FIG. 1(a)

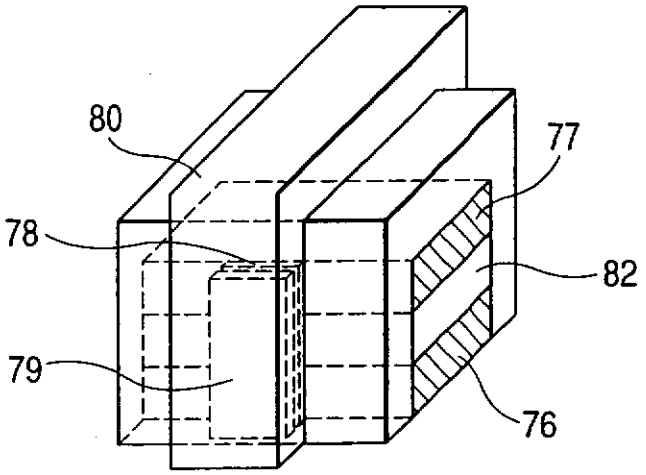


FIG. 1(b)

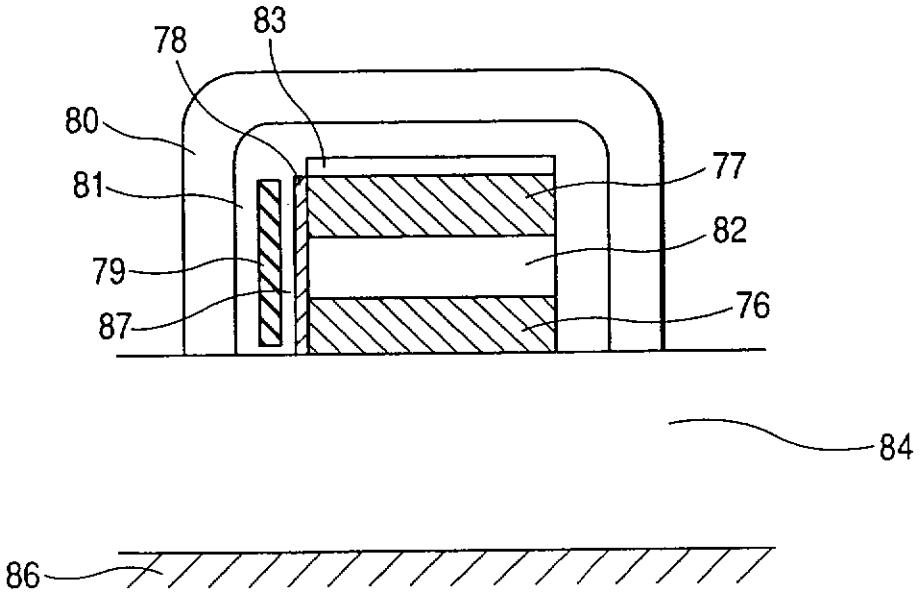


FIG. 2(a)

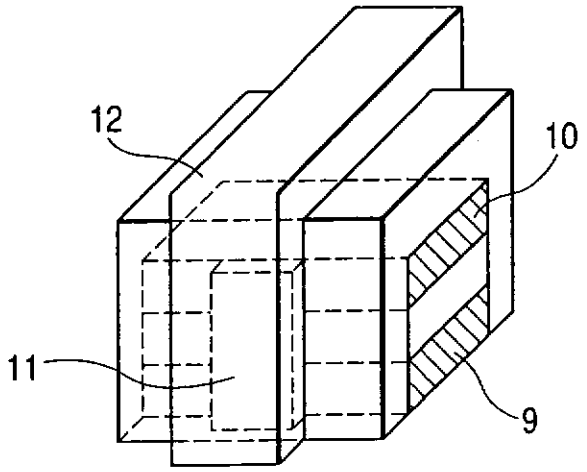


FIG. 2(b)

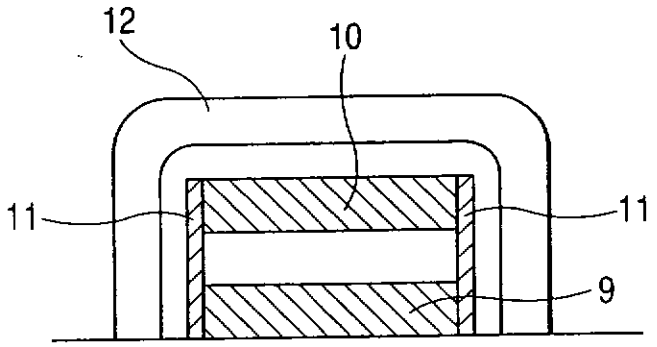


FIG. 3(a)

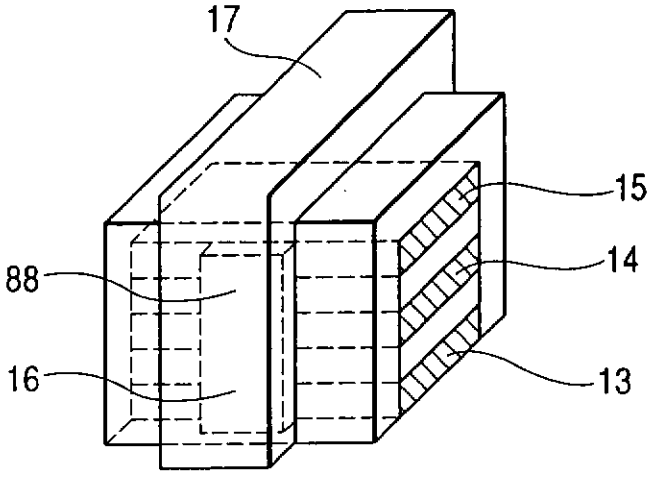


FIG. 3(b)

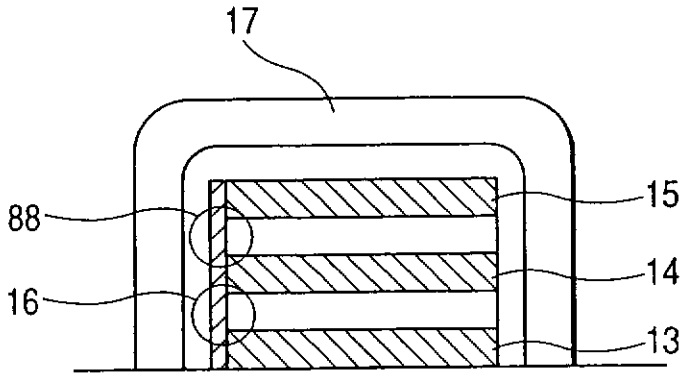


FIG. 4(a)

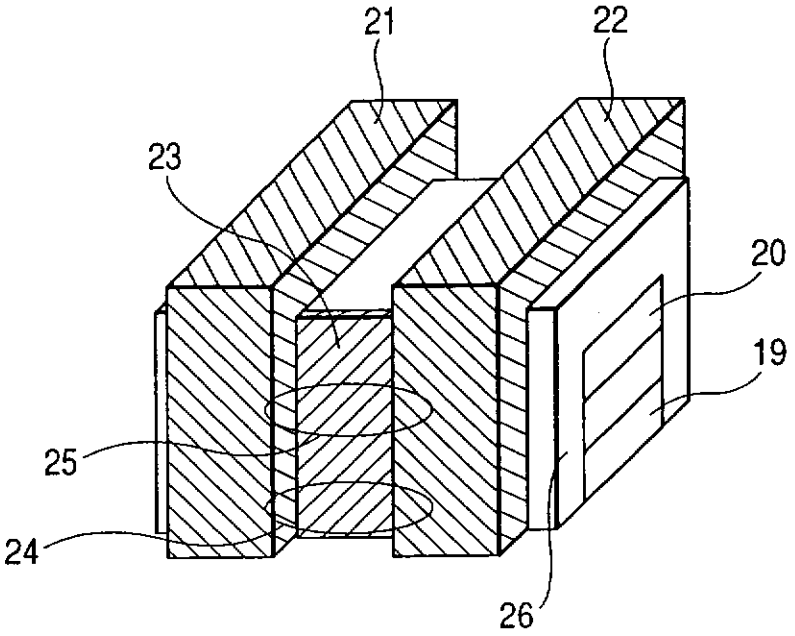


FIG. 4(b)

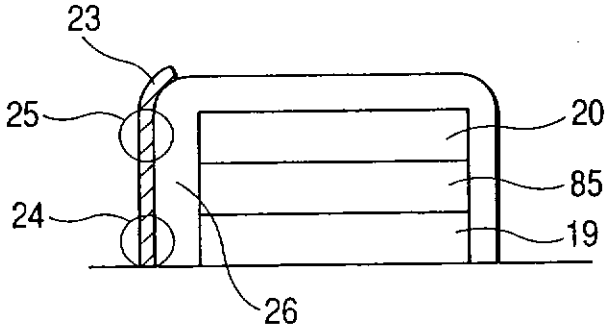


FIG. 4(c)

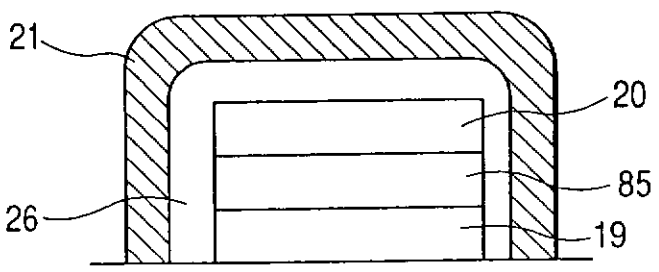


FIG. 5(a)

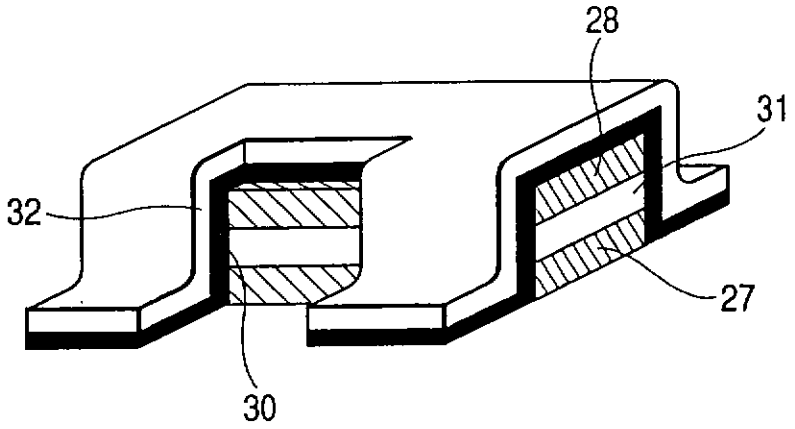


FIG. 5(b)

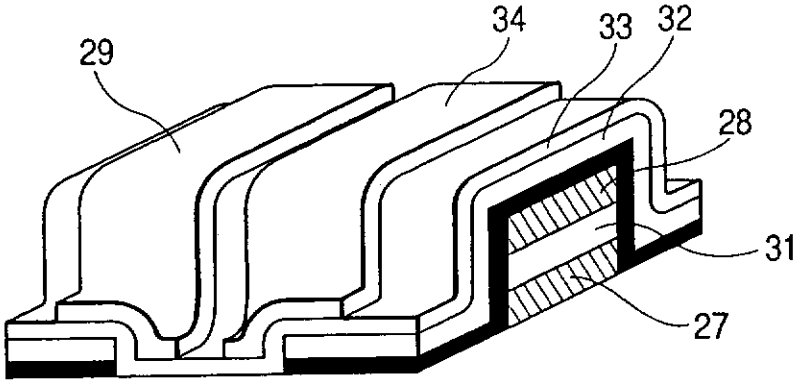


FIG. 6

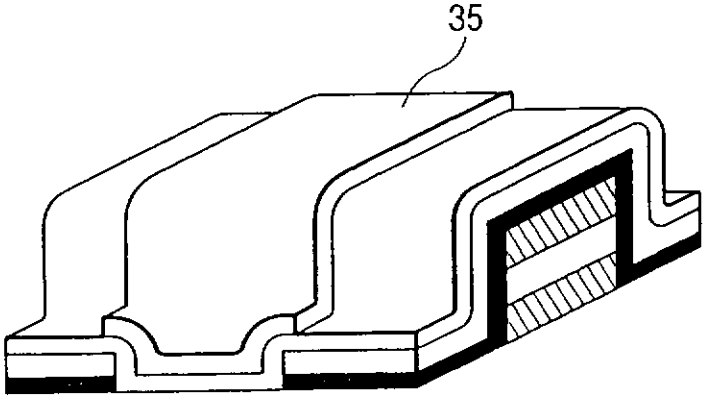


FIG. 7(a)

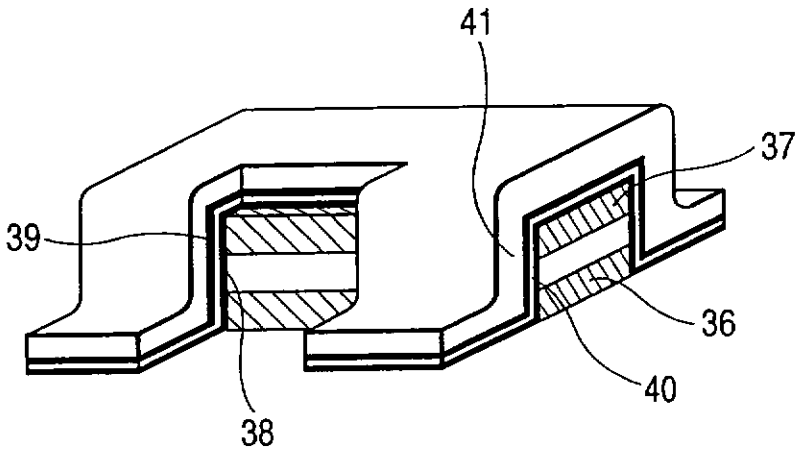


FIG. 7(b)

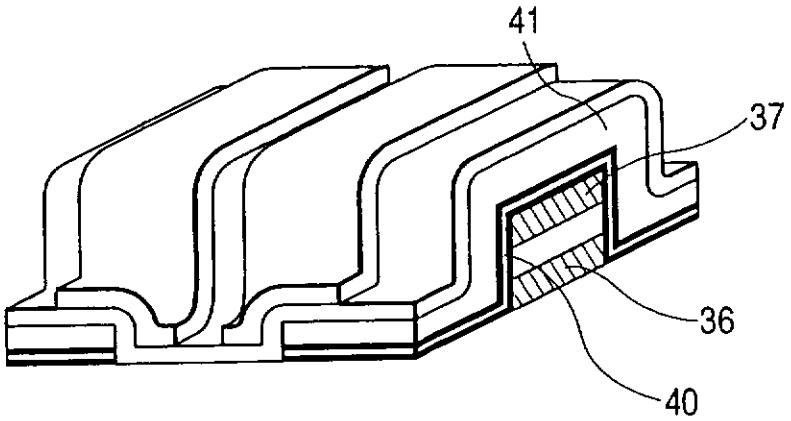


FIG. 8(a)

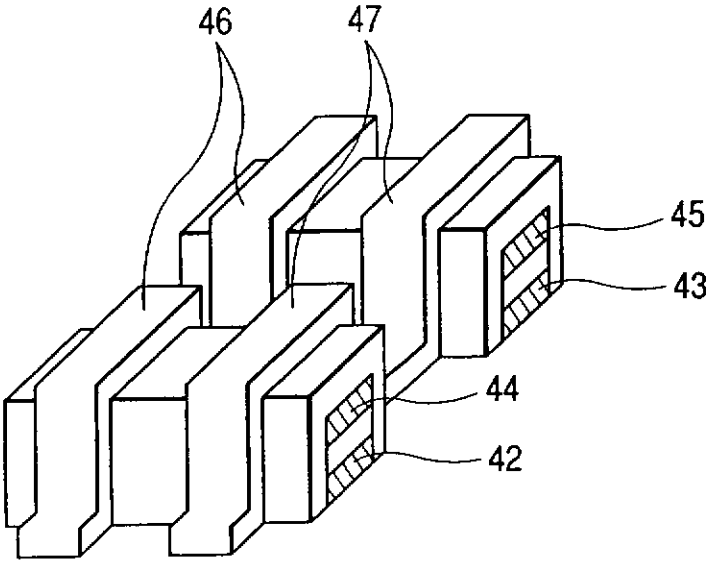


FIG. 8(b)

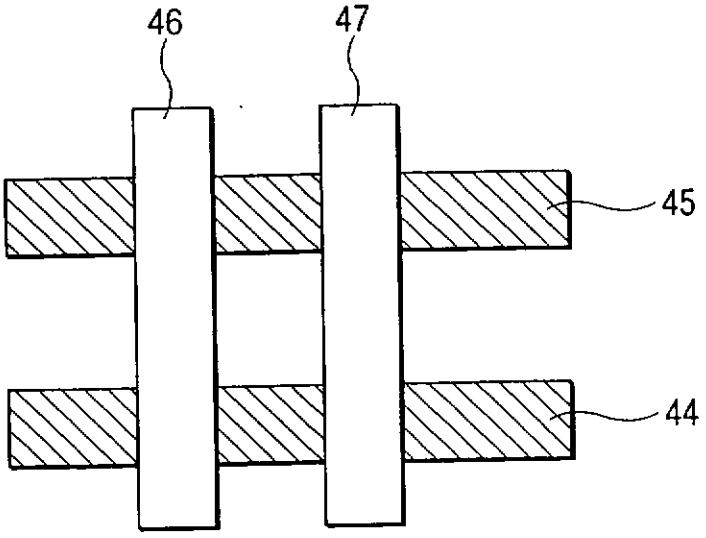


FIG. 9(a)

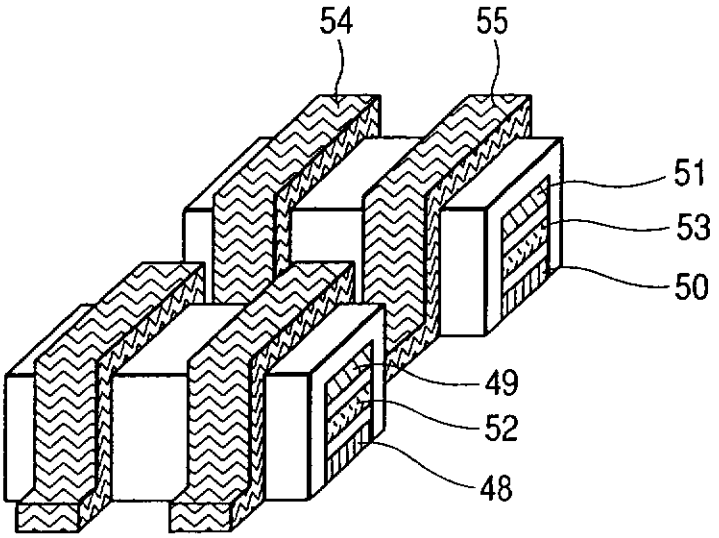


FIG. 9(b)

