

LOWERCASE WORDS ARE 1-BYTE KEYPAD WORDS. UNDERSCORES ARE SPACES.

[SZ=333]

## SIMON

BY BRETT BILBRAY [MODIFIED BY MIKE WHITE] EB VERSION

ENTER>DE.;clear

```
1 if D=1&(V)=66;Y=W
2 if D=2&(V)=49;Y=X
3 if D=3&(V)=32;Z=T+3
4 if D=4&(V)=M;Z=-893
5 if B<2&(22)=0;if B=0for D=-Cto C;next D
6 if B=1if G#3if 2-JY(1)=Cif (JX(1)+1)÷2+C=Dgoto 6+(I>120),I,I+1
7 if B=1if G=3if TR(D)goto 7+(I>120),I,I+1
8 if B=2gosub 0;&(23)=0;&(21)=S;&(22)=0;for H=50to 0step -1;&(20)=H;&(V)=150-H;f
or I=0to S;next I;next H;PL.;&(S)=71;gosub 0
9 &(22)=0;data Y,R,T;if (G<3)+(B#2)return
10 H=D÷2;data CX,12-MxRM,20-Mx(D÷3),1;if B<3TV=D+N;CX=CX-6
11 if B>1CC=2;TV=88
12 return
16 default ;clear ;CX=-12;print "SIMON";CX=-45;print "BY_BRETT_BILBRAY";data CX,
-30,-M;print "[REWRITTEN]";CX=-42;print "[BY_MIKE_WHITE]";data NT,P,-36,CY,2;pri
nt "(BRBLVERSION)",
17 %(28668)=>17-2;data BC,7,0,74,0;&(0)=7;&(2)=74;&(3)=0;&(9)=20;for B=27to 25st
ep -1;circle 0,8,B,7;next B;for B=2to M;C=9
18 if PX(B,C+1)#3goto 18,C,C+1
19 line B,U,0;line B,C,9;line -B,U,0;line -B,C,9;line -B,6,0;line -B,S-C,9;line
B,6,0;line B,S-C,9;next B;box 0,8,50,3,7;box 0,8,3,50,7;box 0,8,20,3,4;box 0,8,3
,20,4;circle 0,8,12,7;for B=1to 11;C=9
20 if PX(B,C+1)#3goto 20,C,C+1
21 line B,8,0;line B,C,4;line -B,C,4;line -B,S-C,4;line B,S-C,4;line B,8,4;next
B;circle 0,8,11,7;circle 0,8,U,7;for B=-4to 4step 4;box B,U,2,2,6;next B;box -4,
6,4,2,6;box 3,6,6,2,6;for F=0to 999;next F
22 %(28668)=24576;gosub 0,C,2;gosub 0,J,0,0;if CX=-Ubox 0,U,2,2,F-3;if F=3&(9)=6
3;goto S
23 gosub R,CX,-15,N,7;print "GAME"
24 for H=21to 23;data CX,15,N,6;if &(H)=Sgoto 26,G,M-H
25 G=&(28)÷86+1;TV=G+N;next H;if TR(1)=0goto M
26 data NT,0,15,N,7;TV=G+N;gosub 0,C,2;if G>1goto 32,I,4
27 data NT,P,-33,40;print "DIFFICULTY"
28 for H=21to 23;if &(H)=Sgoto 31,I,M-H
29 I=&(28)÷64+1;if &(23)=8goto 31,I,4
30 data CX,33,40,6;TV=I+N;next H;if TR(1)=0goto 28
31 data NT,0,33,40,7;TV=I+N;gosub 0
32 gosub 0,B,-M;if G=3for D=1to 4;gosub U,B,1;next D
33 data E,0,Ix6+1;A=0;if I=4F=30
34 @(A)=rnd(4);if G=3if @(A)=J)+(A)=K)goto 34
35 gosub 0,C,2;gosub X,B,0,E
