

Taking charge of your DRAGON

PCN
micropaedia
PULL OUT AND KEEP

TANDY ADD-ONS

Besides having a common processor heritage, the Dragon and the Tandy Color Computer can also share lots of peripherals and software.

Although there can be difficulty in loading Tandy cassette software on the Dragon, cartridge software will usually run without incident. Among the games in this cartridge software range are Chess, Super Bustout (presumably a version of the traditional Breakout game), Checkers, Bridge, adventure and lots of space games.

Serious software

Tandy also handles more serious cartridges to run on the Dragon/Color Computer including word processors and spreadsheets. You should be warned, however, that a thorough test of the Tandy cartridge you want run on the Dragon is pretty much a prerequisite.

There is one Tandy word-processing cartridge that apparently works properly with the Dragon in every respect — except when it comes to trying to print out the text you've written. So make sure the program you want to use will work in the way you think it will work.

Peripherals are a much easier proposition. Most Tandy printers will work without modification on the Dragon — giving Dragon owners access to everything from the little CGP115 four-pen colour printer/plotter right up to Tandy's big Daisywheel II machine.

Print and plotters

The plotter rolls in at the price of £149 and can produce both graphics and highly readable text. The plotter moves at a speed of 12 characters per second, so it's no great speedster compared to common dot-matrix printers.

You'll have to sacrifice paper width — the CGP115 uses non-standard 4½in wide paper — for the extra colour and plotting facilities of the device. But at almost half the cost of most dot-matrix printers, the width shouldn't bother you too much.

The text size is programmable in 63 increments from 80 characters to one per line, and even the smaller of those increments is a good deal more readable than most low cost printer output.

The CGP115 can, of course, be controlled to draw colours as well — with the length of a line being programmable in 480 steps, and height in 999 steps.

Tandy also offers a flat-bed plotter for use with Dragon, but the price on that is a bit steeper — about

three times the price of the machine itself. Selling at £699, the Tandy FP215 Flatbed Plotter/Printer can use four colour pens — red, blue, green and black — and will draw text or graphics horizontally or vertically using the Basic LPRINT command. The speed is about half that of the CGP115 and features a step size of one millimetre.

Tablets and books

Perhaps more interesting is Tandy's graphics input tablet. Again it's not cheap — priced at £299 — but the GT-115 X-PAD boasts simple computer-aided-design facilities which it is claimed are ideal for architectural design, business and other graphics applications. The bottom row of the pad lets you specify the colours you want to use for drawing, while the top row features a list of keywords that can be accessed merely by pointing at them on the tablet.

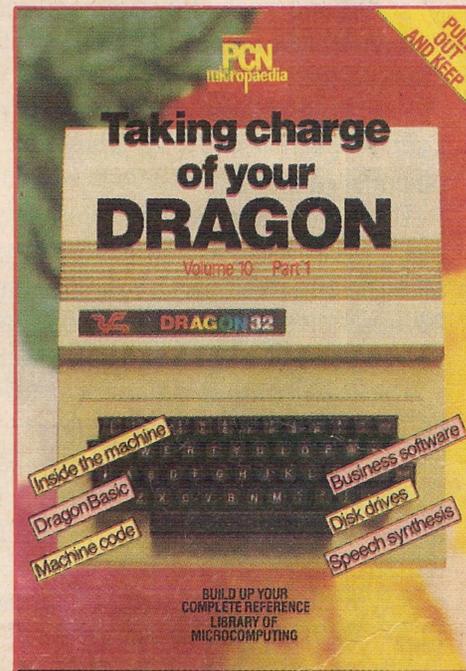
The software to run the device comes on a cartridge and requires Extended Basic to operate, although it's not quite clear whether the extended Basic that comes with the Dragon is extended enough to run the X-PAD with no problems. Tandy says it should work, although this again would be something you should try with your system first before buying.

One Tandy peripheral you might not think of first off is the Tandy Color Computer range of books. Tandy's manuals for the Color Computer come in two books, which cover a greater range than the equivalent tomes for the Dragon. And since they use largely the same Basic, all the principles you learn from a Tandy manual will apply on the Dragon.

Of course, other publishers produce books on the Tandy Color Computer and again many of them will also apply to the Dragon. You'll find a large supply of such books in the U.S., where the Dragon until recently was unheard of and the Color Computer has long been a strong seller.

And of course, the Tandy joysticks can be used with the Dragon — they sell for £9.95 and can be purchased from most Tandy computer centres.

PART 2



VOLUME 10

Below is the Tandy Color Computer and many of its peripherals. Most of these peripherals can be used with the Dragon, including the Color Computer joysticks, many of the games cartridges, as well as many Tandy printers and plotters.

Unlike Dragon add-ons, however, Tandy products can only be purchased from Tandy dealers at either a Tandy Computer Centre or an ordinary Tandy outlet. Although some smaller Tandy outlets may not stock all the computer components, they can be ordered from any Tandy store.



THE SOUND OF YOUR DRAGON

When it comes to sound, the Dragon is a little short of breath. All the bits and pieces which generate sound are in the right places, but the end result can be disappointing — the electronic Dragon produces a croak rather than a roar.

Part of the blame can be laid on documentation — the section on sound in the manual is limited — but the loudspeaker system is obviously a bit naff.

There are two commands which produce sound. Sound X,Y outputs a beep, where X is the pitch and can be chosen between 1 and 255, with 89 sitting at about middle C. Y defines the duration of the beep — again you have a choice between 1 and 255. Each number represents a 16th of a second, so a Y of 32 should give you a hearty two-second beep.

The PLAY command allows you to be a little more creative, but is more complicated to implement. PLAY is followed by a string of commands.

You can specify the note as a normal musical expression, from A to G. O sets the octave, which is expressed by a number between 1 and 5. L, followed by a number 1 to 255, sets note-length, T, with 1 to 255, sets tempo, and V, with a choice of 0 to 31, sets the volume.

A typical PLAY statement might be:
PLAY 'V20;T200;L150;01;AGFEDCE'

Sharps can be played by using the # symbol following the note letter. A minus (-) generates a flat.

One of the limitations of Basic-generated sound is the intrusion of the computer's interrupt routine. The interrupt is a special program executed 50 times a second, and is used to control the Timer function on the Dragon. Normally, the interrupt goes completely unnoticed by the user, but during sound generation it can cause a 'rattling' interference, especially on the high-frequency notes.

It is possible to POKE to disable the function, but it is re-enabled each time the computer meets a SOUND or PLAY statement.

The PLAY command is capable of generating the pedestrian type of tune — *Three Blind Mice* and so on — but the speed of the Basic tends to become a limitation when anything more complicated is attempted. You and your Dragon won't be able to join The Human League, but between the two of you you should be able to generate some good effects to punctuate your games programs.

