

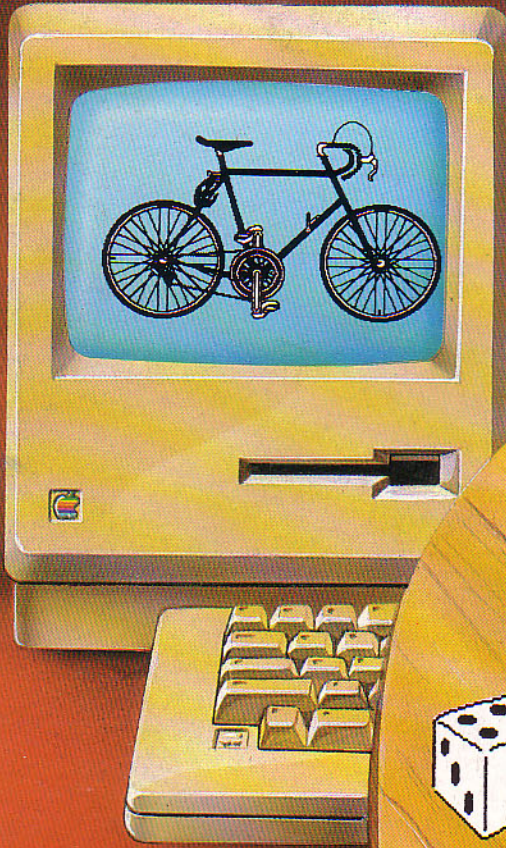


A Database Publication

# apple user

Vol. 4 No. 10 October 1984 £1

-the new name for  
**Windfall**



## The electronic artist

We review Click Art and Mac the Knife, and take a detailed look at the new programming tool, MacForth

**How to graph  
maths functions**

**How simulations  
help demonstrate  
complex problems**

**How Apples can  
speed up production**

**Flashcalc, Spectrogram,  
Blackboard and Waterbell:**  
We evaluate a spreadsheet, colour  
system, printer card and disc drive



**Appletips**

● Drive-cleaning, DOS-dabbling, Remarkable REMs and bizarre listings. **40**

**Application**

● Osia Koranteng tells how Apples are improving efficiency in a clothing factory. **30**



**Books**

● Reviews of books on games programming, Pilot and the Apple IIc. **44**

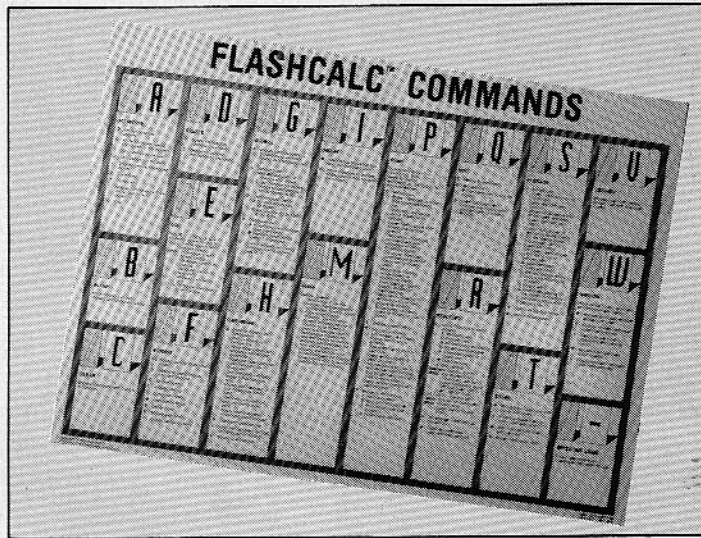
**Feedback**

● Macintosh - the good and the bad, Apple II family compatibility, protection, encryption ... and much more. **64**



**Games**

● Cliff McKnight goes into space with the



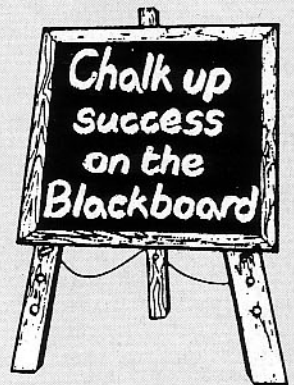
**News**

● Hemel shake-up ... Fat Mac arrives ... MEM/DOS relaunched ... tax man challenged. **5**

**Pascal**

● Gordon Findlay's tutorial series continues. **42**

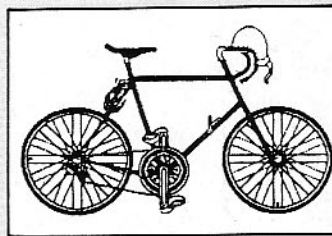
**Reviews**



● Max Parrott has a busy time with the Blackboard printer card ...

... and the CW/P Waterbell disc drive. **60**

**Macintosh**



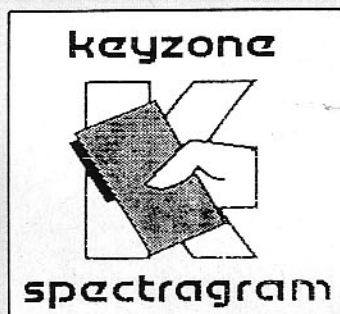
● Eugene Evans finds MacForth the answer to his problems, while Cliff McKnight uses Click Art and Mac the Knife to make up for his lack of artistic ability. **33**

**New products**

● Three pages of ideas on how to spend your money. **56**

Gruds, makes a fortune discovering oil and loses it on a one-arm bandit. Julian Brewer blasts away at Aquatron. **15**

**Graphics**



● Part VIII of Peter Gorry's Graphics package, including a review of the Spectrogram colour card. **50**

**Simulations**

● Dave Miller offers programs to simulate radioactive decay and Einstein solids. **48**

**Spreadsheet**

● Nick Levy looks at Flashcalc, the latest offering from VisiCorp. **20**

Apple User Managing Editor **Derek Meakin** Features Editor **Cliff McKnight** Technical Editor **Max Parrott**  
 Production Editor **Peter Glover** Design **Heather Sheldrick** Advertisement Manager **John Riding**  
 Sales **Peter Nowell** Marketing Manager **Tracey Nash** Editor-in-Chief Database Publications **Peter Brameld**

Telephone: 061-456 8383 (Editorial) 061-456 8500 (Advertising) 061-480 0171 (Subscriptions) Telex: 667664 SHARET G. Prestel: 614568383.

Published by Database Publications Ltd, Europa House, 68 Chester Road, Hazel Grove, Stockport SK7 5NY.

Trade distribution in the UK and overseas: Contact Steve Fletcher, Circulation Manager of Database Publications at the above address or telephone him on 0293 26945. Apple and the Apple symbol are the registered trade marks of Apple Computer Inc. Apple User is an independent publication and Apple Computer is not responsible for any of the articles in this magazine, nor for any of the opinions expressed.

Writing for Apple User: Articles and programs relating to the Apple are welcome. Articles should preferably be typed or computer-printed, using double spacing. Unsolicited manuscripts, discs etc, should be accompanied by a self addressed stamped envelope, otherwise their return cannot be guaranteed. Unless agreed, material is accepted on an all rights basis.

© 1984 Database Publications Ltd. No material may be reproduced in whole or in part without written permission. While every care is taken, the publishers cannot be held legally responsible for any errors in articles or listings.

Subscription rates for 12 issues, post free:  
 £12 UK  
 £13 Eire (IR £16)  
 £18 Europe  
 £15 USA (surface)  
 £25 USA (airmail)  
 £15 Rest of world (surface)  
 £30 Rest of world (airmail)

**A TOP-LEVEL shake-up at Apple (UK) following the appointment of new boss David Hancock is finally over.**

At least that was the prediction of a company spokesman when contacted by *Apple User*.

"It has finished", he claimed, "for now David Hancock believes he has the company structured for future success".

The night of the long knives down in Hemel Hempstead saw six executives bite the dust. These included such formerly prominent Apple personnel as marketing communications manager Mike Spring and key product men, Nigel Parry and Steve Holmes.

Since then the Apple team has been increased by 10 per cent – with emphasis being placed on strengthening marketing, sales and communications.

The new-look Apple team now boasts Richard Strong as creative communications manager, John Hill sales communications manager, Stuart Bagshaw sales manager and Bob Kissach marketing manager.

Moving over from the Lisa group, Phil Peters is to head up a new team known as the Business Software Unit.

As part of the corporate shake-up, product managers will be known as brand man-

## CP/M for the IIc

THE race to produce a CP/M card for the Apple IIc has been won by a Scottish firm Cir Tech, which claims the product as a world first.

It enables IIc owners to run all the standard CP/M software that runs on the Apple IIe or II+. No modifications to programs is necessary, they say.

The three year old company produces a range of cards for the Apple IIe and Apple II+.

Says design manager John Robertson: "We expect to have dealer samples out this month, and volume production should be going a couple of weeks later".

He added that the price would be around £100 fitted.

# That big Apple shake-up is over

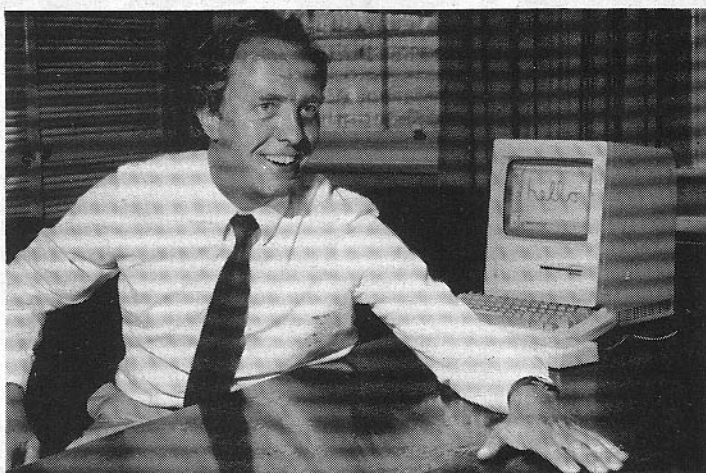
agers. This means that Neil Davison, formerly product manager for the Apple IIc now becomes brand manager for the Apple IIc and IIe.

Now the company is currently looking for a brand manager for Macintosh and Lisa.

Yet another major departure at Apple HQ will be that departments will no longer have secretaries.

From now on there will be "area coordinators" who will "fulfil a more creative role".

"It means doing away with girls having to sit there just to answer the phone or type", said the man from Apple.



*Hello from David Hancock meant Goodbye for some – but that's all over now.*

## FAT MAC WEIGHS IN AT 512k

FAT Mac – the long awaited, more powerful brother of the Macintosh – has finally wobbled onto the scene, weighing in at 512k.

To be known more conventionally as the 512 Macintosh, it was due to be on sale by the start of this month.

At the same time an upgrade memory expansion package is to be made available for current Macintosh users.

The Fat Mac boasts a number of benefits over the 128k machine.

Its greatly enhanced memory facility allows it to run software

considerably faster. And it also offers larger documents and models which result in:

- Eight times the number of pages in MacWrite.
- Ten times the tasks in MacProject.
- Fifty per cent larger spreadsheets.

Fat Mac was born somewhat prematurely as far as Apple (UK) officials were concerned.

"We hadn't in fact expected it for a few weeks yet", admitted a

company spokesman. "It went into production a lot faster than most people anticipated.

"But as the demand was obviously out there, it was decided to meet it as soon as possible".

## Top game for Mac

ONE of the top selling computer games, Broderbund Software's Lode Runner, is to be released soon for the Macintosh.

Lode Runner has won several software industry awards and has been in the top 10 for nearly a year.

The game's popularity is due to its combination of action, strategy and limitless variety made possible by 150 playing screens plus a built-in game generator that allows the creation of custom playing screens.

## MAC SALES SOAR

**A FORECAST that world wide sales of the Macintosh will reach 300,000 this year and 500,000 by the end of 1985 has come from the United States.**

**The claim was made at a two day conference sponsored by Future Computing.**

**Experts told the gather-**

**ing that after only five months on the market the Macintosh is hard on the heels of the IBM PC and fast moving up on the Apple II.**

**It was revealed that the Macintosh is currently the third best selling computer in computer speciality stores in America.**

# A big sales boost is on the cards



APPLE is introducing its own credit card this month backed up by instant credit in some cases.

Dealers are being issued with brochures explaining the new system and containing application forms.

Credit-seeking customers have to be over 21, permanently resident in the UK and have either a cheque guarantee card or an acceptable credit card such as Visa, Access, Amex or Diners.

Speed of approval for the charge card depends on how much the customer wants to spend after he or she fills in the

application.

- Under £100 – immediate approval on completion of the application form.

- £100 to £250 – a phone call is made to Welbeck Finance (who administer the scheme on Apple's behalf) to obtain an approval number. Credit is confirmed immediately with confirmation through the post.

- £250 plus – approval is sought from Welbeck by phone, although in the case of this high credit range (up to £1,500 and occasionally more) the application details must be read over the line. Approval for the purchase should be completed

within 20 minutes.

That's all it takes for qualified customers to leave the premises with a new Apple.

The actual Apple credit cards will be sent out by Welbeck Finance, arriving about seven days following application approval.

Rates for individuals using the card will be slightly higher than Barclaycard or Access, with monthly repayments being a minimum of five per cent or £5, whichever is the greater.

Apple says it regrets the new credit system is not available in Northern Ireland, the Isle of Man or the Channel Islands.

## Primrose goes it alone

PRIMROSE Publishing and the Longman Group have come to an amicable parting of the ways regarding the Apple compatible Tick-Tack instant correspondence kit.

Control, production and marketing of Tick-Tack has been returned to Primrose "in order to ensure closer personal support for users and dealers".

Tick-Tack is the business software package enabling users to compose letters in English and have them automatically translated and typed up in foreign languages.

Based on a data bank of coded building blocks that correspond in all the languages, separate Tick-Tack packs are available for French, German, Spanish, Italian, Swedish, Dutch, Basque, Swiss Romansch and English at £150 each.

Books containing the building blocks cost £15 each.

## Apple's American Dream

APPLE has never received such favourable media attention as it is enjoying at the moment.

Even normally hard-bitten TV reporter Jonathon Dimpleby got caught up in the general enthusiasm when he popped into the Cupertino HQ during his ITV search for The American Dream.

"The staff at Apple are convinced that the future belongs to a new age that they will inherit", he told viewers.

"And perhaps they are right – but only if they outpace and outwit their rivals in that old, ruthless and earthy game called 'cornering the market'".

He paid a courtesy call on top

man John Sculley, who confided: "I believe that awareness is building that if you don't have a personal computer it's like not having a telephone. What it means is that you won't be connected to the world".

Obviously Sculley wasn't aware that Macintosh owners in Britain – despite the sophistication of the world's most advanced personal computer – are still not connected to the world.

The program that will make it possible, MacTerminal, hasn't arrived over here yet.



SYMBIOTIC Computer Systems has launched its own hard disc for Macintosh.

The company's Symbfile mass storage products are now available for Macintosh and Apple IIc.

Because the Macintosh has no expansion slots a special interface unit is provided to form a link between the machine and Symbfile.

This interface can also be used with the Apple IIc for the same purpose.

The initial marketing campaign for the new hard disc will centre on the new 3½in compact Symbfile with the internally-mounted Rodime drive unit.

Symbfile for Macintosh and Apple IIc is available in either 10mbyte, 21mbyte or 42mbyte format. Prices, including interface unit, are £1,995, £2,495 and £3,580 respectively.

## Aid goes on new database

THE Handicapped Persons' Research Unit at Newcastle Polytechnic is setting up a database of software for the handicapped – called Bardsoft.

Each entry will carry a description of the programs, what handicaps they cater for, type of micro, name of supplier, price and so on.

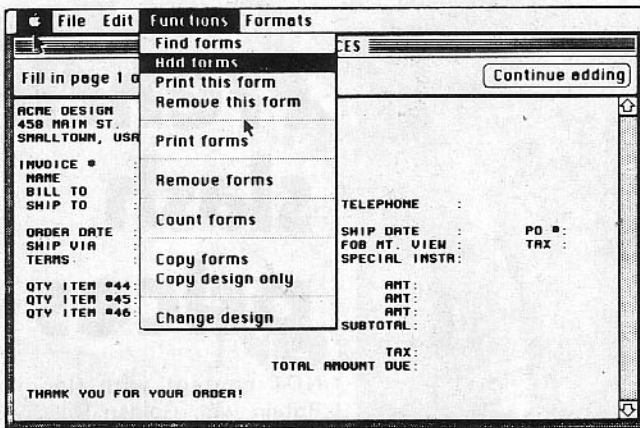
The information will be printed out in response to enquiries.

The idea is to set up a clearing house for information about software to help the handicapped, in the UK and abroad.

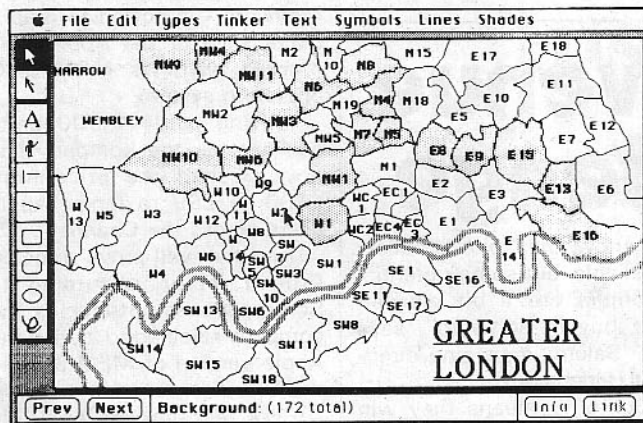
Apple User readers who wish to contribute information should contact Peter Curran at Newcastle Polytechnic, 1 Coach Lane, Coach Lane Campus, Newcastle upon Tyne NE7 7TW.

**ADVANCED Logic Systems of California is claiming its new Z-Engine card "runs CP/M on any micro that even resembles an Apple II, II+ or IIe" Said ALS vice-president of sales, Nathan Schulhof: "We tested the Z-Engine with every Apple and Apple compatible we could find and it has always performed as it should". It will retail for under \$200.**

# Team speeds top Apple programs



PFS: File and Report from Software Publishing



Filevision from Telos

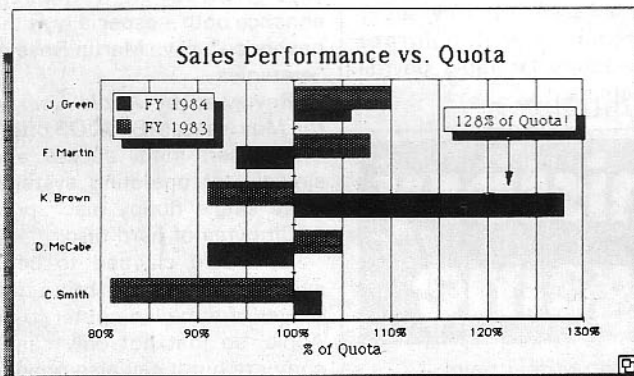


Chart from Microsoft

**APPLE (UK) has set up a specialist team to ensure that British users enjoy instant access to some of the best programs produced world wide.**

Known as the Software Business Unit, its role is to monitor the international software market, handpick major new titles, then get them to the UK with all possible speed.

It has already notched up its first success by securing the recent arrival of eight new releases from the United States – two months earlier than could have been achieved through traditional channels.

## Simultaneous

The coup came as a result of a two week trip to America by Phil Peters, the unit's manager. The major software houses he visited agreed to a simultaneous launch of the products in the UK and USA.

"Normally when a package becomes available in America it takes about a month to get over here", Peters told *Apple User*, "but in the case of those for the Macintosh – as these are – it has been taking up to two months.

So we are pleased to get hold of such hot products so quickly".

All the products – which are now on sale in the UK – have been customised to meet the requirements of the British market. They include:

**Filevision**, a new type of personal computer program, it is a visual filing system for use by anybody concerned with information management.

**PFS: File and Report**, for recording, retrieving, updating and printing information. Information is organised in forms that are user definable and takes full advantage of the Macintosh user interface.

**Microsoft Chart**, a business graphics program that turns numerical data into 40 different kinds of graphs including area, bar, column, line, pie, scatter and combinations.

**Habadex** from Haba Software, is a database which handles lists, reports, labels and customised letters.

**MacCash** from Peachtree Software allows the user to set up a cash book to almost any requirement. It uses the Macintosh user interface.

**DBSPay**, a personal and payroll application written by Developer Business Systems. All tax codes and overseas tax codes are catered for.

Nor will the new Software Business Unit be limiting its activities to the latest offerings States side.

"We will be keeping an eye on products about to be launched all over the world", says Peters.

"Clearly the biggest market is in the United States but we are also seeing interesting developments elsewhere which may result in products arriving from France and Australia in the not too distant future".

## Aid for stroke victims

HELP for deaf people and stroke victims is claimed for an interactive Apple-based system called Visible Speech Aid.

It was developed at Canada's University of Victoria to help teachers and speech therapists. It runs on the Apple II+ or IIe.

Speech patterns picked up on a microphone are analysed and displayed on screen, at the top of which is a target pattern produced by the teacher, or a

stored pattern of the student's best effort.

As the student speaks into the microphone, his voice pattern is shown on-screen below the target.

All patterns can be saved on disc for future evaluation.

To inject a little fun into the work – and to keep children interested – students can play a voice-operated game called Duck Walk.

## How to save £35,000

USING an Apple IIe, four mechanical apprentices at QH motor parts factory, Colwyn Bay, have converted a standard manual lathe to a CNC two-axis lathe for profile machining applications requiring repeatable high precision.

The project resulted in a bolt on system suitable for any centre lathe with a serviceable headstock. An additional feature is that the system does not inhibit the use of the standard features.

The apprentices have evolved a means of modernising a traditional tool room lathe at a cost of under £5,000 for work which might otherwise require a high technology machine tool costing £40,000.

Third year apprentices Simon Baguley and Carl Lewis researched and designed the project and subsequently originated the program for the Apple.

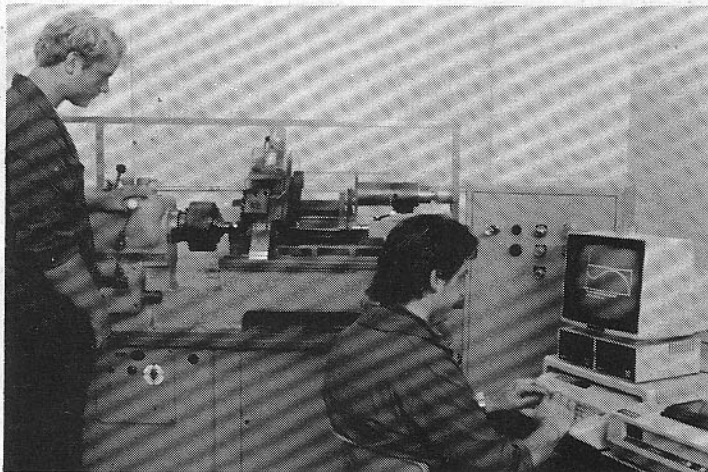
Lyndon Jones and Jeffrey Arch (third and second year apprentices) designed, developed and manufactured all the mechanical parts required for the conversion.

