(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: **0 039 403**
B1

(12) # EUROPEAN PATENT SPECIFICATION

(45) Date of publication of patent specification: **14.03.84**

(51) Int. Cl.³: **G 03 G 15/052**

(21) Application number: **81102041.1**

(22) Date of filing: **19.03.81**

(54) **Xerographic copier including controllable erase system.**

(30) Priority: **05.05.80 US 146714**

(43) Date of publication of application:
11.11.81 Bulletin 81/45

(45) Publication of the grant of the patent:
14.03.84 Bulletin 84/11

(84) Designated Contracting States:
BE CH DE FR GB IT LI NL SE

(56) References cited:
EP - A - 0 016 923
DE - A - 2 856 596

IBM TECHNICAL DISCLOSURE BULLETIN, Vol. 13, No. 12, May 1971 New York T.J. HARRIS "Optical Printer" pages 3757 to 3758
RESEARCH DISCLOSURE, No. 198, October 1980 Havant R.C. McELROY et al. "Code Printer for Photographic Paper" pages 421 to 422, article 19830
Patents Abstracts of Japan Vol. 2, No. 75, 14 June 1978 page 2861E78

(73) Proprietor: **International Business Machines Corporation**
Old Orchard Road
Armonk, N.Y. 10504 (US)

(72) Inventor: **Conly, Douglas Jarvis**
5437 Blackhawk Road
Boulder Colorado 80303 (US)
Inventor: **Baker, Bonnie Belle**
436 Sky Trail Road
Boulder Colorado 80302 (US)
Inventor: **Larson, David Duane**
820 Racquet Lane
Boulder Colorado 80302 (US)
Inventor: **Riddle, Stanley Thomas**
11431 East Summer Trail
Tucson Arizona 85715 (US)

(74) Representative: **Hawkins, Anthony George Frederick**
IBM United Kingdom Patent Operations Hursley Park
Winchester, Hants, S021 2JN (GB)

Courier Press, Leamington Spa, England.

EP 0 039 403 B1

Xerographic copier including controllable erase system

The present invention relates to a xerographic copier including a controllable erase system.

Xerographic devices may be divided into two types, printers and copiers. In the former type images are formed on a photo-conductor device in response to input electrical signals. Usually this type employs a light scanning system employing a laser or cathode ray tube light generator. Another printing arrangement is shown in the IBM Technical Disclosure Bulletin, Vol. 13, Number 12, May 1971 at pages 3757 and 3758. In this system, a row of light emitting diodes are positioned across a photo-conductor device and are driven by a system including five 120-bit shift registers and associated decoders to form character images on the photoconductor. All of the above printing arrangements suffer from the disadvantage that, though they can effect high speed printing, they require extensive logic and control systems which are costly. In copier devices, an image is normally produced on a photoconductor by light reflected from an original document. In order to prevent the formation of dense black areas at the edges of a copy, erase devices are employed to discharge the areas of the photo-conductor outside the image area. Where the image area is variable, the erase devices are controlled to vary the discharged areas to conform with the size of the image areas. A number of arrangements to effect such variable erasure have been proposed. These include selective shuttering of light sources and selective operation of light sources. For the latter of these arrangements, light emitting diodes have been found to be of particular value in view of their controllability.

In certain applications, it is desirable that information, in the form of characters or graphs, is added to a copy. This is of particular value in testing operations when, for example, a test image is produced and this is conveniently accompanied by data indicating various machine operations, such as the total number of copies produced over a period, or the number and type of faults which have occurred. Additionally, such added information may be useful under normal copying conditions, for example to identify a copy or its source.

It is, therefore, an object of the invention to provide the facility in a copier machine to generate symbols on a copy without employing added optical components and extensive extra logic and control devices.

According to the invention, there is provided a xerographic copier including a movable photoconductor device for supporting electrostatic latent images of documents and an erase device for erasing the photoconductor device in an area immediately adjacent an image area thereon, characterised in that said erase device comprises a plurality of light sources positioned adjacent the photo-conductor device, each operable, when energised, to illuminate an individual band on the photo-conductor device as it moves therepast, and arranged such that adjacent bands overlap, and control means coupled to control the energisation of the light sources to illuminate areas of the photoconductor in the configuration of symbolic representations.

The invention will now be described, by way of example, with reference to the accompanying drawings, in which:

Figure 1 is a diagram of an edge erase system comprising a diode block containing ten light emitting diodes;

Figure 2 shows a pattern formed on a moving photoconductor adjacent the diode block depicted in Figure 1;

Figure 3 shows a pattern formed by successive illumination of diodes in the block;

Figure 4 is a chart representing the operation of five of the diodes for producing a preferred character font;

Figure 5 is a representation of a character "one" produced by the five diodes in three successive periods;

Figure 6 is an illustration of the preferred font of the digits zero to four;

Figure 7 is an illustration of the preferred fonts of the digits five to nine;

Figure 8 shows, in pictorial and block form, an embodiment of the invention;

Figure 9 shows a controller for the system of Figure 8; and

Figure 10 is a state diagram representing the operation of the controller in Figure 9.

Figure 1 shows an edge erase device which is positioned adjacent a photoconductor drum in a xerographic copier. This device, as indicated in Figure 8, is positioned at one end of the photo-conductor drum and is normally employed to discharge an edge strip of the photoconductor adjacent the image area. The device includes ten light emitting diodes 0 to 9 which are arranged to illuminate overlapping bands on the photoconductor as it rotates beneath them. When employed in an edge erase function, selected ones of the diodes are energised constantly to define combined illuminated bands of different width in accordance with the position of the edge of the image area. Thus if the image on the drum is of the smallest area, then all of the diodes are illuminated, and as the image area is increased, selected diodes, first 0, then 0 and 1, then 0, 1 and 2 etc. are de-energised. Figure 2 illustrates the pattern on the photoconductor produced by energising all of teh diodes for a period during which the photoconductor moves through a distance indicated by the arrows.

It has now been realised that a variable edge erase device of the type shown in Figure 1 can be

used to produce information on a copy sheet adjacent the image area thereof. Such information may be used to identify a produced copy or its source. In addition it is often desirable to produce output information, as character or graphical symbolic representations, relating to the operation of a copier, particularly when it is under test. Such information may include the number of total hours of machine operation, the total number of copies produced, which may be divided into those made from different sources such as a semi-automatic document feed and an automatic document feeder. Other information may include that related to faults, such as the total number of paper jams which have occurred over a period. This information can, during operation of the copier, be recorded and retained in the control system.

