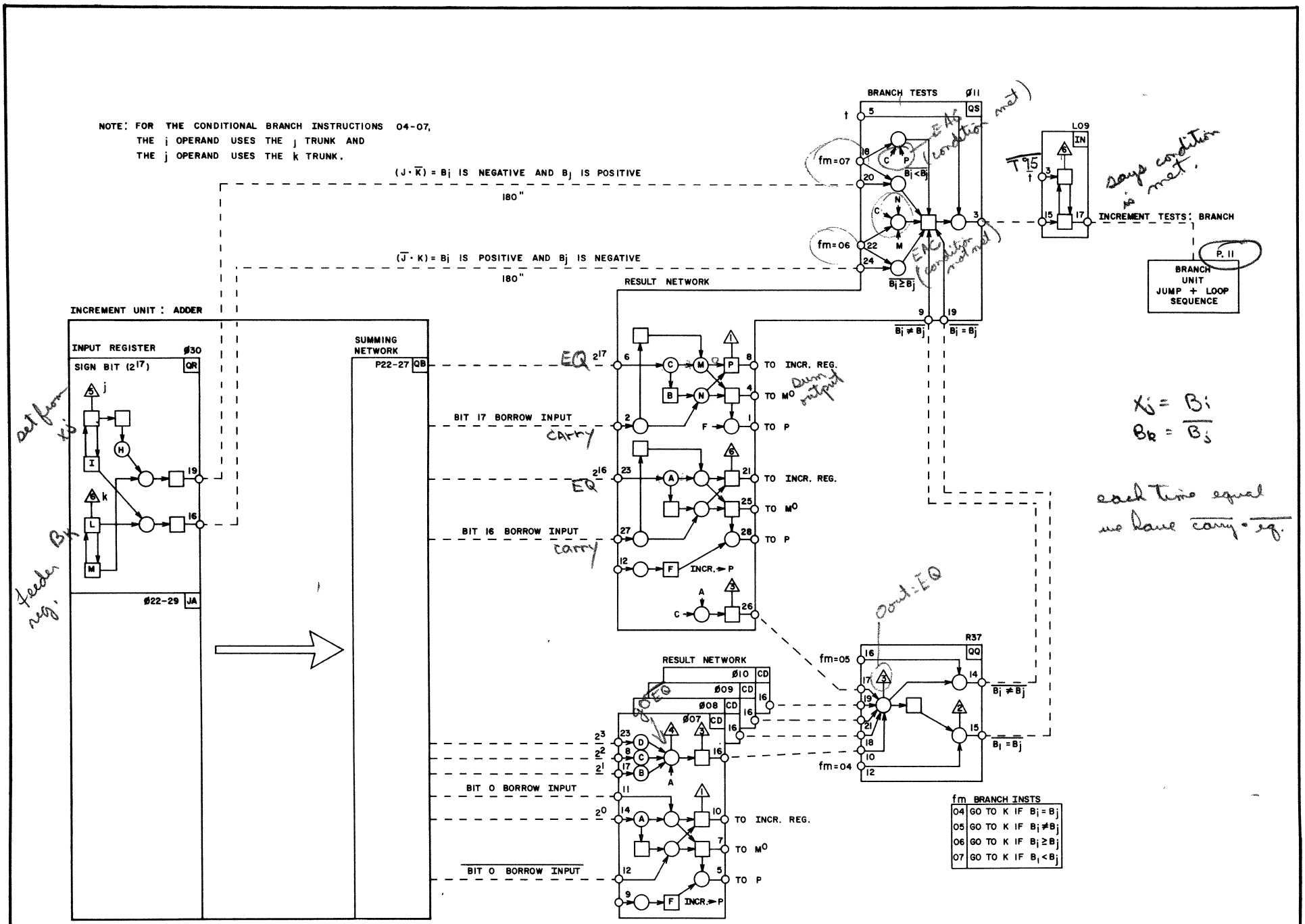
Op. Code	Instruction
02	Go to $B_i + K$
04	Go to K if $B_i = B_j$
05	Go to K if $B_i \neq B_j$
06	Go to K if $B_i \geq B_j$
07	Go to K if $B_i < B_j$

For the Unconditional Jump instruction 02, the Increment Unit adder computes the sum of the two operands and gates the result to the P register. The P + 1 adder is disabled and the result passes through to M^0 and on to the Stunt Box.



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR INCREMENT UNIT P → P+1 → M0	PRODUCT 6601/04	
		SIZE C	DRAWING NO. 60119300 REV BJ
			SHEET 199 189

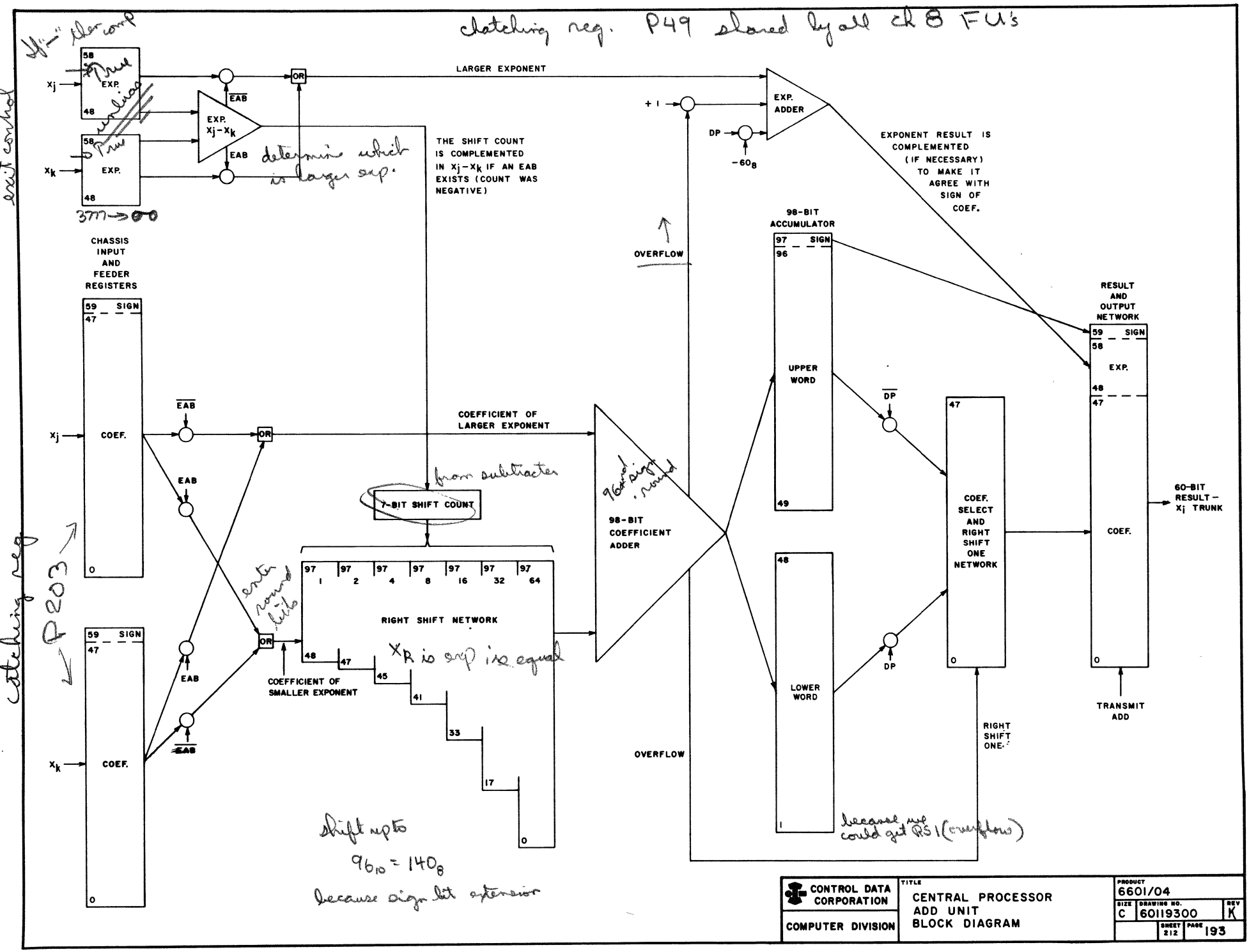
For the conditional tests of equality (04 and 05), the two 18-bit operands are compared bit for bit. For the conditional tests of magnitude (06 and 07), the signs (bit 17) of the two operands are compared. If they are different, the positive number is recognized as the greater of the two. If they are the same, the one's complement difference ($B_i - B_j$) is taken, and the sign (bit 17) of the result is examined. If it is zero, $B_i \geq B_j$; if it is one, $B_i < B_j$. A control bit is then sent to the Branch Unit indicating the success or failure of the test.

