PROMBLOWER KIT

This promblower enables you to blow programs into DM74S571 fusible link PROMs. It is designed to work off the RAM/IO device on an MK14 although it should be possible to use it with other systems.
The power requirements are +5V regulated (from the MK14 board) and between 11V and 15V (which can be the unstabilised input to the MK14 regulator). Two programmes are provided with this kit.

1. The prom-editor programme behaves in a similar way to the MK14 monitor. It enables you to read the contents of a selected location in the PROM (which is assumed to fill the address space ØØØØ to Ø1FF and also to blow the contents of a given address. This is useful for checking the contents of a PROM or for making minor adjustments. The programme is 94 bytes long and is relocatable. A suitable location is ØF2Ø to ØFB4. See figure 1 for a description of its operation.

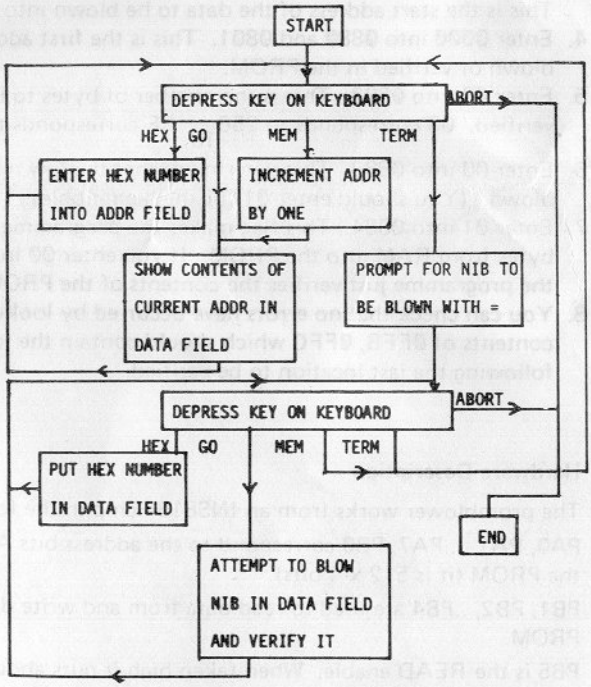


fig. 1

Since the PROMS are 512 x 4 one can only blow half of a byte into a given PROM (the high or low 'nibble' of the byte).

2. The promblower transfers a programme from RAM into a PROM. It blows either the high or low nibble of the bytes of the programme into one PROM. The programme is 80 bytes long and is re-locatable. It fits conveniently into the RAM of the RAM/IO chip. As it stands it can only blow up to 256 bytes at a time so blowing a PROM requires two runs of the programme i.e. from 0000 to 00FF and 0100 to 01FF. Since the programme verifies the blowing as it goes along it has been designed to be used simply to verify a PROM against a programme in RAM. Typical operation is described below.

1. Enter the programme to be blown into the extra RAM space 0B00 - 0BFF
2. Enter 0880 into 0FF9 and 0FFA (the monitor will put this in P1)
3. Enter 0B00 into 0FFB and 0FFC (the monitor will put this in P2) This is the start address of the data to be blown into the PROM
4. Enter 0000 into 0880 and 0801. This is the first address to be blown or verified in the PROM.
5. Enter 00 into 0882. This is the number of bytes to be blown or verified. 00 corresponds to 256_{10}, FF corresponds to 255_{10} etc.
6. Enter 00 into 0883. This determines that the low nibble will be blown. (You should enter 01 for the high nibble).
7. Enter 01 into 0884. This flag makes the programme blow the bytes from RAM into the PROM. If you enter 00 in this location the programme just verifies the contents of the PROM against RAM.
8. You can check that no errors have occurred by looking at the contents of 0FFB, 0FFC which should contain the location following the last location to be verified.

Hardware Description

The promblower works from an INS8154 chip in the following way:-

PA0, PA1 ....PA7, PB0 correspond to the address bits AB0→AB8 for the PROM (it is 512 x 4 bits).

PB1, PB2, ..PB4 are used to read data from and write data to the PROM.

PB5 is the READ enable. When taken high it puts about 4 volts across the PROM, enables the PROM for a short time (about 100μS) and opens the 4016 transmission gate so that data can be read on PB1 PB4.