Even so there are still limitations, especially if you are trying to generate a fairly fast or complicated graphics game. While sound is being used the computer halts everything else to get on with the job — but there is another way.

You can use pre-recorded sound on tape. This has the advantage of going through the television sound system instead of the computer's. The sound can be recorded on the tape just like a program, and then played back at the appropriate moment during the game.

You have to use the AUDIO command, which connects the cassette recorder to the TV speaker, and AUDIO OFF, which disconnects it. MOTORON and MOTOROFF control the cassette motor — timing here is obviously important. When you use MOTORON there is a delay after the relay has switched — this is for the Basic when saving and loading programs, and can be cancelled by POKE 149,0;POKE 150,1.

Remember that you can't mix cassette and Basic sound because if the computer comes across SOUND or PLAY it executes AUDIO OFF.

To get even more flexibility there is always the possibility of going straight to the heart of the matter via machine code. Using machine code is faster and you avoid the 'interrupt' problem mentioned earlier.

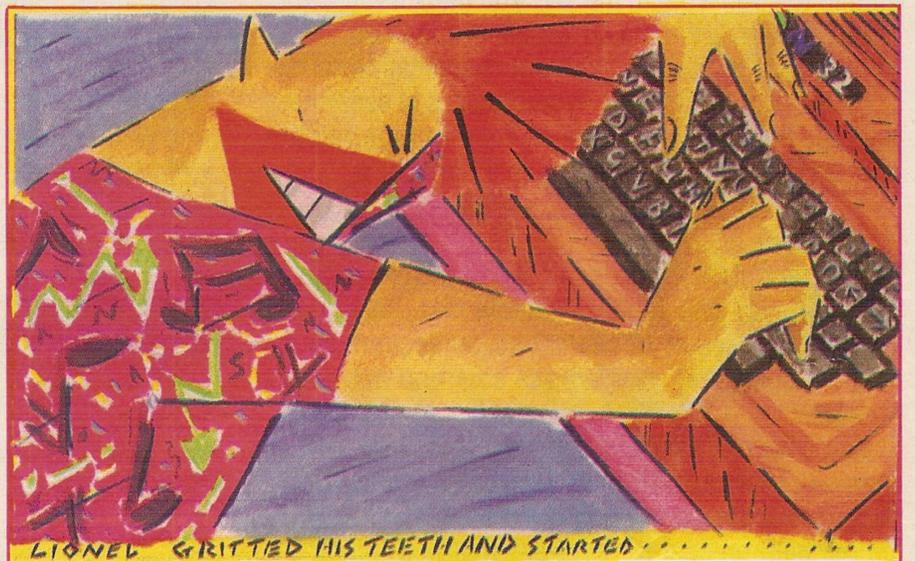
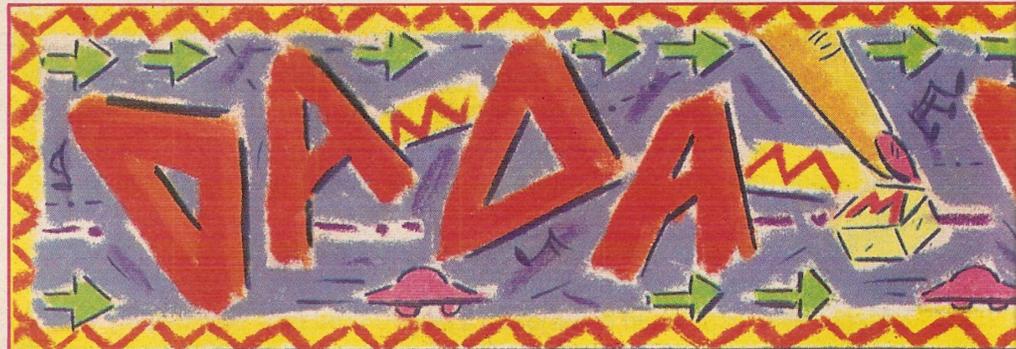
The sound chip is called an 'analog multiplexer' and it controls which sound source is selected from the options 6 BIT (sound/play) CASSETTE (audio on/off) or CARTRIDGE (sound chip in cartridge).

There are three relevant connections to this chip addressed to bit 3 of addresses 65281, 65283, and 65315 respectively, and the table below shows the arrangement for any particular source. The most interesting bit is 6 BIT. For this option, the sound comes from D/A converter (the same D/A converter used for the joysticks).

This is addressed as the six most significant bits of location 65312. With the multiplexer set up for 6 BIT, storing a number at location 65312 produces a click from the TV speaker — the bigger the number the higher the click.

As you would expect there are also packages — both hardware and software — which enable you to generate even more sound if you're really keen. The way the Dragon has been designed means there is great potential for extra sound modules for the computer's games port.

SEL 1	SEL	BDA.	Sound source selected
0	0	1	6 BIT D/A
1	0	1	CASSETTE
0	1	0	CARTRIDGE (ext. chip)
0	0	0	SILENT



So far there is a product from JCB Microsystems, 29 Southbourne Road, Bournemouth BH6 5AE. JCB's Sound Extension Module does just as it says, putting the user in touch with a pre-stored set of sound effects likely to be of use during typical games. Sounds like pistol shots, car horns, explosions and so on are on tap.

With this product you also have the ability to generate your own sounds or musical ditties more easily than with the Dragon's native facilities. The effects can be called into your own programs, and the beauty of it is that it doesn't down the Dragon's processor — it has its

own. Neither does it eat into the user memory.

Which brings us neatly to software packages. This approach is obviously cheaper. The products give you tools to grapple with the Dragon's own facilities more successfully.

Computavoice is Dragon Data's own offering. The machine-language program takes up about 8K, and is loaded via CLOADM into protected RAM above Basic. It uses phonetic codes to build up words and phrases, rather than draw from a built-in set of words. The letters A to Z and numbers 1 to 9 are used as symbols for the phonemes, and other characters alter the stress and insert the punctuation and pauses.

The quality of this product is high and it represents a good buy at £7.95.