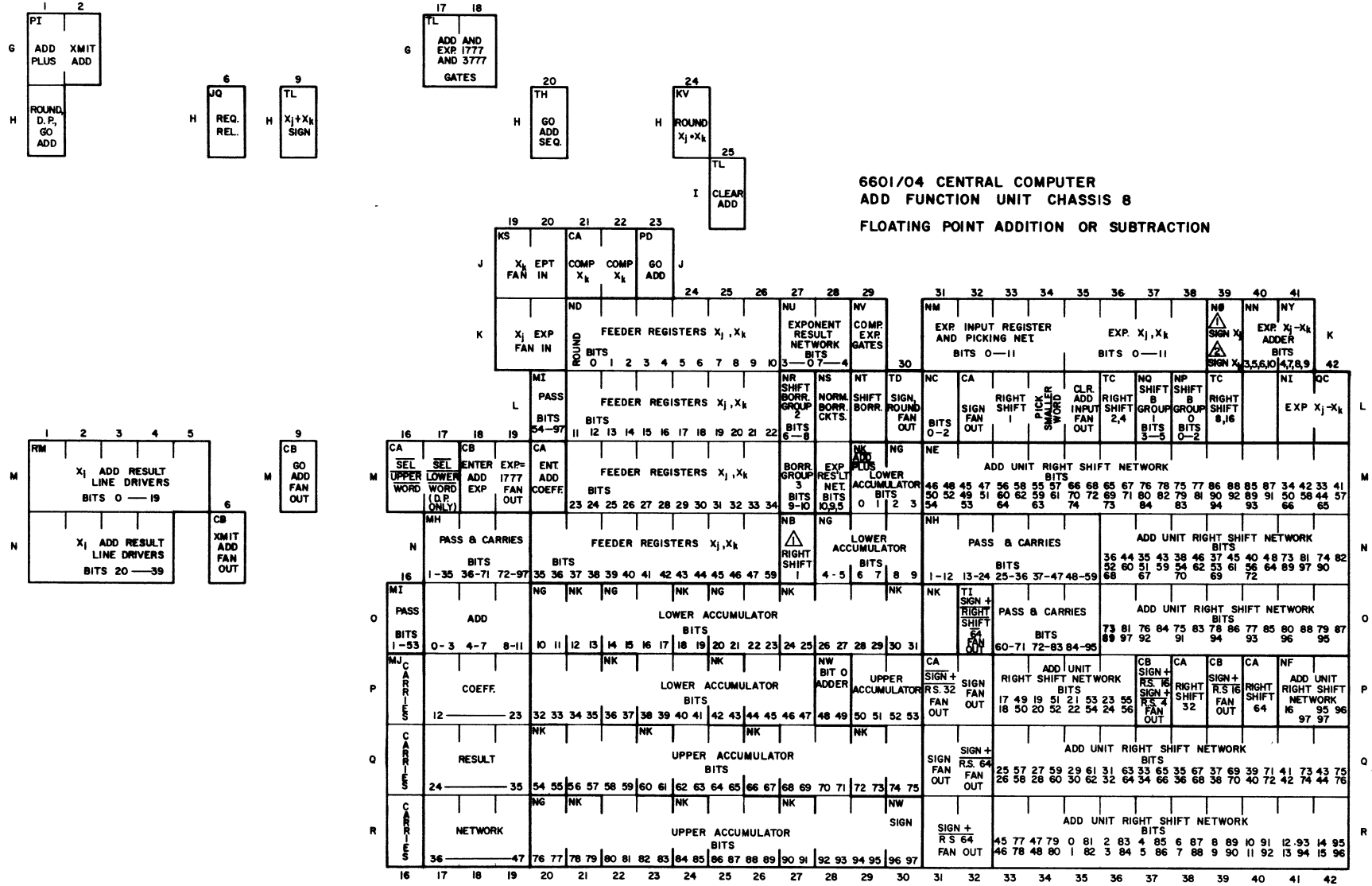


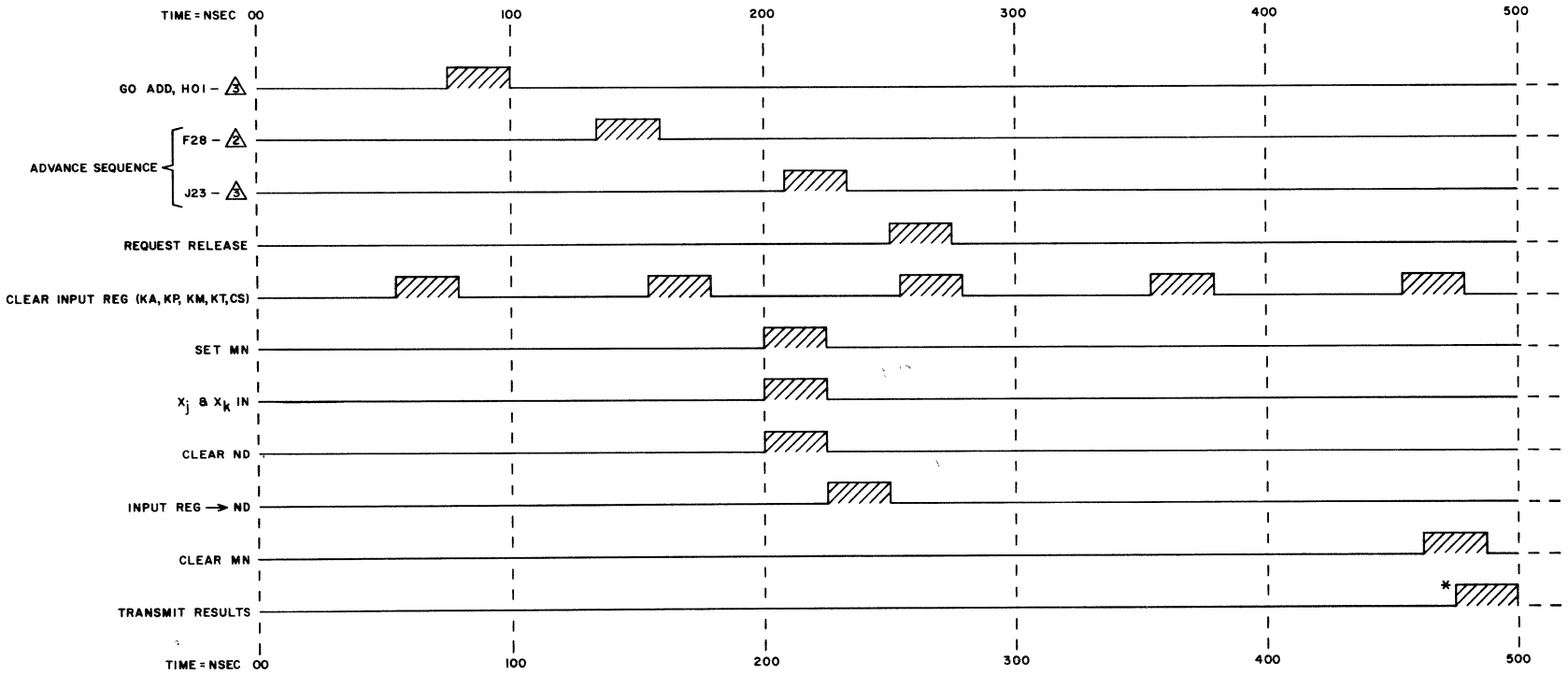
CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR INCREMENT UNIT BRANCH TESTS	PRODUCT 6601/04	
COMPUTER DIVISION		SIZE C	DRAWING NO. 60119300 REV K
			SHEET 200 PAGE 191

Floating point addition and subtraction in the CONTROL DATA 6601 and 6604 Central Processor are performed by the Add Unit located on chassis 8. This unit contains all the hardware necessary to perform the 60-bit, floating point sum or difference of two 60-bit, floating point operands. The time required for a single addition or subtraction is 0.4 microseconds (4 minor cycles).

The add instruction directs the Add Unit to add or subtract the two operands and give either a single or double precision result. The unit may also be directed to round the operands when a single precision result is requested.