## ACCENT ON ICONS

TWO regular contributors to *Apple User* magazine have launched a company to specialise in Macintosh products.

Writers Keith Lander and Mike Glover have formed Icon Technology based at 9 Jarrom Street, Leicester.

It is rumoured that the new company has under wraps a Mac word processor more powerful either than MacWrite or the industry standard Wordstar.



Apprentices operate the adapted lathe

## Inland Revenue face challenge

A SOFTWARE house has thrown down a challenge to the Inland Revenue to find any error in calculation in its payroll or Statutory Sick Pay packages for the Apple.

If any taxman achieves this he will receive an Apple IIe and software – combined value over £1,000 – from the company.

Hilderbay is so confident that its programs are perfect that it is also making the same offer to any user or software reviewer.

"We know that our products are reliable, but some potential customers are a bit worried about buying software", says Mike Salem, managing director of Hilderbay.

"Our offer means they win either way".

The packages, for the Apple II, IIe and IIc, cost £75 each and are aimed at small businesses or departments within larger organisations for rapid payroll calculations.

## Re-write history with war game

**WHEN Hitler invaded Russia in 1941, his plans called for victory in four months. Four years later he was still wondering where he went wrong.**

Now Apple users have a chance to re-write history on their micros with *War in Russia*, a division-level strategic game from Strategic Simulations covering the conflict from June 1941 to December 1944.

Shorter scenarios are provided – Barbarossa 1941, Stalingrad 1942, and Kursk 1943 – allowing players to recreate just one year of the campaign or play out the rest of

the war from that point.

Game mechanics provide for limited intelligence of enemy units and simultaneous entry of orders. During the execution phase, a listing of the divisions involved in the conflict on each side is given.

Additionally, each division is kept track of for strength, experience and fatigue. During solitaire play, the micro directs the Red Army and the game in progress can be saved on disc for later.

*War in Russia* comes with disc, player documentation and laminated maps and costs \$79.95.

## French slash prices

NOT content with flooding Britain with Golden Delicious, the French now seem to want to exploit our Apples.

A company based in Nice – Memsoft – is poised to grab a major slice of the Apple (UK) market with its MEM/DOS operating system.

Having sold 12,000 units back home, the company has now slashed the price from £330 to £99 to tempt Apple users across the Channel.

And that will prove to be Le Crunch factor, according to Somerset-based Strategies, the company retained to launch the Apple version of MEM/DOS in the UK.

"We feel that the Apple IIe and IIc will remain a market for the first time business user and this product will greatly enhance both – especially at the new price", says Martin Rose of Strategies.

Reviewed by *Apple User* in the May issue, MEM/DOS offers multi user, multi access and single user operating systems for a single floppy disc up to 120mbytes of hard disc.

It is also claimed to be a system which will bring the power of a minicomputer to an Apple, so that not only can it compete but it can also provide a more practical solution than larger more expensive computers.

As part of the proposed Memsoft invasion, the product will be available in both disc and card versions – both at £99.

Strategies is currently looking for dealers to market MEM/DOS and the launch will be promoted by a national competition offering major prizes for five end users and five dealers.

However Memsoft will not be inviting the winners to spend time in its own backyard – the French Riviera. Instead the 10 winners will all go to Greece.

IF you've played any of the Sirius adventure games, you'll have met a Grud or two along the way. They're usually hidden in a cave or on an island somewhere, and don't seem to proliferate on Earth.

Once you leave terra firma, though, it's a different story. The solar system is positively crawling with them, if **Gruds in Space** is to be believed.

They're still not exactly friendly little fellows, despite the cheery wave from the Grud on the title screen. However, being practical, they'll do business with you if you've got something they want.

The adventure begins when you receive a transmission from President Fred offering you a reward of \$1 million if you get some fuel to a stranded cargo ship.

The fuel is somewhere on Saturn and the cargo ship is on Pluto. The only useful information you are given are the navigational coordinates for Saturn and the teleport coordinates for the mining camp.

It seems that Arthur Scargill's influence extends even farther than the NUM dares to hope, because you're warned that the miners have become rebellious.

Once again then it's time to boldly go where no hand has set foot. Apart from the imaginative scenario this is pretty much a standard hi-res adventure game.

The graphics are quite good and there are several bits of animation to add extra interest.

# Go imaginative hi-res adventuring with the Gruds



There are some sound effects which can be toggled off, and all the other convenience controls are there too.

Gruds is a fairly big game by single-disc standards and it involves a lot of moving about — a bit *too* much at times. There is an interesting variant on the maze theme using colour coded auto-locking doors and this will keep you to-ing and fro-ing for quite a while.

It's average to hard, but nothing really nasty. You can ask for hints, and occasionally you get one. It's certainly a

game where you should save regularly, if you don't usually.

It's also worth having saved games at various locations because it can get a little tedious entering navigation and teleport coordinates repeatedly. After all, a wrong entry can lead to unexpected visits into Grudless space.

*Title: Gruds in Space*  
*Authors: Chuck Sommerville and Joseph Dudar*  
*Publisher: Sirius Software*  
*Requirements: Nothing special*



## If it moves, zap it

RIPPING your gaze from the all-promising cover art of the package of this Defender type game, you read: "Aquatron — the best of both air, and sea arcade classics, together in one game".

Sounds good, eh?

Things looking up, in goes the disc, on goes the power, the drive whirrs in that unique way, briefly a title screen appears, and mild confusion sets in as the demo made takes command.

That demo, by the way, can provide hours of amusement, watching friends setting their high scores only to be told that they had no more control over the game than you had over your fits of laughter.

What's wrong with a little sadistic fun now and then?

To enjoy the game outside the demo, reading the instructions is not necessary. **Aquatron** is one of those games where you can just sit down and have a good old hack at anything that moves.

However, the game has a complexity not obvious at first. To reach any kind of score and to enjoy the game to the full, it is essential to look through the instructions at least once.

Deserted on an ocean-covered planet with only a base ship as an ally, you are seated at the controls of a mobile fighter. This can cruise over or under the ocean waves in a battle which moves between a blaze in the skies and a submarine hunt in the depths.

Survival relies on capturing or destroying enemy craft. You have blasters, proximity missiles and shields. Ramming the enemy is very effective, but depletes the shields.

The game is over when your shields are gone and there are no base ships left. If only the former is true, the base ship is forfeited for a new fighter.

No two enemies have the same tactic. The nine different enemies are gradually phased in as you progress through the 10 different levels.

Thus the play calls for an ever-changing approach — becoming bored does not figure in this game.

The playing area is very large — two screens deep and 32 screens wide. The fighter is equipped with hyperdrive, allowing quick access to the

distant sectors.

A sector map lies at the bottom of the screen. Each sector is shown as containing yourself, another vessel, or being empty.

On either side of the map are indicator scales, one for the shields and one for the proximity missiles. Both are replaced at the end of each level or when a parachute is intercepted from the base ship. A new base ship is earned every 20,000 points – not easy.

The game demands a joystick, although control via the keyboard is supposedly possible. The special features are all there – freeze-action, multiple players, a high score table of "Aquatron Warriors", and so on.

The graphics are good, and the animation superb. The attention to detail is admirable, notably the splashes as objects hit the water.

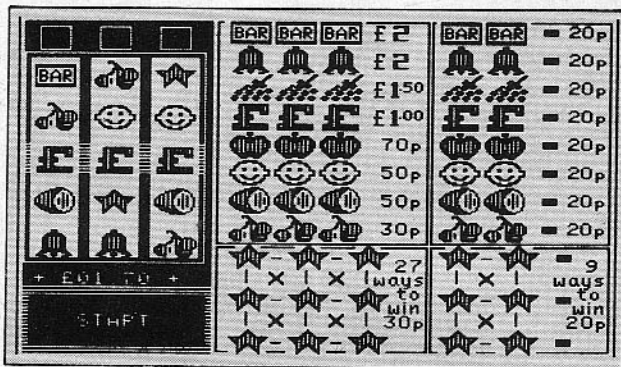
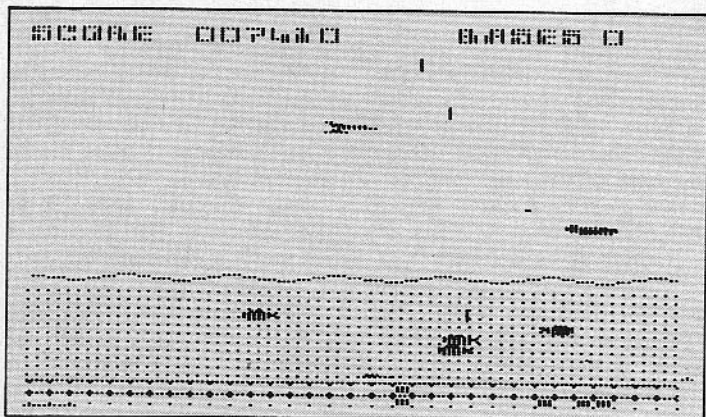
What makes things even more impressive is the speed at which it all happens. The sound effects are not outstanding, but are adequate.

Aquatron seems to have everything that a good arcade game should have. It has certainly proved popular among friends. I'm not sure why it hasn't become a personal favourite.

Perhaps I'm looking for something a little more original, but I can recommend it for Defender fanatics.

Julian Brewer

Title: Aquatron  
Author: Justin Gray  
Publisher: Sierra On-Line  
Requirements: Apple II, II+, IIe



A rare moment in credit with Cherryspin

## Take a bandit home

WOULD you believe that when I was in Las Vegas, I never once played a slot machine? It's true! Mind you, I played a lot of Pac-Man instead.

In view of this, perhaps I'm not the best person to review a fruit machine program, but here goes.

Cherryspin arrived on my desk as a result of my "By British" article a few months ago. It is a realistic simulation of a pub fruit machine, with all the usual hold, gamble and nudge features.

The display is in hi-res colour graphics, although it looks fine in monochrome. There are sound effects which can be turned off.

Of course the one missing ingredient is the money slot. The space bar is used to start the barrels rolling and you're charged 10p per go. The current credit or – more usually – debit is displayed, and there is no limit to how much you can owe the machine.

In fact, there really is only the money and smell of the bar-room missing.

Cherryspin is a well implemented simulation of a one-arm

bandit and has proved to be just as addictive at home as the real thing seems to be in pubs.

At £10 it's certainly value for money.

It's only my innate meanness that stopped me playing the slots in Vegas. Having seen the bills I've run up on Cherryspin, it's just as well.

Title: Cherryspin  
Authors: G.D. Hawker and E.J. Hale  
Publisher: Crystal Software  
Requirements: 48k

## Drilling but not riveting

I'M not usually one to boast. I mean, I've never even mentioned my Cycling Proficiency certificate, have I? However, there are two things I'm proud of – one is that I've never seen Dallas.

You might not think it's much to be proud of, but you'll be glad I haven't seen it when you've finished this review of **Oil's Well** from Sierra On-Line. Just think of all the bad jokes you'll be spared.

Yes, you've struck oil in your back garden but getting it out isn't as easy as you might have thought. The other oil barons are jealous and are trying to sabotage your operation.

They've planted oozies and land mines in your oil fields so

## CLIFF'S COLUMN

HAILING as I do from Liverpool, I'm quite accustomed to telling people that I speak English and Rubbish. However there comes a time when even a seasoned Scouser says: "Enough is enough". I reached that point today.

The straw which broke the camel's back came when I read an advertisement for some educational software which described it as a "beautiful suit of three programs...!" I thought a suit was an article of clothing kept in the wardrobe in case of interview, whereas a suite... well, that's something different.

I suppose the first straw came when I acquired an Apple. The trouble was it had colour, or rather, it didn't have colour, it had "color".

Up to that point I'd been bashing away on teletypes, submitting cards to an ICL mainframe, messing about on a PDP11 and generally not being involved with micros. Well, they hadn't really been invented

then, had they? I remember the stagecoaches were pretty bumpy, too.

You'd think that the speed with which a micro produces a screenful of text would keep me busy enough without noticing spelling. At a deafening 100 characters per second on a teletype, reading the words was a way of taking your mind off the damage to your tympanic membranes. However the more software I encountered the more examples of dreadful spelling I noticed.

Some of the errors have actually added to the enjoyment of otherwise tedious games. For example, I think "blood and destruncion" sounds great. However on discovering "a gleeming device" I had to pause for thought... is there a verb "to gleem" or is this another mistake?

Call me an ivory tower academic if you like. After all, I did manage to scrape a bare O level pass in Eng. Lang. Amateur psychologists may view my



you'd better take care of your drilling equipment.

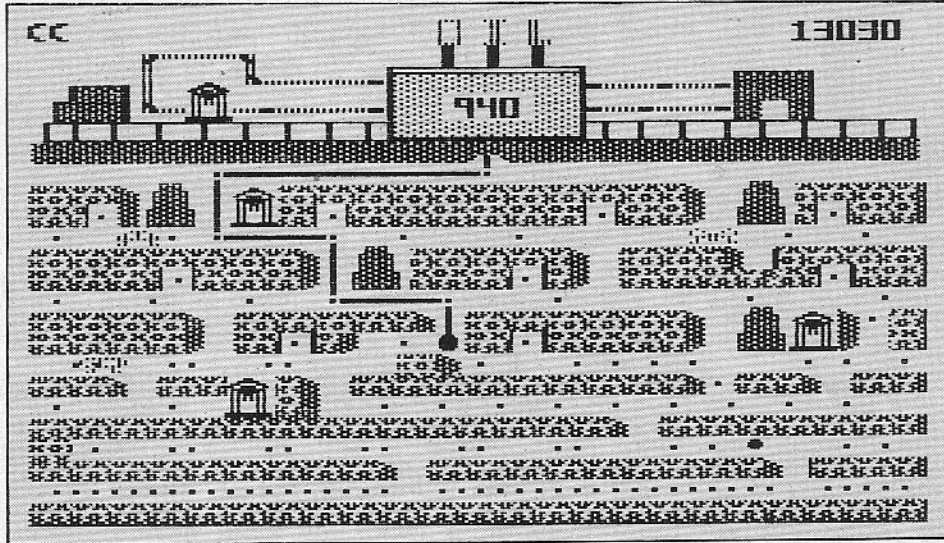
You start at the top of the field and must manoeuvre down, eating up the oil pellets with your drill bit. The oozies move along the seams and will destroy your pipe if they hit it. However, your drill bit will devour them.

There are also travelling land mines which are the opposite of oozies. They are harmless to your pipe but your drill bit will detonate them on contact and get blown up.

When I add that there are also petromins, special oil nuggets (?) that slow down the oozies, you may begin to think that it all sounds a little familiar. Is this "Pac-Man meets JR", you ask yourself?

The answer is partially in the affirmative, because the game obviously owes something to the Pac-Man tradition of maze-munching and power pills.

However, the fact that you are permanently linked to your



*You'll need to refine your joystick technique*

refinery at the top of the screen makes a lot of difference.

As you guide your drill bit down through the seams in order to clear the screen of oil pellets, the oozies and land mines are moving horizontally along them.

This means you are constantly having to retract your bit in order to avoid it being oozled.

However, you must be a little discriminating – if you retract and contact a land mine, you'll lose one of your three bits. If you clear the screen – sorry, oil field – of oil pellets you move on to the next. There are eight in all and as you might imagine, they are increasingly difficult to drain of oil.

The sounds can be toggled off and the game can be paused while you take a breath. High scores are saved, but the scoreboard can be erased if you want to hide the fact that you've been playing.

At first you're offered a choice of three levels of difficulty, so presumably there are 3x8=24 levels altogether.

Although there's nothing on the pack to suggest it, you need a joystick to play this game. Well, you could try with paddles, but you're never offered a keyboard option.

The drill bit moves in the direction last indicated by the joystick, so the only way to go

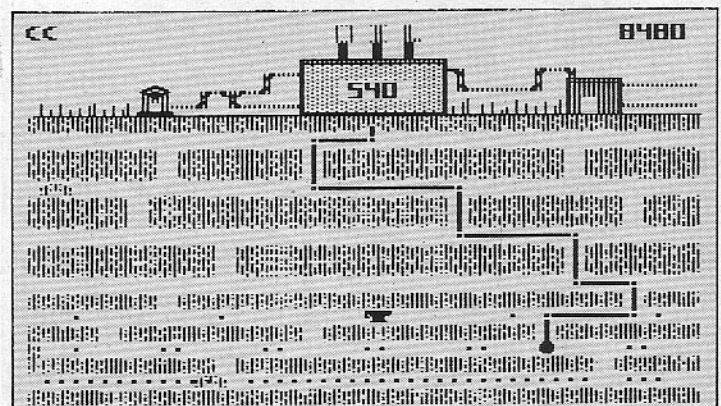
somewhere else is to retract to the last choice-point (by pressing the joystick button) and re-positioning the joystick.

If it's hard to understand, it's probably because it's hard to perform. Oil's Well is one of those games that take a bit of getting used to before you can enjoy it.

However, once you've got the knack, it's a great game that will keep you occupied for hours. It's been very popular in our household, despite being little more than an umbilical Pac-Man.

Nobody has suggested that we dig up our back garden yet... which is why I've resisted the temptation to say that "Oil's Well that ends well"!

*Title: Oil's Well  
Author: Thomas Mitchell  
Publisher: Sierra On-Line  
Requirements: DOS 3.3 and joystick*



*In search of a petromin in Oil's Well*

*concern for things linguistic as a form of compensation for my near failure. Maybe my early toilet training will lead to sesquipedalianism in later life. Who knows?*

*No doubt the modern educationalist would aver that it is the meaning that is important rather than the spelling. Semantics rule OK? Perhaps, then, the person who waits with "baited breath" has maggots in his mouth (or "magots" as I read recently).*

*If arcade games have anything to recommend them – other than sheer enjoyment – it's that they don't have any text. I mean, once you've accepted hi-res, it's a small step to "hi score".*

*Nevertheless, arcade games have presented me with "ameobas" and asked me if I wanted "instructions". I'm still recovering from the music package that asked if I wanted "sincapation".*

*The trouble is that once you notice a spelling mistake, you*

*keep on noticing it every time you use the software. No doubt it's a mark of my pedantry that the thing I remember most about Soft Porn Adventure is the high density of spelling mistakes.*

*From the software author's point of view, it is perhaps difficult to monitor spelling while involved in the intricacies of op-codes and the like. However, book publishers (and some magazines) employ a breed of super-human being known reverentially as The Proof Reader.*

*Now that software houses are making more profits than traditional publishers, is it too much to expect that they spend some of their hard-earned cash actually ensuring that their products satisfy the rules of grammar, that the words are recognisable even to the holder of O-level English?*

*Can "the disconcerning buyer" ever find "piece of mind" in the software jungle... or am I the only one who cares?*



# A flash of inspiration...

NICK LEVY looks at the latest in spreadsheets

HARDLY a month passes by these days without reading about the launch of at least one new electronic spreadsheet which is supposed to out-perform any others in its class.

It is difficult to keep pace with all these new developments, but when VisiCorp – the publishers and distributors of Visicalc – announce the introduction of a new electronic spreadsheet this must be something special, and indeed it is.

I can only describe VisiCorp's new spreadsheet as the logical next step in the development of Visicalc.

Don't be put off by the name of the new package – Flashcalc – which is not a very awe inspiring title for a piece of software intended for use as a serious financial planning tool, nor judge the product by its price (I saw it recently advertised for a mere £87.50).

Flashcalc would, in my opinion, be good value for money even at three times that price.

Its overall design, command syntax, and built-in functions are based on the conventions pioneered by Visicalc, so anyone familiar with Visicalc would find the transition fairly easy.

But why should one switch to Flashcalc? Comparing the two programs could provide some of the answers, and would perhaps be one of the best ways to evaluate the advantages and disadvantages of using the new package.

So let's look at some of the commands and functions and

other refinements available on Flashcalc which do not exist in Visicalc.

To begin with, it has the facility to produce models with variable column widths.

Suppose you want column A to be 21 characters wide, without at the same time changing the widths of any of the other columns, then all you have to do is to place the cursor anywhere in column A, type /GCC21 (followed by Return) and hey presto, column A becomes wider than any of the other columns.

You could even enter calculations in any column and then reduce its width to zero. Although that column disappears from the screen, you have not lost any of your calculations. This is of great help if you want to print a model showing only the key results and without displaying any of the interim calculations.

To return to the figures in the zero-width column, you have to enter the Goto command > and type /GCCD. The column will reappear on your screen without even having to press the Enter key.

Incidentally, you can also change the width of a column by typing /L followed by a number from zero to 80. This is even easier than typing /GCC.

/L does not appear in the definitive list of Flashcalc commands and I discovered it accidentally when momentarily I thought I was using an almost identical program called Magicalc. (/L is a definitive Magicalc command for chang-

ing the width of individual columns.)

Visicalc cells containing carefully worked out formulae are very vulnerable to being accidentally overwritten. Not so formulae used in Flashcalc because it has the facility to protect cells containing formulae from being overwritten by mistake. This protection is of great help if your models have to be handled by different people.

When it comes to the overall appearance of a spreadsheet, Flashcalc has a number of features which are not available in Visicalc.

First and foremost the problem that many VC users have with trailing zeros: Suppose you want to display 10.0 below 15.3 and add them up. Using VisiCalc these figures will appear as follows:

15.3  
10

25.3

Not so with FC where the same column can be made to look:

15.3  
10.0

25.3

Numbers used in Flashcalc can be formatted to display anything from zero to six decimal places, so you will have no problem to enter figures such as 10.0 or 12.300 or 45.6700 – something which cannot be done on Visicalc.

Furthermore Flashcalc can also be made to display commas every three places, such as 34,750.50 or 250,000

– also not possible with Visicalc.

When it comes to replicating, you can only replicate with Visicalc either one row or one column at a time. On the other hand, using Flashcalc you can replicate a whole block or a complete model in one go!

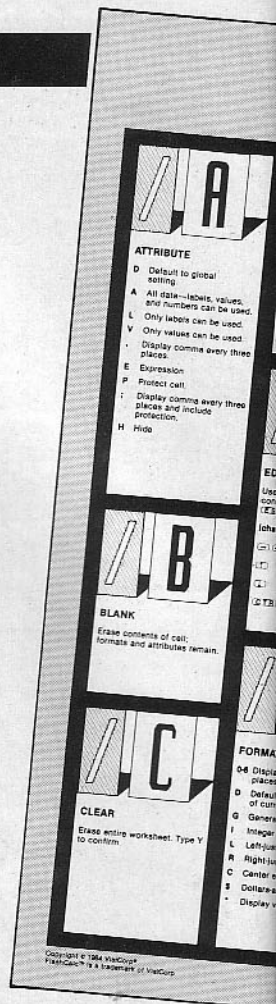
All that this involves is specifying a diagonal source range (for example A1...M20) and then entering just the top left hand cell of the target range followed by Enter.