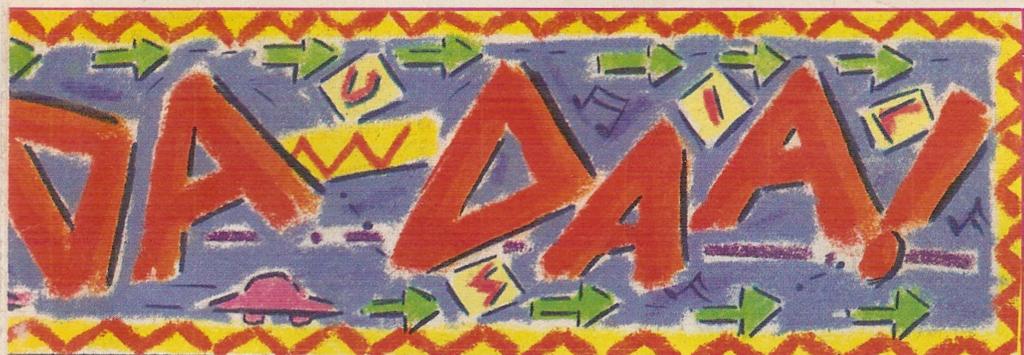
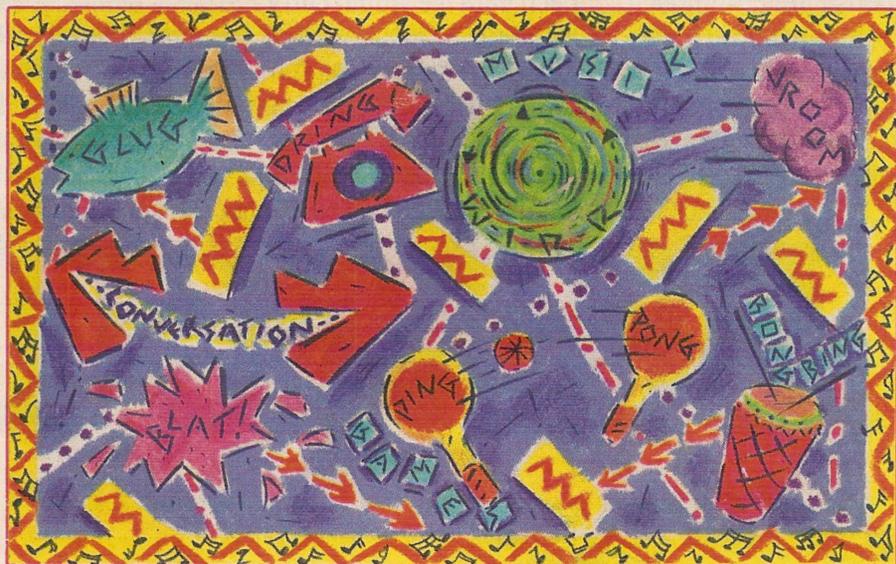
If £7.95 is too much for you, there is an even cheaper software product called Synth at £3.95. This uses a mixture of Basic and machine language and enables you to create sound via joysticks. Instructions tell you how to include the created sound effect within your own programs.

Synth is a fun program and should be considered more a conversation

seven octave range. You can vary the tempo or change the musical keys within the program. The whole thing is driven by an easy-to-understand menu, and the music can be saved as an independent subroutine and can be loaded and executed from other Basic programs.

The expanded Dragon (unavailable at the time of writing) will feature an RS232 interface, encouragement for more third-party add-on merchants to produce sophisticated sound peripherals.

The recently announced disk drives for the Dragon should allow some of the more sophisticated products produced for the similarly designed Tandy Colour Computer to be moved over to the Dragon. These could include The Voice and stereo sound hardware, as well as the 'Talk to me' speech recognition system.



piece or party-stopper than something you'd want to spend long hours learning to master.

It's available from C Woods, 37 Marlpit Lane, Sutton Coldfield, West Midlands.

The Micro Music Map is something else again. It converts standard sheet music into letters for the Dragon's play command. It requires no real musical knowledge to use and costs £3.50 from Cotswold Computers, Park Hill, Hook Norton, Oxfordshire.

The Dragon Selection 2 is from Dragon Data. This allows you to play back your own melodies — there are some nice graphics to go with it as a bonus, and it's also written in an unprotected Basic, so you can list and examine to get a few ideas of your own. Selection 2 costs £7.95 from Dragon dealers.

Some of the games available for the Dragon also make use of the micro's sound facilities. Android Attack from Microdeal software involves speech synthesis. At the beginning of each game the computer announces 'Intruder Alert' and other outbursts like 'Coward' are scattered throughout the game. This adds a nice touch to what is already an excellent game. It's available from Boots or direct from Microdeal (0726) 3456.

Also from Microdeal is the Dragon Composer. This is another tape program, but it's a pretty sophisticated offering and retails at £15.

It allows you to compose your own music and play it back through the television speaker. It gives you the ability to display your own graphics and words on the screen as your masterpiece plays itself through. The Dragon Composer uses four voices (separate channels of sound) over a

MORSES FOR COURSES

The Morse code signal language is probably the simplest use of the Dragon's SOUND command that you could imagine.

Below is a simple five line program to allow you to generate Morse code messages on your Dragon. It demonstrates both the syntax and use of SOUND as well as how the ELSE operation works in Dragon's Microsoft Basic.

The listing runs as follows:

```
5 A=0
10 A$=INKEY$: IF A$<>" " THEN A=ACS(A$)
20 IF A>64 AND A<94 THEN SOUND 160,6 ELSE GOTO 30
30 IF A>49 AND A<57 THEN SOUND 160,2 ELSE GOTO 40
40 A=0:GOTO 10
```

When entered and RUN, the program will generate long tones (or dashes) if you press any of the upper case alpha keys (A to Z) and short tones (or dots) if you press any of the numeric keys (1 to 9).

So to generate the traditional SOS message: dot dot dot dash dash dash dot dot dot, you would press a numeric key three times, then an alpha key three times and then a numeric key again three times.

The morse code table below will give you an introduction to Morse code, which can be practiced using the demonstration program.

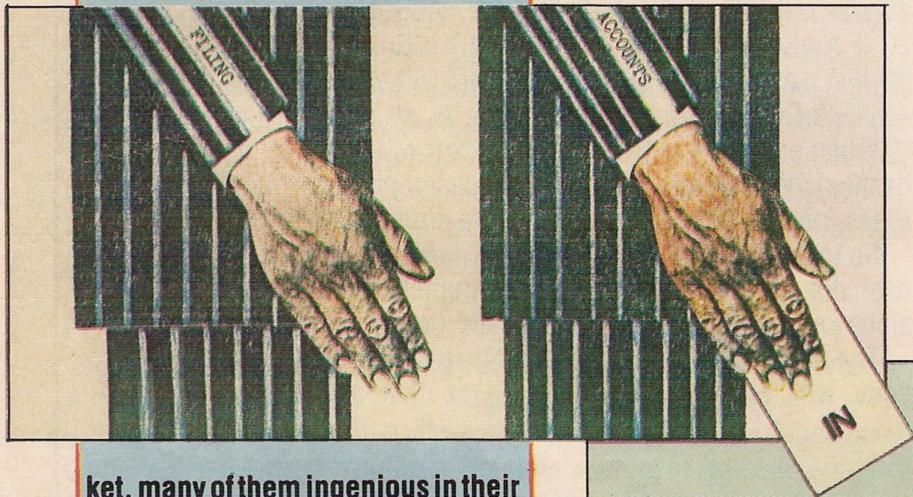
A . —	S
B —	T —
C —	U . . —
D — . .	V . . . —
E .	W . — —
F . . — .	X — . . . —
G — — .	Y — . . . —
H	Z —
I . .	1 . —
J . — — —	2 . . —
K — . —	3 —
L . — . .	4 —
M — —	5
N — .	6 —
O — — —	7 —
P . — . .	8 — —
Q — . . . —	9 — —
R . — .	0 — —

WP MINUS THE FRILLS

A dedicated word processor with a letter-quality printer can cost upwards of £2,000. The Dragon can provide many of the facilities of such a machine for about £500 — providing you are prepared to make compromises.