Figure 3 shows a series of illumination bands produced by the diodes 2 to 9 for successive movements of the photoconductor over distances P0 to P5. It should be noted that diodes 0 and 1 are not energised in order to provide a gap between the image and characters produced by the diodes. From the pattern shown in Figure 3, by selective energisation of the diodes, numeric, alphabetic and special characters can be produced, though the following description will, for simplicity, be restricted to the production of numeric digits.

The font used to produce the digits comprises five rows using the diodes five to nine in the three successive positions illustrated in Figure 4. A digital character one, illustrated in Figure 5, is generated by activating diode nine in positions two and three, diode eight in position two, diode seven in position one, diode six in position four, and diode five in positions two to four.

Figures 6 and 7 illustrate how the characters from zero to nine can be formed in an analogous way.

In Figure 8, a copy drum 81 includes a photoconductive surface 82. Mounted adjacent to the photoconductive surface, but not touching it, is a diode block 83, shown as including ten diode positions. These correspond to the positions shown in Figure 1 as explained above only those numbered five to nine are to be used for generating the digits.

A sensor 84 is used to supply emit count signals to a controller 85 for determining the position of the photoconductive surface 82. The controller 85 supplies output signals which set various bits in an output register 86 which are coupled by a cable 87 to the individual light emitting diodes in the block 83. When a particular bit in the output register 86 is set, the corresponding connected LED in the block 83 is energised. When the bit is reset in the output register 86, the corresponding LED is de-energised.

Figure 9 illustrates a controller which can be used in the system of Figure 8. The explanation of the operation of Figure 9 can be more easily understood with the aid of the state diagram of Figure 10. Two clocked J-K flip-flops 91 and 92 form the state control portion of the controller. The states are numbered zero through three, corresponding to the binary values in the flip-flops 91 and 92 and decoded by a decoder 93. The timing control is provided by a zero-crossing detector 94 which is coupled to the a-c power line and provides an output signal ZX each time the alternating current input power voltage has a zero value. Such zero-crossing detectors are well known in the art and need no further explanation for an understanding of the invention.

The timing signal ZX is also applied to a divide-by-three circuit 95 which produces a control signal ZX3 which occurs every third zero-crossing of the input power.

It is assumed that the initial controller state is zero during which time the data to be printed is loaded into a data shift register 96 by a means not shown but which is well known in the art. During the state zero, a four-stage character counter 97 is cleared.

Also provided is a two-stage H-counter 98 which provides part of the character address as will be explained in greater detail below. A Read-Only Memory 99 (ROM) is also supplied. The ROM has a memory address register 910 and a data register 911. Table I illustrates the ROM data to produce the font of Figures 6 and 7 using five in-line LED's for simplicity of explanation rather than the offset LED's as shown in the block 83 (Figure 8).

The ROM address is a combination or concatenation of two bits of the two-stage H-counter 98 and the four bits of the binary coded decimal digit being displayed which is in the left-hand stage 961 of the data shift register 96. The output data comprises eight bits, the first five corresponding to the five diodes to be controlled, the least significant three bits being zero and not used.

The output data from the data register 911 is gated to the output register 86 (Figure 8) by a network of AND gates 914 which is enabled by the state three (S3) signal from the decoder 93.

The three columns of the font are identified by the first three values of the two-stage H-counter 98, viz., 00, 01, and 10. When the two-stage H-counter reaches a value of three, the character in the shift register stage 961 has been printed. The value of three in the H-counter 98 is detected by an AND gate 915 which produces a signal H=3 and by an inverter 916 which supplies the signal H≠3.

It is assumed that each copy sheet to be printed will contain eight characters. These are counted by the CH-counter 97. A NAND gate 917 provides an output signal indicating that the character count is not equal to eight.

In the operation of the controller, the output gates 914 are enabled at every third zero-crossing of the power, i.e., every S3 state. Each of the columns, therefore, has a width equal to the linear travel of the photoconductor in a period equal to three zero-crossings, i.e., 1/20-th of a second.

The detailed operation of the controller is as follows. The EC signal, which indicates that the drum

is in proper position to begin forming the characters, sets the flip-flop 92 causing the controller to enter state one. During state one, the information is read from the ROM at the address determined by the H-counter bits and the BCD character bits of the character in the stage 961. With the flip-flop 92 set, the ZX3 signal activates an AND gate 920 which causes the ZX signal to set the flip-flop 92. The CH≠8 signal from the NAND gate 917 is high so that the flip-flop 92 remains in the set state. This causes the controller to enter state three which gates the data register 911 to the output register via the gates 914. Also, the state three signal increments the H-counter 98 by one. At the next zero-crossing, the machine returns to state one if the H-counter has not reached the value of three. This is caused by the ZX signal clocking the J-K flip-flop 91 with only the K input (H≠3) signal high. Initially, the first two ROM address bits are 00 and the other four bits are the BCD equivalent of the character being written. Therefore, at the next ZX3 time, the BCD digits of the address are the same but the H-counter now has a value of 01 so that the second column of the data controlling the digit are accessed from the ROM and at state three are gated to the output register. This sequence continues until after the third column is printed and the H-counter is incremented to a value of three.

When H=3, the next zero-crossing causes the controller to enter state two, caused by clocking the flip-flop 92 with both inputs high, the K input signal being the high output signal from the AND gate 921.

In state two, the data register 96 is shifted to place the next digit in stage 961 and the H-counter 98 is cleared to zero. Also, the CH-counter 97 is incremented by one. With the CH≠8, the next zero-crossing causes the machine to enter state one where the above sequence is repeated for the second digit.

The above-described operation is repeated until all eight characters have been printed at which time the low signal from the NAND gate 917 at the J input of the flip-flop 92 causes the flip-flop 92 to be reset and the controller assumes the original state zero.

In a copier incorporating a microprocessor for effecting operational control, it is possible for the microprocessor to perform the functions of the above described controller in addition to its other functions. By doing this, the cost of the hardware is reduced, and flexibility is increased. For example differing character fonts may be generated. The following description details the operation of a microprocessor to control the edge erase device.

Microprocessor are well known in the art and commercially available. The following description and flowcharts can be applied to any of the available microprocessors. Appendix B hereto describes the conventions employed in the flowcharts (Charts I to III). The numbers in the right-hand column identify the hexadecimal address of the first instruction in the attached programs related to the associated step. The attached programs can be used on a microprocessor such as that described in U.S. Patent Specification No. 4170414. Appendix A summarizes the relevant instructions.