When it comes to replicating formulae, you will still need to know, even with Flashcalc, when to respond with R (Relative) and when with N (No-change). This is probably the most tricky part of Visicalc or Flashcalc that every user has to learn.

But there is more to FlashCalc's replicate command than the ability to replicate blocks of data.

Have you had the experience of formatting a long column of figures (which were not configured by a formula) to two decimal places?

Using Visicalc you will have to format each figure



# FLASHCALC™ COMMANDS

**GLOBAL**

- A Assign attribute settings globally to all cells not having local attributes (see /A)
- C Column width (first key) Set width of column with cursor.
- C Set columns not having single column setting to typed number (0-26)
- O Order of calculation
  - R By rows (A1 to B1, A2 to B2, ...)
  - C By columns (A1 to A254, B1 to B254, ...)
- R Recalculation
  - A Automatic—recalculates after every entry
  - M Manual—recalculates when it is pressed
- F Format all cells not having individual format settings (see /F)

**INSERT**

- R Insert a row above the row that contains the cursor.
- C Insert a column to the left of the column that contains the cursor.

**PRINT**

In response to first two of /S or /S/L, type /P for partial worksheet, or /P/T/QRN for entire worksheet.

- O Set options:
  - Prefix—Directory to store file
  - Disk comment's port
  - Upper left—Upper left corner of area to be printed
  - Lower right—Lower right corner of area to be printed
  - File format—Formulas command will print DIF file in row or column order.
  - Paper feed—Set for single sheets or continuous paper
  - Print spacing—Set for single, double, or triple spacing
  - Page select—Whether to advance to top of next page after printing
  - Setup string—Control codes to be sent to your printer.
- L Display Layout menu.

**QUIT**

- C Return to worksheet.
- O Set options:
  - Prefix—Directory to use for storing
  - B Leave the FlashCalc program and load another program without loading ProDOS

**STORAGE**

- O Set options:
  - Prefix—Directory to use for storing
  - Upper left DIF file only—Upper left corner of area to be saved or loaded
  - Lower right DIF file only—Lower right corner of area to be saved (Entire DIF file always loads)
  - File format—DIF or FlashCalc format
  - DIF order—Save or load a DIF file in row or column order
  - S Save worksheet
  - L Load worksheet
  - U Display utilities menu
  - L Lock a file from being changed
  - U Unlock a file
  - C Display the catalog of file names in directory
  - I Initialize a floppy disk
  - D Delete a file
- B Return to Storage menu.
- C Return to worksheet.

**VERSION**

Display copyright notices, version numbers, and serial number.

**WINDOW**

- R Place cursor on row or column to contain window border
- H Horizontal—create window at row with cursor
- V Vertical—create window at column with cursor
- 1 Return to one window, displaying window with cursor
- S Synchronize—windows scroll together
- U Unsyncronized—windows scroll independently

**MOVE**

(loaded, /loaded)

- M Move a row if the row number changes between coordinates. Move a column if the column letter changes. The first coordinates indicate the row or column to be moved. If moving a row, the second coordinates indicate the row that should be moved down to accommodate the row. If moving a column, the second coordinates indicate the column to be moved right to accommodate the new column.

**HARDWARE**

- O Move to options list.
- Set options:
  - Prefix—Directory to contain file
  - Video card—Whether computer can display lowercase
  - Keyboard case—Whether computer can type lowercase
  - One wire shift—Whether Apple II Plus has this wire
  - Printer slot—Number of slot containing printer card
  - Video slot—Number of slot containing video card
  - Video driver—Whether video card needs a driver
- S Save configuration file.
- L Load configuration file that contains option settings
- C Return to worksheet.

**REPLICATE**

- L Limit: Type one or more options:
  - A Attributes only
  - F Formulas only
  - N All references no change
  - V Disturbed values or labels only
  - C Contents only
  - End options
- Source in row (A1..G1) for range
- row segment in column (A1..A8) for column segment; upper left and lower right (A1..D8) for block
- End options
- Target One set of coordinates copies once.
- Range copies several times.
- N No change—reference is the same
- R Relative—reference relative to new position.

**TITLES**

- H Horizontal—freeze row with cursor and above
- V Vertical—freeze column with cursor and to left
- B Both—freeze both rows and columns
- N None—remove all titles

**REPEATING LABEL**

Type character(s) to be repeated across width of cell

mainly deal with compound interest calculations (finding a future value, calculating the internal rate of return, etc.).

Unfortunately only a very small proportion of business executives in the UK are making any use of any of the financial functions built into any of the available electronic spreadsheets. So for the time being these additions would not be of great significance to most UK users – but are of enormous help to the few involved in discounted cash flow calculations.

Of the other functions there are two new functions worth mentioning, @ROW and @COL. By using one or both of these functions you can incorporate in your formulae the row and column numbers of the cell in which your formula is entered.

Why on earth should you incorporate the column or row number in any of your formulae? Let me assure you that although it is not one of the most important functions, it all helps to make life using Flashcalc a little bit easier...

The Visicalc manual only devotes eight pages to the use of functions (four in the tutorial and four in the reference section). This is a most inadequate coverage of what is in essence the most important part of Visicalc.

The Flashcalc manual, on the other hand, devotes 51 pages to guide you in the use of functions – a much more thorough coverage.

Flashcalc uses ProDOS, which is Apple's latest disc operating system. But you don't have to convert your Apple to ProDOS in order to use the program, as ProDOS is built into Flashcalc. Because it uses ProDOS it will run faster than Visicalc.

Finally I must congratulate VisiCorp for producing one of the most truly user-friendly manuals to accompany any business software.

By this I mean that the manual, which was obviously carefully thought out and very well presented, can also serve as an excellent practical tutorial.

individually. But using Flashcalc you only need to format one cell and then replicate the selected format option to all the other cells – without affecting the content of any of those cells.

Furthermore, you can also replicate simultaneously a Flashcalc attribute as well as a format, without affecting the content of the cells in the range (displaying a comma every three places is an example of a Flashcalc attribute, displaying one decimal place is a format option).

When it comes to printing, Flashcalc has the facility to fit your spreadsheet into the paper that you are going to use.

This is particularly useful if you have to print long spreadsheets and you want to avoid printing over the perforations that separate the pages when using continuous stationery.

It's done via a menu which allows you to specify such things as the length of your paper (in lines), how many lines you want to print in a page before skipping to the next page, what left margin to leave

and whether your paper feed is set for continuous stationery or single-sheet feed, and so on.

Similarly, loading and saving Flashcalc files is also performed via a menu. As with Visicalc you start by keying /S. A menu appears and you can press S or L in order to Save or Load a worksheet. In either case a catalog of all the files on the disc appears on screen.

Next to each file name there is a number, and all you have to do is to type the file number in order to save or load that file.

Should you attempt to load a file before clearing the screen a warning message appears asking you if you wish to overlay the new file on top of the one already on the screen, or to abort the Load command – another refinement not available on Visicalc.

When you save files by numbers it is very easy to overwrite the wrong file. So to reduce the risk of such occurrences Flashcalc has the facility to Lock files after they have been saved (thus protecting them from being accidentally overwritten) and Unlocking

them before they are to be updated. All this can be performed via the /S menu.

To Save or Load a DIF file using Visicalc you have to type /S/\$ or /S/L. Not so with Flashcalc where it's all predetermined via the /S menu.

If you are an experienced Visicalc DIF file user you may not find this an improvement, but it certainly makes more sense to newcomers in the uses and applications of DIF files.

If you have added memory to your computer, Flashcalc will make use of that extra memory without you having to activate that extra RAM with a pre-boot disc.

And if your model is so large that it cannot be contained in a single disc, Flashcalc has the facility to enable you to save your model on two discs – another refinement not available in Visicalc.

Next let's look at Flashcalc's functions, that is those built in mathematical formulae that begin with @.

As you would expect by now, Flashcalc has all the functions of Visicalc plus a few more. These

## FROM MICROSOFT

Taking advantage of Macintosh's mouse and rich graphics, we've designed software that works like you, even thinks like you. All our programs share the same plain English commands. So what once took days to learn, now takes hours or minutes to learn with Macintosh.

### MEET THE FAMILY

Our financial whiz is MULTIPLAN, an electronic spreadsheet that actually remembers how you work. Even offers suggestions on spreadsheet set-up.  
**Retail Price £149 + VAT**

When it comes to writing, nothing travels faster than our WORD. Using the mouse, it lets you select commands faster than you can say "cheese".  
**Retail Price £149 + VAT**

Our most artistic child is CHART. It gives you 40 presentation - quality chart and graphic styles to choose from.  
**Retail Price £99 + VAT**

FILE is our most manageable child, and advanced personal record management program.  
**Retail Price £149 + VAT**

And BASIC, the language spoken by nine out of ten microcomputers worldwide, is the granddaddy of them all. Now enhanced to take advantage of the Macintosh mouse, windows and graphics.  
**Retail Price £99 + VAT**

## MEGAMERGE FROM MEGAHOUSE

With MacWrite and Megamerge working together, you'll be able to maintain all your mailing lists and be able to print your form letters with ease! You can include standard "boiler plate" in all letters and documents. In addition MegaMerge will also "chain" all your documents together. Print as many documents together as if they were one!  
**Retail Price: £94.95 + VAT**

## DA VINCI DESIGN PROGRAMS BUILDINGS, LANDSCAPES, INTERIORS.

Hayden have introduced 3 separate program packages, to design a house or an office and then landscape it and arrange the interiors! Each program works with MacPaint to give you hundreds of design selections. You can use each package separately, or in any combination.  
**Each Package, Buildings, Landscapes or Interiors.**  
**Retail Price: £42.95 + VAT**

## MAINSTREET FILER

A filing system and report generator that can handle up to 65,000 records of up to 36 fields in every user designed file. Files can be arranged in alphabetical or numeric order on four different key indexes. Users can change the file design or indexing system after data entry without sorting, re-entering, or losing any existing data.  
Order Code: MAI 01  
**Retail Price: £189 + VAT**

# Mac Power

More for the Macintosh from P&P

Visit us at  
**COMPEC '84**  
OCEANIA, LONDON, NOVEMBER 12-14, 1984  
The computer show where  
the professionals meet

## MAC THE KNIFE

A disk chock full with over 500 illustrations, borders, icons, and maps, all convenient MacPaint files.  
Order Code: MIL 01  
**Retail Price: £29.95 + VAT**

## SARGON III FROM HAYDEN

One of the world's greatest Chess programs, now on the Macintosh! Use the mouse to move your pieces, and pull down the menus to select from 9 levels of play!  
Sargon III also lets you study in detail more than 100 great games in history, or review 45 classic chess problems. Includes a complete library of opening moves.  
Order Code: HAY 14  
**Retail Price: £42.95 + VAT**

## PROFESSIONAL COMPOSER

Is a major breakthrough in music making. The package is a program that lets you score, edit and print music using Apple's new Macintosh. Professional composer puts blank piece of manuscript paper on your computer screen, then lets you use the mouse to enter clefs, notes, rests, time signatures and chords. Even ties, beams, slurs and note ornaments! Using the same mouse, you can edit your music, delete and insert notes etc. Move instantly from one part of your score to another. Change keys. Even listen to what you've written! Because of the Macintosh's excellent graphics, you can print superb sheet music!  
Order Code: MAR 01  
**Retail Price: £429 + VAT**

## MACTOOLS

If you use a Macintosh, you need MacTools. MacTools puts the power of the Macintosh system into an easy to use disk utility package. MacTools integrates many of the standard disk functions into one program, as well as adding several new features such as:  
\* Copy Protect/Unprotect  
\* Verify a Disk  
\* Lock/Unlock Files  
\* Make Files Visible/Invisible  
Of course, MacTools will also perform the more common disk manipulations with ease. You can:  
\* Copy Files

## PC TO MAC AND BACK!

Now your IBM computer can communicate with the Apple Mac! PC to Mac and Back is a complete book/software package which enables you to communicate immediately and directly between both computers. PC to Mac and Back also works as a terminal device, so you can use your computer to access and save information.  
Order Code: DIL 01  
**Retail Price: £79.95 + VAT**

## MAC-JACK

An animated blackjack game for one player. Options include pair splits, double down, and insurance. The betting limit increases as play progresses, and high scores are saved.  
Order Code: DTP 01  
**Retail Price: £39.95 + VAT**

## OMNIS 1, 2 & 3

Three powerful, upgradeable database programs take you from file management to a sophisticated relational/hierarchical system.

Order Codes	Retail Prices
BLY004	£149
BLY005	£295
BLY006	£445

## MAC-SLOTS

Two complete games in one package. A coin operated SLOT Machine, and KENO, non-stop action like you've never seen before. Both games are totally mouse-controlled and let you pit your skill and luck against real Las Vegas odds!  
Order Code: SLF001  
**Retail Price: £59.95 + VAT**

\* Rename Files  
\* Delete Files  
\* Copy a Disk  
\* Rename a Disk  
\* Format a Disk  
all without ever leaving the friendly MacTools environment.  
We've even added pattern matching that makes working with several files a breeze. For example, you can select to copy everything that begins with "Mac", or only those files that contain the letters "Font". And MacTools lets you see all files on a diskette at once, including hidden, locked and protected files.  
Order Code: CEN 03  
**Retail Price: £39.95 + VAT**

## INTRODUCING APPLE MACINTOSH

Published by H.W. Sams & Co. 'Introducing APPLE Macintosh', covers such topics as:

- \* How MacThinking improves productivity
- \* How to get the most from the Mouse
- \* Using all the MacTools
- \* How the 68000 microprocessor works

In addition, you will learn all about communications with the Macintosh. The book provides you with a bushel of information about the Macintosh plus how to become a knowledgeable worker of the eighties.

Order Code: SMS003  
**Retail Price: £10.50**

## THE MAC BOOK

From Microsoft.  
Order Code: MST 70  
**Retail Price: £17.95**

## New Products for Mac

### FOR PRINTING: THE DAISYWHEEL CONNECTION

From Assimilation Process, the team who wrote MacWrite, comes a program which will let you use your Mac with a daisy-wheel, letter quality printer. The package includes interface cable, software program and quick start-up guide.  
Order Code: ASS 01  
**Retail Price: £89 + VAT**

### EPSTART

Your Mac comes complete with a driver to allow it to use an Imagewriter printer. But what if you want to use an Epson? Use our Epstart printer driver, install it on your Mac, connect an Epson (with serial board) and off you go!  
Order Code: SOT 02  
**Retail Price: £39.95 + VAT**

### EMULATION BOARD

Also available is a serial interface for an Epson which makes it behave like an Imagewriter!  
Order Code: HAN 01  
**Retail Price: £99.95 + VAT**

### MACPUBLISHER

A complete text and graphics layout package for small scale publishing, MacPublisher can produce multi-column newsletters, reports, product price lists etc.  
Order Code: BOS 01  
**Retail Price: £89 + VAT**

### HABADEX

A database for Mac which replaces three organisational essentials - the appointment book, the name and address card file, and all those hastily-scribbled lists of 'things to do' which invariably become lost. But, no more! Organise your life with Mac and Habadex.  
Order Code: HAB 01  
**Retail Price: £159 + VAT**

### MACFORTH

This appears to be the only development language available for the MAC at present. This implementation from Creative Solutions is excellent!  
Order Code: CRE 01  
**Retail Price: £109.95 + VAT**

Please add VAT to all prices, except books.

PRICES DO NOT INCLUDE VAT PLEASE ADD 15%  
Head Office: TODD HALL RD., CARRS INDUSTRIAL ESTATE, HASLINGDEN LANCS., BB4 5HU Tel: (0706) 217744 Telex 635740 PETPAM G  
1 GLENEAGLE RD., LONDON, SW16 6AY Tel: 01-677 7631/01-769 1022  
Telex: 919220 PPMICR G  
DALE STREET, BILSTON, WEST MIDLANDS, WV14 7JY Tel: 0902-43913  
Norwegian Agent: Programvare Huset, Okernveien 145, N-OSLO 5. Tel: 47 2 64 55 77

LINES OPEN FOR ORDERS 8am - 6pm Mon-Fri (Lancs) 9am - 1pm Sat (Lancs)  
10am - 3pm Sat (London) 9am - 6pm Mon-Fri (London & Bilston)



AUTHORISED FOR  
IBM PC COMPAG  
APPLE & EPSON  
SALES IN ALL  
OUR OFFICES



MICRO DISTRIBUTORS LTD

THOUSANDS of Apples are now beavering away all over the world, helping factory managers keep track of what's going on.

And if you've never worked in a factory, you may not realise just how much scope there is for things to get out of hand.

Take for example Berghaus, a manufacturing company based in Washington, Tyne and Wear. It employs 300 people on design and manufacture of specialist clothing and equipment.

"Before we computerised our big problem used to be work in progress", says financial director Robert Bacon. "We used to tie up large sums of money in inventory.

"You feed work into the pipeline, and you know it's in there somewhere – and that some day it will come out. But you don't always know where it is.

"Once we found stocks of a model that had been discontinued 12 months before".

But since the company began using micros about 18 months ago, efficiency has increased dramatically.

Berghaus has become a working example of how Apples can be successfully used for corporate strategy and management operations.

New designs and a steady increase in products have underlined the need for modern manufacturing methods supported by an advanced system of management.

In 18 months the company has progressed from one Apple II to a total of eight, including four Apple IIe, one Apple III and three Euro Plus machines. The benefits have proved the investment worthwhile. General oper-

# The end of 'feed it in and hope for the best'

**A clothing company reaps the benefits of letting Apples keep track of their operation. OSIA KORANTENG reports.**

ations both on the factory floor and at management level have become better coordinated.

Management use of the computers covers a wide area including sales forecasting, product scheduling and control, invoicing, motion analysis, bonus calculations and management statistics.

The nature of the company's operations necessitates that most of the software has to be tailored to their specific requirements.

John Welsh is system development accountant at Berghaus. His continual appraisal of the company's software requirements involves him in a considerable amount of re-writing of existing software and

generation of new programs.

He says: "We have adapted the Apple system to suit our particular style of operation and find that as we became more familiar with its capabilities, we quickly refined our programs for use in other areas".

One of the first applications to be introduced was Visicalc. It was used primarily to store and analyse data for sales forecasts and product schedules.

The models used have, over a period, been developed to provide a useful source of management information.

The first serious program to be written by Berghaus was a Simplex linear programming model. This, used with Visicalc, provides a way to trim the factory's capacity to suit the required output while allowing the company to make best use of its resources.

The success of both these applications prompted Berghaus to revise its method of production control.

"As we gathered more and more accurate information, we began to see opportunities for improvement in other departments", said John Welsh.

"We wrote the production controller model out of a need to re-organise our control procedures, and it has been the simplest to write and yet most

dramatic program we have used".

Until the model's introduction, checks on production were done manually every week. But the company found they were often almost a week in arrears.

The production controller program enables products to be found at any stage in their manufacture. It shows the amount of a particular product on order, and whether it has reached the cutting stage, inspection or completion. In this way management has instant knowledge of work in progress.

Updates are now carried out twice a day and accuracy has risen dramatically – from about 60 per cent to 99 per cent.

The flexibility of Apple II has made it possible for Berghaus to use the model to create other programs. Welsh has adapted it to produce a number of reports.

They include an accurate costing schedule used to calculate and monitor line efficiency and production output by time, by machine and by production line. Thus problems can be found and dealt with almost as soon as they appear.

Work in progress evaluations now take only three to five minutes. In the past a three day stock check was needed, followed by two days of calculations.

The work study department has also reaped benefits from the micro. There has been a move away from the physical recording of time to video linked methods.

Workers are recorded on video tape, which is then analysed at length to obtain precise measurements for each element within a single operation.

Mel Gillespie is Berghaus's chief work study engineer. Referring to his use of Apple II for data analysis, he said: "We now have a reliable way of maintaining extremely accurate values on manufacturing procedures as they occur on the factory floor. We also have an easy way of updating stored data as and when changes occur".

A general sewing data (GSD) program, supplied by Methods Workshop in Lancashire, is used for the analysis of work study data. A standard GSD program

```
THERE ARE 254 PRODUCTS ON CURRENT FILE
```

```
DO YOU WANT TO :
```

- 1) ADD PRODUCTS
  - 2) INSERT PRODUCT(S)
  - 3) SETUP NEW FILE
  - 4) CHANGE PRODUCT NAME(S)
  - 5) DELETE PRODUCT(S)
  - 9) END
- ```
ENTER OPTION NO
```

*The product file allows additions and changes to be made if required.*



*System development accountant John Welsh refines the Berghaus production controller model.*

is used for working out staff bonus levels as well as pence per minute costs of labour.

There do, however, appear to be some problems with GSD, mainly due to its being a standard package and therefore not easily adapted to suit specific tasks.

Berghaus find they have to use twice as many discs as they need – a problem, Welsh says, that he would like to overcome in the future, but this would depend on the availability of the source code.

While the Apple has done much to improve efficiency and enhance management operations, Berghaus has been conscious of the need to coordinate micro development with other aspects of the company's growth.

To this end substantial physical changes on the factory floor have been combined with the new computer techniques to enable an even flow of production through the factory.

Alan Brewis, factory general manager, is one of those responsible for many of these changes – the most significant being an overhead rail system.

This runs across the entire production area. Components to make a single product are placed in a bag attached by

pulley to the rail. Each worker then selects the materials relevant to the stage of production, replacing them in the bag ready for the next stage.

Coordination of changes such as this with detailed motion analysis data using the micro has led to a 50 per cent increase in production efficiency over 12 months.

This, together with the new bonus structure based around the GSD program, has led to an improvement in the level of information held on individual employee performance.

The company has developed the Berghaus Bonus Calculator to monitor workers' performances on the factory floor. The program produces a report every morning showing the performance of each worker the day before. From this, accurate measures of individuals' efficiency can be made and bonuses awarded accordingly.

Export invoicing is another area where Apple II has proved invaluable. Berghaus export many of their products worldwide – a situation which, while totally desirable, can cause headaches when it comes to raising invoices in up to 10 different currencies for a product range with over 700 customers.

The company's export invoicing program has been written to deal with this problem. It enables up-to-date price files – in all the required currencies – to be maintained simply by feeding a subset of the stock master file into the Apple.

The program was written using The Last One program generator. This enabled the results of a three month program to be produced in only two weeks.

Although the actual code was both watertight and usable, the size of data and speed of the program (one being too large and the other too slow) has meant a change from the floppy disc drives to an ICE hard disc.

Berghaus has also managed to reduce running time from 40 seconds to just 2 seconds by developing a sort and search system, one of which is a binary chop search.

