VideoBrain™ Family Computer Owner's Handbook

Money Management Series

VB-81 Financier™

VideoBrain™ VB-81 Financier™

Money Management Series

Owner's Handbook

VideoBrain Computer Company
150 South Wolfe Road
Sunnyvale, CA 94086

Program Design: Robert G. Brown

P/N 9320011   Rev. 0   4/78
Printed in the United States of America

Price: $5.00

VB-81 Financial Functions and Variables

LOAN	**Direct reduction loans and bonds**	
	PMT	Periodic payment
	I	Periodic interest rate
	N	Number of payments
	FV	Balloon amount
	PV	Loan amount
SAVE	**Money in a savings account**	
	PMT	Periodic deposit
	I	Periodic interest rate
	N	Number of deposits
	PV	Initial deposit
	FV	Amount in account at end of period N
ROI	**NPV of an investment**	
	N	Number of periods
	C0	Initial investment
	C1	Cash flow in period 1
	⋮	
	CN	Cash flow in period N
	I	Discount rate
	NPV	Net present value
CALC	**Four-function calculator**	

ACP, ACI	**Accumulated principal and interest**
PMT	Periodic payment
I	Periodic interest rate
SPN	Starting period number
EPN	Ending period number
N	Number of payments
FV	Balloon payment
AP	Accumulated principal
AI	Accumulated interest
IPP	**Installment purchase plan**
I	Periodic interest rate
N	Number of payments
PV	Loan amount
PN	Payment period number
PMT	Payment for period PN
AOL	**Add-on loan**
AOI	Annual add-on interest rate
N	Number of payments
PV	Loan amount
PMT	Monthly payment
R78	**Rebate using the rule of 78**
N	Number of payments
PV	Loan amount
PMT	Periodic payment
PN	Payment period number
REB	Rebate on loan
DB, SOD, SL	**Declining balance, sum-of-the-years-digits, and straight-line depreciation**
PV	Cost of asset
FV	Salvage value
N	Number of periods
PN	Depreciation period number
DF	Decline factor (DB only)
DEP	Depreciation for period PN
BV	Book value for period PN

VB-81 Instructions

VB-81 introduces you to computer terms like formula, variable, and function, then shows you how to use them to solve your financial problems. VB-81 displays answers to your financial problems both as numbers and as bars on a bar graph to show you how your answers compare to each other.

This handbook contains definitions of basic financial terms including interest, principal, annuity, and depreciation. You will learn about net present value, rate of return, and different types of loans. You provide VB-81 with necessary information about your financial ventures, then the computer analyzes the data for you.

Many sample problems are set up and solved so you can follow through the same problems on VideoBrain as you read the handbook. All examples are typical savings and investment problems, stated in the same terms you are likely to encounter when considering financial ventures.

Read through this handbook carefully before starting, work out the sample problems, then keep it handy for reference when you are solving your own problems.

Inserting the Cartridge

1. Make sure your VideoBrain computer is attached to your TV as described in the Owner's Manual. Check that power is on.

2. Push the cartridge carrier release button above the VideoBrain keyboard to swing the cartridge carrier door up.

3. With the label facing up, slide the cartridge all the way into the tracks on the underside of the cartridge carrier door.

4. Gently push the cartridge door down into the computer until it locks.

5. Push the **MASTER CONTROL** button. The title of the cartridge should appear on your TV for two seconds.

Table of Contents

Section 1: Operating the Program

Section 2: Solving Formulas

Section 3: Computer Variables

Section 4: Using the Functions

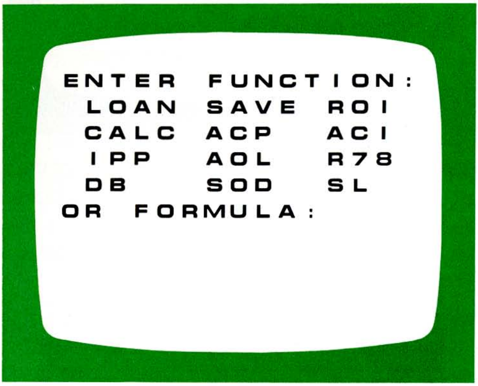
Section 5: Function Descriptions

Appendix A: Financial Formulas

Section 1: Operating the Program

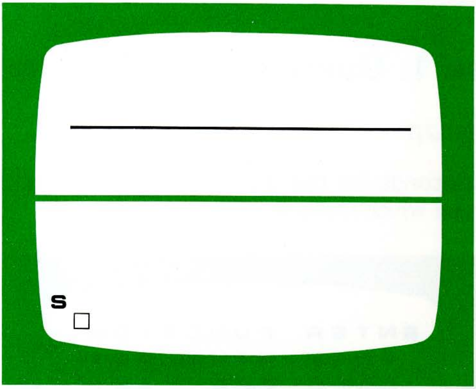
Starting Up

After two seconds the title display on your TV changes to the *menu screen* which looks like this:



Now you must decide whether you want the computer to solve one of the twelve functions or one of your own formulas. To solve a function, type in the name of the function exactly as it appears on the screen (such as LOAN) then type the **NEXT** key. To solve a formula, type in the formula (such as 3+4) followed by the **NEXT** key.

As soon as you type the first key, the menu screen changes to the *split screen* which looks like this:



The computer uses the green field on the top part of the screen for a bar graph. Each time the computer solves a function it draws one bar on the graph. This bar represents the answer to the function. You will learn more about this in section 4 when the functions are defined.

The computer uses the blue field on the bottom part of the screen to display the characters that you type, the questions that the computer asks you, and the answers to functions and formulas.

Typing into the Computer

The picture of the split screen above shows what the screen looks like if the first key you type is an S. The white square after the S is called the *cursor*. The cursor shows you where the next character you type will appear. At first there is an L inside the cursor. This means that when you type a key, the lower character on the key appears on the screen. To type the upper character on a key, first type the **SHIFT** key so a U appears inside the cursor, then type the key you want. To type the lower character, type the **SHIFT** key again so the L reappears inside the cursor.

Each time you type one of the black keys the character appears on the screen above the cursor and the cursor moves one position to the right. You can type 15 characters on one line. If you type **NEXT** before the end of a line, the cursor automatically moves to the first position of the next line. If you type 15 characters on a line, the cursor automatically moves to the first position of the next line and a white square appears in front of the cursor indicating this line is a continuation of the previous line. The computer allows you to have two continuation lines, so the longest formula you can type is three lines. The lines you type are always indented one position from the lines the computer displays.

If you type any mistakes on a line you can correct them by using the **BACK** key. Each time you type the **BACK** key the cursor moves one position to the left and erases the character that was there. For example, say you type SABE when you meant to type SAVE. Just type the **BACK** key twice to erase BE then type VE.

You can not use the **BACK** key to move the cursor to a previous line. If your formula is longer than one line and you made a mistake on a previous line of the formula, type the **ERASE** key then retype the entire formula.

There is enough room in the blue field for four lines. When you finish typing the fourth line, the first line disappears and the remaining three lines each move up a line, leaving the cursor at the beginning of the blank fourth line.

Special Keys

The seven blue keys and one red key are special-purpose keys which affect computer operation in the following way:

MASTER CONTROL tells the computer to stop whatever it is doing and start the program over again. The title screen appears followed by the menu screen.

BACK moves the cursor one position to the left and erases the character that was there. Use this key to back up and correct typing mistakes.

NEXT tells the computer that your line is correctly typed and you want the computer to process it. This means that the computer reads your line, finds out what you want to do (function or formula), and does it.

ERASE tells the computer that the current line (or 2-3 lines for a long formula) is incorrectly typed and you do not want the computer to process it. A white square appears at the cursor position then the cursor moves to the beginning of the next line so you can type a new one.

