United States Patent [19]

Wilber et al.

[11] **3,828,321**

[45] **Aug. 6, 1974**

[54] **SYSTEM FOR RECONFIGURING CENTRAL PROCESSOR AND INSTRUCTION STORAGE COMBINATIONS**

[75] Inventors: **John A. Wilber,** Elk Grove Village; **Verner K. Rice,** Wheaton; **Rolfe E. Buhrke,** La Grange Park, all of Ill.

[73] Assignee: **GTE Automatic Electric Laboratories Incorporated,** Northlake, Ill.

[22] Filed: **Mar. 15, 1973**

[21] Appl. No.: **341,428**

[52] **U.S. Cl.** **340/172.5**

[51] **Int. Cl.** **G06f 11/00,** G06f 13/00

[58] **Field of Search** 340/172.5

[56] **References Cited**

UNITED STATES PATENTS

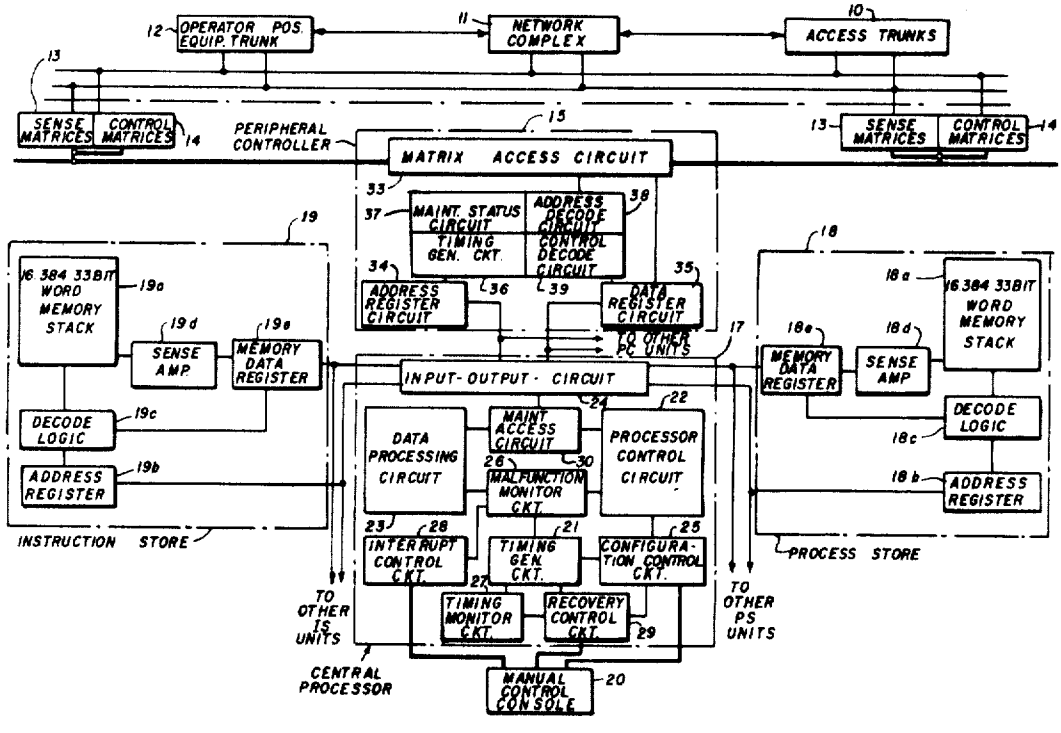
3,386,082	5/1968	Stafford et al.	340/172.5
3,409,877	11/1968	Alterman et al.	340/172.5
3,503,048	3/1970	Avsan	340/172.5
3,562,716	2/1971	Fontaine et al.	340/172.5
3,623,014	11/1971	Doelz et al.	340/172.5
3,641,505	2/1972	Artz	340/172.5
3,654,603	4/1972	Gunning et al.	340/172.5 X

Primary Examiner—Harvey E. Springborn

[57] **ABSTRACT**

A system is disclosed including duplicate copies of central processors and storage means for switching active copies of central processor and primary copies of instruction through all combinations to find a working combination. The reconfiguration may be effected either through a fixed wired recovery control circuit which changes state in predetermined sequence or, alternatively, under program control.

11 Claims, 51 Drawing Figures



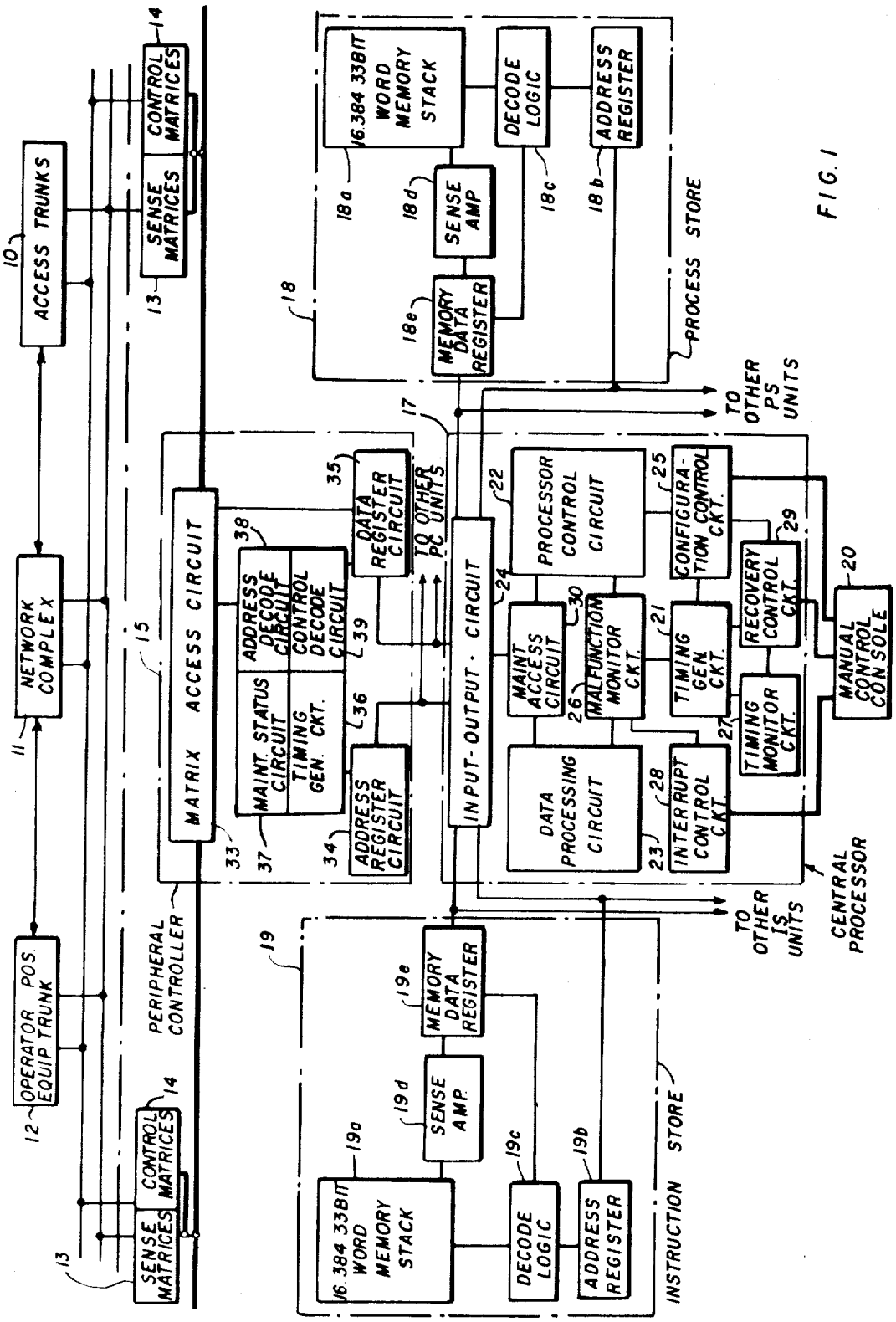

ACCESS TRUNKS
10
NETWORK COMPLEX
11
OPERATOR POS. EQUIP TRUNK
12
SENSE MATRICES
13
CONTROL MATRICES
14
PERIPHERAL CONTROLLER
15
MATRIX ACCESS CIRCUIT
33
MAINT. STATUS CIRCUIT
37
TIMING GEN. CKT.
ADDRESS DECODE CIRCUIT
CONTROL DECODE CIRCUIT
38
ADDRESS REGISTER CIRCUIT
34
DATA REGISTER CIRCUIT
35
TO OTHER PS UNITS
36
39
INPUT-OUTPUT CIRCUIT
17
PROCESSOR CONTROL CIRCUIT
22
CONFIGURATION CONTROL CKT.
25
MAINT ACCESS CIRCUIT
24
MALFUNCTION MONITOR CKT.
30
26
TIMING GEN. CKT.
21
RECOVERY CONTROL CKT.
29
MANUAL CONTROL CONSOLE
20
DATA PROCESSING CIRCUIT
23
INTERRUPT CONTROL CKT.
28
TIMING MONITOR CKT.
27
CENTRAL PROCESSOR
TO OTHER IS UNITS
PROCESS STORE
18
16,384 33BIT WORD MEMORY STACK
18a
DECODE LOGIC
18c
ADDRESS REGISTER
18b
SENSE AMP
18d
MEMORY DATA REGISTER
18e
INSTRUCTION STORE
19
16,384 33BIT WORD MEMORY STACK
19a
SENSE AMP
19d
MEMORY DATA REGISTER
19e
DECODE LOGIC
19c
ADDRESS REGISTER
19b

FIG. 1

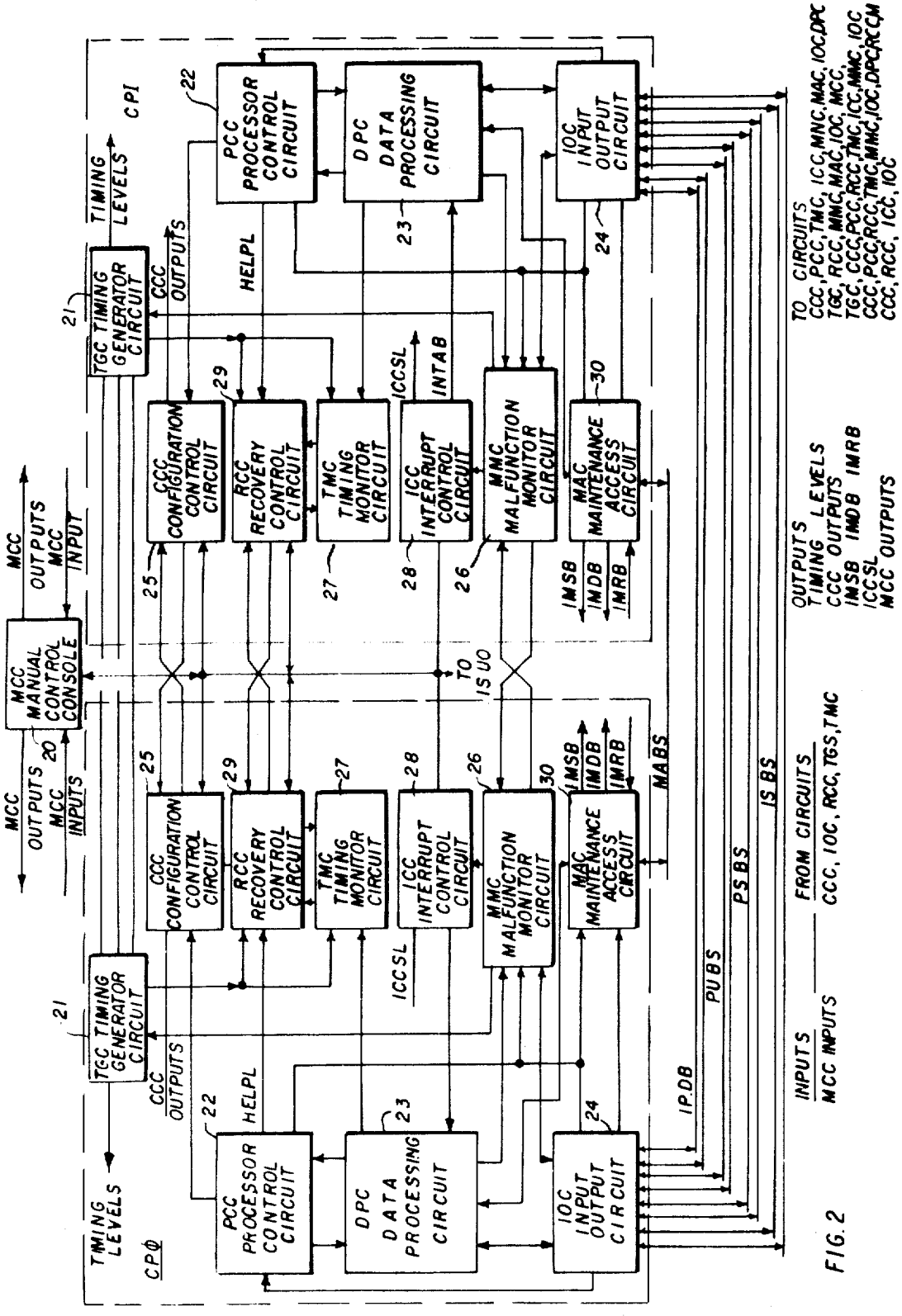

CPI
CPØ
TIMING LEVELS
TGC TIMING GENERATOR CIRCUIT
CCC OUTPUTS
HELPL
PCC PROCESSOR CONTROL CIRCUIT
DPC DATA PROCESSING CIRCUIT
IOC INPUT OUTPUT CIRCUIT
CCC CONFIGURATION CONTROL CIRCUIT
RCC RECOVERY CONTROL CIRCUIT
TMC TIMING MONITOR CIRCUIT
ICC INTERRUPT CONTROL CIRCUIT
MMC MALFUNCTION MONITOR CIRCUIT
MAC MAINTENANCE ACCESS CIRCUIT
MCC MANUAL CONTROL CONSOLE
MCC OUTPUTS
MCC INPUTS
ICCSL
INTAB
IMSB
IMDB
IMRB
TO IS UO
IPDB
PUBS
PSBS
ISBS
MABS
INPUTS
MCC INPUTS
FROM CIRCUITS
CCC, IOC, RCC, TGS, TMC
OUTPUTS
TIMING LEVELS
CCC OUTPUTS
IMSB IMDB IMRB
ICCSL
MCC OUTPUTS
TO CIRCUITS
CCC, PCC, TMC, ICC, MNC, MAC, IOC, DPC
TGC, RCC, MMC, MAC, IOC, MCC,
TGC, CCC, PCC, RCC, TMC, ICC, MMC, IOC
CCC, PCC, RCC, TMC, MMC, IOC, DPC, RCC, M
CCC, RCC, TCC, IOC

FIG. 2

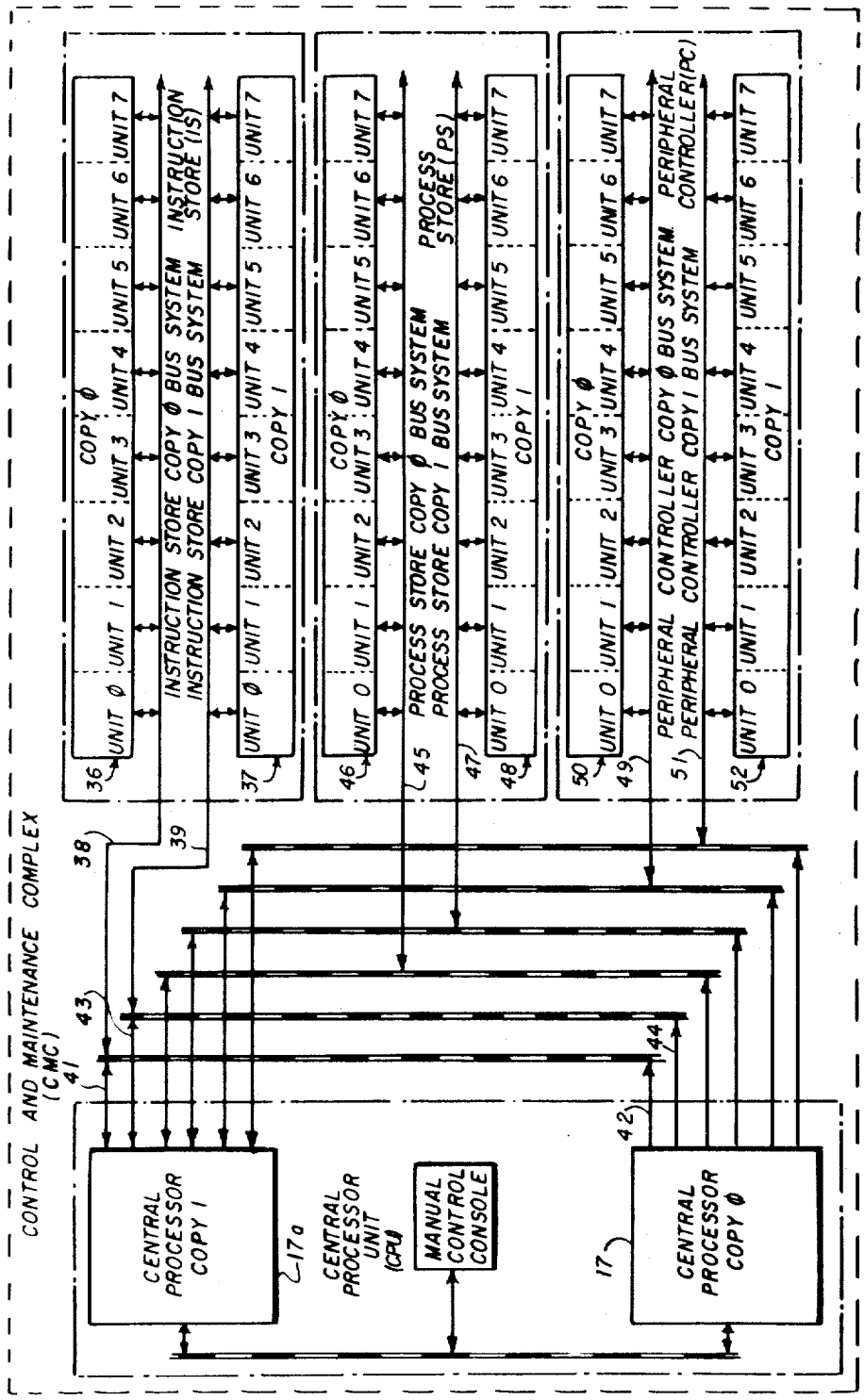


FIG. 2A

CENTRAL PROCESSOR FUNCTIONAL INTERFACE BLOCK DIAGRAM

FIG. 3

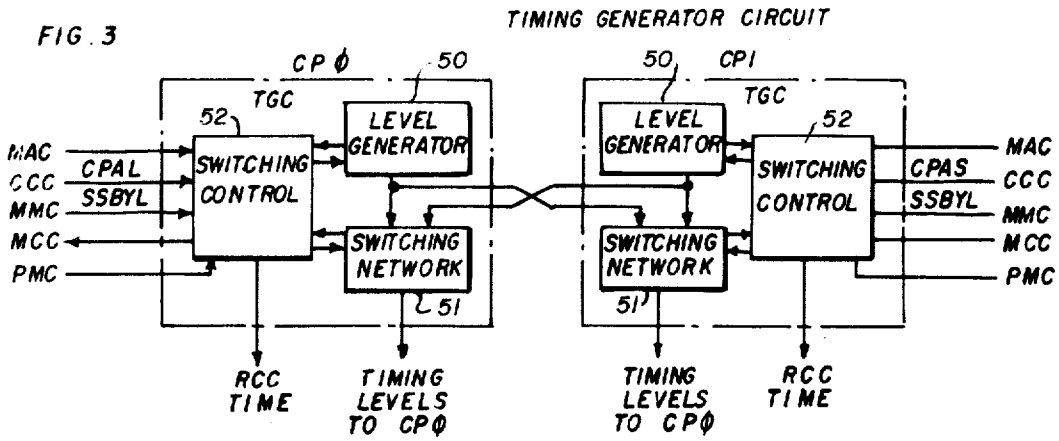

TIMING GENERATOR CIRCUIT
CPØ
50
TGC
52
LEVEL GENERATOR
SWITCHING CONTROL
SWITCHING NETWORK
51
MAC
CCC
MMC
MCC
PMC
CPAL
SSBYL
RCC TIME
TIMING LEVELS TO CPØ
50
CPI
TGC
LEVEL GENERATOR
52
SWITCHING CONTROL
SWITCHING NETWORK
51
CPAS
SSBYL
MAC
CCC
MMC
MCC
PMC
TIMING LEVELS TO CPØ
RCC TIME

FIG. 4

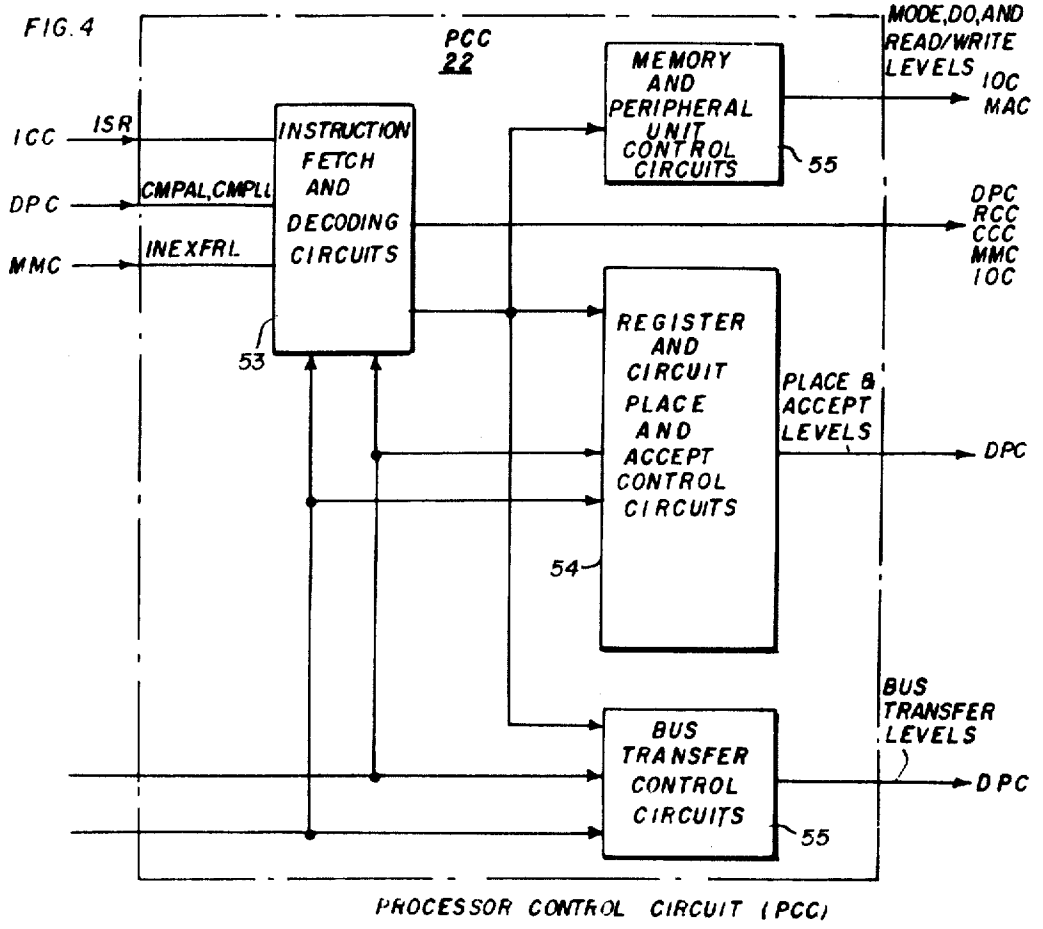

PCC
22
ICC
ISR
DPC
CMPAL,CMPLL
MMC
INEXFRL
INSTRUCTION FETCH AND DECODING CIRCUITS
53
MEMORY AND PERIPHERAL UNIT CONTROL CIRCUITS
55
MODE,DO,AND READ/WRITE LEVELS
IOC
MAC
DPC
RCC
CCC
MMC
IOC
REGISTER AND CIRCUIT PLACE AND ACCEPT CONTROL CIRCUITS
54
PLACE & ACCEPT LEVELS
DPC
BUS TRANSFER CONTROL CIRCUITS
55
BUS TRANSFER LEVELS
DPC
PROCESSOR CONTROL CIRCUIT (PCC)