The program for controlling a copier must perform many tasks in addition to the program being described. Therefore, certain portions of the invention may be found in separate routines. In one embodiment, a STARTOUT routine is used to begin the data output according to the invention. This is flowcharted in Chart I and is used to initialize the data output before printing. The STARTOUT routine performs several functions. It loads the contents of the next register to be displayed into OUTDATA1 (low order byte) and OUTDATA2 (high order byte) registers. These correspond to the data shift register 96 (Figure 9). The data is left-justified to suppress leading zeros and an output character count (CHARCNT) is appropriately incremented to insure that the output of only valid (justified) data. The output control counter is initialized to a value of −1 and the current character (CURRCHAR) and last character (LASTCHAR) registers are loaded with an address to insure that a null output precedes the data. The STARTOUT routine is called by a CZCOUNT routine, a pseudo-emitter routine which is used to control the timing. As shown in Chart I, the STARTOUT routine sets the OUTPTNOW bit at line 2. This bit is used to indicate to another routine (LEDWRITE) that the data is ready for printing. Next, the OUTFETCH subroutine is called. This subroutine fetches and converts the register content to be displayed. It loads the contents from a selected register whose value is expressed in binary and converts it to eight binary coded decimal digits for data output. (OUTFETCH is flowcharted in Chart II below.)

At line 4, the output mode counters are initialized by clearing CHARCNT to zero and setting OUTCOUNT to −1. Next, a loop is performed while CHARCNT≠8 and the higher order digit is equal to zero. The zero is shifted out and the CHARCNT is incremented by one. If an odd-number character is in the high order digit, the program transfers to line 11. Otherwise, as shown as step 9, all the lower digits are shifted two places to the left and the loop repeated. After this is completed, i.e., CHARCNT=8 or the high order digit is not zero, the subroutine returns to the calling routine.

In Chart II, the OUTFETCH routine is detailed. First, the address of the next register to be written is fetched. The DIVIDE routine is called which divides the register value by 10,000. This, in effect, splits the register into two four decimal digit values. Next, the subroutine BINBCD is called to convert the high order digits to be BCD. Conversion of binary to BCD is well known in the art and need not be explained for an understanding of the invention. Next, the binary coded digits are stored in the high output register OUTDATA2, and the BINBCD subroutine is called to convert the lower order digits to BCD which are then stored in the low output register OUTDATA1.

The LEDWRITE routine (Chart III) is used to write the data onto the photoconductor using the variable edge erase lamps. The written image is then developed and transferred to a copy sheet as is well known in the art. The following steps have already been done by the STARTOUT routine before calling the LEDWRITE routine. The data to be printed is stored in the OUTDATA registers in BCD format, the number of trailing digits, i.e., those to be skipped because of zero suppression, is stored in CHARCNT, and an address eleven bytes before the first null character address in the output table has been loaded into the CURRCHAR and LASTCHAR registers. (This address is symbolically indicated as CHARNP0—11.) The output control counter, OUTCOUNT, has been set to −1. The OUTPTNOW bit is tested (step 2) and, if not set, the program branches to step 17 which ends (exits) the routine. Otherwise, the routine goes to step four where it is determined whether it is time to change the output. If the output is not to be changed, the current output pattern is written and the routine ends. The output is changed at every third zero-crossing as signified by a bit which is supplied by another program before the LEDWRITE routine is called. If it is time to change the output, the output change counter is reset and the position counter is incremented by one. A test is made to determine whether the position counter indicates position four. If not, the pointer is changed to the next output state and the LASTCHAR address is tested to determine whether it is beyond the end of the font table. If not, the last character bit pattern is loaded and the current character and last character bits are combined. This will be explained below in more detail. At the step 14, the lamps not to be written are turned off and the lights to be written will be turned on at the next zero-crossing. Next, the character count is checked. If equal to ten, then OUTPTNOW bit is reset and the routine ends. If the character count is not equal to ten, the routine ends and will be resumed on the next zero-crossing. At step nine, if position four was sensed, the counter is returned to position zero and the current table address is moved to the last table address, the character count is incremented by one, and it is determined whether the last character has been written. If so, the CHARNP0 (null character) is moved to the current character.

Because the LED's are offset, some of the next character positions will be encountered before the last character positions of the current character so the bits of the current character and last character are combined. This is done by shifting the last character to the left and ORing the bits of the next character into the resulting low order zero bits.

Table II below represents a font table useful for the offset LED writing where the location of the first character of position zero begins at the hexadecimal address F7B8. The contents of the memory are indicated in hexadecimal characters.

CHART I: STARTOUT ROUTINE

1. enter	
2. set OUTPTNOW bit	EF78
3. call OUTFETCH	EF7E
4. initialize output mode counters	
4a. clear CHAR.CNT to zero	EF83
4b. set OUTCOUNT to −1	EF85
5. WHILE CHAR≠8 & HIGHORDER DIGIT=0	
6. shift LEADING ZERO out	EF92
7. increment CHAR.CNT by 1	EF98
8. ODD NUMBERED CHAR.? (11)	EF99
9. shift ALL LOWER DIGIT TWO PLACES left	EF9C
10. LOOP	
11. load TRAILING NULL CHAR. into CURR. and LASTCHAR registers	EFA3
12. return	EFAC

CHART II: OUTFETCH ROUTINE

1. enter	
2. fetch ADDR. of NEXT REGISTER to be OUTPUT	EFAD
3. call DIVIDE (divide register by 10,000)	EFC9
4. call BINBCD (convert high order digits to BCD)	EFD1
5. store DIGITS in HIGH OUTPUT register (OUTDATA2)	EFD4
6. call BINBCD (convert low order digits to BCD)	EFDA
7. store digits in LOW OUTPUT register (OUTDATA1)	EFDD
8. return	EFE2

CHART III: LEDWRITE ROUTINE

1. begin	
2. OUTPTNOW bit set? (4)	D4EA
3. (17)	D4EE
4. TIME TO CHANGE OUTPUT? (7)	D4F0
5. write CURR. OUTPUT PATTERN	D568

6.	(17)	
7.	reset OUTPUT CHANGE COUNTER	D4F5
8.	increment POSIT. COUNTER by 1	
9.	POSITION 4? (18)	D4F7
10.	point to NEXT OUTPUT STATES	D531
11.	LAST CHAR. ADDR. BEYOND END OF FONT TABLE? (13)	D539
12.	load LAST CHAR. BIT PATTERN	D542
13.	combine CURR. CHAR. and LAST CHAR. bits	D543
14.	turn off bits not to be written-turn on at next zero-crossing	D548
15.	CHAR. COUNT≠10? (17)	D55E
16.	reset OUTPTNOW bit	D563
17.	end	D57A
18.	reset to POSITION 0	D4FC
19.	move CURR. TABLE ADDR. to LAST TABLE ADDR.	D4FE
20.	increment CHAR. COUNT by 1	D502
21.	LAST CHAR.? (28)	D503
22.	fetch NEXT CHAR.	D507
23.	compute CHAR. OUTPUT TABLE ADDR.	D519
24.	is CHAR. an EVEN NUMBER? (26)	D520
25.	(11)	
26.	move NEXT TWO DIGITS to HIGH REGISTER	D524
27.	(11)	
28.	move CHARNPO to CURR. CHAR.	D52A
29.	(11)	