SHIFT places either a U or L inside the cursor to tell the computer to display either the upper or the lower character on the key you type.

SPACE moves the cursor one position to the right. A blank space appears on the screen.

PREVIOUS and **SPECIAL** are ignored by the computer. If you type either one of them nothing happens.

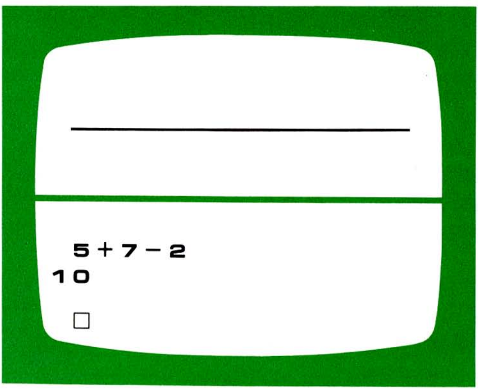
Section 2: Solving Formulas

What is a Formula?

A formula is simply an arithmetic problem that contains numbers separated by one or more of these operators: +, −, ×, ÷, and %. Each operator tells the computer to do one operation. The first four operators are the standard symbols for add, subtract, multiply, and divide operations. The % operator means divide by 100; for example, 30% is the same as 30÷100 or .30. You can represent a decimal number by using a period in the number, and you can represent a negative number by typing a − first. These are examples of formulas:

$$3+7$$
$$8.35-2+3$$
$$150\times 40\%$$
$$7+2\times 3\times 4$$
$$15\div 3+6.2$$
$$45\times 20\%-3+75$$

To solve your formula, type it into the computer then type **NEXT**. The computer solves your formula immediately and displays.the answer on the next line of the screen. For example, if you type 5+7−2 the computer solves it and displays the answer on the screen like this:



How the Computer Solves Your Formula

The computer can only do one operation at a time. If your formula contains more than one operator, the computer follows three rules in deciding which operation to do first, second, and so on until the formula is solved.

Precedence Rule. The computer first does all % operations, then all × and ÷ operations, and finally all + and − operations. For example, if your formula is $3+4\times2$, the computer solves it like this:

$$3+4\times2 = 3+8 = 11$$

Without this rule you might expect the computer to solve it like this:

$$3+4\times2 = 7\times2 = 14$$

A more complex example illustrating this rule is:

$$150-80\times25\% = 150-80\times.25 = 150-20 = 130$$

Left-To-Right-Evaluation Rule. If there are two or more operators of equal precedence, the computer does them in order from left to right. For example:

$$7+8-3 = 15-3 = 12$$

$$8.6\times2-9\div3 = 17.2-9\div3 = 17.2-3 = 14.2$$

Parentheses Override Rule. If you place part of your formula in parentheses, the computer solves that part of the formula first. By using parentheses you can tell the computer to solve a + or − operation before a %, ×, or ÷ operation. For example, by the first two rules:

$$3+4\times5 = 3+20 = 23$$

but by using parentheses:

$$(3+4)\times5 = 7\times5 = 35$$

You can also use parentheses within parentheses; they are called *nested* parentheses. The computer first solves the part of the formula nested within the greatest number of parentheses, then it expands outward to the next set of parentheses. For example:

$$3+(4\times(9+(3+6\times2))) = 3+(4\times(9+(3+12))) =$$
$$3+(4\times(9+(15))) = 3+(4\times(24)) =$$
$$3+(96) = 99$$

Section 3: Computer Variables

What is a Variable?

A variable is a 1, 2, or 3 character name that you can use to represent a number. The first character of the variable name must be a letter of the alphabet. The other characters may be letters or numbers. You can use more than three characters in a name as long as no two variables have the same first three characters. These are examples of variable names:

X F18 A17B DAY N1 STAR

To use a variable you must give it a name and assign it a value in an *assignment statement*. An assignment statement consists of a variable name followed by an = followed by a number. For example, if you type:

A12=5

you have given the variable A12 the value 5. Each time you name and assign a value to a variable, the computer stores the name and value in its memory. If you name a new variable but do not assign a value to it, the computer stores the name in memory and defines the value as unknown. Since the memory is limited in size you can only name 20 variables at a time. If you name a 21st variable the computer erases one of the first 21 variables so there will be room for a new one. The variable that the computer chooses to erase is the variable used least recently.

Pi (π) is a pre-defined variable which always has the value 3.14159265.

Using Variables

Once you have assigned a value to a variable, you can use it in your own formulas in place of numbers. For example, if you type:

CAT=8.65

you can use CAT in a formula such as:

3+CAT

The computer displays 11.65 because 3+CAT = 3+8.65 = 11.65.

You can also use a variable with a known value as part of the assignment statement for another variable, for example:

GO1=4

GO2=7×GO1+2

means that GO2 has the value 30 because GO2 = 7×GO1+2 = 7×4+2 = 28+2 = 30. Now you can use both GO1 and GO2 in other formulas or assignment statements.

If you try to use a variable in a formula before you assign a value to it, the computer cannot solve the problem. For example, if you type:

NEW=OLD+15

the computer displays VALUE ERROR because it does not have a value for OLD.

Once you have assigned a value to a variable, the value stays the same until you assign a new value to it or type **MASTER CONTROL**. To assign a new value to a variable, simply type a new assignment statement for it. For example, if you had typed earlier:

JOE=5

4+JOE

the answer was 9. But now if you type:

JOE=12

4+JOE

the answer is 16 because you assigned the value 12 to JOE.

You can name and assign values to more than one variable in one statement by using more than one =. The computer solves the rightmost part of the assignment statement first, then uses this answer as the value of the variable on the left part. For example, if you type:

B3=8+(B2=40+2)

the computer solves for the value of B3 like this: B3 = 8+(B2=40+2) = 8+(42) = 8+42 = 50.

Displaying Variables

You can display the value of a variable by typing the variable name followed by **NEXT**. Suppose you want to display the value of JOE, just type:

JOE

and the computer displays 12.

If you try to display a variable that you named but did not assign a value to, the computer displays a ? because it does not know the value.

If you either type **MASTER CONTROL** or turn off the power to VideoBrain, the computer forgets the values of your variables. For example, if you type:

X77=35
MASTER CONTROL
X77

the computer displays a ? because the value of X77 was erased when you typed **MASTER CONTROL**.

Sample Problem

Here is a sample problem which shows you how to use variables and formulas to solve a practical problem. Suppose you are planning to buy three rugs for your house, one for the living room, one for the dining room, and one for the hall. You want the living room rug to be 12 feet × 16 feet, the dining room rug to be 9 feet × 12 feet, and the hall rug to be 2 feet × 9 feet. The problem is that you must decide whether to put a rug which costs $24.50 per square yard in all three rooms, or to put that rug only in the living room and put a rug which costs $17.50 per square yard in the other two rooms. You want to solve the problem both ways and see what the actual difference in cost is.