FIG. 5

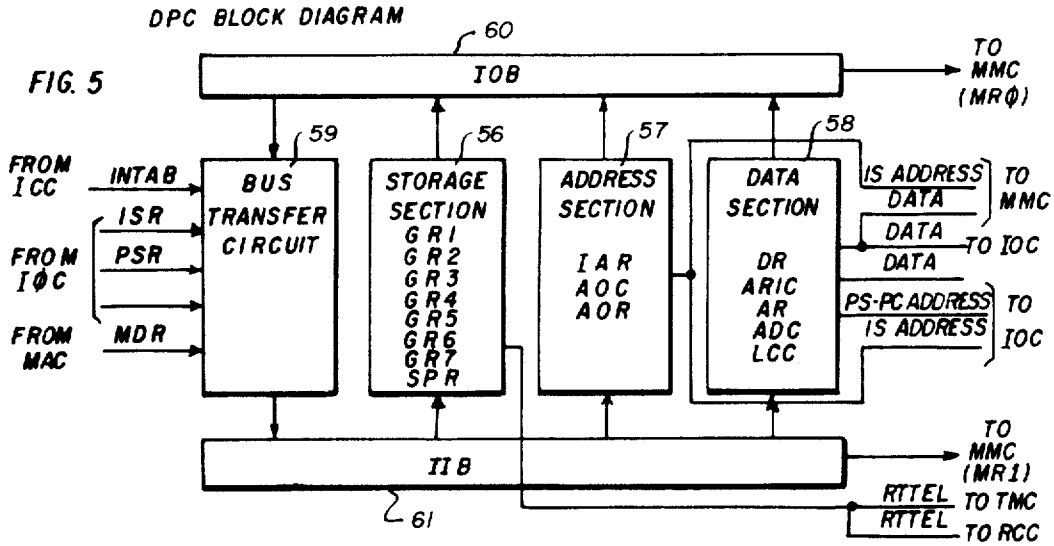
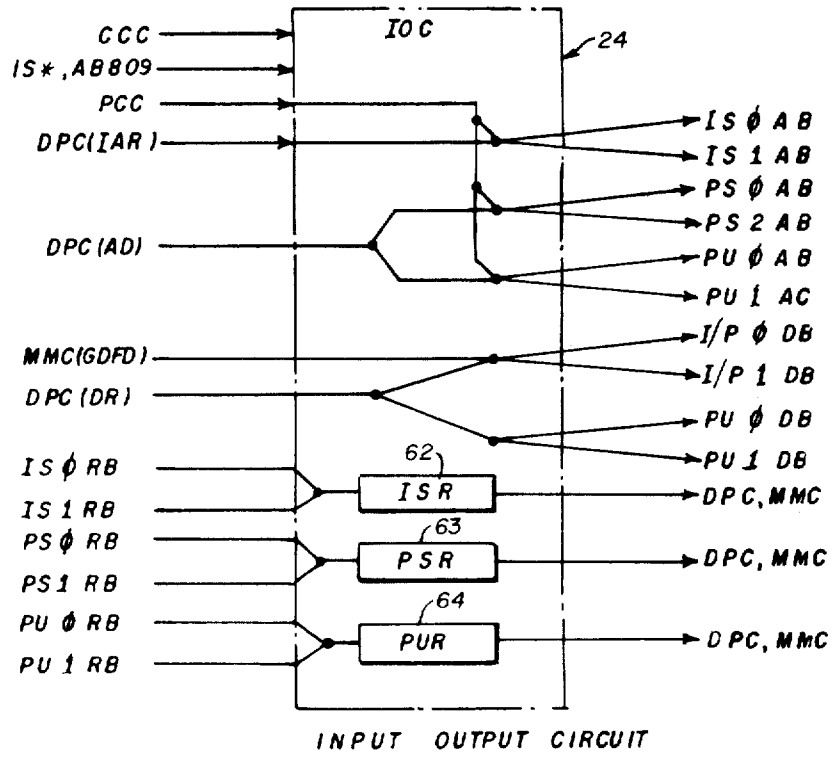

DPC BLOCK DIAGRAM
60
IOB
TO MMC (MRØ)
59
56
57
58
FROM ICC
INTAB
ISR
FROM IØC
PSR
FROM MAC
MDR
BUS TRANSFER CIRCUIT
STORAGE SECTION
GR1
GR2
GR3
GR4
GR5
GR6
GR7
SPR
ADDRESS SECTION
IAR
AOC
AOR
DATA SECTION
DR
ARIC
AR
ADC
LCC
IS ADDRESS
DATA
TO MMC
DATA
TO IOC
DATA
PS-PC ADDRESS
IS ADDRESS
TO IOC
IIB
TO MMC (MR1)
RTTEL
TO TMC
RTTEL
TO RCC
61

FIG. 6


CCC
IS*, AB809
PCC
DPC(IAR)
DPC(AD)
MMC(GDFD)
DPC(DR)
IS Ø RB
IS 1 RB
PS Ø RB
PS 1 RB
PU Ø RB
PU 1 RB
IOC
24
62
ISR
63
PSR
64
PUR
IS Ø AB
IS 1 AB
PS Ø AB
PS 2 AB
PU Ø AB
PU 1 AC
I/P Ø DB
I/P 1 DB
PU Ø DB
PU 1 DB
DPC, MMC
DPC, MMC
DPC, MMC
INPUT OUTPUT CIRCUIT

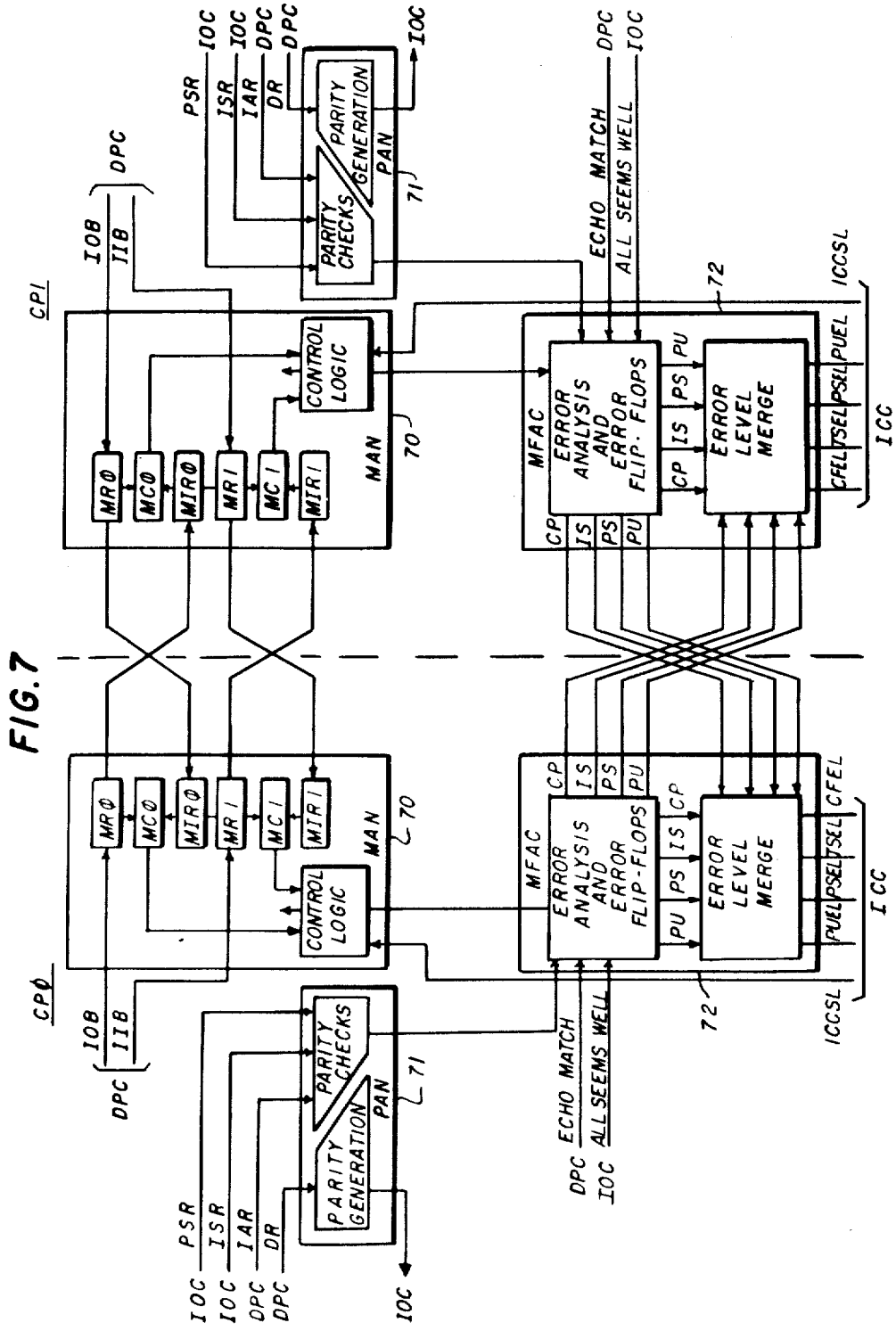

CP1
CP0
IOB
IIB
DPC
PSR
ISR
IAR
DR
IOC
PARITY CHECKS
PARITY GENERATION
PAN
71
MAN
70
MR0
MC0
MIR0
MR1
MC1
MIR1
CONTROL LOGIC
MFAC
ERROR ANALYSIS AND ERROR FLIP-FLOPS
CP
IS
PS
PU
ERROR LEVEL MERGE
72
ECHO MATCH
ALL SEEMS WELL
ICCSL
PUEL
PSEL
TSEL
CFEL
ICC

FIG. 7

FIG.8

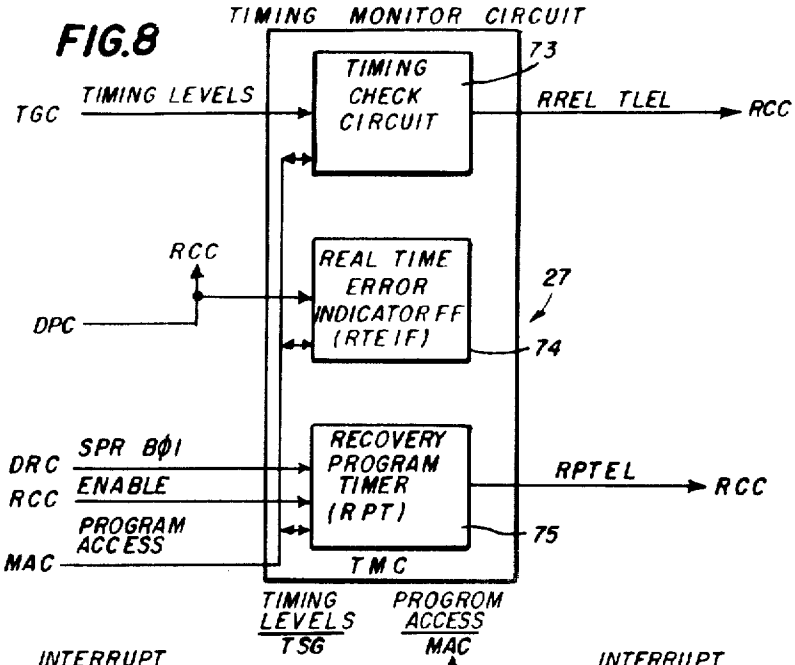


FIG.9

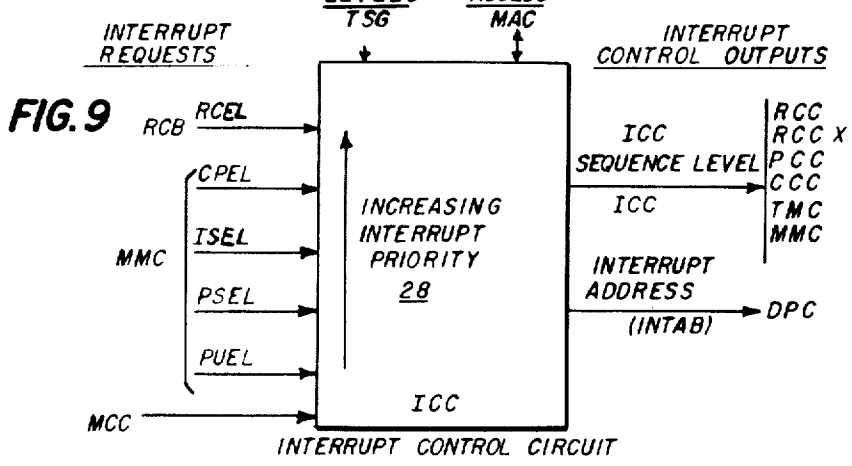


FIG.10

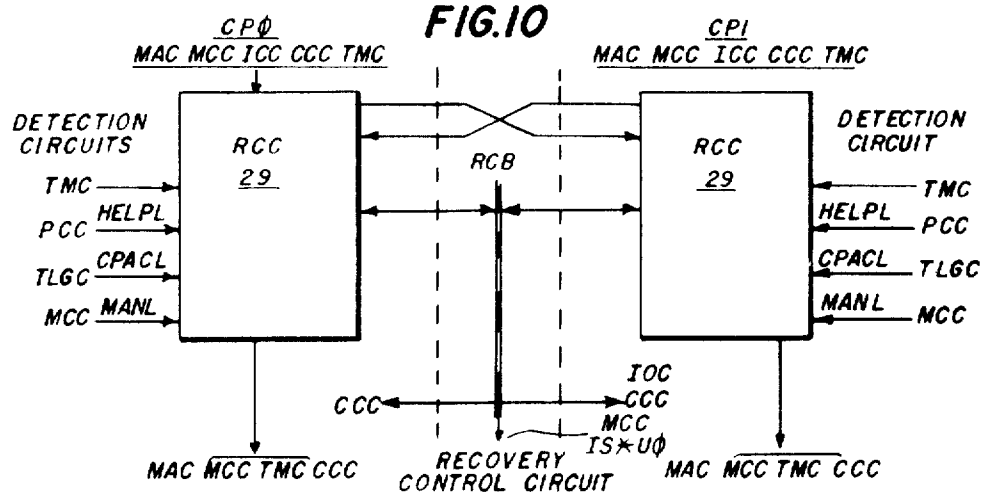


FIG. 11

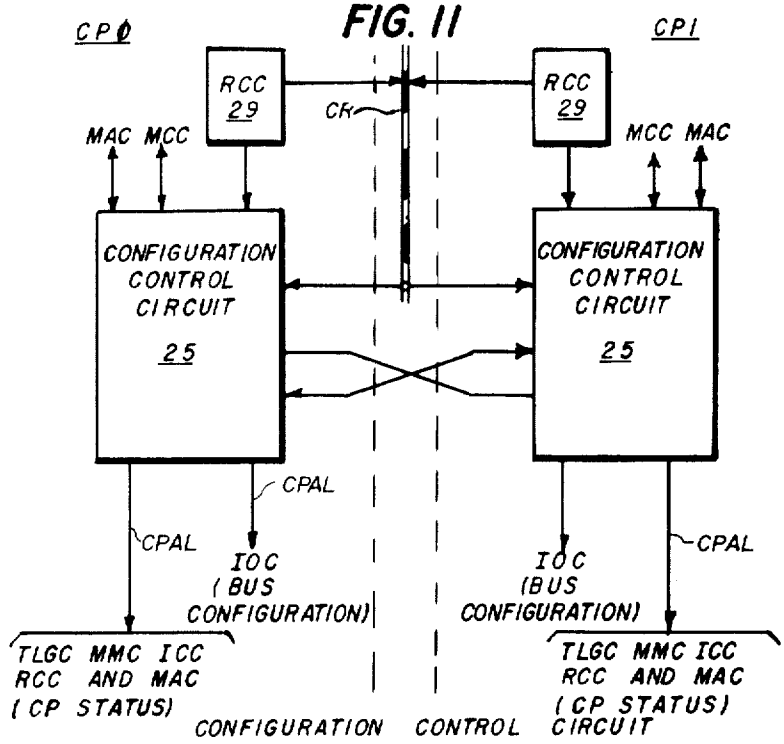

CP∅
CPI
RCC 29
CR
RCC 29
MAC MCC
MCC MAC
CONFIGURATION CONTROL CIRCUIT 25
CONFIGURATION CONTROL CIRCUIT 25
CPAL
CPAL
CPAL
IOC (BUS CONFIGURATION)
IOC (BUS CONFIGURATION)
TLGC MMC ICC RCC AND MAC (CP STATUS)
TLGC MMC ICC RCC AND MAC (CP STATUS)
CONFIGURATION CONTROL CIRCUIT

FIG. 12

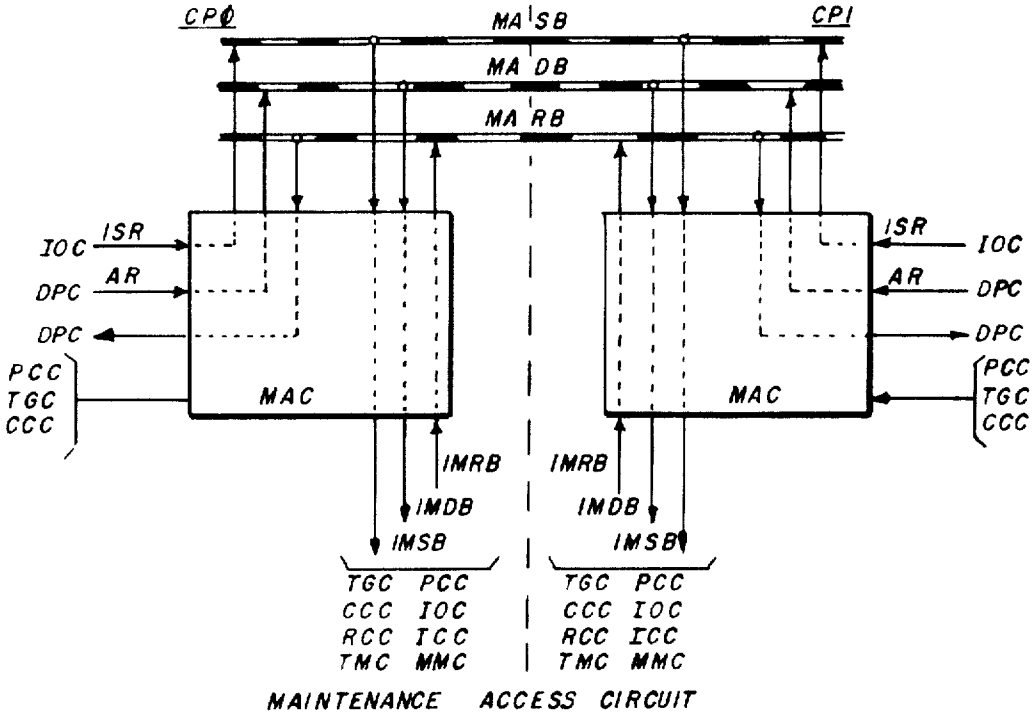

CP∅
MA SB
CPI
MA DB
MA RB
IOC
ISR
DPC
AR
DPC
DPC
PCC
TGC
CCC
MAC
IMRB
IMDB
IMSB
TGC PCC
CCC IOC
RCC ICC
TMC MMC
ISR
IOC
AR
DPC
DPC
PCC
TGC
CCC
MAC
IMRB
IMDB
IMSB
TGC PCC
CCC IOC
RCC ICC
TMC MMC
MAINTENANCE ACCESS CIRCUIT

FIG. 13

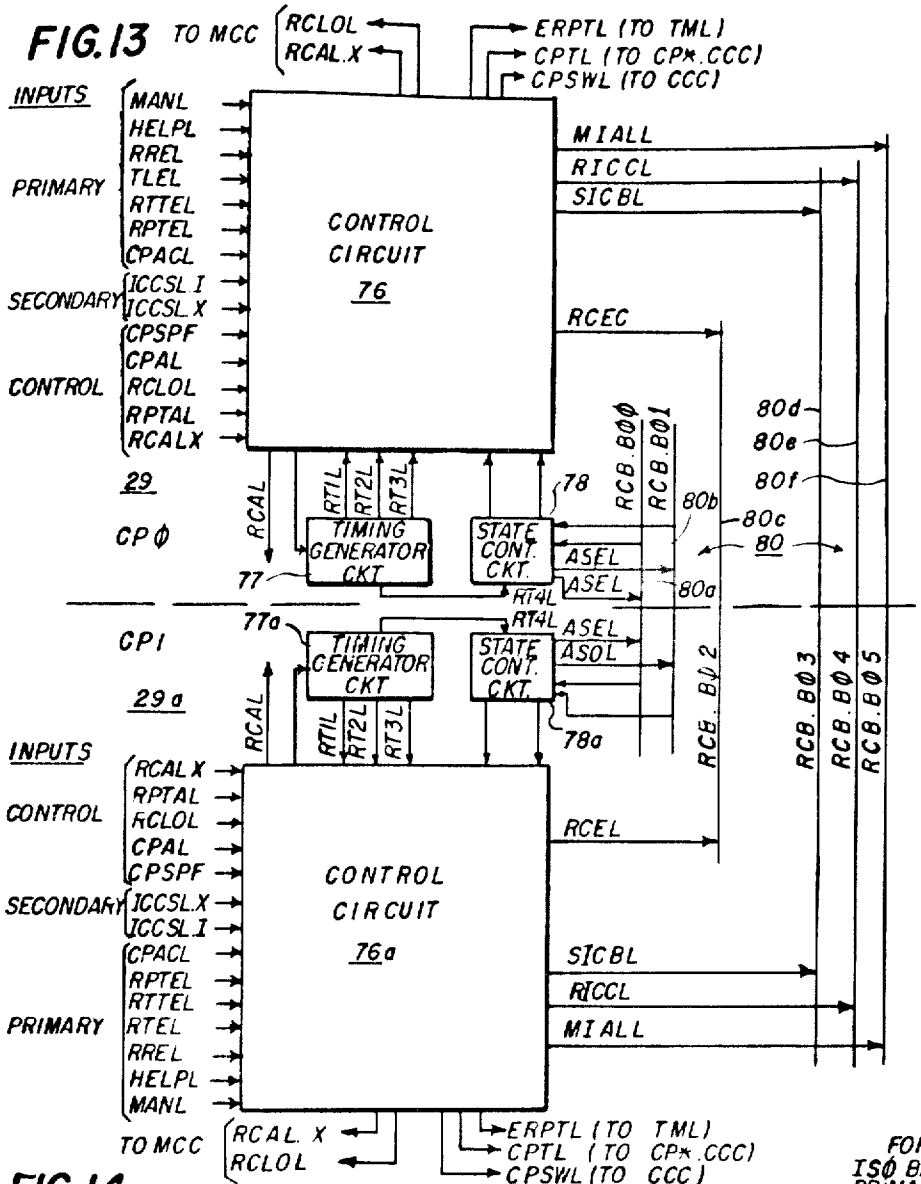
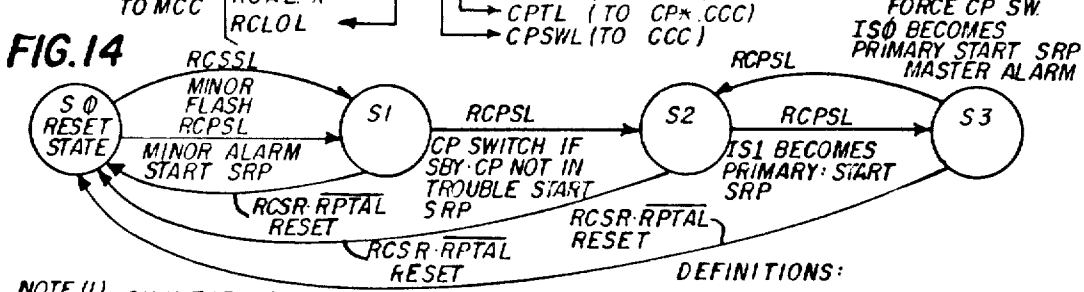


FIG. 14



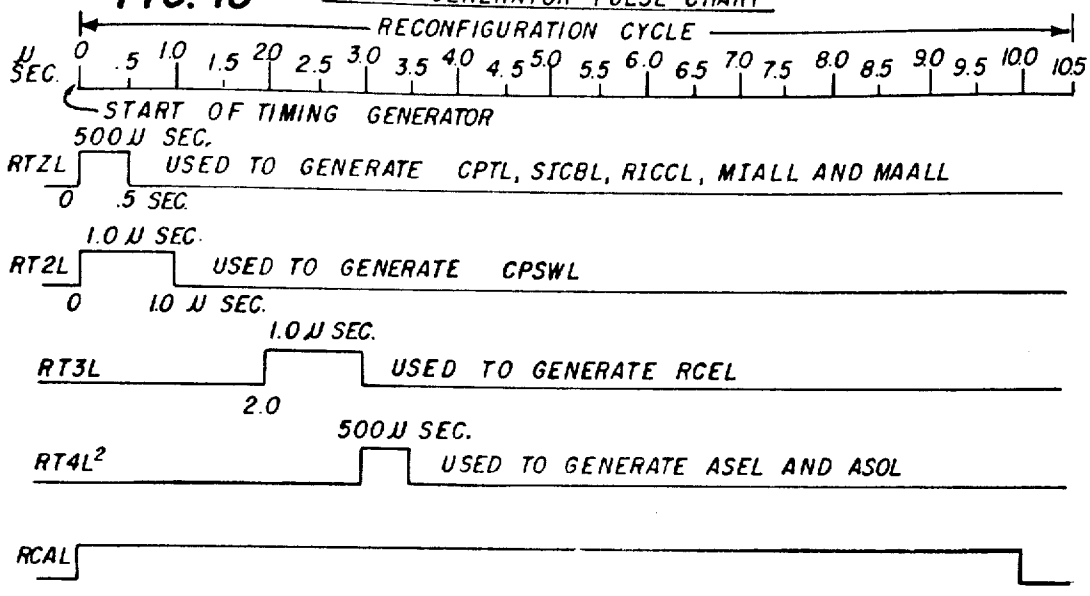
NOTE (1) ONLY THE RCC IN THE ACTIVE CP CAN CAUSE A STATE TRANSITION

(2) BOTH RCC WILL BE IN SAME STATE

DEFINITIONS:

RCSR - RCC STATE RESET
RPTAL - RECOVERY PROGRAM TIMER ACTIVE LEVEL.
RCPSL - RCC PRIMARY START LEVEL
RCSSL - RCC SECONDARY START LEVEL

FIG. 15 TIMING GENERATOR PULSE CHART



NOTE(1) RCC LOCKED OUT TO TRIGGERS FROM START OF CYCLE UNTIL END OF CYCLE

(2) RT4L CAN OCCUR ANY TIME AFTER RT3L AND RT5L; NEW RCC STATE STARTS AT END OF RT4L

FIG. 16 STATE TRANSITION TABLE

STATE	START LEVEL RCSSL	RCPSL	NEXT STATE	ERPTL	RCAL	MIALL	MAALL	RCEL	CPSWL	CPTL	STCBL	RICCL	ASEL	ASOL	FUNCTION
SØ	X		S1	X	X	X							X		PRIME RCC FOR CP SWITCH IN CASE RECOVERY PROGRAM INDICATES ACTIVE CP MALFUNCTION
SØ		X	S1	X	X	X		X					X		START SRP
S1		X	S2	X	X			X	X					X	SWITCH CP'S IF STANDBY IS NOT IN TROUBLE; START SRP
S2		X	S3	X	X						X	X	X		IS1 BECOMES PRIMARY INSTRUCTION STORE START SRP
S3		X	S2	X	X		X	X	X	X		X		X	FORCE CP SWITCH; ISC BECOMES PRIMARY INSTRUCTION STORE START SRP