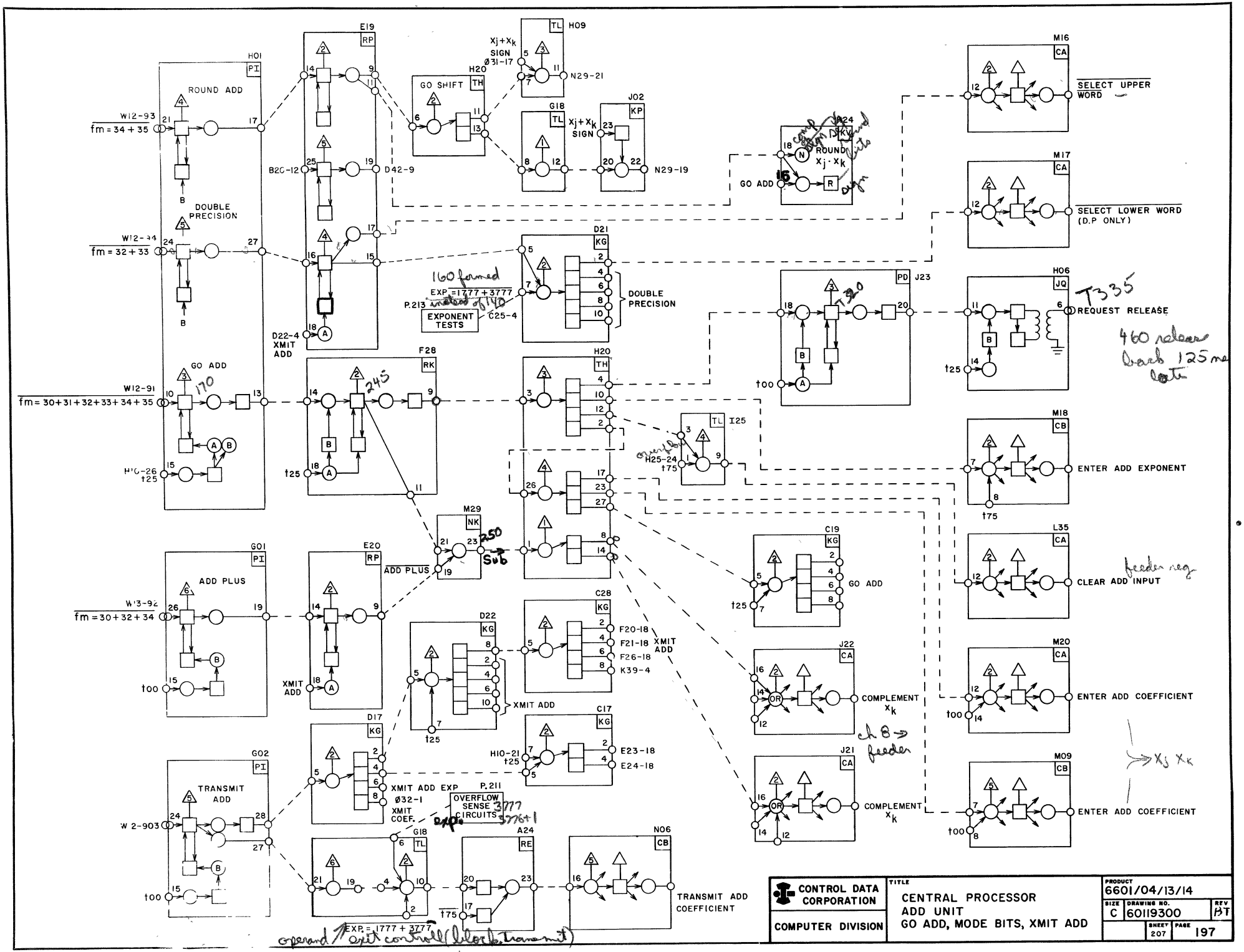

LARGER EXPONENT
EXP. ADDER
+1
DP
-60_8
EXPONENT RESULT IS COMPLEMENTED (IF NECESSARY) TO MAKE IT AGREE WITH SIGN OF COEF.
THE SHIFT COUNT IS COMPLEMENTED IN $X_j - X_k$ IF AN EAB EXISTS (COUNT WAS NEGATIVE)
EXP. $X_j - X_k$
EAB
OR
EXP.
X_j
X_k
CHASSIS INPUT AND FEEDER REGISTERS
59 SIGN
47
COEF.
0
COEFFICIENT OF LARGER EXPONENT
COEFFICIENT OF SMALLER EXPONENT
7-BIT SHIFT COUNT
RIGHT SHIFT NETWORK
97 1
97 2
97 4
97 8
97 16
97 32
97 64
48
47
45
41
33
17
0
98-BIT COEFFICIENT ADDER
OVERFLOW
98-BIT ACCUMULATOR
97 SIGN
96
UPPER WORD
49
LOWER WORD
48
1
COEF. SELECT AND RIGHT SHIFT ONE NETWORK
RIGHT SHIFT ONE
RESULT AND OUTPUT NETWORK
59 SIGN
58
EXP.
48
47
COEF.
0
60-BIT RESULT – X_j TRUNK
TRANSMIT ADD
CONTROL DATA CORPORATION
COMPUTER DIVISION
TITLE
CENTRAL PROCESSOR ADD UNIT BLOCK DIAGRAM
PRODUCT 6601/04
SIZE C
DRAWING NO. 60119300
REV K
SHEET 212
PAGE 193


6601/04 CENTRAL COMPUTER
ADD FUNCTION UNIT CHASSIS 8
FLOATING POINT ADDITION OR SUBTRACTION
PI
ADD PLUS
XMIT ADD
ROUND, D.P., GO ADD
JQ
REQ. REL.
TL
Xj+Xk SIGN
TL
ADD AND EXP. 1777 AND 3777 GATES
TH
GO ADD SEQ.
KV
ROUND Xj•Xk
TL
CLEAR ADD
KS
Xk FAN IN
CA
COMP Xk
COMP Xk
PD
GO ADD
Xj EXP FAN IN
ND
ROUND
FEEDER REGISTERS Xj, Xk
BITS 0 1 2 3 4 5 6 7 8 9 10
NU
EXPONENT RESULT NETWORK BITS 3—0 7—4
NV
COMP. EXP. GATES
NM
EXP. INPUT REGISTER AND PICKING NET
BITS 0—11
EXP. Xj, Xk
BITS 0—11
NB
SIGN Xj
SIGN Xk
NN
NM
EXP. Xj-Xk ADDER
BITS 4,7,8,9
3,5,6,10
MI
PASS
BITS 54—97
FEEDER REGISTERS Xj, Xk
BITS 11 12 13 14 15 16 17 18 19 20 21 22
NR
SHIFT BORR. GROUP 2
BITS 6—8
NS
NORM. BORR. CKTS.
NT
SHIFT BORR.
TD
SIGN, ROUND FAN OUT
NC
BITS 0-2
CA
SIGN FAN OUT
RIGHT SHIFT 1
PICK SMALLER WORD
CLR. ADD INPUT FAN OUT
TC
RIGHT SHIFT 2,4
NQ
SHIFT B GROUP 1
BITS 3—5
NP
SHIFT B GROUP 0
BITS 0—2
TC
RIGHT SHIFT 8,16
NI
QC
EXP Xj-Xk
CA
SEL UPPER WORD
SEL LOWER WORD (D.P. ONLY)
CB
ENTER ADD EXP
EXP= 1777 FAN OUT
CA
ENT. ADD COEFF.
FEEDER REGISTERS Xj, Xk
BITS 23 24 25 26 27 28 29 30 31 32 33 34
BORR. GROUP 3
BITS 9-10
EXP RES'LT NET. BITS 10,9,5
NK ADD PLUS
NG
LOWER ACCUMULATOR
BITS 0 1 2 3
NE
ADD UNIT RIGHT SHIFT NETWORK BITS
46 48 45 47 56 58 55 57 66 68 65 67 76 78 75 77 86 88 85 87 34 42 33 41
50 52 49 51 60 62 59 61 70 72 69 71 80 82 79 81 90 92 89 91 50 58 44 57
54 53 64 63 74 73 84 83 94 93 66 65
MH
PASS & CARRIES
BITS 1-35 36-71 72-97
FEEDER REGISTERS Xj, Xk
BITS 35 36 37 38 39 40 41 42 43 44 45 46 47 59
NB
RIGHT SHIFT 1
NG
LOWER ACCUMULATOR
BITS 4-5 6 7 8 9
NH
PASS & CARRIES
BITS 1-12 13-24 25-36 37-47 48-59
ADD UNIT RIGHT SHIFT NETWORK BITS
36 44 35 43 38 46 37 45 40 48 73 81 74 82
52 60 51 59 54 62 53 61 56 64 89 97 90
68 67 70 69 72
MI
PASS
BITS 1-53
ADD
0-3 4-7 8-11
NG NK NG NK NG NK NK
LOWER ACCUMULATOR BITS
10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
NK
TI
SIGN + RIGHT SHIFT 64 FAN OUT
PASS & CARRIES
BITS 60-71 72-83 84-95
ADD UNIT RIGHT SHIFT NETWORK BITS
73 81 76 84 75 83 78 86 77 85 80 88 79 87
69 97 92 91 94 93 96 95
MJ
CARRIES
COEFF.
12 ——— 23
NK NK NK
LOWER ACCUMULATOR BITS
32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53
NW
BIT 0 ADDER
UPPER ACCUMULATOR
CA
SIGN + RS. 32 FAN OUT
SIGN FAN OUT
ADD UNIT RIGHT SHIFT NETWORK BITS
17 49 19 51 21 53 23 55
18 50 20 52 22 54 24 56
CB
SIGN + RS. 16
SIGN + RS. 4 FAN OUT
CA
RIGHT SHIFT 32
CB
SIGN + RS 16 FAN OUT
CA
RIGHT SHIFT 64
NF
ADD UNIT RIGHT SHIFT NETWORK
16 95 96
97 97
CARRIES
RESULT
24 ——— 35
NK NK NK NK
UPPER ACCUMULATOR BITS
54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75
SIGN FAN OUT
SIGN + RS. 64 FAN OUT
ADD UNIT RIGHT SHIFT NETWORK BITS
25 57 27 59 29 61 31 63 33 65 35 67 37 69 39 71 41 73 43 75
26 58 28 60 30 62 32 64 34 66 36 68 38 70 40 72 42 74 44 76
CARRIES
NETWORK
36 ——— 47
NG NK NK NK NK NW
UPPER ACCUMULATOR BITS
SIGN
76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97
SIGN + RS 64 FAN OUT
ADD UNIT RIGHT SHIFT NETWORK BITS
45 77 47 79 0 81 2 83 4 85 6 87 8 89 10 91 12 93 14 95
46 78 48 80 1 82 3 84 5 86 7 88 9 90 11 92 13 94 15 96
RM
Xi ADD RESULT LINE DRIVERS
BITS 0 —— 19
Xi ADD RESULT LINE DRIVERS
BITS 20 —— 39
CB
XMIT ADD FAN OUT
CB
GO ADD FAN OUT

