

```

1  ^TRAC1M-1P(190801)

10  CLV:OUT0
20  LET[1],1,3,2,0:LET[11],-1,1,0,0,0,-1,1
30  H=2:D=#8A0:V=#900:X=27:Y=21
40  POKE#828,64,24,24,34,96,36,3,27,1,120,25,112,32,48,1,51,1,58,
249,209,112,71
50  POKE#840,64,24,24,34,127,36,3,25,1,120,25,112,1,48,32,51,1,58,
249,209,112,71
60  LC0,0:"READY";
70  IFINKEY()=10ORIN()&7=7OUT7:CLSELSECONT
80  BEEP10,10:LC15,12:"●":LCX,Y:"◎":WAIT20

100  @1P
110  K=INKEY():K=(K)=28ANDK<=31)*(K-27):IF!KK=HELSEH=K
120  IFK<=2GOTO@YK
130  IFK>=3GOTO@TT

200  @YK
210  POKE #82C,96+(2-K)*31:A=#900+(2-K)*31:A=USR(#828,A)
220  SCROLL[K]
230  POKE #844,127-(2-K)*31:A=USR(#840,#8A0)
240  X=X+(2-K)*2-1
250  GOTO400

300  @TT
310  COPYD,V+(32*23)*(4-K),32
320  SCROLL[K]
330  COPY#BE0-(32*23)*(4-K),D,32
340  Y=Y+(4-K)*2-1

400  OUTK+8:IFPEEK(#A8F)LC15,12:"*":LC15,10:"LOSE":RUN
410  LC15,12:"●":WAIT2
420  OUT4,0

500  @2P
510  IF!IN(4)CONT
520  E=IN()&7
530  X=(X+[E+10]+32)%32
540  Y=(Y+[E+14]+24)%24
550  IFSCR(X,Y):LCX,Y:"*":LC15,10:"WIN":RUN
560  POKEY+X+Y*32,#E8
570  IFIN(4)CONTELSELED1:BEEPK*10:LED0

600  GOTO@1P

```

ジャンプワイヤーの結線

1P	2P
IN1	OUT1
IN2	OUT2
IN3	OUT3
IN4	OUT4
GND	GND
OUT1	IN1
OUT2	IN2
OUT3	IN3
OUT4	IN4

```

1  ^TRAC1M-2P(190801)

10  CLV:OUT0
20  LET[1],1,3,2,0:LET[11],-1,1,0,0,0,-1,1
30  H=1:D=#8A0:V=#900:X=3:Y=3
40  POKE#828,64,24,24,34,96,36,3,27,1,120,25,112,32,48,1,51,1,58,
249,209,112,71
50  POKE#840,64,24,24,34,127,36,3,25,1,120,25,112,1,48,32,51,1,58,
249,209,112,71
60  LC0,0:"READY";
70  IFINKEY()=10ORIN()&7=7OUT7:CLSELSECONT
80  BEEP10,10:LC15,12:"○";:LCX,Y:"●";:WAIT20

100  @1P
110  IF!IN(4)CONT
120  E=IN()&7
130  X=(X+[E+10]+32)%32
140  Y=(Y+[E+14]+24)%24
150  IFSCR(X,Y):LCX,Y:"*";:LC15,10:"WIN":RUN
160  POKEV+X+Y*32,#E9
170  IFIN(4)CONTELSELED1:BEEPK*10:LED0

200  @2P
210  K=INKEY():K=(K>=28ANDK<=31)*(K-27):IF!KK=HELSEH=K
220  IFK<=2GOTO@YK
230  IFK>=3GOTO@TT

300  @YK
310  POKE#82C,96+(2-K)*31:A=#900+(2-K)*31:A=USR(#828,A)
320  SCROLL[K]
330  POKE#844,127-(2-K)*31:A=USR(#840,#8A0)
340  X=X+(2-K)*2-1
350  GOTO500

400  @TT
410  COPYD,V+(32*23)*(4-K),32
420  SCROLL[K]
430  COPY#BE0-(32*23)*(4-K),D,32
440  Y=Y+(4-K)*2-1

500  OUTK+8:IFPEEK(#A8F)LC15,12:"*";:LC15,10:"LOSE":RUN
510  LC15,12:"○";:WAIT2
520  OUT4,0

600  GOTO@1P

```

40 行のマシン語部分：画面左端一列のキャラを#8A0～(配列[80]～)にコピー

A=USR(#828,#900) ^このルーチンの開始アドレス#828(配列[20])

R0 = R0 + R1 ^物理アドレス変換
R2 = 24 ^回数カウンタ
R4 = 96
R3 = R0 - R4 ^R3に配列[80]の物理アドレス代入

```

@LOOP
R1 = [R0]
[R3] = R1
R0 += 32
R3 += 1
R2 -= 1
IF !0 GOTO @LOOP
RET

```

スクロール方向に合わせて一部書き換え。50 行の「配列から VRAM への書き戻し」もほぼ同様の処理。