FIG. 19

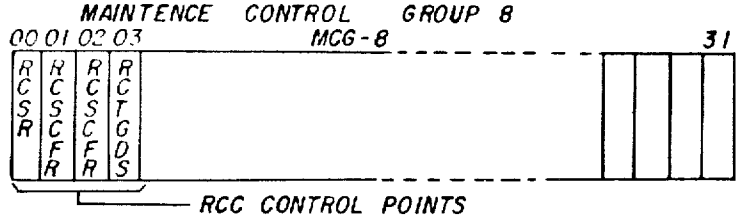


FIG. 20

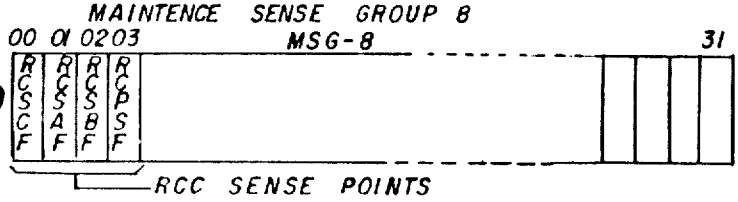
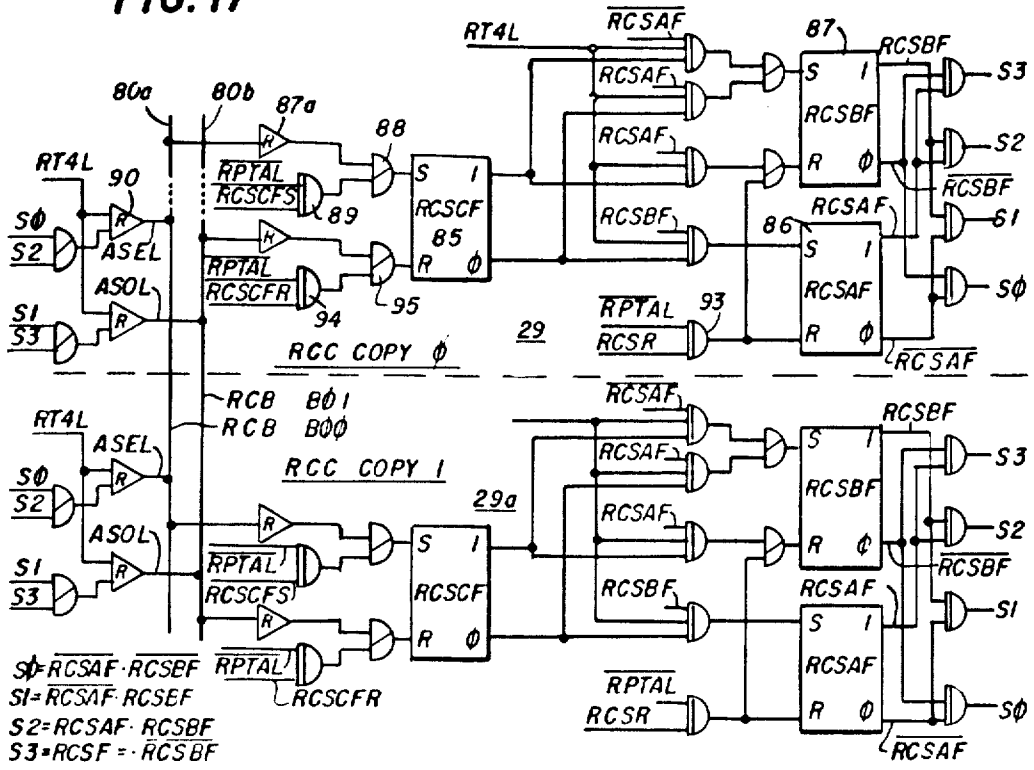


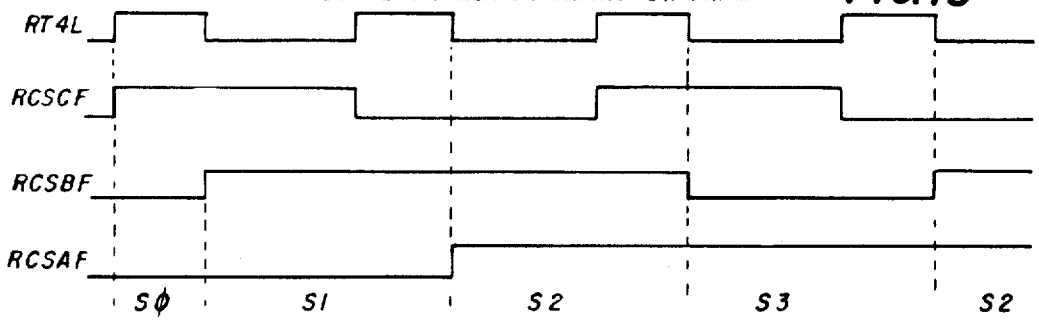
FIG. 17

DUPLEX STATE CONTROL CIRCUIT



STATE TRANSITION TIMING DIAGRAM

FIG. 18



	STATE FLIP FLOP A	B	STATE CONTROLS C	RT4L
(S0)	0	0	0	0
S1	0	0	1	1
(S1)	0	1	1	0
S2	0	1	0	1
(S2)	1	1	0	0
S3	1	1	1	1
(S3)	1	0	1	0
S2	1	0	0	1
(S2)	1	1	0	0

FIG.21

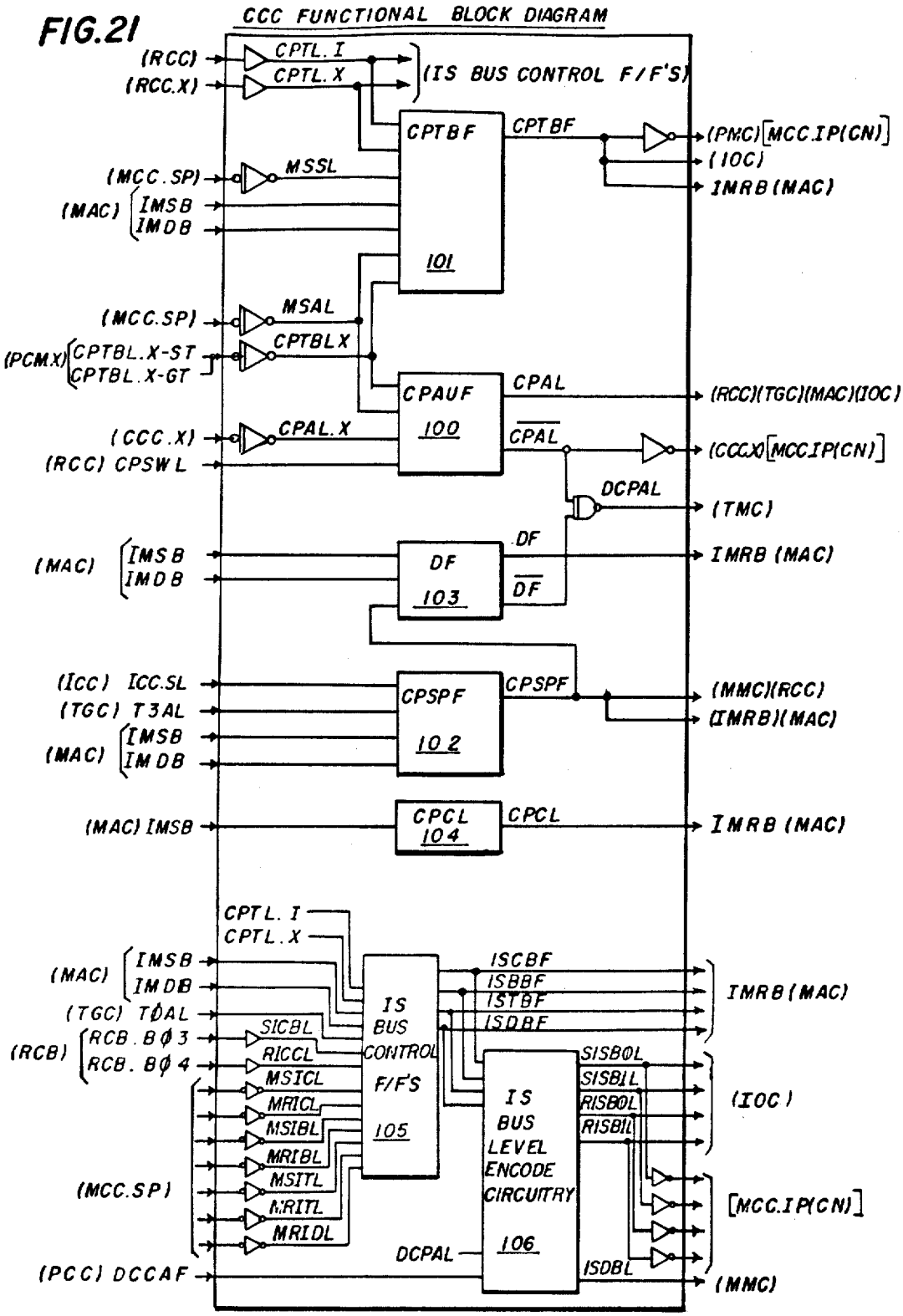
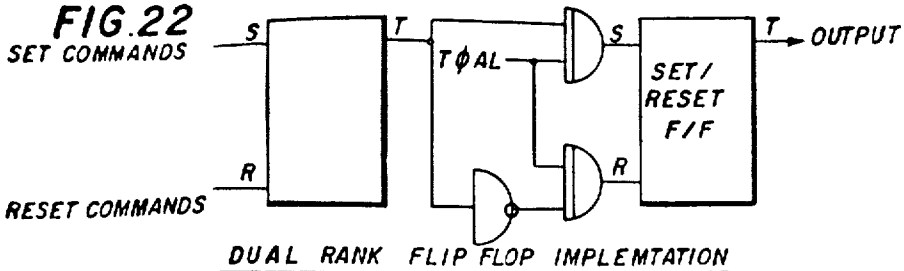

CCC FUNCTIONAL BLOCK DIAGRAM
(RCC)
(RCC.X)
CPTL.I
CPTL.X
(IS BUS CONTROL F/F'S)
CPTBF
CPTBF
(PMC)[MCC.IP(CN)]
(IOC)
IMRB(MAC)
(MCC.SP)
MSSL
(MAC)
IMSB
IMDB
101
(MCC.SP)
MSAL
(PCMX)
CPTBL.X-ST
CPTBL.X-GT
CPTBLX
CPAUF
100
CPAL
(RCC)(TGC)(MAC)(IOC)
(CCC.X)
CPAL.X
CPAL
(CCCX)[MCCIP(CN)]
(RCC) CPSWL
DCPAL
(TMC)
(MAC)
IMSB
IMDB
DF
103
DF
IMRB (MAC)
(ICC) ICC.SL
(TGC) T3AL
(MAC)
IMSB
IMDB
CPSPF
102
CPSPF
(MMC)(RCC)
(IMRB)(MAC)
(MAC) IMSB
CPCL
104
CPCL
IMRB (MAC)
CPTL.I
CPTL.X
(MAC)
IMSB
IMDB
(TGC) T0AL
(RCB)
RCB.B03
RCB.B04
SICBL
RICCL
MSICL
MRICL
MSIBL
MRIBL
MSITL
MRITL
MRIDL
(MCC.SP)
(PCC) DCCAF
IS BUS CONTROL F/F'S
105
ISCBF
ISBBF
ISTBF
ISDBF
IMRB(MAC)
IS BUS LEVEL ENCODE CIRCUITRY
106
DCPAL
SISB0L
SISB1L
RISB0L
RISB1L
(IOC)
[MCC.IP(CN)]
ISDBL
(MMC)

FIG. 22



DUAL RANK FLIP FLOP IMPLEMTATION

FIG. 23

I. S. BUS CONTROL F.F.s C	B	T	D	INST ACT CP SEND	REC	FETCH SBY CP SEND	REC	DATA ACT CP SEND	REC	FETCH SBY CP SEND	REC	IS CONFIGURATION
0	0	0	-	0	0	1	1	0	0	1	1	DUPLEX
0	1	0	-	0/1	0/1	-	0/1	0/1	0/1	-	0/1	MERGED
0	0	1	0	0	0	–	0	0	0	-	0	SIMPLEX
0	0	1	1	0	0	-	0	1	1	-	1	SIMPLEX-DIAGNOSTIC
0	1	1	0	0/1	0	-	0	0/1	0	-	0	SIMPLEX-UPDATE
0	1	1	1	0/1	0	–	0	1/0	1	-	1	SIMPLEX-UPDATE-DIAG.
1	0	0	-	1	1	0	0	1	1	0	0	DUPLEX
1	1	0	-	1/0	1/0	-	1/0	1/0	1/0	–	1/0	MERGED
1	0	1	0	1	1	-	1	1	1	-	1	SIMPLEX
1	0	1	1	1	1	–	1	0	0	–	0	SIMPLEX-DIAGNOSTIC
1	1	1	0	1/0	1	-	1	1/0	1	-	1	SIMPLEX-UPDATE
1	1	1	1	1/0	1	-	1	0/1	0	-	0	SIMPLEX-UPDATE-DIAG.

IS BUS CONTROLS

AND RESULTING CONFIGURATIONS

FIG. 24

IS BUS CONTROL LEVEL EQUATIONS

DEFINITION OF TERMS:

- C - ISCBF BUS CONTROL FLIP-FLOP OUTPUT
- B - ISBBF BUS CONTROL FLIP-FLOP OUTPUT
- T - ISTBF BUS CONTROL FLIP-FLOP OUTPUT
- D - ISDBF BUS CONTROL FLIP-FLOP OUTPUT
- DCPAL - DIAGNOSTIC CP ACTIVITY LEVEL (DCPAL = CPAL v DF)
- DCCAF - DUAL CYCLE CONTROL A FLIP-FLOP (INPUT LEVEL FROM PCC)
- SISBØL - SEND IS* BUS Ø LEVEL
- SISB1L - SEND IS* BUS 1 LEVEL
- RISBØL - RECEIVE IS* BUS Ø LEVEL
- RISB1L - RECEIVE IS* BUS 1 LEVEL
- XEC - EXECUTE INSTRUCTION
- XECN - EXECUTE NON MEMORY INSTRUCTION

$$\text{SISBØL} = \text{DCPAL}\cdot\lceil B \vee \bar{C}\cdot(\bar{T} \vee \bar{D} \vee \overline{\text{DCCAF}} \vee \text{XEC} \vee \text{XECN}) \vee C\cdot T\cdot D\cdot\text{DCCAF}\cdot\overline{\text{XEC}}\cdot\overline{\text{XECN}}\rceil \vee \overline{\text{DCPAL}}\cdot(C\cdot\bar{B}\cdot\bar{T})$$

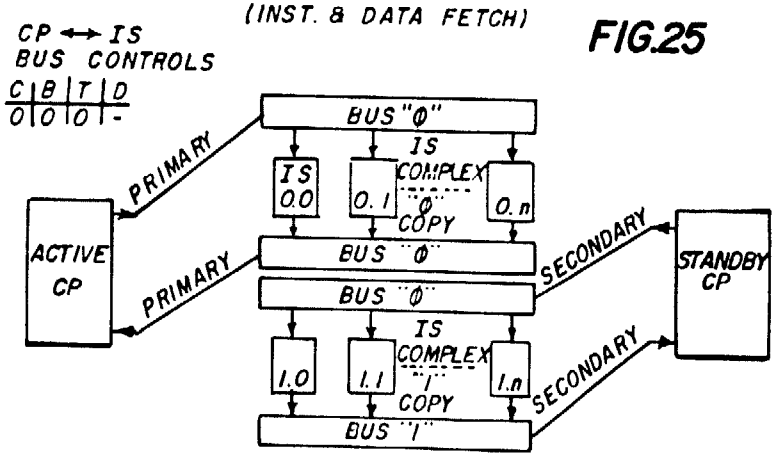
$$\text{RISBØL} = \text{DCPAL}\cdot\lceil B\cdot\bar{T} \vee \bar{C}\cdot(\bar{T} \vee \bar{D} \vee \overline{\text{DCCAF}} \vee \text{XEC} \vee \text{XECN}) \vee C\cdot T\cdot D\cdot\text{DCCAF}\cdot\overline{\text{XEC}}\cdot\overline{\text{XECN}}\rceil \vee \overline{\text{DCPAL}}\cdot\{B\cdot\bar{T} \vee \bar{C}\cdot T\cdot(\bar{D} \vee \overline{\text{DCCAF}} \vee \text{XEC} \vee \text{XECN}) \vee C\cdot(\bar{T} \vee D\ \text{DCCAF}\cdot\overline{\text{XEC}}\cdot\overline{\text{XECN}})\rceil$$

$$\text{SISB1L} = \text{DCPAL}\cdot\lceil B \vee C\cdot(\bar{T} \vee \bar{D} \vee \overline{\text{DCCAF}} \vee \text{XEC} \vee \text{XECN}) \vee \bar{C}\cdot T\cdot D\cdot\text{DCCAF}\cdot\overline{\text{XEC}}\cdot\overline{\text{XECN}}\rceil \vee \overline{\text{DCPAL}}\cdot(\bar{C}\cdot\bar{B}\cdot\bar{T})$$

$$\text{RISB1L} = \text{DCPAL}\cdot\lceil B\cdot T \vee C\cdot(\bar{T} \vee \bar{D} \vee \overline{\text{DCCAF}} \vee \text{XEC} \vee \text{XECN}) \vee \bar{C}\cdot T\cdot D\cdot\text{DCCAF}\cdot\overline{\text{XEC}}\cdot\overline{\text{XECN}}\rceil \vee \overline{\text{DCPAL}}\cdot\lceil B\cdot\bar{T} \vee C\cdot T\cdot(\bar{D} \vee \overline{\text{DCCAF}} \vee \text{XEC} \vee \text{XECN}) \vee \bar{C}\cdot(\bar{T} \vee D\cdot\text{DCCAF}\cdot\overline{\text{XEC}}\cdot\overline{\text{XECN}})\rceil$$

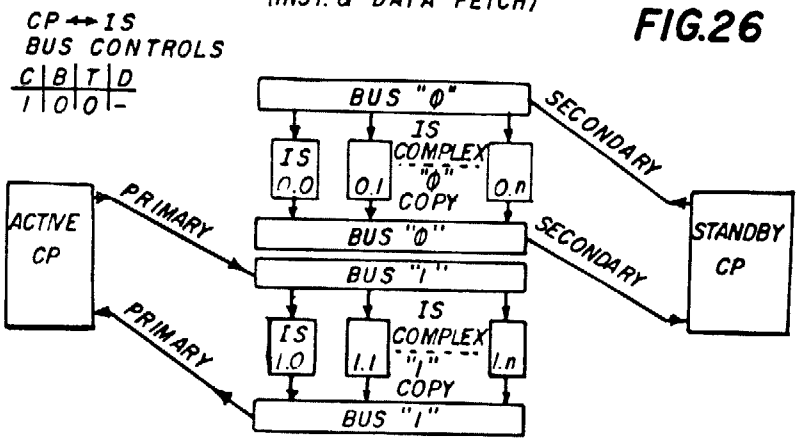
IS CONFIGURATION - DUPLEX
(INST. & DATA FETCH)

FIG.25



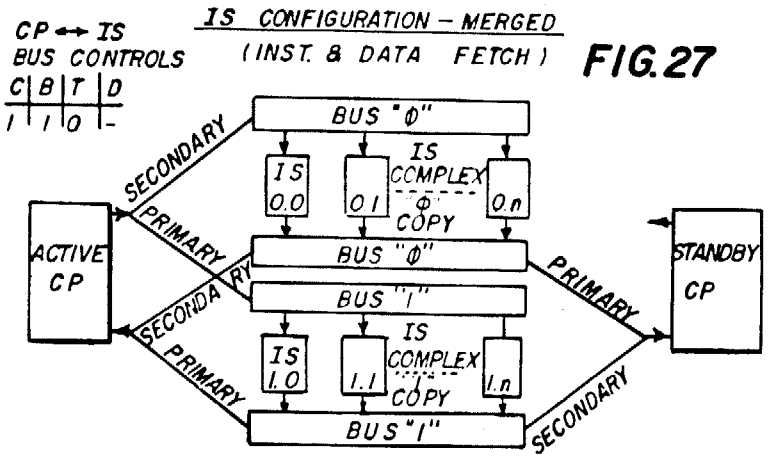
IS CONFIGURATION - DUPLEX
(INST. & DATA FETCH)

FIG.26



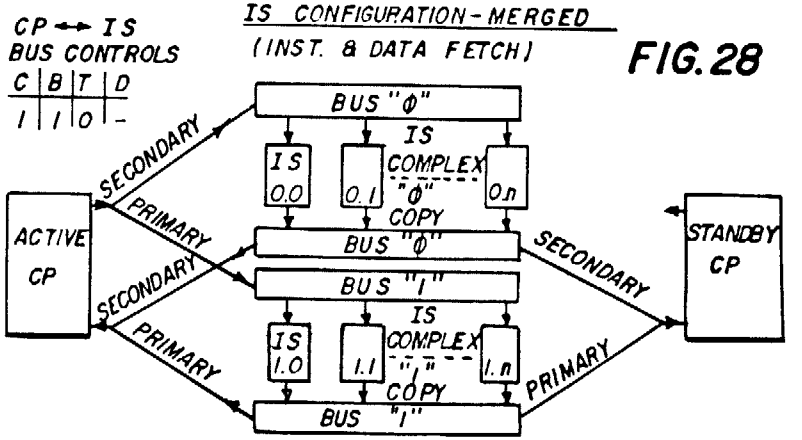
IS CONFIGURATION - MERGED
(INST. & DATA FETCH)

FIG.27



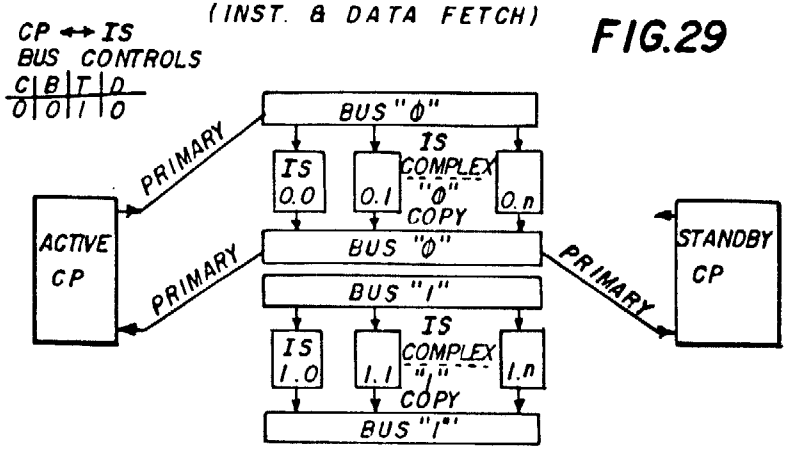
IS CONFIGURATION - MERGED
(INST. & DATA FETCH)

FIG.28



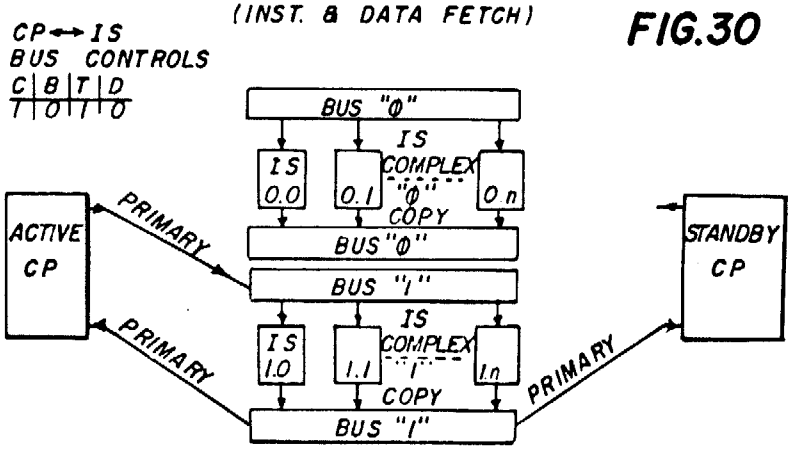
IS CONFIGURATION - SIMPLEX
(INST. & DATA FETCH)

FIG.29



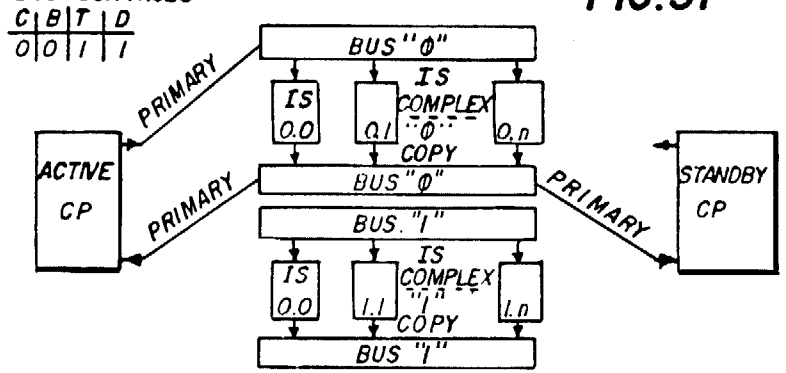
IS CONFIGURATION - SIMPLEX
(INST. & DATA FETCH)

FIG.30



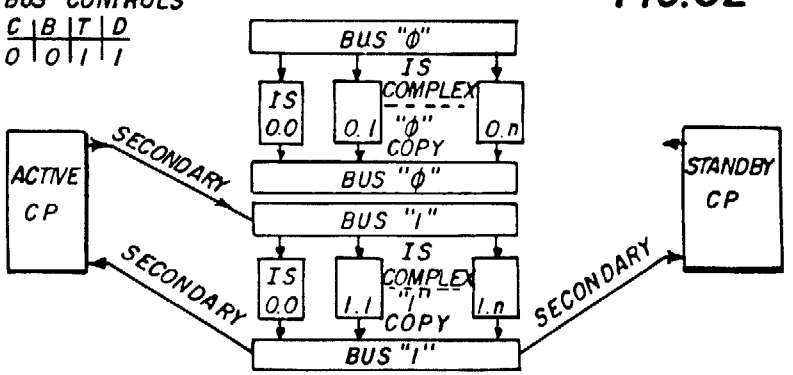
IS CONFIGURATION - SIMPLEX DIAGNOSTIC
(INST. FETCH)

FIG.31



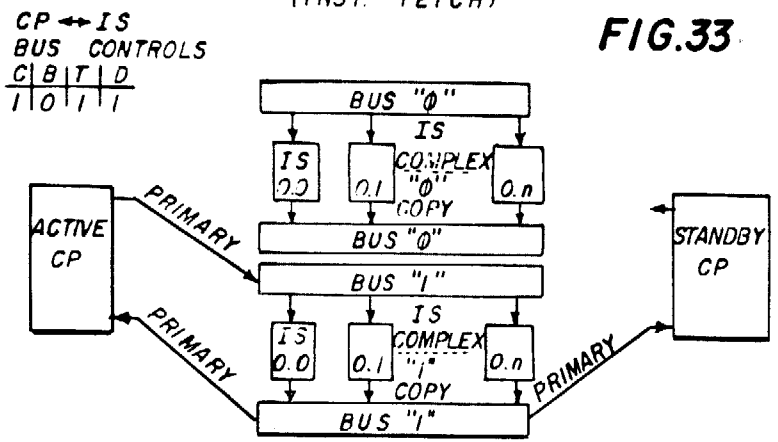
IS CONFIGURATION - SIMPLEX DIAGNOSTIC
(DATA FETCH)