First define the variables for the formula. For this problem the variables are the areas (in square feet) of each of the rooms and the costs (per square foot) of each of the rugs. Use names that are easy to remember like LR (area of living room rug), DR (area of dining room rug), and HAL (area of hall rug). Define the areas of the rooms like this:

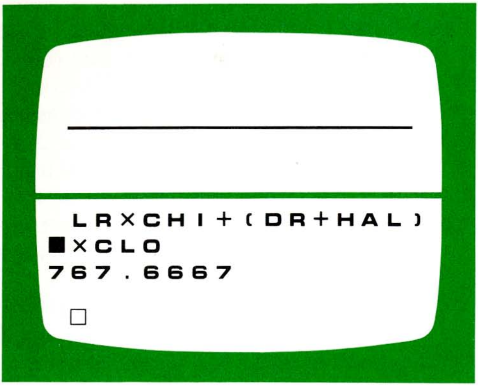
LR=12×16
DR=9×12
HAL=2×9

Next, define the costs as variables. The costs as advertised are per square yard, but the room dimensions are in feet so you must divide each price per square yard by 9 to give you the price per square foot. So you can define the costs like this:

CHI=24.50÷9
CLO=17.50÷9

Now you want to solve the two formulas to compare costs. In the first formula the computer figures out how much it costs to have the rug costing $24.50 per square yard in all three rooms. This formula is (total area in square feet) × (cost per square foot of 24.50 per square yard rug) = (LR+DR+HAL)×CHI. Type this formula followed by **NEXT** and the computer displays the answer like this:

The other formula computes the cost of putting the rug costing $24.50 per square yard in the living room and a rug costing $17.50 per square yard in the other two rooms. This formula is (area of living room rug) × (cost per square foot of 24.50 per square yard rug) + (area of dining room rug + area of hall rug) × (cost per square foot of 17.50 per square yard rug) = LR×CHI+(DR+HAL)×CLO. Type this formula followed by **NEXT** and the computer displays the answer like this:



Now you can compare the costs, $865.67 to $767.67, and decide which rugs to buy.

Section 4: Using the Functions

What Are Functions?

A function is a program in the computer that solves a pre-defined formula when you assign values to the variables used in the formula. The computer already knows the formula; it asks you to assign values to each of the variables in it. For example, you can use the AOL function to compute what your monthly payments will be if you take out an add-on-interest loan. The amount of money, add-on-interest rate, and number of months are the values you must assign to variables before the computer can compute your monthly payments.

Solving Functions

To solve a function type in the name of the function followed by the **NEXT** key. The computer asks you to assign a value to each of the variables it needs to solve the function. In the AOL function described above, the computer needs values for these variables:

AOI	Yearly add-on-interest rate
N	Number of months to pay off loan
PV	Amount of loan

The computer asks you to assign a value to each of these variables by displaying the variable name followed by an =, for example PV=, and leaving the cursor immediately after the =. You type in the value of PV followed by **NEXT**, for example PV=500 (you only type the 500). You can type a number like 500, a formula like 10×50, or a formula containing variables like OLD+200. If you use a variable such as OLD in the formula, you must have already assigned a value to it.

If you have recently used a function where you defined one of the variables, for example PV, the computer asks you if you want to use the same value of PV or assign a new one. For example, say you just solved the AOL function and assigned 500 to PV. Now you want to solve the SAVE function which also uses PV. Instead of displaying PV=, the computer displays PV=500. If you want to use the same PV value just type the **NEXT** key. If you want to change PV to 400, type the **BACK** key three times to erase the 500, then type 400 followed by **NEXT**.

If you want to change a value and can't get to it using the **BACK** key, you can use the **ERASE** key. This may happen when the computer asks you if you want to use the same definition of a variable, and the variable and its definition are larger than one line. For example, if the future value (FV) is defined from a previous function to be 17,392,000,000,000, the computer displays:

FV=1739200000000
■0 □

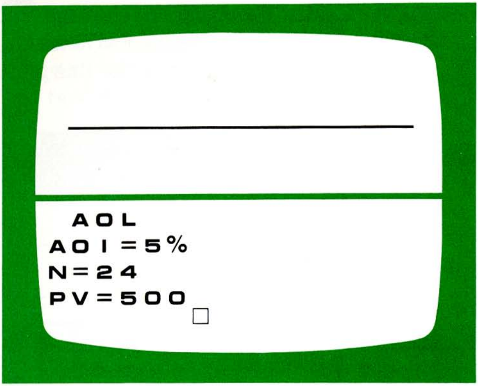
If you want to use FV=18,000,000, you can not move the cursor back to the = because the = is on a previous line. Type **ERASE** to move the cursor to the next line, then type FV=18000000.

Cancelling Functions

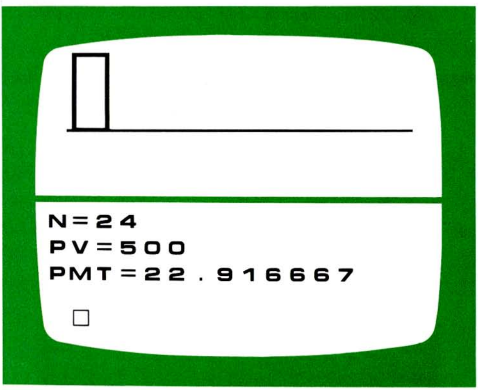
If you decide that you do not want to solve a function after you began it and the computer is already asking you for the definitions of the variables, you can cancel the function by typing the **ERASE** key followed by **NEXT**. The computer displays a white square at the cursor position, displays ENDED on the next line, moves the cursor to the beginning of the line after ENDED, and waits for you to type the next formula or function name.

Answers to Functions

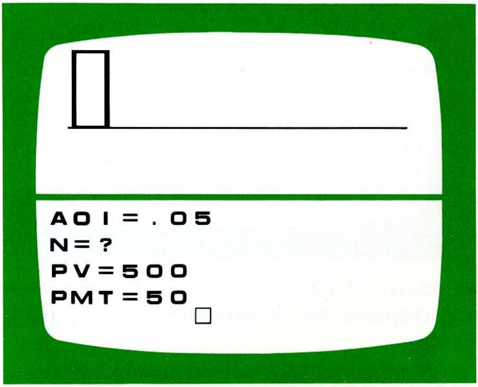
Once you define all the variables for a function the computer solves the formula and displays the answer, both as a number in the bottom part of the screen and as a bar in the top part of the screen. Results are generally accurate to at least 5 significant figures. Say you defined the variables for the AOL function and the screen looks like this:



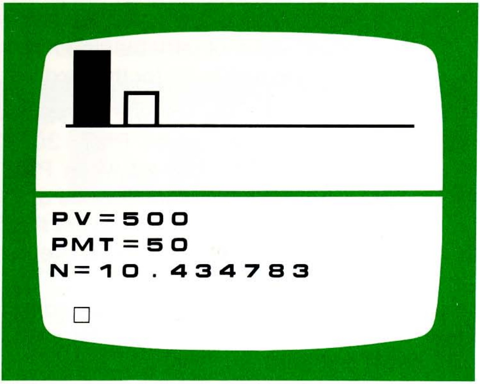
As soon as you type **NEXT** after the 500, the computer solves the formula and displays the answer (amount of payment) on the screen like this:



Several functions let you solve for more than one of the variables used in their formulas. For example, in the AOL function you can solve for PV, AOI, or N as well as PMT. Suppose you already know how much you want your monthly payments to be, say $50 (PMT=50). You know how much you need to borrow (PV=500) and you know what the interest rate is (AOI=.05), but you want to know how long it will take you to pay off the loan if you pay $50 each month. In other words you want the computer to solve for N. To do this, just type a ? when the computer displays N=. This tells the computer that it must ask you to define PMT instead of solving for it as in our first example. So the setup for this problem looks like this:



Type **NEXT** and the computer solves the function and displays:



so you can pay off your loan in 11 months.

Repeating Functions:
Sensitivity Analysis & Schedule Generation

You may want to solve the same function more than once, using a different value for one of the variables each time. This is called *sensitivity analysis.* For example, you want to know how long it takes to pay off an add-on-loan if you pay $30, $40, $50, or $60 per month.

Also, you may want to solve the same function for several time periods. This is called *schedule generation.* For example, you want to calculate the depreciation on a duplex for each of the next five years.