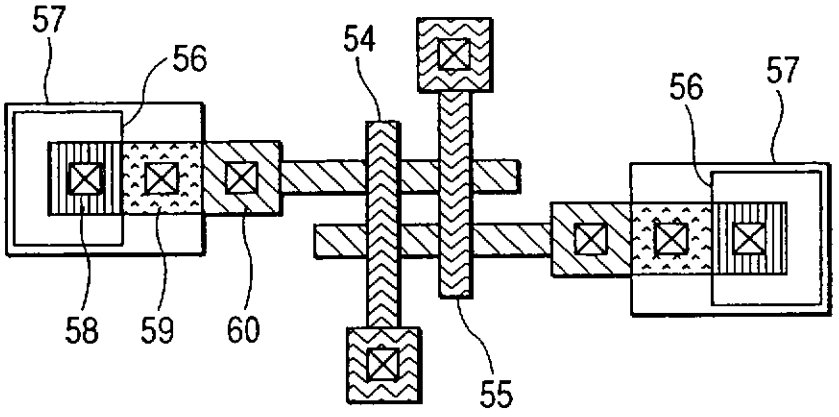


FIG. 10(a)

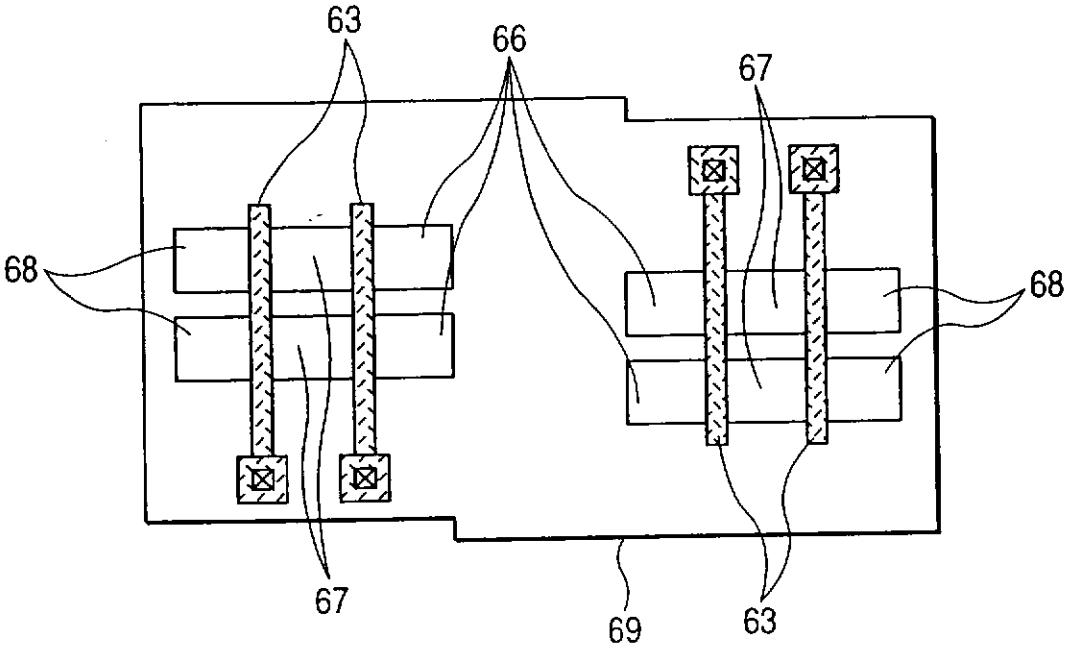


FIG. 10(b)

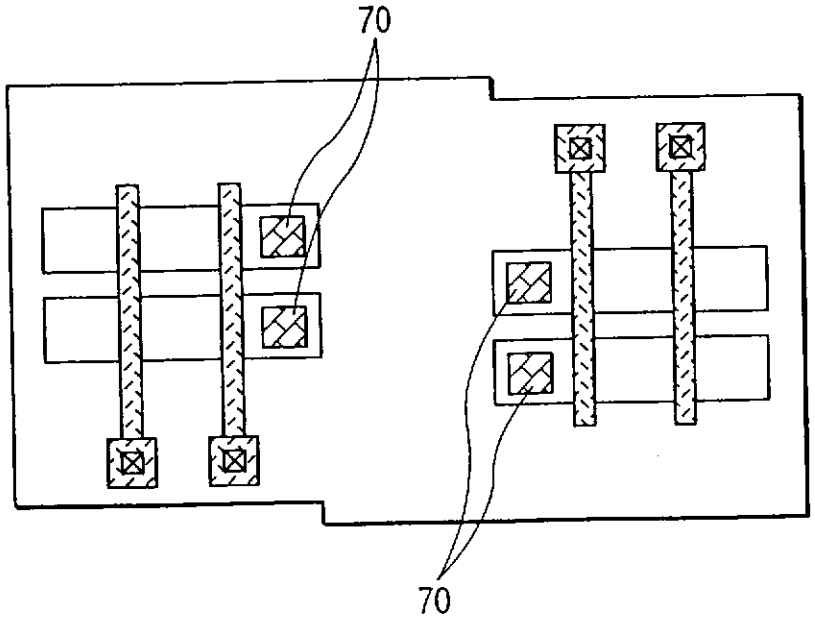


FIG. 11(a)

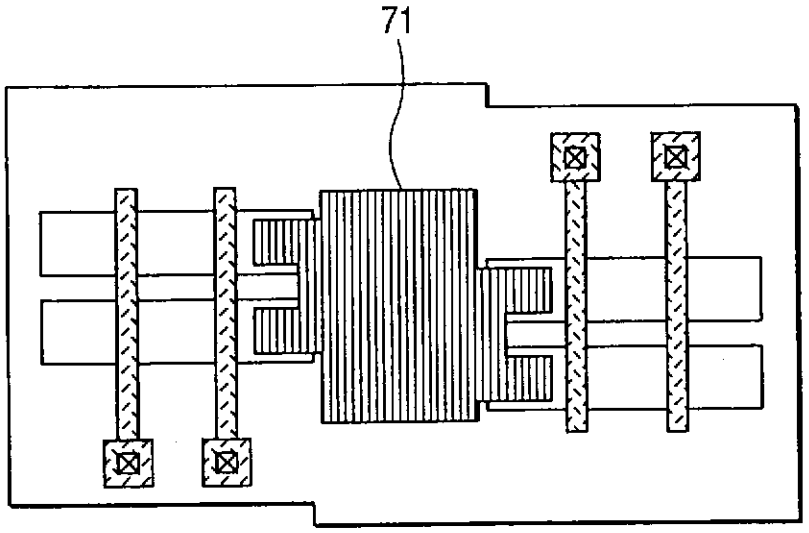


FIG. 11(b)

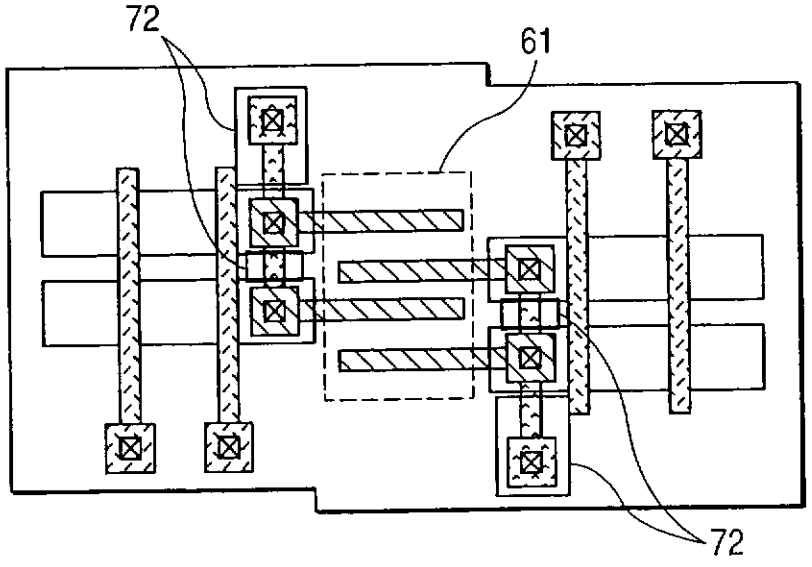


FIG. 12(a)

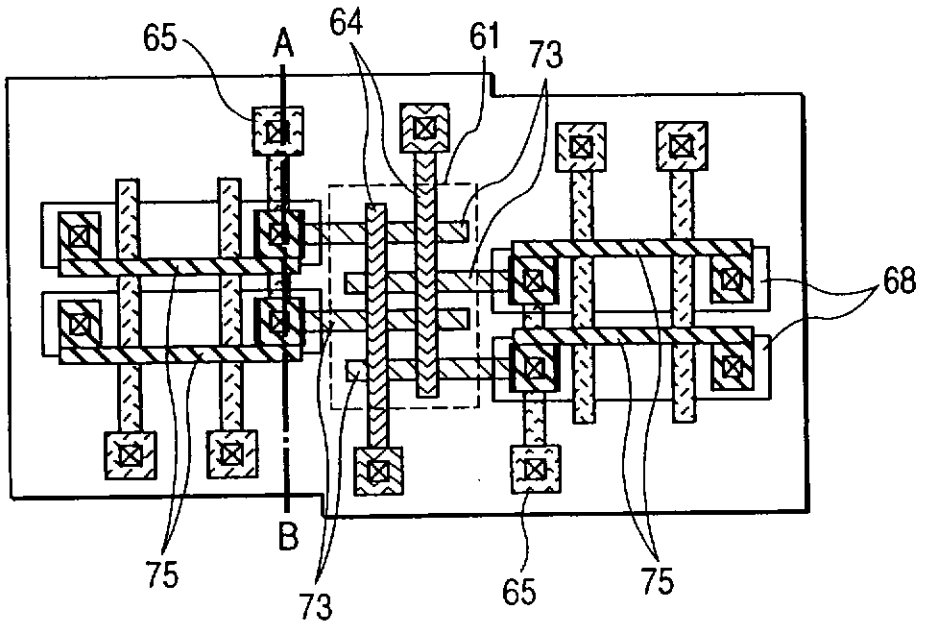


FIG. 12(b)

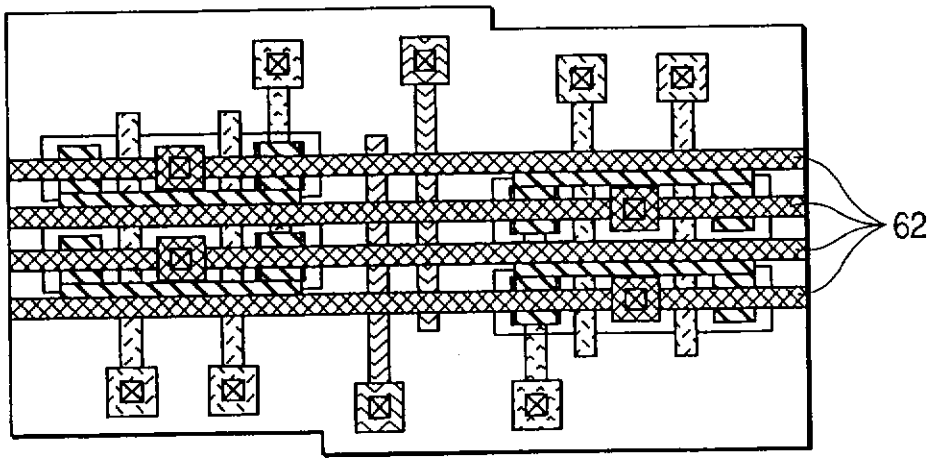


FIG. 13(a)

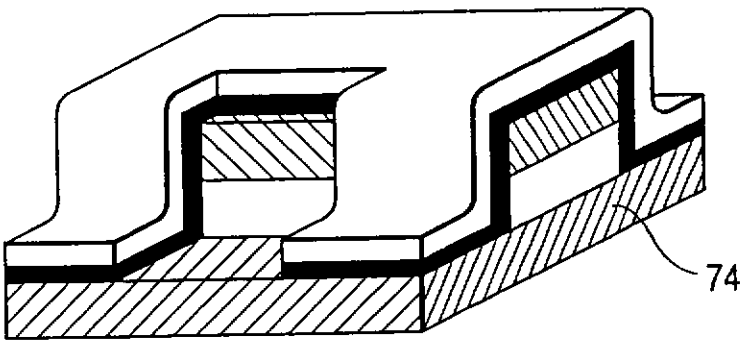


FIG. 13(b)

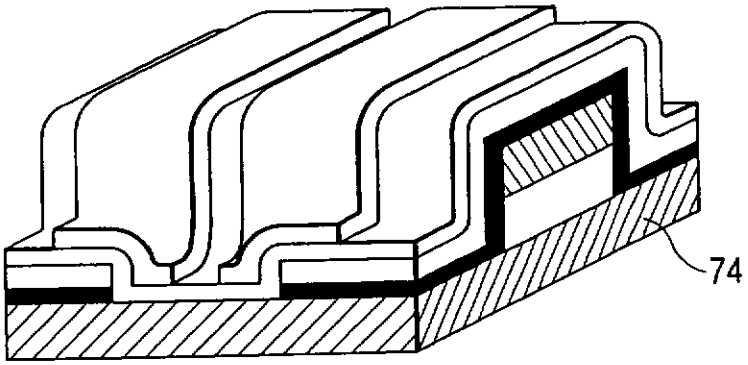


FIG. 14

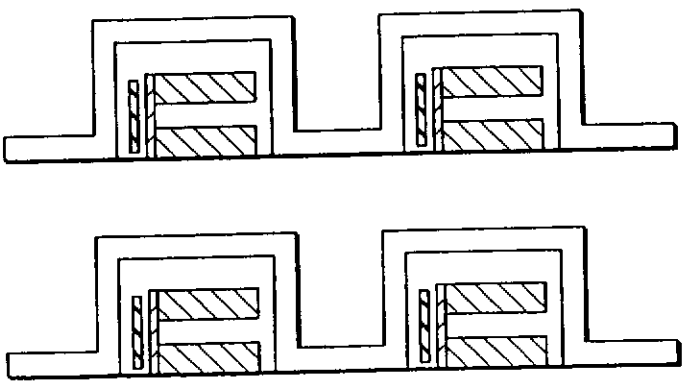


FIG. 15(a)

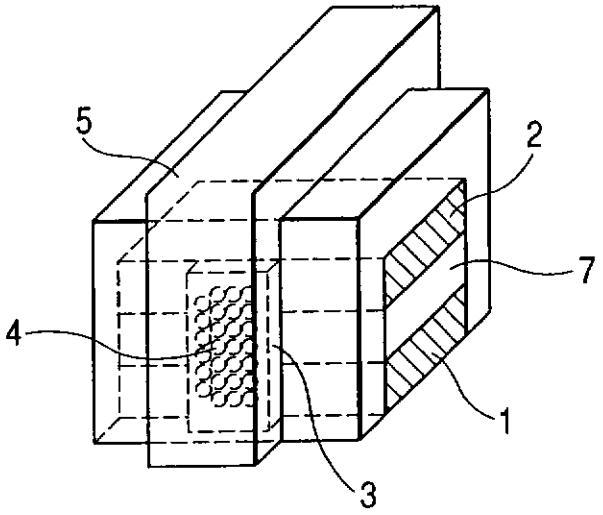


FIG. 15(b)

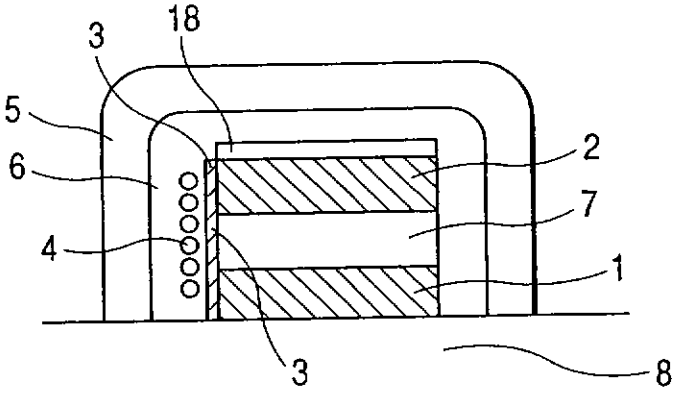


FIG. 16(a)

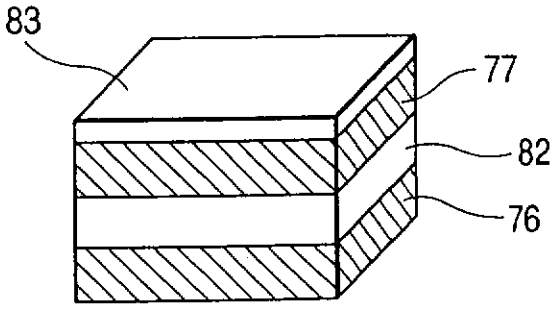


FIG. 16(b)

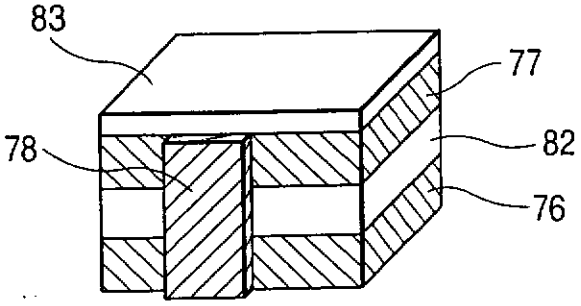


FIG. 17

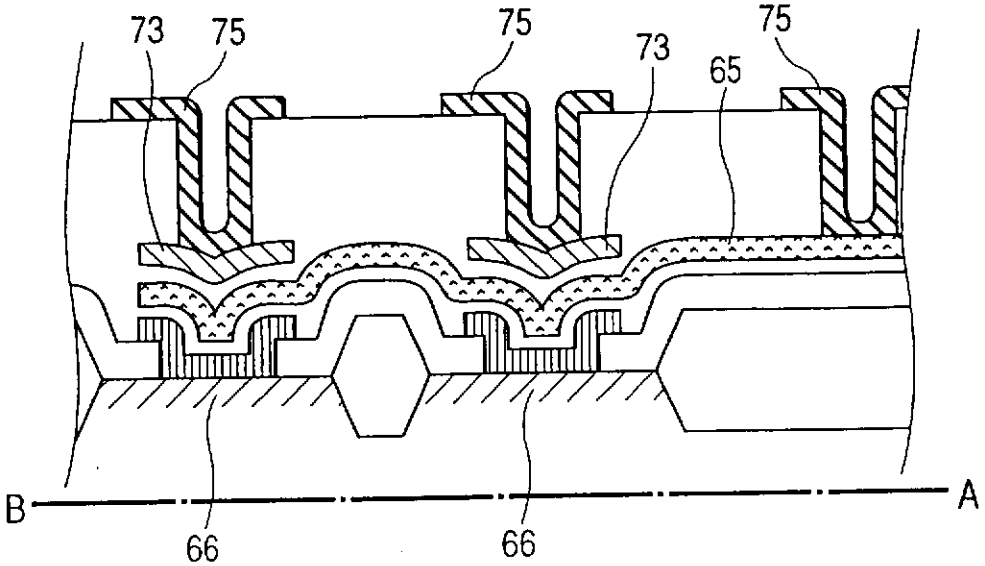


FIG. 18(a)

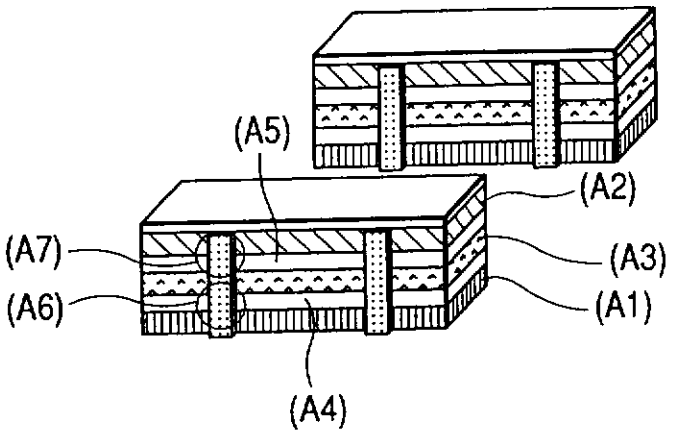


FIG. 18(b)

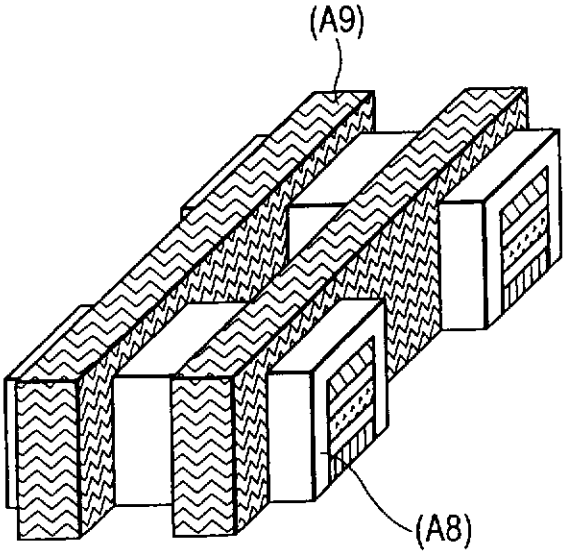


FIG. 19

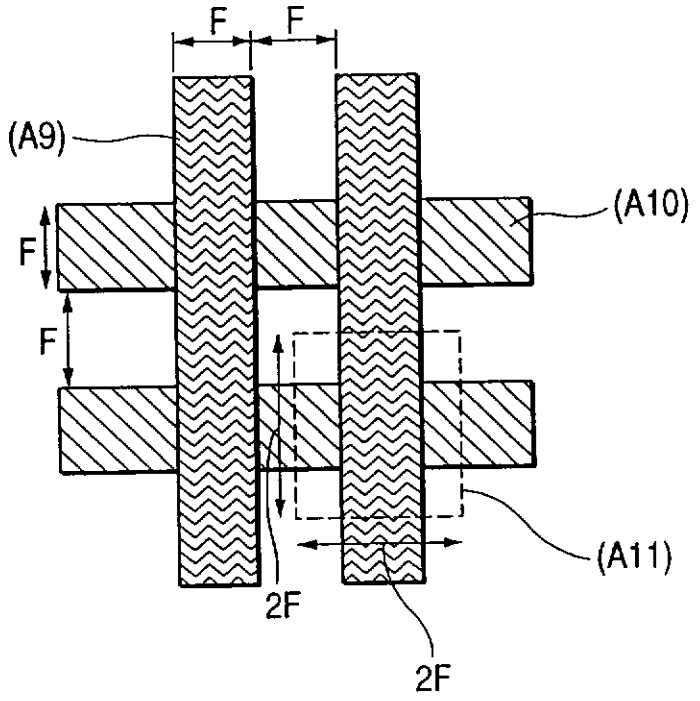


FIG. 20

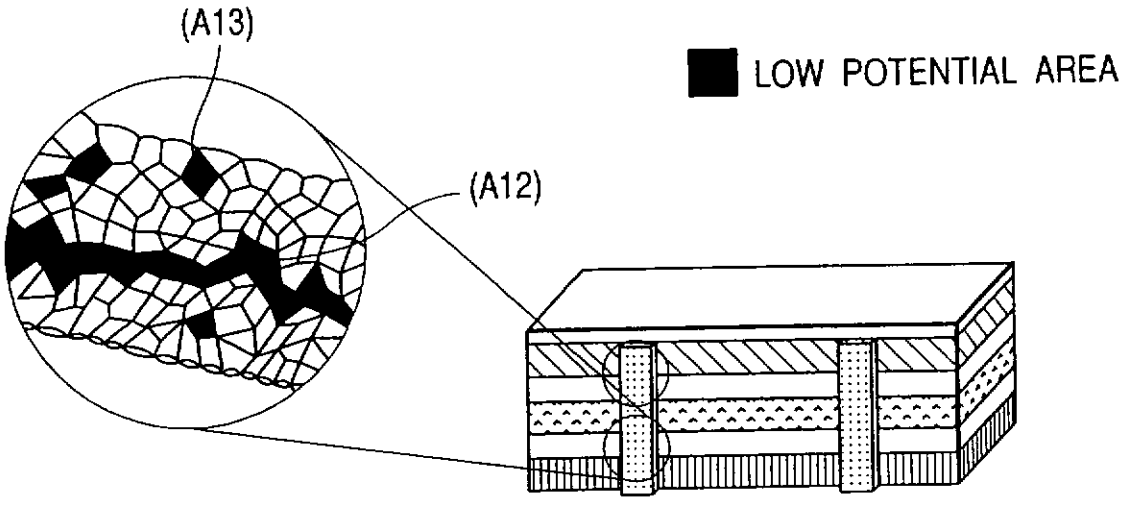


FIG. 21

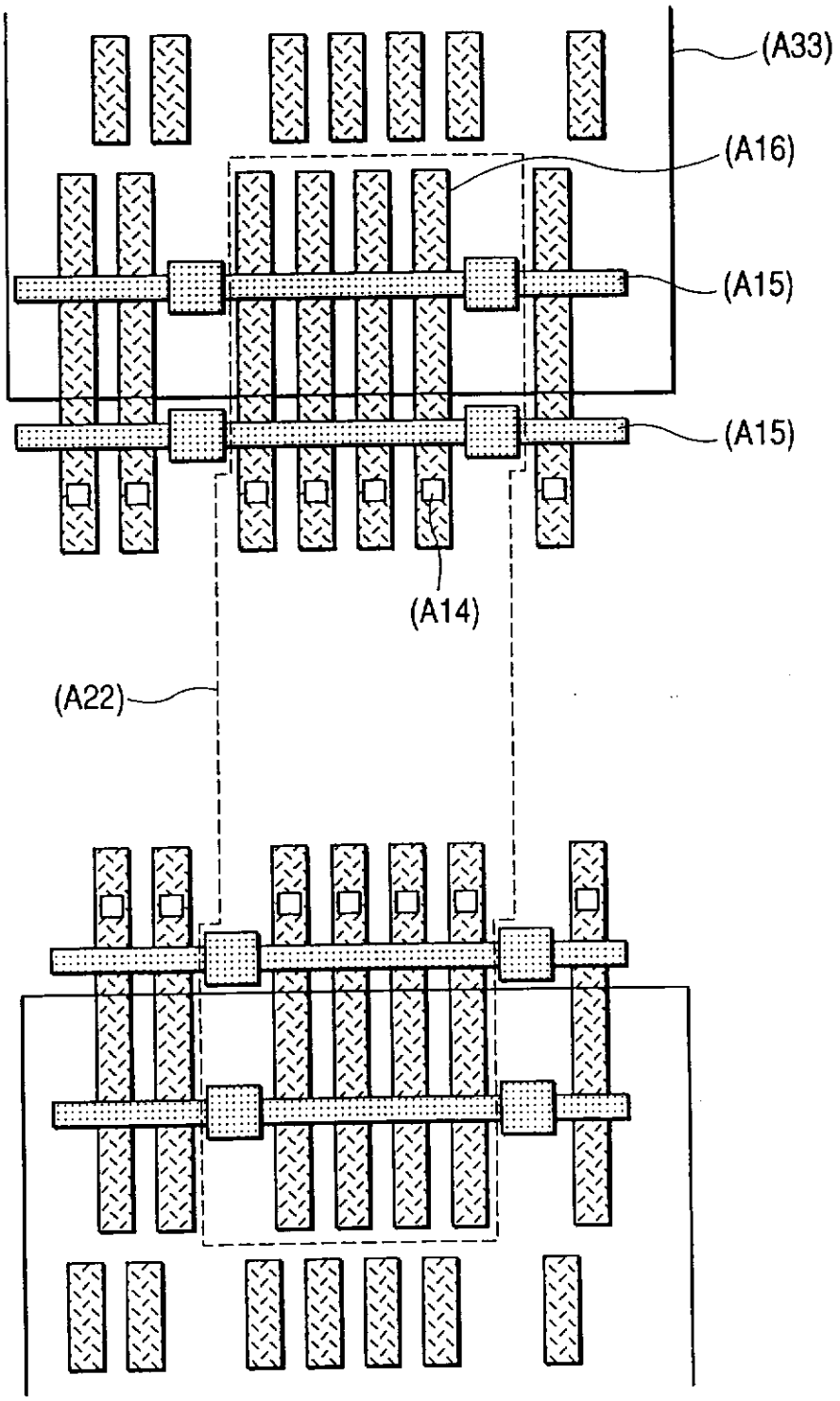


FIG. 22

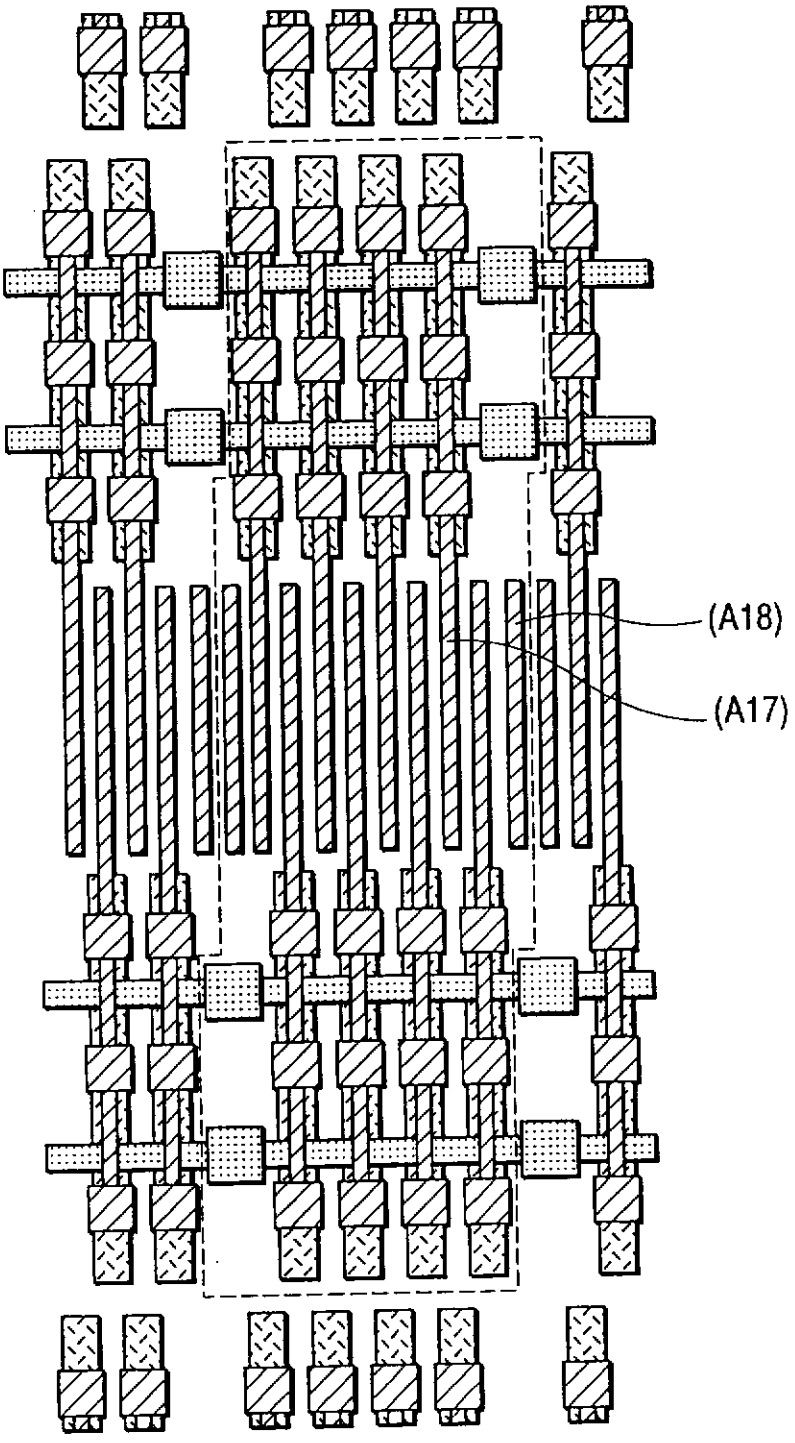


FIG. 23

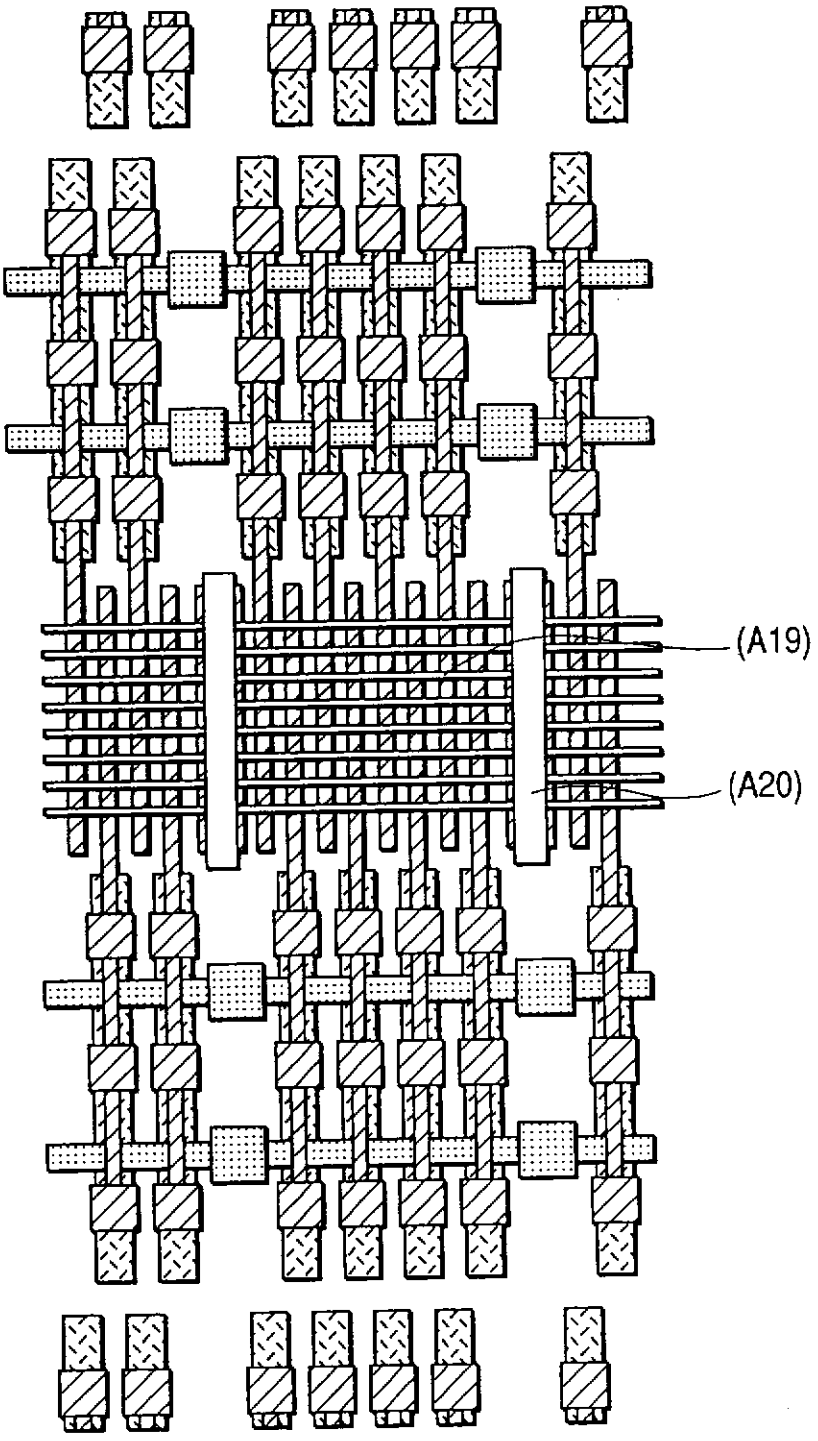


FIG. 24

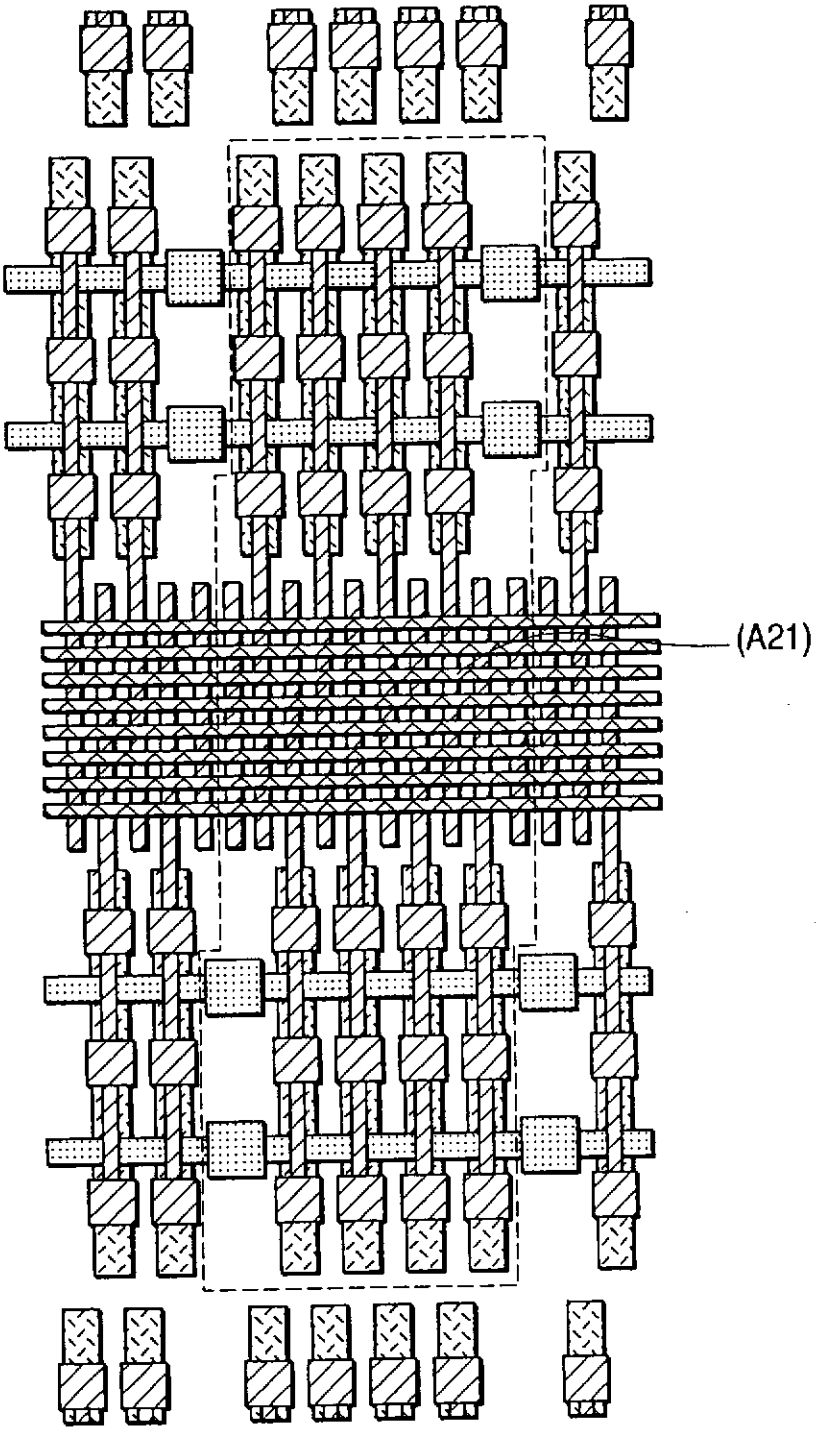


FIG. 25

