CALCULATOR CORNER

This month Dick Pountain has handed over to reader N. Horwood who relates his experience of using the TI59 with printer for accountancy; the moral being that the calculator can often provide a cost effective alternative to the micro if you make a realistic *analysis of the data requirements of your application.*

NEW COURSES FOR TEXAN HORSES

I was in two minds when I bought the TI59 some two or three years ago, knowing nothing about programming, but having been involved in a user capacity in industry. Initially I tried to develop programs without the printer, but, except for very short and uncomplicated programs, I found editing and debugging extremely frustrating and time consuming. It wasn't long before I was forced to buy the print cradle which transformed the whole activity of program development. Even so, not being a recently trained mathematician but merely a middle aged Engineer, it took several months of all my spare time to become reasonably fluent in the Texas dialect. I have nothing but praise for the great assistance, patiently given to me by the TI people at Bedford, usually with me on the phone in yards of paper tape, having forgotten to read the excellent manual properly!

Once having gained confidence both in my ability to write useful programs and in the reliability of the machine mechanically and electronically, it occured to me that here was a very powerful device. Admittedly the storage capacity for bulk data was limited, but the logic and arithmetic capacity was of a very high order. I experimented with some simple applications in industrial costing and accountancy and found that by choosing the type of program structure which gave user prompting on the printer (which doubled as result annotation and dealt with the general form of the data processing so that a program was as flexible as possible — thus minimising the need to change magnetic cards for slightly altered requirements) it was possible to use this machine in a commercial context.

There are drawbacks. For example the 20 column printout is a little narrow for traditional layout, and while the printing rate of about two lines per second free running is acceptable, the data transfer rate round logic loops and into the four print registers is slow, and complex programs can be frustrating. There are ways of speeding up loops by using direct addressing, but this is only to be recommended after you are certain that no further changes will be required to the program and where there is plenty of spare space. Another trick is to use any spare data registers for print code instruction storage, thus releasing program space for logic instructions. There's no doubt that commercial type programs use enormous memory space for text, but in spite of these limitations I have been able to apply this machine to a wide variety of office and factory applications.

One of the first of these was my own (self-employed) accounts, incorporating all the requirements for VAT and the Inland Revenue, apportionments etc. Of course the program had to be cleared by the local VAT office, but, believe it or not, they were most helpful. Leading on from this I developed programs for the most used Retailers VAT special schemes, also cleared by the local friendly VAT man for general use. The ability of this machine to provide printed prompts in this type of application is invaluable since the routine is not in daily use, and although the users crib sheet sets out the procedure clearly enough, it's not the same as the point of use prompt. In fact I have found the printed prompt/title easier to use and less strain than most VDUs. It also provides the permanent record of the transaction. Okay, it's long and narrow, but it can easily be folded concertina-wise, and for less than £300 complete (including VAT) it must be good value.

I am not an accountant, although for my sins I have much to do with that fraternity. One of the mysteries associated with the running of any business is the manipulation of the simple figures derived from the purchase and sale of various items or services. The end result has to be a set of accounts. One of the beauties of the "59" is that data as well as program steps can be stored on magnetic cards. This means that data can be input and processed and stored in RAM and then transferred to magnetic card, just like tape or disc. The data may then be saved for future addition or modification. This is possible due to the four way split of the data banks, with each bank independent of the others.

Admittedly relatively small volumes of data can be stored at a time, but even so there is enough to allow a full set of analysed company accounts. The technique is to print and annotate automatically every cash book entry, giving a clear audit trail and hard copy of the input. This can be organised to segregate Capital and Revenue and Dr and Cr in batches of register locations. Program No. 1 carries out the trial balance with full annotation; with the data still intact a second program is overlaid from another magnetic card to produce the Profit and Loss A/C in more or less standard format, and yet another program is overlaid on this data to give the Balance sheet. For presentation purposes a matching set of descriptive headings may be preserved on another magnetic card to suit the user's needs. The whole set can be put together and Xeroxed for filing etc. It looks almost traditional!

One of the side benefits of the wider use of the TI59 has been the development of some "utility" programs which might be of general use. One is what I have called "Auto-Folio". This enables any value to be keyed in, along with its destination register number. Upon entry the value is loaded into that register and summed to what is already there, the value printed together with, on the same line and at the far right of the tape, the register, or folio number. This provides a much more compact listing than the Trace mode which prints everything twice, once before the destination and function code and again after.

The memory capacity allows up to 99 registers to be so addressed, 01 to 99. At the end of a batch of entries a listing routine prints an annotated statement of every register and clears all registers ready for a new batch. It takes about three seconds including printing for each entry to be completed. About the right tempo for the small keyboard and for the type of data analysis commonly required in say a Work Study office or laboratory. This program is shown in detail as it gives the encoding of numerical characters for loading into the No.4 print register. The Nop statements are left in the program in case it is required to speed up the action slightly by converting to direct addressing without having to change the program location numbers.