PB6 is the WRITE enable. When taken high the promblower puts about 10.5V across the PROM, closes the transmission gates (even if PB5 is high) and waits about 10μS before enabling the PROM for 10μS.* If any

of the lines PB1→PB4 are high when the PROM is enabled the corresponding data line will have about 10.5V volts on it and the appropriate bit becomes blown.

*The hardware timing is performed by the CMOS 4001 and various resistors and capacitors.

Software Description of Promblower Programme

1. The promblower programme first looks to see if the flag HIGH is set and then selects the appropriate nibble of the location pointed to by P1. P1 is then auto-indexed to point to the next location. The nibble is stored as the variable NIB.
2. The variables REPEATS and ATTEMPTS are then both set to 5. This allows for 5 attempts to blow a location in the PROM before reporting an error. It also arranges for the location to be blown four times after a successful blow to ensure proper blowing.
3. The variable MASK determines which bit of the nibble is being blown. If BLOW=0 the MASK is zeroed and the PROM location just verified against the variable NIB. If BLOW = 0 a nibble is to to be blown and MASK is originally set to X'08 (i.e. the left hand bit of the nibble is blown first).
4. The I/O chip is then configured as all outputs except for PB1→PB4 which will read data. The outputs are all taken low. If there have been too many attempts to blow a location the programme will abort here. If the location has been blown and verified five times (REPEATS=0) the variable LEN is decremented by one and if non-zero the next nibble is read using P1 as indicated above in section 1. The programme then takes the READ enable (PB5) high and puts the next address to be blown in the PROM on PA0 PB0.
5. If MASK=0 (just verifying or blowing complete) the programme reads the nibble in the PROM on PB1→PB4 and compares it with NIB. If it is correct the variable REPEATS is reduced by one. If it is incorrect the variable ATTEMPTS is reduced by one. It then loops back to 3.
6. If MASK ≠0 the programme waits for the READ enable to clear before configuring PB1→PB4 as outputs. It then puts MASK.AND.NIB on PB1→PB4 (the next bit of the nibble to be blown) and takes WRITE enable high. This should blow the appropriate bit in the PROM. MASK is the shifted right and we return to 4).

PROMBLOWER AND VERIFIER PROGRAMME

```
0880    PROM      =        0880
0882    LEN       =        0882
0883    HIGH      =        0883
0884    BLOW      =        0884
0885    MASK      =        0885
0886    NIB       =        0886
0887    REPEATS   =        0887
0888    ATTEMPTS  =        0888
0000              =        00889
0889    C103      GETNIB   LD HIGH(1)
088B    9808               JZ LOW
088D    C601               LD @1(2)
088F    1C                 SR
0890    1C                 SR
0891    1C                 SR
0892    1C                 SR
0893    9004               JMP GOTNIB
0895    C601      LOW      LD @1(2)
0897    D40F               ANI X'0F
0899    C906      GOTNIB   ST NIB(1)
089B    C405               LDI X'05
089D    C907               ST REPEATS(1)
089F    C908               ST ATTEMPTS(1)
08A1    C104      NEWMASK  LD BLOW(1)
08A3    9804               JZ NEXTBIT
08A5    C408               LDI X'08
08A7    C905      NEXTBIT  ST MASK(1)
08A9    C4FF               LDI X'FF
08AB    C9A2               ST ODA(1)
08AD    C4E1               LDI X'E1
08AF    C9A3               ST ODB(1)
08B1    C400               LDI X'00
08B3    C9A0               ST PORTA(1)
08B5    C9A1               ST PORTB(1)
08B7    C108               LD ATTEMPTS(1)
08B9    9808               JZ END
```