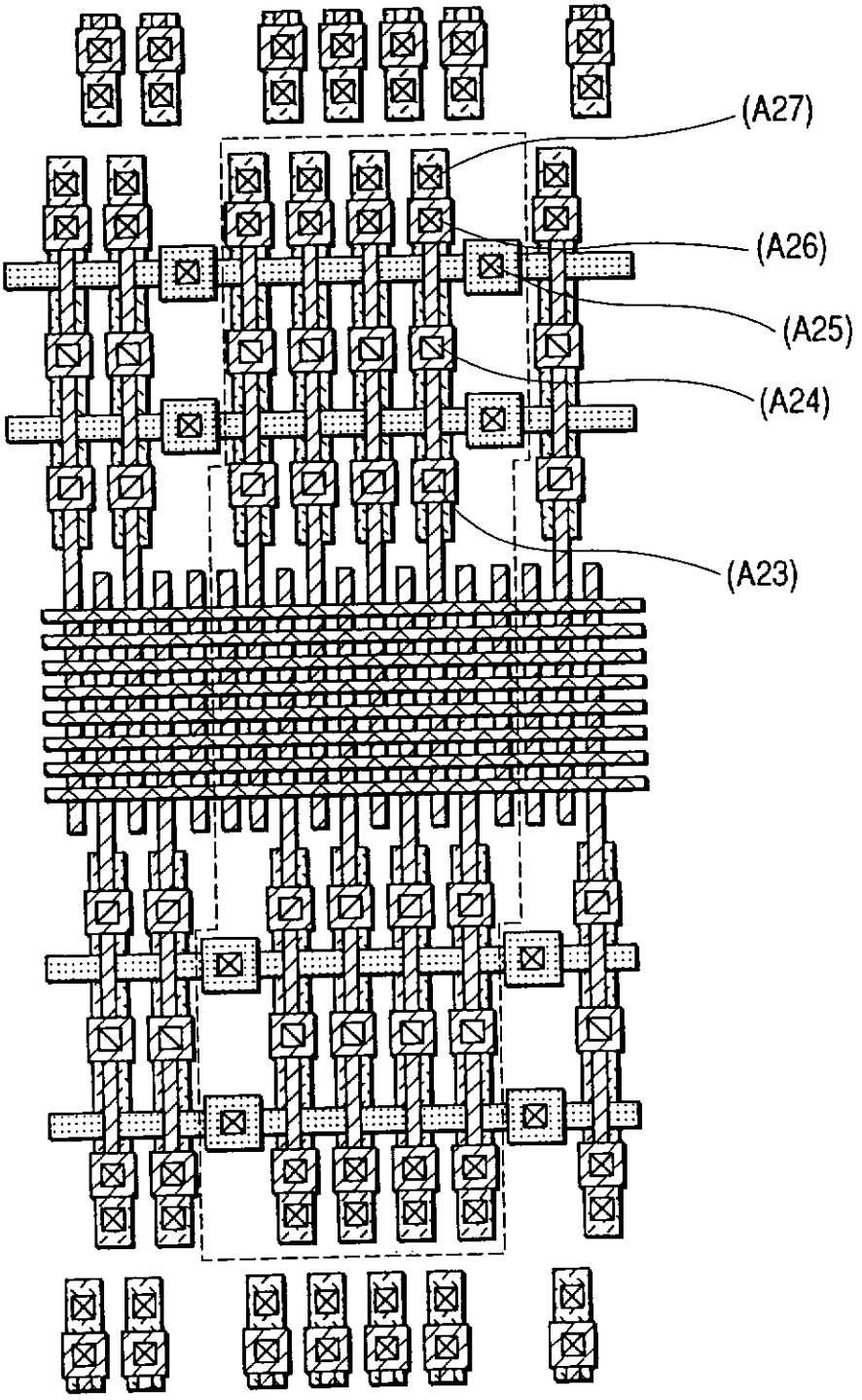


FIG. 26

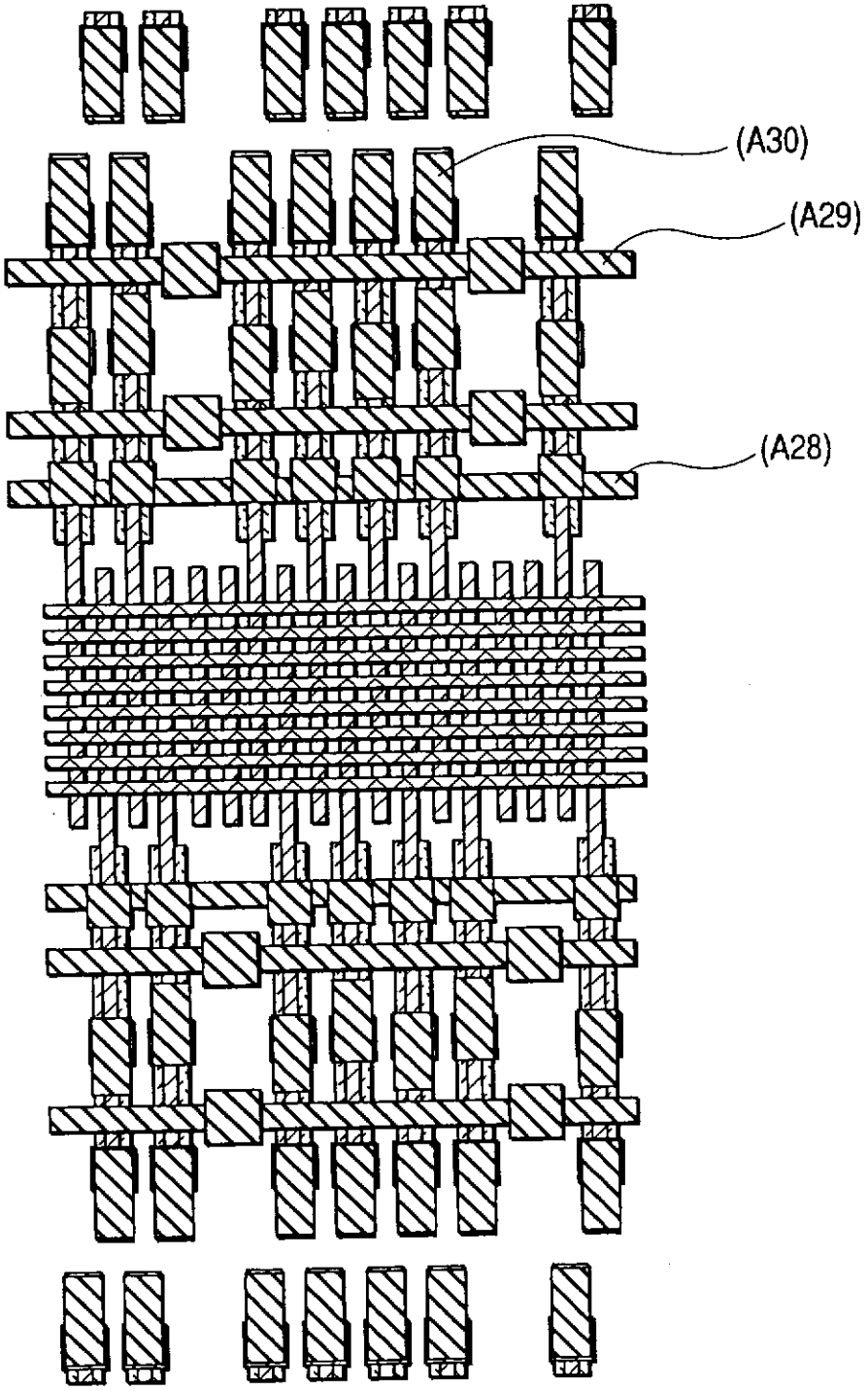


FIG. 27

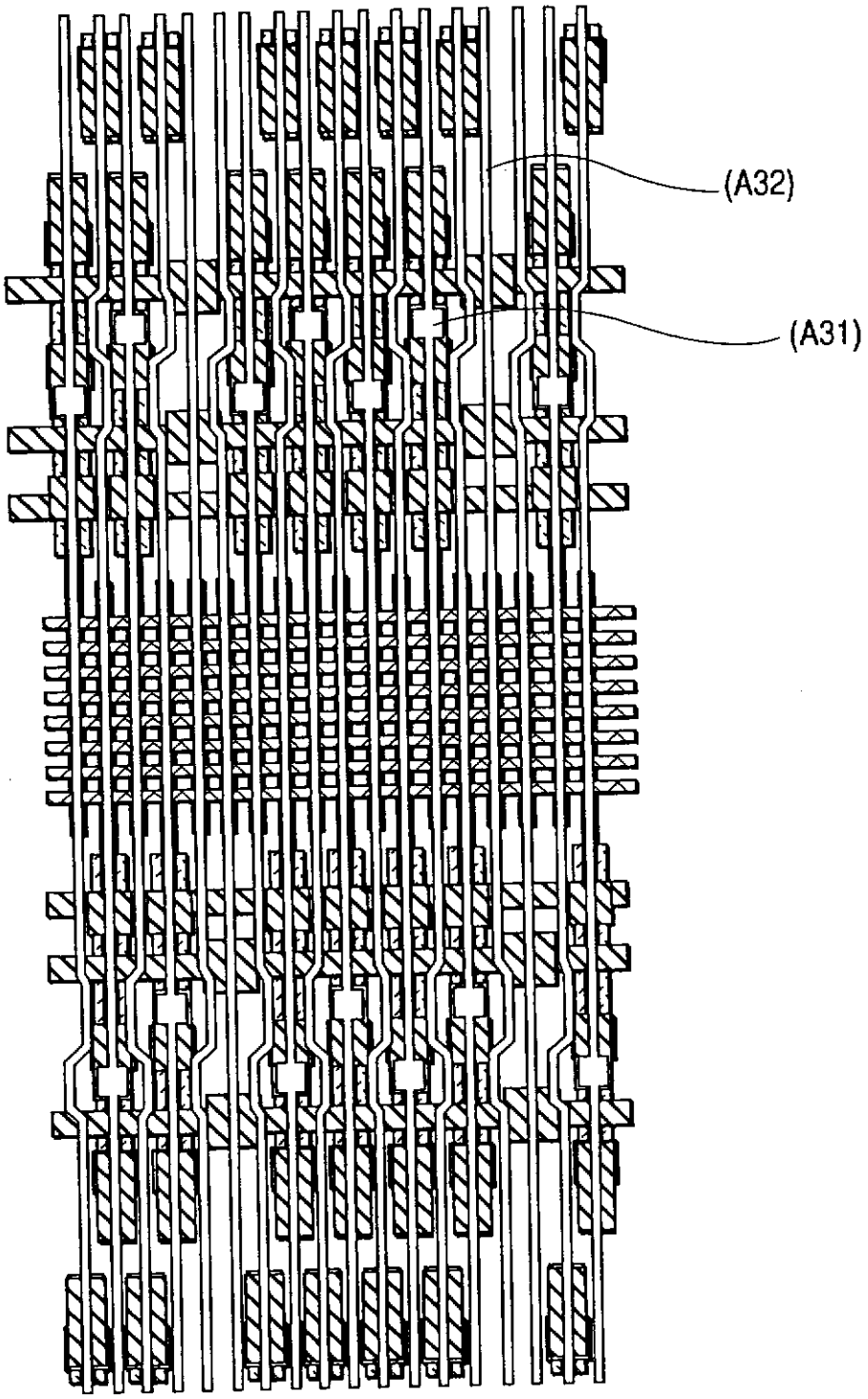


FIG. 28

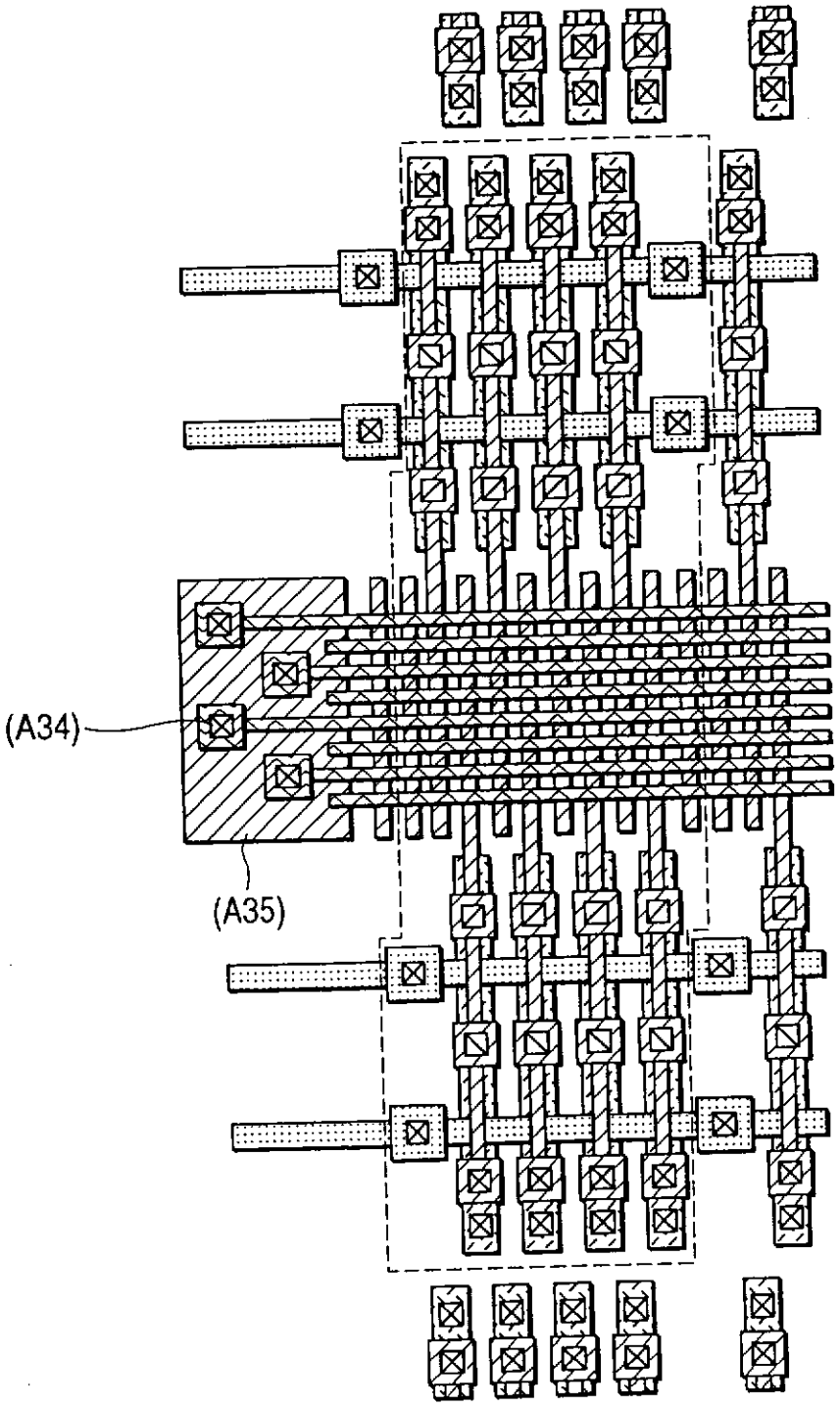


FIG. 29

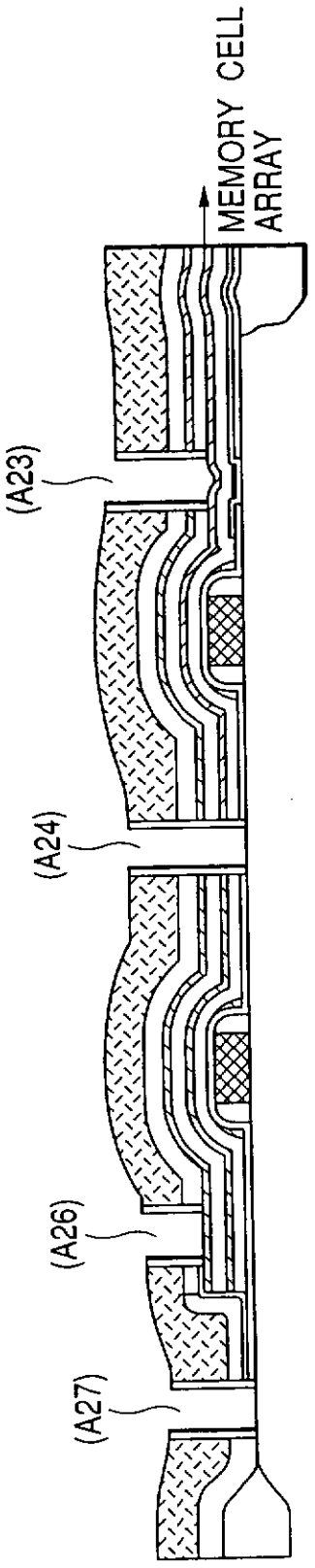


FIG. 30(a)

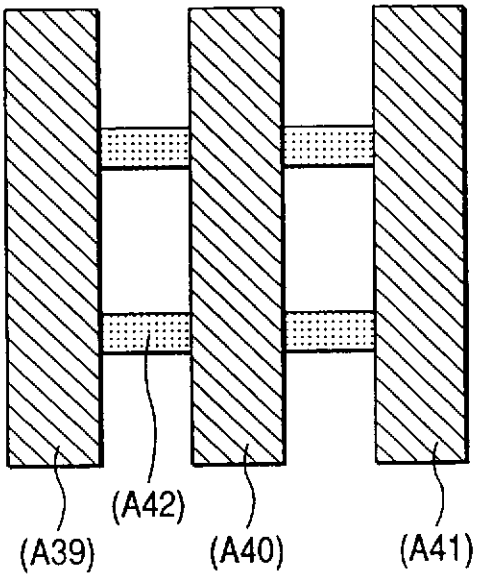


FIG. 30(b)

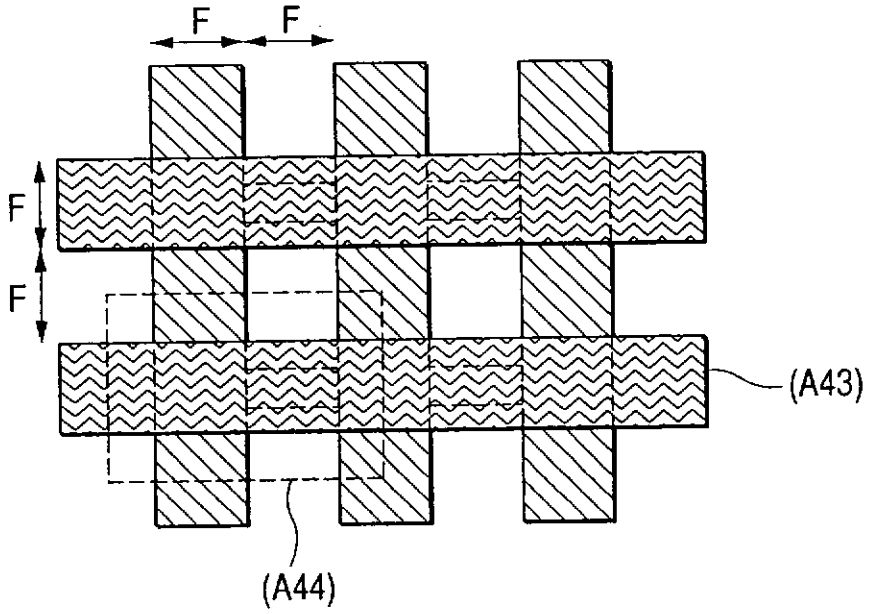


FIG. 31(a)