36 for H=0to A;D=0;for I=0to 60;if G=3for D=1to 4;if (D=J)+(D=K)+(TR(D)=0)next D
;D=0
37 if G<3data B,JX(1),JY(1);if Bif Cdata B,B+1,2-C,B÷2+C
38 if D=0next I;goto N,C,1,@(H)
39 if D#@&(H)goto N,C,1
40 gosub 1,B,1;next H;if (A=4)+(A=8)+(A=12)E=E÷2
41 data CX,-42,-32,6;print A+1;if A=Fgoto 46,E,D,4,1
42 A=A+1;if G#2goto 34
43 D=0;for H=0to 60;data B,JX(1),JY(1);if Bif Cdata B,B+1,2-C,B÷2+C
44 if D=0next H;goto N,A,A-1,2,1
45 @(A)=D;gosub 1,B,1;goto 36
46 for H=1to 8;gosub 1,B,0,N,E;next H;if (A<30)+(G=3)goto 52
47 for I=1to S;gosub 1,B,0,N,rnd(4);next I;data C,0,0
48 F=4;gosub 1,B,2;if G#3goto 52
49 if J=0goto 33,I,4,K,D
```

```

50 for H=1to 4;if (J=H)+(K=H)+(D=H)next H
51 goto 46,D,H,D
52 if A>Lfor H=0to A;@(31+H)=@(H);next H;L=A
53 gosub R,CX,-12,-M;print "(1)_LAST";CX=-12;print "(2)_LONGEST";CX=-12;print "(3)_START",;if G=3for D=1to 4;gosub U,B,3-2x(D=E);next D
54 for I=21to 23;gosub P,CX,-M,-8x(&(28)+86)-M,6;TV=97;if &(I)=Sgoto 56,C,(I-26)
x8
55 next I;if TR(1)=0goto 54
56 gosub P,CX,-M,C,7;TV=97;if C=-40box 0,U,2,2,F-3;NT=0;MU=74;goto 22,NT,P
57 gosub Q,B,N;CX=-42;if C=-Mbox -4,U,2,2,1;print A+1;gosub X,B,0,0;gosub W;box
-4,U,2,2,1;goto 54
58 box F,U,2,2,1;print L+1;for E=0to L;gosub 1,B,0,0,@(31+E);next E;gosub W;box
F,U,2,2,1;goto 54
255 if C=1box -1,6,2,2,2;box 0,U,2,2,2;box 0,9,2,2,2;box 0,6,U,1,2;box 0,5,U,3,3
;XY=FB;data FB,FC,XY;&(2)=FB;&(3)=XY;return
256 if C=2for D=0to 200;next D;return
257 if C=0box 0,8,12,6,4;circle 0,8,6,6;circle 0,8,5,6;C=3;&(U)=200;return
258 gosub R,F,3;box 0,U,13,6,4;box -4,U,2,2,2;box 3,U,2,2,2;box 0,8,1,2,2;CALL28
070;&(17)=200;&(22)=200
259 for K=0to 0step -1;&(V)=K;&(20)=K;data FB,74-K,K;&(2)=FB;&(3)=FC;next K;gosu
b 9;NT=0;BC=263;for J=62to S2+2;MU=@(J)+100;MU=RM;next J;NT=P;BC=7;&(S)=71;retur
n
8192 C=CY;box CX,-32,6,M,4;return
8224 data NT,P,-15,B,7;print "SCORE:";return
10697 box 0,-32,160,M,4
10700 box 0,44,160,S,4;return
11465 for H=0to A;gosub 1,D,@(H);next H;return
ENTER>clear ;for A=62to 166step 1;print "@(",#1,A,;input ")="@(A);next A
ENTER>[NUMBER STRING] PRESSING [GO] AFTER EACH ENTRY
(NOTE) [READ EACH COLUMN DOWN WHEN ENTERING]

