The New Nation

142

Aritmmetic througn BASIC

Naser Siddiqui

In your programme, you may make calculations first and then use the result in another statement, or you can indicate calculations within the statement. Supposing that all variables have previously been defined, these two program segments will have the same results :

```
600x  A 0 B  C
610y  D/E
620   PRINT X, Y
or,
600   PRINT A 0 B—C, D/E
```

In the above descriptions we have seen arithmetic in the PRINT statement, but this is not the only place one can use numbers. Any statement requiring numeric parameters can use actual numbers which are constants, variables, or numeric expressions. You can use these arithmetic operations to make calculalions for the numbers required in other statements. For example, it may be needed to calculate a row and a column to begin printing or drawing. Some one may want to draw to one point with coordinates calculated in terms of the coordinates of adother point. If any one is interested to use SOUND commands with variables and expressions rather than constant numbers he may do so through BASIC.

In order to do that he would be required to be familiar with some commands like FOR-NEXT, IF-THEN. The statement FOR-NEXT may have variable index limits while IF-THEN statements may compare constants, or they may compare constants with variaples, or they may compare numeric expressions. Functions such as SIN(x) can use numeric expressions in the argument. The following statements are given to illustrate the points mentioned above :

```
200 PRINT INT(A B RND(1)01)
210 IF BLUE> RED+1 THEN600
220 PRINT AT(ROW+2, Col+5),
                    NAMES
230 SOUND F, D04, V+2
240 CIRCLE(X08+-A. Y+B),
                    R+5
250 LINE (X, Y)+(X+L, Y+W)
260 PRINT SGN(D0E0(F-G)/8)
270 FOR K+A+BTO C-D STEP
                (+1)AS
```

Let us see how to gr ab the day number. First we read a record and assign the three date bytes to XS, YS and ZS respectively. Now we want to convert a part of ZS to a number. The first bit we ignor and the last two bits also we ignore since we do not need them.

The logical 'AND' operator can be used to "wipe out" unwanted bits. If we get AND as well as the ZS variable with the bit pattern 01111100 we would be erasing bits 7, 1 and O; the bits for the day—6,5,4,3 and 2—are not changed by AN Ding with the 1's in the bit program. Thus our BASIC statement might be read as DAY: (ASC(Z[illegible]) AND 01111100 except BASIC does not have a binary data type and would not understand the O's and 1's.

If the variable DAY is printed, we would not still get a number between 1 and 31 as we are not quite through. Since we are working in binary, we divide 4(2S) to shift two places of decimals to the right. Thus we see that DAY—(ASC(RS) AND & A7c)/4 is the correct BASIC statement; printing DAY will print a value between 1 and 31.

If anyone is interested to buy a new flat he may use BASI to do the needful part of his arithmetic calculations. By inserting the amount of owners loan, its length, and its interest rate the owner can see quickly how much his monthly payments will be, how much will be the total amount of interest that he will have to pay over the life of the loan.

In this way the scope of using BASIC in arithmetic is expanding gradually.