The screen size of the Dragon, 32 columns by 16, makes the operation of a word processor problematic. It means that only about 13 lines of text can be displayed on screen at any one time, which makes it difficult to format and edit without much scrolling of the text. The lack of a genuine upper and lower case facility contributes to the problems with the Dragon.

There are a number of word-processing programs on the mar-



ket, many of them ingenious in their manipulation of the text. Most of them have the usual facilities — formatting, copy, delete and search — but they suffer from the limitations of the Dragon screen size and display.

The one that doesn't is Microdeal's Telewriter word-processing program. With skilful use of high resolution graphics the program creates a new character set. A genuine 51 column by 25 display with upper and lower case letters is achieved.

The entire thing is worked and controlled by software — there are no hardware modifications to be done.

Telewriter allows the user to input text. About 18,000 bytes, which is about 3,000 words or eight A4 pages, are available for use after the program is loaded.

One can format the text before printing to any configuration, and left, right upper and lower margins can be defined by using the software.

Characters per line and lines per page are also software controlled. And embedded direct format commands can be hidden in the text to give the user control over formatting changes during printing.

To suggest that the Dragon can be used as a business machine causes raised eyebrows in some circles. Even Dragon Data only four months ago was saying that the machine wasn't designed as a business machine, and wasn't recommended for that purpose. But imminent arrival of the long-awaited Dragon disk system should put paid to this.

Nevertheless in the 12 months since its launch many people have tried to use the machine in business, and at least two software houses have started to specialise in Dragon business application software. They make use of the Dragon's typewriter style keyboard and ability to link up with most Centronics type printers.

Although it has relatively large RAM memory at 32K, the machine's keyboard does not respond well to touch-typing — if you try to type too fast there is a tendency to lose letters, and the fuller the machine's memory gets, the slower the response time. Another less obvious drawback in the use of the Dragon at high speed is its slow clock time — less than 1Mhz. For business use, this means that lengthy calculations such as sorting or filing can take some considerable time.

To assess the importance of these drawbacks,



DOWN TO

With only 32K of memory and a screen display of 32 by 16, the Dragon does present problems for business applications, but the

consider what you want the computer to do for you. Keep your accounts? Keep the tax man and VAT man happy? Organise your mailing list and keep track of stock? All of these amount to a desire to save yourself some time — those endless nights and weekends spent huddled over the books add up.

The essential tools are a Dragon 32K computer, a cassette machine, TV set and a reasonably good printer — and of course, unless you have the knowledge or time to write it yourself, good application software.

Over the last few months the amount of business software has increased quite considerably and two firms, MST and Abacus, have been pushing their business application programs heavily.

With only 32K of memory and a screen display of 32 by 16, the Dragon does present problems for business applications, but the software available overcomes much of that by clever writing and constant menus. Because cassettes must currently be used for storage, loading and saving is time consuming. But the machine is still powerful enough to allow a small businessman to achieve something approaching an accounting system which will certainly find favour with an accountant.

Once you have the software you must start transferring your data to it. This is a problem with any machine, and the Dragon is no more complicated than others.

The advice of your accountant would probably be useful before you embark on the lengthy and time consuming task of entering the initial data. The software programs dealt with here are well explained and user friendly, and should serve as a good introduction to using the Dragon in business.

Stock control

The stock control programs available will handle about 200 stock items on one data file. This means that if your company has more than that, you'll need to employ more than one file — therefore careful management of the filing system is essential if you are using a cassette machine to store the data files.

Each data file would carry the basic details of each stock item, a description of the item, the supplier, number in stock, minimum stock level, maximum level and buying-in and selling prices. With the program it should be possible to check your stock level instantly, together with the total value of stock in hand, for audit purposes. The Dragon is quite capable of

doing this, and the MST and Abacus versions allow the information to be printed out for a permanent record.

Invoice/statement

Printed invoices are always impressive. MST's invoice/statement program allows the initial information — your own details, company name, address and VAT number — to be loaded onto a data cassette first. This information is then automatically printed out on the top of an invoice each time you send one out.

The customer's name and address is then entered and that too can be saved for future use. Details of the transaction are then inputted, and the computer will compute any VAT additions, charges for carriage and any discounts allowed. The result is a very professionally produced invoice or statement and much time saved.

The data cassette will only record the customer's name and address, since one of the limitations of the Dragon's memory is that to try and keep a trading record within that program would severely limit the number of invoices and customers that one can use within one run.

This means that it is essential to use a program such as this with a business accounts program.

Business accounts

A good business accounts program is the most powerful tool for the small businessman. The Dragon has the capability to handle this, and the programs we looked at all did basically the same thing. Like all program applications none of it works unless the information is accurate. There is little the computer can do if mistakes are made on input, but our test programs did call up error messages if obviously erroneous data was being put in.

Both the programs for the Dragon came with good instructions, but before use it was essential to read them carefully. Once you have decided how you are going to operate a program of this kind you must keep to a pattern. Neither program was difficult to operate and both were menu operated on screen. If one keeps the data up to date it is possible to keep a regular check on your business position.

List of creditors and debtors are immediately available, as is a printout of your current trading position. Again the limitations of the Dragon memory mean that only about 160 transactions can be kept on file at any one time, so you may have to employ more than one data file, saving each one onto cassette as it is

spreadsheet or calculator programs around, but one that works very successfully is a cartridge made for the Tandy Color machine. Called Speculator, the program comes with a well written, easily understood instruction book.

Using a cartridge leaves more memory available for large and complicated spreadsheets, but the operating speed of the Dragon at 1Mhz does mean that the calculations can take a very long time, although it would still take many hours to do the same thing with a pocket calculator. Again the screen display size makes a spreadsheet difficult to use, as only a maximum of 13 rows and four columns can be seen on screen at any one time.