TABLE I: SINGLE ROW ROM DATA

ROM ADDRESS	ROM DATA
00 0000	11111000
00 0001	00000000
00 0010	10111000
00 0011	10101000
00 0100	11100000
00 0101	11101000
00 0110	11111000
00 0111	10000000
00 1000	11111000
00 1001	11100000
00 1010	00000000
01 0000	10001000
01 0001	11111000
01 0010	10101000
01 0011	10101000
01 0100	00100000
01 0101	10101000
01 0110	10101000
01 0111	10111000
01 1000	10101000
01 1001	10100000
01 1010	00000000
10 0000	11111000
10 0001	00000000
10 0010	11101000
10 0011	11111000
10 0100	11111000
10 0101	10111000
10 0110	00111000
10 0111	11100000
10 1000	11111000
10 1001	11111000
10 1010	00000000

TABLE II: FONT TABLE

F7B8	800080808080808000	CHAR 0—9, null—Position P0
F7C3	018080808181818080818100	CHAR 0—9, null—Position P1
F7CE	A201A2A282A3A282A28200	CHAR 0—9, null—Position P2
F7D9	6322632301226223630300	CHAR 0—9, null—Position P3
F7E4	2240222222222042222200	CHAR 0—9, null—Position P4
F7EF	400000404040404000404000	CHAR 0—9, null—Position P5

APPENDIX A

INSTRUCTION MNEMONIC	HEX VALUE	NAME	DESCRIPTION
AB (L)	A4	Add Byte (Low)	Adds addressed operand to LACC (8-bit op.)
AI (L)	AC	Add Immed. (Low)	Adds address field to LACC (16-bit op.)
AR	DN	Add Reg.	Adds N-th register contents to ACC (16-bit op.)
A1	2E	Add one	Adds 1 to ACC (16-bit op.)
B	24,28,2C	Branch	Branch to LSB (+256, —256, +0)
BAL	30—33	Branch And Link	Used to call subroutines (PC to Reg. 0, 1, 2, or 3)
BE	35,39,3D	Branch Equal	Branches if EQ set (See B)
BH	36,3A,3E	Branch High	Branch if EQ and LO are reset (See B)
BNE	34,38,3C	Branch Not Equal	Branch if EQ reset (See B)
BNL	37,3B,3F	Branch Not Low	Branch if LO reset (See B)
BR	20—23	Branch Reg.	See RTN
CB (L)	A0	Compare Byte (Low)	Addressed byte compared to LACC (8-bit op.)
CI (L)	A8	Compare Immed. (Low)	Address field compared to LACC (8-bit op.)
CLA	25	Clear Acc.	ACC reset to all zeroes (16-bit op.)
GI	A9	Group Immed.	Selects one of 16 register groups (also controls interrupts)
IC	2D	Input Carry	Generate carry into ALU
IN	26	Input	Read into LACC from addressed device (8-bit op.)
j	0N,1N	Jump	Jump (forward or back) to PC (15—4), N
JE	4N,5N	Jump Equal	Jump if EQ set (see J)
JNE	6N,7N	Jump Not Equal	Jump if EQ reset (See J)
LB (L)	A6	Load Byte (L)	Load addressed byte into LACC (8-bit op.)
LI	AE	Load Immed.	Load address field into LACC
LN	98—9F	Load Indirect	Load byte addressed by reg. 8—F into LACC (8-bit op.)
LR	EN	Load Register	Load register N into ACC (16-bit op.)
LRB	FN	Load Reg./ Bump	Load reg. N into ACC and increment; ACC to Reg. N (N=4—7, C—F) (16-bit op.)
LRD	FN	Load Reg./ Decr.	Load reg. N into ACC and decrement; ACC to Reg. N (N=0—3, 8—B) (16-bit op.)
NB (L)	A3	And Byte (Low)	AND addressed byte into LACC (8-bit op.)
NI (L)	AB	And Immed. (Low)	AND address field into LACC (8-bit op.)
OB (L)	A7	Or Byte (Low)	OR addressed byte into LACC (8-bit op.)
OI (L)	AF	Or Immed. (Low)	OR address field into LACC (8-bit op.)

APPENDIX A (contd.)

INSTRUCTION MNEMONIC	HEX VALUE	NAME	DESCRIPTION
OUT	27	Output	Write LACC to addressed device
RTN	20—23	Return	Used to return to calling program (See BAL)
SB (L)	A2	Subtract Byte (Low)	Subtract addressed byte from LACC (8-bit op.)
SHL	2B	Shift Left	Shift ACC one bit left (16-bit op.)
SHR	2F	Shift Right	Shift ACC one bit right (16-bit op.)
SI (L)	AA	Subtract Immed. (Low)	Subtract address field from LACC (16-bit op.)
SR	CN	Subtract Reg.	Subtract reg. N from ACC (16-bit op.)
STB (L)	A1	Store Byte (Low)	Store LACC at address (8-bit op.)
STN	B8—BF	Store Indirect	Store LACC at address in Reg. 8—F
STR	8N	Store Reg	Store ACC in Reg. N (16-bit op.)
S1	2A	Subtract One	Subtract 1 from ACC (16-bit op.)
TP	9N	Test/Preserve	Test N-th bit in LACC (N=0—7)
TR	BN	Test/Reset	Test and reset N-th bit in LACC
TRA	29	Transpose	Interchange HACC and LACC
XB (L)	A5	XOR Byte (Low)	Exclusive-OR addressed byte into LACC (8-bit op.)
XI (L)	AD	XOR Immed. (Low)	Exclusive-OR address field into LACC (8-bit op.)

Notes:

ACC (Accumulator) is 16-bit output register from arithmetic-logic unit

—LACC signifies herein the low ACC byte; HACC, the high byte
—all single byte operations are into low byte
—register operations are 16-bit (two-byte)
—8-bit operations do not affect HACC

EQ (equal is a flag which is set:
if ACC=0 after register AND or XOR operations;
if ACC (low byte)=0 after single byte operation;
if a tested bit is 0;
if bits set by OR were all 0's;
if input carry=0;
if compare operands are equal;
if bit shifted out of ACC=0;
if 8th bit of data during IN or OUT=0.