FIG.32



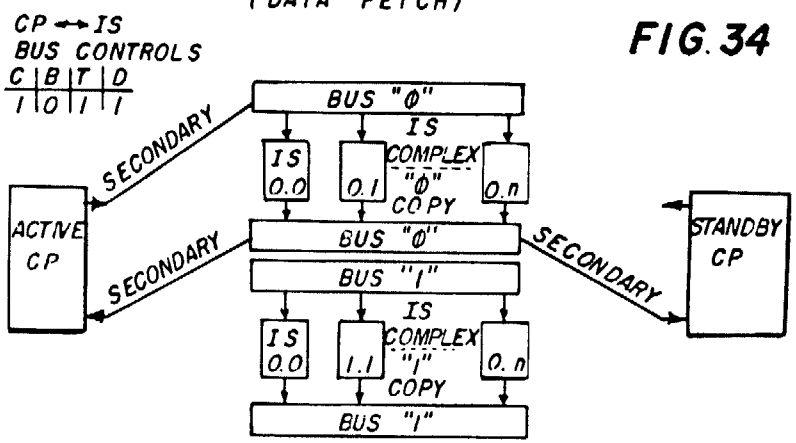
IS CONFIGURATION — SIMPLEX DIAGNOSTIC
(INST. FETCH)

FIG.33



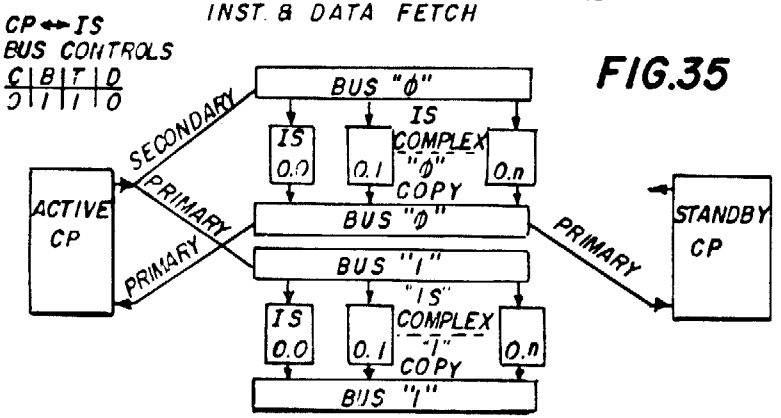
IS CONFIGURATION — SIMPLEX-DIAGNOSTIC
(DATA FETCH)

FIG.34



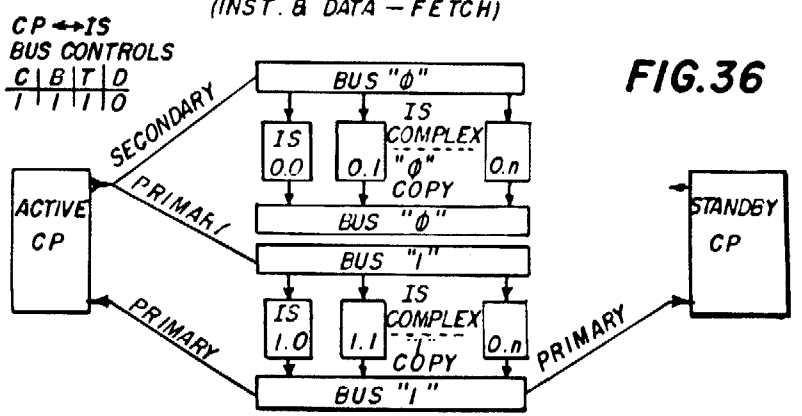
IS CONFIGURATION — SIMPLEX UPDATE
INST. & DATA FETCH

FIG.35



IS CONFIGURATION — SIMPLEX UPDATE
(INST. & DATA — FETCH)

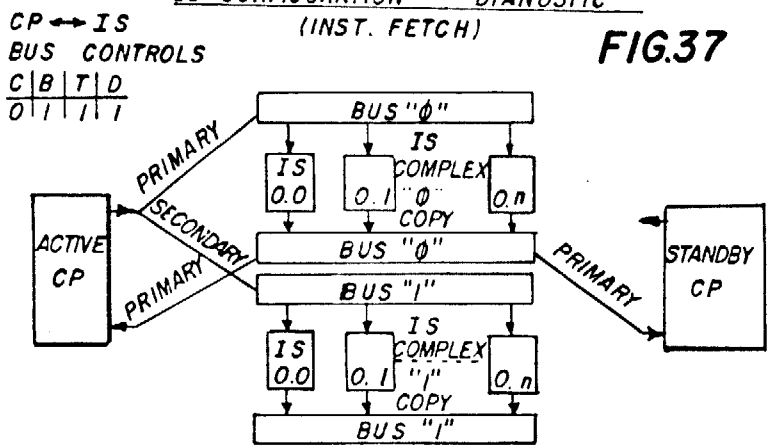
FIG.36



IS CONFIGURATION — SIMPLEX UPDATE DIANOSTIC

(INST. FETCH)

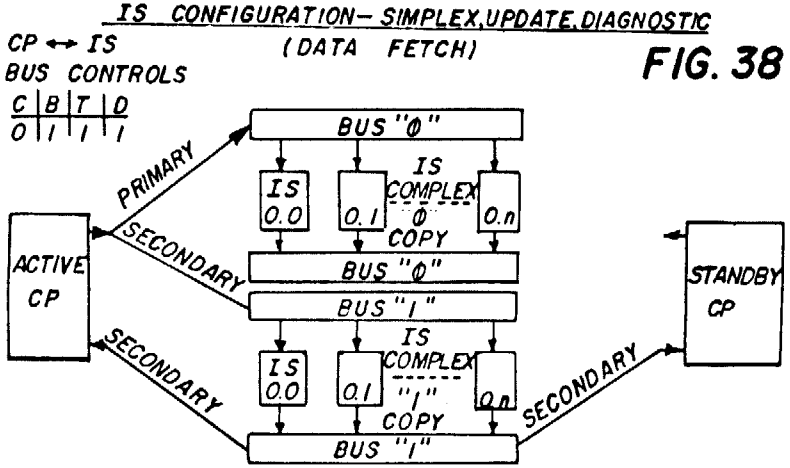
FIG.37



IS CONFIGURATION— SIMPLEX,UPDATE,DIAGNOSTIC

(DATA FETCH)

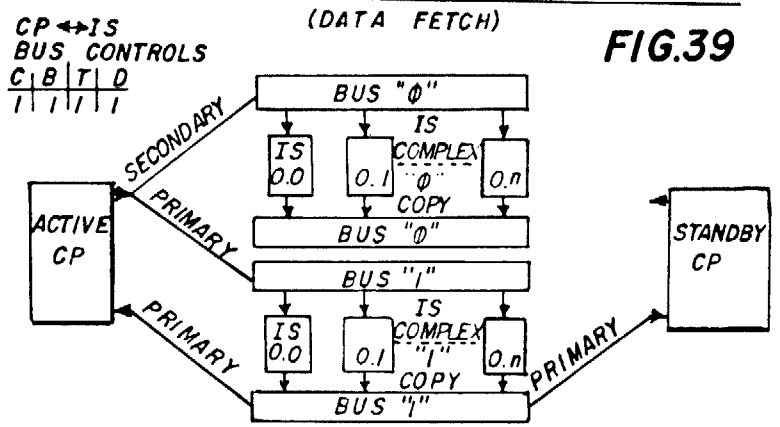
FIG. 38



IS CONFIGURATION — SIMPLEX UPDATE DIAGNOSTIC

(DATA FETCH)

FIG.39



IS CONFIGURATION — SIMPLEX UPDATE DIAGNOSTIC

(DATA FETCH)

FIG. 40

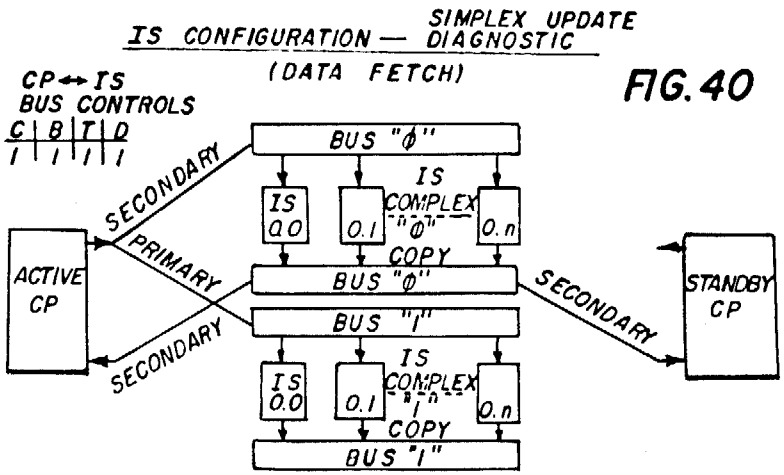


FIG. 41

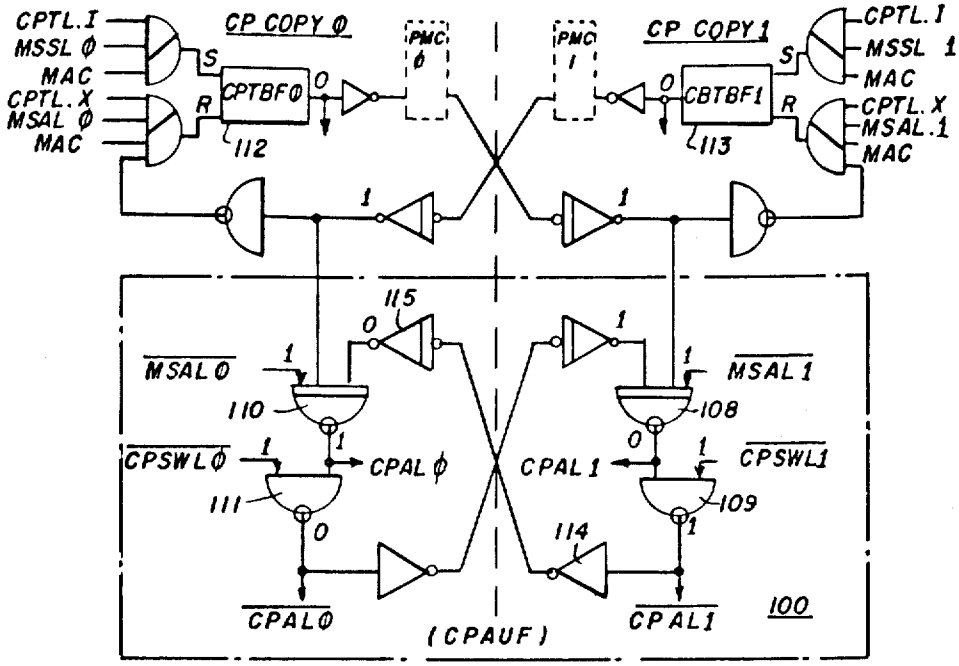


FIG. 42

ACTIVE AND STANDBY CP COPY —
SWITCHING AND NON-SWITCHING SEQUENCES

SWITCHING

ASSUME : CP COPY Ø = ACTIVE
CP COPY 1 = STANDBY
CPTBF 1 = RESET (0)
$\overline{CPSWL\emptyset}$ BECOMES 0

NON SWITCHING

ASSUME : CP COPY Ø = ACTIVE
CP COPY 1 = STANDBY
CPTBF 1 = SET (1)
$\overline{CPSWL\emptyset}$ BECOMES 0

T →

	0	1	2	3	4	5	6
CPAL Ø	1	1	1	1	1	1→0	0
$\overline{CPAL\emptyset}$	0	0	0→1	1	1	1	1
CPAL 1	0	0	0	0→1	1	1	1
$\overline{CPAL\ 1}$	1	1	1	1	1→0	0	0
$\overline{CPSWL\emptyset}$	1→0	0	0	0	0	0→1	1

	0	1	2	3	4	5	6	7	8	9
CPAL Ø	1	1	1	1	1	1	1	1	1	1
$\overline{CPAL\emptyset}$	0	0	0→1	1	1	1	1	1→0	0	0
CPAL 1	0	0	0	0→1	1	1	1	1	1→0	0
$\overline{CPAL\ 1}$	1	1	1	1	1→0	0	0	0	0	0→1
$\overline{CPSWL\emptyset}$	1→0	0	0	0	0	0	0→1	1	1	1

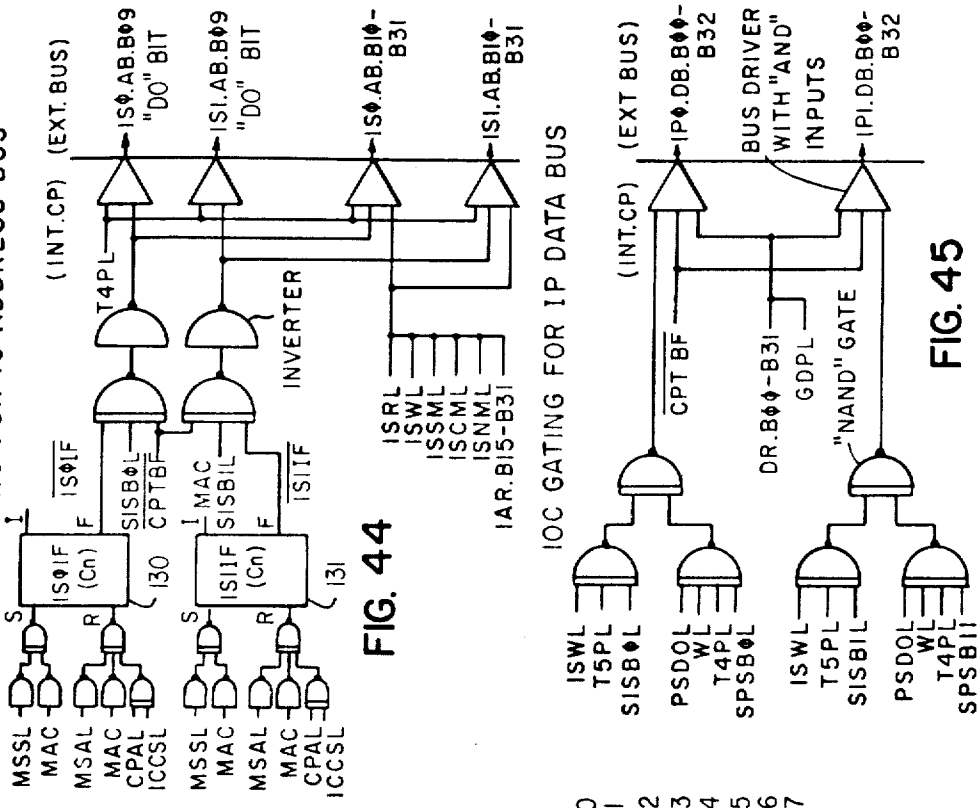


FIG. 44

FIG. 45

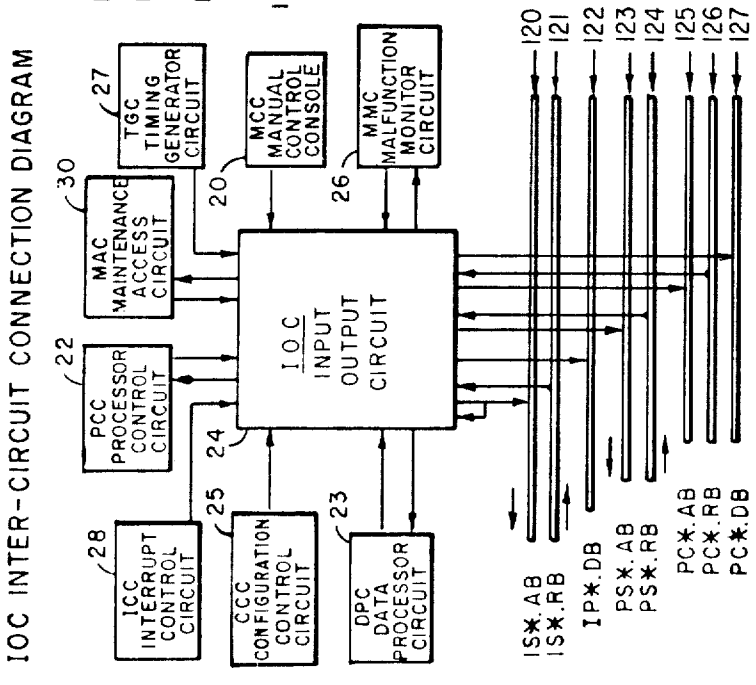


FIG. 43

FIG. 46

IO C GATING FOR IS RETURN BUSES

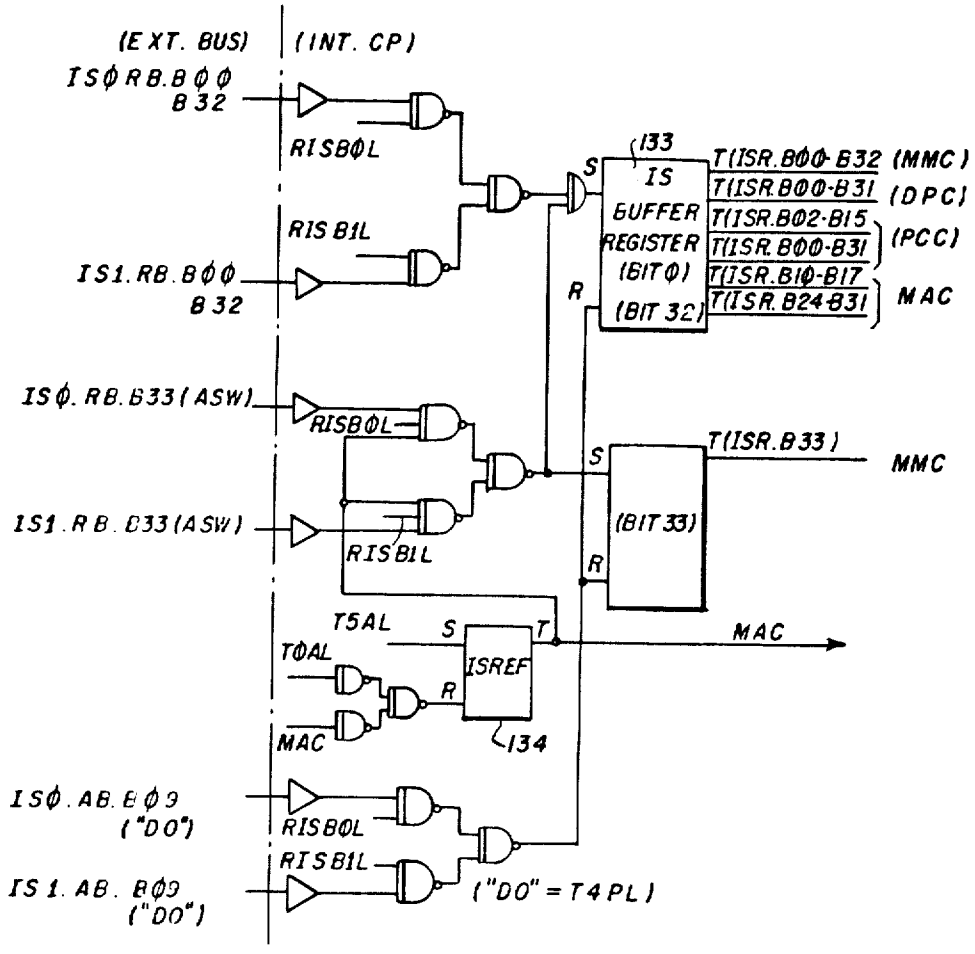


FIG. 47

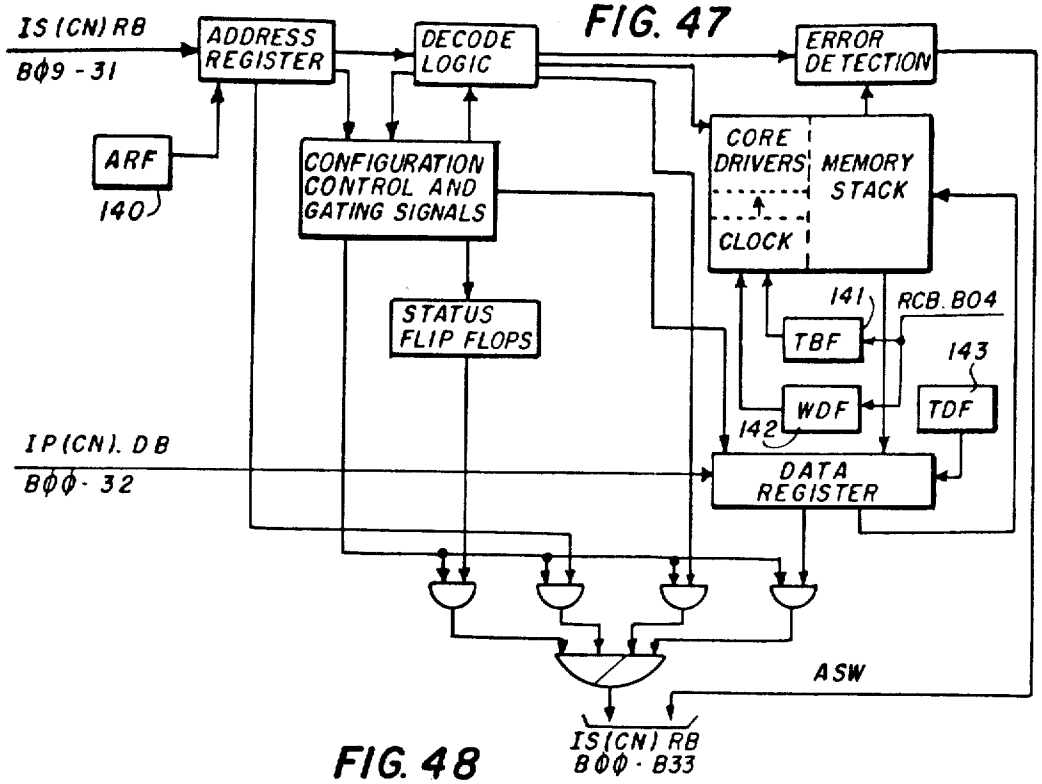
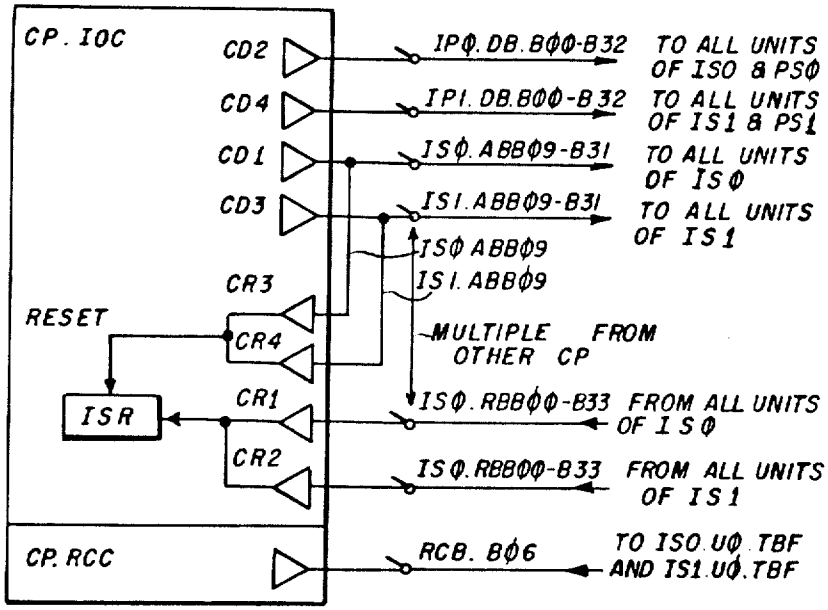
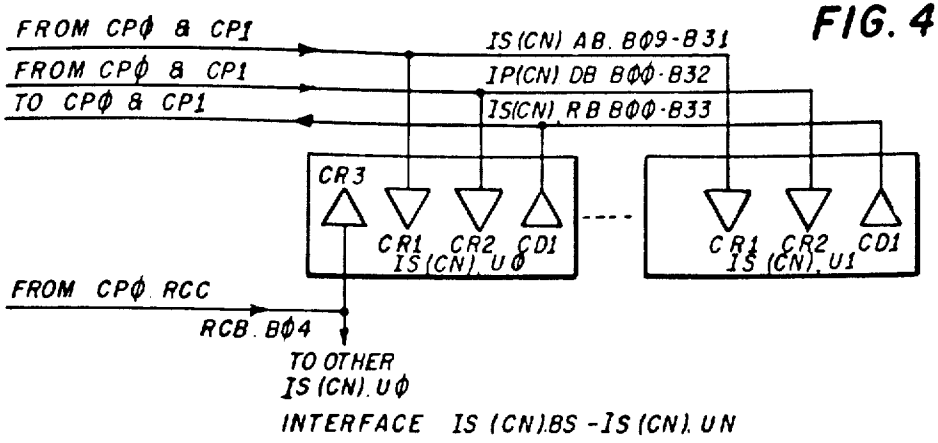


FIG. 48



INTERFACE CPΦ - ISΦ BS

FIG. 49



INTERFACE IS (CN).BS -IS (CN). UN

FIG. 50

IP (CN). DB. BØØ-B32

DR 0	DR 1	DR 2	DR 3	DR 4	DR 5	DR 6	DR 7	DR 8	DR 9	DR 10	DR 11	DR 12	DR 13	DR 14	DR 15	DR 16	DR 17	DR 18	DR 19	DR 20	DR 21	DR 22	DR 23	DR 24	DR 25	DR 26	DR 27	DR 28	DR 29	DR 30	DR 31	P 32

INSTRUCTION AND PROCESS STORE DATA BUS UNDER FOLLOWING CONDITIONS:

OPERATIONAL PULSE — WRITE
MODE PULSE — CONTROL
ADDRESS — MOD 8 = 5
UNIT SELECT — PS (CN). UN = 1

FIG. 51

IP(CN)DB BØØ-B32

																TDF
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

	AVF	$\overline{AVF}$	ARF		BVF	$\overline{BVF}$					WDF		TBF	$\overline{TBF}$	
17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32

INSTRUCTION AND PROCESS STORE DATA BUS UNDER FOLLOWING CONDITIONS:

OPERATION PULSE — WRITE
MODE PULSE — CONTROL
ADDRESS — MOD 8 = 3.5
UNIT SELECT — PS (CN). UN = 1

SYSTEM FOR RECONFIGURING CENTRAL PROCESSOR AND INSTRUCTION STORAGE COMBINATIONS

BACKGROUND AND SUMMARY

The present invention relates to a data processing system for use in connection with a digital communication system; and more particularly, it relates to such systems having duplicate copies of central processors and instruction storage means for obtaining higher reliability, and wherein circuits are provided, upon the detection of a fault in one of these units, for reconfiguring the combination so as to provide an active copy of central processor and primary instruction store which is operative and capable of performing the required functions.