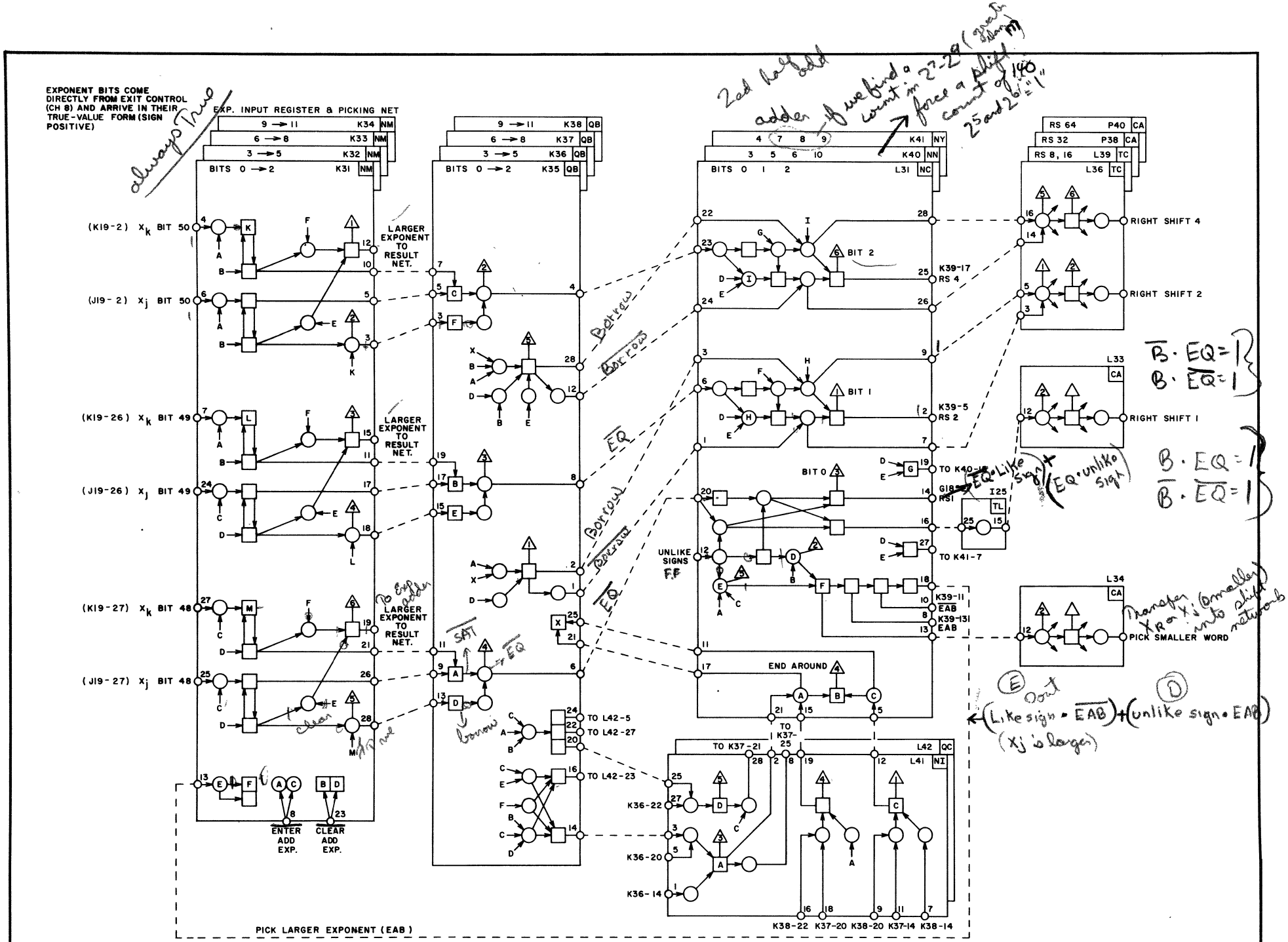


* EARLIEST POSSIBLE TIME — NO RESULT REGISTER CONFLICT

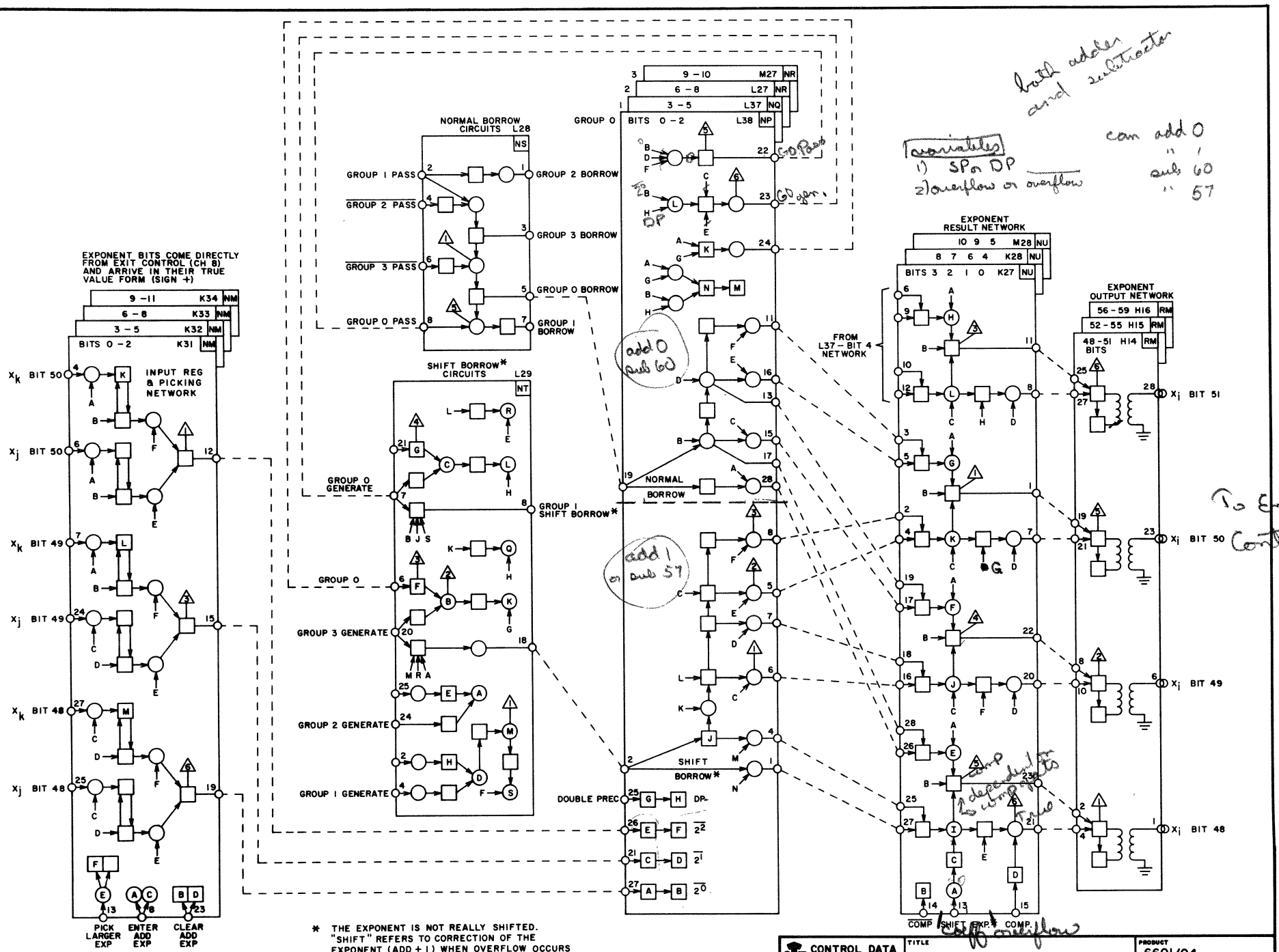
CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR ADD UNIT TIMING CHART	PRODUCT 6601 SIZE C · DRAWING NO. 60119300 · REV D SHEET 215 · PAGE 195	



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR ADD UNIT GO ADD, MODE BITS, XMIT ADD	PRODUCT 6601/04/13/14	
		SIZE C	DRAWING NO. 60119300 REV BT
		SHEET 207	PAGE 197



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR ADD UNIT EXPONENT $X_j - X_k$, SHIFT COUNT, PICK SMALLER WORD	PRODUCT 6601 SIZE C · DRAWING NO. 60119300 · REV BT SHEET 220 · PAGE 199	