The computer provides a special type of assignment statement which allows you to assign a range of values to a variable. The computer uses each value once to solve the function. This type of assignment statement consists of the variable name followed by an =, followed by three numbers separated by the words TO and BY, like this:

VAR = VL TO VH BY VI

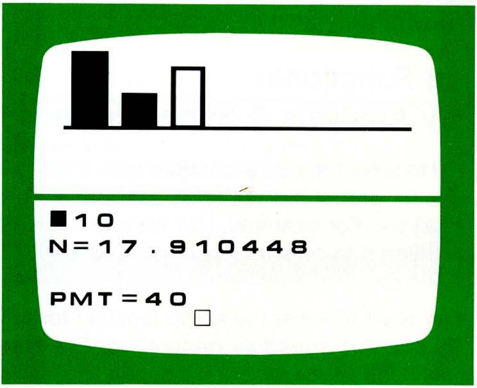
where the variables are defined like this:

VAR	Variable you are defining
VL	Lowest value of variable, first value used
VH	Highest value of variable used
VI	Amount to add to VL for the next value to use

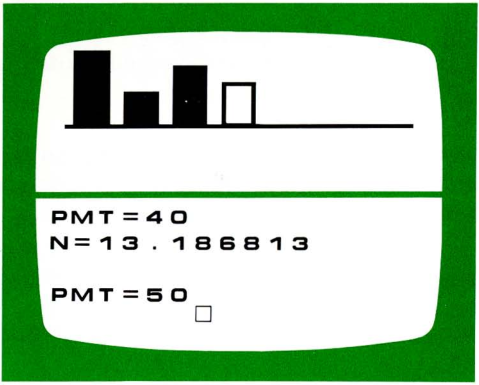
In the last AOL sample problem say you want to solve for N when AOI=.05 and PV=500, and first when PMT=30, then when PMT=40, then when PMT=50, and last when PMT=60. The lowest value of PMT (VL) is 30, the highest value of PMT (VH) is 60, and the amount to add to each variable to get the next (VI) is 10. The assignment statement for PMT looks like this:

PMT=30 TO 60 BY 10

This tells the computer to solve the AOL function four times by increments of 10. The first time, the computer solves the AOL function using PMT=30 and displays:



which means that when PMT=30, it takes 18 months to pay off the loan. The computer displays PMT=40 to tell you it is ready to solve the function again, this time using PMT=40. When you type **NEXT**, the computer displays:



which means that when PMT=40, it takes 14 months to pay off the loan. Now it is ready to solve the problem using PMT=50. The computer continues solving the problem with different values of PMT until it solves the function using the highest PMT value (60 in this case).

Whenever the computer displays the next PMT value, you can end the problem by typing **ERASE** and **NEXT**. The computer displays ENDED and waits for you to type the next formula or function. If you want to continue the problem but use a different value for PMT, type **BACK** to erase the value shown, then type the new value followed by **NEXT**.

You can use this type of assignment statement for any one of the variables in a function.

You can use formulas enclosed in parentheses as values of the variables VL, VH, and VI. For example, you could use this formula:

I=(7%÷12) TO (10%÷12) BY (1%÷12)

You can also use assignment statements which do not include either BY VI or TO VH, like this:

VAR=VL TO VH

VAR=VL BY VI

If you do not include BY VI, the computer automatically adds 1 to VL each time it executes the function until VL=VH. For example, if you want to generate a depreciation schedule using the values 4, 5, 6, 7, and 8 for PN, type:

PN=4 TO 8

This tells the computer to solve the depreciation function five times, starting with PN=4 then adding 1 to PN each time it solves the function until it solves the function using PN=8.

If you do not include TO VH the computer adds VI to VL and solves the function for each new value of VL until you stop it by typing **ERASE** and **NEXT**. For example, if you want to solve the AOL function using the values 1000, 1500, 2000, 2500 and so on for PV, type:

PV=1000 BY 500

This tells the computer to solve the AOL function over and over, adding 500 to the previous value of PV each time.

This type of statement is useful when you want to assign the value to the variable (using the **BACK** key) each time the computer solves the function. For example, if you want to first solve the AOL function using 5% for AOI, then assign a value to AOI each time after, type:

AOI=5% BY 1%

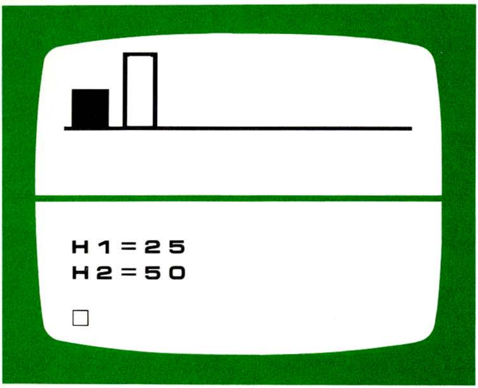
After the computer solves the function using AOI=5%, it displays AOI=0.06 to tell you it is ready to solve the function using AOI=6%. Type the **BACK** key three times to erase .06 and type a new value for AOI. This allows you to use arbitrary values that are not a certain amount larger than the previous value. Note that in this case it is unimportant what VI is; you could say BY 5% or BY 1.5% because you are going to reset the value anyway.

Using the Bar Graph

Each time the computer solves a function it draws a bar on the bar graph. The bar is above the line if the answer is greater than zero and below the line if the answer is less than zero. The graph is big enough for seven bars. When the computer solves the eighth function it replaces the first bar by the bar representing the answer to the eighth function. The ninth function replaces the second bar and so on. The lightest colored bar is the last bar drawn.

The computer scales the bars automatically to show you how the answers compare to each other. When it adds a bar, the computer may rescale the other bars so the new bar fits in.

The computer uses the variable names H1, H2, . . ., H7 to represent each of the bars, where H1 is the first, H2 is the second, and so on. When you assign a value to these variables (just like any other variable) the computer draws the specified bar. For example if you type H1=25 and H2=50 the bar graph looks like this:



You can always find out what number a bar represents by typing the variable name for the bar (such as H3) followed by **NEXT**. The computer shows you the value on the next line. You can also use the bar graph variables in formulas; for example, H3=H2−H1 sets the third bar equal to the difference between the first and second bars. You can erase a bar entirely by defining it as ?, as in H4=?, which erases the fourth bar.

Section 5: Function Descriptions

VB-81 provides these twelve financial functions:

SAVE	Future value of money in a savings account
LOAN	Present value of a direct reduction loan
ACP	Accumulated principal on a direct reduction loan
ACI	Accumulated interest on a direct reduction loan
AOL	Periodic payment for an add-on-interest loan
IPP	Periodic payment for an installment payment plan
R78	Rebate on a loan paid off early
ROI	Net present value of an investment
SL	Depreciation and book value (straight-line method)
SOD	Depreciation and book value (sum-of-the-years-digits)
DB	Depreciation and book value (declining balance)
CALC	Answers to arithmetic problems

You can also use most functions to solve for other variables used in the function.

This section contains a description of each of the twelve functions. It defines each function, lists all variables used in the function, and gives one or more examples showing you how to use the function. First, some financial terms are defined for you.

Financial Terms

Interest is an amount of money which one person pays to use another person's money. If you deposit money in a savings account you receive interest from the bank because the bank has use of your money. If you borrow money from a bank you pay interest to the bank because you have use of the bank's money. The amount of interest is dependent upon the principal, the life of the loan, and the periodic interest rate.

Principal (PV). The principal is the amount of money borrowed or loaned. If you borrow $500 from a bank, the principal is $500.

Life of Loan (N). The life of a loan is the amount of time needed to pay off a loan. The life of a loan is measured in payment periods. The number of payment periods for a loan is the number of payments you must make to pay off the loan. For example, if you pay $100 each month for two years to pay off the loan, the life of the loan is 24 months because you must make 24 payments.