The invention will be described in connection with a particular system as disclosed in copending, co-owned applications of Brenski, et al., entitled "Control Complex for TSPS Telephone System", Ser. No. 289,718, filed Sept. 15, 1972, and Schulte, et al., entitled "Maintenance Access Circuit for Central Processor of Digital Communication System", filed Jan. 2, 1972, Ser. No. 320,020. The subject matter of these applications are incorporated herein by reference.

In brief, the present invention deals with recovery control circuitry which monitors malfunction detection circuits which cause system recovery program interrupts. The detection inputs to the recovery control circuit (RCC) are produced either by timing generation check circuitry in the timing monitor circuit, error levels detected in the data processing circuit, or by the recovery program timer or an active unit change detected by the timing generator circuit, or under program direction by a specific instruction executed by the processor control circuit, called a HELP instruction.

Only the RCC in the active central processor accepts triggers or inputs, and initiates system recovery action. The RCC in the standby central processor is kept in synchronism with the active RCC but cannot bring about independent change.

When an input or "trigger" is received at the active RCC, it executes a wired-logic reconfiguration program and then requests the interrupt control circuit to execute a system recovery program interrupt which causes the system to enter into a system recovery program designed to isolate the fault. If the system recovery program cannot be completed (i.e., the configuration currently including the active central processor and primary instruction store is not operable), another RCC trigger occurs.

Each consecutive RCC trigger causes the recovery control circuit to force one of the four combinations of central processor and instruction store. When an operative combination or configuration is found, the system recovery program completes the recovery and reconfiguration process without further intervention by the recovery control circuit.

The configuration control circuit responds to the switch signals generated by the recovery control circuit as it reconfigures the CP/IS configuration; and the input/output circuit brings about a change in primary instruction store, when directed by switching to the indicated instruction store bus system.

THE DRAWING

FIG. **1** is a functional block diagram of a TSPS System, including a Control and Maintenance Complex;

FIG. **2** is a functional block diagram showing redundant copies of the Central Processor and their associated busing systems;

FIG. **2A** is a functional block diagram showing communication between both copies of the Central Processor and duplicate copies of the Instruction Store, Process Store, and Peripheral Controller;

FIG. **3** is a functional block diagram of the Timing Generator Circuit of the Central Processor;

FIG. **4** is a functional block diagram of the Processor Control Circuit of the Central Processor;

FIG. **5** is a functional block diagram of the Data Processing Circuit of the Central Processor;

FIG. **6** is a functional block diagram of the Input/Output Circuit of the Central Processor; pg,5

FIG. **7** is a functional block diagram of the Malfunction Monitor Circuit of the Central Processor;

FIG. **8** is a functional block diagram of the Timing Monitor Circuit of the Central Processor;

FIG. **9** is a functional block diagram of the Interrupt Control Circuit of the Central Processor;

FIG. **10** is a functional block diagram of the Recovery Control Circuit of the Central Processor;

FIG. **11** is a functional block diagram of the Configuration Control Circuit of the Central Processor;

FIG. **12** is a functional block diagram of the Malfunction Monitor Circuit of the Central Processor;

FIG. **13** is a functional block diagram of duplicate copies of the Recovery Control Circuit;

FIG. **14** is a diagrammatic showing of the various states that are assumed by the Recovery Control Circuit;

FIG. **15** is a timing chart for the timing cycle of the RCC;

FIG. **16** is a state transition table illustrating the various signals generated by the RCC as it undergoes state transition;

FIG. **17** is a block diagram of the duplex state control circuitry;

FIG. **18** is a timing diagram and sequence diagram for the state control circuitry of FIG. **17**;

FIGS. **19** and **20** indicate respectively the maintenance control and sense groups for RCC;

FIG. **21** is a block diagram of a portion of the Configuration Control Circuit;

FIG. **22** is a functional block diagram of a dual rank flip-flop;

FIG. **23** is a chart showing the IS bus controls;

FIG. **24** is a table indicating the IS bus control level equations;

FIGS. **25–40** illustrate the various IS/CP configurations;

FIG. **41** is a logic diagram illustrating the Central Processor Active Unit Flip-Flop;

FIG. **42** is a sequence chart illustrating the operation of the circuitry of FIG. **41**;

FIG. **43** is a functional flow diagram of the input/output circuit;

FIGS. **44–46** are functional block diagrams illustrating IOC gating for the various Instruction Store Buses;

FIG. **47** is a functional block diagram of an Instruction Store Unit;

FIGS. **48, 49** are schematic diagrams showing the interface between the IS and CP, and between individual units of each IS;

FIG. **50** shows the IS Data Bus format for one Write Control mode; and

FIG. **51** shows the IS Data Bus format for another Write Control mode.

DETAILED DESCRIPTION

I. Introduction—TSPS

The primary function of the TSPS System is to provide data processor control of the various functions in toll calls which in the past have been performed by operators but have not required the exercise of discretion on the part of the operator. At the same time, the system must permit operator intervention, as required. Thus, various trunks from an end office to a toll center pass through the TSPS System, and these are commonly referred to as Access Trunks, functionally illustrated in FIG. **1** by the block **10.**

The access trunks **10** are connected to and pass through access trunk circuits in a network complex **11** which is physically located at the same location as the TSPS base unit, and the network complex **11** permits the system to access each individual trunk line to open it or control it, or to signal in either direction. There is no switching or re-routing of trunks or calls at this location. Each trunk originating at a particular end office is permanently wired to a single termination in a remote toll office while passing through a TSPS network complex or trunk circuit en route.

The various access trunks may originate at different end offices, but regardless of origin, they are served in common by the TSPS System and the operators and traffic office facilities associated with that system. Hence, the equipment interfaces with various auxiliary equipment incidental to gaining access to the throughput access trunks, including remote operator positions, equipment trunks, magnetic tape equipment for recording charges, and various other equipment diagrammatically illustrated by the block **12.** Additional details regarding the network complex **11** and the auxiliary equipment and communication lines **12** for a TSPS System may be obtained from the *Bell System Technical Journal* of December, 1970, Vol. 49, No. 10.

The present invention is more particularly directed to one aspect of the data processor which controls the telephony—namely the maintenance circuitry in the Central Processor (CP) which controls the systems and performs call processing as well as maintenance and recovery functions. The Central Processor is shown in simplex form within the chain block **17** of FIG. **1.**

It will be observed that the telephony equipment is about three orders of magnitude in time slower, on the average, than is necessary to execute individual instructions in modern high-speed digital computers. For example, for the present system a clock increment for the Central Processor is 4 microseconds whereas the trunk circuits are sampled every 10 milliseconds. Hence many functions can be performed in the Central Processor, including internal and external maintenance, table look-ups, computations, monitoring of different access trunks, system recovery from a detected fault, etc. between the expected changes in a given trunk.

The TSPS System uses a stored program control as a means of attaining flexibility for varied operating conditions. Reliability is attained by duplicating hardware wherever possible. A stored program control system consists of memories for instructions and data and a processing unit which performs operations, dictated by the stored instructions, to monitor and control peripheral equipment.

A Control and Maintenance Complex (CMC) contains the Instruction Store Complex (IS*), Process Store Complex (PS*), Peripheral Unit Complex (PC*), and the Central Processor Complex (CP*). The asterisk designates all of the circuitry associated with a complex, including the duplicate copy, if applicable.

The interface between the telephony equipment and the data processor is the Peripheral Unit Complex which includes a number of sense matrices **13** and control matrices **14** together with a Peripheral Controller diagrammatically indicated by the chain block **15.**

The principal elements of the data processing circuitry include the Central Processor (CP) **17,** a Process Store (PS) enclosed within the chain block **18,** and an Instruction Store (IS) enclosed within the chain block **19.** A computer operator or maintenance man may gain manual access into the Central Processor **17** by means of a manual control console **20,** if desired or necessary.

The Instruction Store (IS) **19** which consists of two copies, contains the stored programs. Each copy has up to eight units as shown in block **19** and includes two types of memory:

1. A read-only unit **19***a* containing a maximum of 16,384 thirty-three bit words.
2. Core Memory in remaining units containing a maximum of seven units of 16,384 thirty-three bit words per unit. Individual words are read from or written into IS by CP **17,** as will be more full described below.

Each IS unit **19** of the eight possible is similar; and they are of conventional design including an Address Register **19***b* receiving digital signals representative of a particular word desired to be accessed (for reading or writing as the case may be). This data is decoded in the Decode Logic Circuit **19***c*; and the recovered data is sensed by sense amplifiers **19***d* and buffered in a Memory Data Register **19***e* which also communicates with the Central Processor **17.**

The Process Store (PS) **18** contains call processing data generated by the program. The PS (also in duplicate copies) comprises Core Memory units **18***a* containing a maximum of eight units of 16,384 thirty-three bit words for each copy. Individual words are read from or written into PS by CP in a manner similar to the accessing of the Instruction Store **19,** just described. That is, an Address Register **18***b* receives the signals representative of a particular location desired to be accessed; and this information is decoded in a conventional Decode Logic Circuit **18***c*. The recovered information is sensed by sense amplifiers **18***d* and buffered in Memory Data Register **18***e*.

The CMC communicates with the telephony and switching equipment through matrices **13, 14** of sense and control devices. Any number of known design elements will work insofar as the instant invention is concerned. The sense and control matrices **13, 14** are each organized into 32 bit sense words and 32 bit control words. On command of CP, PC samples a sense word and returns the values of the 32 sense points to CP. Each control point is a bistable switch or device. To control telephone and input/output equipment, CP sets

a word of control points through PC. PC together with the sense and control matrices comprise the Peripheral Unit Complex (PU).

CP sequentially reads and executes instructions which comprise the program, from IS. The CP reads and executes most instructions in 4 microseconds (one machine cycle time). Those instructions that access IS require 8 microseconds require two machine cycles to be executed and are referred to as "dual cycle" instructions.

The instructions obtained from the IS can be considered "Directives" to the CP specifying that it is to perform one of the following operations:

a. Change and/or transfer information inside the CP in accordance with some fixed rule.

b. Communicate with the IS or PS by requesting the IS/PS to either;

1. Read a 33 bit word from a specified location, or

2. Write a 33 bit word into a specified location.

c. Communicate with the PC by requesting PC to either;

1. Read a specified 32 bit from sense point word, or

2. Write into a specified 32 bit control point word.

d. Perform maintenance operations internal to CP by either;

1. Reading from a maintenance sense group, or

2. Writing into a maintenance control group.

The Control and Maintenance Complex may be viewed from two levels: a processing level and a maintenance level. At the processing level (which includes the control and maintenance of the telephone equipment) the CMC appears to be an unduplicated, single processor system as in FIG. **1.** At the maintenance level (which here refers only to CMC maintenance) the CMC consists of duplicated copies of the units in each complex, as seen in FIG. **2.**

The duplication within the CMC is provided for three purposes:

1. In the event that a failed unit is placed out-of-service, its copy provides continued operation of the CMC.

2. Matching between copies provides the primary means of detecting failures.

3. In-service units can be used to diagnose an out-of-service unit and report the diagnostic results.

Each complex within the CMC may be reconfigured (with respect to in-service and out-of-service units) independently of the other complexes to provide higher overall CMC reliability.

The CMC operation is monitored by internal checking hardware. In the event of a malfunction (misbehavior due either to noise or to failure), the CP is forced into the execution of a recovery program by a maintenance interrupt.

When the malfunction is due to failure, the recovery program will find the failed copy and place it out-of-service. When at least one complete set of units in each complex can be placed in-service, the fault recovery program will terminate after reconfigurating the CMC to an operational system. If a good set of units in each complex cannot be found, the fault recovery program continues until manual intervention occurs.

To facilitate the recovery operation, a hierarchy of in-service copies are defined:

1. One Central Processor must always be in the active state, only the active CP can change the configuration of the CMC,

2. If the other CP is in-service, that CP is the standby CP, and

3. The in-service copies of Instruction Store, Process Store, and Peripheral Control Units are designated as primary and secondary where the primary copies are associated with the active CP.

Each Peripheral Control Unit may also be designated as active or standby; only the active Peripheral Control Unit controls telephone equipment through the sense and control points. Further, the duplicate copies of IS are designated active and standby according to which one (called the "active" one) is associated with the primary CP.

II. The Central Processor—An Overview

The CP circuits provide two specific functions: processing and maintenance. The processing circuits provide a general purpose computer without the ability to recover from hardware failures. The maintenance circuits together with the processing circuits provide the CMC with recovery capability.

The Central Processor is divided into 10 circuits. The first four provide the processing function.

1. Timing Generator Circuit (TGC), designated **21,**

2. Processor Control Circuit (PCC), **22,**

3. Data Processing Circuit (DPC), **23,** and

4. Input/Output Circuit (IOC), **24.**

The above four processing circuits are described herein only to the extent necessary to understand the present invention. Additional details may be found in the above-referenced copending application of Brenski, et al., Ser. No. 289,718.

The remaining circuits in the CP provide the maintenance function and these include:

5. Configuration Control Circuit (CCC) **25,**

6. Malfunction Monitor Circuit (MMC) **26,**

7. Timing Monitor Circuit (TMC) **27,**

8. Interrupt Control Circuit (ICC) **28,**

9. Recovery Control Circuit (RCC) **29,** and

10. Maintenance Access Circuit (MAC) **30.**

In FIG. **2,** there is shown duplicate copies of each of the above circuits in the Central Processor, with like circuits having identical reference numerals.

Turning back to FIG. **1,** a pair of Peripheral Controllers is associated with each Peripheral Control Unit (PCU). Each Peripheral Controller **15** includes the following circuits which are also described in more detail in the above-references Brenski, et al., application Ser. No. 289,718:

1. A Matrix Access Circuit **33,**

2. An Address Register Circuit **34,**

3. A Data Register Circuit **35,**

4. A Timing Generator Circuit **36,**

5. A Maintenance Status Circuit **37,**

6. An Address Decode Circuit **38,** and

7. A Control Decode Circuit **39.**

The functional interface between the Central Processor, and other system equipment, is shown in functional block diagram form in FIG. **2A.** As can be seen, there is intercommunication between both copies of the Central Processor designated **17** and **17***a* respectively and the manual control console. Maintenance personnel can monitor the status and manually reconfigure the control and maintenance complex from this console.

As can also be seen in FIG. 2A, both Central Processor copies have direct, two-way communication links between each other, via internal bus **35**, and with both copies of Instruction Store, designated **36** and **37** respectively, via their associated bus systems **38** and **39**. Similar communication is provided with the Process Store, and the Peripheral Controllers. This interface is provided by six separate bus systems.

I. An Instruction Store copy **0** bus system (IS0.BS) is designated **38**. This interfaces both copies **17***a*, **17** of the Central Processor via buses **41, 42** with each of the 8 units (IS0.U0 through IS0.U7) that form Instruction Store copy **0** (IS0) generally designated **36.**

II. An Instruction Store copy **1** bus system (ISI.BS) is designated **39**. This interfaces both copies of the Central Processor via buses **43, 44** with each of the 8 units (IS1.U0) through IS1.U7) that form Instruction Store copy **1** (ISI), generally designated **37.**

III. A Process Store copy **0** bus system (PS0.BS) is designated **45**; and it interfaces both copies of the Central Processor with each of the 8 units (PS0.U0 through PS0.U7) that make up Process Store copy **0** (PS0), generally designated **46.**

IV. A Process Store copy **1** bus system (PS1.BS) is designated **47**; and it interfaces both copies of the Central Processor with each of the 8 units (PS1.U0 through PS1.U7) that make up Process Store copy **1** (PS1), generally designated **48.**

V. A Peripheral Controller copy **0** bus system (PS0.BS) is designated **49**; and it interfaces both copies of the Central Processor with each of the 8 Peripheral Controllers (PC0.U0 through PC0.U7) in Peripheral Control copy **0** (PC0), generally designated **50.**

VI. A Peripheral Controller copy **1** bus system (PC1.BS) is designated **51**; and it interfaces both copies of the Central Processor with each of the 8 Peripheral Controllers (PC1.U0 through PC1.U7) in Peripheral Control copy **1** (PC1), generally designated **52.**

Each copy of the Peripheral Control bus system contains an address bus (PC0.AB and PC1.AB), a return bus (PC0.RB and PC1.RB), and a data bus (PC0.DB and PC1.DB). Each copy of the process store bus system contains an address bus (PS0.AB and PS1.AB) and a return bus (PS0.RB and PS1.RB). Each copy of the Instruction Store bus system contains an address bus (IS0.AB and IS1.AB, and a return bus (IS0.RB and IS1.RB). Each copy **0** of the Instruction Store bus system and the ProcessStore bus system share the same data bus: Instruction Store and Process Store copy **0** data bus (IP0.DB). Each copy **1** of the Instruction Store bus system and the Process Store bus system also share the same data bus: Instruction Store and Process Store copy **1** data bus (IPI.DB).

This data bus sharing by Instruction Store and Process Store affects the sequence of instructions that are to be executed by the Central Processor. An instruction directing the Central Processor to access (read from or write into) Process Store requires only one machine cycle, while an instruction directing the Central Processor to access Instruction Store requires two machine cycles. This means that the Central Processor can execute Process Store instructions in sequence, one after the other, for as long as needed, and it can also execute an Instruction Store instruction immediately following a Process Store instruction. However, it cannot execute two Instruction Store instructions, in sequence, nor can it execute a Process Store instruction immediately after an Instruction Store instruction, because of the shared data bus. The Central Processor will have been in the execution of an Instruction Store instruction only one machine cycle of the two required, when it starts executing the next instruction in sequence, and these two instructions cannot use the same data bus (IP0.DB or IPI.DB) simultaneously.

It is believed that a better understanding of the present invention will be obtained if there is an understanding of the overall function of each circuit in the CP, realizing that there are duplicate copies of the CP.

II. A. Processing Circuits of Central Processor

Timing Generator Circuit (TGC)

The Timing Generator Circuit **21** of FIGS. **1** and **2** (TGC) creates the timing intervals for the Central Processor. A more detailed functional block diagram for the TGCs of both Central Processors is shown in FIG. **3.**

The TGC includes a level generator circuit **50** and creates eight timing intervals (or "levels" as they are referred to) every 4 μseconds. Each pulse is picked off a delay line. For each timing interval, TGC produces a 500 nano second (ns) timing interval place level (PL) and a 400 ns. timing interval accept level (AL). Each sequence of 8 timing intervals is called a cycle. Nearly all sequential control in the CP is provided by the timing interval place and accept levels.

Generally, the timing interval place levels are used to gate information out of flip-flop storage while timing interval accept levels are used to accept information into flip-flop storage.

The TGC in each CP generate timing levels. To assure synchronism between CP's, Timing levels generated in the active CP control both CP's. A switching network **51** actuated by a switching control circuit **52** in each TGC transmits (if it is in the active CP) or receives the timing levels from the active TGC, and supplies them to the CP circuits. The standby CP may be stopped by directing the TGS in the standby CP to inhibit reception of timing levels. The TGS also notifies the Recovery Control Circuit **29** (RCC) and Timing Monitor Circuit **27** (TMC) for maintenance purposes whenever the CP's active/standby status changes.

Processor Control Circuit (PCC)

The PCC **22** (see FIG. **4** for a more detailed functional block diagram) includes instruction fetch and decode circuits **53** which decode each instruction and generate the control signals required to execute the instruction and to read the next instruction from IS.

The instructions are performed in the DPC **23** by a sequency of data transfers—one in each of the eight timing intervals. Each data transfer is controlled by three simultaneous command from the PCC to the DPC:

1. A register place command (generated in block **54**) which places a DPC register or circuit on the Interval Output Bus of the PCC.

2. A Bus Transfer Command (generated in bus transfer control circuits **55**) which transfers the information on the Internal Output Bus to the Internal Input Bus, and

3. A Register Accept Command (also generated in block **54**) which gates the information on the Internal Input Bus to a DPC register.

The PCCalso provides auxiliary commands to the DPC such as the selection of the function to be provided by the Logic Comparator Circuit (LCC).

Memory and peripheral unit control circuits **55** of the PCC provide the control signals to the IOC including the mode bits to be transmitted to these complexes.

The instruction fetch logic of block **53** controls an Instruction Address Register IAR, Add One Register AOR, and the instruction store read for the next instruction. The next instruction is read from the Instruction Store simultaneously with the execution of its predecessor.

The PCC also decodes the HELP instruction which is an input to the RCC that initiates a system recovery program interrupt. The instructions RMSG, WMSG, and WMCP are decoded by the PCC but are executed by the Maintenance Access Circuit **30** (MAC). The Malfunction Monitor Circuit **26** (MMC) requires decoded instructions levels from the PCC in order to sample malfunction detection circuits.

DATA PROCESSING CIRCUIT (DPC)

The DPC **23** (see also FIG. **5**) contains the registers of the CP and the circuits required to perform arithmetic, logical, decision, and data transfer operations on the information in these registers. The General Registers (GR**1**, . . . , GR**7**), in the Storage Section **56,** the Special Purpose Register (SPR), also in Storage Section **56,** and the Instruction Address Register (IAR) in the Address Section **57** are the program accessible registers. These registers and the operations which are performed on these registers by individual instructions are described more fully in the above-referenced application.

The remaining registers [Data Register (DR) and Arithmetic Register (AR) in Data Section **58,** the Selection Register (SR), and Add One Register (AOR)] and circuits (Logic Comparator Circuit (LCC), Add Circuit (ADC) the Add One Circuit (AOC), and the Bus Transfer Circuit **59** (BTC) provide the data facilities required to implement the instruction operations on the program accessible registers.

A 32 bit Internal Input Bus (IIB) **60** is the information source for all DPC registers. In general, the DPC registers and circuits as well as other CP circuits place information on the 32 bit Internal Output Bus (IOB) **61.** The Bus Transfer Circuit (BTC) **59** transmits information from the IOB **61** to the IIB **60.** The information can be transferred in six ways which include complementing or not complementing the information, exchanging **16** bit halves (with or without complementing), or shifting the information left or right one bit.

A logic and compare circuit (LCC) provides a 32 bit logical AND, NOR, or EQUIVALENCE of the AR and DR and also matches the AR and DR. The ADD Circuit (ADC) provides the sum of the left half of the AR and the right half of the AR. The ADC is used for addition and subtraction and to generate PS and PU addresses. The 17 bit Instruction Address Register (IAR) is used to address the Instruction Store. The Add-One-Circuit (AOC) increments the right most 16 bits of the IAR by one. The AOC is used to compute the next instruction address (one plus the current address) which will be used if a Program Transfer does not occur.

Input Output Circuit (IOC)

The primary function of the IOC **24** (see also FIG. **6**) is to provide the interface through which the Central Processor complex (CP*) gains access to the non-CP complexes (IS*, PS*, and PC*) via the external bus system. As seen diagrammatically in FIG. **6,** the IOC sends data and addresses from the CP to the non-CP complexes and also receives and buffers data transmitted to the CP from non-CP complexes. The external bus system, used to transmit information between CP* and the non-CP complexes, comprises the Instruction Store Address Bus (IS*.AB), Process Store Address Bus (PS*.AB), Peripheral Control Address Bus (PC*.AB), Instruction Store-Process Store Data Bus (IP*.DB), Peripheral Control Data Bus (PC*DB), Instruction Store Return Bus (IS*.RB), Process Store Return Bus (PS*.RB), and Peripheral Control Return Bus (PC*.RB).

Each bus consists of two copies which are associated with corresponding copies of IS*, PS*, and PC*. At the processing level, the IOC may be considered to use both copies of the bus without distinction between the copies. To provide the reconfiguration capability (maintenance level), the IOC transmits on or receives from copy **0,** copy **1,** or both copies of a particular bus. The choice of bus copies is determined by the Configuration Control Circuit **25.**

There are three buffer registers in the IOC: the Instruction Store Register (ISR) designated **62,** the Process Store Register (PSR) **63,** and the Peripheral Unit Register **64.** These registers communicate with both copies of the Return Buses from IS, PS and PU respectively; and they send received data to the DPC **23** and MMC **26,** as shown.

II. B. Maintenance Circuits

The functions performed by the CP maintenance circuits include the following:

1. System configuration control (CCC **25**),
2. Malfunction detection (MMC **26,** TMC **27,** DPC **23**),
3. Recovery program initiation (ICC **28**),
4. Recovery program monitoring (RCC **29,** TMC **27**),
5. Maintenance program access to CP circuits (MAC **30,** MMC **26**), and
6. Manual system control (MCC **20**).

The CMC detects malfunctions as follows:

1. By matching, between CP copies, all data transfers in the CP Data Processing Circuit (MMC),
2. By parity checking of all memory read operations (MMC),
3. By monitoring internal checks by the IS*, PS*, and PC* (all-seems-well checks),
4. Address echo matching of addresses sent to IS*, PS*, and PC* with the echo address returned by the complex (DPC),
5. Timing level generation checking (TMC), and
6. Excess program time checking (DPC).

When a malfunction is detected by MMC **26,** the Interrupt Control Circuit (ICC) **28** may initiate a maintenance interrupt to a recovery program. The recovery program attempts to locate the faulty unit, remove it from service, and reconfigure the complexes to a working system. The execution of the recovery programs are monitored by the TMC **27** and the RCC **29.** The system recovery program is initiated (reinitiated) by the TMC **27** and the RCC **29** when higher level recovery is required. The Timing Monitor Circuit monitors recovery programs through the Recovery Program Timer (RPT)

in the TMC 27 (see FIG. 8). If a recovery program fails to remain in synchronism with this timer, the TMC initiates (or re-initiates) the system recovery program through the Recovery Control Circuit. The execution of a HELP instruction may also initiate (re-initiate) the system recovery program directly through the RCC.