Payroll

At present there are few payroll programs around for the Dragon. One from Abacus will produce an itemised payslip for up to 50 employees, with tax and NI deductions built in. The program takes just over 20 minutes to run with a full 50 employees, once the data had been loaded from tape and any changes have been made. At the end of each run the data tape has to be resaved and the information is updated for next week's run.

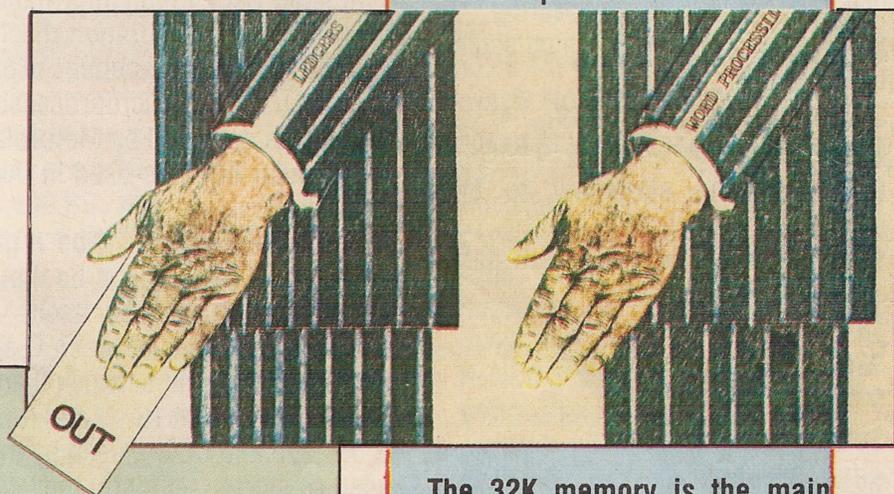
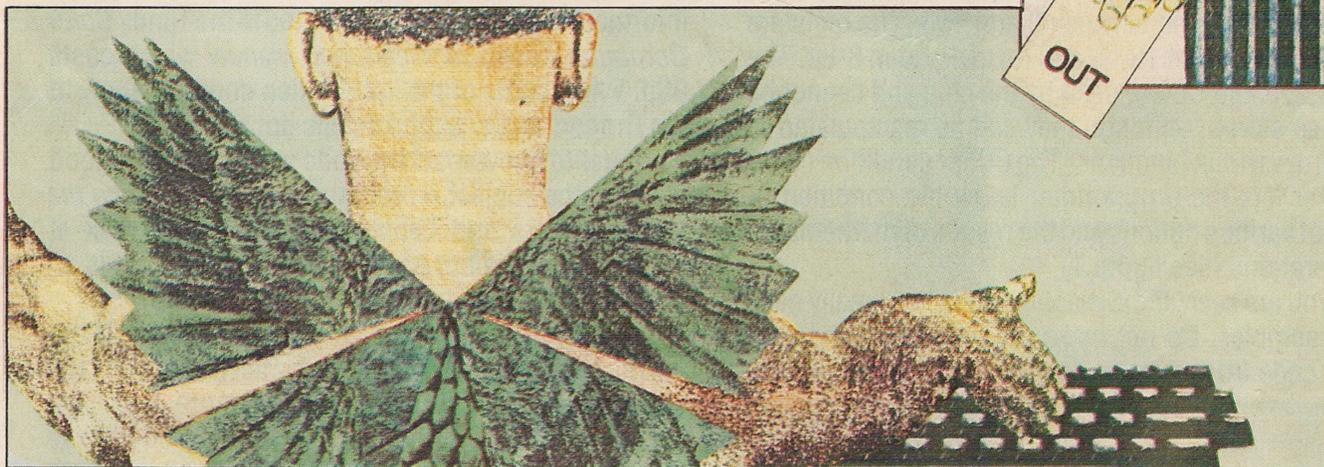
DATABASE DRAGON

Once the new user has finished playing with the computer for games it's a question of looking around for other uses of the machine. Most people will say that they would look at the machine for keeping records or a database.

There are already quite a large number of database programs available for the Dragon. Most of them are basically the same, allowing the user to input names, addresses, telephone numbers and any other information that may be of use. Information has to be available for fast access and the facility to select the type of information required is essential. A handy addition is the ability to dump the data onto a printer.

BUSINESS

play of 32 × 16, the Dragon does present software available overcomes much of that.



filled. Each month it would be necessary to bring forward the data onto fresh data files. But although the use of disks won't extend the machine memory, it will mean that each month you would be able to open another file on the same disk and be able to access the information very much quicker.

Mailing lists

Once you've balanced your books it will then be necessary to mail out your invoices, statements or even cheques. For any business application a good mailing list is essential. Both the ones we tried were comprehensive, user friendly and efficient to use. One of them had a particularly useful function in that you were asked to categorise your entries, so that if you wished to place your suppliers in one category, your customers in another and even your private files in yet another it made for efficient sorting and retrieval. The other program gave you the opportunity to rewrite one of the program lines so that you could line-up your printer to be able to print labels accurately.

Spreadsheet

Spreadsheets are becoming an increasingly useful tool for the business user. There are a number of

The program keeps a running total from week to week for incorporation into P35, P45 and P60s. It doesn't yet handle the new SST regulations, but the company promises that there will be a program available shortly to do this. Neither does this program allow for items such as pension schemes, union subscriptions and BUPA. It would be a relatively easy task to incorporate them into the program, but this would reduce the amount of memory available and therefore the total number of employees that the program can handle.

Many software houses, including the ones mentioned above, have indicated that many more business application programs will become available for the Dragon when the long awaited disk systems become generally available. At present the Dragon, although capable of operating some business applications, is comparatively slow because of the necessity of using cassettes for data storage.

Because of these limitations you'd have to look at all the possibilities before you went out and bought a Dragon as a business machine. But if you already have the machine and you run a small business, then the Dragon can provide you with a useful business tool for a very economical price.

The 32K memory is the main restriction on the number of files one can keep in the computer at any one time. But most programs will allow between 200 and 250 files to be loaded, and this is more than adequate for the average home user.

Most of the programs available are cassette based. They usually have the facility for the user to define the fields for storage and search.

Tandy produces a cartridge called Color File for the Tandy Color Computer. This works on the Dragon, and has the added advantage of leaving a larger amount of memory available for storage. Because it is on cartridge it is instantly loaded into the machine, which halves the access time since all the user has to do is load the data cassette before operating the program.

The cartridge has a number of 'dedicated' files programmed into it, including home inventory, useful for insurance purposes, for example, and car details, for servicing records.

There is also one file that leaves the user free to make his or her own choice, thereby creating a unique database.

DRAGON

LOCATION

Dragon Data recently released a complete listing of the Dragon memory map and a list of miscellaneous memory locations. Below we present some of the important memory locations as compiled by Dragon. Next week we'll give you the full annotated memory map of the Dragon in Part III of Dragon Micropaedia.