Staff attitudes towards the Apple have been at worst, apprehensive and at best, a welcome revelation into the computer era.

For one payment clerk, it meant confronting a computer for the first time. "We tried to combat her fear of technology by writing a set of 'human' error messages into the program", said Welsh.

"We found that by using amusing, chastising phrases, we could help the operator to relax". So with phrases like "Rubbish, try again!" and a numeric pad specially written for Apple's keyboard, operators new to the bonus program have found little trouble getting used to the system.

The numeric pad has been designed to form a layout similar to that of a calculator using letters from the Apple keyboard. A software routine converts the comma, J,K,L,V,I, and O characters to zero, 1, 2, 3, 4 and 6 respectively.

Management use of Apple also covers quality control analysis. The company hopes to develop an export order control system to run in parallel with the export invoicing system and an asset register. Both will be tailor made.

Joint managing director Peter Lockey believes that the integration of the Apple has extended the company's avant garde approach in design to management and factory floor operations.

For Berghaus, the integration of Apple II into their manufacturing operations has proved a success. The future looks just as promising with more and more use being made of the micro.



**MacForth**

# The answer to programming problems - MacForth

**THERE** comes a time in the life of every Mac owner when he begins to tire of playing with MacPaint. Thus was my case, a totally addicted "mouse-scribbler" with few other applications to occupy my time and still have so much fun.

I had to shake the habit, so off I went to consult my MacDoctor. His answer to all my problems, "Program your own applications", of course. But what with...

That's where my search began for a programming language with which to write my own applications.

So what is available for the proud Macintosh owner? At present very little. Apple have MacBasic, MacPascal, MacLogo and a Macintosh assembler up for grabs...at least they will have eventually, because only MacBasic is yet available for sale.

Outside of Apple only two languages are available. One is MS-Basic from Microsoft, a version of their current Basic extended to make use of Mac's capabilities.

The major disadvantage here is that it is only an interpreter and is therefore comparatively slow.

That left only one language available virtually over the counter, MacForth by Creative Solutions, who have been involved in making Forth available on various 68000-based computers since 1979.

MacForth is a derivative of Multi-Forth, their standard version, but tuned to take advantage of the Mac's additional facilities. It didn't take long for me to realise that this was the answer to my problems.

What makes Forth so different from other languages is the way in which it expands with your programming experience. As you would generate a subroutine to perform a specific task in Basic, you generate a new command in Forth.

This command is then added to the vocabulary of the language and is available for use in future programs. So as you

program more and more in Forth, you develop your own commands based on the supplied primitives and create virtually your own tailor-made language.

For example:

```
In Basic: 10 REM *** DELAY SUBROUTINE ***
           20 FOR T=1 TO 1000000
           30 NEXT T
           40 RETURN
```

```
In Forth: :DELAY 1000000 1 DO LOOP ;
           (Delay is now added to the language and
           can be used in any future programs.)
```

I tried these two examples and was astonished at the comparative timings. The results are:

|           |             |           |
|-----------|-------------|-----------|
| MacForth  | Ver. 1.00   | 15.4 secs |
| MacBasic  | Pre-release | 10 mins   |
| MacPascal | Pre-release | 15 mins   |
| MS-Basic  | Pre-release | 33 mins   |
| Applesoft | Standard    | 17 mins   |

I tried the Basic version on several Basics which I had available at the time.

These examples should not be taken too seriously as the timing was done on a wristwatch and some of the versions of the languages are

pre-release, so take note of the version column.

The important aspect to remember is that all things are relative. Applesoft, for example is only running on a 1MHz 6502, whereas the rest run on a

8MHz 68000. Also MS-Basic (Microsoft Basic) is interpretive, MacBasic and MacPascal are half interpreted and half compiled and MacForth is compiled.

A final thought on these timings was put forward by a friend who was not impressed at all. He wrote a loop from 1 to 1,000,000 in 68000 machine code and we still could not time how long it took. In fact 1 to 16,777,215 was still too quick (this is a single 32 bit register being decremented from \$FFFFFFF to 1, the full word length).

MacForth now appeared to be the language to use. It is

incredibly fast, relatively easy to learn, particularly if you are an assembly language programmer, and on the Macintosh allows you to use all the Mac goodies, although more on this later.

So what can you expect when you buy MacForth? The version I received was Level 1 version 1.0. An A5 binder has a 250 page manual inside. It also includes a single disc and the usual license agreement to register you as a certified MacForth user. You will then receive all updates and information on packages to go with the system.

The binder manual is not the final print version and since receiving this I have seen a wire ring version similar to the standard Apple manuals. I assume that this will be available to all registered users in the near future.

At present only a handful of packages are available associated with the MacForth package. They are: Backgammon, RW4TH, EXTEND4TH, and STRINGFORTH.

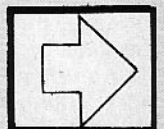
The standard unprotected Mac disc contains the relevant system files and MacForth. There are some small example programs and an interactive tutorial.

The examples include a real time analog clock, a bouncing ball and a spiral drawing program, all very mundane but interesting examples of the language.

The most fascinating aspect of these examples is the way in which they are presented. Each example has a window on screen of its own. To start a program you simply click the mouse within that window.

The window can be manipulated as a standard Mac window. That is, you can drag it around, close it, make it grow and shrink. This is a prime

**By EUGENE EVANS**



## MacForth

example of how you can generate your own integrated system in Forth. But do not expect the programs to run simultaneously, as each can only be activated individually.

The interactive tutorial, which is on disc, is an extremely well-written introduction on how to use the MacForth operating system. It is a technique I would recommend to other commercial software producers who are attempting to include a tutorial with their own package.

When you start up the tutorial you have two windows which split the screen vertically. In the left hand window is the MacForth OS and in the right hand window the tutorial.

The latter is a long scrolling text window which contains all the explanations, 100 screens worth. So you can read through this at your own pace and, if you need to, re-read whole sections simply by scrolling up and down through the window.

Several examples are given throughout and you are invited to activate the OS window and try them, then you click within the tutorial window to continue where you left off.

The MacForth Editor is probably the most disappointing aspect of the whole product, even though it is still adequate for the writing of software.

It is actually a separate window within your programming environment, which can be activated at any time to allow you to edit a block of your program.

A MacForth program in the Editor takes the form of a whole series of blocks which you then tell MacForth to accept as if you typed them in at the keyboard. This has the effect of entering the new commands which you are defining.

The Edit menu gives you the list of available commands. Most work along the same lines as those in MacWrite.

Stamp puts your initials and the date on that block of the program and Clean wipes a block of the edit space to allow you to start afresh. There is also

a series of commands to allow you to copy, move, delete or save a series of these blocks.

I found the Editor to be adequate but not as free format as it could be - I would have preferred something like MacWrite. This surely will be fixed later on in the development of the product.

The reason this version of Forth is so different from all the others on the market is simply because it runs on the Mac and allows you to access all the facilities that usually you take for granted in the Mac OS.

These include pull-down menus, mouse interface, windows, the Filer and printer commands and - probably the most impressive of all - the Toolkit and Quickdraw sub-routines in ROM. I will attempt to cover each of these aspects of MacForth in turn.



It is not within the scope of this article to act as a Forth tutorial but most of the examples should be self-explanatory. The important thing is to show just how flexible the language actually is.

The menu bar is the means to access most commands available in a Mac application. For this purpose MacForth supplies a selection of commands for the generation of your own menu bars. These allow the user to activate windows or separate items of code from the menu bar and stay consistent with the Mac user interface.

Here is an example of how to create the definition for the menu bar:

```
0 "My Menu" EXAMPLE NEW.MENU
  item 1<b<u;item 2/2;item 3<i<" EXAMPLE APPEND.ITEMS
DRAW.MENU.BAR
```

Here the actual command is in bold letters and the parameters which are being passed to it are prior to the command on the line.

The command **NEW.MENU** sets up a menu item called **EXAMPLE** which contains the text "My Menu". The zero at the

start is menu insertion point and tells the menu handler where to place the menu on the new menu bar.

The zero means append it to the present contents. **EXAMPLE** is actually a constant which you have set previously and will indicate the number of the menu on the menu bar. You can have 16 menus and these are numbered 0 to 15.

Of course there will be problems if there is too much text to fit on the line. The next line actually defines what items are to go into the menu. You simply pass the command **APPEND.ITEMS** the name of the menu you wish to access, here **EXAMPLE**, and a string to define the actual contents. The above definition looks like this:



The command string allows you to put in as many menu items as you like and gives you various options on how they look.

You can make them bold, italic, outlined, shadowed or underlined. You can show them as a disabled element in the menu. You can make it possible to activate the option on the menu by a command key (the same way that command-x in MacWrite does a cut) and place a character in front of the menu item, such as a tick mark or apple (the tick mark works like the one in the Style menu in MacWrite).

Then to actually activate the menu bar definition you execute **DRAW.MENU.BAR**. This then replaces the old menu bar with yours.

You will, of course, wish to make the menu items actually do something when they are activated, so you can assign a

different command to each.

The structure of a menu bar and the menus can be altered later with a selection of further commands. So that, for example, if a command in a menu was disabled from use and therefore in grey, it can be enabled later when it is in use, by using the command **ITEM.ENABLE**.

These are just examples of how easy it is to set up your own menu bar. Due to the structure of Forth you can make the items a lot more complex. In fact, whole programs can be activated from a single pull-down menu.

Another good thing about the way in which the software handles menus is how most of the work (for example highlighting, causing the menu to pull-down and selection of the menu) is handled by the OS. All you are doing is defining the way you want it to work.

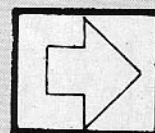
The mouse being the major part of the user interface to the Mac OS and all its functions, it is handled mostly by MacForth itself. On the whole, most tasks performed by the mouse are handled automatically, such as highlighting menu items, dragging a window, opening and closing window, and so on.

There is however a selection of primitives which allow you to access the mouse directly, to enable you to use the mouse for more specific functions in your own applications.

These commands include the ability to read the condition of the mouse button, functions to return the position of the mouse either as a x,y location within the screen or as a x,y within the window that it occupies at the time.

More advanced commands record events generated by the user clicking within windows and the menu bar and thus generating an event, the Mac equivalent of an interrupt.

By using a combination of





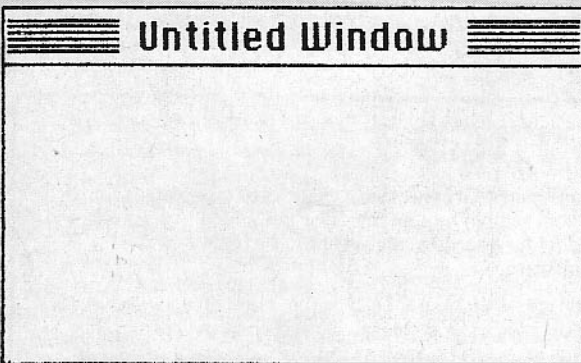
## MacForth

these and other commands associated with mouse-generated events, it is possible to detect such functions as the mouse being clicked within the desktop or within any window that is presently active.

Most of this work is handled automatically by the other commands in the system. The mouse functions are used mainly within the definitions of these commands and are therefore rarely seen.

The window-handling software within MacForth allows you to define your own window layout within your application, but it also allows you to generate simple windows to get started using standard defaults.

A default window looks like this:



To create this window on the screen you simply type:

```
NEW.WINDOW MY.WINDOW
MY.WINDOW ADD.WINDOW
```

The command **NEW.WINDOW** creates the window you see above, using the default values and assigns the internal name "MY.WINDOW". This name is not the one which is used as the title to the window, but your means of referencing it later on in your program.

**ADD.WINDOW** then adds that window to your desk top and causes it to be displayed on the screen, in among all your other applications.

There is a set of commands related to the window manager which allows you to define your own parameters, instead of the default values. These include

**W.TITLE** to set up your own title, **W.BOUNDS** to allow you to set the position of the window on the screen initially and **W.ATTRIBUTES** which allows you to define whether the window has a close box or a size box. As with the menu commands you are able to change any of these parameters later. This code would produce the window shown at the bottom of the page:

```
NEW.WINDOW REAL
  "Real Window" REAL W.TITLE
  CLOSE.BOX SIZE.BOX + REAL W.ATTRIBUTES
REAL ADD.WINDOW
```

The most important aspect of windows on the Mac is, of course, the fact that you can assign a different function or program to each one. MacForth

the user clicks outside that window the editor can be closed and perform any cleaning up needed, such as to save the text typed to disc.

As with all the commands in the MacForth vocabulary, the window-handling commands allow you to stay consistent with the Mac user interface and make your applications look and act like all the other programs on the market.

Quickdraw is the name given to a large proportion of the internal 64k ROM of the Mac or the Toolkit. It is a set of subroutines which handle all the interaction between the OS and the screen. The subroutines within the ROM handle all the aspects of the graphics interface.

What most Mac owners do not realise is that the toolkit ROM contains subroutines which allow you to do all the tricks so well demonstrated on MacPaint. They are not part of the main program. All MacPaint does is to make use of the 24k of 68000 subroutines to access the screen as quickly as possible. Now with MacForth these routines become available to you.

This is what struck me as being the most interesting aspect of MacForth, the ability to write your own applications with fabulous graphics displays without having to sit down for

weeks on end attempting to write the machine code primitives.

Now these primitives are under your control MacForth is, at the moment, the only development language which allows you to access these routines as flexibly and quickly as it does.

The Quickdraw interface is the largest and most complex part of MacForth and it would be useless to attempt to cover all the available commands. Let's just say that if you can do it with MacPaint then you can write a Forth routine which will produce a similar effect.

The ability to access all the routines would make it possible to produce something as powerful, fluid and fast as MacPaint but without all the hassle of assembly language. All of the examples given on the disc are of a graphical nature and demonstrate well what can be achieved.

The MacForth vocabulary is very large and covers just about every aspect of the Mac system which you would wish to emulate. The only objects which appear to be missing are the Dialogue boxes, but what you have to remember is that these may be covered in Level 2 of the software. As this provides all the access routes to all the Toolkit subroutines, you will be able to do everything which you take for granted on the Mac easily.

A lot of very advanced commands cover the more technical aspects of MacForth such as the internal system routines for running the lan-

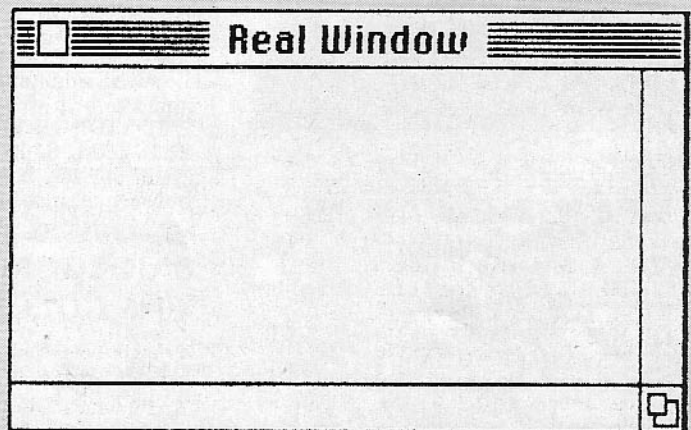
allows you to do the same with your applications. You use the command **ON.ACTIVATE** to assign a command to the window, like:

```
My.window ON.ACTIVATE CODE
```

This now means that whenever the window assigned "My.Window" is activated (clicked within by the mouse), the command CODE will be executed. CODE is not a system command, but one which you have defined previously and performs the job of that window.

For example, you could create a simple text editor and assign it to its own window. You could call this routine EDIT and give the window the same name.

Whenever the user clicks within that window, the editor will be activated and whenever



guage itself. You are also given access to the internal timers, the clock, the loudspeaker.

For advanced users there is also the DEBUG and TRACE commands on the options menu. These allow you to use the programmer's switch to enter trace mode and pressing the interrupt key will stop the program running.

The trace routine will list the names of the commands as they are executed along with the address of the routine and the stack contents.

There are also commands for advanced error handling and your own structuring of the memory map of the system. I feel that these should be left to the more experienced user though.

The programmer who produces a product using MacForth may feel that it has commercial value and that he would like to market it as a stand alone product. It is obvious though, that he does not want to give away the source, but only a stand alone version of the program that appears as just another application to the user.

For this reason CSI have three levels of the MacForth package available.

**Level 1** is the version of the package I am now describing and is the standard version of the MacForth OS. It is described in the manual as the hobbyist version. I feel that this is untrue and that Level 1 can be put to good use by even the experienced user.

**Level 2** is said to be for the professional user. It extends the vocabulary, adding more advanced graphics, a 68000 assembler and floating point. This level is not at present available and buyers ordering it will receive Level 1 and will be told that as soon as it is ready they will be sent Level 2.

For this reason exactly what the extra features will incur remains a mystery, but the advanced graphics are most interesting. Most programmers will certainly be interested in the assembler, as this will be the only means to write 68000 for the Mac if it is out before



Apple's own assembler system.

**Level 3**, for program developers, includes the run-time package and allows the programmer to generate stand-alone applications. The package includes further support from CSI and additional documentation, of which I have no details at the present. Also 250 "right to execute" licences for the commercial developer, to enable him to sell the runtime package in his own programs. (Further licences will be available at \$5 each).

As I mentioned previously, only Level 1 of MacForth is currently available and this will set you back about £120. I feel that for what this language is capable of, it is worth every penny.

I think it should be worth that little bit extra to get Level 2, but that is only from the viewpoint of a professional developer who wants to get to grips with 68000.

Whether or not any software houses will actually take up the offer from CSI for the full run-time package and produce commercial products written in MacForth is yet to be seen.

I can see, though, that unless the outlook improves on the language scene for the Mac developer, CSI is going to have very little competition.

Thanks to P & P Micro Distributors for lending us the review copy.



NO BOZOS



NO OPEN FLAME

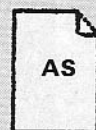


NO PREPPIES



NO SMOKING

By **CLIFF McKNIGHT**



**AS** someone with a scientific / engineering background, it came as an eye-opener to me when I worked for a time in a school of art. I used to gaze in awe as the graphics students worked with pencil, scalpel and Cow Gum.

When I first encountered MacPaint, its ability to cut, copy and paste reminded me of the graphics students. However, although the software gave me the scalpel and gum, I still lack the essential artistic ability.

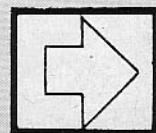
Recently, though, I've been playing with two packages which in some way compensate for this lack of ability.

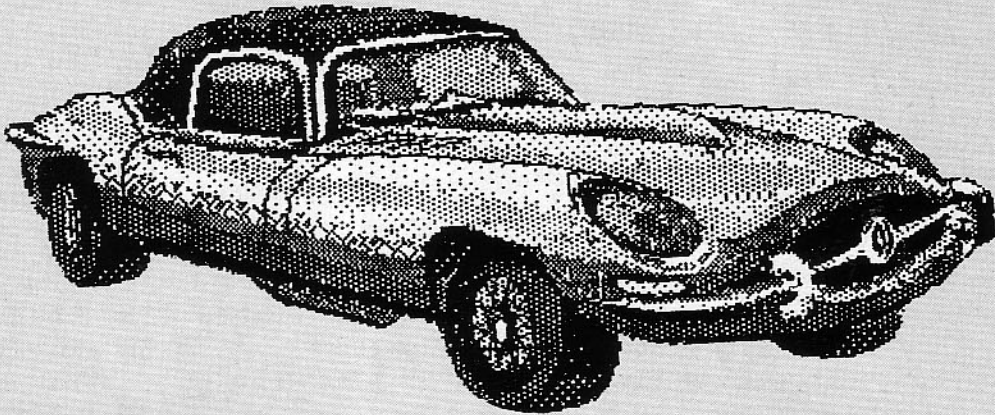
Both Click Art and Mac the

Knife are portfolios of MacPaint documents just waiting to be cut and pasted into any other MacPaint - or even MacWrite - document.

Click Art comprises 28 pages and contains "over 100" images. In comparison, Mac the Knife has only 18 pages, yet contains "over 500" images. As you might imagine, the difference lies largely in the size of the images.

However, there is also a noticeable difference in the type





of image contained in each package.

For example, several of the Click Art pages contain remarkable sketches of such people as Boy George and Albert Einstein. The emphasis seems to be on Art with a capital A, although there are several pages of graphic images.

Mac the Knife, on the other hand, has more little graphic images. One page, for example, contains 90 "international icons" like the male and female signs that you see on toilet doors. It even has some disc labels for your 3½in MacDiscs.

Click Art comes in a chunky little 4 x 4 plastic box, into which is folded the approximately A3-sized sheet of documentation.

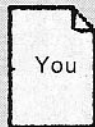
This tells you how to access the pictures via MacPaint and how to use the Scrapbook in order to get your pictures into MacPaint or MacWrite documents. It also tells you how to make a back-up copy.

Returning the registration card automatically enrolls you in the Click Art Users' Group. As a member, you get a newsletter containing examples and tips on

applications.

If you submit an idea which is printed in the newsletter you'll receive the next Click Art portfolio free. So far there are plans for business, education and design applications.

Mac the Knife comes in an A5 plastic folder complete with 24-page booklet. There are instructions on backing up the disc, printing a catalog, installing either of the two special fonts, and generally getting the most from the package.



You are invited to submit your own MacPaint images for consideration in future volumes of Mac the Knife. If they are used you'll receive a \$20 fee and a free copy of the volume in which they appear.

However, volume two has already been planned and will be devoted to lots of special fonts.

Apart from the differences I've already mentioned, there is one major difference between these two packages — price.

Click Art is £39.95, while Mac the Knife is only £29.95.

I can't see any obvious reason for one costing £10 more than the other. The time taken to draw the "artistic" pages of Click Art is compensated for by the fact that Mac the Knife has five times as many images.

Despite the undoubted quality of the Click Art sketches, they must have limited usefulness. I mean, how many times does the average graphic artist need a picture of Boy George? Even in America, a sketch of JFK can only be used so often.

On balance I think Mac the Knife would stand more chance of being useful and therefore represents better value for money.

I haven't produced any masterpieces, but I have done some customised cartoons for the people around the office and have done some nice things for my children. Anyone who can use MacPaint could make use of these packages.

I've had great fun playing with both of them — thanks to P & P Micro Distributors for lending us review copies.

*Border by Mac the Knife*

# Keep it clean..

**i** It is good practice to clean your disc drives from time to time using a drive cleaning kit. If you are like me you will eventually get around to buying one.

The instructions with the kit tell you to place the cleaning disc in the drive and to turn on the drive for 30 seconds. If you are using an Apple then it is easy to do this in the case of drive 1 - you just boot the drive and leave it spinning for 30 seconds then press Reset.

However in the case of drive 2, this cannot be done. So you probably verify a file on the drive a couple of dozen times to keep it spinning.