Periodic Interest Rate (I). Interest is generally expressed as an annual percentage rate. The periodic interest rate is the annual interest rate divided by the number of payment periods in a year. When the computer asks you to define I, you can express it as a percent like 5%, or as a decimal like .05.

You must define I in terms of payment periods, not the annual rate. For example, say you borrow $500 at an 8% annual interest rate with monthly payments of $50. The principal is 500, the payment period is one month, and the annual interest rate is 8%. Since the length of a payment period is one month, the periodic interest rate is 1/12 of the annual interest rate, so I=.08÷12 or I=8%÷12.

Balance (BAL). The balance is the amount still owed on a loan. It is equal to the amount of principal left to pay. As you pay off a loan the balance decreases.

Annuity / Periodic Payment (PMT). Depending upon the type of loan you get, your periodic payments may not be the same each period. If your periodic payment is the same each period, it is called an *annuity*. If you pay the annuity at the *end* of each payment period it is called an *ordinary annuity*. Mortgages are usually ordinary annuities; you set up the mortgage and make your first payment one month later. If you pay the annuity at the *beginning* of the payment period, it is called an *annuity due*. Your rent is usually an annuity due; you pay rent at the beginning of each month.

For the LOAN, SAVE, ACP, and ACI functions you must specify whether the periodic payment is an ordinary annuity or annuity due. At first (after you type **MASTER CONTROL**) the functions assume the payment is an ordinary annuity. If you want to change to annuity due, type BEGIN followed by **NEXT**. If you want to change back to ordinary annuity, type END followed by **NEXT**. Whenever you type BEGIN or END the computer displays either WAS BEGIN or WAS END to tell you the previous setting.

It may seem unimportant whether you make your payments at the beginning or the end of the month. However, if you solve the same problem, first with annuity due, then with ordinary annuity, you can see the difference.

For example suppose you cannot decide whether to buy a car or lease it. Either way you will have $125 payments each month for the next three years at 10% annual interest rate. At the end of the 3 years you will own the car and continue to use it rather than sell it. You want to find out whether it is cheaper to buy the car (ordinary annuity) or lease it (annuity due). Use the LOAN function and define the variables like this:

PMT=125	Monthly payment
I=10%÷12	Monthly interest rate
N=3×12	3 years of monthly payments
FV=0	No balloon payment

The computer solves for the present value of the loan which represents the actual cost (the cost if you paid the whole amount now rather than in payments over a 3-year period). The computer displays PV=3873.9045 which means it costs you $3873.91 to buy the car. Now type BEGIN to tell the computer that you want to use annuity due. The computer displays WAS END. Use the LOAN function again without changing the values of any of the variables. This time the computer displays PV=3906.1869 which means it costs you $3906.19 to lease the car for the same period. You can see that an annuity due loan costs you $32.28 more than an ordinary annuity loan.

That example illustrates the difference between ordinary annuity and annuity due when you borrow. Now look at a situation where you save. Suppose you deposit $200 in a saving account and plan to deposit $25 each month for the next 2 years at 7% compounded monthly. You want to find out whether it is more profitable to deposit the $25 at the end of each month or the beginning. Use the SAVE function to calculate the total amount of money in your account at the end of two years, and define the variables like this:

PMT=25	Monthly payment
I=7%÷12	Monthly interest rate
N=2×12	2 years of monthly payments
PV=200	Initial deposit

The first time the computer calculates the future value of your account assuming annuity due (you typed BEGIN for the last example) and displays FV=875.7321. Now type END and solve the SAVE function again without changing the values of any of the variables. This time the computer calculates the future value assuming ordinary annuity and displays FV=871.987. You can see that in this savings problem, annuity due earns you $3.75 more.

These two annuity examples illustrate the difference to you between ordinary annuity and annuity due. If you are saving, investing, or receiving money, an annuity due is more profitable. But if you are paying money, an ordinary annuity is better because you pay less.

VB-81 Functions

SAVE solves for the future value of an investment like a savings account, where you make an initial deposit followed by equal periodic deposits for a certain number of periods at a certain periodic interest rate. SAVE uses these variables:

PMT	Periodic deposit
I	Periodic interest rate
N	Number of deposits, N⩾0
PV	Initial deposit
FV	Future value, amount in the account at end of period N

SAVE solves for any one of these variables. Be sure to specify BEGIN for annuity due or END for ordinary annuity.

Example: You deposit $500 in a savings account and you plan to deposit $50 at the beginning of each month for the next 3 years. The annual interest rate is 6% compounded monthly. How much money will you have in this account at the end of the 3 years? Type BEGIN since you make the deposit at the beginning of the month, then define the variables like this:

PMT=50	Monthly payment
I=6%÷12	Monthly interest rate
N=3×12	3 years of monthly payments
PV=500	Initial deposit

The computer solves for the future value and displays FV=2574.9796; you will have $2574.98 in this account in 3 years.

Example: Suppose you want to plan for your child's college education in 5 years and you want to know how much you need to deposit in your savings account each month so you will have $10,000 available. Initially you have $1000 in the account, and the annual interest rate is 5% compounded monthly. What is your monthly payment if you pay at the end of the month? Type END for ordinary annuity and define the variables like this:

PMT=?	Solve for monthly payment
I=5%÷12	Monthly interest rate
N=5×12	5 years of monthly payments
PV=1000	Initial deposit
FV=10000	Amount you want to have in account

The computer solves for your monthly payment and displays PMT=128.17446 which means you must deposit $128.17 each month to have $10000 in 5 years.

Example: Now suppose you open a savings account which has an annual interest rate with a compounding period that is different from the payment period. In this case, the periodic interest rate (I) is not equal to the annual interest rate divided by the number of payment periods in a year because the interest is not calculated at the payment period, it is calculated at the compounding period. To use the SAVE function to solve for PMT, N, PV, or FV when the compounding period is not equal to the payment period, you must first solve for the equivalent periodic interest rate.

To do this you need to solve the SAVE function twice. Look at a problem similar to the first example, but with a different compounding period. Say you deposit $500 in the bank and plan to deposit $50 at the beginning of each month for 3 years. The annual interest rate is 6% compounded quarterly. How much will you have in the account at the end of 3 years? Type BEGIN since you deposit at the beginning of the month. First use SAVE to solve for FV using compounding periods. Define the variables like this:

PMT=0	No periodic payment
I=6%÷4	Annual interest rate ÷ Number of compounding periods per year

N=4	Number of compounding periods per year
PV=1	Initial deposit of $1

The computer solves for future value and displays FV=1.0613635 which is the future value of $1 left in the bank for 1 year with interest computed on it every compounding period. Now use SAVE to solve for the equivalent periodic interest rate, and define the variables like this:

PMT=0	No periodic payment
I=?	Equivalent periodic interest rate
N=12	Number of payment periods per year
PV=1	From previous SAVE
FV=1.061363	From previous SAVE

The computer solves for the equivalent periodic interest rate and displays I=0.004975202.

Now use this monthly interest rate to solve for the future value of your account in 3 years. Define the variables like this:

PMT=50	Monthly payment
I=0.00497520	Equivalent monthly interest rate
N=12×3	3 years of monthly payments
PV=500	Initial deposit

The computer solves for the future value and displays FV=2573.5198; you will have $2573.52 in this account at the end of 3 years. You can see that compounding quarterly earns you $1.46 less than compounding monthly as in the first example.

LOAN solves for the principal of a direct reduction loan with a balloon payment. In a direct reduction loan the periodic payments are all equal, but the portions of each payment applied towards the principal and interest are different for each payment. At first, most of the payment is applied toward the interest, then gradually more of each payment is applied to the principal. This is because you pay interest only on the unpaid balance which decreases with each payment.