- \$19** Beginning of text.
- \$1B** Beginning of Simple Variable Space.
- \$1D** Beginning of Array pointer table.
- \$1F** End of storage in use (= 1st free location) set to [\$1B] on clearing.
- \$21** Top of stack (from which stack grows down)
- \$23** Top of string free place.
- \$27** Highest RAM available to Basic.
- \$2D** Points to statement to be executed.
- \$2F** Text pointer for Basic warm start on reset.
- \$31** DATA line number.
- \$33** DATA memory pointer.
- \$35** INPUT pointer.
- \$68** Current line number.
- \$6F** Current device. 0=VDU, -1=cassette, -2=printer.
- \$70** End of file flag: 0=character found.
- \$71** RESTART FLAG. If [\$71] ≠ \$55 then a full cold start will be performed on RESET.
- \$72/73** Restart vector. If [\$71] = \$55 and \$72 points to a NOP then restart will be to that NOP (else coldstart).
- \$78** Cassette status: 0=closed, 1=input, 2=open for output.
- \$87** Last key pressed (but may well have been cleared by Break check).
- \$9D/E** EXEC address.
- \$9F** Start of a short self modifying routine to read in the next useful character for Basic, continued in ROM.
- \$A6/7** Text pointer (part or self modifying code). Points to current active byte.

Continued on page 223

THE MACHINE CODE MYTH

There is a long-established myth that machine code is somehow extraordinarily complicated and difficult to write. This probably stems from the fact that completed programs are often listed in lines, and lines of hexadecimal numbers make little sense to anyone. The real difficulty with machine language is not that it is complicated but that it is *so* simple. The problems arise when trying to simplify the strategy of chess or the effects of an arcade game down to the relatively few instructions that machine code has.

In an article of this size, one cannot hope to explain all of the ins and outs of the computer's native language. But enough information to encourage new users to start with Dragon machine code, and to be helpful to the more experienced user, can be provided.

The Dragon's microprocessor is the MC6809E, which was designed by Motorola — the designers of the popular 6502 as used in the BBC, Atari and Vic computers.

The 6809 has five 16-bit registers and four 8-bit registers. A register can be thought of as a memory location in the processor itself — and 16-bit registers can hold numbers between 0 and 65535, while the 8-bit versions hold between 0 and 255. The registers are X, Y, U, S, PC in 16-bit and A, B, DP, CC in 8-bit. The S, PC, DP and CC are used by the chip and cannot be used as stores, although they can be manipulated to perform various functions. The CC, or condition code, register lets the programmer know the condition or state of various options and the results of mathematical operations (see figure 1).

If you are keen about machine code, you really need an assembler. Do not make the mistake of trying to write code from a book. The assembler allows you to enter mnemonics or abbreviations as a sort of language, and then converts this into the numbers you see listed in programs.

To get the idea, let's look at a simple program which will display each character in the corner of the screen.

```

10      CLR A
20 loop  ST A 1024
30      INC A
40      CMP A # 255
50      BLS loop
60      RTS
    
```

Mnemonics are just abbreviations of the functions they represent. In line 10, CLR A means to clear the A register, *ie* set it to zero. Line 20 means store the contents of the A register into the memory location 1024 — the corner of the Dragon's text screen. Line 30 represents increment A — *ie* Let A=A+1, line 40 reads compare A to 255, while 50 reads branch, if result is less than or same, to 'loop'. Finally, RTS in line 60 means return from subroutine — like a RETURN from a GOSUB.

If you entered and ran this on an assembler you would find that the characters are displayed so fast as to be unreadable — in machine code, games are often intentionally slowed down to make them playable.

This is fine until your program wants to access the outside world — via keyboard, screen, printer or

cassette. The Basic ROMs which are built into the Dragon are just one long 16K machine code program. Certain routines can be accessed from user's routines to perform useful tasks, such as reading the keyboard. See page 223 for a list of the more useful routines. All are called using the JSR instruction, which reads as 'Jump to subroutine' — similar to a Basic GOSUB command.

Due to Dragon Data's agreement with Microsoft, the writers of the Basic, few details are available about ROM, but you can get a booklet from Dragon Data which gives some useful subroutines and addresses, together with details of the monitor and joystick socket connections. Contact the Customer Support Department at Dragon Data and ask for the *Information for Machine Code Users* booklet, which is available free.

Sound Micropaedia part 2 covered Dragon sound generation from machine code, but graphics are somewhat more complicated. As the 6809 has no I/C lines, other devices and chips are accessed as memory locations via PIAs. The graphics mode is controlled by location FF22 and SAM registers FFC0 to FFD3. Useful information about this is to be found in the Tandy *Color Computer Technical Reference Manual* which costs 99p. While some of the information does not apply to the Dragon, many useful details do.

To get to know machine code you will need a good book on the subject. I recommend *Programming the 6809* by Zaks and Labiak, published by Sybex at around £12. It is not a very 'friendly' book, but is an excellent reference guide as well as a tutorial.

An assembler will also save you a lot of time. There are now many of these available for the Dragon on tape and cartridge — while cartridges cost more, they have the advantage of using next to no RAM memory, leaving it free for your programs. The two best I have seen are DASM from Compusense and Dragon Data's Alldream assembler/editor.

Taking the programming a little further, there are several different addressing modes that can be used. Immediate addressing is when the data is in the program and follows the instruction, as in 1d A # 10 which means let A=10. Extended addressing is where the contents of the address are used as data, as in 1d A 10, which means let A = PEEK(10). Indexed addressing allows the contents of another register to point to the data to be used, as in 1d A, X, which means A = PEEK(X). Inherent is not an addressing mode, but is used when no data is required, such as for the instruction CLR A. Finally, Extended indirect takes the form 1d A [1024], which means A = PEEK(PEEK(1024)+256*PEEK(1025)). Using combinations of these gives you tremendous power over your Dragon's microprocessor.

There are two Basic commands which are not adequately explained in the Basic manual, USR and VARPTR. USRn(x) calls the *n*th routine defined by DEF USRn = Address and passes the data 'x' to the routine. Due to a ROM bug, USRn will not work if *n* is not zero, but any character in front of the number, *eg* a zero or

hash sign, will rectify this. That is A = USR9(x) will not work, but A = USR09(x) or A = USR#9(x) works perfectly.

VARPTR's only use is as the argument for a USR command. If used on a string variable, as in A = USR02(VARPTR(X\$)), it gives the address of a five byte string descriptor. The first byte is the length of the string, and the third and fourth contain the address of the start of the string. See the list of ROM routines below for details of how to pass data to machine code from a USR argument.