LO (low is a flag which is set: (always reset by IN, OUT, IC)
if ACC bit 16=1 after register operation;
if ACC bit 8=1 after single byte operations;
if logic operation produces all ones in LACC;
if all bits other than tested bit=0;
if ACC=0 after shift operation;
if compare operand is greater than ACC low byte.

MACRO MNEMONIC	NAME	DESCRIPTION
BC	Branch on Carry	Branches if carry is set
BCT	Branch on Count	Reg. decremented and branch if not zero result
BHA	Branch on High ACC	Used after compare
BL	Branch on Low	Branches if LO is set
BLA	Branch on Low ACC	See BNC; used after compare
BNC	Branch Not Carry	Branches if carry is reset
BNLA	Branch on Not Low ACC	See BC; used after compare
BNZ	Branch Not Zero	Branches if previous result was not zero
BR	Branch via Register	Same as RTN instruction
BU	Branch Unconditionally	Same as BAL instruction
CIL	Compare Immed. Low	Uses low type of indicated constant in CI address field
DC	Define Constant	Reserves space for constant
EXP2	Express In powers of 2	Opcode set to binary
JC	Jump on Carry	See BC
JL	Jump on Low	See BL
JNC	Jump on No Carry	See BNC
JNH	Jump Not High	See BNH
LA	Load Address	Generates sequence LIH, TRA, LIL
LBD	Load Byte Double	Bytes at addr. and addr. +1 to ACC
LID	Load Immed. Double	Same as LA
LIH	Load Immed. High	Uses high byte of constant in LI address field
LIL	Load Immed. Low	Uses low byte of constant in LI address field
NOP	No Operation	Dummy instruction—skipped
RAL	Rotate ACC Left	Generates sequence SHL, IC, A1
SCTI	Set Count Immed.	Generates CLA, LI, STR
SHLM	Shift Left Multiple	Shifts specified number of times to left
SHRM	Shift Right Multiple	Shifts specified number of times to right
SRG	Set Register Group	Same as GI
STDB	Store Byte Double	ACC to addr. +1 and addr.
TPB	Test & Preserve Bit	Generates sequence LB, TP
TRB	Test & Reset Bit	Generates sequence LB, TR, STB
TRMB	Test & Reset Multiple Bits	Same as TRB but specifies multiple bits
TRMR	Test/Reset Mult. Bits in Reg.	Gemerates LR, NI, STR
TS	Test and Set	Same as OI instruction
TSB	Test & Set Byte	Same as TS but byte is specified in addition to bit

MACRO MNEMONIC	NAME	DESCRIPTION
TSMB	Test & Set Multiple Bytes	Same as TS but specifies multiple Bits
TSMR	Test & Set Mult. Bits in Reg.	Generates LR, OI, STR
LZI	Zero & Load Immed.	Generates CLA, LI

Notes:
(Label) DC* causes the present location (*) to be associated with the label.
L and H, in general, are suffixes indicating low or high byte when 16 bit operands are addressed.

APPENDIX B
Conventions used in flowcharts

1. Each step is numbered.
2. The first state is "begin" for in-line programs and "enter" for subroutines.
3. In-line programs terminate with "end" and subroutines, with "return".
4. A step number in parentheses as a statement or part of the statement indicates a branch to the step.
5. A test statement is followed by a question mark; the question mark is followed by a step number in parentheses indicating the branch to be taken if the test result is true.
6. The call command invokes an off-line subroutine which returns to the following step or statement upon completion.
7. The WHILE (logical statement) . . . LOOP command executes the statements enclosed thereby as long as the logical statement is true.
8. Indentations are used to improve readability but have no significance otherwise.

Claims

1. A xerographic copier including a movable photoconductor device (81) for supporting electrostatic latent images of documents and an erase device (83) for erasing the photoconductor device in an area immediately adjacent an image area thereon, characterised in that said erase device comprises a plurality of light sources (0—9) positioned adjacent the photoconductor device, each operable, when energised, to illuminate an individual band on the photoconductor device as it moves therepast, and arranged such that adjacent bands overlap, and control means (85, 86) coupled to control the energisation of the light sources to illuminate areas of the photoconductor in the configuration of symbolic representations.

2. A copier according to claim 1 further characterised in that said erase device comprises an edge erase device positioned at one side of the photoconductor device.

3. A copier according to claim 1 or claim 2 further characterised in that said light sources are arranged in rows in a direction orthogonal to the direction of movement of the photoconductor device, the light sources in adjacent rows being offset such as to effect said overlapping of adjacent bands.

4. A copier according to any of claims 1 to 3 further characterised in that the controller, when effecting control of the erase device to form said symbolic representations, causes at least the light source immediately adjacent the image area to remain unenergised thereby to effect separation between the image area and the representations.

5. A copier according to any of claims 1 to 4, further characterised in that said light sources are light emitting diodes.

Revendications

1. Copieur xérographique comprenant un dispositif photoconducteur mobile (81) pour porter des images latentes électrostatiques de documents et un dispositif d'effacement (83) pour effacer le dispositif photoconducteur dans une zone immédiatement adjacente à une zone d'image portée par celui-ci, caractérisé en ce que ledit dispositif d'effacement comprend une pluralité de sources lumineuses (0—9) disposées adjacentes ou dispositif photoconducteur, chacune fonctionnant, lorsqu'elles sont excitées, pour illuminer une bande particulière sur le dispositif photoconducteur lorsque celui-ci passe devant ladite source lumineuse excitée et agencées de façon que des bandes adjacentes se chevauchent, et des moyens de commande (85, 86) accouplés pour commander l'excitation des sources lumineuses pour illuminer des zones du photoconducteur selon la configuration de représentations symboliques.

2. Copieur selon la revendication 1 caractérisé en outre en ce que ledit dispositif d'effacement comprend un dispositif d'effacement des bords disposé d'un côté du dispositif semiconducteur.

3. Copieur selon la revendication 1 ou 2 caractérisé en outre en ce que lesdites sources lumineuses sont agencées en rangées dans une direction orthogonal par rapport au sens du mouvement du dispositif semiconducteur, les sources lumineuses dans des rangées adjacentes étant décalées de façon à provoquer ledit chevauchement des bandes adjacentes.

4. Copieur selon l'une quelconque des revendications 1 à 3 caractérisé en outre en ce que l'unité de commande, lorsqu'elle assure la commande du dispositif d'effacement pour former lesdites représentations symboliques, provoque au moins le maintien en l'état non excité de la source lumineuse immédiatement adjacente à la zone d'image, ce qui permet d'assurer la séparation entre la zone d'image et les représentations.