After a couple of performances of this routine, you realise it is too much like hard work and consign the kit to the back of a drawer and forget it with a vague resolution to look into writing a program sometime to do the job - which is what I did.

Last week I bought a pack of discs which contained a free drive cleaning kit. Now it is one thing to throw an expensive kit in the back of a drawer - it is quite another not to use a free kit.

So I got out my reference books and got down to writing a program.

The program which is listed here does the job. It first notes the currently selected slot and drive, then asks the user to input the slot of the drives to be cleaned, and whether the slot controls one or two drives.

The user is then prompted to place the cleaning disc in the drive to be cleaned and press Return. The drive motor is turned on and a delay loop keeps the motor spinning for about 30 seconds then the motor is switched off.

If there are two drives to be cleaned, the procedure is repeated for the second drive.

Finally the original slot and drive are selected. The program can be re-run if necessary where more than one disc controller card is being used.

**Colin J. Davies**

```

100 HOME : PRINT "DRIVE
      CLEANING PROGRAM"
110 PRINT
120 REM REMEMBER ORIGINAL
      SLOT & DRIVE
130 OFFSET = PEEK ( -
      18953):TEMP = OFFSET
140 DRIVE = PEEK ( - 18952)

150 PRINT "SLOT NO. FOR
      CONTROLLER CARD -->";
160 GET A$: IF VAL (A$) < 1
      OR VAL (A$) > 7 THEN 160
170 S = VAL (A$): PRINT A$
180 OFFSET = S * 16
190 PRINT "TWO DRIVES (Y/N)
      -->";
200 GET A$: IF A$ < > "Y"
      AND A$ < > "N" THEN 200
210 Y = 0: IF A$ = "Y" THEN Y
      = 1
220 PRINT A$
230 PRINT

240 PRINT "PLACE CLEANING
      DISK IN DRIVE ";
250 IF Y = 1 THEN PRINT "1":
      GOTO 270
260 PRINT
270 PRINT "& PRESS RETURN"
280 GET A$: IF A$ < > CHR$
      (13) THEN 280
290 REM ENGAGE DRIVE 1
300 DUMMY = PEEK ( - 16246 +
      OFFSET)
310 REM MOTOR ON
320 DUMMY = PEEK ( - 16247 +
      OFFSET)
330 FOR K = 1 TO 28000: NEXT
340 REM MOTOR OFF
350 DUMMY = PEEK ( - 16248 +
      OFFSET)
360 IF Y = 0 THEN 490
370 PRINT
380 PRINT "PLACE CLEANING
      DISK IN DRIVE 2 "
390 PRINT "& PRESS RETURN"

400 GET A$: IF A$ < > CHR$
      (13) THEN 400
410 REM ENGAGE DRIVE 2
420 DUMMY = PEEK ( - 16245 +
      OFFSET)
430 REM MOTOR ON
440 DUMMY = PEEK ( - 16247 +
      OFFSET)
450 FOR K = 1 TO 28000: NEXT
460 REM MOTOR OFF
470 DUMMY = PEEK ( - 16248 +
      OFFSET)
480 REM RESTORE ORIGINAL
      DRIVE
490 OFFSET = TEMP
500 IF DRIVE = 2 THEN 530
510 DUMMY = PEEK ( - 16246 +
      OFFSET)
520 GOTO 540
530 DUMMY = PEEK ( - 16245 +
      OFFSET)
540 END

```

**i** Three tips for your collection:

- The symbol for locked in DOS 3.3 is an \*. You can change this to any other character, for example by:

```
POKE 44515,ASC("L")+128
```

or any other character between the quotes.

- DOS 3.3 filetype symbols are located between 45991 and 45998 in the following order:

```

T - Text file
I - Integer Basic
A - Applesoft Basic
B - Binary file
S - unknown
R - Relocatable object file
A - unknown
B - unknown

```

To change these:

```
POKE N,ASC("Z")+128
```

where N=45991-45998 and Z can be any character.

- The text "Disk volume" is stored backwards starting at 46000, so it is therefore possible to change this to your own personal title not longer than 11 characters. Remember to POKE the message in backwards. To

make any of these modifications permanent on your disc you must format a blank disc with your new DOS using the form: INIT Filename.

**Jason W. Smith**

**i** While playing around on my Apple II+, I came across a useful tip for REM statements.

By ending and starting your REM statement with a few Ctrl-Js you can have many more blank lines above and below your words and therefore make them stand out much more than using an asterisk (\*) or any other symbols.

**Paul Kahn**

**i** Zero page location 50 (\$32-INVFLG) controls the video format. The value 255 in this location gives normal output, 127 flashing output and 63 inverse output.

The program POKES location 50 with various values and then LISTs itself. The result has peculiar

effects on the LISTING of any program. When POKING the location with value 223, the computer listing seems to display all commands and variables.

When POKING the location with value 135, the computer lists the entire program with CHR\$(7) - Bell.

The other values produce various other listing effects.

To change the listing back to normal you can POKE 50,255, or just type NORMAL.

**M. Richardson**

```

10 FOR N = 1 TO 9
20 HOME : PRINT "LIST"
30 READ A
40 POKE 50,A
50 LIST
60 POKE 50,255
70 PRINT : PRINT "PRESS
      ANY KEY: ";
80 GET OK$
90 NEXT N
100 DATA 127,135,161
110 DATA 162,164,168
120 DATA 176,192,223

```

# Restrict your variables to make life easier

Part IX of our teach-yourself series by GORDON FINDLAY

IN this tutorial series we have met several data types: Integer, Real, String, Char, Boolean and arrays of these. Pascal has many other types, and we should look at some of these.

It is often desirable to impose limits on the values which a variable can take – say restricting it to being a positive integer, or a single digit. It is easy to do this in Pascal, unlike most languages.

This is to make programs easier to follow, and easier to make correct.

By restricting the values we can be sure that the micro will tell us if something does go wrong, for example if a value which should be positive becomes negative.

This is the sort of declaration used:

```
VAR I,J,K : 0..9;
```

This restricts the variables I, J and K to being single (integer) digits, between 0 and 9. This type is called a subrange.

You have already seen these in array declarations – the type of an array subscript is always a subrange.

Subranges are usually of integers, or characters:

```
VAR CH : 'A'..'Z';
```

declares a variable which is always an upper case letter.

The programmer may define his own types. Suppose you are working on a program which frequently manipulates days of the week. It will probably make programming much easier to define a 'days of the week' type.

This is done in a type statement, which will appear in the declaration section, just before the variables declarations.

A type is given an identifier; and the possible values of the type are enumerated in a list – watch the punctuation:

```
TYPE DAYOFWEEK = (SUNDAY, MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY, SATURDAY)
```

Once you have done this, what can you do with it? Assign-

ments and tests behave naturally:

```
VAR DAY : DAYOFWEEK;
...
BEGIN
...
DAY=SATURDAY;
...
IF DAY=SUNDAY THEN...
```

Notice that the variable DAY will be one of the values listed in the type statement. This scheme means it is unnecessary to use a range of integers as you would in many languages; and then have trouble remembering whether 1 represented Sunday or Monday.

This scheme falls over when you try to output a day. If you try WRITELN(DAY) you can expect a syntax error, probably number 116 (error in type of standard procedure parameter) or 103 (identifier is not of the appropriate class).

You will probably need to output days several times, so why not write a procedure to do the job:

```
PROCEDURE OUTDAY(DAY:DAYOFWEEK);
BEGIN
CASE DAY OF
SUNDAY: WRITELN('SUNDAY');
MONDAY: WRITELN('MONDAY');
TUESDAY: WRITELN('TUESDAY');
WEDNESDAY: WRITELN('WEDNESDAY');
THURSDAY: WRITELN('THURSDAY');
FRIDAY: WRITELN('FRIDAY');
SATURDAY: WRITELN('SATURDAY');
END (case)
END (outday)
```

As you can see, a user defined type is perfectly all right

in a case statement. The effect, of course, is to convert the variable to a string for output only.

If you have tried the Turtlegraphics unit, you might have had a compile time message which gives a list of the declarations in that unit.

One of these is a type, SCREENCOLOR, which you can use as the type of a variable representing a colour.

There is a type statement:

```
SCREENCOLOR = (NONE, WHITE, REVERSE, RADAR, BLACK1, GREEN, VIOLET, WHITE1, BLACK2, ORANGE, BLUE, WHITE2)
```

and you may declare variables as, say:

```
VAR X: SCREENCOLOR;
```

Subranges of user defined types are also permitted:

```
TYPE DAYOFWEEK = (SUNDAY, MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY, SATURDAY);
WORKDAY = MONDAY .. FRIDAY;
```

The "subrange" FRIDAY .. SUNDAY isn't allowed however, as FRIDAY is listed after SUNDAY in the original (DAYOFWEEK) declaration.

Although you cannot directly input or output a variable of enumerated type, you can use them in case statements (as above), in FOR loops:

```
FOR DAY := MONDAY TO THURSDAY DO...
```

in comparisons:

```
(IF DAY = WEDNESDAY THEN....)
```

and so on. There are two useful

built in functions for dealing with them as well:

SUCC ('successor') gives the next value in the list. SUCC(MONDAY) is TUESDAY. SUCC(SATURDAY) is undefined.

PRED ('predecessor') is the reverse of SUCC, PRED(FRIDAY) = THURSDAY.

ORD gives the position in the list, starting from zero. ORD(MONDAY) = 1.

It is also sensible to use < and > with enumerated types. < means 'is listed before', so MONDAY < WEDNESDAY. Of course, > is just the reverse.

It is not permissible to have the same identifier in more than one enumerated type. This isn't allowed:

```
TYPE MONTH = (JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC);
WINTER = (MAY, JUN, JUL, AUG);
```

What is allowed is:

```
WINTER = MAY..AUG;
```

which defines WINTER not as a separate type, but as a subrange of MONTH.

Another important benefit to be gained by the use of these enumerated types is that they prevent foolish operations. In Basic, it is easy to use integers as representing days say. We might code them as 1 for Sunday, 2 for Monday, and so on.

Thus it would make sense to write DAY = 2. It would also make sense to write DAY = DAY + 1 to advance to the next day of the week. However the operation DAY = DAY \* 1.75 would also be allowed. So would DAY = SQR(DAY).

These operations don't make sense in terms of days of the week, but are permitted by the programming language.

Pascal will not allow these operations, except for the (entirely meaningful) DAY := SUCC(DAY) corresponding to the Basic DAY = DAY + 1.

In Pascal, the language itself helps maintain the security of

the program, by not allowing meaningless operations.

The type statement may be used to define types, such as arrays, which previously we have written in full. Instead of:

```
VAR A:ARRAY[1..10] OF INTEGER;
```

we could declare a:

```
TYPE TENARRAY = ARRAY[1..10] OF
INTEGER;
```

and:

```
VAR A : TENARRAY;
```

This might save some typing if we needed this type of array in several places in our program, but more importantly, it allows us to use array parameters in PROCEDURE declarations:

```
PROCEDURE(A,B,C : TENARRAY);
```

You might have already discovered that the apparently equivalent declaration, with the array type spelt out in full, is not accepted.

Put simply, the types of all procedure (and function) parameters must be just one word, so more complex types must be defined beforehand.

That is a bit of a simplification I think, but it is what is done in practice anyway.

Pascal is one of the few programming languages which allows variables to represent sets. When we worked with arrays, we found we operated primarily by passing through the array sequentially, using a loop.

Very seldom did we work with the whole array, and it is almost obligatory to work through the array in order, from first to last.

A set is rather like an array, except that the elements are not in any particular order. The elements of a set are not identified by subscript, and are not directly accessible. Two sets with the same members only listed in a different order are equal.

Sets are declared in the usual way:

```
VAR X : SET OF CHAR;
```

The type may be declared separately, as:

```
TYPE CHARSET = SET OF CHAR;
```

and the variable declaration may refer to this new type. The effect is the same.

Set variables are defined or assigned using the usual ':=' operator, by listing the elements between square brackets:

```
X := ['A', 'E', 'I', 'O', 'U']
```

or by giving a subrange:

```
X := ['A'..'Z']
```

defines X as a set containing all the capital letters.

There are operations on sets, but unfortunately, those used are standard mathematical symbols which also have other meanings.

The first is the operation of set UNION. The union of two sets X and Y is written X+Y, and is the set formed by putting the two together, but recording duplicates only once.

Set INTERSECTION is represented by \*.X\*Y is the set formed by taking only the elements which are in both X and Y. Some examples:

```
X := ['A', 'S', 'D', 'F', 'G'];
```

```
Y := ['Q', 'A', 'S', 'E'];
```

X + Y is the set ['A', 'S', 'D', 'F', 'G', 'Q', 'E']

and: X \* Y is the set ['A', 'S']

There is also the operation of set DIFFERENCE: X - Y is the set formed from the elements of X which are not in Y.

Set comparisons are catered for as you might expect. Two sets are EQUAL if they have the same members, regardless of the order they occur in:

```
['a','s','d'] = ['s','a','d']
```

The opposite is obviously the 'not equal' operator, <>. The symbol '<=' is pressed into use to represent set INCLUSION:

X<=Y means that all the members of set X are also to be found in set Y, but not necessarily the other way round.

But I have saved the most useful set operator for last - the operator representing set MEMBERSHIP.

The operation is written IN, and A IN Y is true only if A

represents one of the members of Y. Be extremely careful over types here.

If Y is a SET OF CHAR, then A must be a variable of type CHAR. This program fragment will illustrate:

```
VAR X,Y:CHAR
    VOWELS: SET OF CHAR;

BEGIN
    VOWELS := ['A', 'E', 'I', 'O', 'U',
              'a', 'e', 'i', 'o', 'u']
    X := 'A';
    Y := 'B';
```

X IN VOWELS is TRUE,  
Y IN VOWELS is FALSE.

In programming, sets are most useful for making several tests. Suppose we were working on a text handling system, and needed to know whether a character was a vowel or not, perhaps as part of a hyphenation routine.

Here are two ways of asking the question:

```
IF (CH = 'A') OR (CH = 'a') OR
(CH = 'E') OR (CH = 'e')...
IF CH IN VOWELS THEN....
```

Quite a saving in length, and much clearer also.

This becomes particularly useful when getting responses from a user. Two typical situations are getting a yes/no response, in which the key pressed should be Y, y, N or n and getting a menu selection, in which the correct responses will vary.

One way of checking a response is valid is to check that the response is in the appropriate set of responses. I have written two functions which do this:

```
TYPE CHARSET = SET OF CHAR;
....
FUNCTION CHOICE(OKSET:CHARSET):CHAR;
VAR CH : CHAR;
    GOOD : BOOLEAN;
BEGIN
    REPEAT
        READ(CH);
        GOOD := CH IN OKSET;
        IF NOT GOOD
            THEN WRITE(CHR(7))
```

```
        ELSE WRITELN(CH)
    UNTIL GOOD;
    CHOICE := CH
END;

FUNCTION YES:BOOLEAN
VAR CH:CHAR;
BEGIN
    CH := CHOICE(['Y', 'N', 'y', 'n']);
    YES := CH IN ['Y', 'y']
END;
```

An explanation...

CHOICE returns a character value. It keeps reading characters until the character read is in the set OKSET passed to it as a parameter. The first acceptable character is echoed; any unacceptable characters are beeped.

YES is a Boolean function, which reads the keyboard until a suitable yes or no response is obtained, and returns a value indicating which.

The functions might be used like this (missing procedures display a menu and ask questions):

```
SHOWMENU; (*missing procedure*)
CH := CHOICE(['A', 'D', 'R', 'E', 'a',
             'd', 'r', 'e']);
(* choices for add, delete, replace
records, or exit program *)
IF CH IN ['E', 'e']
    THEN BEGIN
        ASKQUESTION; (* missing procedure *)
        IF YES THEN EXIT(PROGRAM)
    END;
```

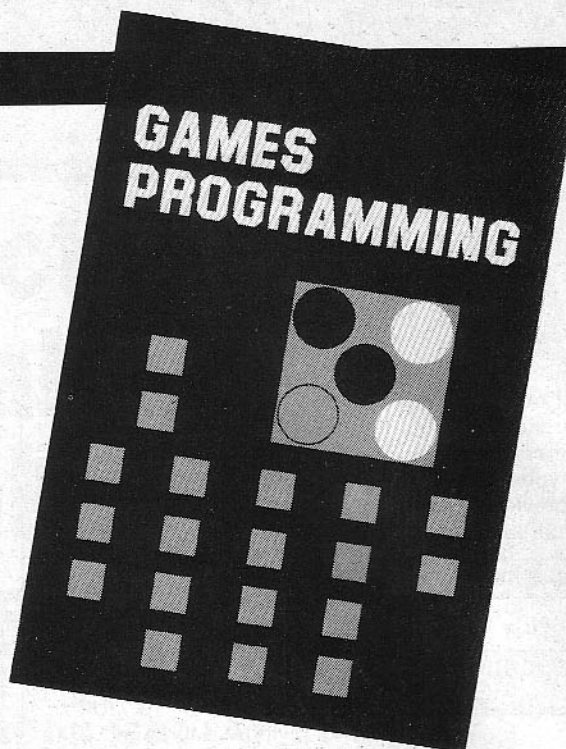
The EXIT(PROGRAM) statement is often used in this way to bail out of a program gracefully. When executed it returns you immediately to the command level of the Pascal system.

These functions illustrate the use of sets as a neat way of allowing for any number of tests. They allow the user to type in upper or lower case without having his response rejected.

As well as making programs easier to write and more likely to be correct, the use of set types is faster than any of the other ways of handling a large number of tests.

I haven't thought of any disadvantages!

# It's make or break time..



*Title: Games Programming*  
*Author: Eric Soloman*  
*Publisher: Cambridge University Press*  
*Price: £7.50*

WHO has not dreamed of writing a marvellous computer game? Instant fame, vast sums of money, write-ups in *Apple User*...?

Well, if you want to go on dreaming, don't read this book. It could either force you to write your game, or persuade you that you can't do it after all: definitely a 'make or break' book.

The author aims to take the novice programmer from a rudimentary knowledge of Basic to the fundamental concepts of serious programming, although the "non-professional programmer who has recently purchased his machine", for whom he says he is writing, may well find sections such as "mathematical techniques for games" beyond him.

It is interesting that an academic approach has been applied to what is usually considered a non-academic subject. But the book is still very readable for the most part.

The slant is definitely towards intellectual games rather than monster-zapping, with the micro considered as opponent, second, moderator, law-giver and research tool.

This is no mere collection of program listings. There is a good discussion of program design, with the "top-down" approach strongly favoured.

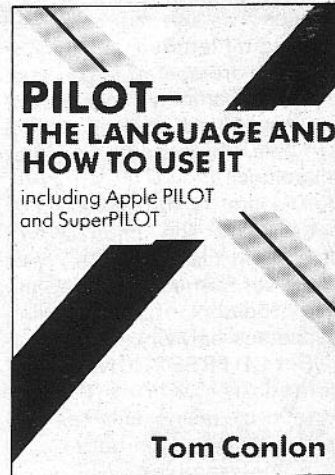
Coding in any particular language is put firmly, and rightly, into second place, and the section on "Which language?" shows a refreshing open-mindedness, setting out the strengths and weaknesses of each high level language.

Some examples from existing games are used to illustrate various techniques, and there is a useful index, plus a list of references for further study.

From writing a simple Basic program to writing a serious game is a huge step, and the book requires a lot of hard work – and O-level maths – on the part of the reader. But it's all there if you're prepared to work at it.

Be warned, however. This book could turn your fantasies into realities... or it could put you off the idea altogether.

Gill Reeve



**PILOT - THE LANGUAGE AND HOW TO USE IT**

including Apple PILOT and SuperPILOT

Tom Conlon

## Sailing through troubled waters

*Title: Pilot - The Language And How To Use It*  
*Author: Tom Conlon*  
*Publisher: Prentice/Hall International*  
*Price: £7.95*

It didn't quite make News at Ten – but Tom Conlon has recently attempted the impossible...

In one relatively slim volume, 221 pages including the index, he has tried to provide not only an introduction to the authoring language Pilot, but also discuss in detail its implementation on Apple micros through Apple Pilot and Apple Superpilot.

There are two types of beginner Pilot user – the person who

approaches it having already learned a previous programming language, and the complete beginner with no experience of micros at all.

The manuals – Editors and Language Reference – which accompany the Apple Pilot package are admittedly designed for a person already familiar with the Pilot language. Even knowledge of another language is very helpful when using them.

The total beginner is often lost when dealing with these manuals by themselves.

We might therefore expect a book on Pilot to cater for the total beginner, and Tom Conlon has aimed his book at the

person with "no previous computer experience". How well does it succeed?

Trying to evaluate a book like this is very difficult owing to the fact that it is hard to remember what it was like being a total beginner – not knowing a disc drive from a DIM statement or a FOR loop from Fortran.

The book divides into two more or less, equal parts. The first is a description of the Pilot language covering the basic Common Pilot.

Thus, in this first part, we have text display, branches, match routines, the use of strings and number handling and then three general chapters covering lesson development and testing.

The second part is no more than a re-hash of the Apple Pilot and Apple Superpilot manuals, and the topics covered are done just as well, if not better, in the manuals themselves.

Anyone buying the book having already invested in Apple Pilot/Superpilot would be disappointed by this second part.

The success of the book will thus depend, I believe, on its appeal to the total beginner, and here it partly succeeds and partly fails.

While the language is competently explained and all aspects are covered, I found it rather pedantic in approach.

As an appendix, there is a listing of a sample Pilot lesson on shingle beach fishing.

The main text of the book is peppered with small fragments of Pilot programming but we

never see these as parts of a whole Pilot lesson.

I think Mr Conlon missed an ideal opportunity here – to use fragments from his own sample lesson to illustrate his points. This would have helped unite the somewhat fragmentary text.

Minor grouses include the program listing in the appendix which is badly copied and unreadable in places (for example page 198), some very strange figures (such as figures 2.2, 3.4 and 9.1) and no indication of other implementations of Pilot.

On the other hand there are some very useful summary charts and I rather like the slightly jokey style, although that could grate with some

readers.

A success then?

I would not recommend the book to experienced programmers; if you know Pilot already then you will learn very little from this text. If you know another language then the Apple Pilot/Superpilot manuals are probably better.

If you are using Pilot on another machine, then the book does give a good insight into what you are missing by not having Apple Pilot.

For the complete beginner, you might find the book adequate but unimaginative. As a fan of Apple Pilot though, I hope it succeeds.

**Brian Rushton**

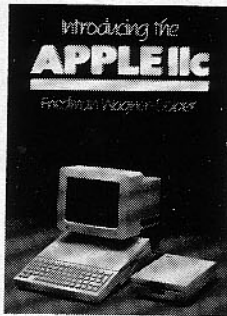
## A case of more haste less impact

*Title: Introducing The Apple IIc*

*Author: Friedman Wagner-Dobler*

*Publisher: Pitman*

*Price: £6.95*



THE letter which accompanied this book proudly pointed out that the Apple IIc was only launched seven weeks before publication.