Another useful method of auto-routing data is to incorporate within a number a two digit code, or a series of two digit codes which may be decoded after splitting out of the original number. The example shown uses standard 0p. codes, and can split a number of the form abcdefgh into four pairs, i.e. ab; cd; ef; gh, stored in four data registers ready for use as indirect addresses. There is a more complicated approach available, using the pseudo-Op. code HIR. This utilises the eight "pending result" registers, not normally available to the user except by inference when using nested parentheses. The great advantage in using them in the Code Splitter and similar programs is that precious data registers are not sacrificed in the splitting up process, and may be utilised for the safe storage of working data. However, be warned! There are many pitfalls in using these pseudo-ops., which is probably why they have not been included in the standard manual; however, used with care they increase the capability of this great little machine even further.

Another word of warning. If this machine, or similar magnetic card readers, are to be used more generally in industry, commerce, and schools (as their price level warrants) care must also be taken to ensure that the card reading mechanisms of different machines which have to share families of magnetic cards are matched. I have occasionally needed to adjust the card feed speed of a satellite machine to match the one which originally wrote the program on to the card. This is not

Continued on P. 127

Code Splitter

```
000  76 LBL
001  11  A
002  99 PRT
003  42 STO
004  04  04
005  59 INT
006  48 EXC
007  04  04
008  22 INV
009  59 INT
010  65  ×
011  01  1
012  00  0
013  00  0
014  95  =
015  48 EXC
016  04  04
017  55  ÷
018  01  1
019  00  0
020  00  0
021  95  =
022  42 STO
023  03  03
024  59 INT
025  48 EXC
026  03  03
027  22 INV
028  59 INT
029  65  ×
030  01  1
031  00  0
032  00  0
033  95  =
034  48 EXC
035  03  03
036  55  ÷
037  01  1
038  00  0
039  00  0
040  95  =
041  42 STO
042  02  02
043  22 INV
044  59 INT
045  65  ×
046  01  1
047  00  0
048  00  0
049  95  =
050  48 EXC
051  02  02
052  59 INT
053  42 STO
054  01  01
055  99 PRT
056  43 RCL
057  02  02
058  99 PRT
059  43 RCL
060  03  03
061  99 PRT
062  43 RCL
063  04  04
064  99 PRT
065  98 ADV
066  98 ADV
067  98 ADV
068  91 R/S
069  00  0
```

Press A to enter data.

Press B to list and clear

Sample Output

```
  123456.78
        12.
        34.
        56.
        78.
      4.05
        0.
        0.
        4.
        5.
```

AUTO FOLIO

```
000  76 LBL
001  23 LNX
002  65  ×
003  01  1
004  00  0
005  00  0
006  95  =
007  82 HIR
008  32  32
009  82 HIR
010  12  12
011  69 OP
012  04  04
013  22 INV
014  86 STF
015  01  01
016  82 HIR
017  15  15
018  69 OP
019  06  06
020  74 SM*
021  00  00
022  91 R/S
023  76 LBL
024  25 CLR
025  85  +
026  03  3
027  95  =
028  87 IFF
029  01  01
030  23 LNX
031  68 NOP
032  82 HIR
033  02  02
034  76 LBL
035  34 FX
036  86 STF
037  01  01
038  82 HIR
039  14  14
040  59 INT
041  61 GTO
042  24 CE
043  68 NOP
044  76 LBL
045  11  A
046  82 HIR
047  05  05
048  91 R/S
049  42 STO
050  00  00
051  55  ÷
052  01  1
053  00  0
054  95  =
055  82 HIR
056  04  04
057  22 INV
058  59 INT
059  65  ×
060  01  1
061  00  0
062  95  =
063  76 LBL
064  24 CE
065  82 HIR
066  03  03
067  07  7
068  32 X:T
069  82 HIR
070  13  13
071  77 GE
072  25 CLR
073  68 NOP
074  85  +
075  01  1
076  95  =
077  87 IFF
078  01  01
079  23 LNX
080  68 NOP
081  82 HIR
082  02  02
083  61 GTO
084  34 FX
085  68 NOP
086  76 LBL
087  12  B
088  98 ADV-
089  98 ADV⁻
090  01  1
091  22 INV
092  90 LST
093  98 ADV
094  98 ADV
095  98 ADV
096  47 CMS
097  91 R/S
098  00  0
```

Press A to enter data.

Press R/S to enter address

Calculator Corner cont. from p.107

too difficult when you know how, but it should not be necessary. Texas Instruments please note.

In spite of the minor problems, and no machine is perfect, I hope that I have been able to demonstrate the extra scope available with the application of some imagination to the use of these inexpensive but powerful calculators.

Postscript

Mr Horwood has expressed his willingness to answer any further enquiries about these applications. You can write to him at: 11 Melville Hall, Holly Road, Edgbaston, Birmingham B16 9NJ (and not to PCW please!).

Here, to stop you all pestering me, is the answer to the problem which Rolf Howarth set in his TI59 Pseudo Op codes article:-

It is possible to examine the internal routines which perform the P/R, decimal/sexagesimal and statistic operations by keying in the following:

10 Op 17 (6 Op 17 for the TI58)
99 STO 00
Pgm 01 A
CLR D.MS

Then press LRN and examine the program using SST. Note that on pressing BST, Del or Ins they do not behave normally, and may crash the calculator. The program is 380 steps long, after which garbage is found — which must presumably have some purpose. Any guesses?

The routines start at the following steps:

Op 12	000
Op 15	047
Op 14	058
x	067
Op 11	084
INV x	107
Op 13	149
Σ+	192
INV Σ+	213
INV P/R	250
P/R	284
D.MS	303
INV D.MS	341