5. Copieur selon l'une quelconque des revendications 1 à 4 caractérisé en outre en ce que lesdites sources lumineuses sont des diodes électroluminescentes.

Patentansprüche

1. Xeroraphisches Kopiergeraet mit einem bewegbaren Zwischenbildtraeger-Einrichtung (81), die elektrostatische latente Bilder und ein Loeschsystem (83) zum Loeschen einer Zwischenbildtraeger-Einrichtung in einem unmittelbar an einem Bildfeld anliegenden Feld, dadurch gekennzeichnet, dass das Loeschsystem eine Vielzahl von an der Zwischenbildtraeger-Einrichtung anliegenden Lichtquellen (0—9), wo jede erregte Lichtquelle zur Beleuchtung eines einzelnen Bandes auf der davor laufenden Zwischenbildtraeger-Einrichtung bewirkt werden kann, und wo sie so angeordnet sind, dass es eine Ueberlappung der anliegenden Baender erfolgt, sowie gekoppelte Steuermittel (85, 86) zum Steuern der Erregung der Lichtquellen aufweist, um Felder aus Zwischenbildtraegers innerhalb des Musters von symbolischen Darstellungen zu beleuchten.

2. Kopiergeraet nach Anspruch 1, weiterhin dadurch gekennzeichnet, dass das Loeschsystem eine auf einer Seite des Zwischenbildtraegers angeordneten Einrichtung zum Loeschen des Randes aufweist.

3. Kopiergeraet nach Anspruch 1 oder 2, weiterhin dadurch gekennzeichnet, dass die Lichtquellen in in einer zu der Bewegungsrichtung des Zwischenbildtraegers rechtwinklige Richtung angeordneten Spalten zusammengefasst sind, wobei die anliegenden Spalten von Lichtquellen so verschoben sind, um Ueberlappung der anliegenden Baender zu bewirken.

4. Kopiergeraet nach einem der Ansprueche 1 bis 3, weiterhin dadurch gekennzeichnet, dass die Steueranordnung, wenn sie das Loeschsystem zur Bildung der symbolischen Darstellungen so steuert, dass mindestens die unmittelbar an dem Bildfeld anliegende Lichtquelle in einem aberregten Zustand gehalten ist, um so die Trennung zwischen dem Bildfeld und den Darstellungen zu bewirken.

5. Kopiergeraet nach einem der Ansprueche 1 bis 4, weiterhin dadurch gekennzeichnet, dass die Lichtquellen lichtemittierende Dioden sind.