Often in a direct reduction loan the last payment is not equal to the other payments. This last (and different, usually larger) payment is called a *balloon payment*. In some loans you make the balloon payment after the final periodic payment. However, all functions assume you pay the balloon payment at the same time as the final periodic payment.

LOAN uses these variables:

PMT	Periodic payment
I	Periodic interest rate
N	Number of payments, N≥0
FV	Balloon payment
PV	Loan amount

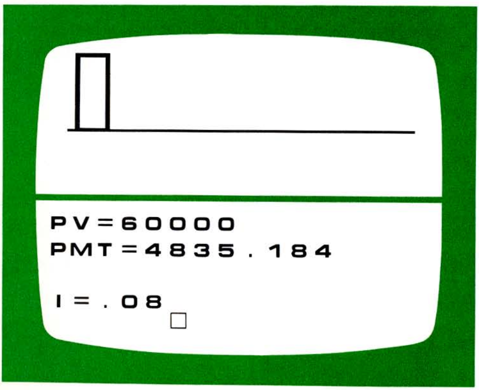
LOAN solves for any one of these variables. Be sure to specify BEGIN or END for annuity due or ordinary annuity.

Example: Suppose you want to take out a $60,000 house mortgage and you want to know what your approximate annual payments will be if the interest rates are 7%, 8%, 9%, and 10%. Typically, a mortgage has a life of 30 years with no balloon payment option.

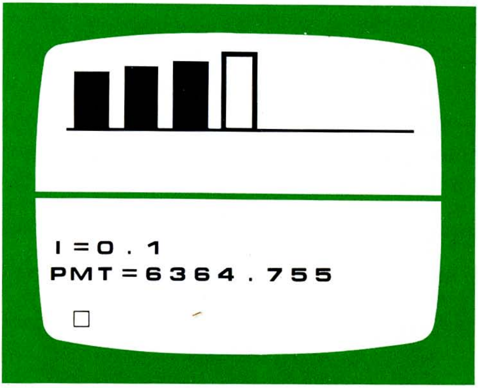
Type END for ordinary annuity and define the variables like this:

PMT=?	Solve for annual payment
I=7% TO 10% BY 1%	
N=30	Life of loan in years
FV=0	No balloon payment
PV=60000	Mortgage amount

The computer solves for PMT and displays:



which means that at 7% interest your payments are approximately $4,835.18 per year. Type **NEXT** and the computer solves for the annual payment with a 8% interest rate. After the fourth answer, when I=10%, the display looks like this:



Example: Suppose you want to buy a new car. You can pay $90 at the end of each month for 4 years at a 10% annual interest rate. You don't think you will be able to afford a balloon payment at the end. When you go shopping for a car, what priced car should you look for? Type END for ordinary annuity and define the variables like this:

PMT=90	Monthly payment
I=10%÷12	Monthly interest rate
N=4×12	4 years of monthly payments
FV=0	No balloon payment

The computer solves for the principal and displays PV=3548.5345 which means you should look for a car that costs less than $3550.

ACP solves for the accumulated principal paid on a direct reduction loan between two payment periods, possibly including a balloon payment. The accumulated principal between two periods is the amount of money from your payments which is applied toward the principal in this time period. ACP uses these variables:

PMT	Periodic payment
I	Periodic interest rate
SPN	Starting period number, the first period to start computing accumulated principal from, 1≤SPN≤EPN
EPN	Ending period number, the last period to compute the accumulated principal through, SPN≤EPN≤N
N	Number of payments
FV	Balloon payment
AP	Accumulated principal, amount paid toward principal from period SPN through EPN

ACP solves only for AP. Be sure to specify BEGIN or END for annuity due or ordinary annuity.

Example: You want to take out a $45,000 house mortgage with an 8.5% annual interest rate for 30 years with a $3000 balloon payment and monthly payments of $344.20. How much money from payments 20 through 30 is applied to the principal? Define the variables like this:

PMT=344.20	Monthly payment
I=8.5%÷12	Monthly interest rate
EPN=20	First period for computation
EPN=30	Last period for computation
N=30×12	30 years of payments
FV=3000	Balloon

The computer solves for the accumulated principal and displays AP=331.63386; you pay $331.63 toward the principal from periods 20 through 30.

ACI solves for the accumulated interest paid on a direct reduction loan between two payment periods, possibly including a balloon payment. The accumulated interest between two periods is the total amount of money from your payments which is applied toward the interest in this time period. ACI uses these variables:

PMT	Periodic payment
I	Periodic interest rate
EPN	Starting period number, the first period
SPN	to start computing accumulated interest from, $1 \leq SPN \leq EPN$
EPN	Ending period number, the last period to compute the accumulated interest through, $SPN \leq EPN \leq N$
N	Number of payments
FV	Balloon payment
AI	Accumulated interest, amount paid toward interest from period SPN through EPN

ACI solves only for AI. Be sure to specify BEGIN or END for annuity due or ordinary annuity.

Example: Using the same mortgage problem as in the previous ACP example, you can use ACI to calculate the accumulated interest on the mortgage for the same time periods. You do not have to change the values of any of the variables; ACI and ACP use the same variables to solve for AI and AP. So unless you typed **MASTER CONTROL** or redefined some of the variables since doing the ACP example, you only have to type **NEXT** after the computer displays each variable.

The computer solves for the accumulated interest and displays AI=3454.5661 which means that $3,454.57 of your payments in this period are applied to the interest, compared to the $331.63 applied to the principal.

AOL solves for the periodic payment in an add-on-interest loan. In an add-on-interest loan all payments are equal but you calculate interest using the add-on-interest rate. AOL uses these variables:

AOI	Annual add-on-interest rate
N	Number of payments in months, $N \geq 1$
PV	Loan amount
PMT	Monthly payment

AOL solves for any one of these variables.

Example: Suppose you take out an add-on-interest loan for $5,000 with a 5% annual add-on-interest rate over 3 years. How much is your monthly payment? Define the variables like this:

AOI=5%	Annual interest rate
N=36	3 years of monthly payments
PV=5000	Amount of loan

The computer solves for the payment and displays PMT=159.72222; your monthly payment is $159.73.

Example: To illustrate how an add-on-interest loan is often not as good as it first appears, you can convert the add-on-interest rate from the previous example to a true interest rate by using the LOAN function. Type END, and use these definitions:

PMT=159.722	Monthly payment
I=?	Solve for periodic interest rate
N=36	3 years
FV=0	No balloon payment
PV=5000	Amount of loan

The computer solves for the interest and displays I=0.007758756 which is the monthly interest rate. Multiply this by 12 to get the annual rate: $I \times 12 = 0.09310507 = 9.31\%$. The 5% add-on-interest rate is equivalent to 9.31% true interest rate.

IPP solves for the payment for a particular period on an installment purchase plan where the principal is paid in equal payments, but the total payment includes an interest charge on the unpaid balance. IPP calculates the payment amount for a specified period, and uses these variables:

I	Periodic interest rate
N	Number of payments
PV	Loan amount
PN	Period number to compute the payment amount for, $1 \leq PN \leq N$
PMT	Payment for period PN

IPP solves for any one of the variables, except PN.

Example: You buy a stereo system for $800 at an annual interest rate of 16% with one year to pay for it. How much will your first 10 payments be? Define the variables like this:

I=16%÷12	Monthly interest rate
N=12	1 year of monthly payments
PV=800	Price of system
PN=1 TO 10	Want PMTs for 10 periods

The computer displays the PMTs for the 10 periods; the 10th month you pay $69.34.

R78 solves for the amount you save by paying off a loan before the original life of the loan is reached, using the Rule of 78. The amount of money you save is called a *rebate*. R78 uses these variables:

N	Number of payments
PV	Loan amount
PMT	Periodic payment
PN	Period number when you pay off the loan, 0≤PN≤N
REB	Rebate

R78 solves only for REB.