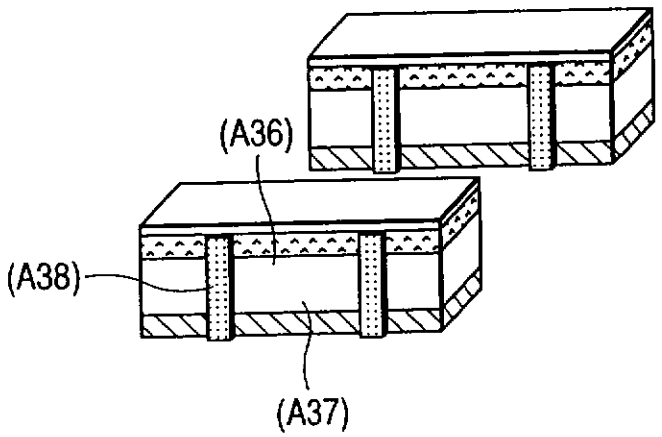


FIG. 31(b)

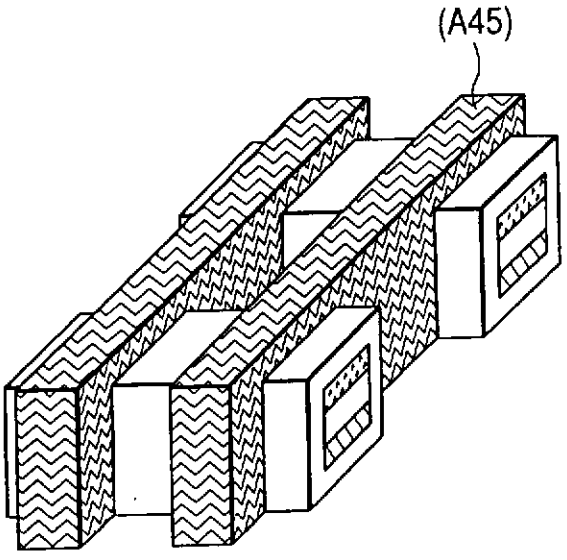


FIG. 32

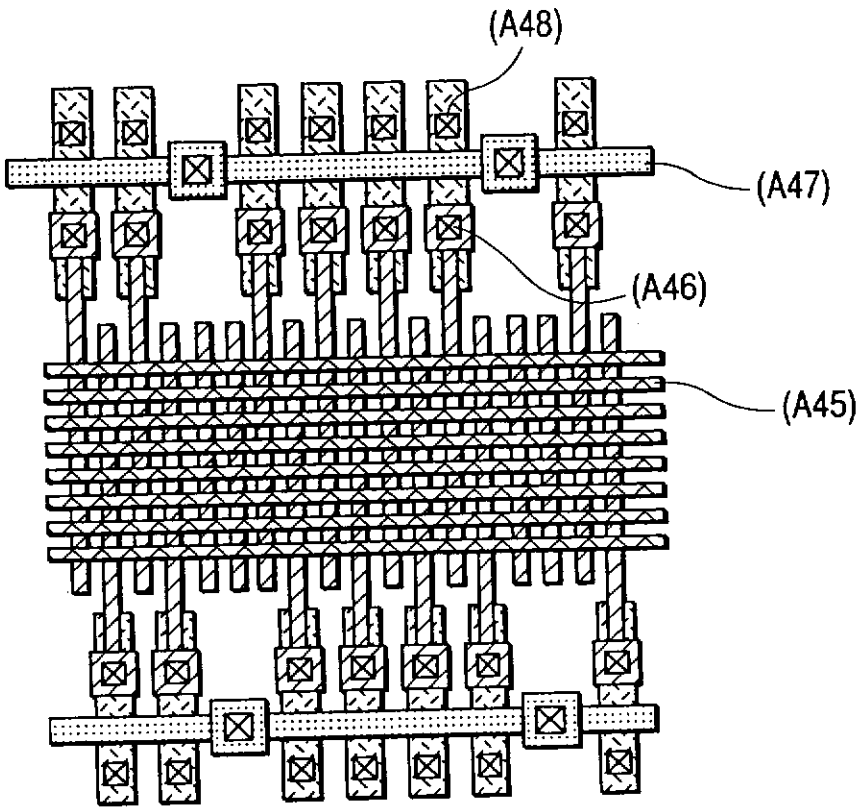

(A48)
(A47)
(A46)
(A45)

FIG. 33

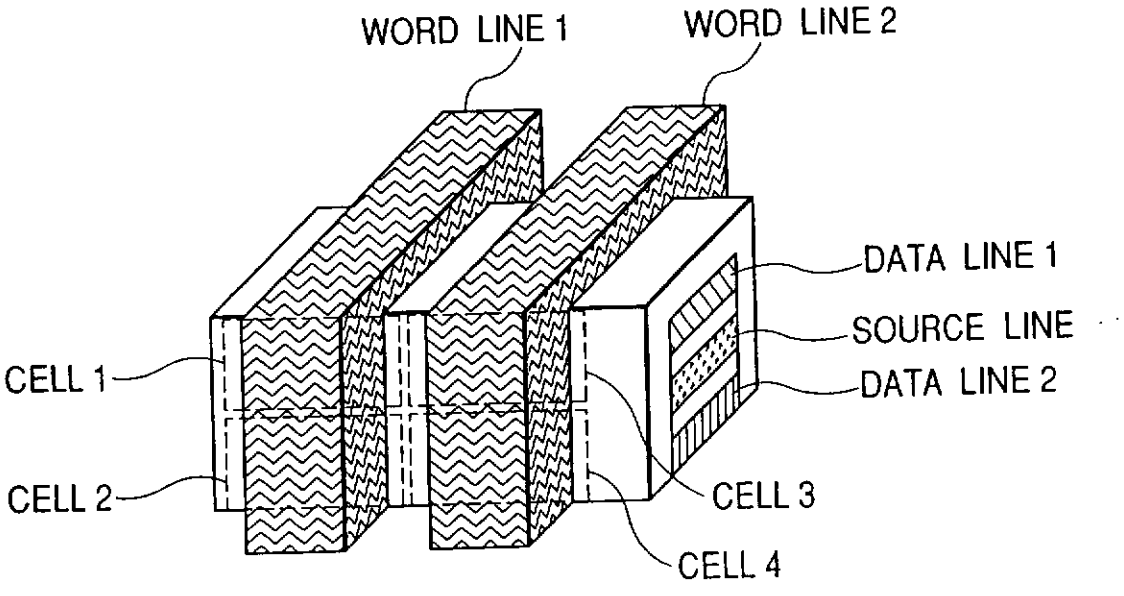
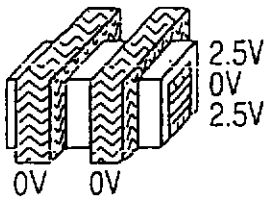
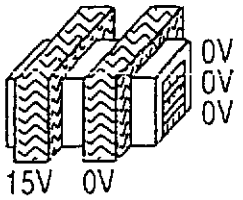
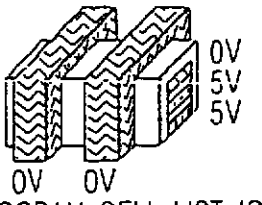
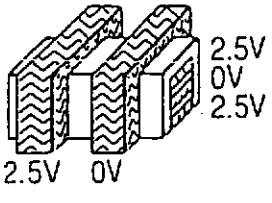
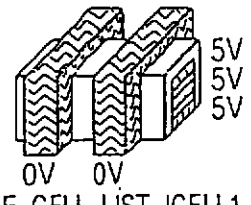
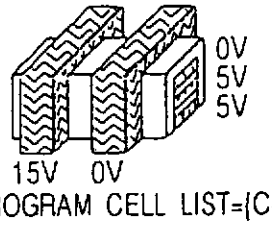
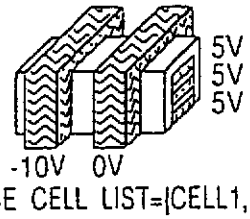
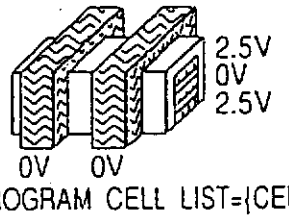
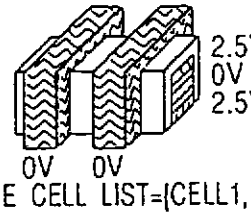
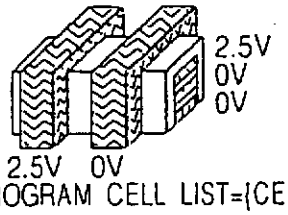
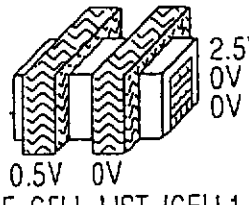

WORD LINE 1
WORD LINE 2
DATA LINE 1
SOURCE LINE
DATA LINE 2
CELL 1
CELL 2
CELL 3
CELL 4

FIG. 34

STEP FLOW

READ OUT SEQUENCE	ERASE SEQUENCE	PROGRAM SEQUENCE
STEP1 : PRE CHARGE  	STEP1 : PRE WRITE  ERASE CELL LIST={CELL1, CELL2}	STEP1 : DATA INPUT  "1"PROGRAM CELL LIST={CELL1}
STEP2 : READ/SENSE  	STEP2 : READ DATA  ERASE CELL LIST={CELL1, CELL2}	STEP2 : PROGRAM  "1"PROGRAM CELL LIST={CELL1}
	STEP3 : ERASE  ERASE CELL LIST={CELL1, CELL2}	STEP3 : PRE CHARGE  "1"PROGRAM CELL LIST={CELL1}
	STEP4 : PRE CHARGE  ERASE CELL LIST={CELL1, CELL2}	STEP4 : PROGRAM VERIFY  "1"PROGRAM CELL LIST={CELL1}

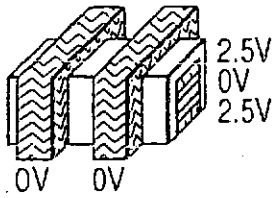
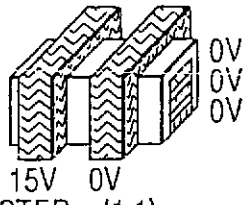
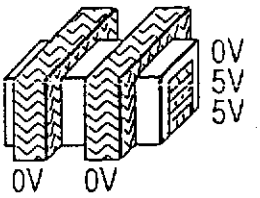
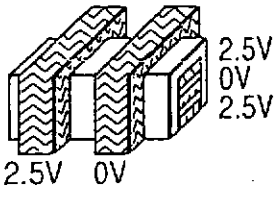
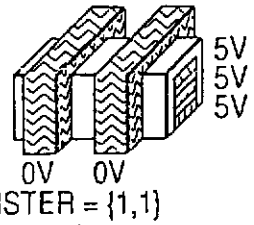
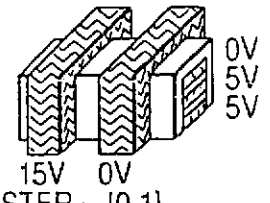
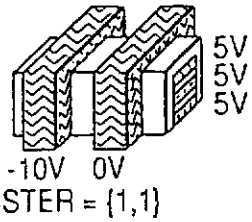
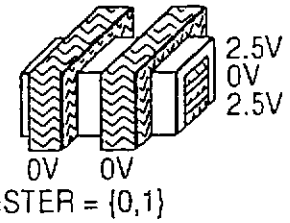
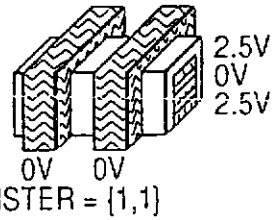
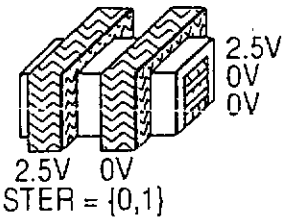
	STEP5 : ERASE VERIFY
	
	
	ERASE CELL LIST={CELL1, CELL2}
		STEP5 : CELL LIST RENEWAL
		if(data line voltage high)
		then
		delete from cell list
		end if
		"1" PROGRAM CELL LIST={CELL1}
		("1" PROGRAMMING FOR CELL 1 IS FINISHED)

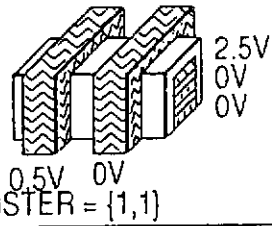
	STEP6 : CELL LIST RENEWAL
	if(voltage of data line low)
	then
	delete from cell list
	end if
	ERASE CELL LIST={CELL1}
	(CELL 2 DELETED, CELL 1 UNDELETED)
		STEP6 : DETERMINATION OF ENDING PROGRAMMING LOOP
		if(cell list empty)
		then loop end
		else return to step 1
		end if

	STEP7 : DETERMINATION OF ENDING ERASE LOOP
	if(cell list empty)
	then loop end
	else return to step 2
	end if	

FIG. 35

STEP FLOW

READ OUT SEQUENCE	ERASE SEQUENCE	PROGRAM SEQUENCE
STEP1 : PRE CHARGE  	STEP1 : PRE WRITE  REGISTER = {1,1}	STEP1 : DATA INPUT  REGISTER = {0,1}
STEP2 : READ/SENSE  	STEP2 : READ DATA  REGISTER = {1,1}	 REGISTER = {0,1}
	STEP3 : ERASE  REGISTER = {1,1}	STEP3 : PRE CHARGE  REGISTER = {0,1}
	STEP4 : PRE CHARGE  REGISTER = {1,1}	STEP4 : PROGRAM VERIFY  REGISTER = {0,1}

	
	
	REGISTER = {1,1}
		STEP5 : CELL LIST RENEWAL
		if(data line voltage high)
		&(register bit "0")
		then
		substitute "1" for register bit
		end if
		REGISTER = {1,1}
		(PROGRAMMING FOR CELL 1 IS FINISHED)

	STEP6 : CELL LIST RENEWAL
	if(data line voltage low)
	&(register bit "1")
	then
	substitute "0" for register bit
	end if
	REGISTER = {1,0}
	(CELL 2 DELETED, CELL 1 UNDELETED)
		STEP6 : DETERMINATION OF ENDING PROGRAMMING LOOP
		if(all of the register bits "1")
		then loop end
		else return to step 1
		end if

	STEP7 : DETERMINATION OF ENDING ERASE LOOP
	if(all of the register bits "0")
	then loop end
	else return to step 2
	end if	

FIG. 36

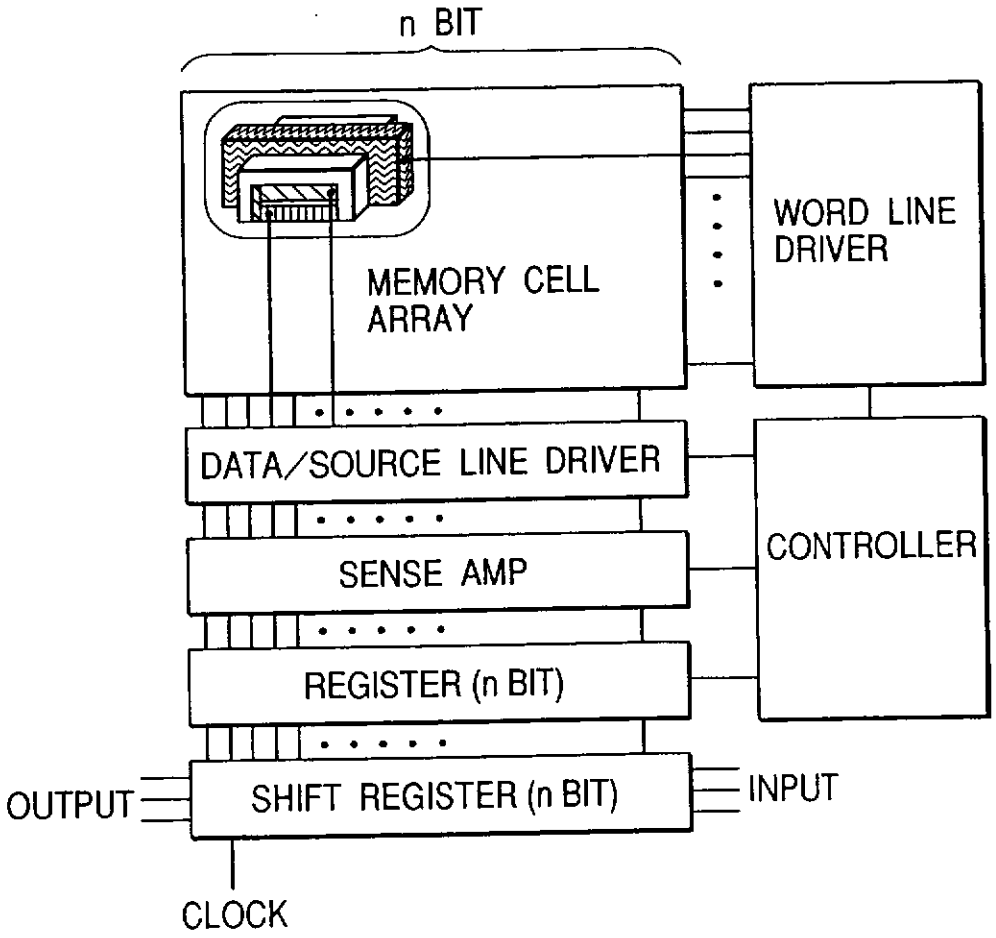
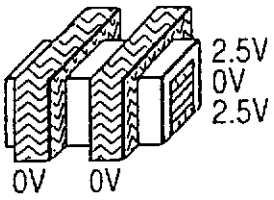
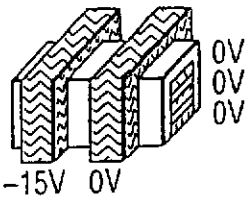
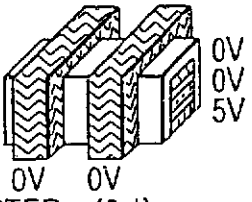
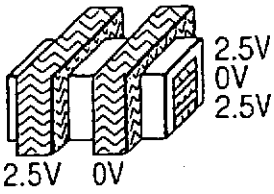
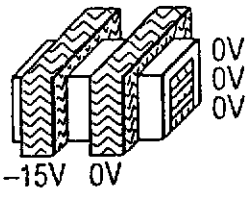
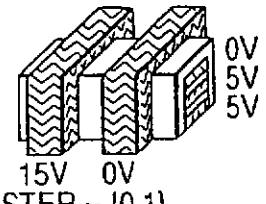
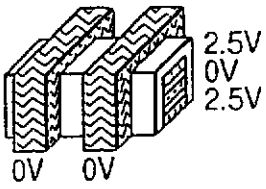
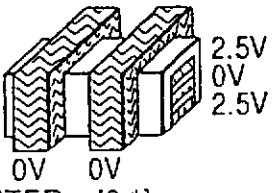
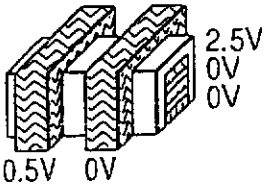
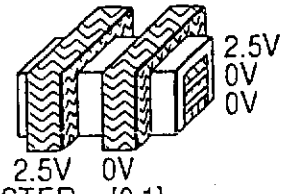

n BIT
MEMORY CELL ARRAY
WORD LINE DRIVER
DATA/SOURCE LINE DRIVER
SENSE AMP
REGISTER (n BIT)
CONTROLLER
OUTPUT
SHIFT REGISTER (n BIT)
INPUT
CLOCK

