



RND(100) can be as small as $1-50 = -49$.

By repeatedly MOVING to the centre of the explosion and then drawing a relative line up to 50 units in any direction and colour, we can produce a visually exciting effect.

Now let's consider how we can produce sound effects to accompany the explosive graphics on screen. It is possible to create quite sophisticated sounds using BBC BASIC's SOUND and ENVELOPE commands. SOUND can generate either pitched notes (so you can play a tune) or 'noise' for special sound effects. The ENVELOPE command is used in conjunction with SOUND to 'shape' the sound you hear, and thus simulate musical instruments.

SOUND has four numbers (or parameters) associated with it, which control the characteristics of the tone produced. These are as follows:

SOUND C,A,P,D

- C is the channel number, which must be a value between 0 and 3. Channels 1 to 3 produce pitched notes but channel 0 is for special sound effects. This is the channel that we shall use here.
- A is the amplitude, or, in plain terms, the volume of sound produced. Maximum volume is set when A is -15 and ranges to silence when A is zero. Positive values of A are used to select the shape of the sound and are used together with the ENVELOPE command.
- P is usually the pitch of the note — in other words how high or low the sound is.
- D is the duration or length of time that the tone sounds for. Values of D from 0 to 254 cause the note to hold for a time measured in units of 1/20th of a second. Hence, to hold a note for one second the value of D must be set to twenty. If D has the value -1 then the sound will continue until you take action to stop it.

The explosion sound effect clearly needs to have maximum volume, and thus our command will have the maximum amplitude, signified by a value of -15. The pitch value is slightly more complex. When used with channel 0, P takes values from 0 to 7 according to these rules:

P=0	High frequency pulse
P=1	Middle frequency pulse
P=2	Low frequency pulse
P=3	Pulse, frequency set by channel 1
P=4	High frequency noise
P=5	Middle frequency noise
P=6	Low frequency noise
P=7	Noise, frequency set by channel 1

The group of four pulse waveforms produce 'buzzer'-type sounds, whereas the noise waveforms produce what sounds like a television set before it has been correctly tuned. The SOUND command we will use in our program is:

```
SOUND 0,-15,4,40
```

which produces a high pitched hiss, lasting two seconds (40/20) with the volume set to maximum. Here is the complete listing for the procedure:

```
3550DEF PROCexplode(x_explode,y_explode)
3560REM ** SOUND EFFECT **
3570SOUND 0,-15,6,50
3580REM ** SET FLASH RATE **
3590*FX9,20
3600*FX10,50
3610FOR I=1 TO 100
3620MOVE x_explode,y_explode
3630VDU19,2,RND(15),0,0,0
3640GCOL 0,RND(3)
3650PLOT 1,RND(100)-50,RND(100)-50
3660NEXT I
3670PROCreset
3680ENDPROC
```

THE 'RESET' PROCEDURE

After an explosion there are several tasks to do:

- We must reduce the number of lives remaining and check to see if all the lives have been used.
- We must clear up the mess made on the screen by the explosion.
- We must reposition the detector and assistant in their starting position.

The first of these is easily accomplished by use of a count that is incremented each time the procedure is called. If the count exceeds four after incrementation then a special variable, end flag, is set to one to signal the fact that the game is over.

Having tested for the end of game, we have to clear up the explosion. Several methods could be employed here, such as PRINTing blank spaces over the area of the explosion. However, the method we use is to clear the screen, relay the mines and reprint the time and scores information. To make it slightly fairer, we should relay only the same number of mines as were left before the explosion. This can be easily calculated from the score. As each mine is worth 150 points and there were 50 originally, we can calculate the number of mines left and pass it to the 'lay mines' procedure.

The mine detector and assistant are repositioned by re-initialising their co-ordinates (by calling the initialise variables procedure), and then calling the position chars procedure. The full Reset listing is shown below. Add this and the explode procedure to your program.

```
3880DEF PROCreset
3890count=count+1
3900IF count>4 THEN end_flag=1:ENDPROC
3910CLS
3920VDU19,2,2,0,0,0
3930CLOUR 2
3940PROCinitialise_variables
3950mines_left=50-score/150
3960PROClay_mines(mines_left)
3970PROCdraw_border
3980PRINTTAB(2,27);"Time"
3990PRINTTAB(2,28);"Score"
4000PRINTTAB(11,28);score#
4010PRINTTAB(2,29);"Hi score"
4020PRINTTAB(11,29);hi_score#
4030remaining_men#=LEFT$(men#,4-count)
4040CLOUR 1
4050PRINTTAB(2,30);remaining_men#;" "
4060CLOUR 2
4070PROCposition_chars
4080ENDPROC
```

Also, the following modification is needed on the temporary calling program (see page 405):

```
60 UNTIL TIME>12099 OR end_flag=1
```