MALFUNCTION MONITOR CIRCUIT (MMC)

The MMC 26 (seen in more detail in FIG. 7) provides the following maintenance functions:

1. Detection of malfunctions during the execution of programs,
2. Classification of malfunctions into CP*, IS*, PS*, and PC* caused malfunctions,
3. Indication of a CP, IS, PS, or PC malfunction occurrence to ICC in each CP,
4. Storage of malfunction indications on error flip-flops,
5. Storage of the address of the instruction being executed when a maintenance interrupt occurs,
6. Special facilities for use by recovery programs,
7. Access to standby CP for extraction of diagnostic data through the match facilities,
8. Facility to monitor standy CP executing off line maintenance programs (Parallel Mode), and
9. Facilities for routining the MMC itself.

The Malfunction Monitor Circuit **26,** shown is divided into the following three sub-circuits:

1. MAtch Network (MAN), designated **70,**
2. PArity Network (PAN), designated **71,** and
3. Malfunction Analysis Circuit (MFAC), designated **72.**

The MAtch Network (MAN) provides all inter-Central Processor matching facilities. In addition to malfunction detection, the match network can be used for extracting diagnostic data from the standby CP for routining the match network itself. The control logic within the MAN controls the match network according to match modes selected by the maintenance programs.

The PArity Network 71 (PAN) contains all the Parity Circuits used in checking the transmission and storage of information in the Instruction Store (IS*) and Process Store (PS*).

The Malfunction Analysis Circuit **72** monitors malfunction detection signals from

1. MAN (inter CP matching),
2. PAN (parity checks),
3. DPC (address echo match), and
4. IOC (all-seems-well signals).

The malfunction detection signals are sampled according to the timing intervals and instructions being executed. When a malfunction is detected an error flip-flot associated with the detection circuit is set to be used by maintenance program to isolate the source of the malfunction.

The malfunction analysis circuit classifies the malfunction according to its most likely cause (CP*, IS*, PS*, or PC*) and a corresponding error level (CPEL, ISEL, PSEL, or PUEL) is sent to the Interrupt Control Circuit (ICC) in both CP's.

Timing Monitor Circuit (TMC)

The TMC 27 (FIG. 8) provides three timing malfunction detection circuits:

1. Timing check circuit **73** which checks the timing levels generated by TGC,
2. A Real Time Error FF (RTEIF) **74** which monitors the state of the overflow of the Real Time Timer RTT in DPC, and
3. A Recovery Program Timer (RPT) **75** which monitors recovery program execution.

Most failures of the active Timing Generator Circuit (TGC) do not cause inter-CP mismatches. These failures are detected by the TGC checking circuitry of the active TMC. The output of this Circuit is monitored by the active Recovery Control Circuit (RCC).

Failures of the standby TGC will cause inter-CP mismatches and are detected by the Malfunction Monitor Circuit. The standby RCC ignores error outputs of the standby TMC.

RTT, which is located in the DPC, has both an operational and a maintenance function. It provides real time synchronization for the operational programs and a sanity check on the execution. The RTT is a fourteen bit counter which is incremented by one every CP cycle (4 microseconds). The program may read or modify RTT through the Special Purpose Register (SPR). In this manner, RTT can provide time intervals of up to 65 milliseconds for the operational programs. The programs, however, must reinitialize RTT often enough to prevent the overflow from occurring. The active RCC monitors the RTT overflow. If the overflow occurs, RTEIF is set and the RCC initiates the system recovery operation.

RPT checks the execution of the Recovery programs. Rpt is a seven bit counter which, when enabled, is incremented by one every CP cycle. RPT is enabled whenever a maintenance interrupt occurs and is disabled by the recovery program through MAC when recovery is completed.

The active RCC monitors the RPT of the active TMC and initiates further system recovery operations if the recovery programs fail to reset the RPT in the correct interval. The RPT has two checking modes. When first enabled by a maintenance interrupt, the recovery program must check into the RPT through the SPR exactly every 128th cycle.

The recovery program may change the checking mode to permit check-in before the 128th cycle. In the second mode, checkins may not be more than 128 CP cycles apart. The recovery program changes the checking mode or disables the RPT through MAC and must do it at exactly the 128th cycle.

Interrupt Control Circuit (ICC)

The ICC **28** (FIG. **9**) controls the execution of maintenance interrupts. A maintenance interrupt is a one-cycle wired transfer instruction which causes the CMC to begin execution of a recovery program. The malfunction detection circuits in the CP initiate maintenance interrupt whose execution takes precedence over the execution of any other CP instructions.

The ICC provides five maintenance interrupts:

1. System Recovery.
2. CP recovery,
3. IS recovery,
4. PS recovery, and
5. PU recovery.

When an interrupt occurs, the ICC products an ICCinterrupt Sequence Level (ICCSL) which controls the execution of the interrupt in the other CP circuits. The recovery program address corresponding to the interrupt is also placed on the INTerrupt Address Bus (IN-

TAB) to the Data Processing Circuit, from which it is sent to the IS.U0 as the address of the next instruction to be executed.

The Malfunction Monitor Circuit initiates the CP, IS, PS, and PU recovery interrupts. The Recovery Control Circuit or the Manual Control Console initiates the system recovery interrupt. An interrupt may be initiated by either circuit during the execution of an operational program when a malfunction occurs. During the execution of a recovery program additional interrupts may occur as a part of the recovery process.

To handle simultaneous interrupts and interrupts during execution of a recovery program, the ICC produces maintenance interrupts according to a priority structure. The system recovery interrupt has highest priority and cannot be inhibited. The CP, IS, PS, and PU interrupts follow respectively in descending order of priority. A CP, IS, PS, or PU interrupt can occur if the interrupt itself or a higher priority interrupt has not already occurred. CP, IS, PS, and PU interrupts may be individually inhibited by the maintenance programs.

Recovery Control Circuit (RCC)

The RCC **29** (shown in duplicate copy in FIG. **10**) monitors the malfunction detection circuits which cause system recovery program interrupts. The detection inputs to the RCC (RCC triggers) are produced by the timing generation check circuit in the TMC, error level from the DPC, the Recovery Program Timer in the TMC, a HELP instruction executed by the PCC, CP active unit change detected by the TGC, and a manual request from the MCC.

Only the active RCC accepts triggers and initiates system recovery action. The RCC in the Standby CP is kept in synchronism with the active RCC but cannot affect the operation of the CMC.

When a trigger to the active RCC occurs, the RCC executes a wired logic reconfiguration program and then requests the ICC to execute a system recovery program interrupt. If the system recovery program cannot be completed (i.e., the configuration is not operable), another trigger occurs. Each consecutive trigger causes the RCC to force one of the four combinations of CP*, and IS*.U0 configurations CP0-IS0.U0, CP1-IS0.U0, CP1-ISI.U0, and CP0-ISI.U0). When an operating CP*-IS*.U0 configuration is selected, the system recovery program completes the recovery and reconfiguration process without further intervention by the RCC.

Configuration Control Circuit (CCC)

The CCC **25** (FIG. **11**) defines the system configuration by controlling:

1. CP* status, and
2. The CP*-IS&, CP*-PS*, and CP*-PC* configurations.

The CP status is specified by:

1. The active CP indication,
2. The standby CP trouble status, and
3. The CP-CP error signal status (separated CPs or coupled CPs).

Each of the IS*, PS*, and PU*, has a bus system (address bus, data bus—the PS and IS share a data bus, and return bus). Each copy within IS*, PS*, and PU* is permanently associated with an individual bus copy. The CCC defines the CP*-IS*, CP*-PS*, and CP*-PC* configurations by specifying the bus copy on which each CP copy sends and receives.

The CCC first defines a primary bus copy for each of the IS, PS, and PC bus systems. The active CP always sends and receives on the primary bus. The standby CP sends and receives according to the specific bus configuration. For each primary bus copy selection, four bus configurations can be defined:

1. DUPLEX specifying that the standby CP sends on and receives from the non-primary bus copy,
2. SIMPLEX specifying that the standby CP receives from the primary bus copy while the non-primary bus copy is not used,
3. MERGED specifying that the active CP sends on both bus copies and both the standby and active CP's receive from both bus copies (i.e., the return buses are merged), and
4. SIMPLEX-UPDATE specifying that the active CP sends on both bus copies to update the secondary memory copies but the standby CP receives from the primary bus copy only.

The duplex bus configuration is used when both CP's and all units on both buses are in-service. The simplex configuration is used when a unit on the secondary bus is out of service. The merged configuration is used when units on both the primary and secondary buses are out-of-service. The update configuration is used while updating an in-service unit on the secondary bus.

A diagnostic bus configuration is also available for IS* which is used in the diagnosis and recovery of IS*.

Maintenance Access Circuit (MAC)

The MAC **30** (FIG. **12**) provides maintenance program access to the CP circuits. Read Maintenance Sense Group (RMSG) is an instruction which allows a group of 32 sense points from either the active or the standby CP to be read into a general register (GR1-GR2 of the Data Processor Circuit **23**, see FIG. **5**). Write Maintenance Control Group (WMCG) and Write Maintenance Control Point (WMCP) are instructions which respectively allow the program to write a group of 32 maintenance control points or a single control point in either the active CP, the standby CP, or both CPs. In this context, "writing" means that each maintenance control point sets or resets one or more flip-flops.

Although the instructions are decoded and controlled by the PCC, as explained more fully in the above-identified Brenski, et al., application Ser. No. 289,718, MAC selects the control groups, transmits write data from the DPC to the maintenance control groups selected, and reads maintenance sense groups returning data to the DPC.

Maintenance sense and control groups in either the active or standby CP are always selected by the MAC in the active CP only. Write data for maintenance control groups is also always taken only from the MAC in the active CP. In other words, only the MAC in the active CP can execute MAC instructions.

Power Monitor Circuit (PMC)

A Power Monitor and Control Circuit (PMC) (see FIG. 41) controls the actions necessary to turn power on or off from a CP or controls the actions necessary to remove power from a CP in which there is a defective power supply.

In case of trouble in a power supply of a CP copy, the PMC will remove all remaining power supplies from that copy.

When power is turned back onto the CP, the PMC will guarantee that the power can be turned on only to the standby CP while keeping the other CP active.

III. Recovery Control Circuit

The Recovery Control Circuit (RCC) **29** is the hardware part of a hardware-software emergency package. On indications that there are faults in the currently active CP [i.e. CP(A)] — primary IS combination, the RCC causes an interrupt into the System Recovery Program. If this program fails to pass certain checks, RCC forces a new CP-IS simplex combination and restarts the System Recovery Program. This is repeated until a CP(A), IS (primary) combination is found which does not exhibit errors or faults.

The RCC has to take action under failure conditions that are not covered by error signals from MMC **26.** We can list two broad classes:

A. Failure conditions that affect equally both CMC copies

Examples:

1. Faults in TGC of CP(A). These may affect both CP's.
2. Program faults that cause CP's to lose real-time control.

B. Failure conditions in the CP(A)-IS combination during execution of a recovery program. These programs are always entered with CP(S) i.e., the Standby Central Processor stopped. Their execution thus depends on fault-free operation of CP(A) and its primary IS, and has to be monitored by RCC.

The failure signals feeding into RCC are described below.

Before getting into the specific details of the RCC, a broader summary of its structure and operation will be made.

Duplication: There is one RCC in each CP and these are designated respectively **29** and **29***a* in FIG. **13.**

States: An RCC can be in one of four states called **S0, S1, S2,** and **S3.**

Activity: An RCC is called active when it is executing an RCC-cycle; otherwise it is called dormant. Only the RCC in CP(A) can become active (initiate a RCC-cycle) in response to error signals. The RCC which initiates an RCC cycle remains in control during the entire cycle.

RCC Cycle: During an RCC Cycle, the active RCC:

1. assumes a new state; commands the other RCC to assume the same state,
2. sets up new CP(A)-IS combinations, if required,
3. generates and transmits a signal called RCEL to ICC (to start System Recovery Program), if required,
4. always starts recovery program timer **75** in the Timing Monitor Circuit **27** (FIG. **8**), and
5. allows some 10 μ for restoration of CP error indicators before returning to dormancy.

When a CP status change takes place during an RCC cycle, it is executed before the end of this cycle. Thus, the active CP does not necessarily contain the active RCC.

Timing: Each RCC contains an internal timing generator that produces the various timing pulses in an RCC cycle (RCC is independent of TGC **21**).

Each of the RCC's **29, 29***a* of FIG. **2** includes: A control circuit designated **76** and **76***a* respectively; a timing generator circuit designated **77** and **77***a* respectively; and a state control circuit designated **78** and **78***a* respectively.

These inputs to RCC, as seen in FIG. **13,** can be divided into three groups:

Primary Error Inputs

These are indications of malfunctions that require a transfer to the System Recovery Program and may require CP-IS reconfiguration. Primary error inputs are not cross-coupled between CP's.

1. MANL—Manual Level. Generated at Manual Control Console **20** (MCC) by maintenance man; causes an RCC cycle.
2. HELPL—Help Level. Generated by PCC **22** when a HELP instruction is decoded as indicated in the above-referenced Brenski, et al. application, Ser. No. 289,718. This provides a means to enter System Recovery Program under software control.
3. RREL — Repetition Rate Error Level. Generated by Timing Check Circuit **73** of TMC **27** (see FIG. **5**) when timing levels are not received at proper rate.
4. TLEL — Timing Level Error Level. Generated by Timing Check Circuit **73** of TMC **27** when one or more timing levels are not received properly.
5. RTTEL — Real-Time Timer Error Level. Generated by DPC when Real-Time Timer (which is a counter) exceeds its maximum count because the program did not reset it in time. This is an indication that program has lost real-time control. The Real Time Timer is a part of the Special Register (SPR) included in the Storage Section **56** of the DPC (FIG. **5**), as disclosed more fully in the referenced Brenski, et al. application, Ser. No. 289,718, the disclosure of which is also incorporated herein by reference.
6. RPTEL — Recovery Program Timer Error Level. Generated by Recovery Program Timer **75** of TMC when Recovery Program Timer is not punched properly by a recovery program.
7. CPACL — Central Processor Activity Change Level. Generated by TGC whenever status of CP changes. This change could of course have been provoked by RCC, (CPACL is ignored in this case) but the circuit considers the case that the change is caused elsewhere. In that case RCC has to initiate System Recovery Program.

Secondary Error Inputs

These inputs never cause the RCC to enter the System Recovery Program or to change the CP-IS configuration. Their purpose is to alert RCC that an interrupt to a recovery program is taking place.

1. ICCSL.I — Interrupt Control Circuit Sequence Level (see FIG. **9**). Generated by ICC in the CP that houses RCC, whenever interrupt (caused by

MMC **26**, MCC **20**, or RCC **29** itself) takes place.

2. ICCSL.X — Interrupt Control Circuit Sequence Level External. This signal is cross coupled from the other CP.

Control Inputs

The following control inputs affect RCC response to the primary and secondary error inputs described above.

1. CPAL — Central Processor ACtivity Level. Generated by CCC **25** of FIG. **11**. CPAL = 1 indicates that the "home" CP is active (only RCC in CP(A) may initiate RCC cycle).
2. RPTAL — Recovery Program Timer Activity Level. Generated by Recovery Program Timer **75** of TMC when Recovery Program Timer **75** (FIG. **8**) is timing a recovery program.
3. RCLOL — RCC Lockout Level. Generated by MCC **20** when a maintenance man disables RCC.
4. RCAL.X — RCC Activity Level External. Generated by RCC in other CP. RCAL = 1 when a RCC is active; it inhibits other RCC from becoming active.
5. CPSPF — CP Separate FF. Generated by CCC. When CPSPF = 1 the "home" CP is to ignore error indicators from other CP.

RCC STATE SEQUENCES AND ACTIONS

Starting Conditions for RCC Cycle

An RCC can only become active in response to an error signal if all conditions below are fulfilled.

a. It is located in active CP (indication: CPAL = 1).

b. The maintenance man is not preempting RCC actions (indication: RCLOL - 0).

c. The RCC in either CP is not active (indication: RCAL.X = RCAL.I = 0).

We now define a primary RCC starting level RCPSL (caused by primary error inputs) and an RCC secondary starting level RCSSL (caused by secondary error inputs). Either level starts an RCC cycle, and they are stored in corresponding flip flops in the control circuitry **76**, namely RCPSF and RCSSF (not shown).

$$RCPSL = \overline{RCAL} \cdot CPAL \cdot \overline{RCLOL} \cdot \overline{RCAL.X} \cdot (MANL \vee HELPL \vee RREL \vee TLEL \vee RTTEL \vee RPTEL \vee CPACL).$$

$$RCSSL = \overline{RCAL} \cdot CPAL \cdot \overline{RCLOL} \cdot \overline{RCAL.X} \cdot (ICCSL.I \vee \overline{CPSPF} \cdot ICCSL.X) \cdot S0$$

As discussed below **S0** is a signal indication that the RCC is in state **0**. An RCC cycle is started by triggering the timing generator circuit **77** into a timing cycle. It will be observed that both primary and secondary start levels are dependent on CPAL which is the signal indicative of which CP is active. Hence, only the RCC in the active CP can be triggered into an RCC cycle.

State Sequences

FIG. **14** shows the four RCC states (**S0, S1, S2, S3**). Possible transitions between states are shown by arrows. The condition for a transition is shown above the arrow and if this transition takes place during an RCC cycle the most important RCC actions are shown below the arrow.

S0 is the reset or dormant state of an RCC. Note that transitions into **S0** do not involve an RCC cycle. RCC is reset by program control. MAC produces the RCC reset level RCRSL. More details can be obtained from the copending, co-owned application of Schulte, et al. for "Maintenance Access Circuit for Central Processor of Digital Communication System", filed Jan. 2,1972, Ser. No. 320,020 Note also that the reset is executed only when the Recovery Program Timer is reset (RPTAL = **0**). This is to prevent accidental resetting of RCC during execution of recovery programs monitored by the RPT. Normally all other transitions involve execution of an RCC cycle. (Note these transitions can also be caused under software control by MAC, in which case no RCC cycle is started).

Transitions from **S0** to **S1** are caused by primary and secondary start levels. In both cases a minor alarm indicator at MCC **20** is set. Only primary start levels cause entrance into System Recovery Program (SRP). The secondary start level merely notifies RCC that a lower-priority Recovery Program is being started.

Transitions into **S2** and **S3** are caused by primary start level only. These transitions always cause entrance to SRP. In addition, they cause the following CP-IS reconfigurations.

S1 to **S2**: CP status switched provided that CP(S) does not have its Trouble Flip Flop CPTBF set.

S2 to **S3**: **IS1** is forced to become primary IS.

S3 to **S2**: CP status switched, regardless CPTBF state of CP(S). **IS0** is forced to become primary IS.

During this RCC cycle a major alarm is started at MCC.

The reconfiguration rules have been set up to minimize the number of RCC cycles/SRP attempts until a working CP-IS combination is found. It turns out that at most six attempts are necessary. As an example, assume:

CP0 is active CP

IS1 is primary IS.

A primary malfunction indication occurs because of simultaneously detected malfunctions in both **CP0** and **IS1**.

Attempt	RCC State Change	Active CP	Primary IS
1	S0 → S1	CP0	IS1
2	S1 → S2	CP1	IS1
3	S2 → S3	CP1	IS1
4	S3 → S2	CP0	IS0
5	S2 → S3	CP0	IS1
6	S3 → S2	CP1	IS0

The number of RCC cycles required to reconfigure the CP* - IS* into a working system will be minimal if **IS0** is defined to be the primary copy whenever possible.

Returning to the RCC block diagram of FIG. **13**, all interconnections between the two RCC's **29** and **29*a*** and between the RCC's and other circuits are shown. Each RCC includes:

a. a control circuit, **76**, **76*a***

b. a timing generator, **77**, **77*a***, and

c. a state control circuit, **78**, **78*a*** respectively.

A Recovery Control Bus (RCB) generally denoted **80** is used to transmit AC signals from the active RCC to circuits in both CP(CN) and IS*.U0. DC levels are used for signals which do not require cross coupling between CP's or have only one destination.

The control circuit **76** or **76*a*** initiates a cycle by activating the timing generator **77** or **77*a*** via either the primary (RCPSL) or further ECC actions (except the En-

able Recovery Program Timer Level ERPTL, as will be described presently) are purely a function of RCC timing pulses and RCC state. The Timing Generator has control circuitry for determining the sequence and length of output signals. The RCC state defines which output signals are required to perform the reconfiguration.

The timing generator circuits **77, 77***a* may be constructed according to anyone of a number of conventional techniques. In one embodiment it includes a delay line which is pulsed at the beginning of an RCC cycle. The timing levels are picked off the appropriate distances along the line to generate delayed pulses; and the width of these pulses may be expanded or contracted with monostable circuits. See the above referenced Brenski, et al. application, Ser. No. 289,718, particularly FIG. **44** and accompanying text.

RCC Cycle Time

An RCC cycle is 10.0 to 10.5 μ sec. long, as seen in FIG. **15.** This is sufficient to allow all malfunction detection circuits to reset in the active central processor following completion of reconfiguration actions. The longest reconfiguration sequency occurs as a result of a central processor switch. The cycle must be long enough to allow the repetition rate timer in the Timing Check Circuit **73** of TMC to reset after the active central processor starts executing the interrupt. The total RCC cycle time is the sum of:

a. 2.0 μ sec.; the time required by the RCC from the start of the cycle to completion of the CP switch function,

b. 6.0 μ sec.; the time required by the Timing Generator Circuit (TGC) to start the active central processor, and

c. 2.0 μ sec.; the time required, after the CP is started, to initialize the repetition rate timer.

Minimum RCC cycle time = 2.0 μ sec. + 6.0 μ sec. + 2.0 μ sec. = 10 μ sec.

Timing Within RCC Cycle

Within each cycle 4 sequencial stages of action occur in the RCC:

a. generation of appropriate configuration control signals as a function of existing RCC state,

b. generation of request for SRP interrupt if required,

c. advancing RCC to next state, and

d. guard period to allow CP malfunction detection circuits time to reset. Sequencing of these actions is accomplished by 5 timing pulses generated by the RCC clock. FIG. **15** shows the 5 pulses (RT1L through RT4L and RCAL), the length of each, and their placement in the RCC cycle.

RT1L and RT2L — used to generate configuration the control signals indicated in the drawing.

RT3L — used to generate request for SRP interrupt.

RT4L — used to generate advance state signals.

RCAL — RCC Activity Level.

FIG. **16** shows the RCC output signals required for each state transition initiated by an RCC cycle. The purpose and characteristics of each is given below:

ERPTL: Enable Recovery Program Timer

ERPTL = MANL + HELPL + RREL + TLEL + RTTEL + RPTEL + CPACL + ICCSL.I + ICCSL.X·$\overline{\text{CPSPF}}$

This signal is generated whenever any one or more of the input error indicators is true.

Coupling: DC to TMC of home CP(CN).

Signal Length: Same as input signal (not less than 500 ns.)

RCAL: RCC Activity Level; RCAL in one CP becomes RCAL.X in the other CP.

This signal is true whenever the RCC is active; i.e., a cycle is in process.

Coupling: DC to opposite CP(CN).

Signal Length: Same as RCC cycle time (minimum of 10 μ sec., see FIG. **15**)

Function: Disable RCC in other CP(CN) during RCC cycle.

RCEL: Recovery Control Error Level

RCEL = RT3L·RCPSF where RCPSF is true whenever an RCC cycle is initiated by RCPSL as defined above.

Function: Upon receiving RCEL the ICC will generate an interrupt to SRP.

Coupling: AC to CP*.ICC via RCB.B02 (to ICC's in both CP's).

Signal Length: 1μ.

Placement: RCEL cannot be generated until all configuration commands have been issued and have had time to stabilize. Therefore, RCEL is generated 2.0 μ sec. after start of cycle (see CPSWL below).

CPSWL: CP Switch Level

CPSWL = RT2L·(S1 v S3).

Function: The active CP will become standby and standby will become active if the standby CP trouble FF is not set.

Coupling: DC to CCC of home CP.

Signal Length: 1 μ minimum.

Note: The minimum signal length is a function of the number of gates in series in each leg of the CPAUF which is disclosed below. The reconfiguration time is twice CPAUF switch time (2μ). This allows for relapse in case the CP trouble FF is set in the standby.

Placement: CPSWL is generated at start of cycle.

CPTL: CP Trouble Level

CPTL = RT1L·S3.

Function: Set the CP trouble FF in home CP, reset CP trouble FF in other CP, and reset Instruction Store Copy Bus FF in both CP(CN).

Coupling: AC to CP*.CCC.

Signal Length: 500 μ sec.

Placement: At start of cycle.

SICBL: Set Instruction Store Copy Bus Level SICBL = RT1L·S2.

Function: Set the Instruction Store Copy Bus FF in CP*.CCC.

Coupling: AC to CP*.CCC via RCB.B03.

Signal Length: 500μ sec.

Placement: At start of cycle.

RICCL: RCC Instruction Store Configuration Control Level

RICCL = RT1L·(S2 v S3)

Function: Reset ISBBF, set ISTBF and reset ISDBF instruction store bus configuration FF's. Also resets TBF in both IS(CN).U0

Coupling: AC to CP*.CCC and IS*.U0 via RCB.B04.

Signal Length: 500 μ sec.

Placement: At start of cycle.

MIALL: Minor Alarm Level

MIALL = RT1L·S0.

Function: Set minor alarm FF in manual control console.

Coupling: AC to MCC via RCB.B05.
Signal Length: 500 μ sec.
Placement: At start of cycle
MAALL: Major Alarm Level
MAALL = RT1L·S3.
Function: Set major alarm FF for home CP in manual control console.
Coupling: DC to MCC.
Signal Length: 500 μ sec.
Placement: At start of cycle.
ASEL: Advance State Even Level
ASEL = RT4L·(S0 v S2).
Function: Cause state counter to advance in both RCC's from S0 or S2 to S1 or S3 respectively.
Coupling: AC to CP*.RCC via RCB.B00.
Signal Length: 500 μ sec.
Placement: Any time after RT3L.
ASOL: Advance State Odd Level (see 5.3)
ASOL = RT4L·(S1 v S3).
Function: Cause state counter to advance in both RCC's from S1 or S3 to S2.
Coupling: AC to CP*.RCC via RCB.B01.
Signal Length: 500 μ sec.
Placement: Any time after RT3L.

In the above, S0, S1, S2 and S3 are signals representative of the RCC's being in those respective states. The generation of the signals is shown in FIG. 17 and will be better understood presently.