```

3232	4350	4848	5198	5198	4898	5498	3232
3232	5153	4851	5148	5298	5098	5198	3232
5348	4851	5051	4398	5198	4998	5048	3232
4848	5352	5348	5098	5055	5648	4848	3232
5348	4848	4854	5198	9850	4848	9849	3232
5251	4850	4854	5148	4848	4898	3232	3232
5148	4848	4850	4848	4898	4998	3232	3232
4350	5050	4848	4848	4948	5098	3232	
5151	4843	4848	4398	9849	5198	3232	
4848	4950	4898	5098	4898	5398	3232	
4848	5048	5348	5198	4948	4998	3232	
4843	4349	9853	5148	5598	5098	3232	
5051	5052	4898	4398	4945	5198	3232	
5148	4848	5298	5098	9851	5353	3232	

```

ENTER>clear ;for A=28070to 28191step 2;print "%(",#1,A,;input ")="%(A);next A
ENTER>[NUMBER STRING] PRESSING [GO] AFTER EACH ENTRY
(NOTE) [READ EACH COLUMN DOWN WHEN ENTERING]

```

-14859	-14488	22479	9474	15949	2199	-14894	274
-3699	-147	20041	27869	9040	10423	-1775	-13271
33	7733	17742	11086	-11502	15559	28806	-1007
14592	27720	8530	11078	6927	-494	28963	6275
-207	28103	-2816	-14082	-11400	10371	29219	236
-6833	-1567	-10811	-14274	4613	-493	29475	
13823	-11807	4581	1568	-11399	8258	-20968	
30	-3647	27873	11086	6913	445	318	

```

ENTER>clear ;for A=27842to 27869step 2;print "%(",#1,A,;input ")="%(A);next A
ENTER>[NUMBER STRING] PRESSING [GO] AFTER EACH ENTRY
(NOTE) [READ EACH COLUMN DOWN WHEN ENTERING]

```

24	-1661
48	10
255	18
8192	10700
8224	11465

10697 10697

16 -1661

ENTER>clear ;A=0

ENTER>print ";goto S";:print %(24576),1810[REC]

# Simon Machine Code

Winner!

INTERRUPT

27866	6CPA	C9-CC	(Green Color 1)	6DBF	6P	(char. string)
	6CPB	29-2C	Red color 2	6DC0	E1	POP HL
27868	6CDC	B3-86	Yellow color 3	1	F9	LD SP, HL
	6CPD	F9-FC	Blue color 4	X(RET)6DC2	E1	POP HL
	6CDF0	42-43	(Flasher Byte)	3	D1	POP DE
27872	6CE0	C9-CC	(FA Variable)	4	C1	POP BC
	6CE1	C9-90	(Scan Line)	5	F1	POP AF
28070	6DAG	ES	Push AF	6	C9	RET
	7	C05	Push BC	(Character String) 7	57	W
	8	D05	Push DE	8	49	I
	9	E9	Push HL	9	4E	N
	A	21	LD HL, 0000	A	4E	N
	B	00	(clear)	(End of L2 Buffer) B	45	E
	C	00	(HL)	C	52	R
	D	39	Add HL, SP	D	21	!
	E	31	LD SP, 4FFF	E	00	(End)
	F	FF	(move stack)	(stack) 28011 6DCE	F5	Push AF
6DB0	4F		(Do screen)	6DD0	C5	Push BC
	1	ES	Push HL	1	D5	Push DE
	2	FF	RET 38	2	E5	Push HL
	3	39	(string disp.)	3	11	LD DE, 6CE1
	4	1E	(X position)	4	E1	(scan line)
	5	00	(Y position)	5	6C	(Address)
	6	68	(magic byte)	6	1A	LD A, (DE)
	7	C7	(Address of)	7	21	LD HL, 6CPD
	8	6D	(char. string)	8	DD	(Bottom)
	9	FF	RET 38	9	6C	(colors)
	A	39	(string disp.)	A	4E	LD C, (HL)
	B	1E	(X position)	B	2B	Dec HL
	C	49	(Y position)	C	46	LD B, (HL)
	D	6C	(magic byte)	D	2B	Dec HL
6DB E	C7		(Address of)	6D E	FE	CP C8

Interrupt

Routine

6DDE	C8	(Bottom Line)
6DE0	3E	LD A, C8
1	C8	(Bottom Line)
2	20	JR NZ, 06
3	06	(To Address)
4	4E	LD C, (HL)
5	2B	Dec HL
6	46	LD B, (HL)
7	3E	LD A, 50
8	50	(Top Line)
9	23	Inc HL
<del>6DEA</del> Add. 6DEA	12	LD (DE), A
<del>6DEB</del> B	D3	Out (OF), A
C	0F	(scan line)
D	1B	Dec DE
E	78	LD A, B
F	D3	Out (09), A
6DF0	09	(Left color)
1	12	LD (DE), A
2	79	LD A, C
3	D3	Out (01), A
4	01	(Right color)
5	1B	Dec DE
6	2B	Dec HL
7	1A	LD A, (DE)
8	B7	OR A
9	28	JR Z, C7
A	C7	(To R Ret.)
B	3C	Inc A
C	12	LD (DE), A
D	FE	CP 83
6DFE	83	(2nd switch)