FIG. 37

STEP FLOW

READ OUT SEQUENCE	ERASE SEQUENCE	PROGRAM SEQUENCE
STEP1 : PRE CHARGE  	STEP1 : PRE WRITE  	STEP1 : DATA INPUT  REGISTER = {0,1}
STEP2 : READ/SENSE  	STEP2 : ERASE  	STEP2 : PROGRAM  REGISTER = {0,1}
	STEP4 : PRE CHARGE  	STEP3 : PRE CHARGE  REGISTER = {0,1}
	STEP5 : ERASE VERIFY  	STEP4 : PROGRAM VERIFY  REGISTER = {0,1}

	STEP6 : DETERMINATION OF ENDING ERASE LOOP
	if(all of the voltage of the data lines are high)
	then loop end
	else return to step 2
	end if
		STEP5 : CELL LIST RENEWAL
		if(data line voltage high)
		&(register bit "0")
		then
		substitute "1" for register bit
		end if
		REGISTER = {1,1}
		(PROGRAMMING FOR CELL 1 IS FINISHED)

		STEP6 : DETERMINATION OF ENDING PROGRAMMING LOOP
		if(all of the register bits "1")
		then loop end
		else return to step 1
		end if

FIG. 38

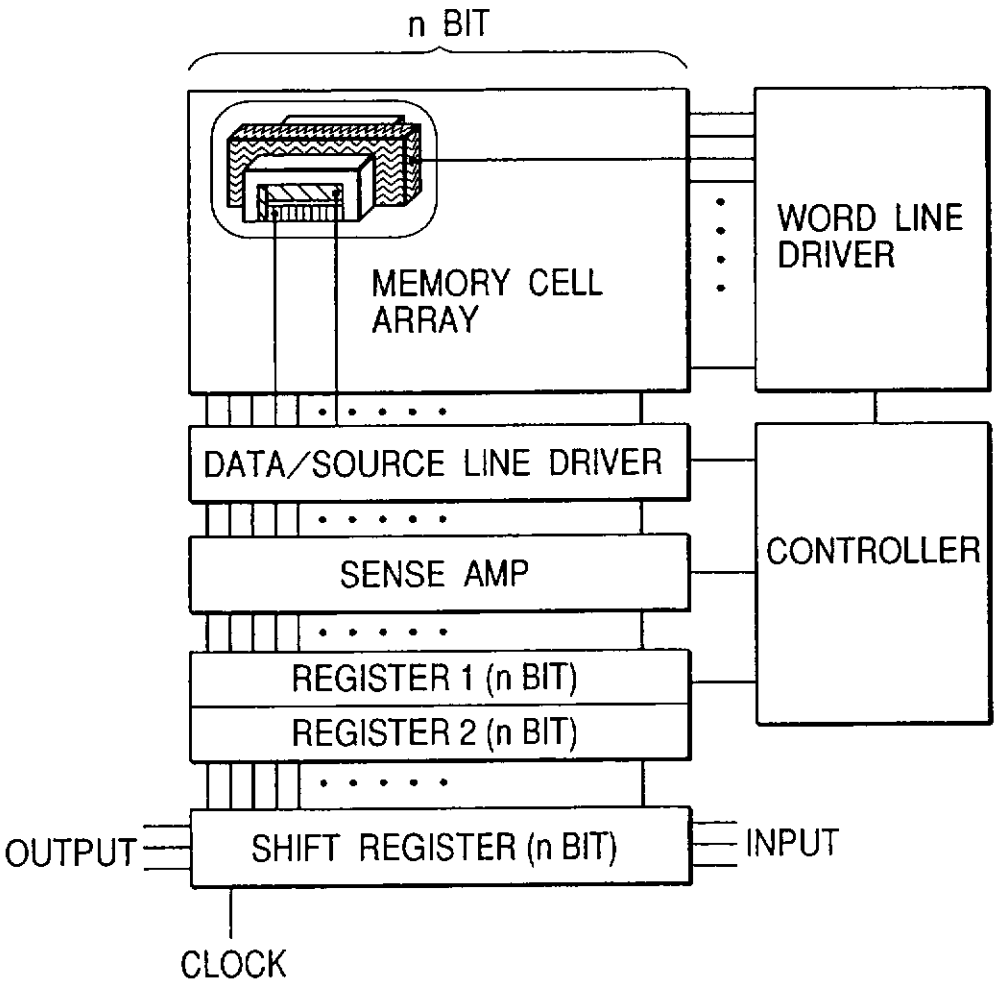
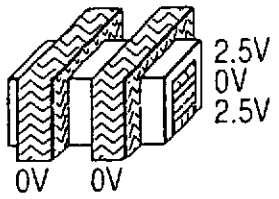
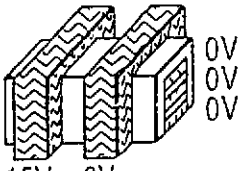
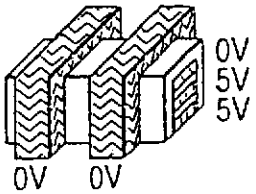
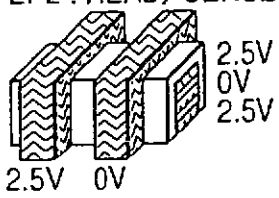
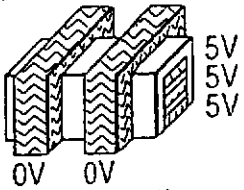
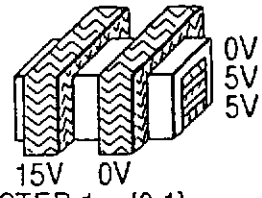
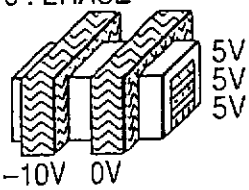
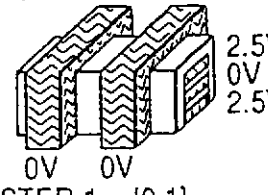
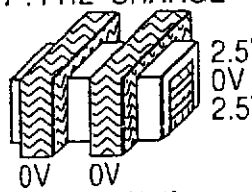
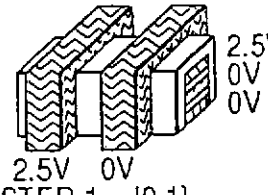

n BIT
MEMORY CELL ARRAY
WORD LINE DRIVER
DATA/SOURCE LINE DRIVER
SENSE AMP
REGISTER 1 (n BIT)
REGISTER 2 (n BIT)
CONTROLLER
OUTPUT
SHIFT REGISTER (n BIT)
INPUT
CLOCK

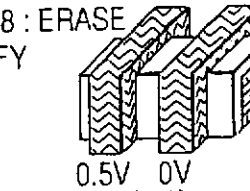
FIG. 39

STEP FLOW

REFRESH SEQUENCE		
READ OUT/HOLD	ERASE	REPROGRAM
STEP1 : PRE CHARGE  	STEP4 : PRE WRITE  REGISTER 2 = {1,1}	STEP11 : DATA INPUT  REGISTER 1 = {0,1}
STEP2 : READ/SENSE  	STEP5 : READ DATA  REGISTER 2 = {1,1}	STEP12 : PROGRAM  REGISTER 1 = {0,1}

STEP3 : HOLD DATA
if(data line voltage low)
substitute "1" for register1 bit
else
substitute "0" for register1 bit
end if
REGISTER 1 = {0,1}
{DATA OF CELL 1 "1",
DATA OF CELL 2 "0"}
	STEP6 : ERASE
	
	
	REGISTER 2 = {1,1}
		STEP13 : PRE CHARGE
		
		
		REGISTER 1 = {0,1}

	STEP7 : PRE CHARGE  REGISTER 2 = {1,1}	STEP14 : PROGRAM VERIFY  REGISTER 1 = {0,1}

	STEP8 : ERASE VERIFY
	
	
	REGISTER 2 = {1,1}
		STEP15 : CELL LIST RENEWAL
		if(data line voltage high)
		&(register1 bit "0")
		then
		substitute "1" for register1 bit
		end if
		REGISTER 1 = {1,1}
		(FINISHED PROGRAMMING CELL 1)

	STEP9 : CELL LIST RENEWAL
	if(data line voltage low)
	&(register2 bit "1")
	then
	substitute "0" for register2 bit
	end if
	REGISTER 2 = {1,0}
	(CELL 2 DELETED, CELL 1 UNDELETED)
		STEP16 : DETERMINATION OF ENDING REPROGRAM LOOP
		if{all of the register2 bits "1"}
		then next word line
		else return to step 11
		end if

	STEP10 : DETERMINATION OF ENDING ERASE LOOP if{all of the register2 bits "0"} then go to step 11 else return to step 5 end if	

FIG. 40

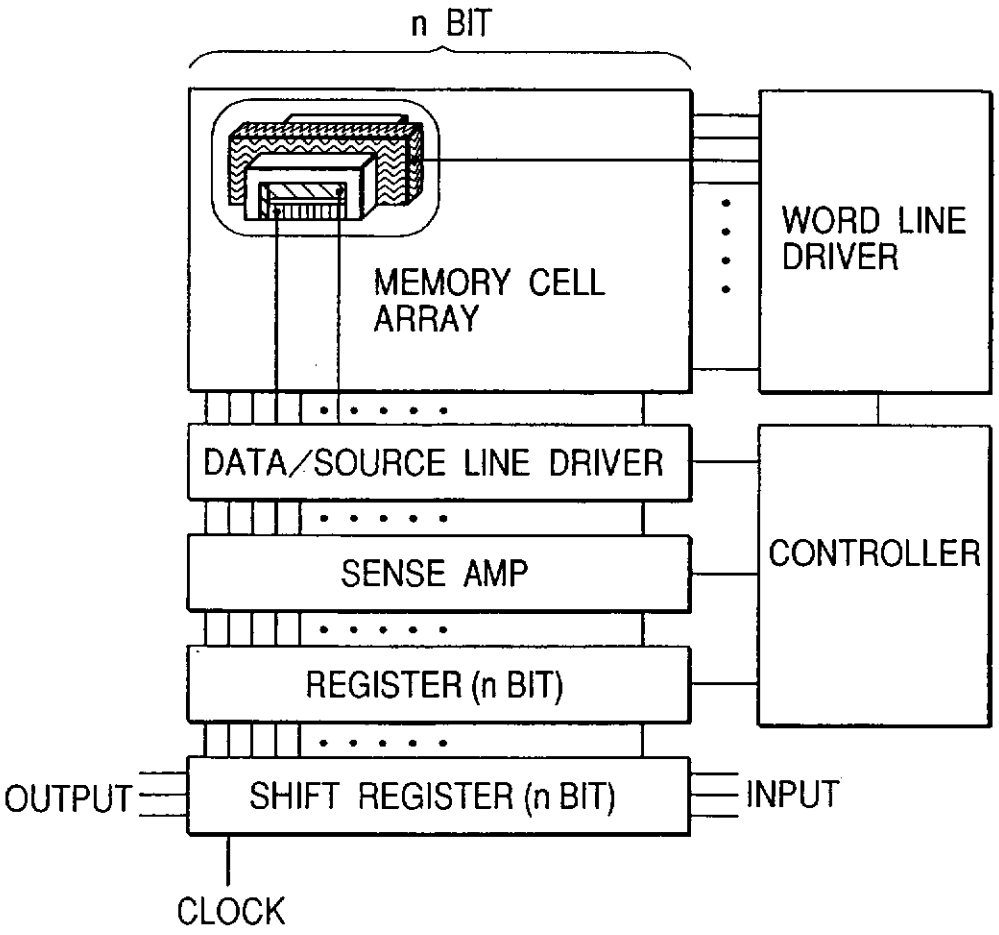
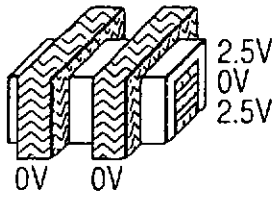
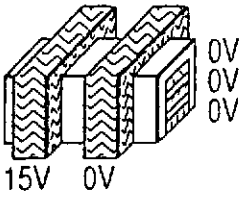
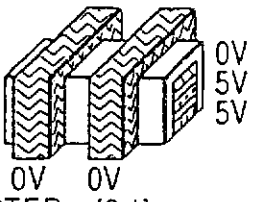
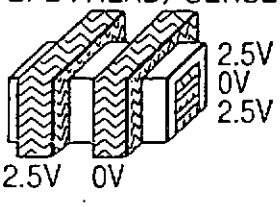
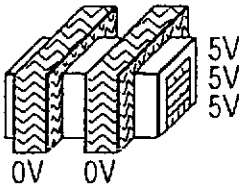
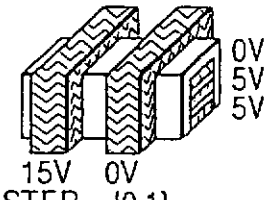
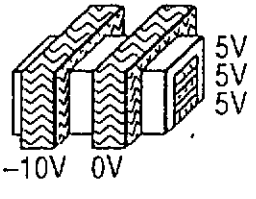
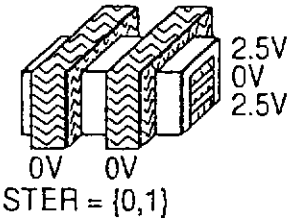
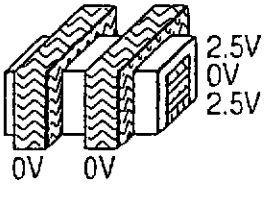
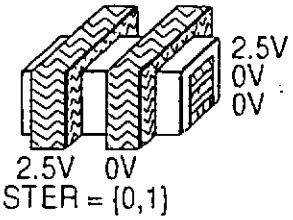

n BIT
MEMORY CELL ARRAY
WORD LINE DRIVER
DATA/SOURCE LINE DRIVER
SENSE AMP
REGISTER (n BIT)
CONTROLLER
OUTPUT
SHIFT REGISTER (n BIT)
INPUT
CLOCK

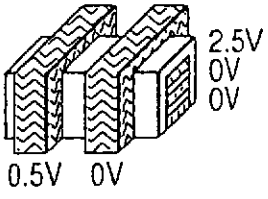
FIG. 41

STEP FLOW

REFRESH SEQUENCE		
READ OUT/HOLD	ERASE	REPROGRAM
STEP1 : PRE CHARGE  	STEP4 : PRE WRITE  	STEP10 : DATA INPUT  REGISTER = {0,1}
STEP2 : READ/SENSE  	STEP5 : READ DATA  	STEP11 : PROGRAM  REGISTER = {0,1}

STEP3 : HOLD DATA
if(data line voltage low)
substitute "1" for register bit
else
substitute "0" for register bit
end if
REGISTER = {0,1}
{DATA OF CELL 1 "1",
DATA OF CELL 2 "0"}
	STEP6 : ERASE
	
	
		STEP12 : PRE CHARGE
		
		
		REGISTER = {0,1}

	STEP7 : PRE CHARGE  	STEP13 : PROGRAM VERIFY  REGISTER = {0,1}

	STEP8 : ERASE VERIFY
	
	
		STEP14 : CELL LIST RENEWAL
		if(data line voltage high)
		&(register bit "0")
		then
		substitute "1" for register bit
		end if
		REGISTER = {1,1}
		(FINISHED PROGRAMMING CELL 1)

	STEP9 : DETERMINATION OF ENDING ERASE LOOP
	if(all of the voltage of the data lines are high)
	then go to step 10
	else return to step 5
	end if
		STEP15 : DETERMINATION OF ENDING REPROGRAM LOOP
		if(all of the register bits "1")
		then next word line
		else return to step 10
		end if

FIG. 42

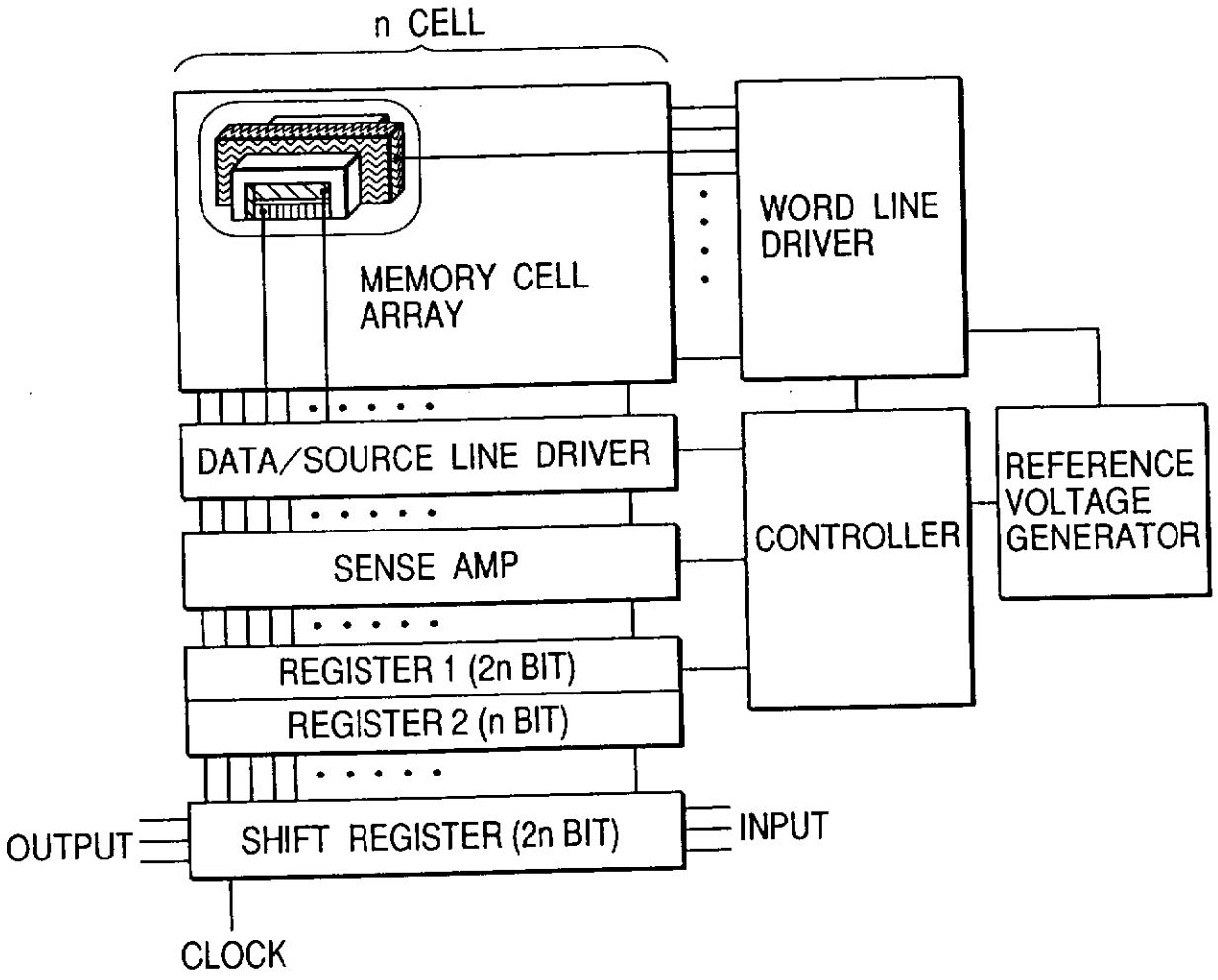

n CELL
MEMORY CELL ARRAY
WORD LINE DRIVER
DATA/SOURCE LINE DRIVER
SENSE AMP
REGISTER 1 (2n BIT)
REGISTER 2 (n BIT)
CONTROLLER
REFERENCE VOLTAGE GENERATOR
OUTPUT
SHIFT REGISTER (2n BIT)
INPUT
CLOCK

