plot. Here is the complete listing for the snipe procedure:

```
3110DEF PROCsnipe
3120ystart=RND(750)+220
3130yfinish=RND(750)+220
3140dx=32:dy=(yfinish-ystart)/32
3150GCOL 3,3
3160PROCline
3170IF POINT(x,y)=1 THEN PROCexplode(x,y) ELSE PROCline
3180ENDPROC
```

And this is the line procedure listing:

```
3450DEF PROCline
3460SOUND0,-8,4,5
3470x=xstart:y=ystart
3480MOVE x,y
3490REPEAT
3500DRAW x,y
3510x=x+dx:y=y+dy
3520UNTIL x>xfinish OR POINT(x,y)=1
3530ENDPROC
```

THREE ADDITIONAL FEATURES

As we saw in the last instalment, quite complicated sounds can be generated by the BBC Micro. For those of you with a musical bent, we shall now add a short tune to the program. To make things as simple as possible we shall only use one channel. The tune can be played by simply specifying the frequency and duration of each note in the tune.

```
4090DEF PROCmusic
4100REM ** 1ST BAR **
4110SOUND1,-8,213,5
4120SOUND1,-8,209,5
4130SOUND1,-8,213,5
4140SOUND1,-8,209,5
4150SOUND1,-8,213,5
4160SOUND1,-8,193,5
4170SOUND1,-8,205,5
4180SOUND1,-8,197,5
4190REM ** 2ND BAR **
4200SOUND1,-8,185,20
4210SOUND1,-8,165,5
4220SOUND1,-8,185,5
4230SOUND1,-8,193,20
4240REM ** 3RD BAR **
4250SOUND1,-8,165,5
4260SOUND1,-8,193,5
4270SOUND1,-8,197,20
4280ENDPROC
```

Title Page: We can use the ideas of Exclusive OR plotting and relative point plotting to produce an interesting title sequence. This procedure draws the word MINES using high resolution graphics. Every new line drawn in the word is plotted relative to the last, so we can position the entire word anywhere on the screen simply by specifying the start point. If we plot the word and then replot in Exclusive OR before moving up and repeating the action, we can make the word appear to float up the screen. GCOL0,129 sets the background colour to red. Performing a subsequent CLG colours the whole screen red. At the same time, we can also play the tune defined above by calling PROCmusic. The information held in PROCmusic is processed rather more quickly than it is played, so a buffer is used to store SOUND information until it can be played. This means that the processor is free to move on to do other things while the tune is still playing.

Skill Factors: To make the game a little more challenging, we can employ the idea of skill factors. After the title has been displayed we shall ask for a number between 0 and 9, which will be stored in the variable skill. This can then be used to increase the number of mines on the minefield and the rate of sniping across the area. The first of these can be done by making a small alteration to the setup procedure given previously (see page 405). Change lines 1930 and 1940 to:

```
1930factor=skill*3+30
1940PROClay_mines(factor)
```

In addition, when we relay the mines during the reset procedure, we must calculate the number of mines remaining by changing line 3950 to:

```
3950mines_left=factor-score/150
```

The full listing for the title page procedure is:

```
1300DEF PROCtitle_page
1310GCOL 0,129
1320CLG
1330GCOL 3,3
1340PROCmusic
1350Y=100:X=0
1360REPEAT
1370X=X+20:Y=Y+50
1380FOR I=1 TO 2
1390PROCmines
1400NEXT I
1410UNTIL Y>700
1420:
1430PROCmines
1440PRINTTAB(0,20)"Skill factor (0-9)?"
1450PROCmusic
1460REPEAT
1470skill=GET-48
1480UNTIL skill>-1 AND skill<10
1490ENDPROC
1500:
1510DEF PROCmines
1520PLOT4,X,Y
1530REM ** LETTER M **
1540PLOT1,0,200
1550PLOT1,80,-100
1560PLOT1,80,100
1570PLOT1,0,-200
1580REM ** LETTER I **
1590PLOT0,40,0
1600PLOT1,80,0
1610PLOT0,-40,0
1620PLOT1,0,200
1630PLOT0,-40,0
1640PLOT1,80,0
1650REM ** LETTER N **
1660PLOT0,40,-200
1670PLOT1,0,200
1680PLOT1,120,-200
1690PLOT1,0,200
1700REM ** LETTER E **
1710PLOT0,160,0
1720PLOT1,-120,0
1730PLOT1,0,-200
1740PLOT1,120,0
1750PLOT0,-40,100
1760PLOT1,-80,0
1770REM ** LETTER S **
1780PLOT0,280,60
1790PLOT1,0,40
1800PLOT1,-120,0
1810PLOT1,0,-100
1820PLOT1,120,0
1830PLOT1,0,-100
1840PLOT1,-120,0
1850PLOT1,0,40
1860ENDPROC
```

Up to this point we have been using a temporary calling program (given on page 394) to knit our procedures together, but now we have assembled all the procedures that are required for the main program loop of the game. Erase the temporary calling program (lines 10 to 70) and enter the following listing:

```
2020DEF PROCloop
2030REPEAT
2040PROCupdate_time
2050PROCtest_keyboard
2060rand=RND(50-skill)
2070IF rand=1 THEN PROCsnipe
2080 UNTIL TIME>12099 OR end_flag=1
2090ENDPROC
```

Our calling program can now be written. Enter these lines:

```
10e0hi_score$="00000"
1110MODE5
1120REM ** TURN OFF CURSOR **
1130VDU23;8202;0;0;0;
1140PROCtitle_page
1150CLS
1160PROCsetup
1170:
1180PROCloop
```

In the next and final instalment of the course, we shall look at producing the end-of-game scenario and present a complete listing of our program.