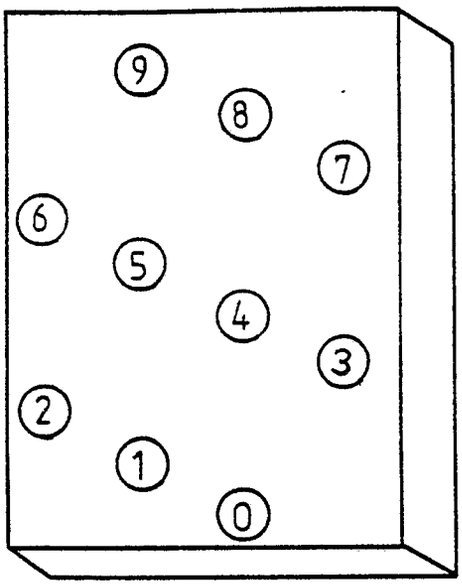

FIG. 1

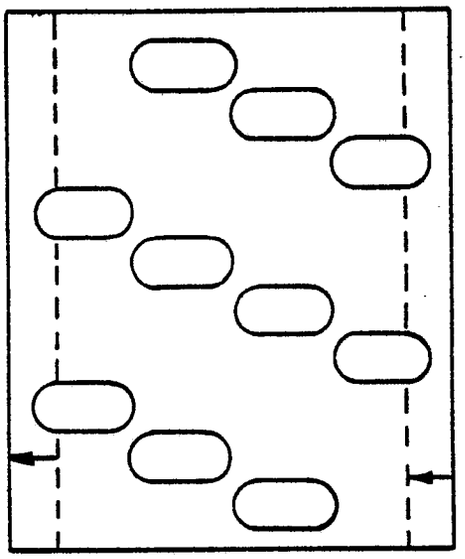

FIG. 2

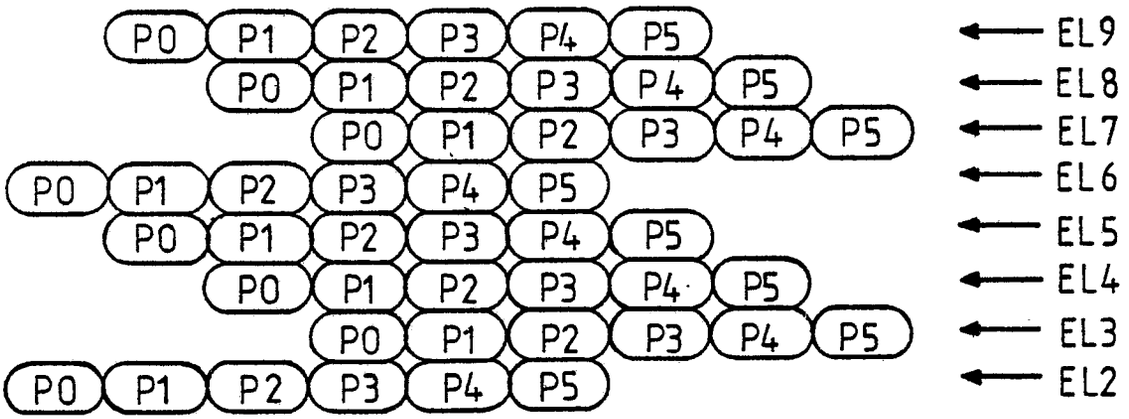

FIG. 3

P2	P3	P4	EL9
P1	P2	P3	EL8
P0	P1	P2	EL7
P3	P4	P5	EL6
P2	P3	P4	EL5

FIG. 4

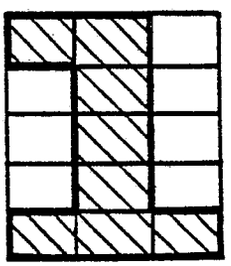

FIG. 5

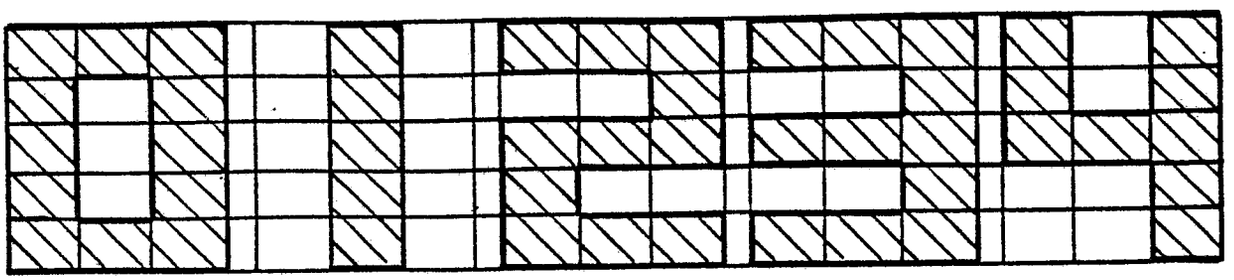

FIG. 6

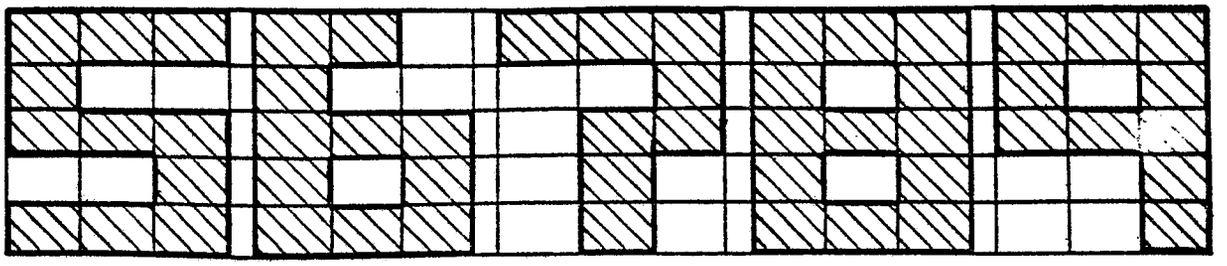

FIG. 7

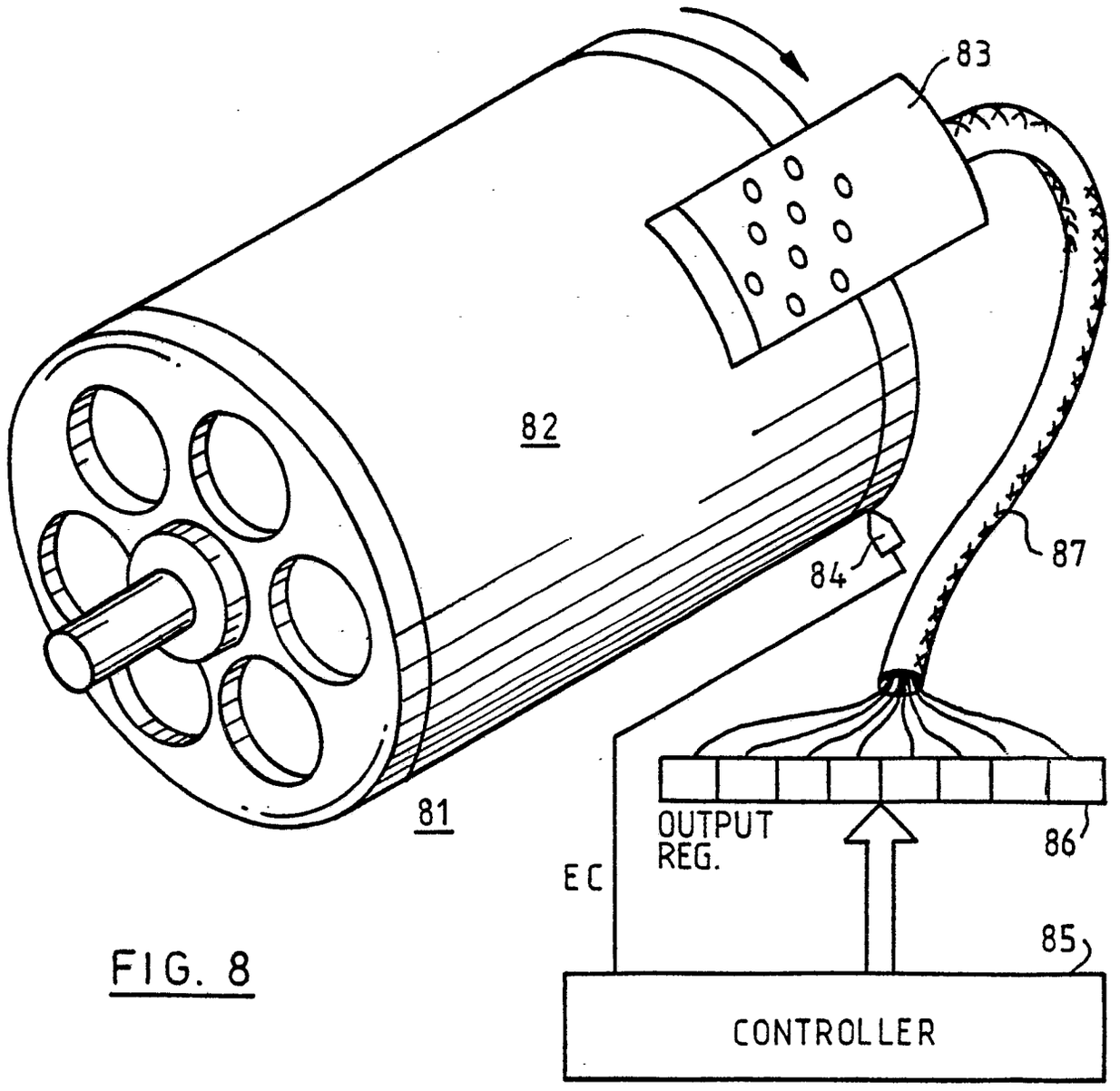

FIG. 8

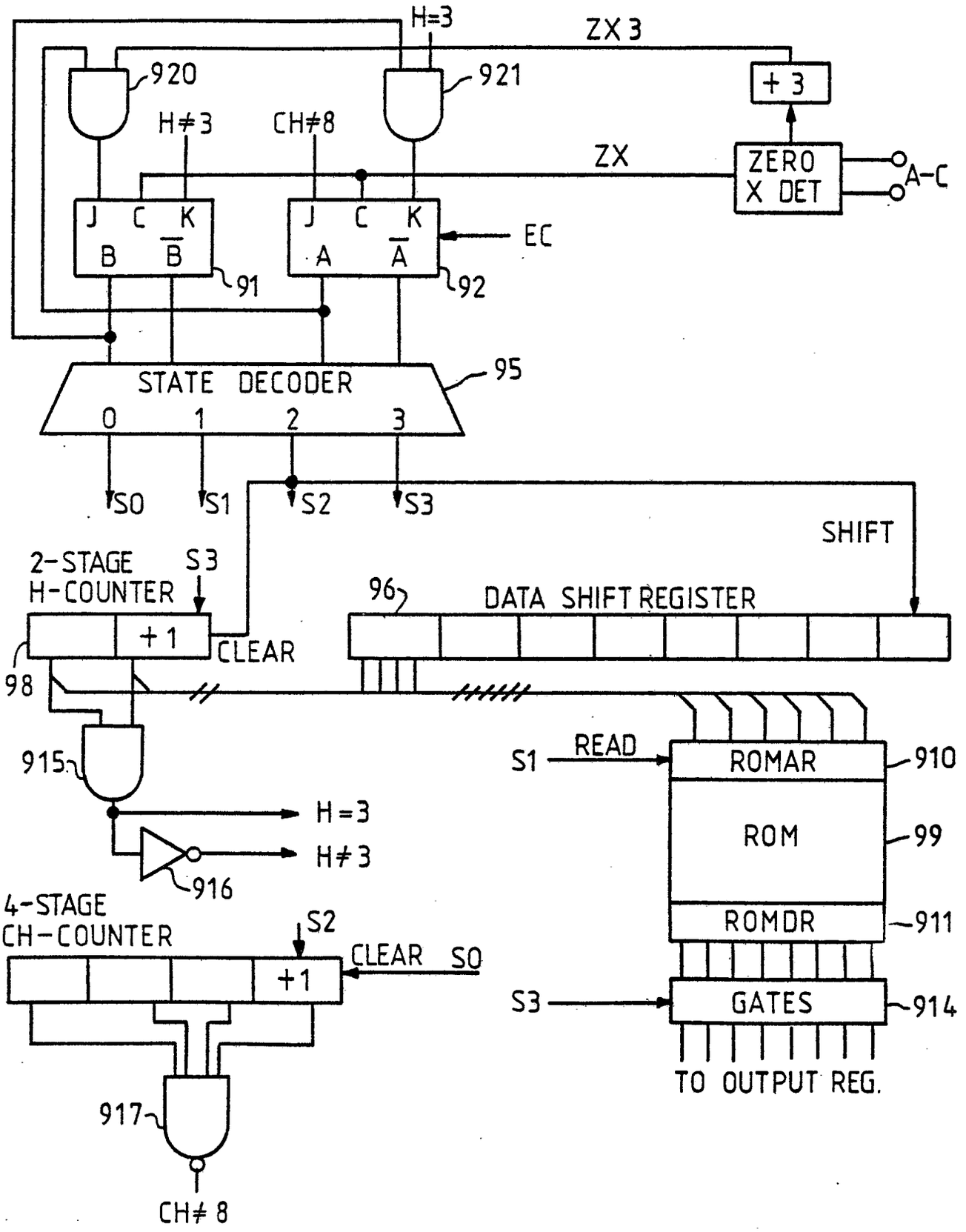
H=3
ZX3
920
921
+3
H≠3
CH≠8
ZX
ZERO
X DET
A-C