IF HIGH = 0 THEN BLOW LOW NIBBLE

GET HIGH NIBBLE

GET LOW NIBBLE

STORE NIBBLE
ALLOW FOR 5 ATTEMPTS TO BLOW AND FOR 5 REPEATS

IF JUST VERIFYING SET MASK TO ZERO

MASK FOR LEFT HAND BIT OF NIBBLE

CONFIGURE I0 PORTS FOR VERIFICATION

ZERO ALL LINES TO PROM BLOWER

IF THERE HAVE BEEN 5 ATTEMPTS THEN GIVE UP

```
08BB  C107            LD REPEATS(1)      IF THERE HAVE BEEN LESS THAN 5
08BD  9C0D            JNZ SETPORT        REPEATS BLOW THE NIBBLE AGAIN
08BF  B902            DLD LEN            SEE IF THERE ARE MORE NIBBLES TO BLOW
08C1  9C01            JNZ NEXT
08C3  3F      END     XPPC 3             FINISH
08C4  A901    NEXT    ILD L(PROM)        INCREMENT PROM ADDRESS
08C6  9CC1            JNZ GETNIB
08C8  A900            ILD H(PROM)
08CA  9OBD            JMP GETNIB
08CC  02      SETPORT CCL                SET UP LINES TO READ FROM PROM
08CD  C100            LD H(PROM)
08CF  F420            ADI X'20
08D1  C9A1            ST PORTB(1)
08D3  C101            LD L(PROM)
08D5  C9A0            ST PORTA(1)
08D7  C105            LD MASK(1)         IF MASK IS ZERO JUST VERIFY THIS PROM ADDRESS
08D9  9814            JZ VERIFY
08DB  D106    BITBLOW AND NIB            MASK OFF ONE BIT OF NIBBLE
08DD  01              XAE
08DE  8F01            DLY X'01           WAIT FOR READ ENABLE TO CLEAR
08E0  C9A3            ST ODB(1)
08E2  40              LDE
08E3  70              ADE
08E4  F460            ADI X'60
08E6  F100            ADD H(PROM)
08E8  C9A1            ST PORTB(1)        BLOW THIS BIT OF NIBBLE
08EA  C105            LD MASK(1)         SHIFT MASK TO RIGHT
08EC  1C              SR
08ED  90B8            JMP NEXTBIT
08EF  C1A1    VERIFY  LD PORTB(1)        READ PROM
08F1  1C              SR
08F2  D40F            ANI X'0F           CHECK IF NIBBLE IS CORRECT
08F4  E106            XOR NIB
08F6  9804            JZ OK
08F8  B908            DLD ATTEMPTS(1)    NIBBLE BAD SO MAKE ANOTHER ATTEMPT
08FA  90A5            JMP NEWMASK
08FC  B907    OK      DLD REPEATS(1)     NIBBLE OK SO REPEAT IF NECESSARY
08FE  90A1            JMP NEWMASK
```