6DFF	28	JR Z, 13
6E00	13	(To R Address)
1	FE	CP 42
2	42	(1st switch)
3	20	JR NZ, BD
4	BD	(To R Ret.)
5	01	LD BC, C9 2C
6	2C	(color 2 on)
7	C9	(color 1 off)
8	11	LD DE, 86 F9
9	F9	(color 4 off)
A	86	(color 3 on)
<del>6E0B</del> Add. B	70	LD (HL), B
C	23	Inc HL
D	71	LD (HL), C
E	23	Inc HL
F	72	LD (HL), D
6E10	23	Inc HL
1	73	LD (HL), E
2	18	JR AE
3	AE	(To R Ret.)
<del>6E14</del> Add. 6E14	3E	LD A, 01
5	01	(Restart Counter)
6	12	LD (DE), A
7	01	LD BC, C9 29
8	29	(color 2 off)
9	CC	(color 1 on)
A	11	LD DE, 83 FC
B	FC	(color 4 <del>on</del> )
C	83	(color 3 <del>on</del> ) off
D	18	JR EC
6E1E	EC	(To R Address)

Character  
number code  
SIMON

190 FB	2E26	00
7	1F	
8	00	
9	00	
A	00	
B	E4	
C	E0	
D	00	
E	03	
F	04	
2E30	18	
1	00	
2	0C	
3	04	
4	06	
5	00	
6	10	
7	04	
8	01	
9	00	
A	10	
B	04	
C	01	
D	00	
2E3E	20	

F	04	
2E40	00	
1	80	
2	20	
3	04	
4	00	
5	80	
6	40	
7	0E	
8	00	
9	40	
A	40	
B	31	
C	80	
D	40	
E	40	
F	40	
2E50	40	
1	40	
2	80	
3	40	
4	40	
5	20	
6	80	
2E57	80	

2E58	20	
9	20	
A	FF	
B	80	
C	3F	
D	E0	
E	80	
F	80	
2E60	20	
1	20	
2	80	
3	40	
4	40	
5	20	
6	40	
7	40	
8	40	
9	40	
A	40	
B	31	
C	80	
D	40	
E	40	
F	0E	
2E70	00	

2E71	40	
2	20	
3	04	
4	00	
5	80	
6	20	
7	04	
8	00	
9	80	
A	10	
B	04	
C	01	
D	00	
E	10	
F	04	
2E80	01	
1	00	
2	0C	
3	04	
4	06	
5	00	
6	03	
7	04	
8	18	
2E89	00	

FB	ZEBA	00
	8	E4
	C	E0
	D	00
	E	00
	F	1F
	ZE90	00
	1	00
<hr/>		
Fc	1812E92	00
	3	00
	4	00
	5	00
	6	00
	7	1B
	8	00
	9	00
	A	00
	B	FB
	C	E0
	D	00
	E	03
	F	FB
	ZEAO	FB
	1	00
	ZEAZ	0F

ZEAZ	FB
4	FE
5	00
6	0F
7	FB
8	FE
9	00
A	1F
B	FB
C	FF
D	00
E	1F
F	FB
ZEBO	FF
1	00
2	3F
3	F1
4	FF
5	80
6	3F
7	C0
8	7F
9	80
A	3F
ZEBB	80

ZEBC	3F
D	80
E	7F
F	80
ZECC	3F
1	C0
2	7F
3	00
4	1F
5	C0
6	00
7	00
8	00
9	00
A	7F
B	00
C	1F
D	C0
E	7F
F	80
ZECD	3F
1	C0
2	3F
3	80
4	3F

ZECS	80
6	3F
7	C0
8	7F
9	80
A	3F
B	F1
C	FF
D	80
E	1F
F	FB
ZEEO	FF
1	00
2	1F
3	FB
4	FF
5	00
6	0F
7	FB
8	FE
9	00
A	0F
B	FB
C	FE
ZEED	00