The example game on page 224 is written entirely in machine code. The program is quite long, and should be entered with the loader program given. Alternatively, you could put the code in data statements so that the program can be saved as Basic, otherwise save it using the CSAVEM command.

The game is for two players with two joysticks. It uses a graphics mode not available from Basic. This gives a fairly high resolution display using all nine colours. Your aim is to kill off your opponent's Droids

by breaking down his shields and shooting them before he kills your Droids. You can fire only through gaps in the moving force field — otherwise the missile will rebound and hit your own shields. The winner is the first to destroy all of the opponent's Droids. You can select the speed of the game at the beginning. Once entered, use EXEC 30000 to start the game.

Figure 1: Registers in the 6809

ACCA (8-bits)	ACCB (8-bits)
DP (8-bits)	CC (8-bits)
IX (16-bits)	
IY (16-bits)	
SPU (16-bits)	
SPS (16-bits)	
PC Program Counter (16-bits)	

These are the registers in the 6809 processor. There are 5 16-bit registers, X, Y, U, S and PC, and four eight-bit registers, A, B, CC and DP.

6809 vectors for the Dragon

SW13	100 hex
SW12	103 hex
SW1	106 hex
NM1	109 hex
IRQ	10C hex
FIRQ	10F hex

See PCN issue 24 for details of the Dragon's interrupt handling.

USEFUL ROM ROUTINES

33649	Recover and give OK prompt
35623	Return value in 'D' of value in A = USRn(value)
38266	Print value in 'D' to channel number
46080	Boot Basic interpreter
46410	Print character whose code is in 'A' to channel number
48101	Return ASCII code of any key pressed in 'A'
48466	Update joystick values stored in locations 346 to 349
48591	Turn on relay
48604	Turn off relay

Note: Channel number is set by location 111, 0=screen, 255=tape, 254=printer.

Continued from page 222

- \$B0** Address of start of USR function address table.
- \$B2** Foreground colour.
- \$B3** Background colour.
- \$B4** Active colour.
- \$B6** Graphics mode.
- \$B7/8** Top address of current graphics screen.
- \$B9** Count of bytes in a row of graphics.
- \$BA/B** Base address of current graphics screen.
- \$BD/E** Current X position.
- \$BF/C0** Current Y position.
- \$112-4** Timer \$114 cycles 0-\$FF in about 5s.
- \$120** Start of "STUB0" — Used to set up entry points for Basic keywords.
- \$120** Number of normal reserved words.
- \$121** Address of normal reserved words list.
- \$123** Address of statement despatch table.
- \$125** Number of function reserved words (tokens will be preceded by \$FF).
- \$126** Address of function reserved word list.
- \$128** Address of function despatch table.
- \$12A** STUB 1: normally a dummy to make end of stubs. Layout as for STUB0 except:
- \$12D** address of statement despatch routine.
- \$132** address of function despatch routine.
- \$134** STUB 2 (as above).
- \$134-147** Are normally used for the USR function address.
- \$15E-1AF** are subroutines (length 3 bytes) used by the Basic ROM. By default they return immediately with a RTS, but these can be modified to jump to your own code to provide extensions to the Basic. Some of the more useful ones:
- \$167** Input a character.
- \$16A** Output a character.
- \$182** Read an input line.
- \$18B** Evaluate an expression.
- \$18E** User trap } error handler
System error trap }
- \$194** RUN
- \$19A** Read in next statement (after returning, the keyboard is polled for break, so increasing the return address by 4 will disable BREAK).
- \$1A3** Crunch Basic line for storing.
- \$1A6** Decrunch Basic line for output.