PROM EDITOR PROGRAMME

```
ØF2Ø  C4ØF   START     LDI H(RAM)          POINT P2 AT RAM
ØF22  36               XPAH 2
ØF23  C4ØØ             LDI L(RAM)
ØF25  32               XPAL 2
ØF26  C4Ø1   DISPLAY   LDI H(DISPD)        SHOW ADDRESS CONTENTS AND WAIT FOR KEY
ØF28  37               XPAH 3              HEX KEY IS ENTERED INTO ADDRESS FIELD
ØF29  C43F             LDI L(DISPD)-1      MEM INCREMENTS ADDRESS BY ONE
ØF2B  33               XPAL 3              TERM SHIFTS TO BLOWING ROUTINE
ØF2C  3F               XPPC 3              GO CONFIRMS CONTENTS OF ADDRESS
ØF2D  9ØØ6             JMP ADROK           COMMAND KEY DEPRESSED
ØF2F  C41A             LDI L(ADR)-1        SHIFT HEX NUMBER INTO ADDRESS FIELD
ØF31  33               XPAL 3
ØF32  3F               XPPC 3
ØF33  9ØF1             JMP DISPLAY
ØF35  E4Ø3   ADROK     XRI X'Ø3            TERM KEY DEPRESSED
ØF37  9811             JZ BLOW
ØF39  4Ø               LDE
ØF3A  E423             XRI X'23            GO KEY DEPRESSED
ØF3C  9C06             JNZ NOBLOW          MEM KEY DEPRESSED
ØF3E  AA0C             ILD ADL(2)
ØF4Ø  9C02             JNZ NOBLOW
ØF42  AA0E             ILD ADH(2)
ØF44  C4ØØ   NOBLOW    LDI X'ØØ            ZERO MASK FOR SIMPLE VERIFICATION
ØF46  CA12             ST MASK(2)
ØF48  901C             JMP DOIO            PUT '=' PROMPT ON DISPLAY
ØF4A  C448   BLOW      LDI X'48
ØF4C  CAØ2             ST D3(2)            HEX KEY BECOMES NIBBLE TO BE BLOWN
ØF4E  C43F             LDI L(DISPD)-1      GO KEY CAUSES IT TO BE BLOWN
ØF5Ø  33               XPAL 3              MEM OR TERM KEYS GET OUT OF BLOW ROUTINE
ØF51  3F               XPPC 3              COMMAND KEY DEPRESSED
ØF52  9ØØ5             JMP CHKBLOW         HEX KEY STORED AS NIBBLE TO BE BLOWN
ØF54  4Ø               LDE
ØF55  CA0D             ST WORD(2)
ØF57  90F1             JMP BLOW            REMOVE '=' PROMPT ON DISPLAY
ØF59  C4ØØ   CHKBLOW   LDI X'ØØ
ØF5B  CAØ2             ST D3(2)
ØF5D  4Ø               LDE                 MEM OR TERM KEY DEPRESSED
ØF5E  E422             XRI X'27            GO KEY DEPRESSED
ØF6Ø  9CE2             JNZ NOBLOW
ØF62  C4Ø8             LDI X'Ø8
```

```
0F64  CA12          ST MASK(2)
0F66  C408  DOIO    LDI H(IO)
0F68  35            XPAH 1
0F69  C400          LDI L(IO)
0F6B  31            XPAL 1
0F6C  C4FF          LDI X'FF
0F6E  C922          ST ODA(1)
0F70  C923          ST ODB(1)
0F72  C20E  POWERUP LD ADH(2)
0F74  D401          ANI X'01
0F76  CA0E          ST ADH(2)
0F78  F420          ADI X'20
0F7A  C921          ST PORTB(1)
0F7C  C20C          LD ADL(2)
0F7E  C920          ST PORTA(1)
0F80  C212          LD MASK(2)
0F82  9C13          JNZ BITBLOW
0F84  C4E1  VERIFY  LDI X'E1
0F86  C923          ST ODB(1)
0F88  C121          LD PORTB(1)
0F8A  1C            SR
0F8B  D40F          ANI X'0F
0F8D  CA0D          ST WORD(2)
0F8F  C400          LDI X'00
0F91  C920          ST PORTA(1)
0F93  C921          ST PORTB(1)
0F95  9089          JMP START
0F97  02    BITBLOW CCL
0F98  D20D          AND WORD(2)
0F9A  01            XAE
0F9B  8F01          DLY X'01
0F9D  40            LDE
0F9E  70            ADE
0F9F  F440          ADI X'60
0FA1  F20E          ADD ADH(2)
0FA3  C921          ST PORTB(2)
0FA5  8F01          DLY X'01
0FA7  C400          LDI X'00
0FA9  C920          ST PORTA(1)
0FAA  C921          ST PORTB(1)
0FAD  C212          LD MASK(2)
0FAF  1C            SR
0FB0  CA12          ST MASK(2)
0FB2  90BE          JMP POWERUP
```