J C K
B B̄
91
J C K
A Ā
92
EC
STATE DECODER
95
0 1 2 3
S0 S1 S2 S3
SHIFT
2-STAGE
H-COUNTER
S3
+1
CLEAR
96
DATA SHIFT REGISTER
98
915
H=3
H≠3
916
READ
S1
ROMAR
910
ROM
99
ROMDR
911
4-STAGE
CH-COUNTER
S2
+1
CLEAR
S0
S3
GATES
914
TO OUTPUT REG.
917
CH≠8

FIG. 9

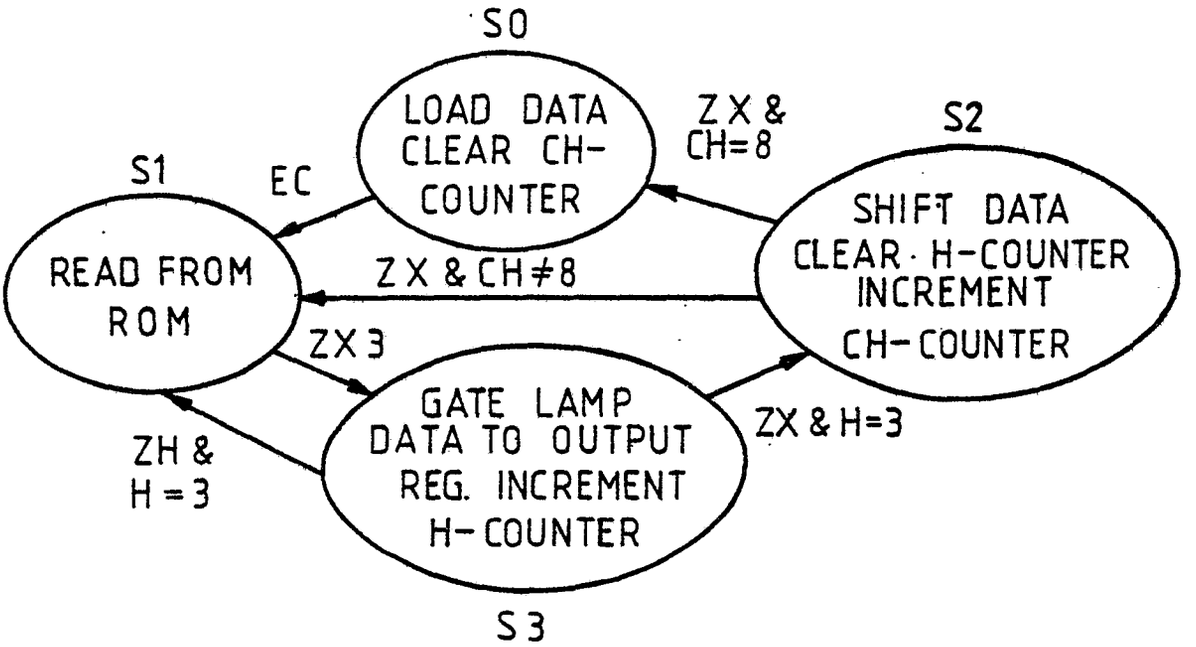
S0
LOAD DATA
CLEAR CH-
COUNTER
ZX &
CH=8
S2
S1
EC
READ FROM
ROM
ZX & CH≠8
SHIFT DATA
CLEAR · H-COUNTER
INCREMENT
CH-COUNTER
ZX 3
GATE LAMP
DATA TO OUTPUT
REG. INCREMENT
H-COUNTER
ZX & H=3
ZH &
H=3
S3

FIG. 10