This either means that the author had a review machine some time ago, or that the book was thrown together very quickly to cash in on the launch.

My guess is that it was the latter.

This slim volume (116 pages) spends the first 22 pages on general chat – mostly about software – and the tail end is padded out with things like a six-page description of “the Apple tree” and a four-page “Basic dictionary”.

As far as I can make out, the book is written to help you decide if you are “a suitable mind appliance owner”. Of course, your bank manager

might have something to say about that too.

With this sort of aim, the introductory chat seems reasonable. But why bother with an appendix describing the cable connections from an Apple IIc to a Disk II? Or a detailed description of how to get into terminal mode?

According to the introduction, these are included to help you save money.

Even the pages between the chat and the appendices aren't all about the IIc. For example, there are about 12 pages devoted to giving you “a flavour of what computer languages look like”.

Having said all that, the book is well-written in a clear style and it is obvious that the author knows what he is talking about, as you might expect from the technical director of a software house. My impression is that the publishers have done the author a disservice by rushing out “the first book to be published in the UK on the amazing new Apple IIc”.

**Pete Gilbert**

*“An exceptionally good program . . .”*

**Susan Curran, author of *The Penguin Computing Book*.**

*“. . . quite remarkable value for money.”*

**Hugh de Glanville, Editor of ICPUG Newsletter.**

*“. . . its programming interface . . . Adds incredible power to ‘ordinary’ database management.”*

**Benn Dunnington, Info 64.**

*“. . . a sophisticated database system.”*

**Peter Irwin, Users Newsletter (ICPUG).**

  
**Precision  
Software**



## Radioactive decay

A COMMON practical exercise for students to appreciate radioactive decay is to get them to roll dice, which represent atoms, and to remove those which land on a pre-selected side.

While this has merits, it does occupy considerable time, is often hard to control – dice being rolled off tables, etc – and the students often cannot get the feeling of true radioactive decay because of the small number of dice used.

This small number also makes the results from these experiments very irregular.

One solution is to let a computer roll the dice, count up those which have decayed, and then plot a graph of the number of undecayed atoms against time.

Since a larger number of dice is used, the resulting graph will more closely approach an exponential decay curve.

This program also allows the investigation of the decay curves of the radioactive daughters produced by the decay of the parents.

Effectively, the program "rolls" all the dice which have not decayed and then removes those which have landed on the

# Demonstrate complex problems...with the aid of simulations

**DAVE MILLER presents outlines of two simulations which should help demonstrate potentially difficult topics to students. Although space limitations do not permit us to print the actual programs, readers who would like copies should send a blank disc to *Apple User* together with a stamped addressed envelope for its return.**

pre-selected side. The graph of the number of undecayed atoms against time is then updated.

In fact, this is not really what happens, but the method used has no practical differences from the above-described algorithm.

Each pixel on the second hi-res screen represents a die

and therefore an atom. As each atom decays, its corresponding point on the screen is extinguished, thus giving a two-dimensional representation of actual decay.

For radioactive parents to stable daughter decay the whole of the screen starts off as undecayed parents, and each

atom decays into a stable daughter – signified by the point going black – but, for radioactive parent to radioactive daughter decay, a compromise with the Apple's hardware has to be made.

Each pixel on the Apple, and its look-alikes, can only have two logical states: black or white, colours being achieved by combining various patterns of pixels.

So the program must find some way of representing three states on one pixel (undecayed parent/decayed parent, undecayed daughter/decayed daughter and stable grand-daughter).

This is impossible, so the screen is split into two halves. The left half contains the undecayed parents and the right half the undecayed daughters.

As the program proceeds, a parent in the left side will decay into a radioactive daughter. This will be shown by the point on the left side being removed (parent decayed) and being placed in the corresponding position on the right side (undecayed radioactive daughter).

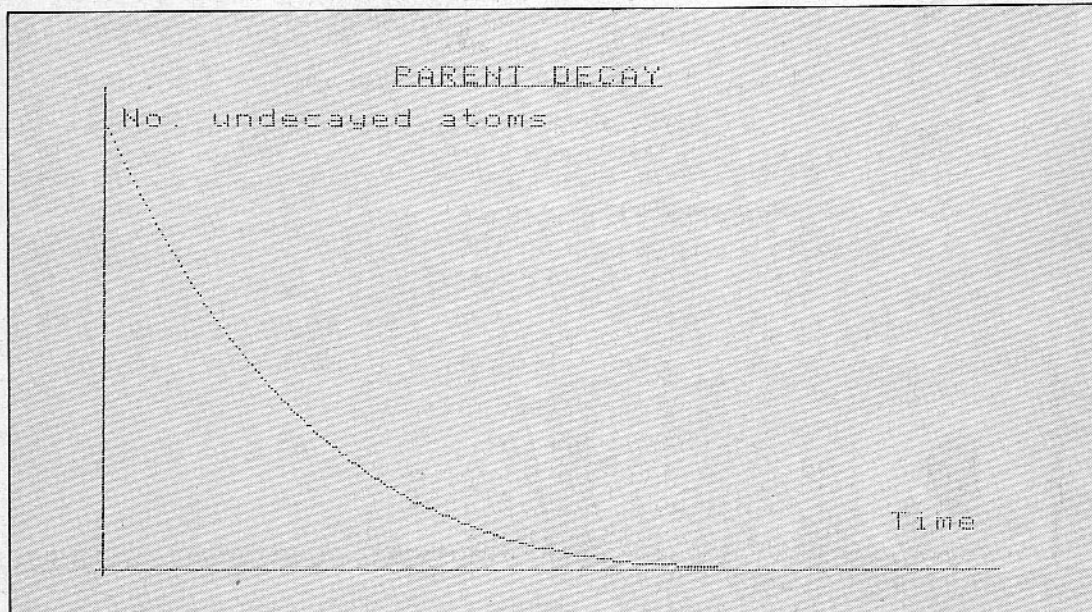


Figure 1: Final graph produced by radio-active parent to stable daughter decay

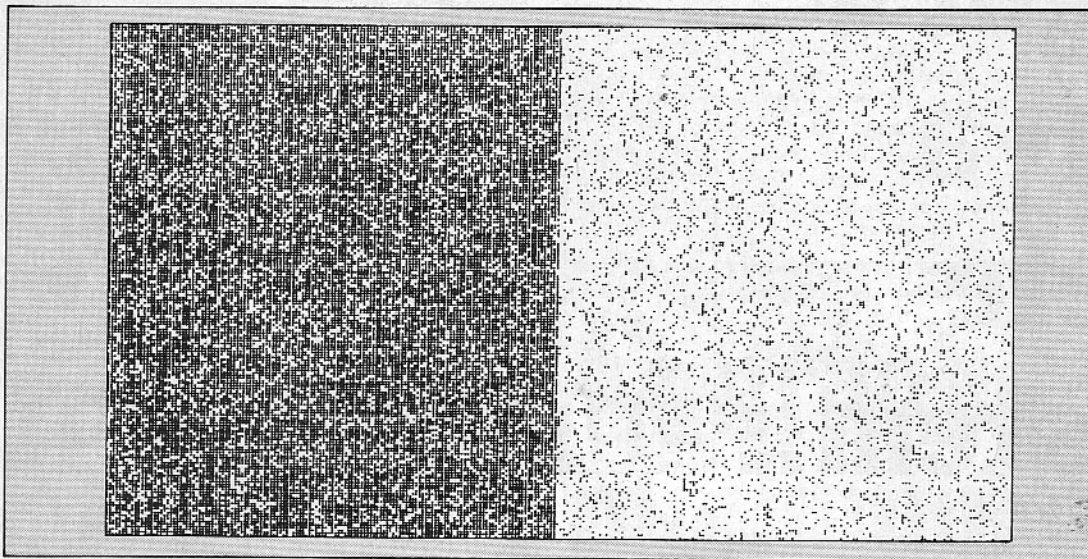


Figure II: Radio-active parent to radio-active daughter decay just started

When this daughter decays into a stable grand-daughter, the point on the right side will be removed.

As can be seen, this is a fairly complicated method of going about things but it does seem to be the only practical method which allows both parents and daughters to be visible at the same time, while not drastically reducing the total number of atoms used.

It will also be seen that when the program is carrying out a parent and daughter decay, the total number of atoms is half that of a parent to stable daughter decay.

The half-lives of both the parent and daughter atoms are the same.

There are two versions of the Basic controlling program – an HRCG and a non-HRCG.

## Einstein solids

WHEN students investigate energy transfer between atoms, they often use practical analogies to help with their understanding. One such method is described here.

A piece of paper is split into 36 squares of equal area – six rows and six columns.

Each square is referred to by its X and Y coordinates, and in each square two pennies are placed. Two dice are rolled, one for the X coordinate and one for the Y coordinate.

This roll chooses a square from which a penny will be taken. Another roll determines the square which will receive this penny. If the first chosen square has no pennies then

nothing is done.

After each move, the numbers of squares with 0, 1, 2, 3... 10 pennies are recorded. This data can be used to plot graphs at various stages of the experiment for analysis.

The squares represent atoms, and the pennies represent the energy levels (in one dimension) inherent in these atoms.

No pennies are equivalent to an atom at its ground state and subsequent numbers of pennies represent various stages of atomic excitation.

Along with this practical experiment, many schools show a Nuffield film of the graphic output of a mainframe doing this same experiment but plotting an energy distribution as the simulation proceeds.

This visual display gave me

the idea to write a similar, but more flexible, program for the Apple.

I must thank Mr I.D. Taylor for his help and ideas.

An array of 256 atoms undergoes random energy transfer while an energy distribution graph from energies  $n=0$  (ground state) to  $n=20$  is plotted.

The following are assumed to be true for the simulation:

□ The 256 atoms exist in a two dimensional array of 16 rows and 16 columns.

□ Energy levels of atoms are denoted by the expression 'n=N' where 'N' is the number of quanta in excess of the ground state which the atom possesses.

□ No energy can enter or leave the atomic array.

□ All atoms are of the same element and isotope.

□ The energy levels of all the atoms represent the inherent oscillation energy in one dimension only in excess of the atoms' ground state (so energy  $n=0$  is the ground state).

□ Any atom can transfer quanta to any other atom in the array irrespective of their relative positions.

□ Atoms at high energy act in the same way as those at low energy. Atoms are excited but never ionised.

The program uses the High Resolution Character Generator – HRCG – as supplied on the DOS Toolkit disc.

It will work with other hi-res character generators if they work with Basic, although some sections of the Basic program will have to be modified.

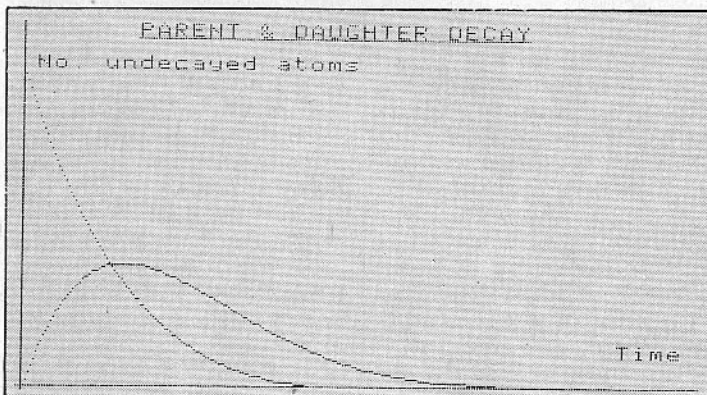


Figure III: Graph produced by radio-active parent to radio-active daughter decay

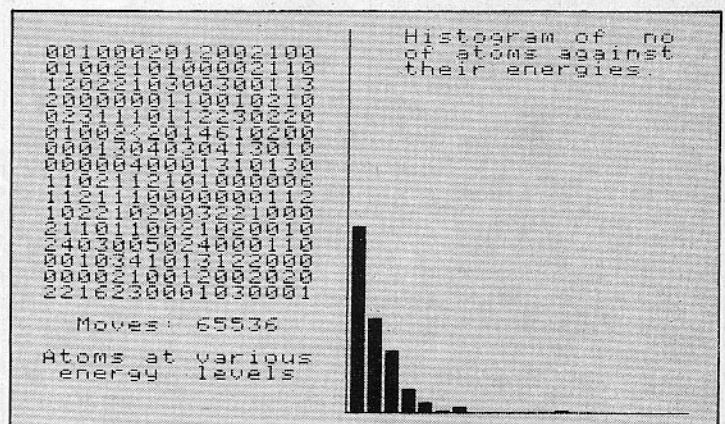


Figure IV: All atoms with one quantum simulation complete

# We've got a system - a complete new system..

**ANYONE who has connected an Apple to a colour TV or monitor will know that it has problems with colour. There are a number of technical reasons for this, most of which arise from the fact that the Apple has been around a long time now, and it adopted an approach that saved on memory at the cost of flexibility.**

As far as the user is concerned, the problems manifest themselves most visibly in hi-res graphics mode.

Text is blodgy, multi-hued and generally unreadable, colours only plot in either odd or even positions, adjacent colour areas spill over into each other or have coloured fringes, and coloured lines break up the more vertical they become.

In addition, the better the monitor used the more the coloured areas take on a stripey look. Generally, apart from games, I don't like using colour very much and my colour monitor remains switched off.

I have tried a number of colour cards in the past and some do improve the situation considerably - but the new Spectrogram card from Keyzone is streets ahead.

It isn't so much a colour card as a complete colour system for the Apple II and IIe. In fact I found it more entertaining than many games!

It is by far the largest and most complex colour card I've seen and has a variety of connector locations, switches

and leads which do different things depending on the type of RGB monitor used and whether you have an Apple II or IIe.

The card comes with a disc containing a host of demo and set up programs and a 64 page manual. The size of the manual gives some indication of the range of facilities offered.

The first task is to install the card, which turns out to be no simple matter. The cable from card to monitor ends in bare wires and you have to solder them to the appropriate connector for your monitor.

If you are unhappy about this you will have to get a friendly dealer - I'm sure there must be one - to do it for you. In fact you might want help over the whole installation procedure.

The next step is to decide what type of monitor you have - standard linear, high linear or TTL. If it's the latter then most of the card's special features won't work.

Since I didn't know which my Kaga monitor was I set up for standard linear as indicated and everything worked fine. The manual suggests what you expect to see if you've chosen the wrong one, so it isn't crucial to get it right first time.

Having set up the card and monitor we must now install the card in the Apple itself. This first requires soldering across two jumper locations, labelled X7, on the Apple motherboard. These are easy to locate and it is simple to do as long as you have a soldering iron.



The final step is to connect leads, with clip hook ends, to two of the pins on chip C14. I found this rather tricky to do even after I'd located the correct chip (the number is underneath it!) and I had to lever the chip up a bit to get the clips on properly.

I don't want to imply that the installation is impossibly difficult, but it is a bit more involved than simply plugging the card into slot 7. However by following the instructions carefully I had no problems and everything worked first time. The results are well worth the effort.

The card works in three main modes - the first is just a simple colour card. I found the Apple colours were bright and crisp with a good white in text mode. It even did a reasonable job on 80 column text, which is often very poor with cheaper cards.

The second mode produces just two colours on text and hi-res screens alike. This mode is accessed by simple POKES to three memory locations.

All text and graphics must be generated in black and white by your programs, but the card offers the facility to redefine black and white to any of the 16 lo-res Apple colours. This gives 240 useful colour combinations - although many of them shouldn't be looked at if you have a hangover (I particularly like violet on lime green).

The advantage of this for hi-res graphics is that it removes all the colour anomalies and fringing effects completely,

which in turn renders text and lines clear and sharp.

By choosing different pairs of colours on successive occasions a program can give plenty of colour to the visual display while keeping maximum clarity and resolution. I used this mode quite a lot - especially with Doublestuff (see last month's column) where colour problems can be even greater than usual.

The real fun occurs with the third mode of operation. This allows the user to redefine the Apple colours (16 lo-res, 6 hi-res) from a possible choice of 255 colours. The card does this by providing separate control of the red, green and blue components of each colour.

There are eight red, eight green and four blue levels. This offers a range of colours and shades from brilliantly garish to muted pastel.

Each of the 16/6 colours can be redefined by a simple POKE so a program can continually change the colour palette available. The user defined set can be switched out, and normal colours restored, at any time.

The disc contains several demonstration programs that make extensive use of this facility. The most striking are certainly the animation routines - Snow Demo and Sunset Demo. These use the redefinition capabilities to create simple animation.

The trick is to draw the moving parts in different HCOLORS at their various locations in the same picture.

The HCOLORS are then redefined to be the same as the background in all but one position. By successively redefining colours the different shapes appear or disappear instantly.

The technique is limited by the fact that there are only six hi-res colours to play with, but the shapes can be as complicated as you wish. The disc and manual contain simple animation programs to allow automatic setting up of such effects. I'm sure that with clever hi-res page switching the effects could be greatly enhanced.

The most important of the programs, however, is the Set Card option. This is a menu driven suite of programs that provide systematic procedures for altering the card parameters. The programs are well written, very simple to use, and adopt a common format in hi-res and lo-res.

In the Colour Chart options the user is presented with 16 (lo-res) or 6 (hi-res) blocks of colour and given the ability to alter all the features of the card interactively.

In this way the effects can be seen immediately and each colour can be worked on in turn. This can be quite addictive and I learnt a lot about mixing colours and how to get various subtle shades.

The settings can be saved to disc as a small User Preset program, although the resulting program didn't do anything on my version.

I soon found that the problem was that a variable RAM – used to enable the redefine model – had not been set, so it was easy to correct.

Another option allows one to use a normal hi-res picture instead of the coloured blocks when redefining the colours. This is wonderful if you want to change the colour scheme of your favourite games.

I now have a pastel version of Zaxxon which is much easier on the eyes, as well as another version – for those trying to beat my score – which is truly nauseous.

The card also has a mode which reduces the striped appearance of large coloured areas, albeit with a slight loss of fine detail, which I also like in

some games.

You don't have to do anything to the games disc itself to achieve the colour changes. All you do is redefine the colours before warm booting (PR#6) the game – so it doesn't matter if it is on a protected disc.

Finally there is a stereoscopic demo to provide 3D pictures on the Apple screen. This uses the

anaglyph technique of viewing two slightly displaced red and blue images of the object through red and blue glasses. Unfortunately they don't provide the glasses, and the ones I got were red and green.

The ability to redefine colours means that, in principle, one can compensate for this by using a red and green anaglyph, but in

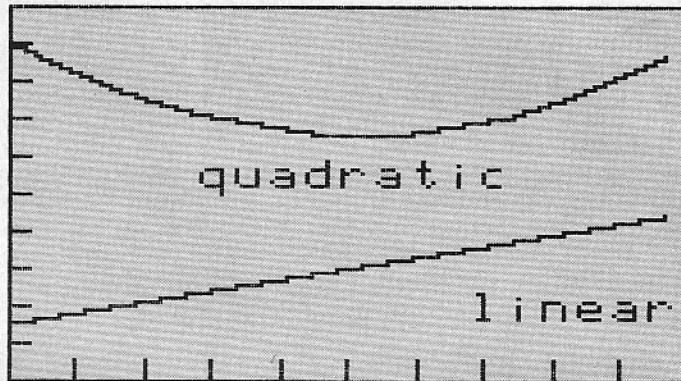
practice I wasn't too successful at it.

All in all this is the best colour card I have used on an Apple and if you are serious about using colour regularly it is well worth looking at. Nothing will ever overcome all the inherent colour problems on Apples but Keyzone have gone a long way towards it.

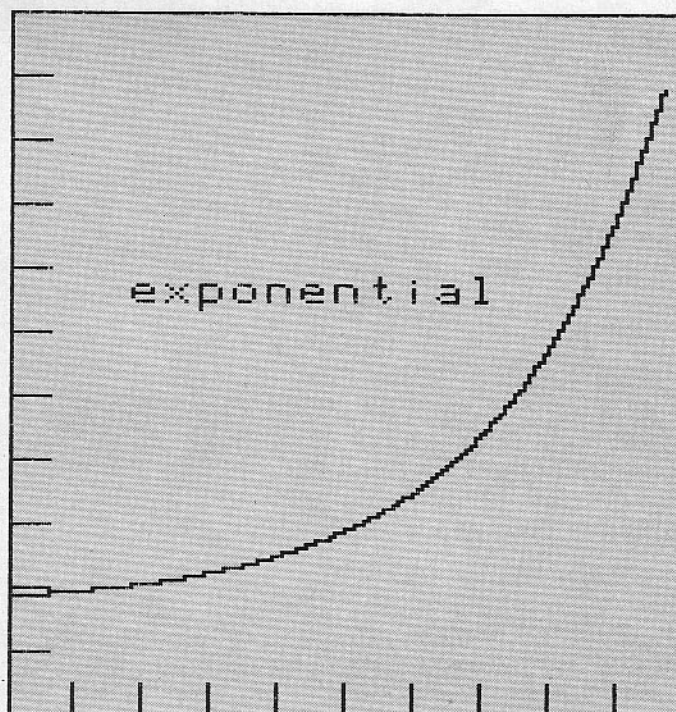
## Creating the Apple User Graphics Library

## Part VIII

# Maths graphs – with the minimum of effort



Above and below: Samples of output using three of the four functions provided



A REQUIREMENT of any graphs package is that it should be able to produce graphs of mathematical functions. This is as important to the business user with his trend analysis as the engineer with his stress calculations.

This month we add two routines to the Apple User library to achieve this with the minimum of effort.

If you have been following the series you will be familiar with the format. The routines should be typed in after the previous ones and the program set to run past hi-res Page 1.

The example program presented here uses several of the routines from previous months and, as the library expands, we can see how a few, relatively simple, program lines can produce a complex three graph display.

Before writing the routines I decided on several conditions to meet. These are:

- The function is in "user" units, so it can be typed in just as it would be written.
- All clipping at the edge of the graph window is automatic – if it goes off screen it just isn't plotted.
- A default setting automatically graphs the function over the range XMIN to XMAX.
- The default setting can be overridden to graph over a restricted range.
- New functions can be added with minimum of difficulty.
- All functions use a common

# GRAPHICS

technique to pass parameters.  
 □ Variable step size allows for coarse, but fast, lines or fine, but slow, ones.

As done previously we relegate most of the task to subroutines and use the main program just to set up the minimum of information necessary.