SET UP MASK FOR MOST SIGNIFICANT BIT OF NIBBLE
POINT P1 AT IO AREA

CONFIGURE BOTH PORTS FOR OUTPUT

TRUNCATE ADDRESS TO LIE IN THE RANGE 0000 TO 01FF

SET UP READ LINE AND HIGH ADDRESS BIT FOR PROMBLOWER

SEND LOW ADDRESS BITS TO PROMBLOWER

IF MASK IS NON ZERO BLOW A BIT

CONFIGURE PORTB FOR READING NIBBLE

READ NIBBLE AND STORE FOR DISPLAY

ZERO ALL SIGNALS TO PROMBLOWER

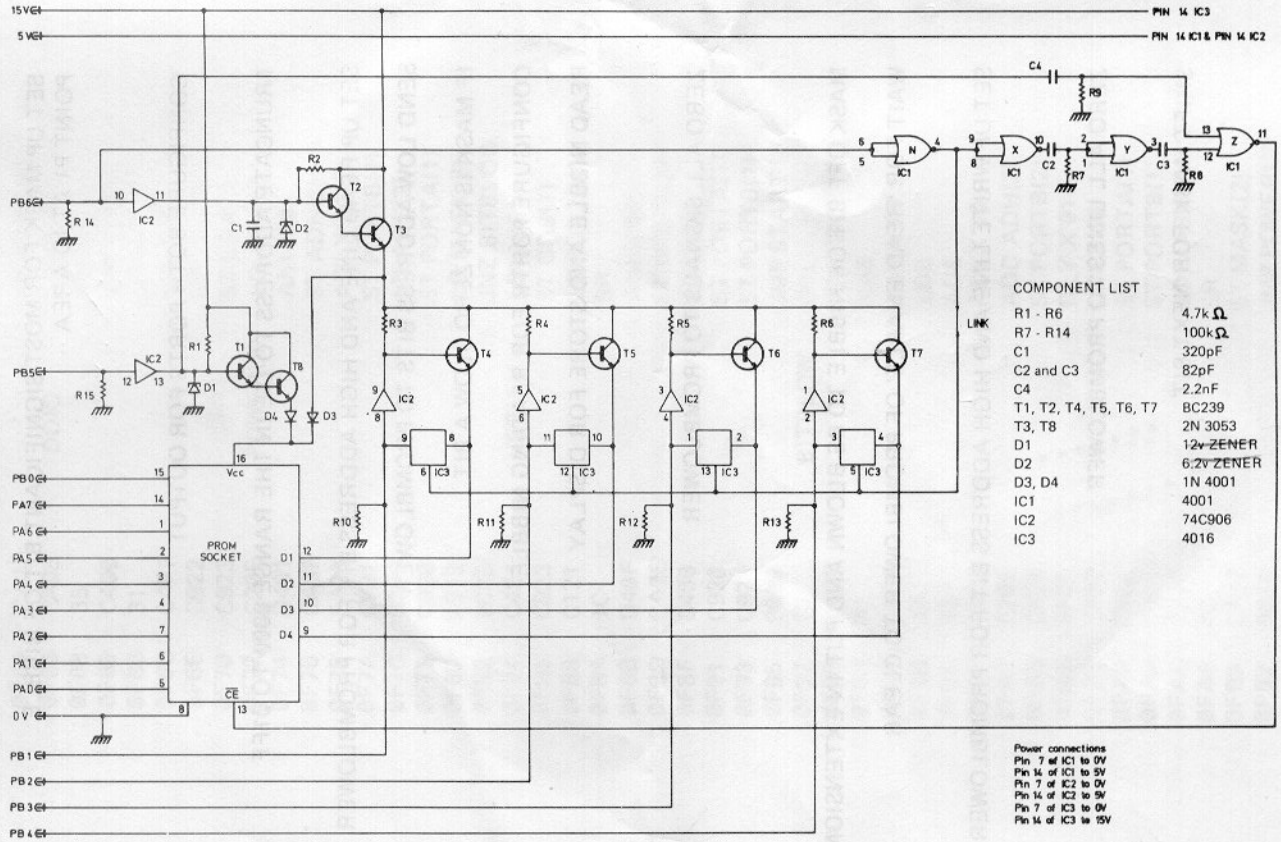
MASK OFF BIT OF NIBBLE TO BE BLOWN AND PUT IN EXTENSION REGISTER

WAIT FOR 'READ ENABLE' OF PROMBLOWER TO CLEAR

SET UP WRITE LINE AND HIGH ADDRESS BIT FOR PROMBLOWER

ZERO ALL LINES TO PROMBLOWER

SHIFT MASK FOR NEXT BIT



COMPONENT LIST

R1 - R6	4.7k Ω
R7 - R14	100k Ω
C1	320pF
C2 and C3	82pF
C4	2.2nF
T1, T2, T4, T5, T6, T7	BC239
T3, T8	2N 3053
D1	~~12v ZENER~~ 6.2V
D2	~~6.2V ZENER~~ 12V
D3, D4	1N 4001
IC1	4001
IC2	74C906
IC3	4016

Power connections
Pin 7 of IC1 to 0V
Pin 14 of IC1 to 5V
Pin 7 of IC2 to 0V
Pin 14 of IC2 to 5V
Pin 7 of IC3 to 0V
Pin 14 of IC3 to 15V

PROM PROGRAMMER CIRCUIT

SCIENCE OF CAMBRIDGE

ISSUE 2 9.1.1979