Character  
Number  
Code

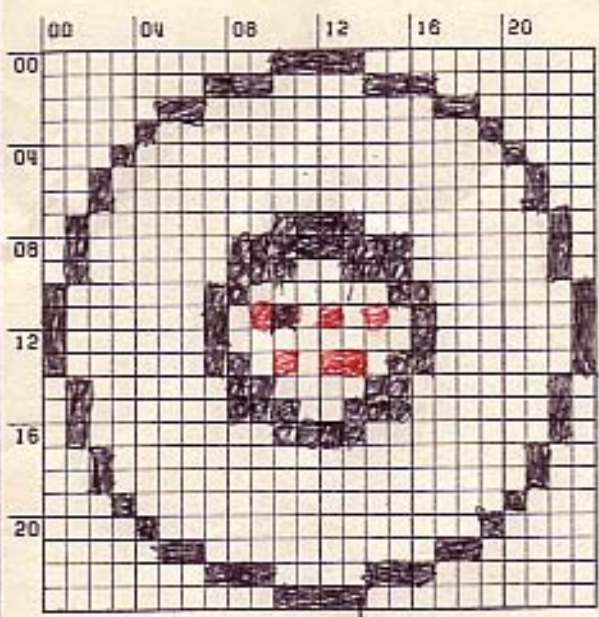
Gimon

FCV	2EEE	03	2F07	<del>7</del>	00	2F20	00	2F39	00
	F	FB	8		00	1	00	A	00
	2EFO	FG	9		00	2	00	B	00
	1	00	A		00	3	00	C	00
	2	00	B		00	4	00	D	00
	3	FB	C		00	5	00	E	00
	4	EO	D		00	6	00	F	00
	5	00	E		00	7	00	2F40	00
	6	00	F		00	8	00	1	00
	7	1B	2F10		00	9	00	2	00
	8	00	12		00	A	00	3	00
	9	00	2		00	B	00	4	00
	A	00	3		00	C	00	5	00
	B	00	4		00	D	00	6	00
	C	00	5		00	E	00	7	00
	D	00	6		00	F	15	8	00
BC 192	2EFE	00	7		00	2F30	00	9	00
	F	00	8		00	1	00	A	00
	2F00	00	9		00	2	00	B	00
	1	00	A		00	3	00	C	00
	2	00	B		00	4	00	D	00
	3	00	C		00	5	00	E	00
	4	00	D		00	6	00	F	00
	5	00	E		00	7	13	2F90	00
	2F06	00	2F1F		00	2F7B	00	2F91	00



icv			Font	
	2F52	00		80
	3	00		00
	4	00		00
	5	00		04
	6	00		16
	7	00		26
	8	00		2E
	9	00		
	A	00		
	B	00		
	C	00		
	D	00		
	E	00		
	F	00		
	2F60	00		
	1	00		
	2	00		
	3	00		
	4	00		
	5	00		
	6	00		
	7	00		
	8	00		
	2F69	00		