FIG. 43

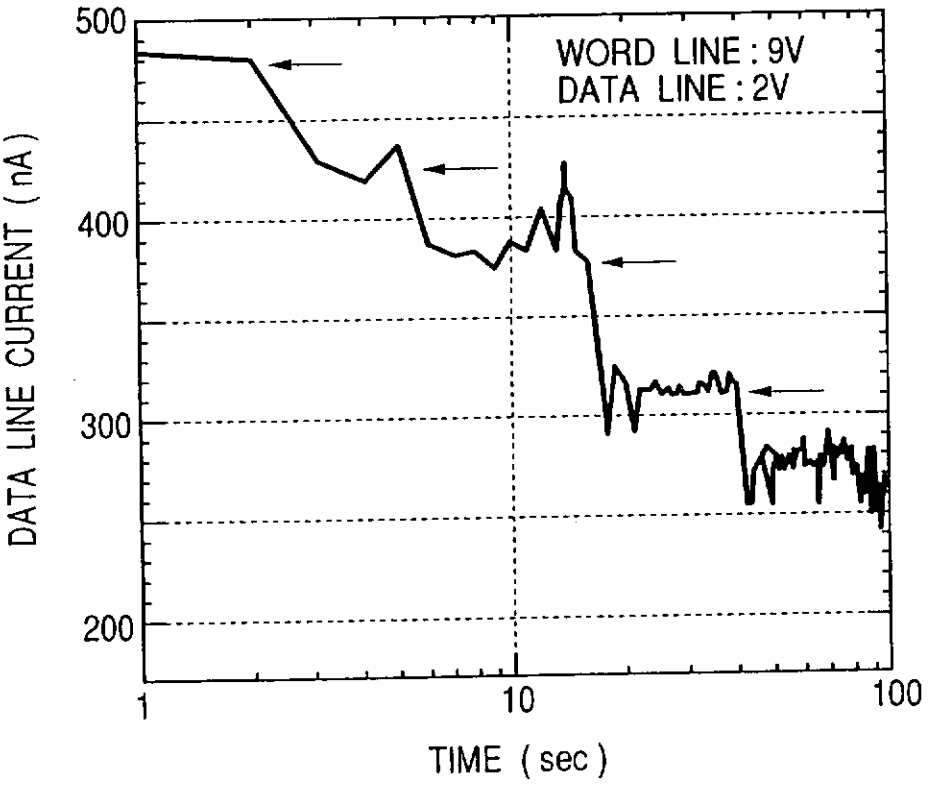
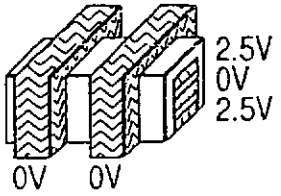
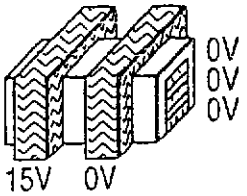
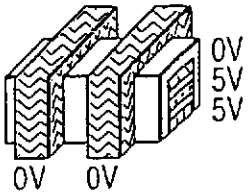
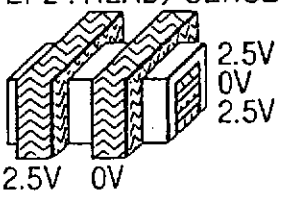
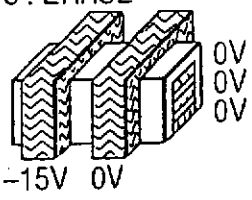
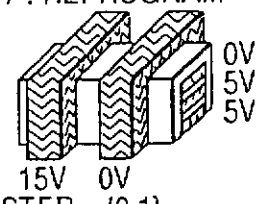

500
400
300
200
1
10
100
WORD LINE : 9V
DATA LINE : 2V
DATA LINE CURRENT (nA)
TIME (sec)

FIG. 44

FLOW

REFRESH SEQUENCE		
READ OUT/HOLD	ERASE	REPROGRAM
STEP1 : PRE CHARGE  	STEP4 : PRE WRITE  	STEP6 : DATA INPUT  REGISTER = {0,1}
STEP2 : READ/SENSE  	STEP5 : ERASE  	STEP7 : REPROGRAM  REGISTER = {0,1}

STEP3 : HOLD DATA
if(data line voltage low)
substitute "1" for register bit
else
substitute "0" for register bit
end if
REGISTER = {0,1}
{DATA OF CELL 1 "1",
DATA OF CELL 2 "0"}		

FIG. 45

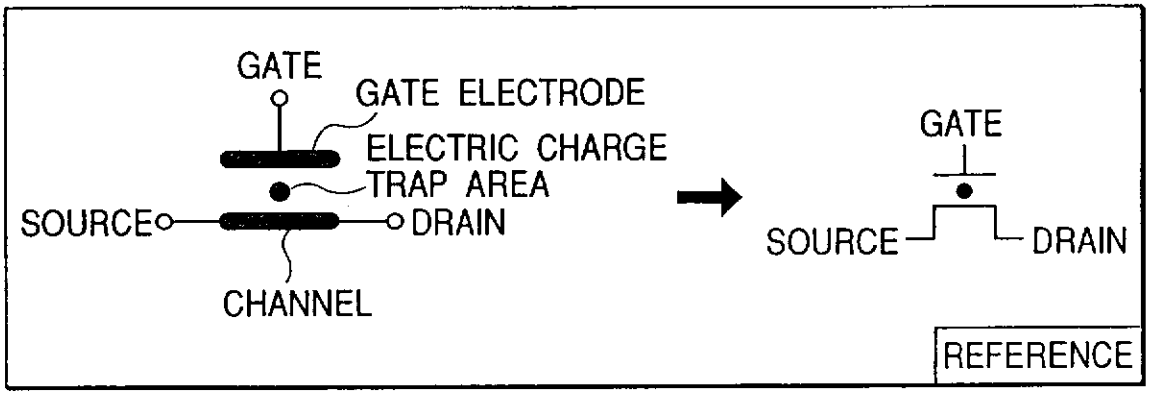

GATE
GATE ELECTRODE
ELECTRIC CHARGE
TRAP AREA
SOURCE
DRAIN
CHANNEL
GATE
SOURCE
DRAIN
REFERENCE

FIG. 46

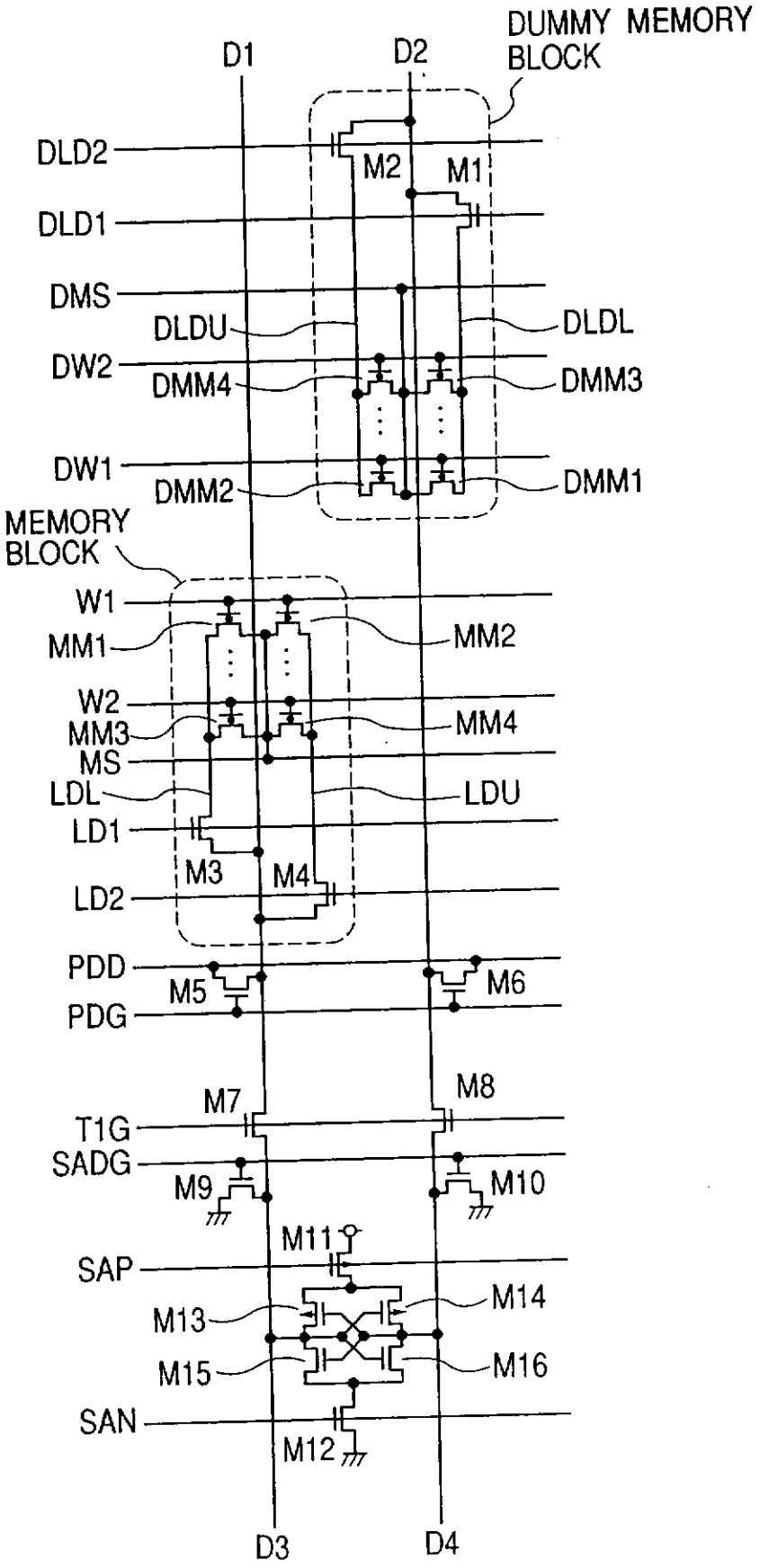
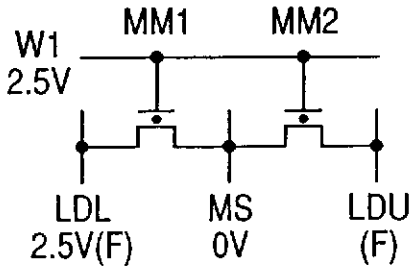
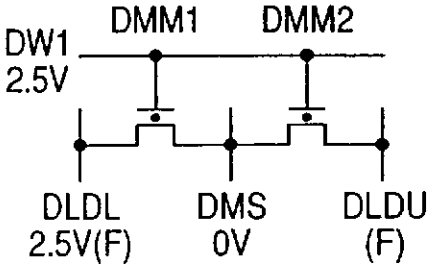
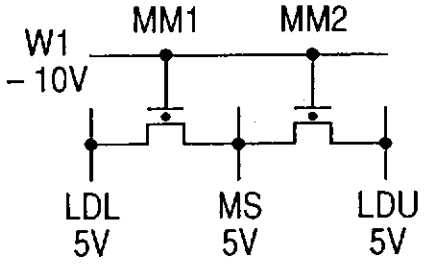
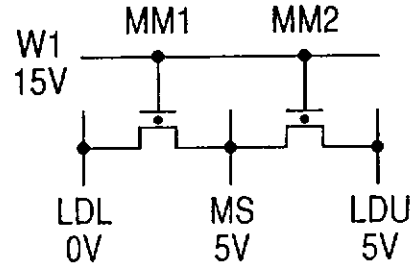
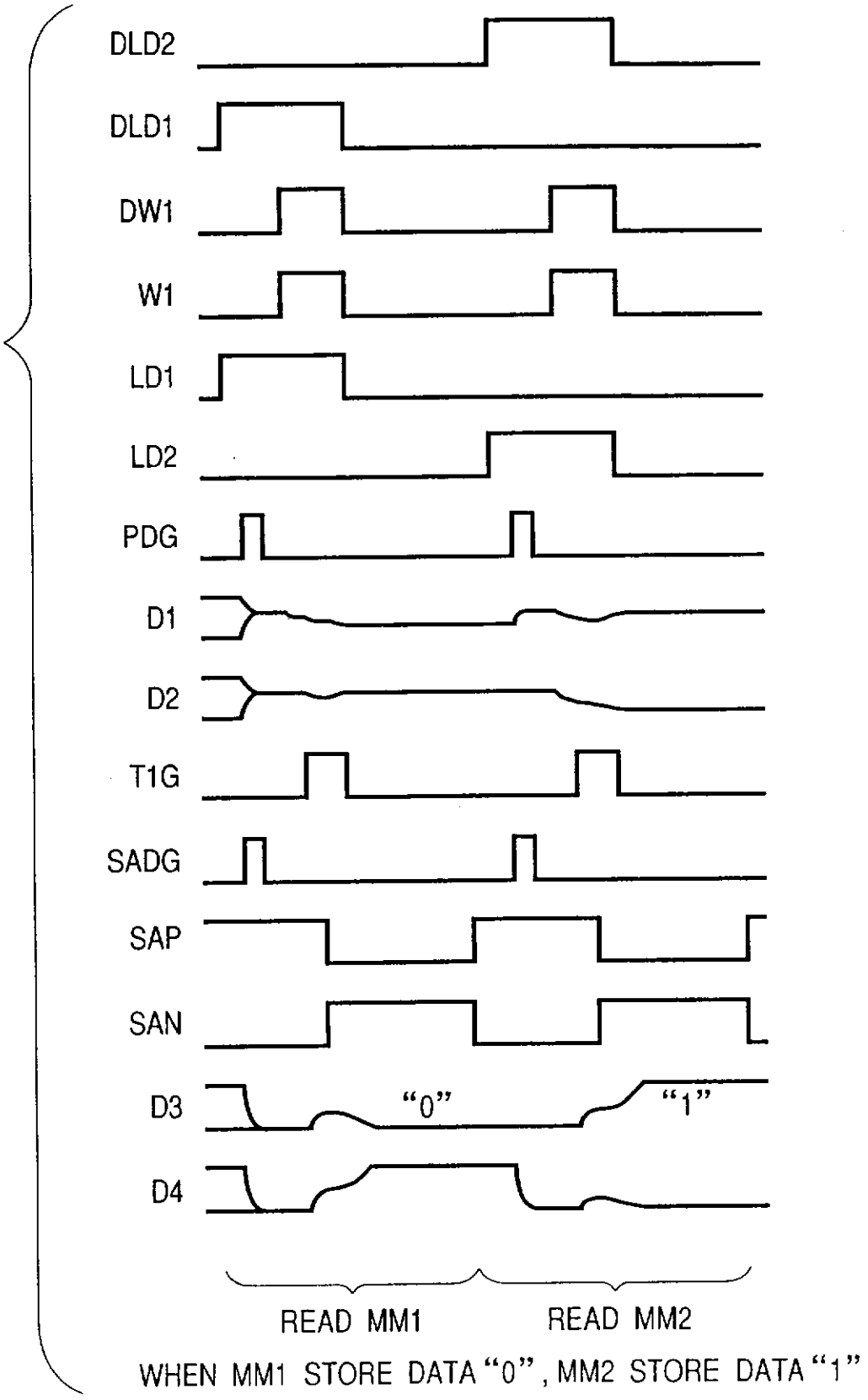

DUMMY MEMORY BLOCK
D1
D2
DLD2
M2
M1
DLD1
DMS
DLDU
DLDL
DW2
DMM4
DMM3
DW1
DMM2
DMM1
MEMORY BLOCK
W1
MM1
MM2
W2
MM3
MM4
MS
LDL
LDU
LD1
M3
M4
LD2
PDD
M5
M6
PDG
M7
M8
T1G
SADG
M9
M10
M11
SAP
M13
M14
M15
M16
SAN
M12
D3
D4

FIG. 47

	MEMORY CELL	DUMMY MEMORY CELL
READ OUT	 READ MM1	
ERASE		NO USE
PROGRAM	 MM1 WILL STORE "1" MM2 WILL STORE "0"	NO USE

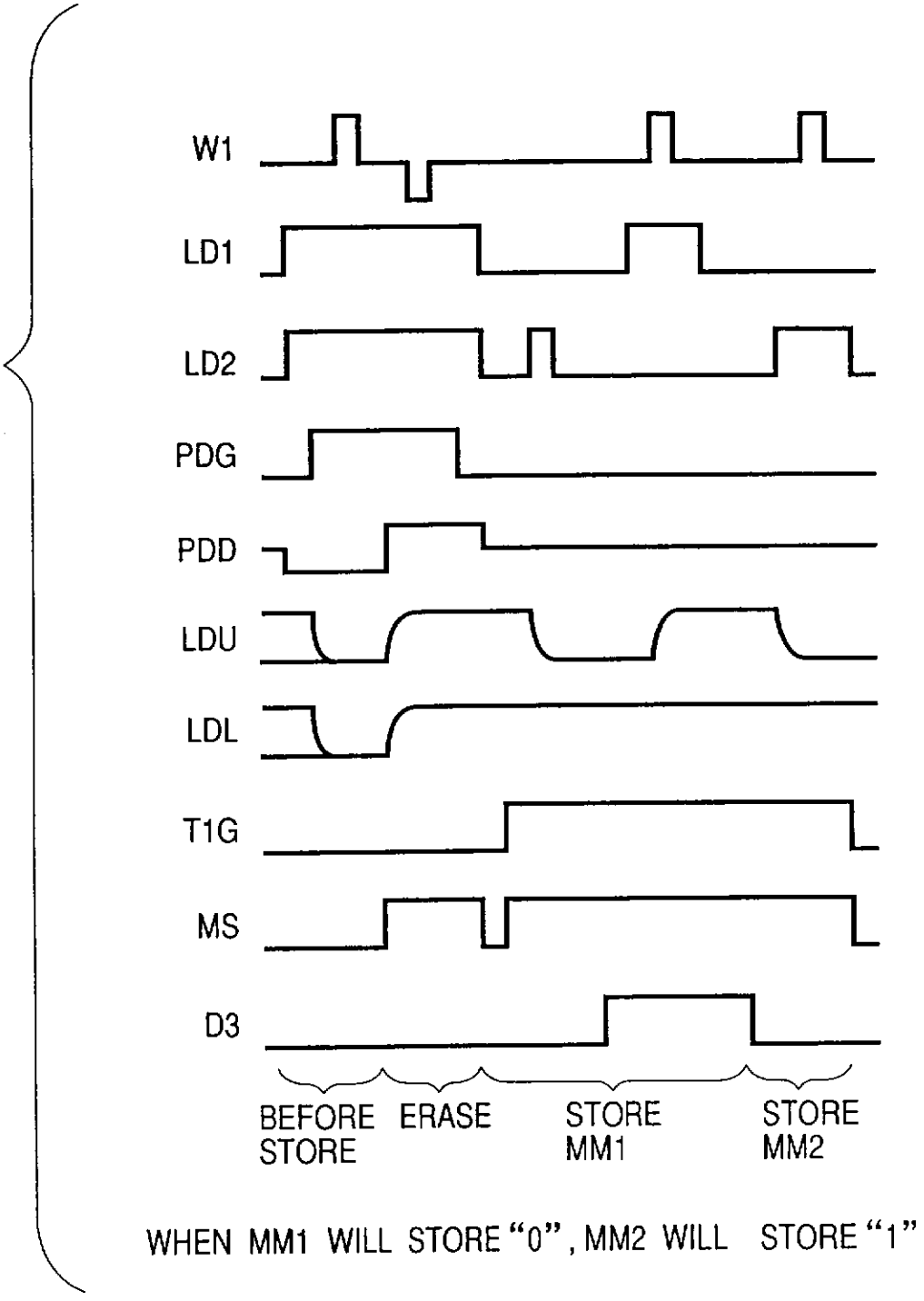
"F" INDICATES FLOATING

FIG. 48



WHEN MM1 STORE DATA "0", MM2 STORE DATA "1"

