



RAISE THE ALARM

In the last section of Workshop, we designed and built a mains relay box that allows us to control mains voltage devices with suitable software. In this instalment, we look at the design of some programs to use the mains relay box and demonstrate some simple domestic applications.

The mains relay box is designed to control a mains supply to any device that is plugged into it. In response to a low voltage signal, the box makes or breaks the mains power fed to the socket mounted on the box. The mode of operation is such that the mains supply to the socket is maintained while there is a low voltage current supplied to the relay. Therefore, we can trigger the relay directly from the low voltage output box we built earlier in the course (see page 574). The supply of mains power from the relay box will mirror exactly the low voltage current supplied to the relay from the low voltage output box. Thus, the control of mains supply can be achieved by the same software techniques used to control low voltage devices.

If, for example, the mains relay's low voltage leads are connected to the positive and negative connections of line 0 on the output box, and it is plugged into a mains socket, then a mains current will be supplied to the socket on the relay box when bit 0 of the user port data register is sent high. Whenever bit 0 is sent low then the mains supply to the relay box's socket will stop. Up to four mains relay boxes can be connected to the low voltage output box and switched in this way.

We can make use of this simple switching arrangement to develop a number of control systems that make use of everyday household appliances. First, let's try the following simple project, in which we make use of a tape recorder to program your micro to respond 'verbally' to pressure on a pad.

To begin, we need to record a series of phrases, such as 'You're treading on my pad', followed by 'You've just done it again' and 'Look, I'm warning you!', and so on. Once the messages have been recorded, we will connect the pad and recorder to our user port system and write some software to trigger off the phrases, one at a time, in response to repeated pressure on the pad.

We have to make the following connections to the user port system:

- 1) Plug the mains relay voltage leads into the positive and negative terminals of line 0 on the low voltage output box.
- 2) Plug the supply lead to the mains relay box into a wall socket and switch it on.

- 3) Connect the two pressure pad leads across the positive and negative terminals of line 7 of the buffer box.

The main problem in designing the software to run this system is ensuring that the tape recorder is switched on and off with precision when a message is played. Before we can write a program therefore we must accurately time each message and enter this data into the controlling program. Timing can be done using your micro's internal timer or a stopwatch. If there are three phrases on the tape lasting for periods of T(1), T(2) and T(3) seconds then we can write a program which, on activation from the pressure pad, turns on the tape recorder for the correct time period for each successive message. If the timing of the phrases is done accurately then each phrase should be ready to start when the tape recorder is switched on.

The following programs (for the Commodore 64 and the BBC Micro) will turn the tape recorder on for three successive time intervals – T(1), T(2) and T(3) – in response to triggers from the pressure pad. You must set these variables to your own timed values for the tape you record.

BBC Micro

```

10 REM BBC TREADING PROGRAM
20 DIM T(3)
30 DDR=&FE62:DATREG=&FE60
40 ?DDR=127:REM L7 INPUT
50 ?DATREG=0:REM ALL OFF
60 CLS
70 FOR I=1 TO 3
80 INPUT"TIME INTERVAL (SECS)":T(I)
90 NEXT I
100 :
110 FOR L=1 TO 3
120 CLS
130 REPEAT
140 UNTIL(?DATREG AND 128)=0:REM L7 LOW
150 ?DATREG=1:REM TURN ON TAPE
160 TIME=0:REM START TIMER
170 REPEAT
180 UNTIL TIME>T(L)*100
190 ?DATREG=0:REM TURN TAPE OFF
200 NEXT L
210 END

```

Commodore 64

```

10 REM CBM 64 TREADING PROGRAM
20 DD=56579:DATREG=56577
30 POKEDDR,127:REM L7 INPUT
40 POKEDATREG,0:REM ALL OFF
50 PRINTCHR$(147):REM CLEAR SCREEN
60 FOR I=1 TO 3
70 INPUT"TIME INTERVAL (SECS)":T(I)
80 NEXT I
90 :
100 FOR L=1 TO 3
110 IF(PEEK(DATREG)AND128)<>0 THEN 110
115 POKEDATREG,1:REM TURN TAPE ON
120 T=T:REM INITIALISE TIMER
130 IF T(L)>>(T1-T)/60 THEN130
140 POKEDATREG,0:REM TURN TAPE OFF
150 NEXT L
160 END

```