STATE CONTROL

The state control circuit 78, 78*a* is shown in FIG. 17. FIG. 18 shows the state control timing and the state transition table. The primary functions of the state control circuit are to determine which states both RCC's 29 and 29*a* should be in, and to indicate which state they are in fact in.

Before turning to the detailed showing of FIG. 17, FIG. 13 shows six bits, B00-B05 of the Recovery Control Bus (RCB) 80, and these are designated respectively 80*a*–80*f*. Lines 80*a*, 80*b* representing respectively bits B00 and B01 communicate the state control circuits 78, 78*a* in the two copies of the RCC. Bit B00 carries the ASEL signal, and bit B01 carries the ASOL signal. ASEL stands for Advance State Even Level, and ASOL stands for Advance State Odd Level. These signals are AC pulses and they are cross coupled to both RCC copies as indicated. Only the RCC in the active CP may initiate an RCC cycle; therefore only one RCC State Control Circuit 78, 78*a* will receive timing level RT4L which is necessary to produce the signals ASEL and ASOL, as will be explained presently. Recovery Control Bus bits 03, 04 and 05 are coupled between both RCC copies and carry the following signals respectively: SICBL, RICCL, and MIALL. It will be observed that RCB, B04 (RICCL) is fed directly to the Instruction Stores to set the TDF and TBF flip-flops, as will be discussed.

Turning then to FIG. 17, the state control circuit 29*a* is similar to the state control circuit 29, and only the latter need be described in detail in order to understand the invention completely. The state control circuit 29 includes a Reconfiguration State Control Flip-Flop C (denoted 85 and sometimes referred to as RCSCF; a Reconfiguration State Control Flip-Flop A (denoted 86 and sometimes referred to as RCSAF); and a Reconfiguration State Control Flip-Flop B (denoted 87 and sometimes referred to as RCSBF).

In the drawing, the circuit indicated by the symbol 87 is a cable receiver, the circuit indicated by the symbol 88 is an OR gate; the circuit indicated by the symbol 89 is an AND gate; and the circuit indicated by the symbol 90 is a cable driver (both inputs of which must be pulsed in order to obtain an output). As indicated, the signals ASEL and ASOL are generated respectively on the lines 80a and 80b at the time RT4L dependent upon whether the CP is in an even state (S0 or S2) or an odd state (S1 or S3).

The flip-flops 86, 87 indicate which state the RCC is in, and the AND gates responsive to the outputs of these flip-flops generate the levels S0-S3, as illustrated.

The Reconfiguration State Control Flip-Flop C 85 in both RCC copies responds to the signals ASEL and ASOL—ASEL causes the flip-flop 85 to be set when RT4L is pulsed, and ASOL causes it to be reset. When RT4L ends, the state indicating flip-flops 86, 87 will assume the next sequential state (see timing diagram and chart of FIG. 18). That is, these latter flip-flops are arranged as a counter, counting sequentially through the various states that the RCC may assume, as disclosed above.

It is an important feature of the circuitry shown in FIG. 17 that an inactive RCC, if activated, will eventually achieve synchronous operation with and be controlled by the RCC in the active CP. This is due to the logic circuitry shown and the separation of control pulses into the even and odd levels, as described. For example, if the inactive RCC is in S2 and the active is in S1 or S3, the inactive RCC will not even respond to RT4L since an RSEL level will be generated, and the RCSCF is already in the "ONE" state. The input logic circuitry to flip-flops 86 and 87 will prevent their response until synchronization is eventually attained.

All three flip-flops in the state control circuit can be software controlled via the Maintenance Access Circuit (MAC). The RCSCF can be set (signal RCSCFS), reset (signal RCSCFR), and sensed via MAC (level RCSR). Note, however, that the MAC set and reset commands will have no effect if the recovery program timer 75 of the TMC 27 (FIG. 8) is active ($\overline{\text{RPTAL}}$=1) since the AND gate 98 is enabled only by $\overline{\text{RPTAL}}$. In other words, if a recovery program is being run, MAC cannot reset the RCC to State 0. The location of the MAC control and sense points associated with the RCC are shown in FIGS. 19 and 20.

Because an RCC cycle only occurs as a result of a malfunction, no definite statements can be made about the CP status at the start of the recovery program. A few general comments follow:

- CP general registers; the contents are not directly affected by RCC, or ICC.
- The standby CP is stopped (by the ICC).
- All error cross coupling is disabled (by CCC and MMC).
- No bus configurations are modified except where explicitly defined by RCC state transition.
- The address + 1 or address + 2 of the instruction being executed at time of malfunction or RCC cycle is stored in match register 0 (by ICC and MMC) so that the program can return.

Only the RCC state control circuit 78, 78*a* can be accessed directly by program. Access is provided via the Maintenance Access Circuit (MAC). The following maintenance access control points are provided.

a. RCSCFS: Reconfiguration State Control Flip-Flop Set

This control point will set the RCSCF to "1". If the recovery program timer is active, the control point will have no effect on RCSCF (see gate **89**).

b. RCSCFR: Reconfiguration State Control Flip-Flop Reset

This control point will reset the RCSCF to "0" via AND gate **94** (also receiving $\overline{\text{RPTAL}}$) and OR gate **95.**

c. RCSR: RCC State Reset

This control point will reset state control section flip-flops RCSAF and RCSBF along with the RCC Primary Start Flip-flop (RCPSF) which is included in the control circuit **76** and generates the RCPSL start signal.

FIG. **19** shows the RCC control point layout in Maintenance Control Group **—8.** If the recovery program timer is active (RPTAL=1), all of the above control points are disabled.

Four maintenance access sense points are provided.

a. RCSAF: RCC State A Flip-Flop
b. RCSBF: RCC State B Flip-Flop
c. RCSCF: RCC State Control Flip-Flop
d. RCPSF: RCC Primary Start FF.

The layout of these points are shown in FIG. **20.**

Before returning to call processing, the recovery programs must reset the RCC to its normal state **S0.** First RCSCF must be reset via MAC control point RCSCFR (MCG-**8**,**B02**). Following this, all other RCC FF's can be reset via MAC control point RCSR (MCG-**8.B00**).

The reset function of both RCSCF and RCSR is disabled if the recovery program timer is not reset.

The above two part sequence will cause the RCC state counter to go to **S0** from all other RCC states **S1, S2,** and **S3.**

The execution of a HELP instruction (see FIG. **81** of above-identified Brenski, et al. application Ser. No. 289,718) causes the RCC to immediately go through a reconfiguration cycle whenever the RCC is in states **S1, S2** or **S3.** This feature is provided so that an RCC trigger can be generated by the program whenever it leaves the normal path in a recovery program. For example, if a recovery program contains a decision transfer instruction where the outcome of the decision should always be true, the transfer should be followed by a HELP instruction. Thus, if the CP fails to make the transfer a RCC trigger is generated.

Program Monitoring

Program Monitoring During Normal Operation

As mentioned above and more fully disclosed in the Brenski, et al. application Ser. No. 289,718 incorporated herein by reference, the Data Processing Circuit **23** (FIG. **5**) contains the Real-Time Timer as a part of the Special Register SPR which is a continuously running counter. The counter is incremented once during each CP cycle time (4μ sec.). The programs must continually reset the counter such that the count never exceeds 16383 cycles. If the count is allowed to reach 16384 an RCC trigger (RTTEL) is generated and a reconfiguration cycle will result.

Program Monitoring During Recovery

In addition to the Real-Time Timer, the Recovery Program Timer **75** of TMC (FIG. **8**) is also employed during the execution of the recovery program. This timer requires that the recovery program "punch in" every 128 instructions. If the program punches in either early or late, an RCC trigger (RPTEL) is generated. The Recovery Program Timer also has a special mode of operation which allows operation similar to that of the Real-Time Timer; the program must "punch in" such that the count never exceeds 127 cycles.

Input Summary

SOURCE	MNE-MONIC	NAME
TMC	RREL	Repetition Rate Error Level
		The Repetition Rate Error Level becomes true if a selected timing level in the TGC circuit does not occur every 4 μ seconds.
		Signal Characteristics:
		Coupling: DC Coupled.
		Signal Duration: Until timing level starts again.
TMC	TLEL	Timing Level Error Level
		This signal to RCC is true if any timing level is missing or occurs in the wrong time.
		Signal Characteristics:
		Coupling: DC Coupled.
		Signal Duration: 500 nanoseconds.
DPC	RTTEL	Real Time Timer Error Level
		The Real Time Timer counts the number of Basic Order Cycles (BOT) executed by the Processor (A BOT is 4 microseconds). The program timer will provide a true level when the count reaches 16284, approximately 65 milliseconds, unless it is reset anytime prior to this by the program. It's function as an RCC input is to keep a gross check on program execution. It makes the RCC take action if the program should get lost or enter endless loops, and thus lose its call processing ability.
		Signal Characteristics:
		Coupling: DC Coupled.
		Signal Duration: 3.5 – 4.0 μ seconds.
TMC	RPTEL	Recovery Program Timer Error Level
		The Recovery Program Timer counts BOT's and will provide a true level when the count reaches binary 128 or if the timer is reset at anytime other than when the count is binary 127, unless the Recovery Timer is in its special mode.
		Its function as an RCC input is to keep a stringent running time check on all fault recovery programs. It makes the RCC take action if fault recovery programs run amuck due to a fault.
		Signal Characteristics:
		Coupling: DC Coupled
		Signal Duration: Until reset by ICC.
TGC	CPACL	Central Processor Activity Change Level
		The CPACL level is true whenever the central processor's change their active standby status. Its principal function as an input to the RCC is to cause the RCC to generate a interrupt command to the ICC circuit whenever the CP's switch.
		Signal Characteristics:
		Coupling: DC Coupled
		Signal Duration: Approximately 1.5 μ seconds
PCC	HELPL	Help Instruction Level
		A true input is provided whenever the instruction decoder decodes a HELP instruction in the program. Its function as an input to the RCC is to provide a means for the program to initiate an RCC cycle.
		Signal Characteristics:
		Coupling: DC Coupled
		Signal Duration: Approximately 3.5 μ seconds.
MCC,MSCP	MANL	Manual Interrupt Level
		A true input is provided whenever the button requesting an RCC Cycle from the maintenance console is depressed.
		Its function as an RCC input is to allow the maintenance man to initiate RCC action when it is required.
		Signal Characteristics:

-Continued

Input Summary

SOURCE	MNE-MONIC	NAME
		Coupling: AC Coupled. Signal Duration: 1 μ second.
ICC	ICCSL.I	Interrupt Control Circuit Sequence Level Interval. The ICCSL.I level is generated whenever the interrupt circuit in same CP(CN) generates an interrupt of any priority. Its function as an RCC input is to inform the RCC that ICC is taking action and that the RCC should start the recovery program timer and advance its self to state S1 but take no other action. ICCSL.I is effective in the RCC only if RCC is in S0 the reset state. Signal Characteristics: Coupling: DC Coupled. Signal Duration: Approximately 3 μ seconds.
ICC	ICCSL.X	Interrupt Control Circuit Sequence Level. This is the same as ICCSL.I except it is generated by the opposite CP(CN).
CCC	CPAL	Central Processor Activity Level. This level when true signifies that the RCC in which it is connected is the active RCC.
RCC.X	RCAL.X	RCC Activity Level External. This level is generated by the external RCC circuit when it is active and performing a RCC cycle. Its function in the internal RCC circuit is to prevent the internal RCC from accepting principle inputs during the transition of the internal RCC from a standby to active status, during an external RCC initiated CP switch. This level is necessary to allow the initiating RCC to complete all RCC actions. RCAL eliminates timing synchronism problems between RCC's and simplifies gating.
RCB.B00	ASEL	Advance State Even Level. This level is generated by the active RCC to step the state counter in both RCC's. Its function as an RCC input is to keep the states of the RCC's in synchronism.
RCB.B01	ASOL	Advance State Odd Level. Same as ASEL.
IMDB.nn		Internal Maintenance Data Bus
IMDB.B00	RCSR	RCC State Reset Command. This level is a general reset signal to the RCC. Signal Characteristics: Coupling: DC Coupled. Signal Duration: 500 μ seconds.
IMDB.B01	RCSCFS	These levels allow the program
IMDB.B02	RCSCFR	to advance the RCC state counter.
IMDB.B03	RCTGDS	Start SBY RCC delay line; for diagnostic use only.
MCC.MCSP	RCLOL	RCC Lockout Level. This level is generated by the maintenance control console and allows the maintenance man to disable all RCC action when he wishes to select a configuration manually. Signal Characteristics: Coupling: DC Coupled. Signal Duration: Duration switch generated.
CCC	CPSPF	Central Processor Separate Flip-Flop. The level inhibits all inter CP error level communication. Its purpose in the RC is to inhibit reception of ICCSL.X when CPSPF is true. Signal Characteristics: Coupling: DC Coupled
TMC	RPTAF	Recovery Program Timer Active Flip-Flop. This level from the TMC indicates that the cycle timer is running. Its function in the RCC is to prevent the RCC from being reset to the idle state and to prevent MAC routining of the State Counter when the timer is running. Signal Characteristics: Coupling: DC Coupled. Signal Duration: Approximate duration as long as timer is running.

Output Summary

DESTINATION	MNE-MONIC	NAME
CP*.RCC	RCAL	RCC Activity Level. This level is generated when the active RCC enters its cycle and remains up until it finishes its cycle. It serves to keep the other RCC deactivated until the cycle is complete and this guards against the other RCC taking over in the event of a CP switch caused by the active RCC. Signal Characteristics: Coupling: DC Coupled. Signal Duration: Approximately 10 μ seconds.
RCB.B00	ASEL	Advance State Even Level. This level is generated by the RCC during its cycle to advance the RCC state counters to their next state. The active RCC advances the state of both its self and the standby RCC. Signal Characteristics Coupling: AC Coupled Signal Duration: 500 nanoseconds.
RCB.B01	ASOL	Advance State Odd Level. Same as ASEL.
TMC	ERPTL	Enable Recovery Program Timer Level. This signal is generated by the RCC for any principle input being true irrespective of whether the RCC is the active RCC or the standby RCC. ERPTL is coupled only to the internal TMC. Signal Characteristics: Coupling: DC Coupled Signal Duration: This signal is generated directly from the principle inputs and as such depends on the length of the input. The minimum length is 500 nanoseconds the maximum is undefined.
RCB.B02	RCEL	Recovery Control Error Level. This level to the ICC circuit is generated by the RCC when a interrupt to System Recovery Program is required. Signal Characteristics: Coupling: AC Coupled. Signal Duration: 1 μ second.
RCB.B03	SICBL	Set Instruction Store Copy Bus Level. This level selects Instruction Store Bus 1 to be the primary bus. SIBL is coupled to the CCC circuits in both the active and the standby CP's via RCB Reconfiguration Bus. Signal Characteristics: Coupling: AC Coupled. Signal Duration: 500 nanoseconds
CP*.CCC	CPTL	Central Processor Trouble Level. This level sets the CP trouble flip-flop in same CP(CN) as the RCC, resets the CP trouble F/F in opposite CP(CN), and selects Instruction Store Bus 0 to be primary bus. Signal Characteristics: Coupling: AC Coupled. Signal Duration: 500 nanoseconds.
RCB.B04	RICCL	RCC Instruction Store Configuration Control Level. This level resets the trouble flip-flops in IS*.U0, resets the ISBBF in both CP(CN), sets the ISTBF in both CP(CN) and resets the ISDBF in both CP(CN). Signal Characteristics: Coupling: AC Coupled. Signal Duration: 500 nanoseconds.
CCC	CPSWL	Central Processor SWitch Level. This level causes the active and standby Central Processors to switch if the standby Central Processor is not marked in trouble. CPSWL is coupled only to the internal CC.

-Continued

Output Summary DESTINATION	MNE-MONIC	NAME
	Signal Characteristics: Coupling: DC Coupled. Signal Duration: 1 μ second.	
RCB.B05	MIALL	MInor ALarm Level
		This level is provided to the CAM whenever the RCC advances out of the reset state.
	Signal Characteristics: Coupling: AC Coupled. Signal Duration: 500 nanoseconds.	
IMRB.B01	RCSAF	RCC State A Flip-Flop
IMRB.B02	RCSBF	RCC State B Flip-Flop
IMRB.B00	RCSCF	RCC State Control Flip-Flop
IMRB.B03	RCPSF	RCC Primary Start Flip-Flop
09MCC.MCP(CN) CAM		
		These flip-flops are sense points to the Central Processor to enable it to view the states of the RCC State Sequence Counter and the Primary Start Flip-Flop of the RCC.
	Signal Characteristics: Coupling: DC Coupled. Signal Duration:	The State Sequence Counter will provide a steady level unless it is viewed during a RCC cycle.
MCC.MCP(CN)	RCAL.X	RCC Activity Level External
		The RCAL.X signal that is received from other RCC is wired to a lamp on MCC.
MCC.MCP(CN)	RCLOL	RCC Lockout Level
		The RCLOL signal that is received from the MCC is returned to the MCC and displayed in a lamp.

IV. Configuration Control Circuit

The Configuration Control Circuit (CCC) (blocks **25** in FIGS. **1** & **2** and seen in FIG. **11**) contains 13 flip-flops which statically define the configuration that exists in the Control and Maintenance Complex (CMC) communication channels. These 13 flip-flops are separated into four groups with respect to function they perform. The four functions are to:

a. Indicate the status of the Central Processors (CP's) and cross coupling.

b. Indicate the bus configuration between the CP's and the Instruction Store Complex (IS*).

c. Indicate the bus configuration between the CP's and the Process Store Complex (PS*).

d. Indicate the bus configuration between the CP's and the Peripheral Unit Complex (PU*).

CCC flip-flops can be altered by one (and for certain flip-flops by all) of the following three ways:

1. Program Control (via Maintenance Access Circuit "MAC").

2. Hardware Control (via Recovery Control Circuit "RCC").

3. Maintenance Personnel Control (via Manual Control Console "MCC").

The nomenclature used is the same as used in the above-identified Brenski, et al. application Ser. No. 28-9,718. The CCC will be disclosed herein only to the extent necessary to understand how it generates signals to the IOC to set up the various IS, CP combinations called for by the previously described RCC.

The Configuration Control Circuit is organized into four sections: The CP Status Group, the CP-IS Bus Configuration Group, the CP-PS Bus Configuration Group, and the CP-PU Bus Configuration Group.

CP Status Group

The Central Processor Status Group (see FIG. **21**) contains four flip-flops and a CP copy identification level:

1. Central Processor Active Unit Flip-Flop **100** (CPAUF)

This flip-flop specifies whether the CP in which it resides is the active or the standby copy. The CPAUF is actually implemented as one flip-flop with one half in each CP copy, as will be described in connection with FIG. **41**. This insures that a standby-standby condition will not exist between the two CP copies. Also if power is removed from the active copy the standby CP automatically becomes the active copy.

2. Central Processor Trouble Flip-Flop **101** (CPTBF)

The CPTBF when set, indicates that the CP copy in which it resides is marked in trouble. The functions of this flip-flop are to inhibit the CP from sending to the IS, PS and PU complexes and to cause the other CP to become the active unit.

3. Central Processor Separate Flip-Flop **102** (CPSPF)

When this flip-flop is set: (a) it inhibits the Malfunction Monitor Circuit (MMC) error levels from being sent to the other CP: (b) inhibits the reception of the error signals generated and transmitted by the other CP; and (c) inhibits reception of ICCSL.X at the RCC. When reset, it provides a reset command to the Diagnostic Flip-Flop **103** (DF).

4. Diagnostic Flip-Flop **103** (DF)

This flip-flop provides the means for diagnosing the send and receive bus level circuitry within the CC of the standby CP which is developed, in part, with the function "DCPAL". Without the DF flip-flop the function "DCPAL" could never be "true" in the standby CP thereby making it impossible to diagnose standby CP circuitry composed with the "DCPAL" function.

The DF may be set (via MAC) in either CP only if the CPSPF flip-flop in the respective CP is set, since the CPSPF, when reset provides a constant reset command to the DF. DF's becoming set in the active CP has no effect.

5. Central Processor Copy Level **104** (CPCL)

The program normally refers to a CP copy as active or standby but during diagnostics the copy (**0** or **1**) identity of the active unit is required in order to notify the maintenance personnel as to which CP copy is faulty.

CP-IS Bus Configuration Group

CP-IS Bus Configuration Group, generally indicated by the block **105** in FIG. **21** contains four flip-flops which indicate the 24 possible CP-IS bus configurations (12 with **CP0** active, 12 with **CP1** active). The instruction Store Bus Control Flip-Flops in both CP copies are normally in identical set or reset states and should be altered by their set or reset commands through MAC simultaneously. The four flip-flops are:

1. Instruction Store Copy Bus Flip-Flop (ISCBF)

When ISCBF = 1;

defines **IS1**.Bus System (**IS1**.BS) as the primary bus which the active CP is to communicate on.

When ISCBF = 0:

defines IS0.BS as the primary bus which the active CP is to communicate on.

2. Instruction Store Both Buses Flip-Flop (ISBBF)
When ISBFF = 1:
a. active CP sends on both IS buses.
b. both CP's receive from both IS buses (except when ISTBF = 1, see 3).

3. Instruction Store Trouble Bus Flip-flop (ISTBF)
When ISTBF = 1:
both CP's receive from primary IS bus only.

4. Instruction Store Data Bus Flip-Flop (ISDBF)
ISDBF only has significance when ISTBF = 1 and only effects data being sent to the IS* during the first cycle of write instructions and data being received from IS* during the first cycle of read instructions. ISDBF provides a useful tool for diagnostics of a standby IS unit.
When ISDBF = 1:
a. both CP's receive data from secondary bus.
b. active CP sends data on secondary bus.
When ISDBF = 0:
a. both CP's receive data on primary bus.
b. active CP sends data on primary bus.

SOURCE	MNEMONIC	DESCRIPTION
RCC	CPSWL	CP Switch Level (Reset CPAUF)
	CPTL.I	CP Trouble Level Internal (Set CPTBF and reset ISCBF)
RCC.X	CPTL.X	CP Trouble Level External (Reset CPTBF and ISCBF)
RCB	RCB.B03 (SICBL)	RCC Bus, bit 03 (Set ISCBF (Level)
	RCB.B04 (RICCL)	RCC Bus, bit 04 (RCC-IS Configuration Control Level, initiates reset of ISBBF and ISDBF, also the set of ISTBF)
CCC.X	CPAL.X	CP Activity Level External (integral part of CPAUF)
PMC.X	CPTBL.X-ST	CPTBF Level External-Signal Transmission (Functional part of reset commands and also a set command for CPAUF; reset command for CPTBF)
	CPTBL.X-GT	CPTBF Level External-Ground Transmission (Twisted pair ground of CPTBL.X-ST)
ICC	ICCSL	ICC Sequence Level (Part of set command for CPSPF)
MCC.SP	MSICL	MCC Set ISCBF F/F Level
	MSIBL	MCC Set ISBBF F/F Level
	MRIBL	MCC Reset ISBBF F/F Level
	MSITL	MCC Set ISTBF F/F Level
	MRITL	MCC Reset ISTBF F/F Level
	MRIDL	MCC Reset ISDBF F/F Level
	MSAL	MCC Set Active Level (Set CP Active Unit Flip-Flop and Reset CP Trouble Flip-Flop)
	MSSL	MCC Set Standby Level (Set CPTBF)
PCC	DCCAF	Dual Cycle Control A F/F (indicates time interval T2 of first cycle through T2 of second cycle of the dual cycle instructions: RIS, WIS, WISN, RISN, RISA, WISD, RIST, XEC and XECN).
TGC	T0AL	Timing Interval 0 Accept Level (used for the set or reset of the second stage of the Dual Rank ISCBF, ISBBF and ISTBF flip-flops.
	T3AL	Timing Interval 3 Accept Level (specifies when the CPSPF F/F may set as a function of the ICCSL input being true).
MAC	IMSB.B26 .B29	Internal Maintenance Select Bus Bits 26 and 29 — selects CCC flip-flops for status read-out (RMSG instruction) or status control (WMCP and WMCG instructions).
	IMDB.B00-B12 B16-B28	Internal Maintenance Data Bus, Bits 0 through 12, and 16 through 28 — this data bus provides the control bits that specify set or reset of CCC flip-flops during the execution of a WMCP or WMCG instruction. For MCG-6, Bit 00 sets ISCBF, Bit 01 sets ISBBF, Bit 02 sets ISTBF, Bit 03 sets ISDBF; Bits 10–12 set respectively CPTBF, CPSPF and DF; Bits 16–19 reset respectively ISCBF, ISBBF, ISTBF and IBDBF; and Bits 27–28 reset respectively CPTBF, CPSPF and DF.

—Continued

USER	MNEMONIC	DESCRIPTION
RCC	CPAL	Central Processor Activity Level (Central Processor Active Unit Flip-Flop Output)
	CPSPF	CP Separate Flip-Flop (CPSPF F/F Output Level)
TGC	CPAL	Central Processor Activity Level (Central Processor Active Unit Flip-Flop Output)
MMC	CPSPF	CP Separate Flip-Flop (CPSPF F/F Output Level)
	ISDBL	Instruction Store Data Bus Level — (Represents condition of ISTBF and ISDBF flip-flops are set). Inhibits IS Error Level generation within the MMC during IS* data fetch operations.
CCC.X	CPAL	Central Processor Activity Level Integral part of CPAUF F/F)
PMC	CPTBF-ST	Central Processor Trouble F/F-Signal Transmission (Used for Reset of CPTBF and set or reset of CPAUF in CCC.X) routed through power "ON" switch in PMC.
	CPTBF-GT	CPTBF-Ground Transmission (Twisted Pair ground of CPTBF-ST)
IOC	CPTBF	Central Processor Trouble Flip-Flop — This level, when true, will inhibit the CP from sending to the IS,PS and PU complexes.
	CPAL	Central Processor Activity Level
	SISB0L	Send Instruction Store Bus 0 Level
IOC	SISB1L	Send Instruction Store Bus 1 Level
	RISB0L	Receive Instruction Store Bus 0 Level
	RISB1L	Receive Instruction Store Bus 1 Level
MAC	IMRB.B00-B13	Internal Maintenance Return Bus, Bits 0 through 13 — the CCC sense points selected by the IMSB.XX signals are gated onto this bus during the execution of a RMSG instruction. Bits 00 - 03 sense respectively ISCBF, ISBBF, ISTBF and ISDBF; and bits 10–13 sense respectively CPTBF, CPSPF, DF and CPCL.
	CPAL	Central Processor Activity Level (Central Processor Active Unit Flip-Flop Output).
TMC	DCPAL	Diagnostic CP Activity Level — When false, inhibits the Repetition Rate Error Indication Flip-Flop (RREIF) from becoming set in the standby CP.