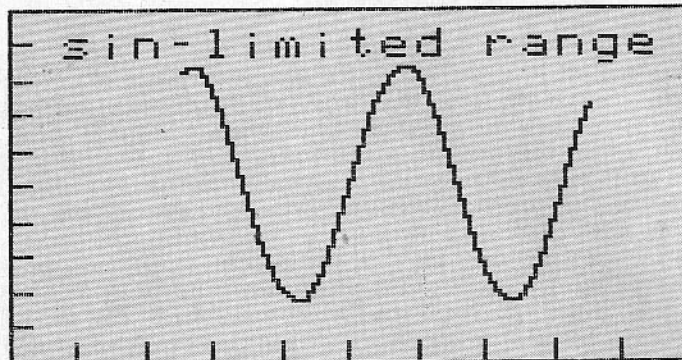
There are two routines here – the Function Plotter, and the Function Library. The first handles the plotting of the function itself, while the second is a library of useful functions.

By separating the task of plotting a function from the actual function code we can easily add new functions at any time. This approach is slower than having the code in the plotting loop itself, but it is very much easier to update.

The Function Plotter is controlled by an array ZF. Again the default size of 10 is used so it doesn't need dimensioning explicitly. It plots the function in the presently set HCOLOR and using the current graph window.

ZF(1) contains the library number of the function – that is, where it is in the library.

ZF(2) controls the speed of



Sample plot over a restricted range

the plotting. A value of 0 will plot the function every pixel. A value of 1 plots values every 5 pixels with straight lines in between, a value of 2 uses 10 pixel steps, and so on.

ZF(3) selects the range over which to graph the function. A value of 0 goes over the whole range. A value of 1 selects the restricted range option.

ZF(4),ZF(5) contain XMIN and XMAX for the restricted range – in user values. These are only used if ZF(3)=1 and providing they aren't outside the graph window.

Lines 180 and 190 in the example program show how to select speed=2 (10 pixel steps) and the full graph range. ZF(8)–ZF(10) are used internally

by the routine.

The Function Library contains the functions that can be plotted. It uses the value in ZF(1) to select the function and line 44270 directs the program to the correct place. When you add another function you must update this line to include it. A non-valid library number causes the routine to return with a value of zero.

All functions are treated in a similar manner and they all make use of an array ZA(0)–ZA(10) to pass information in. On entry the X value (user units, not screen position) is in ZA(0), on exit ZA(0) contains the Y value (user units also). The rest of the array ZA(1)–ZA(10) is available for

parameters.

For instance, let's consider function 1 – a general straight line. It has the form:

$$Y = a_1 + a_2 X$$

The parameters  $a_1$  and  $a_2$  might be decided by a calculation, statistical fit or some other means. All that is required is that their values be put in ZA(1) and ZA(2) before calling the function plotter routine.

The library contains four functions as examples – linear, quadratic, exponential and sine. A glance at these will show how easy it is to add your own function.

The example program plots all four library functions on three separate graphs. Each graph has different ranges and occupies a different screen area. The labelling is done using the machine code text routine from the August *Apple User*, but you could just as easily have used the earlier Basic version.

At present you would have to put numbers on the axes yourself, but in the not-too-distant future I shall present a routine to annotate the graphs automatically.

Peter Gorry

```

100 REM
EXAMPLE PROGRAM
110 GOSUB 42400: REM SHAPE TABLE LOADER
120 ZC = 3:ZP = 1:ZF = 1: GOSUB 40000: REM SET PAGE
130 ZM(1) = - 50:ZM(2) = 50:ZM(3) = - 50:ZM(4) = 50: REM USER VALUES
140 ZM(5) = 0:ZM(6) = 135:ZM(7) = 90:ZM(8) = 0: REM SCREEN VALUES
150 GOSUB 40200: REM SET MAPPINGS
160 ZB(1) = 10:ZB(2) = 10: GOSUB 40400: REM SET BORDER
170 REM PLOT FUNCTIONS
180 ZF(2) = 2: REM SPEED=2
190 ZF(3) = 0: REM FULL RANGE
200 ZA(1) = - 20:ZA(2) = 0.3: REM PARAMETERS
210 ZF(1) = 1: GOSUB 43800: REM LINEAR
220 ZA(1) = 15:ZA(2) = 0:ZA(3) = 0.01
230 ZF(1) = 2: GOSUB 43800: REM QUADRATIC
240 ZS(4) = 1: REM LOWERCASE TEXT
250 ZX = 20:ZY = - 30:ZS% = "LINEAR": GOSUB 43470
260 ZX = - 20:ZY = 5:ZS% = "QUADRATIC": GOSUB 43450
270 ZM(5) = 140:ZM(6) = 279:ZM(7) = 160:ZM(8) = 0: REM NEW PART OF SCREEN
280 ZM(1) = 0:ZM(2) = 10:ZM(3) = - 10:ZM(4) = 50
290 GOSUB 40200: GOSUB 40400: REM MAPPINGS AND BORDER
300 ZA(1) = 1:ZA(2) = 0.4: REM EXPONENTIAL PARAMETERS
310 ZF(1) = 3:ZF(2) = 0: GOSUB 43800: REM EXP SLOW SPEED
320 ZX = 2:ZY = 30:ZS% = "EXPONENTIAL": GOSUB 43450
330 ZM(1) = - 10:ZM(2) = 10:ZM(3) = - 1.5:ZM(4) = 1.5:ZM(5) = 0:ZM(6) = 135:ZM(7) = 180:ZM(8) = 95
340 GOSUB 40200: GOSUB 40400
350 ZA(1) = 1:ZA(2) = 1:ZA(3) = 0
: REM SIN PARAMETERS
360 ZF(3) = 1:ZF(4) = - 5:ZF(5) = 7: REM RANGE VALUES
370 ZF(1) = 4: GOSUB 43800: REM SIN
380 ZX = - 8:ZY = 1.2:ZS% = "SIN-LIMITED RANGE": GOSUB 43450
390 END :
43800 REM
FUNCTION PLOTTER
43810 REM PLOTS A FUNCTION ACCORDING TO CONDITIONS HELD IN ZF(1)–ZF(5)
43820 REM 1 FUNCTION LIBRARY NUMBER
43830 REM 2 SPEED 0=SLOWEST, 0 R N%5
43840 REM 3 RANGE 0=FULL, 1=SELECTED VALUES
43850 REM 4 XMIN FOR SELECTED RANGE
43860 REM 5 XMAX FOR SELECTED RANGE
    
```

```

43870 REM 8-10 USED INTERNALLY
43880 REM SET UP RANGE
43890 IF ZF(3) = 1 THEN GOTO 43
910
43900 ZF(9) = ZM(5):ZF(10) = ZM(6
); GOTO 43940: REM FULL RAN
GE
43910 ZF(9) = INT ( FN XCN(ZF(4)
) + .5):ZF(10) = INT ( FN X
CN(ZF(5)) + .5)
43920 IF ZF(9) < ZM(5) THEN ZF(9
) = ZM(5)
43930 IF ZF(10) > ZM(6) THEN ZF(
10) = ZM(6)
43940 ZF(8) = 1: IF ZF(2) > 0 THEN
ZF(8) = ZF(2) * 5: REM SPEE
D STEP SIZE
43950 ZQ = 0: REM PLOT FLAG TO Z
ERO
43960 FOR XP = ZF(9) TO ZF(10) STEP
ZF(8): REM PLOT FUNCTION
43970 ZA(0) = FN UXCN(XP): REM
USER X
43980 GOSUB 44200: REM FUNCTION
LIBRARY

```

```

43990 YP = FN YCN(ZA(0)): REM Y
SCREEN VALUE
44000 IF XP < ZM(5) OR XP > ZM(6
) OR YP > ZM(7) OR YP < ZM(8
) THEN ZQ = 0: GOTO 44030
44010 IF ZQ = 1 THEN H PLOT TO
XP,YP: GOTO 44030
44020 H PLOT XP,YP:ZQ = 1
44030 NEXT
44040 RETURN :
44200 REM

```

#### FUNCTION LIBRARY

```

44210 REM LIBRARY OF AVAILABLE
FUNCTIONS
44220 REM ON ENTRY THE FUNCTION
LIBRARY NUMBER MUST BE IN Z
F(1)
44230 REM AND THE USER X VALUE
IN ZA(0)
44240 REM THE ROUTINE PUTS THE
Y USER VALUE INTO ZA(0) ON E
XITING

```

```

44250 REM ZA(1)-ZA(10) ARE AVAI
LABLE FOR PARAMETERS
44260 ZX = ZA(0): REM USER X
44270 ON ZF(1) GOTO 44300,44330,
44360,44390
44280 ZA(0) = 0: RETURN : REM NO
T IN LIBRARY
44290 REM LINEAR - Y=A1+A2*X
44300 ZA(0) = ZA(1) + ZA(2) * ZX
44310 RETURN
44320 REM QUADRATIC - Y=A1+A2*X
+A3*X^2
44330 ZA(0) = ZA(1) + ZA(2) * ZX +
ZA(3) * ZX * ZX
44340 RETURN
44350 REM GENERAL EXP - Y=A1*EX
P(A2*X)
44360 ZA(0) = ZA(1) * EXP (ZA(2)
* ZX)
44370 RETURN
44380 REM GENERAL SINE - Y=A1*SI
N(A2*X+A3)
44390 ZA(0) = ZA(1) * SIN (ZA(2)
* ZX + ZA(3))
44800 RETURN :

```

## The Blackboard

The most advanced text/graphics printer card yet produced for the Apple II & IIe. 45 easy to use Text/Graphics commands makes 'The Blackboard' the most powerful printer card for the EPSON printer range and the Apple DMP.

#### Graphics Features

- \* Independent scaling of X & Y axis.
- \* Y axis by up to 8 times, X by up to 5.
- \* Special command to give 1:1 scaling on Epson FX series (print square squares and round circles!).
- \* Dump Page 1 Page 2 or 3 mixtures of both (AND, OR, EOR).
- \* Exclusive pre & post shading feature give grey background.
- \* Plus the usual features of Inverse, Rotated & double size.
- \* Over 700 different ways of printing just one HIRES screen.

#### Text Features

- \* Easily Set Left/Right Margin, line length, form length.
- \* Automatic word wrap.
- \* Exclusive option automatically substitutes F's for S's.
- \* Text screen dump works from any language.
- \* Fully supports Pascal, CP/M and all 80 column cards.
- \* Single, Double or no line spacing.
- \* Select international character sets and alternate fonts.

#### For Programmers

- \* Hex dump mode shows exactly what is being sent to the printer.
- \* Transparent mode lets any character through to the printer.
- \* RAM option lets you program The Blackboard yourself.
- \* Mask or set high bit.

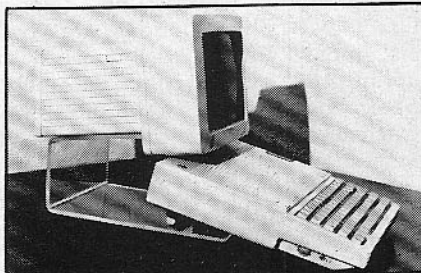
Price £99.00 plus VAT

**LEICESTER**  
computer centre limited

L.V.E. Building, 9 Jarrom Street, Leicester LE2 7DH  
Telephone: 0533 556268 - -

## NOW AVAILABLE! From Leicester Computer Centre Mailmerger for Appleworks™ written by Peter Meyer

Links together the word processing and database functions of Appleworks™ to produce customised letters and documents. Allows customising not only of name and address but also sections of sentences can be changed. Puts an extra dimension to the use of your Appleworks program. Just one of the many programs for your Apple from Leicester Computer Centre.



Only £49.95  
+ VAT  
+ £1 p&p  
Includes back up  
disk and full  
Reference/  
Tutorial  
Documentation

### Monitor stand for Apple IIc

The LCC stand for the IIc monitor places the monitor at a convenient height and angle, directly above the computer. Made from strong 6mm clear perspex.

£14.95 + £1.50 P&P, + VAT

**LEICESTER**  
computer centre limited

L.V.E. Building, 9 Jarrom Street, Leicester LE2 7DH  
Telephone: 0533 556268

Appleworks is a trade mark of Apple computer inc.

IDEALLY, when you connect a printer to an Apple II you should be able to put the interface card into an Apple slot, connect it to the printer, switch on the system and print whatever you want under whatever language system you want without problem.

Not too much to ask, is it?

However, in the past it has often been the case that the printer is perfectly controlled from within Basic but not from CP/M or Pascal without invoking patches to the disc or changes to the card.

I put the Blackboard into my Apple II+, connected it to an old Epson MX-80F/T Mk II - fitted with the Graftrax ROM so that it behaves much like a Mk III - and quickly printed text under DOS 3.3 (from Basic and Applewriter II), under Pascal, under CP/M, and under Metacraft's Forth. Each system worked faultlessly.

That was enough to endear the Blackboard to me. There is, however, much more.

The Blackboard can truly be called a very intelligent printer card. Different versions know how to control the Epson series of printers, the Apple DMP and the Taxan Kaga. Specify your printer when buying the card.

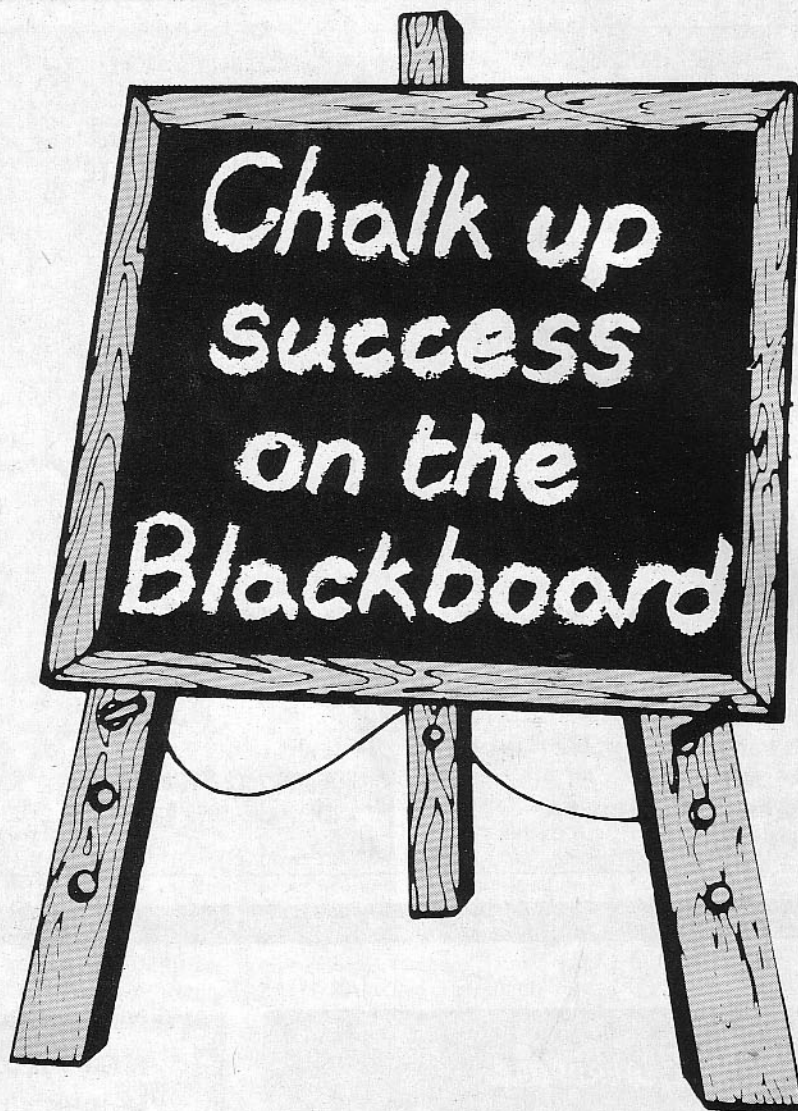
Over the last few years printer cards have grown more intelligent so that now one expects a fair amount of firmware to be on board.

A protocol has grown up which says that the card will be in slot 1 and that Ctrl-I will be interpreted as the start of an interface command string if the card has been selected as the output stream.

Thus, for example, most parallel cards will print 40 columns of characters as the default and can be switched to allow 80 by issuing the command Ctrl-I "80N".

Similarly, if the right printer is connected, several cards will give a screen dump of the Apple's hi-res page 1 by issuing the command Ctrl-I "G".

The Blackboard follows the



By MAX PARROTT

same conventions and uses the same sets of commands as other intelligent printer cards so that most software should cope admirably without change.

However, more complex graphics options are available. Pages 1 and 2 can be printed overlaid one on the other (OR'ed together), complemented (XOR'ed together) or printed such that only common information appears on paper (AND'ed together).

Further, the X scale can be expanded on printing to give x1, x2, x3, x4 and x5 the normal scale and independently the Y scale may be set to x1, x2

x2 $\frac{2}{3}$ , x4 and x8 the normal.

Graphics dumps can also be rotated and printed in double density or in inverse-useful for pictures - or shaded.

During the last couple of months of testing I have used several combinations of the graphics options without major problems. On the FX series of Epson printers the squashed circle appearance is easily overcome with the Blackboard but of course it cannot be overcome, merely minimised, with the older Epsoms.

Graphics dumps were obtained from Basic and Pascal. A Pascal example program is

given in the manual but more interestingly, Blackboard commands can be issued interactively by using Ctrl-Y in place of Ctrl-I.

A normal graphics dump takes 45 seconds, which compares favourably with other cards. I found that under the control of a Basic program, a carriage return had to be sent immediately after a D\$ "PR # 1" to make sure that the graphics dumps occurred.

The manual also says that the graphics dumps can be centred by using the command to set the left margin, but with my set-up this would not work.

Users of Apple's Business Graphics package will be relieved to hear that for £20 a driver package is available from Leicester Computer Centre which takes full advantage of the Blackboard's graphics capabilities.

Besides the graphics utilities, there is a host of other simple commands which will

The Blackboard makes bit image printing easy....

Bit

markings, and create piano reductions.

A special title page formatter gives finished manuscripts a polished look.

Professional Composer saves and files all pieces, whether sketches or symphonies. It costs £429.

● *P & P Micro Distributors Limited, Todd Hall Road, Carrs Industrial Estate, Haslingden, Rossendale, Lancashire, BB4 5HU. Tel: 0706-217744.*

## Working out the odds

FOR managers who want to predict the future comes a software package which is claimed to help you to at least see ahead – and it runs on the Apple II.

The package comes in two parts, available separately. The first program is General Appli-

cation Simulation System, or Gass.

It simulates 10 variables and combines them into one user defined algorithm.

Variables can be random, Boolean, functional or compound. Random variables can be any of 13 different probability distributions, including empirical tables.

It is said to be ideal for things like projecting group health insurance costs.

The second program, Monte Carlo Simulations, as well as simulation, provides statistical analysis. Seven different probability distributions are offered.

The program also works as a front-end to Gass by analysing past experience. It can also, say the makers, enhance the Gass results with extra graphics and reports.

● *Actuarial Micro Software, 3915A Valley Court, Winston-Salem, NC 27106, USA. Tel: 0101 919 765 5588.*

## Ile crib sheet

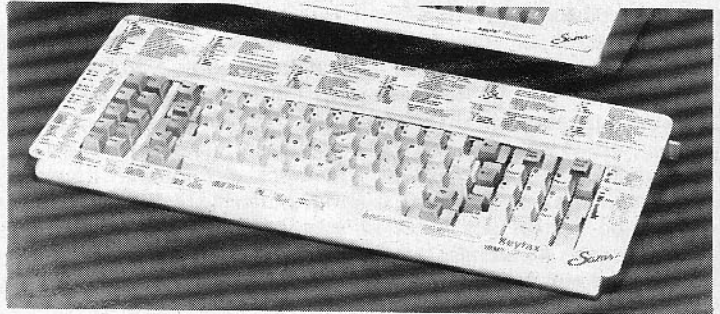
ITT subsidiary Howard W. Sams has produced Keyfax for users of Apple IIe and IBM PC micros.

It provides quick reference data about the control commands and other vocabulary needed for operations with word processors, spreadsheets, DOS and Basic.

It is a two-sided template that fits round the keyboard. Side one displays operating commands for many software packages, while side two provides the main commands for Apple DOS 3.3.

Templates for the Apple include Applesoft Basic (both sides), Applewriter IIe/DOS 3.3, Multiplan/DOS 3.3, ProDOS (both sides) and VisiCalc/DOS 3.3.

● *P&P Micro Distributors, New Hall Hey Road, Rossendale, Lancs. Tel: 0706 217744.*



Keyfax on the Apple IIe

Small Systems Engineering are innovators in the field of microcomputer technology and enhancement products. We now offer the first multi-user hard disk system for the Macintosh.

# KING TOGETHER

Apple Mac users can now share resources and share costs.

The Sunol Winchester disk drive from Small Systems adds 8, 16, 25, 40, 65 or 92 Mbytes of usable storage capacity to the Mac. The system operates with existing programs, appearing to the Mac as a high speed, high capacity floppy drive.

To achieve compatibility with the Mac, we have developed an interface implementing Apple's APPLEBUS\* networking protocol. This allows up to 31 Macs spread out over a building to share a drive at distances of up to 1000', with each user having separate secure work areas.

Interfaces are available to allow not only Macs but also most other popular micros to share the same drive. Micros currently supported include Apple II and III, IBM PC and compatibles, Epson QX, TI, Victor/Sirius, Future, Zenith and Apricot.

For further information on the Sunol range of Winchester disk drives, contact your local Apple dealer or phone Jane Leather or Simon McCarthy on 328 7145, or just fill in the coupon.

**small systems engineering**



**PUTTING WISDOM TO WORK**

Name \_\_\_\_\_ Position \_\_\_\_\_

Company \_\_\_\_\_ (type of business) \_\_\_\_\_

Address \_\_\_\_\_

\_\_\_\_\_ Telephone Number \_\_\_\_\_

If you are interested in distribution opportunities please tick box.

**Small Systems Engineering Limited,**

2-4 Canfield Place, London NW6 3BT Telephone: 01-328 7145. Telex: 264538. (AU1)



## Bubbling over with memories

XBUB is a bubble memory module for the Apple.

The unit, which plugs into any of the Apple slots, is compatible with the Apple II+ and IIe and will run Applesoft, Pascal and CP/M. It is distributed by Xcalibur.

Xbub is used as a conventional disc unit within the Apple where it provides 1mbyte of non-volatile, fast access storage, either in conjunction with conventional floppy discs or as a stand-alone unit.

It has advantages for areas with dust, grime, grease or vibration.

Being non-volatile means that the unit may auto-boot at switch-on and then go on to run using programs and data which have been pre-recorded.

● Xcalibur Computers, Spencer House, 3 Spencer Parade, Northampton NN1 5AB. Tel: 0604-21051.