COLOR MAP

11-

10-

01-

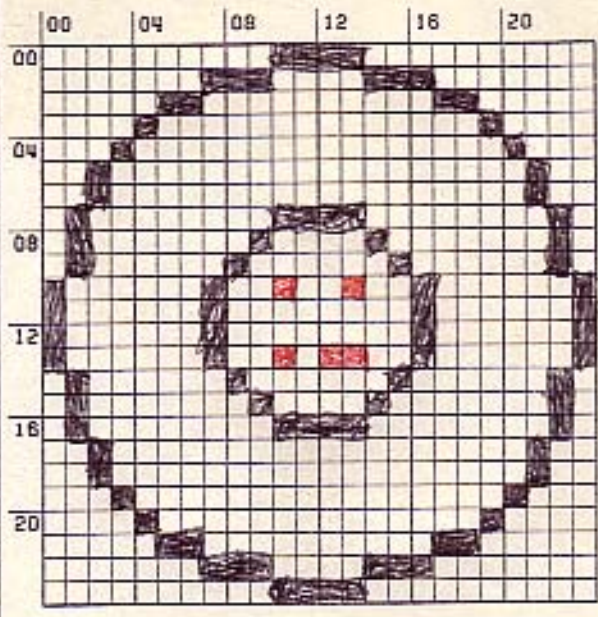
00-

PATTERN DATA

VECTOR OFFSETS

SIZE

ADDRESS BYTES



COLOR MAP

11-

10-

01-

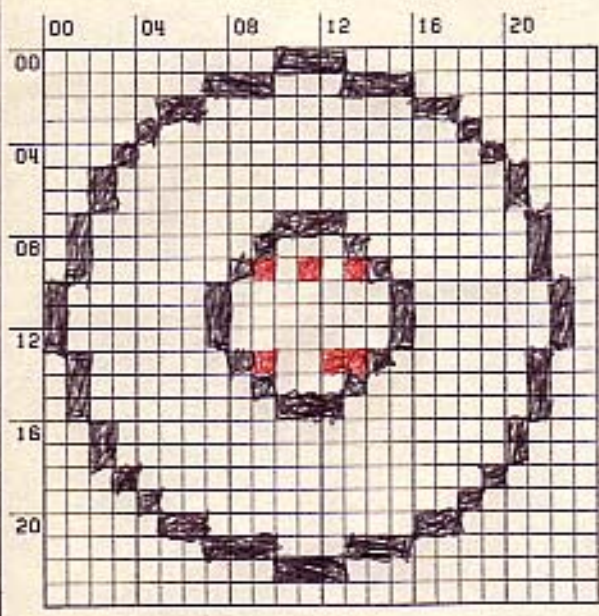
00-

PATTERN DATA

VECTOR OFFSETS

SIZE

ADDRESS BYTES



COLOR MAP

11-

10-

01-

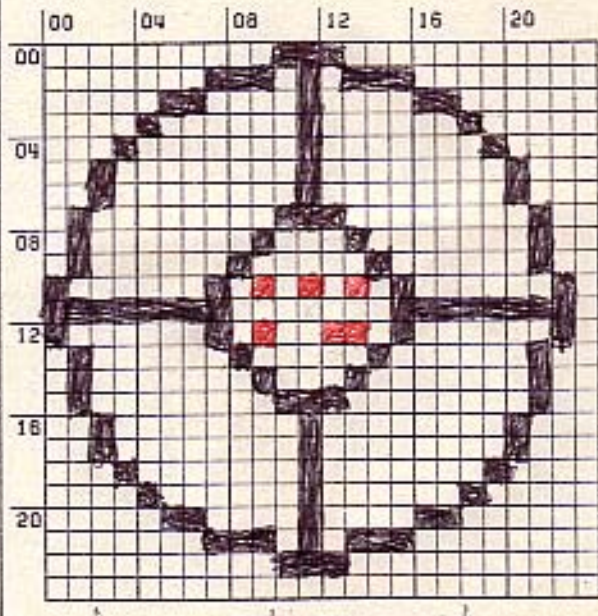
00-

PATTERN DATA

VECTOR OFFSETS

SIZE

ADDRESS BYTES



COLOR MAP

11-

10-

01-

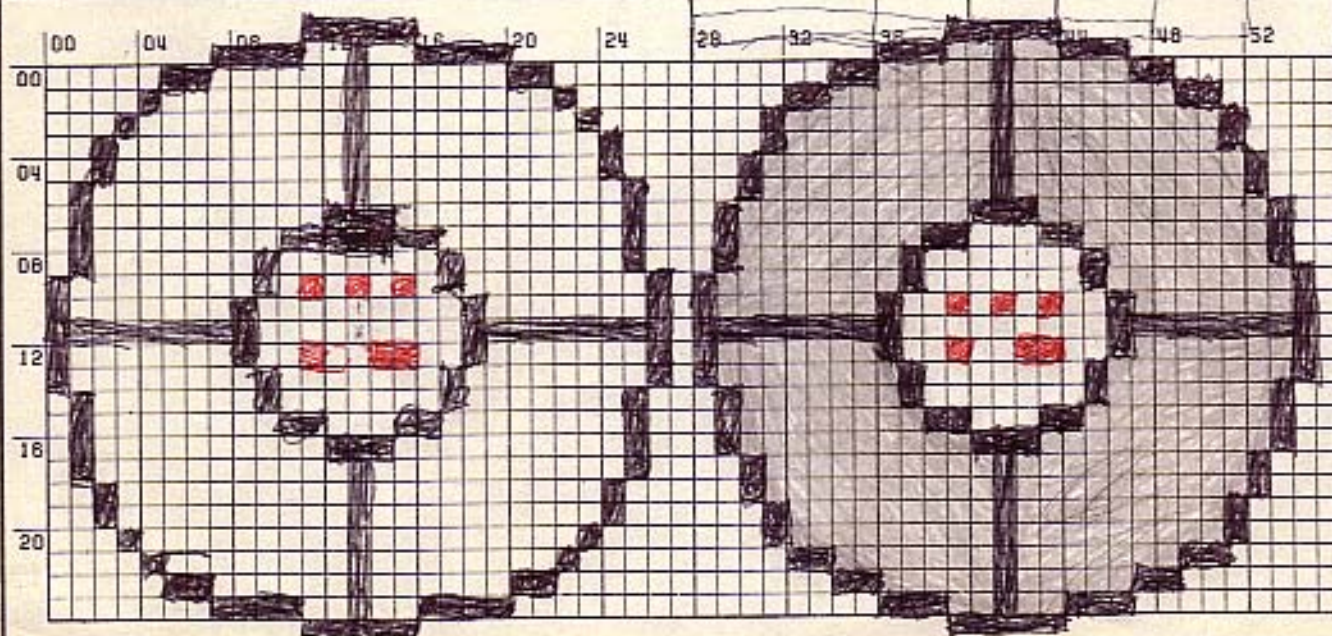
00-

PATTERN DATA

VECTOR OFFSETS

SIZE

ADDRESS BYTES



COLOR MAP

11-

10-

01-

00-

PATTERN DATA

VECTOR OFFSETS

SIZE

ADDRESS BYTES