Example: You borrow $4,000 at 8% annual interest to do repairs on your house. You make monthly payments of $97.65 and you have 4 years to pay. Suppose that after 3 years you sell the house and are able to pay off the loan. How much do you save by paying off the loan early? Define the variables like this:

N=48	4 years of monthly payments
PV=4000	Principal
PMT=97.65	Periodic payment
PN=36	Pay off loan at 36th payment

The computer solves for the rebate and displays REB=45.57959; you save $45.58 by paying off the loan early. You can use R78 to calculate the total amount needed to pay off the loan at a particular period. First use R78 to solve for the rebate. Then type this formula followed by **NEXT**:

PMT×(N−PN+1)−REB

The computer solves it using the values from R78 and displays the answer which is the total amount needed to pay off the loan, $1223.87 in this example.

ROI solves for the net present value of periodic cash flow generated by an investment, for a discount rate. The cash flow in each period may be different, and a cash flow may be negative. Many investments lose money at first (negative cash flow) but hopefully they eventually return a profit. The rate is the minimum desired yield from the investment.

The net present value is the sum of the present values of all the periodic cash flows, assuming a discount rate. If the net present value of an investment is less than 0, then the actual rate of return is less than this rate. If the net present value is equal to 0, the actual rate of return is equal to this rate. If the net present value is greater than 0, the actual rate of return is greater than this rate.

ROI uses these variables:

N	Number of periods, $1 \leq N \leq 15$
C0	Initial investment
C1	Cash flow in period 1
C2	Cash flow in period 2
⋮	
CN	Cash flow in period N
I	Discount rate
NPV	Net present value

ROI solves for either NPV or I.

Example: You want to buy an apartment building for $130,000, rent it for 5 years then sell it. The first 2 years you must put money into improvements, but for 3 years after that you don't foresee major expenses. If the cash flow for the next 5 years is −$2500, −$2000, $8800, $12,200, and $13,400, and you can sell it for $180,000, will this investment return 10% each year? Define the variables like this:

N=5	5 year investment
C0=−130000	Cost of building
C1=−2500	1st year
C2=−2000	2nd year
C3=8800	3rd year
C4=12200	4th year
C5=13400+180000	5th year cash flow plus sale price
I=.10	Discount rate

The computer solves for the net present value and displays NPV=1104.8986. Since NPV is greater than 0, you know this is an investment which returns at least 10% each year.

Example: You can use the ROI function to solve for the internal rate of return which is the rate of return that gives you a net present value of 0. To solve for the internal rate of return in the previous example, use the same values for N and C0 through C5, but define I and NPV like this:

I=?
NPV=0

The computer solves for the internal rate of return and displays I=0.10188507 which means that the internal rate of return is 10.19%.

Depreciation (SL, SOD, DB). Most material assets such as houses, cars, or machines, all decline in value as time passes. Depreciation is the decrease in value of an asset from the time it is new until a later period in time. For example, if you bought a car five years ago for $4,500 and now you can only sell it for $2,000, the depreciation is $2,500. You can not compute the exact depreciation of an asset until you sell it. However, you can estimate depreciation using three different methods: the straight line, sum-of-the-years-digits, or declining balance methods. The functions corresponding to these three methods are SL, SOD, and DB. (Be sure to check IRS regulations if you are using VB-81 for tax purposes.)

SL solves for depreciation per period and book value at any particular period using the straight-line method. Use this function when the amount of depreciation is equal in every period.

SOD solves for the depreciation per period and book value at a particular period using the sum-of-the-years-digits method. SOD assumes the asset depreciates the most the first year, and a little less each year after. This method is based on the sum of the digits from the number of years you have the asset down to 1. For example, if the useful life of a computer system is 5 years, the sum-of-the-years-digits is:

$$5+4+3+2+1 = 15$$

This means the first year the system depreciates 5/15 of the total depreciable value (value less salvage), the second year is 4/15 of the total, the third year is 3/15 of the total, and so on.

DB solves for the depreciation and remaining book value for any particular period using the declining balance method. In this way, the depreciation is calculated by taking a certain fraction of the remaining book value for each period. As opposed to SL and SOD methods, the salvage value is not subtracted initially; but the asset may not be depreciated below the salvage value. The fraction taken is determined by a decline factor, which is a value from 1 to 2 that is authorized for income tax purposes. To compute the fraction, divide the factor by the estimated life (in years) of the asset. For example, if the decline factor is 1.5 and the life of the asset is 5 years, the fraction applied to the remaining book value for each period is 1.5÷5=.3.

The depreciation functions use these variables:

PV	Cost of asset
FV	Future value, salvage value of asset at period N, 0≤FV≤PV
N	Depreciable life of the asset in periods
PN	Period number to find the depreciation and book value for, 1≤PN≤N
DF	Decline factor, 1≤DF≤2 (DB only)
DEP	Depreciation for period PN
BV	Book value for period PN

The depreciation functions solve for DEP and BV.

Example: Suppose you buy a computer system for $22,000 with an estimated salvage value of $8,000 in 5 years. Using the SOD method, how much does the system depreciate the first year and what is the book value after the first year? Define the variables like this:

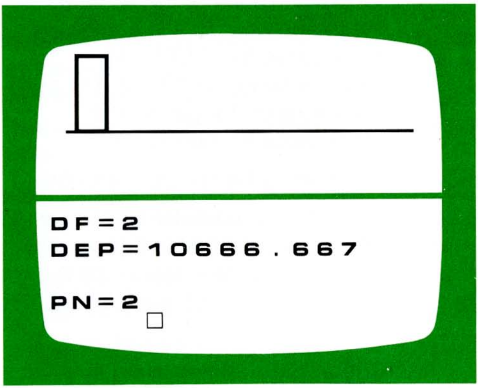
PV=22000	Cost of asset
FV=8000	Salvage value
N=5	Depreciable life
PN=1	Want DEP and BV after 1st year

The computer solves for depreciation and displays DEP=4666.667 which means the system depreciates $4666.67 the first year. Type BV and **NEXT**; the computer displays 17333.333 which means the undepreciated balance is $17,333.33 after the first year.

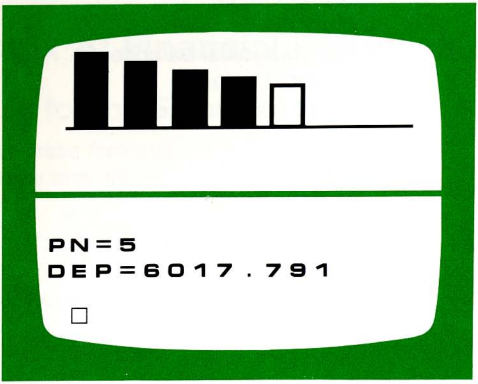
Example: Suppose your realtor recommends that you invest in a new duplex. The realtor points out that one of the advantages is that you can accelerate your depreciation losses on a new building by using the double declining balance method so you pay less taxes. The duplex costs $80,000 and has an estimated salvage value of $10,000 in 15 years. If the decline factor is 2, calculate a depreciation schedule for the first 5 years. Use DB and define the variables like this:

PV=80000	Building cost of duplex
FV=10000	Estimated salvage value
N=15	Depreciable life is 15 years
PN=1 TO 5	Calculate for first 5 years
DF=2	Decline factor

The computer solves for DEP and displays:



The duplex depreciates $10,666.67 the first year. Continue solving the function for other PN values through the 5th year. The last display is:



The duplex depreciates $6017.80 the 5th year. The 5 bars show the depreciation over the first 5 years.

