



LATTICE PATTERNS

In the last instalment, we investigated a set of symmetrical patterns along a line, developing a procedure to draw these. Here, we extend our investigation of transformations to include patterns in two dimensions. To begin with, we look at how LOGO can create the grids on which these patterns are based.

If, instead of translating the motif we defined in the last article (see page 915) along a line, we allow two non-parallel translations to occur simultaneously, then our pattern becomes two-dimensional. We'll begin investigating this set of patterns by using a single dot as our unit.

```
TO DOT
  PD FD 1 BK 1 PU
END
```

The procedure to perform these simultaneous translations is defined as follows:

```
TO GRID :STARTX :STARTY :XSTEP :YSTEP :ANGLE
  DRAW HT PU
  SETXY :STARTX :STARTY SETH 0
  REPEAT 3 [LINE :XSTEP DOWN :YSTEP :ANGLE]
END
```

This procedure draws a grid of nine points. The inputs give the co-ordinates of the starting point, the size of the X and Y steps, and the heading of the corresponding points of the next row from the present row. The LINE procedure:

```
TO LINE :X
  REPEAT 3 [UNIT SETX XCOR + :X]
  SETX XCOR - 3 * :X
END
```

draws a single line of three units across the screen, and then returns the turtle to its starting point. For the moment, the UNIT procedure is simply a dot:

```
TO UNIT
  DOT
END
```

Another procedure:

```
TO DOWN :Y :A
  SETH :A
  FD :Y
  SETH 0
END
```

moves the turtle to the next row (in the way we have used it, this has involved moving 'down' to the next row, which gives the procedure its name), and then restores the heading. The five types of

plane lattice, along with LOGO procedures to draw them, are given in the diagram.

Many different patterns can be obtained from combinations of these basic grids, although it is probably more interesting to change the procedure to draw the line at various angles rather than straight across the screen.

Another line of investigation is to see how the symmetry of each of the grids can be enhanced by giving various forms of symmetry to the unit drawn at each point. There are 17 such patterns and they are all shown in the second diagram. An obvious method of drawing out all these 17 possibilities is to replace the UNIT command in the LINE procedure with a procedure that draws the shape at that point.

THE UNIT SHAPE

The unit shapes are made from a basic motif together with various reflections and rotations. As a demonstration, let's define our basic motif, and call it LIT. (As before, this is state transparent and uses no subprocedures.)

```
TO LIT
  PD
  FD 15
  RT 90
  FD 5
  BK 5
  LT 90
  BK 15
  PU
END
```

We can use the procedures we developed in the previous article to create two procedures: MOTIF and its mirror image, R.MOTIF:

```
DEFINE "MOTIF TEXT "LIT
DEFINE "R.MOTIF REWRITE "LIT
```

We can now define the unit shapes. For example, the units for patterns 7 and 17 are:

```
TO UNIT7
  LT 90 MOTIF RT 90
END

TO UNIT17
  RT 30
  REPEAT 6 [MOTIF R.MOTIF RT 60]
  LT 30
END
```

If you look back at the LINE procedure, you will see that it runs the procedure UNIT. Therefore, in order to draw pattern 7, for example, the procedure UNIT must be changed to read as UNIT7. This we do by using DEFINE.UNIT7 where:

