

## 2

# Budget

This is a tape-based accounting program suitable for personal use by any member of the family. It can be used for club or committee finances and has applications in small businesses. It can handle up to eighty different accounts, each identified by name or code number and can deal with up to two thousand separate transactions at one loading. The well known disadvantages of using tape as a storage medium have been minimised by the design of the program. Data is efficiently compacted before recording it, so as to reduce the length of tape used and to cut down the time required for saving and loading.

As well as the BUDGET program, there is a special OPENING program. This is used when you start a new accounting period (including the first time that you use BUDGET). It allows you to begin the period with amounts already held in various accounts. For example, you may already have savings at the bank when you begin your accounting. Or you may have items of equipment or stock that you want to be included in your accounting.

The BUDGET program displays ledger accounts in the familiar format of double-entry bookkeeping. Do not worry if this sounds complicated to you - it is all very simple when it is the computer which has to do the hard work! There are several examples given to make it easy for you to understand how to use the program.

The program has been designed to do as many useful things as possible, and to handle data efficiently and as fast as possible. The input routines are designed to make it difficult to make mistakes when using the program. The result is that this program is one of the longest in the book. This is inevitable if it is to do its job properly.

## Using the program

Two tapes are required:

- (1) The PROGRAM tape, carrying the OPENING and BUDGET programs.
- (2) The ACCOUNTS tape, carrying the details (NAMES and AMOUNTS) of all the accounts.

You may also want to have an additional ACCOUNTS tape (or tapes) for backup. A C-10 tape is long enough to hold the maximum number of two thousand transactions.

The instructions below are set out in sections for ease of reference.

### *(A) Starting your accounts*

Before loading the OPENING program, work out on paper the amounts of your opening entries. Figure 2. I shows opening entries for a more formal account, such as might be used for a club or small business.

The meanings of the words *credit* and *debit* need to be explained. When we say that an account is in debit, we mean that there is money in it. When you are paid money (say for a job you have done) you debit your CASH account with amount. So a *debited* amount is an amount which is *paid in*.

Conversely, an account which is in credit is one out of which you have paid money. If you pay the plumber for mending a burst pipe, you credit your CASH account with that amount. A *credited* amount is an amount that is *paid out*.

When you begin a home budget account you will enter any sums in the bank or in your pocket as accounts in debit. These are amounts which can be made available for you to spend. You could enter all these amounts in one account, called CASH, or keep separate accounts for BANK and POCKET. Of course, if you owe the bank money, either because of a loan or overdraft or because you have not paid your credit card account, this item must be entered as a credit on your CASH account. The same applies to any amounts you owe to your friends or for unpaid bills. Should any of your friends owe you money then this amount should be entered as a debit. The balance of these amounts will then be brought down as the opening balance on your own home budget account.

If you simply want to start your home account from scratch, use the program to enter a single account. You might call it 'H'KPING' (for 'housekeeping') or 'CASH'. Start it with '0.00' as the amount. You can type 'C' or 'D' in answer to the 'Debit or Credit' question.

---

OPENING ENTRIES	ASSETS	LIABILITIES
	£	£
Equipment	1500.00	
Cash	700.00	
Balance c/d to Capital a/c		2200.00
	2200.00	2200.00

The opening balances of the Equipment account and the Cash accounts are debit. The opening balance of the Capital account is credit.

OPENING ENTRIES	ASSETS	LIABILITIES
	£	£
Equipment	1500.00	
Cash		300.00
Balance c/d to Capital a/c		1200.00
	1500.00	1500.00

The opening balance of the Equipment account is debit. The opening balances of the Cash account and the Capital account are credit.

OPENING ENTRIES	ASSETS	LIABILITIES
	£	£
Equipment	348.50	
Cash		1255.00
Balance c/d to Capital a/c	906.50	
	1255.00	1255.00

The opening balances of the Equipment account and the Capital account are debit. The opening balance of the Cash account is credit.

---

*Fig 2.1 Basic examples of opening entries.*

Opening an account as a business is similar, though there are special names given to certain types of account. If you are starting a new business or other enterprise, the amounts to be accounted for will usually come under three headings (see Fig. 2.1).

In the CASH account: if you have money in the bank this is a debit on the opening balance. If you start with a bank overdraft this is a credit on the opening balance. Remember, a debit is a sum which is owed to you. When you put money in the bank, the bank then owes you that money.

In the EQUIPMENT account: the value of equipment, stock or other items which you already have when the business is started. This is a debit item, in the sense that you might later sell part of the equipment, and the money for this sale would be owed to the business.

In CAPITAL account: this must equal the total of the opening balances of the CASH and EQUIPMENT accounts. It represents what the business owes to the proprietor or to those who have lent money to start the business. Figure 2.1 will help you decide if this is a credit or debit balance.

### *(B) The OPENING program*

Load and run the OPENING program. Note that the programs are able to accept account names (up to 7 characters) which are typed in capital letters (e.g. CASH), or in a mixture of capital and lower-case letters (e.g. Cash). Look at the small lamp beside the CAPS LK FUNC key. If the lamp is on, your typing will appear in capitals. If the lamp is off, typing will appear in lower-case letters and you need to press SHIFT to obtain capitals. To change from one mode of typing to the other, hold down the SHIFT key, then press the CAPS LK FUNC key. The lamp tells you when the computer is in the required mode.

The OPENING program asks you for the date (of opening the accounts), the account names, the opening amounts and whether they are debits or credits. Section G explains how to enter dates; Section H explains how to enter amounts. As you finish entering the details of each account, you are asked 'ALLOK? (Y/ N)' If you are satisfied that you have typed in everything correctly, key ' Y' Otherwise, key ' N' and you are asked to enter all the details of that account again.

Typing ' Y' makes the Electron open an account with the details you have typed in. After each account has been opened, you are asked ' MORE?' Key ' Y' to open another account; key ' N' when you have opened all the accounts you need to start with. There is no need to enter accounts which begin with zero balances, since this is done in the main BUDGET program. The OPENING program allows you to start with between one and nine accounts.

When you have typed in the last account details and have pressed ' N' to indicate that no more accounts are required or when you have

typed in details of nine accounts, the message ' RECORD then RETURN' appears on the screen. Insert the ACCOUNTS tape in the recorder and rewind the tape on fast forward so that it is ready for recording. Then press the RECORD and PLAY buttons on the recorder. Press RETURN on the Electron. After about ten seconds a further ' RECORD then RETURN' appears on the screen. The computer has saved the account names and is now waiting to save the details of amounts. Press RETURN again. The word ' AMOUNTS' appears on the screen, followed by some figures. After about ten seconds the recorder stops and the information has all been saved on tape. The OPENING program has now finished.

### *(C) Starting the BUDGET program*

Now LOAD and RUN the BUDGET program. You are first asked to press the ' SpaceBar when ACCOUNTS tape ready' If you have not already done so, remove the PROGRAM tape from the recorder and replace it with the ACCOUNTS tape. Then press the space-bar. You can then rewind the tape, if necessary, to the beginning of the recording you have just made with OPENING. Next, press PLAY (only). After a few seconds, the ' Searching message is displayed. Your opening balances will then be read into the computer' s memory. This takes about ten seconds. As soon as this is done, the recorder is switched off, and the screen displays the menu.

### *(D) Using the menu*

The menu invites you to choose between four options but, if you have entered only one account when you used OPENING, you can select only option 3 at this stage. If this is the case, you are reminded of it by the two asterisks before the ' 3' Obviously the program needs at least two accounts on its records before it can start to transfer sums from one account to another.

The menu is the central point in the operation of the program. Whenever you select an option (by pressing the key which has the same number as the option) you are sent to the part of the program which performs that task. Afterwards, (unless you select ' 4Finish' ), you are returned to the menu to decide what you want the computer to do next.

The options work as follows:

*(1) Display account:* A list of the account names or codes is displayed and you are asked to choose any one by typing its number, as listed.

You are also asked to type in the number of the month from which you wish the display to begin. Transactions are displayed from the first day of that month. You are then asked to type in the number of the month up to which the account is to be displayed. It will be displayed up to the last day of that month. If you want to see everything on the account, simply type '1' as the 'From' month and '\*12' as the '^To' month. The account is then displayed, showing the opening balance (if any), every transaction and its date, and amounts debited or credited and the balance carried down at the end of the period. If there are many transactions in the account, the display stops when the screen is full. Press the SHIFT key to view the next screenful of transactions. Figure 2.2 shows Mrs Smith's housekeeping account for two weeks.

---

Account name: H' kping

Date	Details	Debit	Credit
01/04	OPEN.BAL.	0.00	0.00
06/04	Cash	50.00	0.00
06/04	Dairy	0.00	3.26
07/04	Garden	0.00	5.10
09/04	Baker	0.00	3.45
10/04	Grocer	0.00	25.63
12/04	TV	0.00	3.95
12/04	Butcher	0.00	9.48
13/04	Cash	50.00	0.00
14/04	Dairy	0.00	3.56
14/04	Car	0.00	8.25
16/04	Baker	0.00	2.96
17/04	Groder	0.00	22.31
18/04	TV	0.00	3.95
18/04	Butcher	0.00	7.95
	Bal c/d	0.00	0.15
		100.00	100.00

---

Fig. 2.2 Mrs Smith's housekeeping account.

(2) *Transaction*: Home users may be unfamiliar with terms such as *credit balance*, and *double entry* accounting. The essence of this accounting system is that money always comes *out of* somewhere and goes *into* somewhere else. We need an account from which money is taken (this account is credited) and an account into which the money is paid (the account which is debited). For example, if Mrs Smith pays

the baker for a loaf of bread, she needs an account called ' Baker and an account called ' Housekeeping'. The Baker account is debited with the cost of the loaf, while the ' Housekeeping' account is credited with the same amount. These two operations are performed together by the computer when you use Option 2. Mrs Smith's housekeeping account will look like Fig. 2.2 and her Baker account will look like Fig. 2.3. You have to open an account for each different kind of item you spend money on. This makes it very easy to check just how much you spend with people such as the butcher, the baker and the electricity board. The program can hold up to eighty different accounts, so allowing you to analyse your expenditure in detail. However, there are some items which could be grouped together under one main heading. For example you would not want separate accounts for shoes, socks, jeans and hats, but would charge all of these to a single account called Clothes.

---

Account name: Baker

Date	Details	Debit	Credit
09/04	H' kping	3.45	0.00
16/04	H' kping	2.96	0.00
	Bal. c/d	0.00	6.41
	Bal. c/d	6.41	6.41

---

Fig. 2.3. Amounts paid from one account always appear in another account

When you select Option 2, a numbered list of account names is displayed. You are asked to key in the number of the account which is to be credited (i.e. the account out of which the sum of money is to be paid) and the number of the account to be debited (i.e. the account into which the money is to be paid). You are also asked the date of the transaction (see Section G). If your accounts begin from January of each year, enter the day of the month (' 01 to ' 29' 30 or ' 31' followed immediately (no space or punctuation) by the month number (' 01 to ' 12'. If your accounts begin in some other month and you want to keep accounts for a whole year, enter ' 01 for the first month in your accounts year, ' 02 for the second month, and so on. Transactions should be entered in order of date as far as possible. The essential point is that all transactions taking place in a given month must be entered before you enter any transactions of the following month.

If, on looking at the list of accounts, you find that one of the

accounts is not on the list, key ' 0' as the account number. This takes you direct to the ' Gpewew account' option, so that you can open the ' missing' account, as described in the next section.

(3) *Open new account:* The first question is ' What's the name of the new account?' Key this in, using all capitals or both capitals and lower-case letters as preferred. Note that account names can be only seven characters long. Any characters in excess of this are chopped off by the computer. Account names, though referred to as such in the program and in this description, need not be namex. You can use codes made up of letters and numerals, or even solely of numerals. They can include punctuation marks, except for commas. Press RETURN after you have typed in the account name. The computer searches its memory to discover if an account of that name has already been opened. If so, it informs you of this fact and then returns you to the menu. If the account is not open already, it displays the name of the account again, so that you can confirm that you have typed it in correctly. If the name you typed had more than seven characters, only the first seven are displayed. You are then asked to confirm that the name is correct. If so, you are returned to the Menu. If not, you are asked to enter it again.

(4) *Finish:* It is essential to use this option at the end of every session using the program. If you simply switch off, all data entered during the current session is lost.

The first message to appear is ' RECORD then RETURN' Rewind the ACCOUNTS tape to the beginning, so that it is ready to record the new data over the old track. Then press RECORD and PLAY on the recorder, followed by RETURN on the Electron. After about ten seconds, another ' RECORD then RETURN' message appears. Press RETURN again. The word ' AMOUNTS' appears, followed by a number. This runs through the hexadecimal values from ' 00' to ' 36' At this point the recorder stops. This process takes slightly over three minutes. The length of time is fixed, and does not depend on how many transactions or accounts there are. It will take no longer even when you have as many as two thousand transactions on memory.

When the recorder stops you are offered the option of taking a backup copy of the transactions. If you decide to do this, which is advisable, put another tape labelled '\*ACCOUNTS' in the recorder. Alternatively, you can record the backup on the same tape, immediately after the first recording. If you key \*Y?, the procedure

described above is repeated. If you press ' N'the program ends. It is now safe to switch off the computer, rerun BUDGET from the beginning, or load and run another program. Before loading another program, press BREAK, to make the full amount of memory available for the new program.

*(E) Using BUDGET again*

The procedure is exactly the same as in Section C, above. The only difference is that you now have a full-length AMOUNTS file on tape. The loading of this takes a little over three minutes, no matter how few or how many transactions are on record. Then use the program exactly as described in Section D.

*(F) Using OPENING again*

This program is used whenever you want to start a new set of accounts. You will need to use it after your accounts have been open for a full year, or when the number of transactions reaches two thousand.

*(G) Typing dates*

The format used for dates requires four figures:

DDMM

where DD are the figures for the day of the month and MM are the figures for the number of the month. Thus 1 st April is represented by ' 0104' ,and 1 November by ' 2111' .if your accounting year doesn't begin on 1st January, you may number the months differently, as described in Section D (2).

Although the program does not prevent you from entering imaginary dates such as 31st February, it does check that the days are not more than 31 and the month is not more than 12. If you get no response when a date is typed in, you have probably typed in an illegal date. Type the correct version on top of the incorrect version.

*(H) Typing amounts*

Sums of money are entered as:

up to 5 digits, for the pounds  
 a full stop (point)  
 two digits, for the pence.

There is no need to press RETURN after entering the second pence digit; the program continues immediately. If you have entered 5 ' pounds' digits, the program will not accept anything other than a full stop, then the two ' pence' digits. Examples of acceptable amounts are:

£30305.45  
£555.06  
£67.00

The largest sum that can be entered is £65535.99. If you try to enter a sum larger than this amount, the computer will wait until it has been overtyped to correct the error.

### *(1) Getting out of trouble*

Although care has been taken to make it difficult or even impossible for the computer to accept incorrect entries, there are still occasions on which you may press the wrong key by mistake. The hints below will help you recover without upsetting the program:

(1) You keyed Options 1 or 2 and then found that the account you thought had been opened had not already been opened. Key ' 0 when asked for the account number required. This takes you directly to Option 3, to open the ' missing' account.

(2) You have keyed Option 3 in mistake for one of the other options, and have no new account to be opened. Type the name of an account which you know is already open. The computer will then tell you that this account is already open and, after a short pause, send you back to the menu.

(3) You have made a mistake in entering details of a transaction and it is already on the account concerned. There is no way of deleting transactions from the accounts. Instead, follow the usual book-keeping practice of making a *contra entry*. For example, if you entered a payment of £25.45 from the cash account to the fuel account, when it should have been £25.54, proceed as follows. Make a payment of £25.45 from the fuel account into the cash account (this is the contra entry). Then make a payment of the correct amount (£25.54) from the cash account into the fuel account.

(4) You accidentally press the ESCAPE key and halt the BUDGET program. Key in ' GOTO 80' and press RETURN. This will take you back to the menu without loss of data.

(5) You accidentally press the BREAK key. You can recover the BUDGET program by typing OLD and then pressing RETURN. Unless you happened to press BREAK at an unfortunate state of the computer's action, it is possible to recover the transaction details, though not the names of any new accounts that you have just opened. With luck, you may be able to remember the names.

Having recovered the program, as described above, delete line 170 by typing '170' then pressing RETURN. Now RUN the program, using the ACCOUNTS tape as usual. It will take only a few moments to load in the account names. Then the menu appears. If you had added any new accounts prior to pressing BREAK, add these new accounts to the list, in the usual way. If you cannot remember their names, use dummy names, such as 'AC1' 'AC2' etc. You will then be able to display these accounts under their dummy names and, from the amounts you find displayed and from your receipts and bank statements you should be able to discover their real names. However, you will have to refer to them by their dummy names from now on.

## Keying in

In OPENING, line 100, there are seven spaces between the quotes. In BUDGET, take particular care with the colons and semicolons, for errors with these can ruin the display. For the same reason, be sure to key in the correct number of 'ticks' after the PRINT statements. Mistyping of some of the numbers could cause chaos in both programs when they are run. Numbers to take especial care with are those in lines 30 and 430 of OPENING and in lines 30, 170, 340, 590, and 1540 of BUDGET. Letter 'o' is not often used in variable names, but we use it in OPENING, in line 80 and in BUDGET, in lines 1170-1190, in the variables MO\$ and MO.

## Program designs

### OPENING

20-40 Initialising, including protecting memory to receive the data (line 30).

50-80 Inputting and checking the date.

90-100 Inputting the account name and shortening it if necessary.

110-210 Inputting and checking the amount.

### 34 *Practical Programs for the Electron*

- 220-240 Credit or debit?
- 250-270 Confirming entries.
- 280-340 Storing details in memory.
- 350-380 More accounts?
- 390-440 Saving NAMES and AMOUNTS files.

#### BUDGET

- 20-70 Initialising.
- 80-110 Waiting to load from tape.
- 120-170 Loading from tape.
- 180-270 Displaying the menu.
- 280-330 Selection of options.
- 340-380 Displaying the list of account names.
- 390 Branch to input transaction details.
- 400-500 Details for account to be displayed.
- 510-560 Calculating the balance of an account for any period prior to the period for which it is to be displayed (i.e. balance brought down).
- 570-650 Displaying headings of account and balance brought down.
- 660-750 Searching memory for transactions on that account, and displaying them, accumulating credit and debit totals.
- 760-880 Calculating balance carried down and total balances and displaying them.
- 890 Return to menu, with warning if memory full.
- 900-930 Inputting name of new account.
- 940-970 Searching to see if an account in this name already exists.
- 980-1060 Confirming account name.
- 1070-1190 Inputting transaction details; account numbers and date.
- 1200-1310 Inputting amount.
- 1320-1420 Confirming that details of transaction are correct.
- 1430-1450 Storing transaction details in memory.
- 1460-1470 Return to menu, with warning if memory is nearly full.
- 1480-1540 Saving names and details to tape.
- 1550-1590 Optional backup.
- 1600-1630 PROCdelay; delay to allow user time to read warning messages.

## Points of interest

The names of accounts are held in the array N\$( ). They are saved in a file ' NAMES' preceded by N, the number of accounts and NR, the number of transactions.

The details of transactions are stored directly in memory in blocks of 7 bytes. Set aside for this purpose are 140(0) bytes of memory, allowing two thousand transactions to be stored in a very compact form. This block of memory extends from address 10576 (&2750) to 24575 (&5FFF). The contents of each block of 7 bytes are:

Byte 0	The number of the account credited.
Byte 1	The number of the account debited.
Byte 2	The day.
Byte 3	The month.
Byte 4	The number of pounds (high byte).
Byte 5	The number of pounds (low byte).
Byte 6	The number of pence.

The actual number of pounds is found by multiplying the contents of Byte 4 by 256 and then adding the contents of Byte 5.

The transaction details are saved and loaded in a file ' AMOUNTS' by using the operating system commands \*SAVE and \*LOAD.

## The program (OPENING)

```

10 REM ** OPENING BALANCES **
20 MODE 4
30 HIMEM=10576
40 DIM N$(9):N=1
50 CLS:INPUT TAB(0,5)"DATE (DDMM)? "D
$
60 IF LEN(D$)<>4 THEN 50
70 DA$=LEFT$(D$,2):IF VAL(DA$)<1 OR V
AL(DA$)>31 THEN 50
80 MO$=RIGHT$(D$,2):IF VAL(MO$)<1 OR
VAL(MO$)>12 THEN 50
90 INPUT TAB(0,7)"ACCOUNT NAME (7 CHA
RS)? "N$(N)
100 N$(N)=LEFT$(N$(N)+"",7)
110 NK=0:AM$="":PRINT TAB(0,9)"AMOUNT
(POUNDS,PENCE)? ";

```

### 36 Practical Programs for the Electron

```
120 A$=GET$
130 IF VAL(A$)=0 AND A$<>"0" AND A$<>"
." THEN 120
140 IF NK=5 AND A$<>"." THEN 120
150 PRINTA$;:IF A$="." THEN 170
160 AM$=AM$+A$:NK=NK+1:IF NK<6 THEN 12
0
170 IF VAL(AM$)>65535 THEN 110
180 NK=0:PE$=" "
190 A$=GET$
200 IF VAL(A$)=0 AND A$<>"0" THEN 190
210 PRINTA$;:PE$=PE$+A$:NK=NK+1:IF NK<
2 THEN 190
220 PRINT'"OPENING BALANCE DEBIT OR C
REDIT (D/C) " ;
230 REPEAT:B$=GET$:UNTIL INSTR("CcDd",
B$)
240 PRINTB$
250 PRINT'"ALL OK? (Y/N)";
260 REPEAT:KEY$=GET$:UNTIL INSTR("YyNn
",KEY$)
270 IF INSTR("Nn",KEY$) THEN N=1:GOTO
50
280 U=HIMEM+7*(N-1)
290 IF B$="C" THEN ?U=N:U?1=0
300 IF B$="D" THEN ?U=0:U?1=N
310 U?2=VAL(LEFT$(D$,2)):U?3=VAL(RIGHT
$(D$,2))
320 U?4=(VAL(AM$) DIV 256)
330 U?5=VAL(AM$)-U?4*256
340 U?6=VAL(PE$)
350 IF N=9 THEN 390
360 PRINT'"MORE? (Y/N)"
370 REPEAT:KEY$=GET$:UNTIL INSTR("YyNn
",KEY$)
380 IF INSTR("Yy",KEY$) THEN N=N+1:CLS
:GOTO 90
390 PRINT:X=OPENOUT "NAMES"
400 PRINT#X,N:PRINT#X,N
410 FOR J=1 TO N:PRINT#X,N$(J):NEXT
420 CLOSE#X
430 *SAVE "AMOUNTS" 2950 298E
440 END
```

## The program (BUDGET)

```

10 REM ** BUDGET **
20 MODE 6
30 HIMEM=10576
40 DIM N$(80)
50 N=0:JJ=1
60 VDU 14
70 FOR J=1 TO 80:N$(J)=STRING$(8," ")
:NEXT
80 REPEAT
90 PRINT TAB(0,5)"Space Bar when ACCO
UNTS tape ready "
100 REPEAT:KEY$=GET$
110 UNTIL KEY$=" "
120 X=OPENIN "NAMES"
130 INPUT#X,N
140 INPUT#X,NR
150 FOR J=1 TO N:INPUT#X,N$(J):NEXT
160 CLOSE#X
170 *LOAD "AMOUNTS" 2950
180 VDU 19,0,4,0,0,0:VDU 19,1,7,0,0,0
190 CLS
200 PRINT TAB(14,2)"** BUDGET **"
210 PRINT TAB(17)"-----"
220 PRINT''TAB(7)"1) Display account"
230 PRINT''TAB(7)"2) Transaction"
240 IF N<2 THEN PRINT''TAB(3)"**";:ELS
E PRINT'';
250 PRINT TAB(7)"3) Open new account"
260 PRINT''TAB(7)"4) Finish"
270 PRINT''TAB(3)"<Select by pressing
numbered key>";
280 REPEAT
290 KEY$=GET$:A=VAL(KEY$)
300 UNTIL A>0 AND A<5
310 IF NR=2000 AND A=2 THEN CLS:PRINT'
'' "There is no more room":PROCdelay:GOT
O 180
320 IF N<2 AND A<>3 THEN 280
330 ON A GOTO 340,340,890,1480
340 CLS:@%=&00002:VDU 19,0,5,0,0,0:VDU
19,1,7,0,0,0
350 FOR J=1 TO 20
360 IF J>9 THEN JJ=0
370 PRINT TAB(JJ,J);J;N$(J);TAB(10,J)J

```

### 38 Practical Programs for the Electron

```
+20;N$(J+20);TAB(20,J);J+40;N$(J+40);TAB
(30,J)J+60;N$(J+60)
380 NEXT
390 IF A=2 THEN 1070
400 REPEAT
410 PRINT TAB(10,21)"Account no.":INP
UT "NA$
420 NA=VAL(NA$)
430 UNTIL NA>-1 AND NA<N+1
440 IF NA=0 THEN 890
450 REPEAT
460 INPUT TAB(0,22)"From which month (
1-12)"SM$;
470 SM=VAL(SM$):UNTIL SM>0 AND SM<13
480 REPEAT
490 INPUT TAB(0,23)"To which month (1-
12)"FM$;
500 FM=VAL(FM$):UNTIL FM>=SM AND FM<13
510 J=0:B=0
520 U=HIMEM+7*J
530 IF U?3=SM OR U?3>SM THEN 570
540 IF ?U=NA THEN B=B-U?4*256-U?5-U?6/
100
550 IF U?1=NA THEN B=B+U?4*256+U?5+U?6
/100
560 J=J+1:IF J<NR THEN 520
570 CLS:VDU 19,0,6,0,0,0:VDU 19,1,4,0,
0,0
580 PRINT'"Account name:"N$(NA)
590 @%=&20209:CD=0:DD=0
600 PRINT'"Date Date Debit
Credit"
610 IF B=0 THEN C=0:D=0:GOTO 660
620 IF B<0 THEN D=0:C=-B
630 IF B>0 THEN D=B:C=0
640 TR$="Balance B/D":CD=C:DD=D
650 PRINT"1/"SM$TAB(7)TR$,DD,CD
660 U=HIMEM+7*J
670 IF U?3=FM+1 OR U?3>FM+1 THEN 760
680 IF ?U<>NA AND U?1<>NA THEN 750
690 M=U?4*256+U?5+U?6/100
700 IF ?U=NA THEN C=C+M:CD=M:DD=0:TR$=
N$(U?1)
710 IF U?1=NA THEN CD=0:D=D+M:DD=M:TR$
=N$(?U)
720 IF ?U=0 OR U?1=0 THEN TR$="OPEN.BA
```

```

L. "
730 DA$=STR$(U?2)+"/"+STR$(U?3)
740 PRINTDA$TAB(7)TR$,DD,CD
750 J=J+1:IF J<NR THEN 660
760 B=C-D
770 IF B=0 THEN 810
780 IF B<0 THEN DD=0:CD=-B
790 IF B>0 THEN DD=B:CD=0
800 PRINT TAB(7)"Bal. C/D",DD,CD
810 PRINT TAB(19)"===== "
820 IF B=0 THEN DD=D:CD=C
830 IF B<0 THEN DD=D:CD=D
840 IF B>0 THEN DD=C:CD=C
850 PRINT TAB(18)DD,CD
860 PRINT''TAB(10)"<Space Bar for Menu
>"
870 REPEAT:KEY$=GET$:UNTIL KEY$=" "
880 @%=&10:GOTO 180
890 CLS:IF N=80 THEN PRINT'"No more a
ccounts can be opened":PROCdelay:GOTO 19
0
900 VDU 19,0,2,0,0,0:VDU 19 1,4,0,0,0
910 PRINT'"What is the name of the ne
w account?"
920 INPUT'N$
930 N$=N$+" " :N$=LEFT$(N$,7)
940 FF=0:FOR J=1 TO N
950 IF N$=N$(J) THEN FF=1
960 NEXT
970 IF FF=1 THEN PRINT'"An account in
this name is already open":PROCdelay:GOT
O 190
980 PRINT'"Account to be opened, name
d:"
990 PRINT'TAB(15)N$
1000 IF N>69 THEN PRINT'"Only ";80-N;"
more accounts allowed"
1010 PRINT'"OK (Y/N)"
1020 REPEAT
1030 KEY$=GET$
1040 UNTIL INSTR("YNyn",KEY$)
1050 IF KEY$="Y" OR KEY$="y" THEN N=N+1
:N$(N)=N$:GOTO 180
1060 GOTO 180
1070 REPEAT
1080 INPUT TAB(0,21)"Credited a/c no.

```

#### 40 Practical Programs for the Electron

```
"NC$
1090 NC=VAL(NC$):UNTIL NC>-1 AND NC<N+1
1100 IF NC=0 THEN 890
1110 REPEAT
1120 INPUT TAB(0,22)"Debited a/c no.  "
ND$
1130 ND=VAL(ND$):UNTIL ND>-1 AND ND<N+1
AND ND<>NC
1140 IF ND=0 THEN 890
1150 REPEAT
1160 INPUT TAB(0,23)"Date (DDMM)  "D$
1170 DA$=LEFT$(D$,2):MO$=RIGHT$(D$,2)
1180 DA=VAL(DA$):MO=VAL(MO$)
1190 UNTIL LEN(D$)=4 AND DA>0 AND DA<32
AND MO>0 AND MO<13
1200 NK=0:AM$="":PRINT TAB(0,24)"Amount
(POUNDS,PENCE)?  ";
1210 A$=GET$
1220 IF VAL(A$)=0 AND A$<>"0" AND A$<>"
." THEN 1210
1230 IF NK=5 AND A$<>"." THEN 1210
1240 PRINT A$;:IF A$="." THEN 1260
1250 AM$=AM$+A$:NK=NK+1:IF NK<6 THEN 12
10
1260 IF VAL(AM$)>65535 THEN 1200
1270 NK=0:PE$=" "
1280 A$=GET$
1290 IF VAL(A$)=0 AND A$<>"0" THEN 1280
1300 PRINT A$;:PE$=PE$+A$:NK=NK+1:IF NK
<2 THEN 1280
1310 AM=VAL(AM$):PE=VAL(PE$)
1320 CLS:VDU 19,0,6,0,0,0:VDU 19,1,1,0,
0,0
1330 PRINT'"The transaction to be ente
red is:"
1340 PRINT'"Credit account:  "N$(NC)
1350 PRINT'"Debit account:  "N$(ND)
1360 PRINT'"Date:  "D$
1370 PRINT'"Amount:  "AM$"
."PE$
1380 PRINT'"'"Is this correct? (Y/N)
";
1390 REPEAT
1400 KEY$=GET$
1410 UNTIL INSTR("YyNn",KEY$)
1420 IF KEY$="N" OR KEY$="n" THEN 340
```

```

1430 U=HIMEM+7*NR:NR=NR+1
1440 ?U=NC:U?1=ND:U?2=DA:U?3=MO
1450 U?4=AM DIV 256:U?5=AM-256*U?4:U?6=
PE
1460 IF NR>1990 THEN PRINT'"Only "2000-
NR"transactions allowed":PROCdelay
1470 GOTO 180
1480 CLS:VDU 20
1490 PRINT''
1500 X=OPENOUT "NAMES"
1510 PRINT#X,N:PRINT#X,NR
1520 FOR J=1 TO N:PRINT#X,N$(J):NEXT
1530 CLOSE#X
1540 *SAVE "AMOUNTS" 2950 5FFF
1550 PRINT"Backup? (Y/N) ";
1560 REPEAT:KEY$=GET$:UNTIL INSTR("YyNn
",KEY$)
1570 IF KEY$="Y" OR KEY$="y" THEN 1500
1580 VDU 15
1590 END
1600 DEF PROCdelay
1610 TIME=0
1620 REPEAT:UNTIL TIME>800
1630 ENDPROC

```

## Variations

These programs work equally well with a disk system. No modifications are required, except that in BUDGET, the message on line 90 may be omitted, together with lines 100 and 110.

It would obviously be an advantage to be able to obtain a hard copy of accounts from time to time. If you have a printer attached to your Electron, consult its handbook to find out how to initiate output to it. Output should be initiated at line 580 by inserting a suitable statement at the beginning of that line. Output should be disabled at the end of line 850.