DRAGON: PART 2

MACHINE CODE DATA FOR DROIDS GAME

```

7530 7E 76 AA 86 32 B7 01 37 B7 01 38 7F 01 39 7F 01 3A 8E 00 00 BF 01
7546 3B BF 01 3D 7F 01 3F 7F 01 40 7F 01 41 8E 01 12 BF 01 42 86 80 8E
755C 06 00 A7 80 8C 1E 00 25 F9 86 8D B7 FF 22 B7 FF C0 B7 FF C3 B7 FF
7572 C5 B7 FF C7 C6 08 8E 07 60 10 8E 79 A5 A6 A0 A7 84 A7 88 1E A6 A0
7588 A7 01 A7 88 1F 30 88 20 10 8C 79 B3 25 E9 30 89 02 00 5A 26 DE 8E
759E 06 00 86 EF A7 02 A7 03 A7 04 A7 88 1C A7 88 1D A7 88 1B 86 9F A7
75B4 88 22 A7 88 23 A7 88 24 A7 88 3C A7 88 3D A7 88 3B 30 88 40 8C 1E
75CA 00 25 D3 BD 76 F1 BD 76 F1 BD 77 50 BD 77 92 BD 77 D4 BE 01 44 30
75E0 1F 26 FC B6 01 39 81 08 27 0C BD 78 AE B6 01 3A 81 08 27 0A 20 D7
75F6 B6 01 37 8E 06 16 20 06 B6 01 38 8E 06 08 C6 20 3D 30 88 C6 0F 86
760C 9F A7 84 A7 01 A7 89 01 00 A7 89 01 01 8D 15 86 80 A7 84 A7 01 A7
7622 89 01 00 A7 89 01 01 8D 05 5A 26 DD 20 1E 34 04 86 3F B7 FF 23 7F
7638 FF 20 86 FF 73 FF 20 1F 89 5A 26 FD 4A 26 F5 86 37 B7 FF 23 35 84
764E 8E 10 E0 10 8E 76 8A A6 A0 84 BF A7 80 A7 88 1F A7 88 3F A7 88 5F
7664 A7 88 7F A7 89 00 9F A7 89 00 BF A7 89 00 DF A7 88 DF 10 8C 76 AA
767A 25 D9 BD 8B E5 81 59 10 27 FE AE 81 4E 26 F3 39 20 20 20 20 41
7690 4E 4F 54 48 45 52 20 47 41 4D 45 20 28 59 20 4F 52 20 4E 29 20 3F
76A6 20 20 20 20 BD BA 77 8E 05 00 9F 88 8E 76 D1 BD 90 E5 BD 8B E5 81
76BC 30 23 F9 81 3A 24 F5 34 02 C6 3A E0 E0 86 FA 3D FD 01 44 7E 75 33
76D2 20 20 45 4E 5A 45 52 20 53 50 45 45 44 20 4C 45 56 45 4C 20 28 31
76E8 20 54 4F 20 39 29 20 3F 00 B6 01 36 27 17 B6 01 41 27 09 8D 29 86
76FE 95 B7 06 0F 20 1E 8D 20 86 80 B7 06 0F 20 15 BD 77 37 7D 01 41 26
7714 05 C6 02 3D 1F 98 B7 01 36 73 01 41 20 CF 7A 01 36 39 8E 1D EF C6
772A BF A6 88 E0 A7 84 30 88 E0 5A 26 F5 39 FC 01 42 3D BE 01 42 30 05
7740 BF 01 42 C1 07 23 04 C0 07 20 F8 C8 03 1F 98 39 BD 8D 52 86 01 5B
7756 81 35 24 0E 81 0A 24 14 7D 01 37 27 0F 7A 01 37 20 0A B6 01 37 81
776C B6 24 03 7C 01 37 8E 06 16 C6 20 B6 01 37 3D 30 88 10 8E 79 B3 C6
7782 09 A6 A0 A7 84 A6 A0 A7 01 30 88 20 5A 26 F2 39 BD BD 52 86 01 5D
7798 81 35 24 0E 81 0A 24 14 7D 01 38 27 0F 7A 01 38 20 0A B6 01 38 81
77AE B6 24 03 7C 01 38 8E 06 08 C6 20 B6 01 38 3D 30 88 10 8E 79 C5 C6
77C4 09 A6 A0 A7 84 A6 A0 A7 01 30 88 20 5A 26 F2 39 7D 01 3F 26 1B B6
77DA FF 00 85 01 27 01 39 86 FF B7 01 3F B6 01 37 C6 20 3D 8E 06 96 30
77F0 8B BF 01 3B BE 01 3B 86 80 A7 84 B6 01 3F 30 86 1F 10 C4 1F C1 0F
7806 27 7D 5D 27 76 C1 1F 27 72 A6 84 81 80 27 64 81 EF 26 0F 86 80 A7
781C 84 A7 88 20 A7 88 E0 BD 79 88 20 59 81 9F 27 ED 81 D5 25 49 81 E0
7832 24 45 B6 01 3F 4C 27 02 30 01 86 80 30 89 FF 20 C6 0E A7 84 A7 1F
7848 30 88 20 5A 26 F6 86 3F B7 FF 23 C6 C8 7F FF 20 73 FF 20 1F 98 4A
785E 26 FD 5A 26 F5 86 37 B7 FF 23 B6 01 3F 81 FF 27 05 7D 01 3A 20 0D
7874 7C 01 39 20 08 86 9A A7 84 BF 01 3B 39 7F 01 3F 39 A6 84 81 80 27
788A 1F 86 01 B7 01 3F 86 3F B7 FF 23 C6 0A 7F FF 20 73 FF 20 86 64 4A
78A0 26 FD 5A 26 F5 86 37 B7 FF 23 BF 01 3B 39 7D 01 40 26 1B B6 FF 00
78B6 85 02 27 01 39 86 01 B7 01 40 B6 01 38 C6 20 3D 8E 06 89 30 88 BF
78CC 01 3D BE 01 3D 86 80 A7 84 B6 01 40 30 86 1F 10 C4 1F C1 0F 27 7D
78E2 5D 27 76 C1 1F 27 72 A6 84 81 80 27 64 81 EF 26 0F 86 80 A7 84 A7
78F8 88 20 A7 88 E0 BD 79 88 20 59 81 9F 27 ED 81 D5 25 49 81 E0 24 45
790E B6 01 40 4C 26 02 30 1F 86 80 30 89 FF 20 C6 0E A7 84 A7 01 30 88
7924 20 5A 26 F6 86 3F B7 FF 23 C6 C8 7F FF 20 73 FF 20 1F 98 4A 26 FD
793A 5A 26 F5 86 37 B7 FF 23 B6 01 40 81 FF 27 05 7C 01 3A 20 8D 7C 01
7950 39 20 08 86 9A A7 84 BF 01 3D 39 7F 01 40 39 A6 84 81 80 27 1F 86
7966 FF B7 01 40 86 3F B7 FF 23 C6 0A 7F FF 20 73 FF 20 86 64 4A 26 FD
797C 5A 26 F5 86 37 B7 FF 23 BF 01 3D 39 34 12 86 3F B7 FF 23 8E 84 D0
7992 A6 80 8A 80 B7 FF 20 8C 89 80 25 F4 86 37 B7 FF 23 35 92 D5 DA DF
79A8 DF DF DF DF DF DF D5 DA DA D5 80 80 80 C5 80 CF C5 CF CF C5
79BE CF 80 CF 80 C5 80 80 80 80 CA 80 CF 80 CF CA CF CF CA CF 80 CA
79D4 80 80 80

```

When the BASIC program is run a hexadecimal number is given to prompt you to enter the code given here. The code numbers should be entered a line at a time. When a line has been typed press Enter, and the computer will then enter that line into memory. When it comes back with the next line number, enter that line until all 55 lines have been entered. Make sure that the numbers are entered correctly, otherwise the game will not run. Note that, except for the last line, there are 22 code number per line. When all the lines have been entered, the computer will ask you to press Enter to save the game on tape. Make sure that your tape recorder is ready before pressing the key. When the game has been saved, the computer will tell you how to run the game (EXEC 30000). If nothing happens when the game is run (EXECed) it is more than likely because you have entered one of the code numbers wrongly. If so, enter the game again. To reload the game from tape use the command CLOADM "DROIDS", then EXEC 30000.

DROIDWARS

The Basic program at right is used to load the main machine code program above. The program makes it quite easy to load the machine code routines, giving you explicit instructions such as when to press which key and what to type in order to get the program loaded.

Once you've got it running, the idea is to kill off your opponent's Droids by breaking down his shields.

```

10 'MACHINE CODE DATA ENTERER
20 CLEAR200,29999
30 FORI=30000 TO 31190
40 PRINTHEX$(I); " ";
50 LINE INPUT A$
60 Z=VAL("&H"+LEFT$(A$,2))
70 POKE I,Z
80 A$=MID$(A$,3)
90 IF A$(">") THEN I=I+1:GOTO 60
100 NEXTI
110 PRINT"PRESS ENTER TO SAVE GAME"
120 EXEC 41194
130 CSAVEM "DROIDS",30000,31190,30000
140 PRINT"USE EXEC 30000 TO START GAME"

```

Contributors: Brian Cadge, David Owen, Ian Scales and Geof Wheelwright.

Design: Nigel Wingrove

Illustrations: Simon Edwards and Becky Rowlands

NEXT WEEK

We conclude our three-part series on the Dragon featuring peripherals for the machine, including a first exclusive look at the Dragon disk drives (including a full Pro-Test in the peripherals section). We'll also have an in-depth look at joysticks, printer, monitors and cassette recorders for the machine.

And in two weeks time, we begin a massive eight-part programming course that tells you Everything You Need to Know About Programming But Were Afraid To Ask. This series will take you right from the questions of What is a Program? to complex issues like How to Make the Best of Forth Programming.