In addition to the four bus control flip-flops this group also contains combinatorial logic circuitry **106** that encode the outputs of the above flip-flops **105** into four bus control levels. These levels enable the activation of the instruction store bus system interface circuits in the IOC that correspond to the configuration specified by the bus control flip-flops. The four bus control levels (which are transmitted to IOC **24**) are:

1. Send Instruction Store Bus **0** Level (SISB0L).

2. Send Instruction Store Bus 1 Level (SISB1L).
3. Receive Instruction Store Bus 0 Level (RISB0L).
4. Receive Instruction Store Bus 1 Level (RISB1L).

The ISCBF, ISBBF and ISTBF flip-flops are of the Dual Rank type seen in FIG. 22 and become set or reset in T0AL of the instruction cycle following the WMCG or WMCP (set or reset) instruction which means that the instruction following the WMCG or WMCP instruction will be the last one to be executed from the old configuration.

FIG. 23 shows a table of bus control flip-flops configurations. FIG. 24 shows bus control level equations for generating the required configuration signals. FIGS. 25–40 represent the 12 possible IS bus configuration with either CP unit active and the other standby.

Turning now to FIG. 41, the CPAUF flip-flop 100 and its associated circuitry is seen in more detail. As mentioned above, this flip-flop specifies whether the CP in which it resides is the active or the standby copy. The signal CPAL is representative of its associated central processors being the active copy. The flip-flop basically comprises four NOR gates 108, 109, 110 and 111, the first two of which are associated with CP1 and the latter two of which are associated with CP0. The two trouble flip-flops CPTBF0 and CPTBF1 are designated respectively 112 and 113 for the two CPs. Because a portion of the CPAUF flip-flop 100 is located in each CP, the interconnections between output and input are effected by means of cable drivers. That is, the symbol designated 114 is an inverting cable driver, and the symbol designated 115 is a double inverting cable driver.

FIG. 41 also shows the circuit location of the Power Monitor Circuits in each CP copy, see the dashed blocks labeled PMC0 and PMC1. The function of the PMC has been described above, just before Section III.

FIG. 42 is a table showing the operation in the CPAUF for both switching and non-switching sequences in the active and standby CP copies. For example, the chart to the left half of FIG. 42 illustrates the six signal level changes in switching CP copy 0 from active to standby. The table to the right-hand portion of FIG. 42 attempts the same switching; however, in this case the trouble flip-flop CPTBF1 designated 113 is set, indicating that copy 1CP cannot assume an active state. The CPAUF therefore reassumes the original status.

V. Input/Output Circuit (IOC)

The Control and Maintenance Complex of the TSPS machine contains four major complexes; Central Processor (CP*), Instruction Store (IS*), Process Store (PS*), and Peripheral Unit (PU*). The CP* requires access to the other complexes; IS* and PS* for instructions and data storage and PU* to control the many functions required to process calls. To accomplish this access the CP* is connected to the IS*, PS*, and PU* via a duplicated system of AC cable drivers, receivers, and buses.

The primary function of the Input/Output Circuit (see 24 of FIG. 1) (IOC) is to provide the interface in the CP* through which the CP* gains access to the external bus system and complexes. The IOC will be disclosed herein only so far as is necessary to understand how reconfiguration of the Information Store and Central Processor copies is accomplished.

The basic functions of the interface between the CP and other complexes are: 1. To identify the complex to use as input or source; 2. To identify the complex to which the source is to transmit; and 3. To determine the timing for transmission. This timing for transmission from source is defined and controlled by the CP*, and the timing for reception is specified by the CP but controlled by the transmitting complex. The functions to be performed are identified by the IOC from the control signal inputs it receives.

Since the sending complex controls the data reception in the IOC, which is due to the IS*, PS* and PU* complexes being asynchronous with respect to the CP*, the secondary function of the IOC is to provide a buffer register for return data from each complex. The inputs and outputs of the IOC are shown in FIG. 6 which also schematically shows the routing of data and the buffer registers: Instruction Store Buffer Register 62, the Process Store Buffer Register 63, and the Peripheral Unit Buffer Register 64, also respectively denoted ISR, PSR and PUR.

Communication channels and subsystems are identified using the following nomenclature and format. The format consists of up to eight alpha numeric characters in the format below and with the following information:

(XXX.XX.XXX)

The three character prefix (first three characters) defines the complex and the copies with the exception of the PU*. Characters one and two identify the subsystem CP, IS, PS, PU, IP (signifies Instruction Store and Process Store subsystems), or the Peripheral Control ($\overline{PC}$) of the PU complex.

* — Complex, the collection of all units defined by the subsystem or PC identifier.
1 — Copy 1 of the complex.
0 — Copy 0 of the complex.

The component characters (characters four and five) specify the bus:
BS — The Bus System.
AB — The Address Bus.
DB — The Data Bus.
RB — The Return Bus.

The last three suffix bits refer to one of "n" bits on a bus. The notation is of the following form:
.BXX — The bit specified, where XX could range from 00 to 33.

Organization of the IOC (see FIGS. 159 and 160 which are functional block diagrams)

In general, the IOC 24 is organized into three sections. They are: CP*-IS* communications; CP*-PS* communications; CP*-PU* communications. Each section contains: AC cable drivers and receivers required for bus system comcunications; buffer register, composed of set-reset flip-flops, for retaining data received from the bus system; and combinatorial logic circuitry for channeling the address and data words onto, or the data word from the appropriate bus(es) of the system.

The bus systems include: (1) the IS* bus system which includes the IS* Address Bus 120, IS* Return Bus 121 and the IP* Data Bus 122 which is shared with the PS*; (2) the PS* Bus system which includes the IP* Data Bus 122, just mentioned, the PS* Address Bus 123, and the PS* Return Bus 124; and (3) the PC* Bus

system which includes the PC* Address Bus **125**, the PC* Return Bus **126** and the PC* Data Bus **127**. It will be observed that whereas the PC* has a separate Data Bus, the IS* and PS* share one.

CP*-IS* Communications

See FIGS. **169–171** of the above-identified Brenski, et al. application Ser. No. 289,718 for word formats on the communication buses.

This section executes the channeling of:

a. The **23** bit IS address word onto one or both of the IS (**0** or **1**) Address Bus(es), namely bits **9** through **31** (IS*.AB.B09-B31).

b. The **33** bit IS data word onto one or both of the IS & PS (called IP) (**0** or **1**) Data Bus(es), bits **0** through **32** (IP*.DB.B00-B32).

c. The **34** bit IS data word from one or both of the IS (**0** or **1**) Return Bus(es), bits **0** through **33** (IS*.RB.B00-B33) into the Instruction Store Register (ISR), to be described presently.

The IOC circuits for each copy of CP are similar, and only one need be described in further detail for a complete understanding of the invention.

IS0IF and IS1IF Operation (see FIG. **44**)

The Instruction Store **0** Inhibit Flip-Flop, denoted **130**, (IS0IF) and Instruction Store **1** Inhibit Flip-Flop **131** (IS1IF) when set, provide individually controllable inhibits of IS0.AB and IS1.AB for diagnostics. These flip-flops can be sensed via MAC and can also be set or reset as follows:

Active CP
Set: via MAC only.
Reset: hardware interrupt occurence or after manual switch of CP's, or via MAC.
Standby CP
Set: via MAC, or after manual switch of CP's.
Reset: via MAC only.

FIG. **45** shows the IOC gating for selecting the IP* Data Bus **122**.

Instruction Store Register (ISR) Operation

As shown in FIG. **46**, the ISR **133** loads a received IS* unit word upon the All Seems Well Level (IS*.RB.B33) becoming true and provided the Instruction Store Register Enable Flip-Flop **134** (ISREF) is set. ISREF isolates the ISR from the IS*.RB between the start of T0AL and the start of T5AL.

The ISR must be reset prior to receiving an IS* unit word. The "DO" command generated by the active CP* is coupled from the IS*.AB.B09 and used for ISR reset in both CP's. This insures the ISR of the stopped standby CP* will never contain more than one IS* unit word so that when the standby CP* is started in sync with the active CP*, correct match operation or instruction execution can be expected.

ISREF Operation

The ISREF flip-flop **134** is primarily an ISR enable flip-flop which when set (by T5AL, the timing interval immediately following the transmission of the IS address word) permits the return data word from an IS* unit to be accepted into the ISR upon recognizing the "all seems well" level becoming true ("all seems well" must be received **100** ns before the end of T7PL). The in T0AL the ISREF is reset which inhibits the ISR from accepting erroneous IS*.RB information. (Noise occurring on the IS*.RB between T7PL and T5PL could have possibly altered the ISR contents).

The ISREF can be sensed and reset via MAC for diagnostic purposes, as shown.

VI. COMMUNICATIONS WITH INSTRUCTION STORE

The Instruction Store complex IS* (**19** of FIG. **1**) consists of two identical copies IS0 and IS1 (denoted **36** and **37** in FIG. **2A**).

Each copy contains up to eight units (numbered **0** through **7**). Each instruction Store unit is identified by its copy number and unit number.

Example: IS0.U3 = Copy **0** of IS Unit **3**

The capacity of an IS Unit is 16,384 words of 33 bits (32 data, 1 for odd parity).

The Instruction Store complex is composed of two types of store units. One unit of each copy is a random access read only unit. All others are 16,384 word read-write random access units. The read only unit is always defined to be Unit **0** of each copy.

Referring to FIG. **47**, the following parts of an IS Unit will be discussed:

a. Address Register (AR) is a twenty-two bit register divided as follows:

Unit Address — 3 bits
Word Address — 14 bits
Commands — 5 bits

An IS Unit accepts an address in AR on receipt of the DO signal from CP. (This means that all units on a bus accept addresses simultaneously).

b. Decode Logic is a straightforward combinational logic network which performs the following functions:

decodes Unit Address. Receives Unit Address is compared with unit number (hard-wired in each unit during installation). Unit proceeds with operation only if these numbers are equal.
decodes commands and performs required operations.
decodes word address for operation involving memory bank.

c. Data Register (DR) is a 33-bit register which:
accepts data from CP (write operation).
accepts data from memory bank (read operation).

d. Status Flip-Flops

These flip-flops modify the reaction of the Unit to commands from CP. The following FF's are included:

WDF: Write-disable FF (denoted **142**)
When WDF = **1**, unit ignores normal "write" commands.
Note: The read-only unit (Unit **0**) does not have a WDF.
TBF: Trouble FF (denoted **141**)
When TBF = **1**, unit ignores all "normal" commands.
TDF: Transmit Disable FF (denoted **143**)
When TDF = **1**, unit performs all memory operations but no information is transmitted over bus.

ARF: Address Register Flip-Flop **140**, described further below.

Instruction Store Bus System

The Instruction Store Bus System IS*.BS (FIG. **2A**) includes two identical copies: IS0.BS and IS1.BS.
IS0.BS connects both CP's with the units of IS0.
IS1.BS connects both CP's with the units of IS1.

IS*.BS can be divided into three buses:

IS*.AB	Address Bus	23 bits (22 address bits, DO).
IP*.DB	Data Bus	33 bits (32 data, 1 odd parity).
IS*.RB	Return Bus	34 bits (32 data, parity, ASW).

The data buses IP*.DB are shared by PS* and IS*. In other words, these buses can be considered as part of both PS&.BS and IS&.BS.

Interface CP*-IS*.BS

In each CP, the input-output circuit IOC contains the AC cable drivers CD, the AC cable receivers CR and the buffer-register (ISR) for IS return data that form the interface with IS*.BS, as shown in FIG. **48.**

CD1 is group of **23** drivers to outpulse address and "DO" to IS0.

CD2 is group of **33** drivers to outpulse data + parity to PS0 and IS0.

CD3 is group of **23** drivers to outpulse address and "DO" to IS1.

CD4 is group of **33** drivers to outpulse data + parity to PS1 and IS1.

CR1 is group of **34** receivers to receive return data from IS0.

CR2 is group of **34** receivers to receive return data from IS1.

CR3 is used to receive the "DO" signal from IS0.AB to reset ISR.

CR4 is used to receive the "DO"signal from IS1.AB to reset ISR.

In each CP, individual groups of CD and CR can be enabled or disabled. This determines the CP*-IS* configuration.

Example: If CP1 is to transmit to IS0, then CD1 and CD2 are enabled. Only one CP is allowed to transmit on a bus, therefore CD1 and CD2 in CP0 should be disabled.

Example: If CP1 is to receive from IS0, then only CR1 and CR3 have access to its buffer register ISR. In this situation CP0 may or may not also receive from IS0, so we cannot say anything about its CR1 and CR3.

Interface IS(CN).BS - IS(CN).UN

The suffix (CN) is used when a statement or a drawing applies equally to copy **0** and to copy **1.**

Each IS Unit contains two groups of AC cable receivers and one group of AC cable drivers, as shown in FIG. **49.**

CR1 is group of **23** receivers to receive address and "DO" from IS(CN).AB.

CR2 is group of **33** receivers to receive data + parity from IP(CN).DB.

CD1 is group of **34** drivers to outpulse return data on IS(CN).RB.

NOTE: Both IS(CN).U0 contain one additional AC cable receiver CR3 connected to RCB.B06. CR3 receives a signal (R1CCL) used to reset the trouble FF(TBF) of both IS(CN).U0.

Interface CP*.RCC-IS*.U0

The Recovery Control Circuit Bus (RCB in FIGS. **48** and **49**) connects directly to Unit **0** of both IS(CN). RCB.B04 which carries the signal R1CCL generated in the Recovery Control Circuit, as shown in FIG. **13,** is used to reset the trouble FF (TBF **141** in FIG. **47**) and transmit disable FF (TDF **143**) in both IS(CN).U0.

The memory unit is capable of "read" and "write" operations in the normal and control modes and only read in the special mode. The control and special modes are used only for diagnostic and maintenance testing. "Read" and "write" commands for each mode give six operations, referred to as cycles.

Normal Modes

Read Normal (RN) Cycle

The RN cycle sometimes referred to as a Read/Restore mode, consists of reading (and subsequently restoring) the **33** bit word at core stack location W specified by the address and storing it into the Data Register (DR). The Unit Selected (PS(CN).UN) transmits contents of the DR on the Process Store Return Bus, PS(CN).RB.

Write NOrmal (WN) Cycle

The WN cycle, sometimes referred to as a clear/write mode, consists of clearing existing data from the addressed unit word and writing into core stack location W the new **33** bit data that was accepted by DR of the addressed PS Unit. The PS(CN).UN transmits address echo on the PS(CN).RB.

Control Modes

Read Control (RC) Cycle

The RC cycle consists of reading non-memory information from specific address dependent points that are internal to the memory unit but not in the memory stack itself. These points assume logical states which may or may not be address dependent since only the three least significant bits of the address field will specify the points to be sensed. Information from the sensed points is gated onto the PS(CN).RB.

Write Control (WC) Cycle

The WC cycle consists of writing information gated from the DB into specific flip-flop points (either special maintenance flip-flops or the data register) determined by the address and which are internal to the memory unit but not in the memory stack itself. This cycle also returns the address echo on the PS(CN).RB.

Special Test Mode

Read Special Test (RST) Cycle

The RST cycle gates information directly to the PS(CN).RB from specific internal address independent points (hard wired into PS(CN).RB gates) whose states are address dependent. The DR and the contents of the core stack are not involved.

Write Special Test (WST) Cycle

The WST places the address echo on the PS(CN).RB. The DR and the contents of the core stack are not involved.

Trouble Flip-Flop (TBF) **141**

This flip-flop, when in the set state, does not allow Read Normal or Write Normal operations. This is accomplished by inhibiting the memory clock. The status of the TBF is determined by the Write Control Cycle when PS(CN).AB − MOD **8** = $\overline{5}\overline{3}$.* TBF sets if IP(CN).DB.B30 was a logical "**1**" during the cycle and TBF resets if IP(CN).DB.B31 was a logical "**1**" during the cycle. If IP(CN).DB.B30 and IP(CN).DB.B31 are both logical "**1**" during the cycle the state of the TBF will be indeterminate, see FIG. **51.** The state of TBF cannot be changed while ARF is set.

Transmit Disable Flip (TDF) **143**

This flip-flop, when in the set state, allows all memory operations but inhibits the transmission of information on the PS(CN).RB. The electromechanical sense points are activated when the TDF is set. The status of the TDF is determined by the Write Control Cycle when PS(CN).AB − MOD **8** = $\overline{5}\overline{3}$. This type of Write

Control operation always leaves the TDF in the reset state unless the IP(CN).DB.B16 was a logical "1" during the cycle.

Address Register Flip-Flop (ARF **140**)

This flip-flop, when in the set state, inhibits the data window strobe as well as the address window strobe for the **14** least significant address bits. This flip-flop will be used for the detection of faults that may occur in the address delay line card as well as the address register card of the memory system that would otherwise go undetected. This flip-flop can be reset with any Write Control Cycle. The ARF flip-flop can be set by a Write Control cycle in the PS(CN).AB − MOD = $\bar{5}\cdot\bar{3}$. The nomenclature PS(CN).AB − MOD **8** = $\bar{5}\cdot\bar{3}$ is used to denote the status of the three least significant bits of the address bus. In the case where MOD 8 = $\bar{5}\cdot\bar{3}$, the octal equivalent of the three least significant bits can be 0, 1, 2, 4, 6 or 7.

Write Disable Flip-Flop (WDF) **142**

This flip-flop, when in the set state, does not allow Write Normal operations. This will be accomplished by inhibiting the memory clock. The status of the WDF is determined by a Write Control Cycle when IS(CN).AB − MOD **8** = $\bar{5}\cdot\bar{3}$. This type of Write Control operation always leaves the WDF in the reset state unless the IP(CN).DB.B28 was a logical "1" during the cycle.

Write Control

This operation controls the states of the maintenance flip-flops described above as well as the flip-flops of the memory data register (DR). The following gives a detailed description of the operation in terms of the operational pulse, mode pulse, and address necessary to perform the operation:

Operational Pulse - Write (W)
PS(CN).AB − Mod **8=5** (see FIG. **50**)
Unit Select PS(CN).UN = **1**
A. Returns Address Echo
B. Places data of IP(CN).DB into memory data registers
C. ASW=**1**
D. TBF=X
E. TDF =O
F. ARF=O
G. WDF=X
ii. AVF=X
I. BVF=X

PS(CN).AB − Mod **8=**$\bar{5}\cdot\bar{3}$ (FIG. 51)
Unit Select PS(CN).UN = **1**
A. Returns Address echo
B. Controls states of the maintenance flip-flops through the IP(CN).DB
C. ASW=**1**
D. TBF=X (will reset unless IP(CN).DB.B**30=1**)
E. TDF=X (will reset unless IP(CN).DB.B**16=1**)
F. ARF=X
G. WDF=X (will reset unless IP(CN).DB.B**28=1**)
H. AVF=X
I. BVF=X

PS(CN).AB - MOD **8=3** See FIG. **5.F.1**
Unit Select PS(CN).UN = **1**
A. Returns Address echo
B. Ignores data of IP(CN).DB
C. Resets ARF if IP(CN).DB.B**21=1**

We claim:

1. A programmable data processing system having first and second central data processors each including processing circuits and maintenance circuits, and first and second storage means, one of said storage means being adapted respectively to be connected in signal communication with one of said central processors and the other storage means being adapted to be connected in signal communication with the other central data processor under normal operating conditions, said maintenance circuits sensing conditions within said system to generate error signals upon detection of faults, said system further including a system recovery program,

wherein the improvement comprises:

subsystem circuit means in each data processor for connecting said first and second central processors with said first and second storage means to form predetermined configurations thereof, one of said central processors being active and the other being passive at any one time, said subsystem comprising

recovery control circuit means including state control circuit means for generating state signals representative of a plurality of predetermined states, and control logic circuit means responsive to said error signals to actuate said state control circuit to generate said state signals in predetermined order as defined by the circuitry of said state control circuit means, said state signals including a central processor switch signal for determining one central processor as being active and corresponding to at least one state change, and further including a storage means switch signal in one of said states for determining one storage means as being primary and being associated with said active central processor corresponding to a desired storage means being coupled with the active central processor, said recovery control circuit means further including means for generating a system recovery signal to initiate said system recovery program when either of said central processor switch signal or said storage means switch signal is generated;

maintenance access circuit means responsive to program controlled signals for generating configuration signals representative of a desired configuration of said central processors and storage means; and

configuration control circuit means responsive to said central processor switch signal and said storage means switch signal of said recovery control circuit means and responsive to said configuration signals of said maintenance access circuit means, and including a first switching circuit means for controlling the switching of said central processors and second switching circuit means for controlling the coupling of said first and second storage means with said first and second central processors.

2. The system of claim **1** wherein said state control circuit means of said recovery control circuit means comprises bistable circuit means for defining said states, said states including **S0, S1, S2** and **S3**, and wired logic circuit means for controlling the state of said bistable circuit means wherein said bistable circuit means can switch in response to said error signals from state **S0** only to state **S1** and from state **S1** only to state **S2** and from state **S2** only to state **S3** and from state **S3** only to state **S2**, said bistable circuit means being capable of being reset to state **S0** from either of states **S1**,

S2 or S3, said recovery control circuit means generating said central processor switch signal only if the standby central processor is capable of becoming active and whenever said bistable circuit means switches from said state S1 to state S2 or from state S3 to state S2, said recovery control circuit means further generating signals calling for said first storage means becoming primary in said state S2 and for said second storage means becoming primary in said state S3.

3. The system of claim 2 wherein said recovery control circuit means further comprises circuit means responsive to instruction signals from said system recovery program for switching from state S1 to state S2, from state S2 to state S3, and from state S3 to state S2.

4. The system of claim 1 further comprising input/output circuit means responsive to said second switching circuit means of said configuration control circuit means for selectively communicating a designated storage means as the primary storage means in a configuration.

5. The system of claim 4 wherein each of said storage means includes an address bus, a data bus and a return bus, and wherein said input/output circuit means include gating means for selectively communicating each of said buses of said storage means for communication with a selected one of said central processors.

6. The system of claim 5 wherein said second switching circuit means of said configuration control circuit means comprises a plurality of bistable circuits responsive to instruction signals received from said maintenance access circuit means for selecting predetermined ones of each of said buses of each of said storage means for selective communication with said central processors in accordance with a desired configuration, whereby independent program control over the resulting configuration may be achieved.

7. A programmable data processing system having first and second central data processors each including processing circuits and maintenance circuits, and first and second storage means provided respectively with first and second bus means for coupling to said data processors, said maintenance circuits sensing conditions within said system to generate error signals upon detection of faults, said system further generating recovery reconfiguration instruction signals under program control,

wherein the improvement comprises:

sybsystem means in each data processor responsive to said error signals only when its associated data processor is active, for reconfiguring said first and second data processors and said first and second storage means to form new configurations thereof thereby to provide an active data processor and a standby data processor, said subsystem means including

recovery control circuit means responsive to said error signals, and including state control circuit means for generating state signals representative of predetermined configuration states defined as S0, S1, S2 and S3, and control circuit means responsive to said error signals, and said state signals to further change the state of said state control circuit and to generate a set storage copy bus signal when said state control circuit is in said S2 state, and a switch processor signal both when said state control circuit is changing from said S1 state to said S2 state or from said S3 state to said S2 state;

configuration control circuit means responsive to said recovery reconfiguration instruction signals, the configuration control circuit means in the active central processor being responsive to said switch processor signal only when the standby data processor is capable of becoming active and including a first circuit for controlling the switching of said data processors; and second bus control circuits for generating bus control signals; and

input/output circuit means responsive to said bus control signals for connecting the designated storage means bus as the primary store bus coupled to the active central data processor.

8. The system of claim 7 wherein said configuration control circuit means includes a processor active unit bistable circuit having a first section including first and second input leads and one output lead associated with one of said central processors and a second section having third and fourth input leads and a second output lead associated with the other of said central processors, said first and third input leads being cross-coupled with the other central processor whereby each of said sections may be made active only when the other section is inactive and said second and fourth input leads are responsive to the switch level signals of its associated recovery control circuit means, and wherein said first and second output leads generate signals representative respectively of an active level in one of said central processors.

9. The apparatus of claim 7 wherein said recovery control circuit means further includes a self-contained timing generator responsive to one of said error signals for generating timing signals to advance the state of said state control circuit means and to control the timing of the generation of said switch processor signal and said switch instruction store signal.

10. The system of claim 7 wherein said first and second bus means of said first and second storage means each includes an address bus, a data bus and a return bus, and wherein said input/output circuit means includes first gating circuitry for controlling the switching of said address bus, second gating circuitry for independently controlling the switching of said data bus, and third switching circuit means for independently controlling the switching of said return bus.

11. The system of claim 7 wherein said bus control circuits of said configuration control circuit means further includes a plurality of bistable circuits responsive to said copy bus signal of said recovery control circuit means and to programmed instruction signals for generating signals representative of a desired configuration of said storage means communicating with said central processors.

* * * * *

UNITED STATES PATENT OFFICE
CERTIFICATE OF CORRECTION

Patent No. 3,828,321 Dated August 6, 1974

Inventor(s) J. A.WILBER, V.K.RICE, R.E. BUHRKE, D.L.SCHULTE

It is certified that error appears in the above-identified patent and that said Letters Patent are hereby corrected as shown below:

Title page, [75] add -- DONALD L. SCHULTE --

Column 39, line 49, change "sybsystem" to -- subsystem --

Signed and sealed this 29th day of October 1974.

(SEAL)
Attest:

McCOY M. GIBSON JR.
Attesting Officer

C. MARSHALL DANN
Commissioner of Patents

M PO-1050 (10-69)

USCOMM-DC 60376-P69
* U.S. GOVERNMENT PRINTING OFFICE : 1969 O—366-334.

Disclaimer

3,828,321.—*John A. Wilber*, Elk Grove Village, *Verner K. Rice*, Wheaton, *Rolfe E. Buhrke*, La Grange Park, and *Donald L. Schulte*, Oak Park, Ill. SYSTEM FOR RECONFIGURING CENTRAL PROCESSOR AND INSTRUCTION STORAGE COMBINATION. Patent dated Aug. 6, 1974. Disclaimer filed Mar. 17, 1975, by the assignee, *GTE Automatic Electric Laboratories Incorporated.*

Hereby enters this disclaimer to claims 1, 4, 5 and 6 of said patent.

[*Official Gazette May 20, 1975.*]