FIG. 49



WHEN MM1 WILL STORE "0", MM2 WILL STORE "1"

FIG. 50

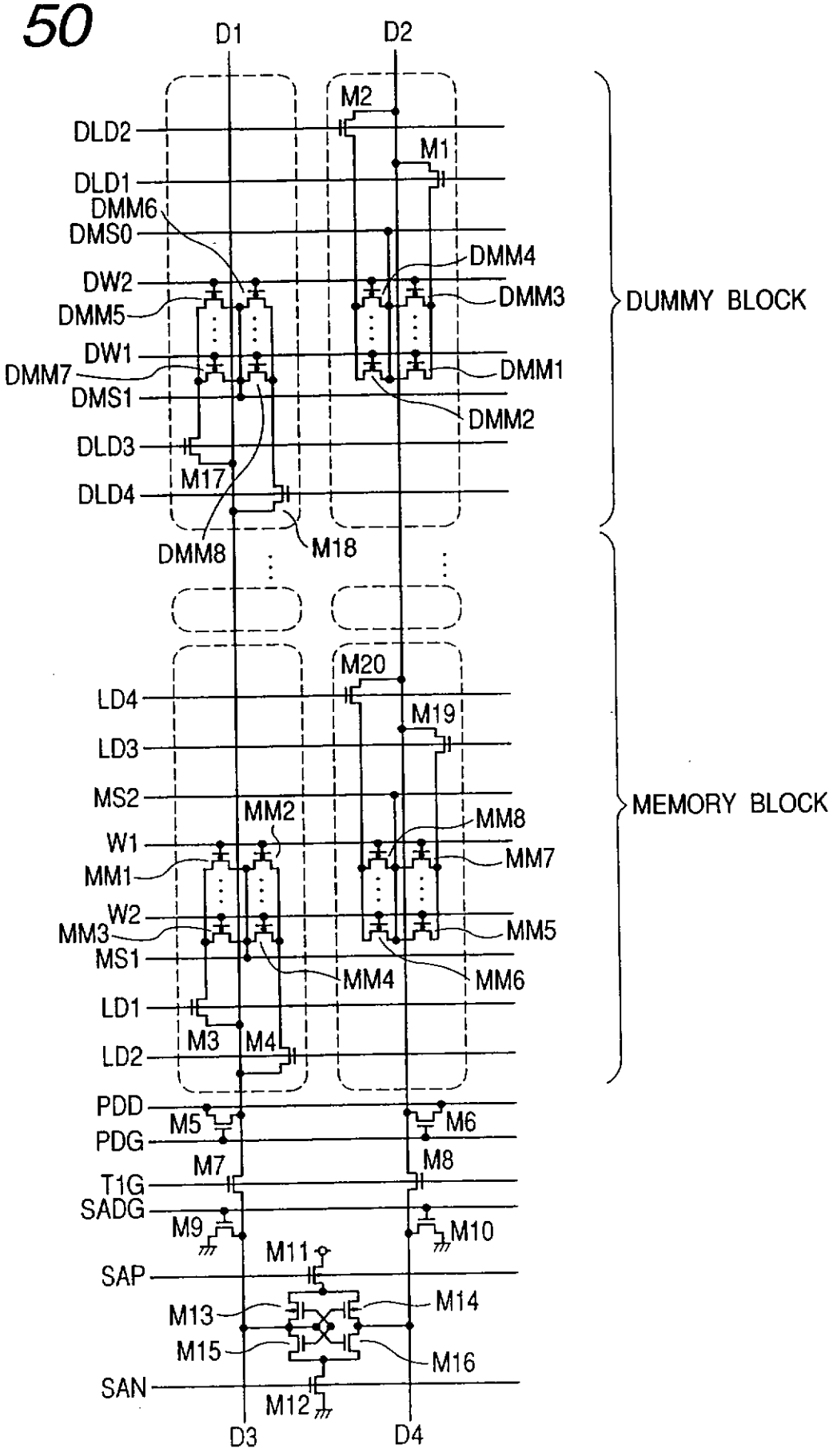

D1
D2
M2
M1
DLD2
DLD1
DMM6
DMS0
DW2
DMM5
DW1
DMM7
DMS1
DLD3
DLD4
M17
DMM8
M18
DMM4
DMM3
DMM1
DMM2
DUMMY BLOCK
M20
M19
LD4
LD3
MS2
W1
MM1
W2
MM3
MS1
LD1
LD2
M3
M4
MM2
MM4
MM8
MM7
MM5
MM6
MEMORY BLOCK
PDD
PDG
T1G
SADG
SAP
SAN
M5
M6
M7
M8
M9
M10
M11
M12
M13
M14
M15
M16
D3
D4

FIG. 51

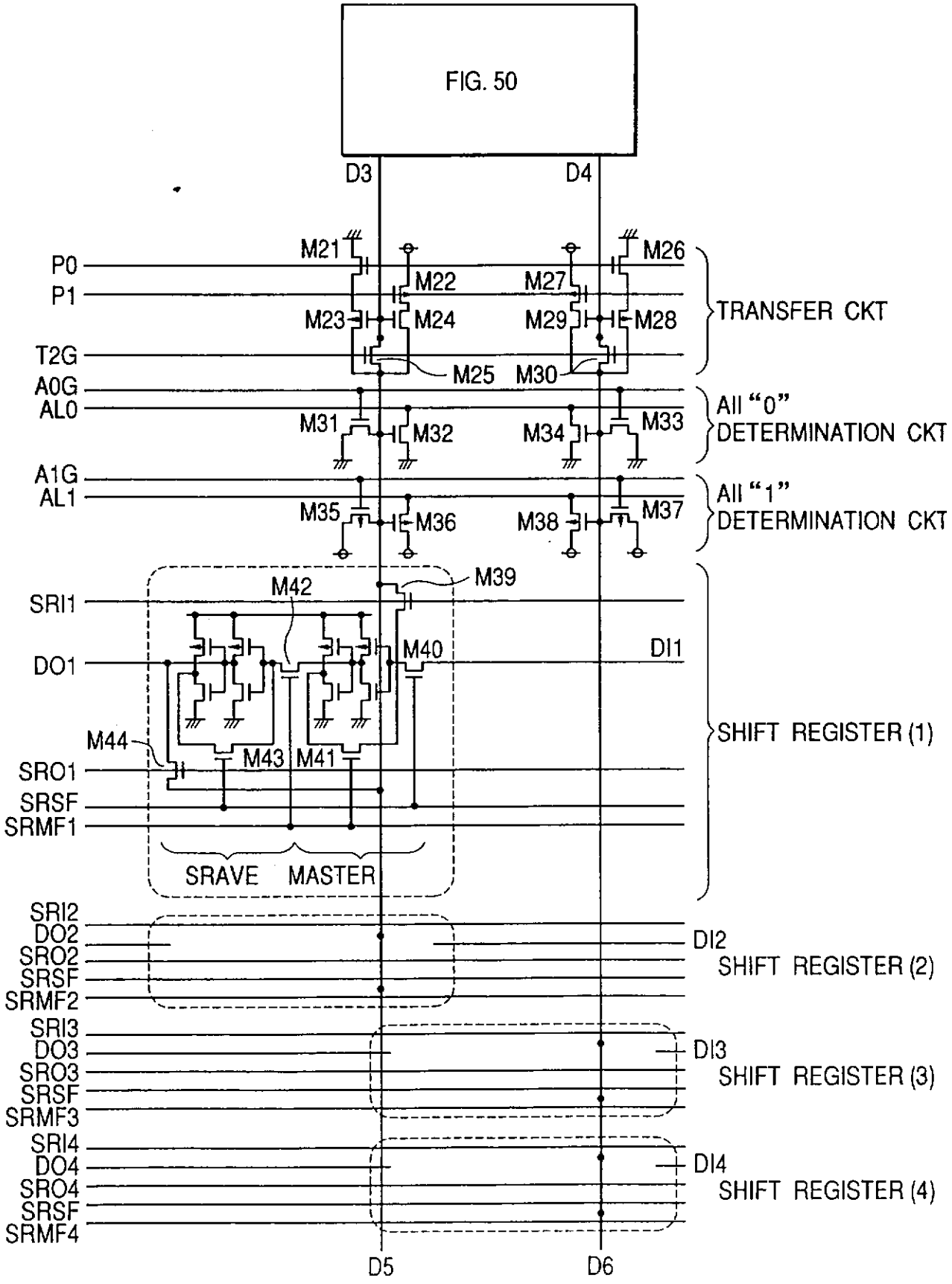

FIG. 50
D3
D4
M21
M22
M23
M24
M25
M26
M27
M28
M29
M30
P0
P1
T2G
TRANSFER CKT
A0G
AL0
M31
M32
M33
M34
All "0"
DETERMINATION CKT
A1G
AL1
M35
M36
M37
M38
All "1"
DETERMINATION CKT
SRI1
M42
M39
DO1
M40
DI1
M44
M43
M41
SRO1
SRSF
SRMF1
SRAVE
MASTER
SHIFT REGISTER (1)
SRI2
DO2
SRO2
SRSF
SRMF2
DI2
SHIFT REGISTER (2)
SRI3
DO3
SRO3
SRSF
SRMF3
DI3
SHIFT REGISTER (3)
SRI4
DO4
SRO4
SRSF
SRMF4
DI4
SHIFT REGISTER (4)
D5
D6

FIG. 52

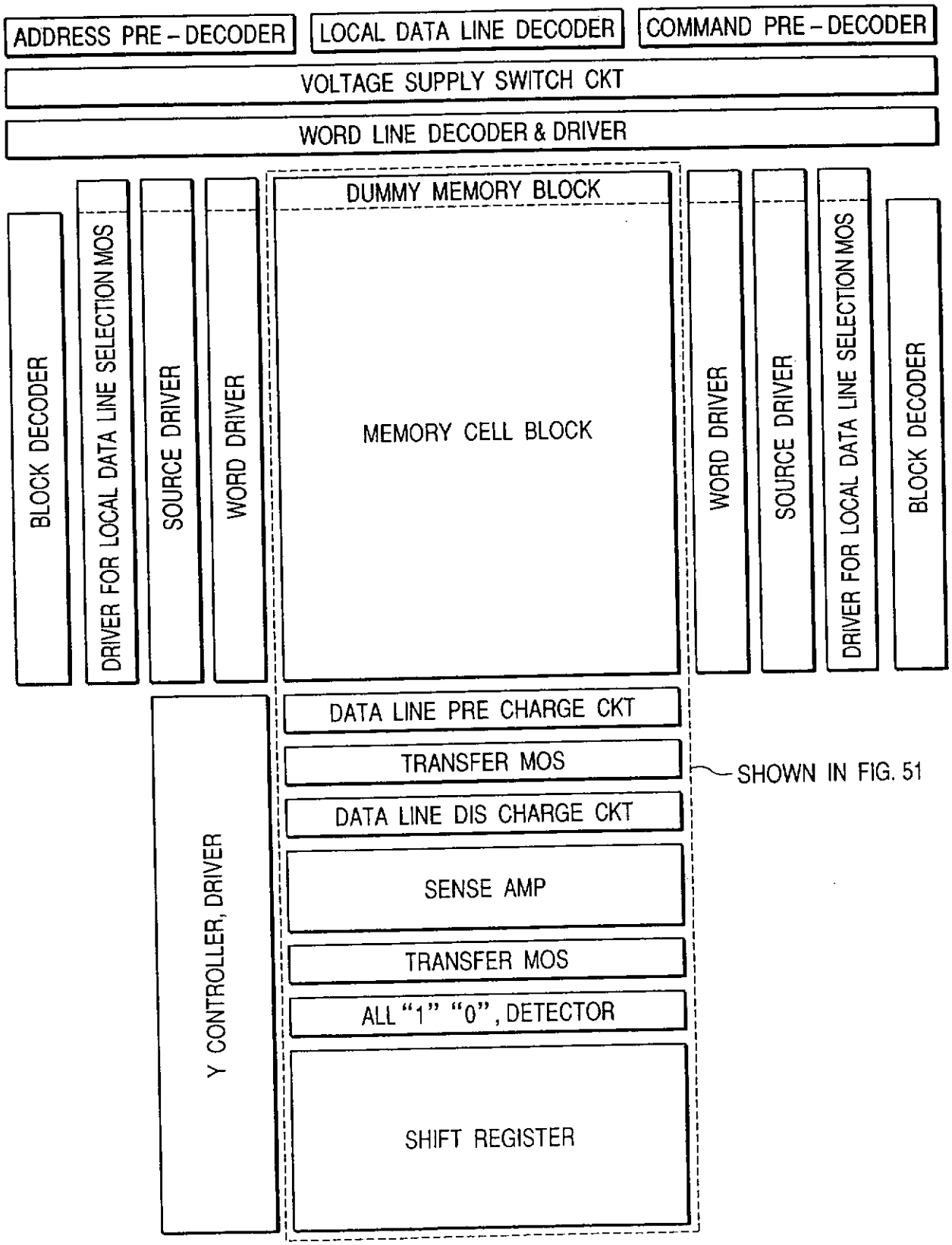
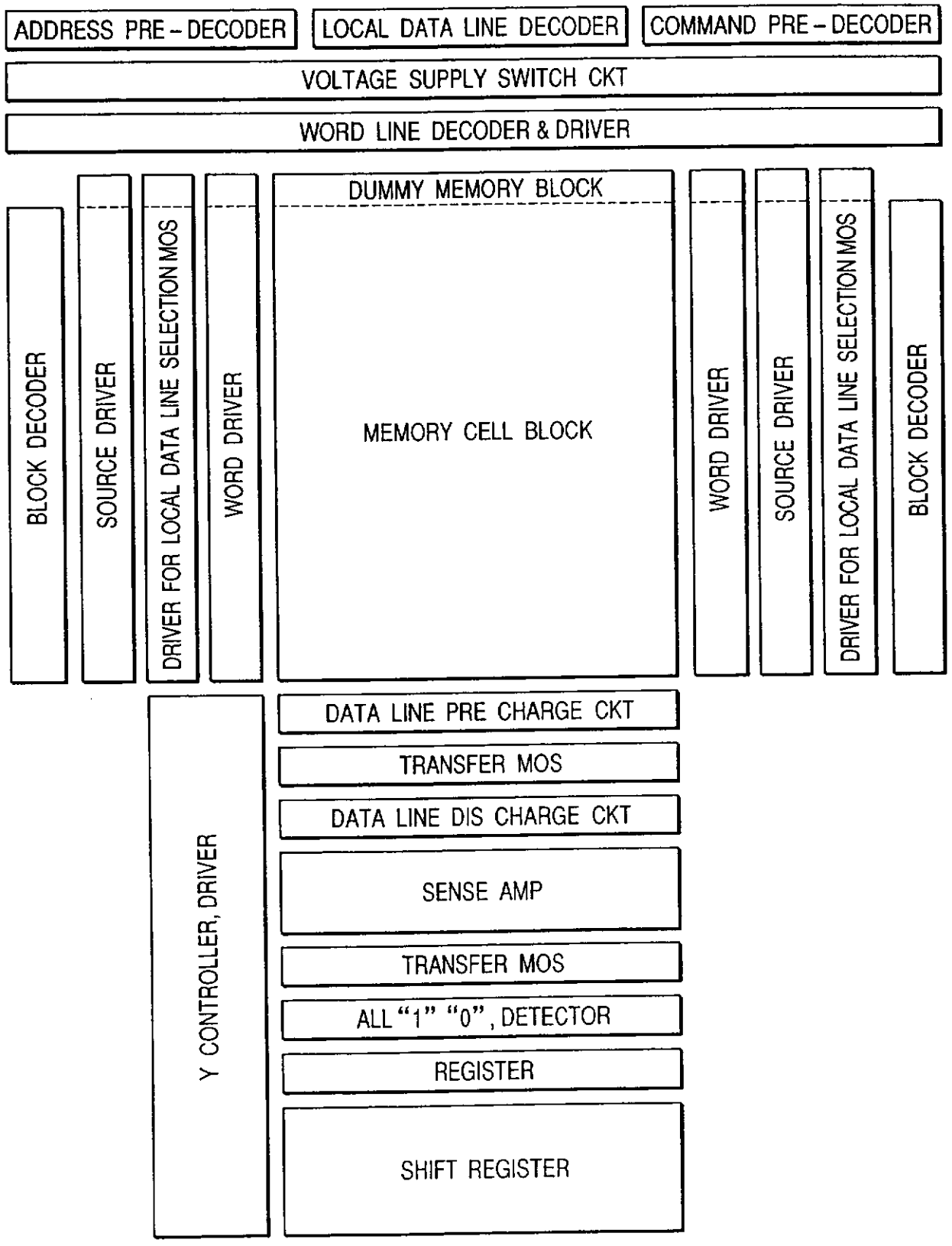

ADDRESS PRE-DECODER
LOCAL DATA LINE DECODER
COMMAND PRE-DECODER
VOLTAGE SUPPLY SWITCH CKT
WORD LINE DECODER & DRIVER
BLOCK DECODER
DRIVER FOR LOCAL DATA LINE SELECTION MOS
SOURCE DRIVER
WORD DRIVER
DUMMY MEMORY BLOCK
MEMORY CELL BLOCK
WORD DRIVER
SOURCE DRIVER
DRIVER FOR LOCAL DATA LINE SELECTION MOS
BLOCK DECODER
Y CONTROLLER, DRIVER
DATA LINE PRE CHARGE CKT
TRANSFER MOS
DATA LINE DIS CHARGE CKT
SENSE AMP
TRANSFER MOS
ALL "1" "0", DETECTOR
SHIFT REGISTER
SHOWN IN FIG. 51

FIG. 53


ADDRESS PRE-DECODER
LOCAL DATA LINE DECODER
COMMAND PRE-DECODER
VOLTAGE SUPPLY SWITCH CKT
WORD LINE DECODER & DRIVER
BLOCK DECODER
SOURCE DRIVER
DRIVER FOR LOCAL DATA LINE SELECTION MOS
WORD DRIVER
DUMMY MEMORY BLOCK
MEMORY CELL BLOCK
WORD DRIVER
SOURCE DRIVER
DRIVER FOR LOCAL DATA LINE SELECTION MOS
BLOCK DECODER
Y CONTROLLER, DRIVER
DATA LINE PRE CHARGE CKT
TRANSFER MOS
DATA LINE DIS CHARGE CKT
SENSE AMP
TRANSFER MOS
ALL "1" "0", DETECTOR
REGISTER
SHIFT REGISTER