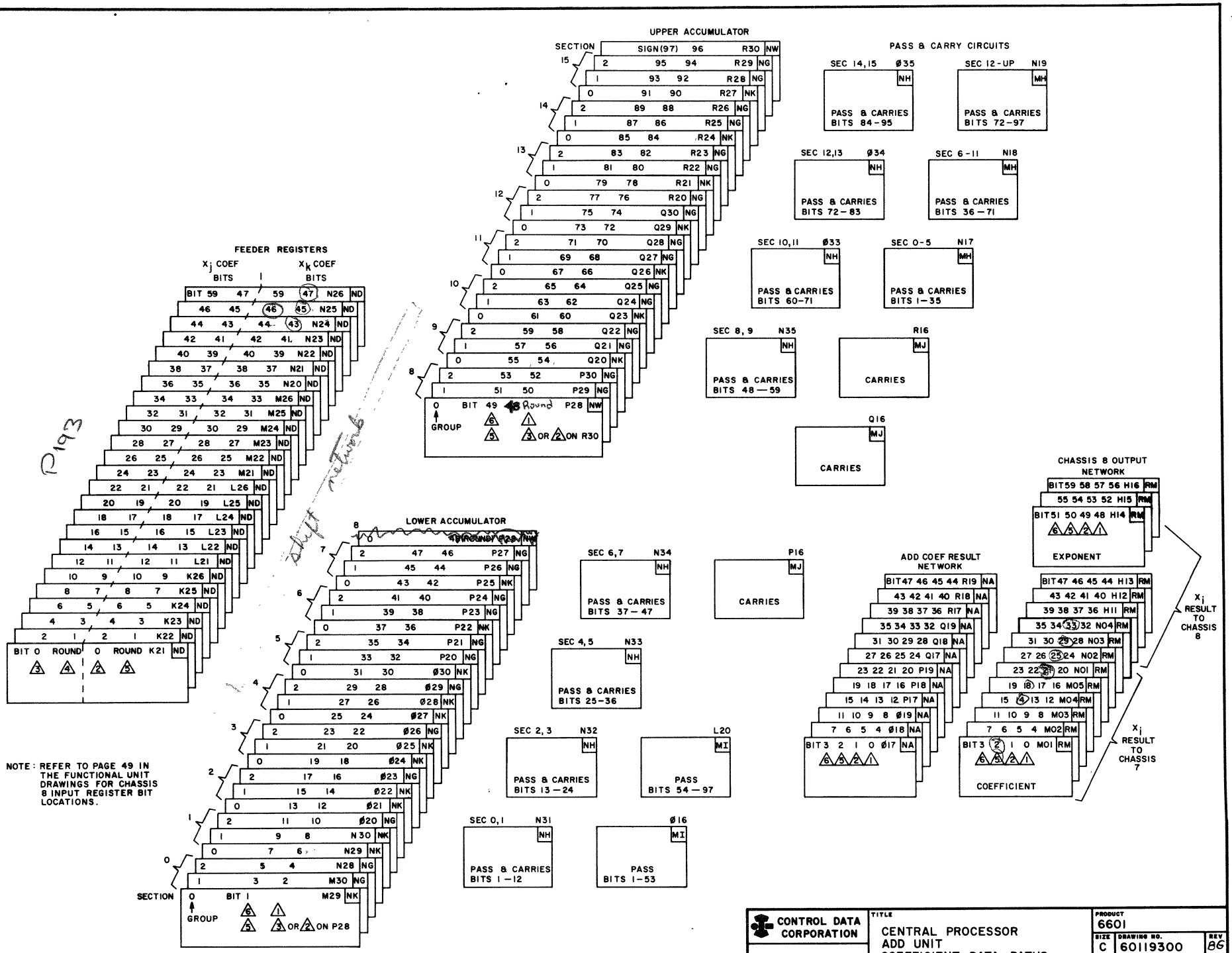
both adders and subtractor

variables
1) SP or DP
2) overflow or overflow

can add 0
" 1
sub 60
" 57

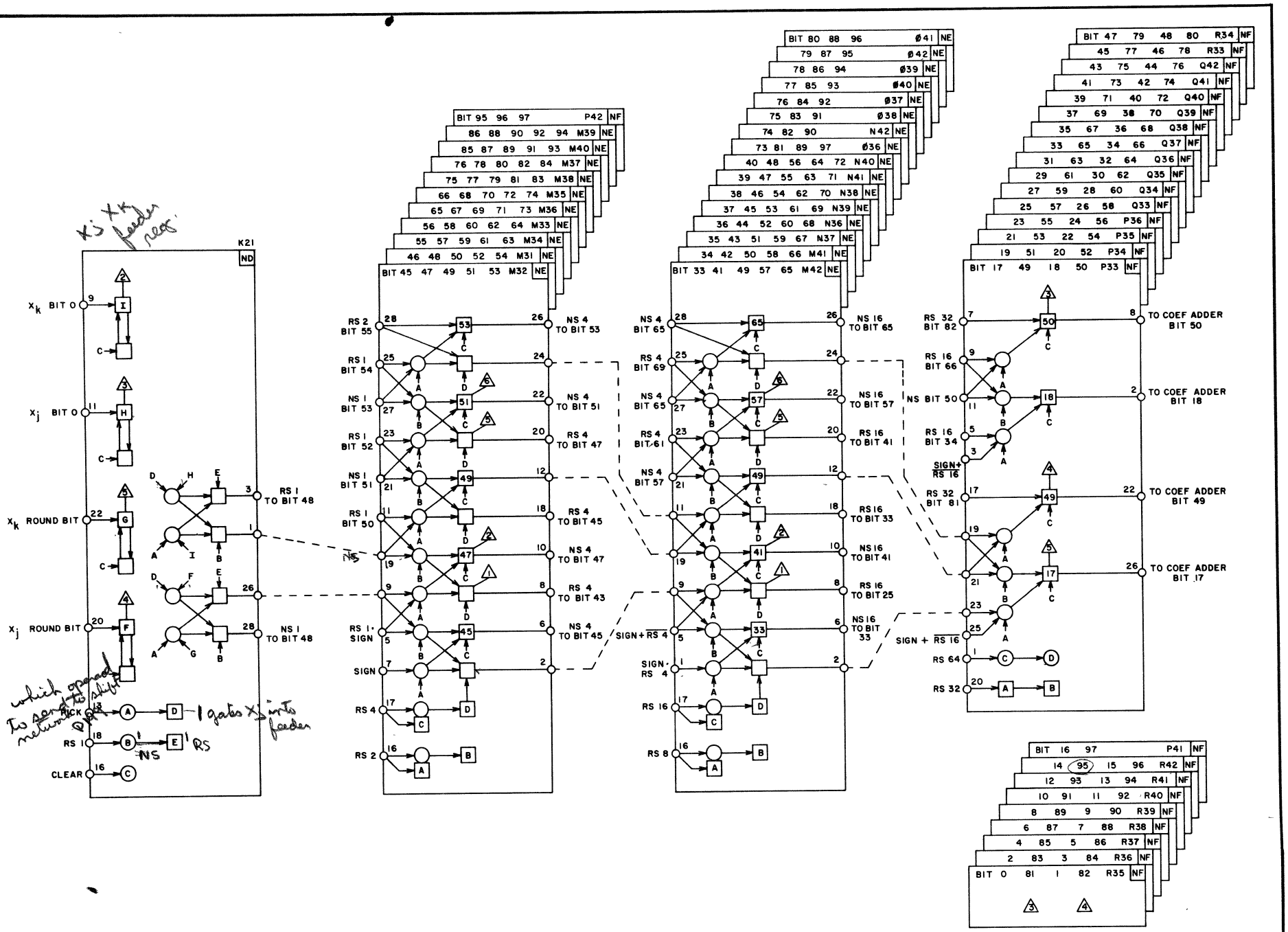
* THE EXPONENT IS NOT REALLY SHIFTED. "SHIFT" REFERS TO CORRECTION OF THE EXPONENT (ADD + 1) WHEN OVERFLOW OCCURS IN THE COEFFICIENT REQUIRING IT TO BE RIGHT SHIFTED.

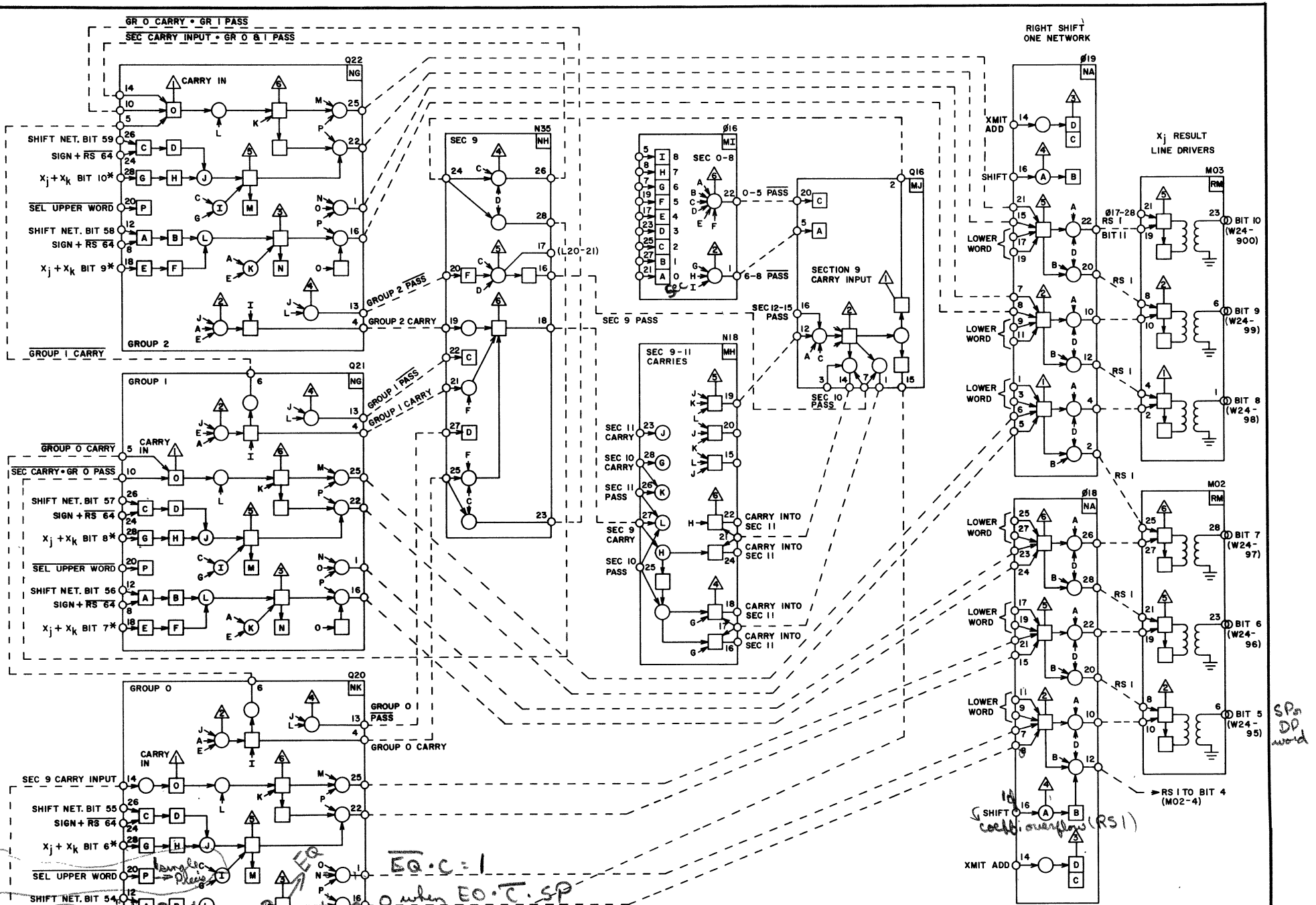
CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR ADD UNIT EXPONENT ADDER, RESULT NET.	PRODUCT 6601/04 SIZE C · DRAWING NO. 60119300 · REV K SHEET 214 · PAGE 201	

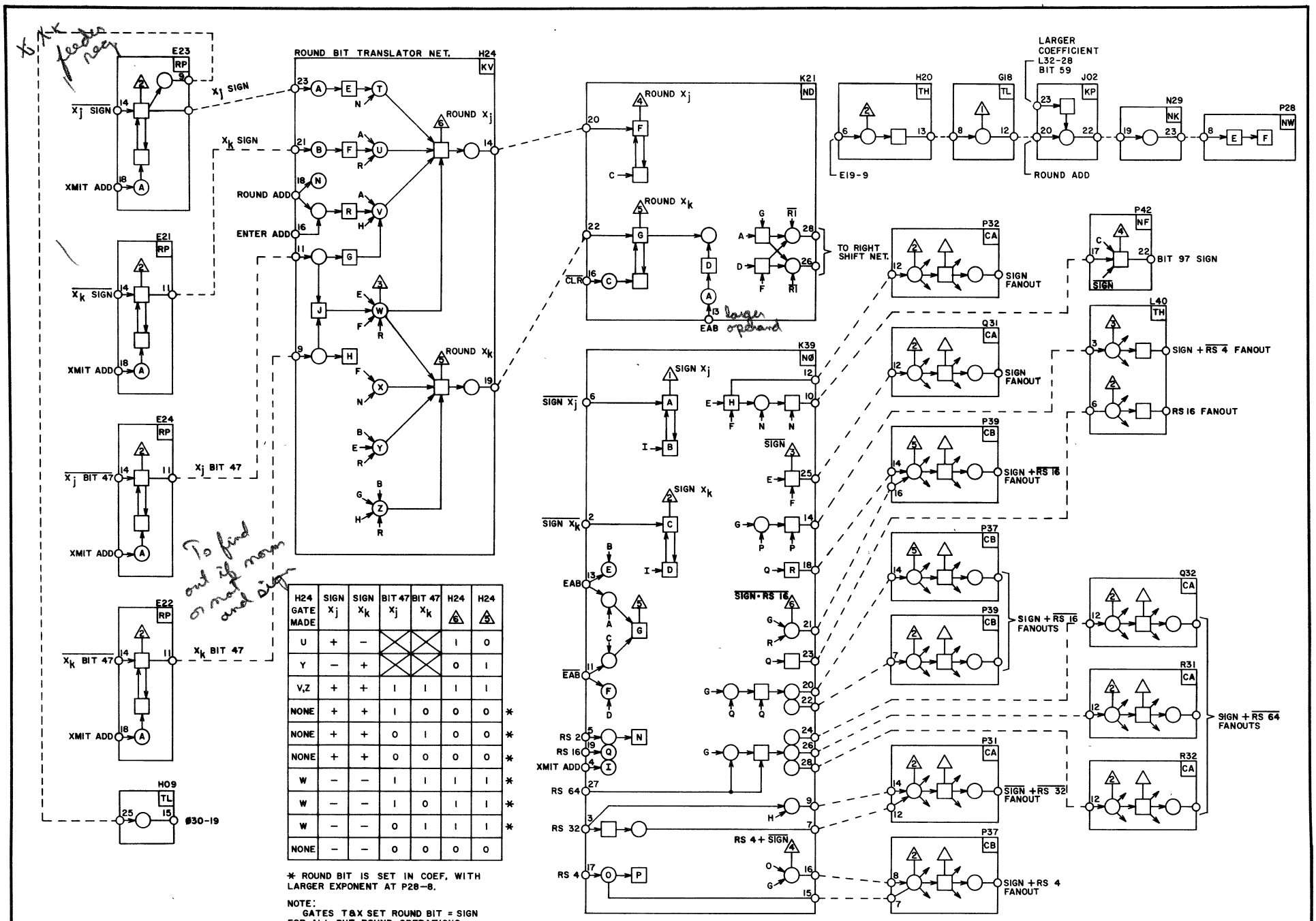


NOTE: REFER TO PAGE 49 IN THE FUNCTIONAL UNIT DRAWINGS FOR CHASSIS 8 INPUT REGISTER BIT LOCATIONS.

CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR ADD UNIT COEFFICIENT DATA PATHS	PRODUCT 6601 SIZE C · DRAWING NO. 60119300 · REV 86 SHEET 222 · PAGE 203	


K21
ND
x_k BIT 0
x_j BIT 0
x_k ROUND BIT
x_j ROUND BIT
RS 1
CLEAR
RS 1 TO BIT 48
NS 1 TO BIT 48
BIT 95 96 97 P42 NF
86 88 90 92 94 M39 NE
85 87 89 91 93 M40 NE
76 78 80 82 84 M37 NE
75 77 79 81 83 M38 NE
66 68 70 72 74 M35 NE
65 67 69 71 73 M36 NE
56 58 60 62 64 M33 NE
55 57 59 61 63 M34 NE
46 48 50 52 54 M31 NE
BIT 45 47 49 51 53 M32 NE
RS 2 BIT 55
RS 1 BIT 54
NS 1 BIT 53
RS 1 BIT 52
NS 1 BIT 51
RS 1 BIT 50
RS 1 SIGN
SIGN
RS 4
RS 2
NS 4 TO BIT 53
NS 4 TO BIT 51
RS 4 TO BIT 47
RS 4 TO BIT 45
NS 4 TO BIT 47
RS 4 TO BIT 43
NS 4 TO BIT 45
BIT 80 88 96 Ø41 NE
79 87 95 Ø42 NE
78 86 94 Ø39 NE
77 85 93 Ø40 NE
76 84 92 Ø37 NE
75 83 91 Ø38 NE
74 82 90 N42 NE
73 81 89 97 Ø36 NE
40 48 56 64 72 N40 NE
39 47 55 63 71 N41 NE
38 46 54 62 70 N38 NE
37 45 53 61 69 N39 NE
36 44 52 60 68 N36 NE
35 43 51 59 67 N37 NE
34 42 50 58 66 M41 NE
BIT 33 41 49 57 65 M42 NE
NS 4 BIT 65
RS 4 BIT 69
NS 4 BIT 65
RS 4 BIT 61
NS 4 BIT 57
SIGN + RS 4
SIGN - RS 4
RS 16
RS 8
NS 16 TO BIT 65
NS 16 TO BIT 57
RS 16 TO BIT 41
RS 16 TO BIT 33
NS 16 TO BIT 41
RS 16 TO BIT 25
NS 16 TO BIT 33
BIT 47 79 48 80 R34 NF
45 77 46 78 R33 NF
43 75 44 76 Q42 NF
41 73 42 74 Q41 NF
39 71 40 72 Q40 NF
37 69 38 70 Q39 NF
35 67 36 68 Q38 NF
33 65 34 66 Q37 NF
31 63 32 64 Q36 NF
29 61 30 62 Q35 NF
27 59 28 60 Q34 NF
25 57 26 58 Q33 NF
23 55 24 56 P36 NF
21 53 22 54 P35 NF
19 51 20 52 P34 NF
BIT 17 49 18 50 P33 NF
RS 32 BIT 82
RS 16 BIT 66
NS BIT 50
RS 16 BIT 34
SIGN+ RS 16
RS 32 BIT 81
SIGN + RS 16
RS 64
RS 32
TO COEF ADDER BIT 50
TO COEF ADDER BIT 18
TO COEF ADDER BIT 49
TO COEF ADDER BIT 17
BIT 16 97 P41 NF
14 95 15 96 R42 NF
12 93 13 94 R41 NF
10 91 11 92 R40 NF
8 89 9 90 R39 NF
6 87 7 88 R38 NF
4 85 5 86 R37 NF
2 83 3 84 R36 NF
BIT 0 81 1 82 R35 NF
CONTROL DATA CORPORATION
COMPUTER DIVISION
TITLE
CENTRAL PROCESSOR
ADD UNIT
RIGHT SHIFT NETWORK
PRODUCT 6601/04
SIZE C
DRAWING NO. 60119300
REV K
SHEET 213
PAGE 205


GR 0 CARRY • GR 1 PASS
SEC CARRY INPUT • GR 0 & 1 PASS
Q22
NG
CARRY IN
SHIFT NET. BIT 59
SIGN + RS 64
$x_j + x_k$ BIT 10*
SEL UPPER WORD
SHIFT NET. BIT 58
SIGN + RS 64
$x_j + x_k$ BIT 9*
GROUP 2
GROUP 2 PASS
GROUP 2 CARRY
GROUP 1 CARRY
Q21
GROUP 1
GROUP 1 PASS
GROUP 1 CARRY
GROUP 0 CARRY
SEC CARRY • GR 0 PASS
SHIFT NET. BIT 57
SIGN + RS 64
$x_j + x_k$ BIT 8*
SEL UPPER WORD
SHIFT NET. BIT 56
SIGN + RS 64
$x_j + x_k$ BIT 7*
Q20
NK
GROUP 0
GROUP 0 PASS
GROUP 0 CARRY
SEC 9 CARRY INPUT
SHIFT NET. BIT 55
SIGN + RS 64
$x_j + x_k$ BIT 6*
SEL UPPER WORD
SHIFT NET. BIT 54
SIGN + RS 64
$x_j + x_k$ BIT 5*
N35
NH
SEC 9
(L20-21)
SEC 9 PASS
Ø16
MI
SEC 0-8
0-5 PASS
6-8 PASS
Q16
MJ
SECTION 9 CARRY INPUT
SEC 12-15 PASS
SEC 10 PASS
N18
MH
SEC 9-11 CARRIES
SEC 11 CARRY
SEC 10 CARRY
SEC 11 PASS
SEC 9 CARRY
SEC 10 PASS
CARRY INTO SEC 11
CARRY INTO SEC 11
CARRY INTO SEC 11
CARRY INTO SEC 11
RIGHT SHIFT ONE NETWORK
Ø19
NA
XMIT ADD
SHIFT
LOWER WORD
Ø17-28 RS 1 BIT 11
RS 1
Ø18
NA
x_j RESULT LINE DRIVERS
M03
RM
BIT 10 (W24-900)
BIT 9 (W24-99)
BIT 8 (W24-98)
M02
RM
BIT 7 (W24-97)
BIT 6 (W24-96)
BIT 5 (W24-95)
RS 1 TO BIT 4 (M02-4)
SP's DP word
EQ·C·1
0 when EQ·C·SP
* THESE BITS ARE OF THE COEFFICIENT WITH THE LARGER EXPONENT.
comp rules
CONTROL DATA CORPORATION
COMPUTER DIVISION
TITLE
CENTRAL PROCESSOR
ADD UNIT
COEFFICIENT ADDER, SEC. 9;
RT. SHIFT ONE NETWORK
PRODUCT
6601/04
SIZE C
DRAWING NO. 60119300
REV ØT
SHEET 221
PAGE 207