CALC lets you use the computer like a simple calculator to do +, −, ×, and ÷ operations. CALC uses one variable called R. As soon as you type CALC followed by **NEXT**, CALC displays the value of R which is 0 unless R has been set with an assignment statement.

The rules for using CALC are simple. If you type a value which is not preceded by an operator, this value is the new value of R. If you type an operator followed by a value, the computer solves the operation, displays the answer, and uses the answer as the new value of R.

To end CALC, type **NEXT** on a blank line. The computer displays ENDED and waits for you to type the next formula or function name.

Example: Suppose you have a checking account. The following sequence shows you how to use CALC to compute the balance:

CALC	Start-up CALC
0	Initially R=0
321.50	Enter your initial balance
321.5	Your initial balance
+1500.12	Deposit your paycheck
1821.62	
−523	Check 1
1298.62	
−290.21	Check 2
1008.41	Your current balance
	Type **NEXT** to end
ENDED	

Appendix A: Financial Formulas

Formulas for the Financial Functions

SAVE uses these formulas ($N \geq 1$):

For ordinary annuity:

$$FV = \sum_{i=0}^{N-1} PMT(1 + I)^{i} + PV(1 + I)^{N}$$

For annuity due:

$$FV = \sum_{i=1}^{N} PMT(1 + I)^{i} + PV(1 + I)^{N}$$

LOAN uses these formulas ($N \geq 1$):

For ordinary annuity:

$$PV = \sum_{i=1}^{N} \frac{PMT}{(1 + I)^{i}} + \frac{FV}{(1 + I)^{N}}$$

For annuity due:

$$PV = \sum_{i=0}^{N-1} \frac{PMT}{(1 + I)^{i}} + \frac{FV}{(1 + I)^{N-1}}$$

ACP and **ACI** use these formulas:

For ordinary annuity:

$$BAL(0) = PV$$
$$BAL(i) = BAL(i - 1) - PRN(i)$$
$$INT(1) = PV \times I$$
$$INT(i) = BAL(i - 1) \times I$$
$$PRN(i) = PMT - INT(i)$$

For annuity due:

$INT(1) = 0$

All other variables are the same as ordinary annuity variables.

$$AP = \sum_{i=SPN}^{EPN} PRN(i) \quad (+\ FV \text{ if } EPN = N)$$

$$AI = (EPN - SPN + 1) \times PMT - AP$$
$$(+\ FV \text{ if } EPN = N)$$

AOL uses this formula:

$$PMT = \frac{PV}{N} \times (1 + AOI \times \frac{N}{12})$$

IPP uses this formula:

$$PMT = \frac{PV}{N} + PV \times I \times (1 - \frac{PN - 1}{N})$$

R78 uses this formula:

$$REB = (PMT \times N - PV) \times \frac{(N - PN) \times (N - PN + 1)}{N \times (N + 1)}$$

ROI uses this formula:

$$NPV = \sum_{i=0}^{N} \frac{Ci}{(1 + I)^i}$$

SL uses these formulas:

$$DEP = \frac{PV - FV}{N}$$

$$BV = FV + (N - PN) \times \frac{PV - FV}{N}$$

SOD uses these formulas:

$$DEP = (PV - FV) \times 2 \times \frac{N - PN + 1}{N \times (N + 1)}$$

$$BV = (PV - FV) \times (N - PN) \times \frac{N - PN + 1}{(N \times (N + 1))} + FV$$

DB uses these formulas:

$$BV = FV$$

$$\text{or } BV = PV \times \left(1 - \frac{DF}{N}\right)^{PN}, \text{ whichever is larger}$$

$$DEP = FV - BV$$

$$\text{or } DEP = PV \times \left(1 - \frac{DF}{N}\right)^{PN-1} - BV, \text{ whichever is larger}$$

Technical Notes

The largest number you can use is approximately 10^{18}.

The smallest number you can use is approximately 10^{-18}.

π has the value 3.14159265.

You should not use very small interest rates like $I < .00001$.

When N is an exponent in a formula, the computer uses the integer closest to N.

When the computer solves for N and N is an exponent in a function, the computer's approximation of N is bounded by the same adjacent integers as N.

Appendix B: Error Messages

Sometimes when you are using VB-81, the computer displays a message on the screen. The messages generally mean that the computer can not do what you are telling it to do. Either the statement you typed is incorrect, or the computer is not able to solve the formula or function. Following is a list of possible error messages and an explanation of each one.

? You are trying to recall a variable that does not have a value assigned to it. For example, you type X8 and **NEXT** without previously assigning a value to X8.

SYNTAX ERROR STMT IGNORED The statement you typed does not make sense to the computer so it ignores the statement. Either you do not have an even number of parentheses in your formula, your formula has a × or ÷ operator immediately after another operator, you are using a variable name that does not begin with a letter, or you are using TO or BY improperly.

VALUE ERROR Either the answer to your formula is greater than 10^{18}, your formula contains variables that are not defined, you are trying to divide by 0, or you are giving an illegal value to DF (DB function), PN (IPP, R78, SL, SOD, or DB functions), SPN (ACP or ACI functions), EPN (ACP or ACI functions), FV (SL, SOD, or DB functions), or N (all functions).

CAPACITY ERROR RESTARTING Your formula has more than 9 nested parentheses.

CAPACITY ERROR STMT IGNORED Either your formula has more than 10 numbers and variables in it, or one of your variables is larger than 10^{18}.

MISSING VALUES The computer can not solve a function using the definitions for the variables that you gave it. Either you are trying to solve for two variables in a function at once, or you are using a function to solve for a variable that the function can not solve for. Some functions can solve only for certain variables. For example, the ROI function solves only for NPV or I. If you try to solve for C0 using the ROI function, the computer displays MISSING VALUES followed by C0= and waits for you to define C0 for it.

I UNDER 0 You are using a function to solve for I (interest) and I is negative. Interest is always a positive number.

I OVER 5 You are using a function to solve for I and I is over 5 which is 500% and not an acceptable interest rate.

N UNDER 0 You are using a function to solve for N (number of periods) and N is negative.

N OVER 512 You are using a function to solve for N and N is greater than 512.

N MUST BE 0-15 You are using the ROI function and did not define N correctly. N must be between 0 and 15 for this function.

ENDED You cancelled a function early by typing **ERASE** and **NEXT**, or just **NEXT** after a blank line.

VideoBrain Cartridge Programs

Money Management

VB-59	The Programmable
VB-81	Financier
VB-1000	Money Manager
VB-1100	Budget System
VB-1200	Information System

Communications

CM01	Timeshare

Education

ED01	Music Teacher 1
ED02	Math Tutor 1
ED03	Wordwise ™ 1
ED04	Wordwise ™ 2
ED05	VideoArtist ™
ED06	Lemonade Stand
ED07	Music Teacher 2
ED08	Number Cross

Entertainment

EN01	Gladiator
EN02	Pinball
EN03	Tennis
EN04	Checkers
EN05	Blackjack
EN06	Vice Versa
EN07	Challenge Racer
EN08	Music Programmer
EN09	Programmable Football

Limited 90-Day Warranty on Cartridges:

For 90 days from the date of purchase, VideoBrain Computer Company will repair any defect in material or workmanship in this Cartridge free of charge.

To obtain warranty service, return the Cartridge postpaid, with sales receipt showing date of purchase, to the VideoBrain Service Center with address shown below.

Under no circumstances will VideoBrain Computer Company be liable for any special, incidental, or consequential damages resulting from use or possession of the VideoBrain or its accessories. However, some states do not allow the exclusion or limitation of incidental or consequential damages, so that the above limitations or exclusions may not apply to you.

This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.

VideoBrain Computer Co.
An Umtech Company
150 South Wolfe Road
Sunnyvale, California 94086
408/737-2680

9320011 0478/C/TL/5K