

the dollar sign.

```

NEW
OK
10 INPUT S$
20 I = INSTR (S$, "$")      'FIND THE $ IN S$
30 X = VAL (MID$ (S$, I+1)) 'FIND VALUE OF DOLLAR AMOUNT
40 PRINT "The amount is"; X
50 GOT010
RUN
? The unit cost is $6.34
The amount is 6.34
? The price is $9.95
The amount is 9.95
? ^C
BREAK AT LINE 10
OK

```

If the value of A\$ is not a legal constant, an FC error occurs and VAL returns 0.

CLEAR and FRE\$

Instead of wasting memory by statically allocating space for each string variable to contain up to 255 characters, Extended XYBASIC stores the values of string variables dynamically in a part of memory called string space. String space is also used for temporary storage during the evaluation of some string formulas. Extended XYBASIC initially allocates 256 bytes of string space, which should be adequate for programs which do not use strings extensively. For more complex string programs, the amount of allocated string space may be changed by using the CLEAR command with a numeric argument. For example,

```

CLEAR 1000
OK

```

tells Extended XYBASIC to allocate 1000 bytes of string space. The argument of CLEAR may be any numeric formula. Of course an OM (Out of Memory) error will occur if you try to allocate more string space than the amount of free memory allows.

Just as before, the CLEAR command (without an argument) may be used to clear all variables; this form of CLEAR leaves the amount of string space unchanged.

You can use the function FRE\$ to find the number of unused bytes remaining in string space, as in the following example.

```

CLEAR 256
OK
PRINT FRE$
256
OK

```