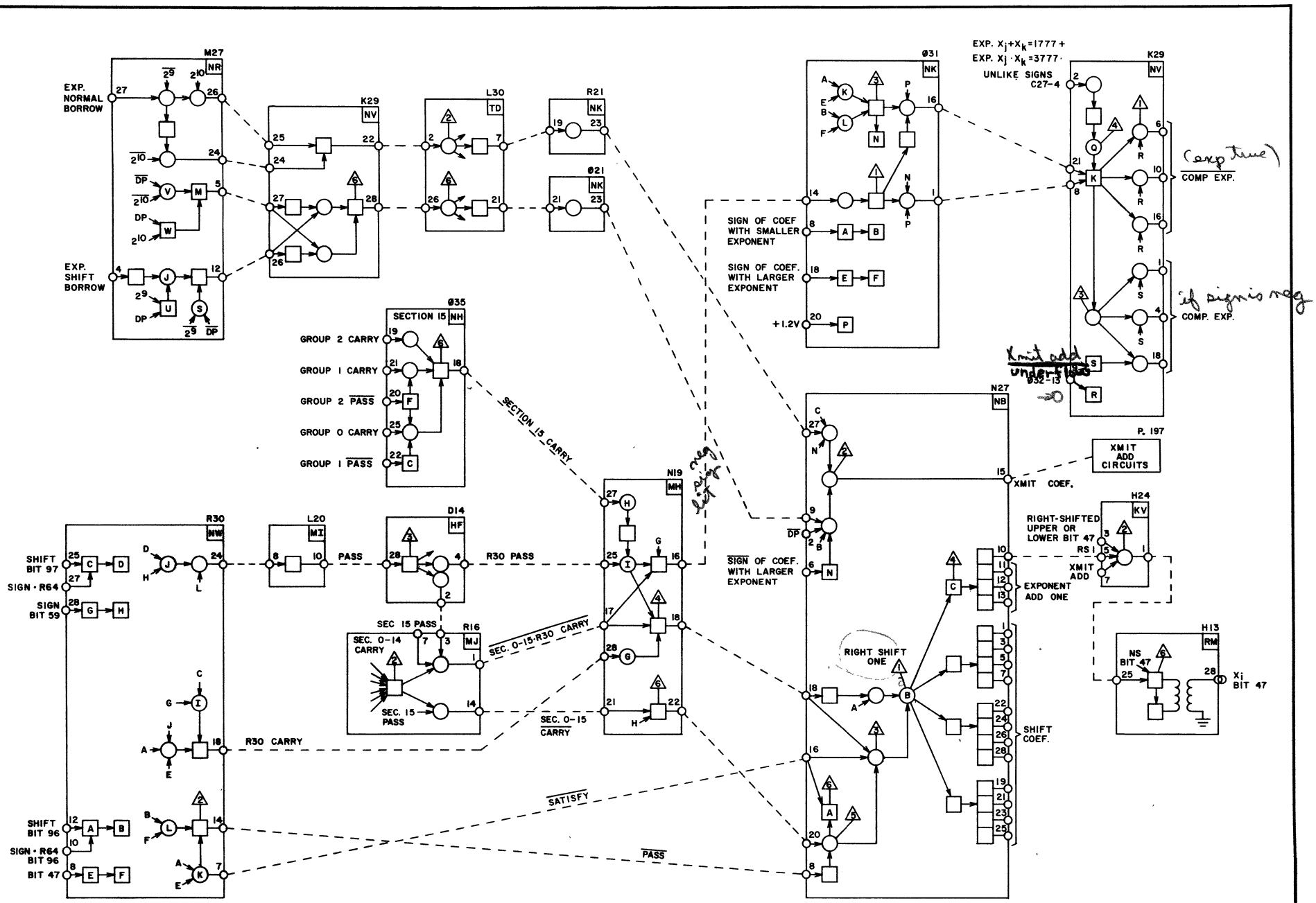


H24 GATE MADE	SIGN X_j	SIGN X_k	BIT 47 X_j	BIT 47 X_k	H24 6	H24 5	
U	+	−			1	0	
Y	−	+			0	1	
V,Z	+	+	1	1	1	1	
NONE	+	+	1	0	0	0	*
NONE	+	+	0	1	0	0	*
NONE	+	+	0	0	0	0	*
W	−	−	1	1	1	1	*
W	−	−	1	0	1	1	*
W	−	−	0	1	1	1	*
NONE	−	−	0	0	0	0	

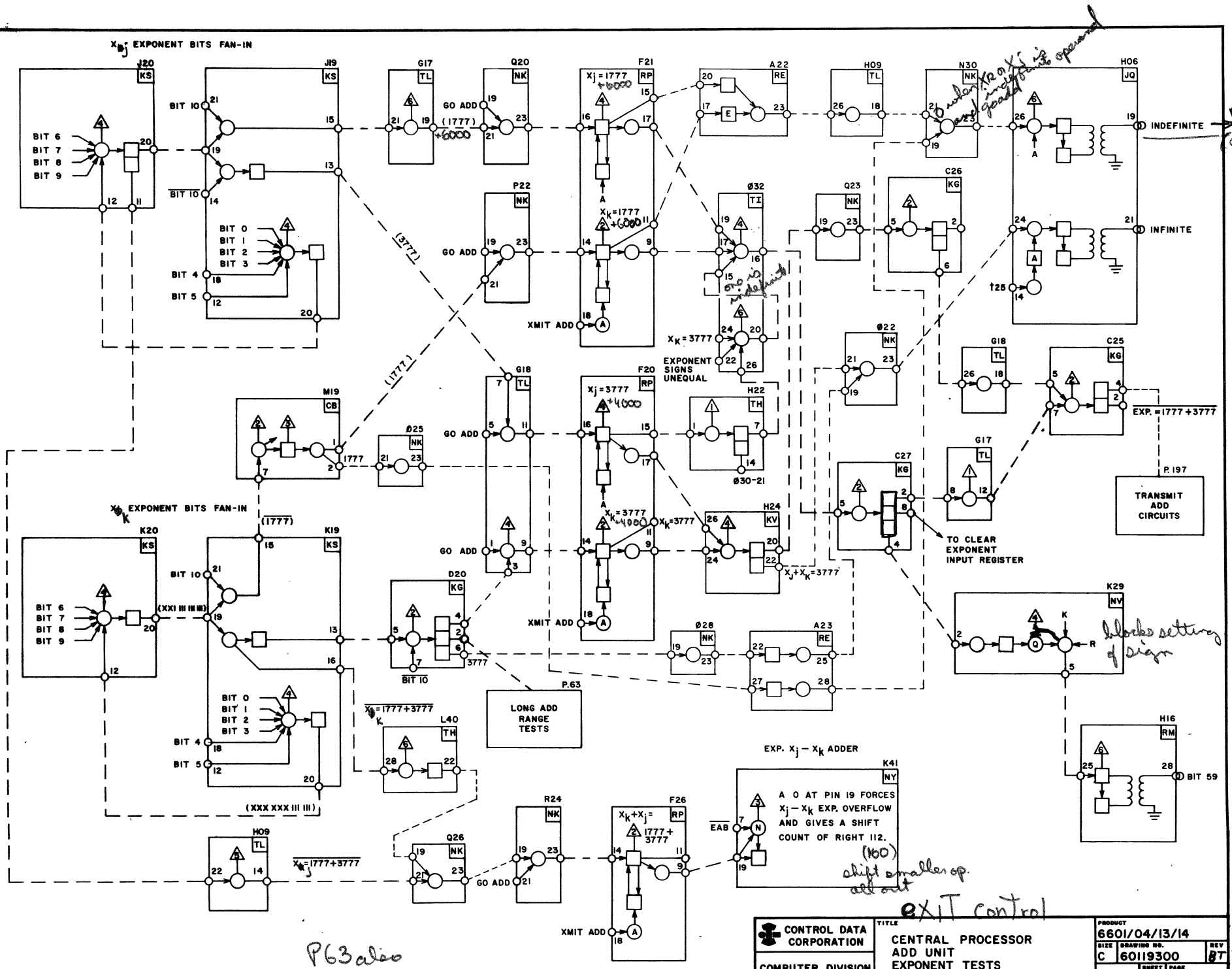
* ROUND BIT IS SET IN COEF. WITH LARGER EXPONENT AT P28–8.

NOTE:
GATES T&X SET ROUND BIT = SIGN FOR ALL BUT ROUND OPERATIONS.

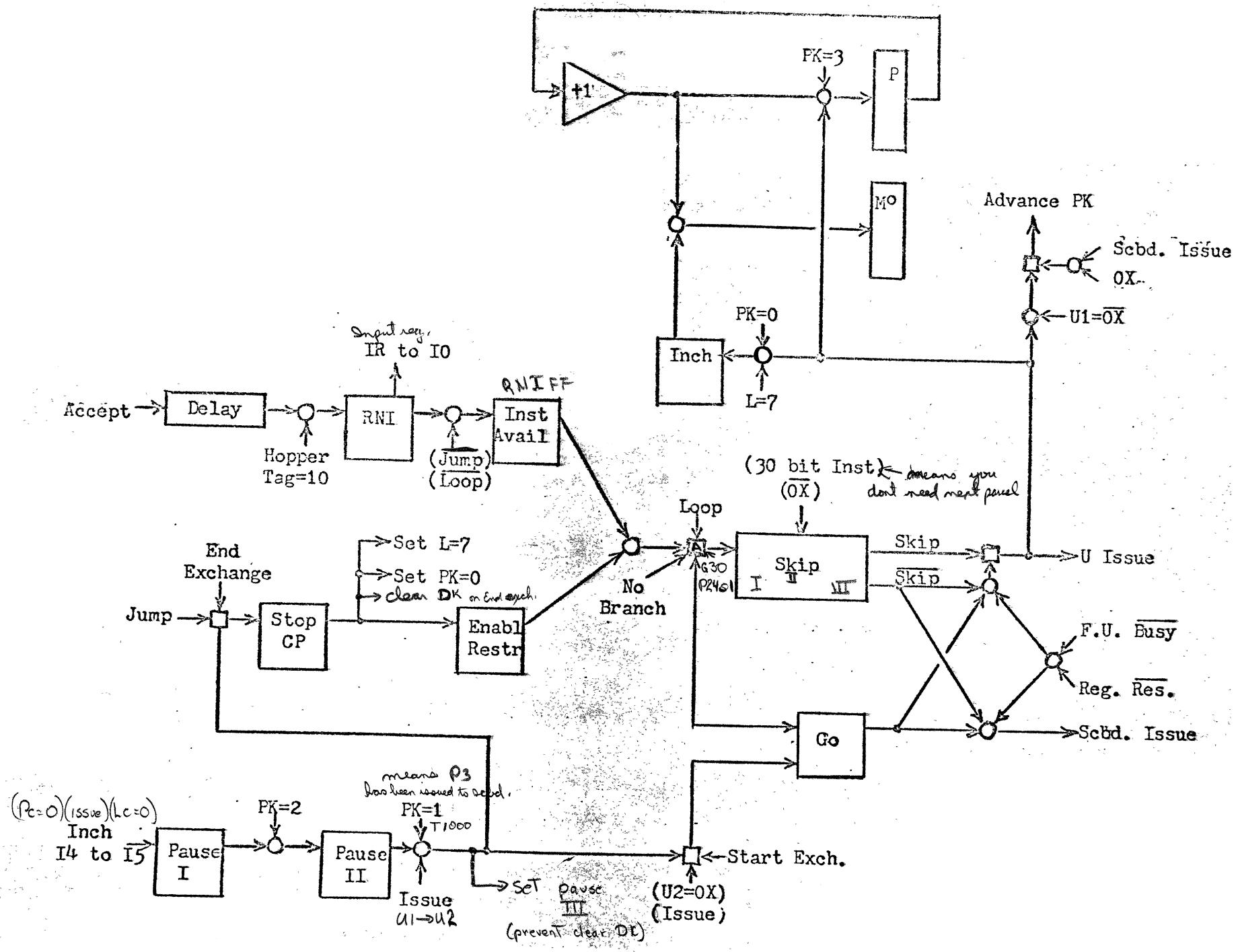
CONTROL DATA CORPORATION	TITLE	CENTRAL PROCESSOR ADD UNIT COEFFICIENT ROUND BIT, SHIFT SIGN EXTENSION	PRODUCT 6601/04/13/14
COMPUTER DIVISION			SIZE C · DRAWING NO. 60119300 · REV BT
			SHEET 223 · PAGE 209



CONTROL DATA CORPORATION COMPUTER DIVISION	TITLE CENTRAL PROCESSOR ADD UNIT COEF. OVERFLOW SENSE NET., SIGN TEST, COMP. EXP.	PRODUCT 6601/04/13/14	
		SIZE C	DRAWING NO. 60119300
		REV 8T	
		SHEET 224	PAGE 211

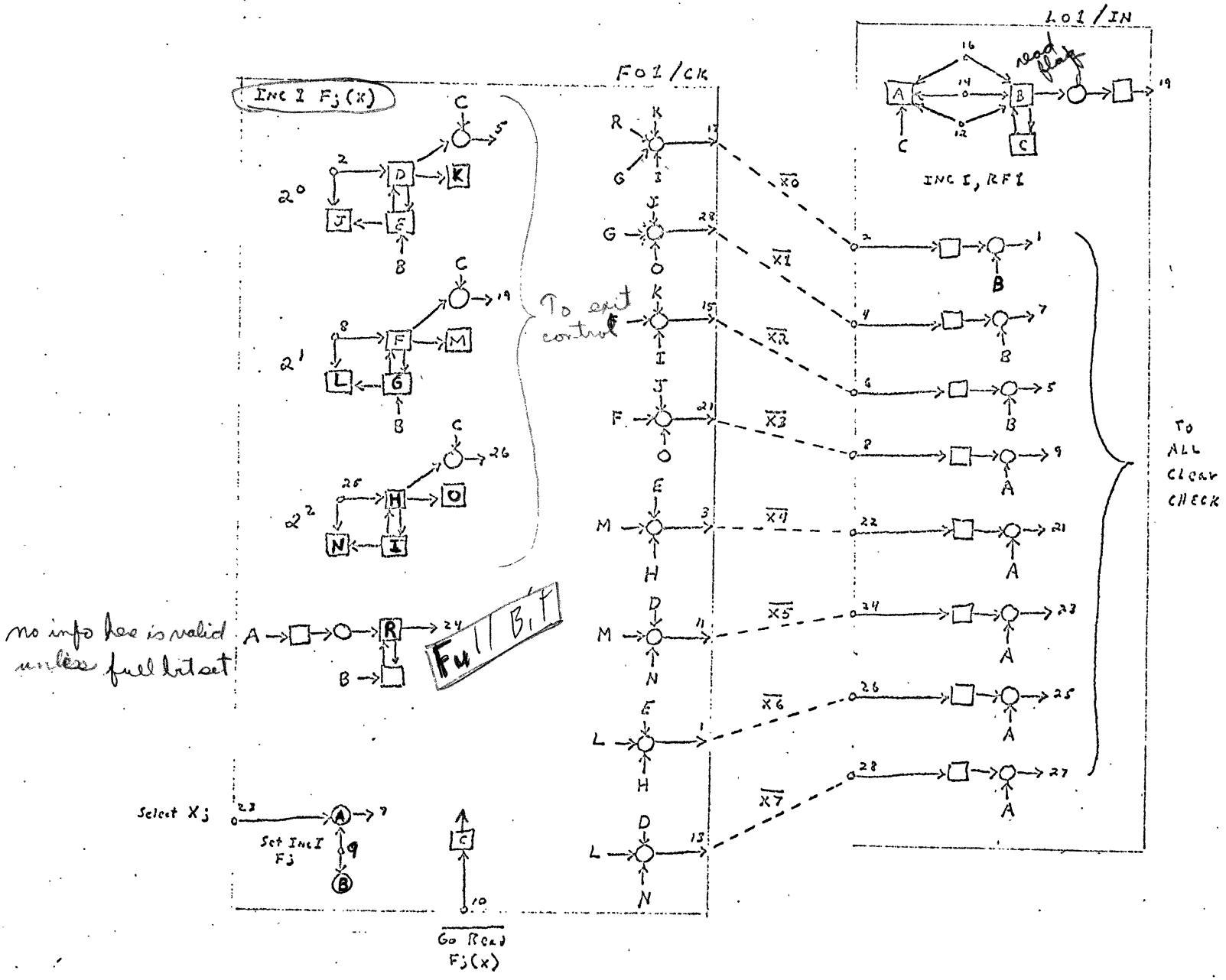


CONTROL DATA CORPORATION	TITLE: CENTRAL PROCESSOR ADD UNIT EXPONENT TESTS	PRODUCT: 6601/04/13/14	
COMPUTER DIVISION		SIZE: C	DRAWING NO.: 60119300
		REV: BT	
		SHEET: 216	PAGE: 213

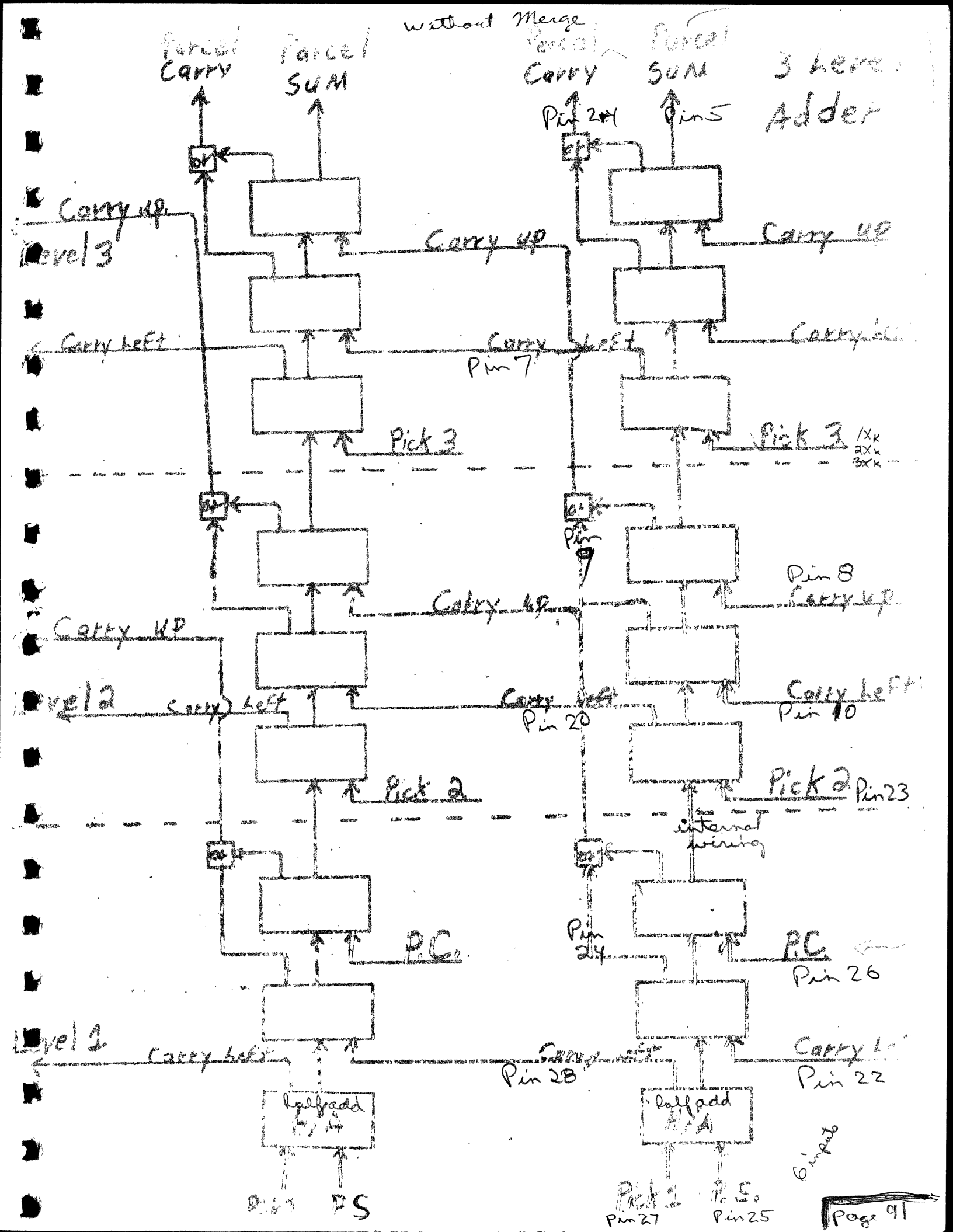


GO CONTROL BLOCK DIAGRAM

Increment I Read Flag I (Fj x)



PRINTS 50


without Merge
Parcel Carry
Parcel SUM
Parcel Carry
Parcel SUM
3 here: Adder
Pin 24
Pin 5
Carry up
Carry up
Carry up
Level 3
Carry left
Carry left
Carry left
Pin 7
Pick 2
Pick 3
1X k
2X k
3X k
Pin 9
Pin 8
Carry up
Carry up
Carry up
Level 2
Carry left
Carry left
Carry left
Pin 20
Pin 10
Pick 2
Pick 2
Pin 23
internal wiring
P.C.
Pin 24
P.C.
Pin 26
Level 1
Carry left
Carry left
Carry left
Pin 28
Pin 22
Halfadd
Halfadd
A
6 inputs
Pick 1
Pin 27
P.C.
Pin 25
Page 91

COMMENT SHEET

MANUAL TITLE CONTROL DATA 6601/04 CENTRAL COMPUTER

CE Diagrams and Circuit Description Manual Vol. 1

PUBLICATION NO. 60119300 REVISION _____

FROM: NAME: _____

BUSINESS ADDRESS: _____

COMMENTS:

This form is not intended to be used as an order blank. Your evaluation of this manual will be welcomed by Control Data Corporation. Any errors, suggested additions or deletions, or general comments may be made below. Please include page number references and fill in publication revision level as shown by the last entry on the Record of Revision page at the front of the manual. Customer engineers are urged to use the TAR.

CUT ALONG LINE

FORM CA231 REV. 1-67 PRINTED IN U.S.A.

NO POSTAGE STAMP NECESSARY IF MAILED IN U. S. A.

FOLD ON DOTTED LINES AND STAPLE

STAPLE

STAPLE

FOLD

FOLD

FIRST CLASS
PERMIT NO. 8241

MINNEAPOLIS, MINN.

BUSINESS REPLY MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN U.S.A.

POSTAGE WILL BE PAID BY

CONTROL DATA CORPORATION

8100 34TH AVENUE SOUTH

MINNEAPOLIS, MINNESOTA 55440

ATTN: TECHNICAL PUBLICATIONS DEPT.
PLANT TWO

CUT ALONG LINE

FOLD

FOLD