



To set the screen colour to light blue and the border colour to dark blue, the following pair of POKE commands is required.

```
POKE53281,14:POKE53280,6
```

The seventh row of the screen starts at location 1264. The second row from the bottom starts at location 1944. We can colour the sea by POKEing the screen code for a reverse space (a space character that is blocked-out) into locations 1264 to 1943, and POKEing 6 into the corresponding colour memory locations. There is an easy way to connect a colour memory location with a corresponding screen memory location: simply add 54272 to the screen memory location.

The screen code for a space character is 32. A reverse character's screen code can be calculated by adding 128 to the normal screen code, so the code for a reverse space is $32 + 128 = 160$. The following few lines of program make use of a simple FOR...NEXT loop to colour the sea:

```
FOR I=1264 TO 1943
POKE I,160:POKE I+54272,6
NEXT I
```

The seabed consists of two rows of a chequered character with a screen code of 102, coloured brown. Again a simple FOR...NEXT loop will do this:

```
FOR I=1944 TO 2023
POKE I,102:POKE I+54272,9
NEXT I
```

The PRINT statement is also an effective method of producing screen displays. Colour and cursor positioning can be controlled from within a PRINT statement either by using the special Commodore control characters or by using CHR\$ codes. We will use the latter method, as that is easier to read in program listings. A full list of CHR\$ codes is given in Appendix F of the user guide. We are interested in those that affect colour and cursor position:

CHR\$(5)	colour white
CHR\$(144)	colour black
CHR\$(147)	clears screen and positions cursor in top left corner
CHR\$(19)	positions cursor in top left corner
CHR\$(17)	moves cursor one place DOWN
CHR\$(145)	moves cursor one place UP
CHR\$(157)	moves cursor one place LEFT
CHR\$(29)	moves cursor one place RIGHT

As part of our screen setup routine SCORE and HI SCORE must be PRINTed on the top line of the screen. CHR\$(19) will ensure that the cursor is at the beginning of the top line. The following command PRINTs the initial score in black:

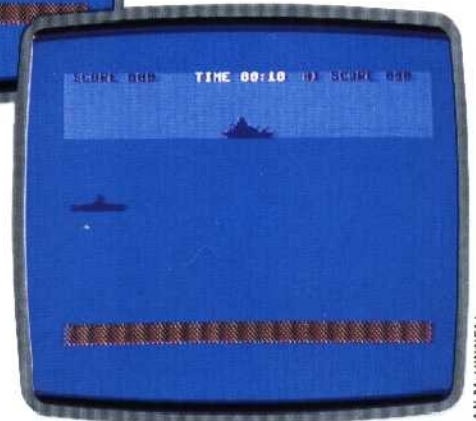
```
PRINT CHR$(19):CHR$(144);"SCORE 000"
```

The HI SCORE is also to be positioned on the top line, but on the right-hand side. The SPC function



Subhunter In Action

The program we are developing in this series is the classic computer graphics game of depth-charging submarines. The Commodore 64's sprite graphics are used to provide smooth animation of the sub and ship



IAN MCKINNEL

allows a number of spaces to be inserted. The PRINT command can now be altered to include the HI SCORE:

```
PRINT CHR$(19):CHR$(144);
"SCORE 000";SPC(16);"HI SCORE 000"
```

The screen setup routine will form a subroutine to the main program, starting at line 1000. Also included is a POKE command that causes all the keys on the keyboard to repeat when they are pressed. This will be used later when the keyboard control routine is discussed. This subroutine can be tested by the following program lines:

```
10 GOSUB 1000: REM SCREEN SETUP
20 END

1000 REM **** SCREEN SETUP ****
1010 PRINT CHR$(147):REM CLEAR SCREEN
1020 :
1030 REM ** COLOUR SEA **
1040 POKE53281,14:POKE53280,6
1050 FOR I=1264TO1943
1060 POKE I,160:POKE I+54272,6
1070 NEXT
1080 :
1090 REM ** SEA BOTTOM **
1100 FOR I=1944TO2023
1110 POKE I,102:POKE I+54272,9
1120 NEXT
1130 POKE 650,128:REM REPEAT KEYS
1140 :
1150 REM ** SCORE**
1160 PRINTCHR$(19); CHR$(144);
"SCORE 000";SPC(16);"HI SCORE 000"
1170 RETURN
1180 :
1190 :
```

Once the routine has been entered, it is a good idea to save it on tape or disk before running it.