## Joystick calculating

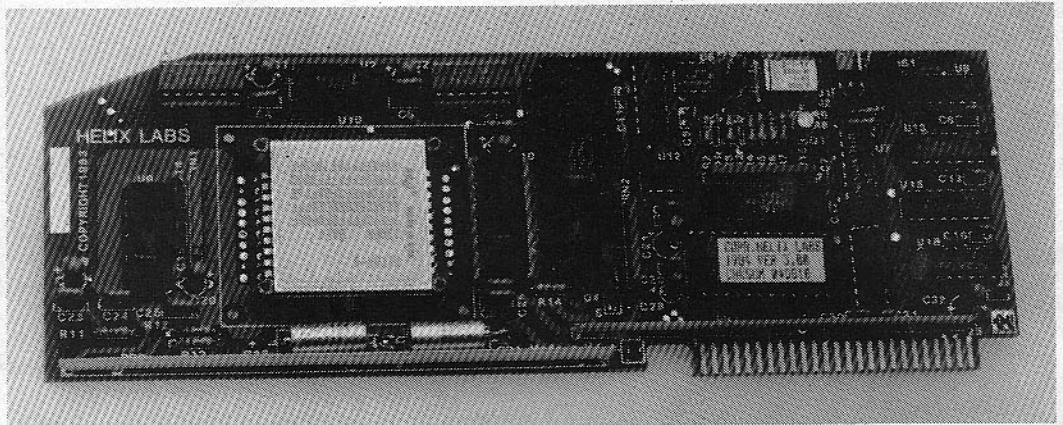
VARICALC — a program for the Apple II, designed to solve engineering, scientific and business equations is announced by Heyden Datasystems.

A novel feature is the use of variators to change variables interactively. The variators may be game paddles, a joystick or the keyboard's arrow keys.

The variator may also be an automated loop with selectable range and step size.

Further, the variator may be the voltage output from a process or instrument. This is interfaced using Heyden's Adalab data acquisition interface card.

Any variable can be used to output a real-time voltage using



Xcalibur's Xbub

the Adalab interface.

Varicalc will simulate complex physical, chemical or mathematical processes, accept real-time voltage input directly into a pre-defined model and solve equations of up to 19 variables.

The system will also plot results as on-screen graphics or on a dot matrix printer. Up to 255 equations may be sorted on disc for quick recall.

● Heyden Datasystems, Spectrum House, Hillview Gardens, London NW4 2JQ. Tel: 01-203 5171.

## ProDOS clock

A clock-calendar card to work with Apple's new disc operating system, ProDOS, is now avail-

able from Apple dealers.

Proclock, manufactured by Practical Peripherals, also works with DOS 3.3, Apple Pascal and CP/M.

While some users may wish to write their own programs for Proclock, there are many applications which will automatically use a clock as soon as it is installed in the micro.

One enhancement to ProDOS is the ability to time and date-stamp files. Whenever a ProDOS disc is catalogued, the time of creation and last modification of each file is displayed.

For DOS 3.3 users, Proclock includes an enhancement called Super-DOS which provides a similar time and date stamping of files.

By running a simple installation program, Super-DOS can be added to existing discs without changing any files.

In addition, the Proclock utilities disc provides various routines for using Proclock in Pascal, as well as automatic updating of the filer.

Proclock emulates three other clock/calendars so it can be used with programs written for these products.

They include Apple Clock from Mountain Computer, Superclock II from West Side Electronics, and Thunderclock Plus from Thunderware.

For those wishing to write their own programs, Proclock is easy to use in Applesoft, Pascal, CP/M, and machine language.

Aside from its usual date/time function, Proclock can also generate precise interrupts for foreground/background programming. Several examples of this feature are supplied on the Proclock utilities disc.

The disc also contains many demonstration programs.

## MAC IN MIND

DEVERILL Business Systems have produced a personnel and payroll application for Macintosh — dbSPay.

It incorporates all the usual payroll system features for £125.

● Deverill Business Systems, 3 Luscombe Road, Poole, Dorset BH14 8ST. Tel: 0202-741391.

□ □ □

MACTOOLS puts the power of the Macintosh system into a disc utility package.

The system integrates many of the standard disc functions into one program, as well as

adding several new features such as copy protect/unprotect, lock/unlock files, verify a disc and make files visible/invisible.

It will also perform the more common disc manipulations with ease, such as copy files or discs, rename files or discs, delete files or format a disc.

Pattern matching has been added to make working with several files simple.

Everything that begins with "Mac" can be copied, as can those files that contain the letters "Font".

● Central Point Software, 9700 SW

Capitol Hwy, Suite 100, Portland OR 97219. Tel: (503) 244-5782.

□ □ □

AN appointment calendar and address book for the Macintosh has been produced by Rodent Software.

The Personal Appointment Diary allows all 12 months to be open at once and utilises the flexible Macintosh cut/copy/paste technique for duplicating, moving, changing and removing entries.

● Rodent Software Limited, The Shanty, Cores End Road, Bourne End, Bucks, SL8 5AL. Tel: 06285-26784.

utilities, and a useful job-time logging program called Time-Clock II.

Proclock features a low-power design, including a 10 year continuous lithium battery so the clock will keep running even if the system is left off for long periods.

● *P&P Micro Distributors, Todd Hall Road, Carrs Industrial Estate, Haslingden, Rossendale, Lancs. Tel: 0706 217744.*

## Accountant with AI

AN integrated business system incorporating accounting, financial reporting, spreadsheet forecasting, job costing, stock control, modelling and payroll functions has been launched by Cybersoft Intelligent Systems.

Artificial intelligence techniques have been used in the product, which is spread over only three discs and is said to overcome the disadvantages of modular software.

Called The Accountant, the new package can process batches of information independently of the operator. Once the day's transactions have been keyed in the machine can be left alone to work on pre-arranged tasks.

Running on the Apple, it has a capacity of 1,000 accounts and costs £800.

● *Reliability Engineering, 24 St. Margarets Street, Canterbury, Kent CT1 2TH. Tel: 0227-60456.*

## Industrial interface

AN integrated industrial and laboratory interface from Digital Design and Development allows the Apple to interact with the real physical world.

The General Purpose Interface System enables scientists and engineers to turn their micros into intelligent data monitors and closed loop controllers.

Sampling rates of 28KHz are

possible on the analog input side. Analog voltages may be generated at 25KHz.

All analog inputs are fully differential and overload protected up to 30 volts. In addition to eight analog inputs and four analog outputs, there are two eight bit digital ports.

Eight separate on/off digital lines are available for controlling valves, motors, relays, etc. Each digital output is capable of delivering 400mA at 50 volts.

Digital inputs allow the micro to sense contact closures, microswitches or read binary and BCD data. Price: £700.

● *Digital Design and Development, 18/19 Warren Street, London W1P 5DB. Tel: 01-387 7388.*

## Statistics handler

A STATISTICS package for the Apple II and IIe with CP/M card and hard disc called Abstat is being launched by Gil Anthony Systems.

Abstat facilities include integral help, full data entry and editing from keyboard, data transformation by user defined algebraic expressions and conditional data set creation.

The report generator supports text inclusion and the package allows multiple variable data sorting.

One and two way analysis of variance, chi-square, (r) matrix, standard deviation, skewness and kurtosis, U-test and t-tests, simple and multiple regression and Spearman rank matrix and

cross tabulation are all catered for.

Abstat needs 56k of memory and 280k of disc storage.

● *Gil Anthony Systems, Shepherds, Tydehams, Newbury RG14 6JT. Tel: (0635) 35831.*

## Business sub-system

QUEST International is the distributor for the new Ampex Pyxis 5¼" Winchester sub-system for a wide range of business micros including Apple.

Installation is said to be simple.

The Pyxis offers fast data access time and a data transfer rate of five megabits/second.

It is available with 5, 10 or 20mbytes formatted capacity which means that even the smallest version can replace up to 40 floppy discs.

The capacity can be increased by the addition of higher capacity discs or up to two disc drives.

● *Quest, School Lane, Chamber's Fold, Hampshire SO5 3YY. Tel: (04215) 66321.*

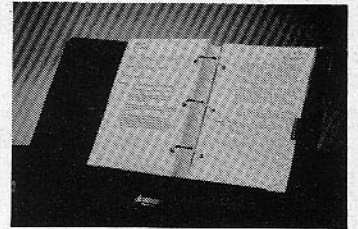
## Easy reader

A MANUAL easel that securely holds all standard sizes of micro manuals and books at a convenient reading angle has

been produced by Inmac.

The easel is made of easy-to-clean bronze-tinted acrylic and has non-slip rubber feet to prevent marking furniture.

● *Inmac (UK) Limited, Davy Road, Astmoor, Runcorn, Cheshire WA7 1PZ. Tel: 09285-67551.*



Easel from Inmac

## Serial Grappler

CALIFORNIA-based Orange Micro has launched a serial version of its Grappler printer interface for the Apple, which has sold 170,000 units since 1981.

The new Serial Grappler features over 25 firmware commands for control of screen text and graphics. These include double hi-res graphics, to access the Apple IIe's highest resolution screen, and an 80 column text screen dump.

There is also a new transparency command, allowing users to bypass all firmware when using software packages that don't work well with the Grappler's logic.

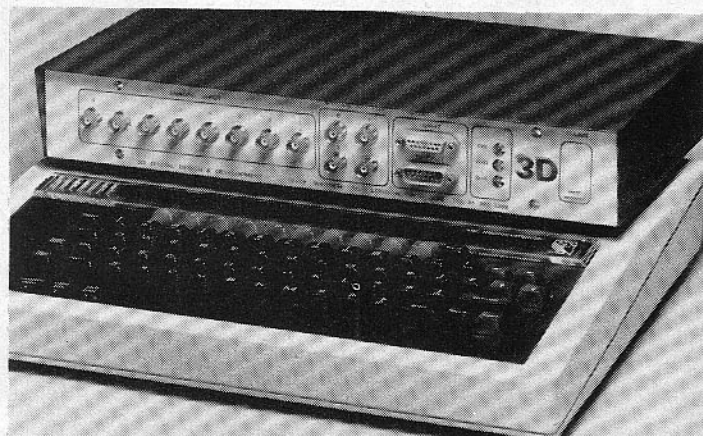
The Serial Grappler is claimed to be the first Apple interface capable of accepting an extra buffer module without using an additional slot.

An optional add-on buffering unit will be available later this year, providing from 16k to 64k of printer memory, or five to 20 pages of text.

The Bufferpak plugs directly onto the Serial Grappler - no cables, no jumpers, no lost slots.

The interface links the Imagewriter and the Apple II, II+ or IIe. It is Grappler+ compatible, so the hundreds of software packages written for the Grappler+ will work.

● *Orange Micro, Inc., 1400 North Lakeview Avenue, Anaheim, CA 92807. Tel: (714) 779-2772.*



Laboratory interface by DDD

A HALF height 5¼in floppy disc drive finished in typical Apple beige with black front, red LED, and black rubber feet, the C/WP Waterbell drive looks good sat by the side of the Apple II or IIe.

It is slightly longer than an Apple drive, measuring 10 x 6 x 2¼in and is slightly darker than the new IIe in colour.

I first connected it to an Apple disc controller card as the first (boot) drive and used it on a II Plus for several days, during the course of which I ran DOS 3.3, Pascal and CP/M programs, together with a variety of protected discs. Each disc behaved faultlessly.

For several days I used it as the second drive to an Apple drive and had no problems with reading and writing to disc.

During this time my only problems were in getting used to the drive's door. It is not like the typical Apple drive which folds down but is a 1¾in bar, projecting about ¼in from the front, which is pressed down to engage the head.

When pressed backwards it operates a sprung lever to eject the disc and lift the head, and this is what gave me the trouble.

I kept pressing the door inwards rather than down when inserting a disc, which promptly self ejected.

I found that the drive has to be at approximately the same height as one's hand, or above it, otherwise it is quite difficult

# It's faultless.. once you're past the door

**MAX PARROTT looks inside the C/WP Waterbell disc drive**

to fully push the disc home.

There was also one other problem. I like to issue a PR#6 command and then close the drive door after the head has stopped banging. This is quite difficult to do with the sliding bar arrangement.

When confident that the drive was working well I decided to see how good it was constructed. It came apart in much the same way as the Apple version. Undo four Philips bolts on the underside and the outer case slides forward to expose the innards.

A small PCB is mounted behind the mechanical parts on four plastic mounts. The cable was securely clamped to the back wall and ended in a socket.

The board bears four chips, a number of discrete components and several holes for testing

posts — which had not been installed.

Three more bolts held the mechanism to the outer case. Undoing these exposed another small PCB. The drive speed adjuster was underneath, so the whole would have to be dismantled to make adjustments. The mechanism and motors were all Japanese, from the Alps Electric Co.

There are three main differences between the Waterbell and Apple drives. First, already mentioned is the device to hold the disc in place. As a disc is inserted it pushes back a sprung metallic lever until it engages with a small peg on the floor of the mechanism which prevents it pushing forward.

Pressing down the bar at the front of the drive allows another small lever to engage with the

side of the sprung lever. When the front bar is pushed in to eject the disc the small lever lifts the sprung lever clear of the peg, allowing it to push the disc out.

This arrangement appeared robust enough, but only time will tell how well it will last.

The second difference is that the presence or absence of the write-protect notch is not detected by a micro switch but by an optical arrangement. Out of curiosity I protected a disc with a piece of clear Sellotape and found that, as expected, the disc had not been write protected.

The third difference is that the read/write head is not moved on its carriage by a peg engaged in a spiral track as on the Apple drives but by a metallic band clamped directly to the head mechanism and to a pulley wheel fitted to the spindle of the stepping motor.

The band is kept taught by another pulley fitted in a sprung mounting. Whether this arrangement gives a better or worse registration I don't know, but the drive certainly behaved perfectly in the time I had it.

Overall this drive appears good value. It would be interesting to test it on an Apple IIc.

**Product:** Waterbell disc drive  
**Price:** £159 including VAT  
**Distributor:** C/WP Computers, Willow Place, London SW1P 1JH. Tel: 01-828 9000.

## Appletips

### Follow the logic

Have you ever written a large program, too large in fact, and then used a utility to compact it and remove remark statements, only to discover that your program no longer works because lines which are gone to are now missing? I have. So I solved this

problem by always putting the dummy instruction called LABEL=LABEL at the start of any line which is the object of a GOTO or GOSUB statement.

For example, if I enter:  
 120 IF A>12 THEN GOTO 190  
 I immediately enter the line:  
 190 LA=LA: REM CONTROL COMES HERE FROM 120

This also helps you to follow the logic of the program at a later date.  
 Marie Donegan

AN Appletip from Graham Shields (August 1983 Windfall) on accessing the mini-assembler from Applesoft, works but my Applesoft program in memory is of course lost.

There are occasions when in the middle of developing a program it would be useful to access the mini-assembler without loss of program.

At the risk of being a bore, once again Derek Turner's lovely ampersand DOS modification can be useful. (See Windfall April 1983 page 21; August 1983 page 29.)

Place at \$BCDF:AD 83 C0 4C 66 F6 which simply selects RAM read

### Once again Derek has the answer

and turns on assembler. Also place at \$9E76:4C DF BC as usual.

INIT your new disc with this resident and on reboot (make sure INTBASIC is on this disc) the & call gets you to mini-assembler, which may be used as normal.

To exit, Reset or !\$FF69 and 300G, which returns you to Applesoft.

J. Taylor

# Mac the magnificent ...

CONGRATULATIONS Apple on creating, in Macintosh, a really first class machine. It must be the greatest single step forward yet towards overcoming the élitist mystique that still pervades many areas of computer applications.

No matter how much the ubiquity of computers is suggested there are still a great many individuals, in all walks of life, that have a poor understanding of computers, if not a downright apprehension of them.

Having been presented with a Macintosh, won by my father in Apple's own competition, I have wasted no time in exploring and exploiting its many

features — while patiently awaiting the release of Mac-Pascal.

In this period I have also observed naïve computer users "at play" with MacPaint, and found that as their confidence grew in producing simple drawings they would actively seek out the more sophisticated features to extend their creativity.

Having become hooked at that level they invariably wanted to know "What else can it do?" Trepidation gave way to curiosity and frequently amazement.

I could not help but think of the shock in store for them if they ever encountered the

infamous "A>", and a blank screen.

I am quite sure that already Mac users will be finding short cuts and useful tricks that their machines can perform. I hope that they will pass them on with the same enthusiasm that has been displayed by Apple II users throughout its esteemed life.

To be going with, I would like to suggest one little trick that I use within MacPaint. Macpainters may have found the rubber to be too large for some fine erasing, in a confined corner, for example.

One solution is, of course, to go into "Fat Bits" and remove the offending pixels one by one. The method I tend to employ is

to select a paintbrush of appropriate size and to paint over the unwanted area in white — instant erasure, on what ever scale you want!

The technique can be applied with any background, in fact, just paint over the extraneous material in the current background pattern and the unwanted lines disappear.

The versatility of Macintosh continues to impress me, and I strongly recommend anyone considering the purchase of a 16/32 bit machine to put the Mac high on their list.

With high resolution colour facilities I believe that it would be truly unbeatable. — **Peter V. Marsh, Sittingbourne, Kent.**

## ... or Mac the paltry?

MR P. Knight (Feedback, Apple User, September 1984) is not alone in having serious doubts about the usability of the Mac. Although 128k sounds big with Microsoft Basic there's a paltry 13k available for programs.

The IIe with Microsoft Basic and CP/M has 35k available.

The Mac operating system gobbles up space for itself, for the hi-res graphics, for the icons and for fonts etc. You can expect up to 50 disc swaps (and 20

minutes) to copy a disc with the built-in drive. You need a Lisa to develop programs.

Software houses have found themselves able to write bigger, more powerful and more competitive software for the IBM PC than was possible before. The small available memory for the Mac means a retrograde step and is why people such as Lotus have presently dropped support for it.

In fact the consensus is don't

buy a Mac without 512k, a second drive and a printer — unless you just want to design pretty magazine covers.

Volume supplies of 256k chips are not now expected before mid-1985 (later than originally forecast). The only way that Apple can stop a Macflop is to design an intermediate version of 256k using 64k RAMs.

There is no slot or expansion bus like the Apple II that can be

used to plug in memory boards.

Finally I want to commend your magazine for allowing a free discussion of these problems to come to light. It had looked as if you were part of Apple's sales department rather than an independent magazine with readers' interests at heart. — **B.E. Stevens, Altrincham, Cheshire.**

● You'll be glad to know the 512k Big Mac has now arrived. See story on Page 5.

## Selling point abandoned?

I READ your article on Appleworks in your May 1984 Apple User with some misgivings. When I bought my computer the Apple II Europlus, some years ago one of the strong selling points that Apple used was that they had a commitment to try and upgrade their computers by expansion cards and other means so that the individual could keep up with the trends by small additions to the system rather than through a throw away

repurchase philosophy.

This type of sales approach was very effective to a relatively hard-up personal user and indeed was followed for many years, making the Apple one of the foremost individual's personal computer.

In the article on what is obviously a very effective program, it states that Appleworks will not operate on any model before the IIe. Surely an expansion card or cards could have been designed to allow this program to run on earlier models of the Apple. Perhaps some of the expansion card producers should put their minds to this as there must still

be a lot of older Apples out there.

What concerns me is the attitude that comes through that if you have an older Apple, bad luck! If this is the new trend in Apple thinking, should I ever get enough together to buy a new personal computer, it would not necessarily be one of the new generation Apples, as I would be concerned that a similar philosophy would render it obsolete fairly rapidly.

I would like to point out that some of the success of the Volkswagen car was that it essentially maintained its form for many years with only minor modifications and did not

become regularly dated.

Apple should still look after its older devotees. Hell hath no fury like a computer owner scorned. — **A.L. Lawrence, Quellerina, Florida.**

● Many people now have doubts about just how "compatible" the Apple II family is. For example, although Quickfile was developed on a II Plus the version available will only run on a IIe.

Even Appleworks has separate versions for the IIe and IIc. This point was made to Apple at the Apple '84 Forum in Slough, but their defence was not convincing enough to reprint here.

## Protecting programs

I HAVE nearly finished writing a program in Applesoft which is long (over 100 blocks), and I wish to know how to protect the disc from being copied using COPYA, and also how to stop someone from changing and viewing the code.

Where can I find more info? — **Tim Stevenson, Harringay, London.**

● There is no easy answer to your problem. To stop people looking at your program in memory you will have to not let them break out of it — change RESET vectors and use ONERR GOTO etc.

You could possibly encode the program. There have been a few magazine articles on this subject, one in Apple User, April 1984. This is a bit simple to crack — we plan to publish some more information and a much harder to crack routine.

Most information on protecting the disc is available in Beneath Apple DOS and the manual which comes with the CIA Files. There has been sporadic information in some American magazines, notably Creative Computing and Nibble.

But do you really need to protect the program? To my mind a protected program for which I pay money is something to be suspicious of. If you are not selling the program, is there really any need to protect it?

**Max Parrott**

make one and the way in which this was done.

I have written such a program and it also includes a number of even more complicated moves to decode it. I can safely say that it is almost, if not, impossible to be decoded without using the key to it.

This key can vary from program to program and the data can also be written to disc as 4&4 encoded data (as against 6&2/5&3).

For people who have already typed the coding program in I have a suggestion. Instead of EOR twice ROR and ROL. This would make it a lot harder to decode than the present one as you are not restricted to just one ROR and ROL (for encoding and decoding the amount of ROR must equal the amount of ROL for example ROR ROR ROR (encode) ;ROL ROL ROL (decode)).

This may mean typing some of the code in again but not very much. — **Giles Gamon, East Molesey, Surrey.**

## Mixing CHAR and string variables

I HAVE read with much interest Gordon Findlay's Pascal Tutorial. In May's edition of Apple User it is stated that there is no way of mixing CHAR and string variables. In fact UCSD Pascal does provide a simple method of moving CHARs into and out of strings.

Pascal allows the individual characters of a string to be referenced in the following way:

```
stringname[i]
```

where 'i' is a number not greater than the string length. This allows direct access to the 'i'th character in the string:

```
VAR STRNG: STRING;
    X: INTEGER;
STRNG := 'Hello';
FOR X := 1 TO LENGTH(STRNG) DO WRITELN(S
TRNG[X]);
```

The above program fragment will print 'Hello' down the left side of the screen. This is mentioned in the Pascal

language manual, but what is not said is that these characters extracted from the string are not strings themselves but are CHARs.

This system can be used to place CHARs into strings provided that the CHAR is replacing an existing character in the string since the index number can not be greater than the string length.

To add a CHAR on to the end of a string the following code could be used:

```
VAR STRNG: STRING;
    CH: CHAR;
STRNG:=CONCAT(STRNG,' ') Add a space on to
STRNG's contents
STRNG[LENGTH(STRNG)] := CH Swap the space for the
character to be added

So Mr Findlay's letter checker (Page 54, May) becomes:
VAR CH: CHAR;
    INPUT: STRING;
    MATCH: STRING[1];
MATCH := ' ' Initialise match string
FOR CH := 'A' TO 'Z' DO Match for upper case
alphabet

BEGIN
    MATCH [1] := CH Move character into
match string
IF POS(MATCH, INPUT) > 0 We've a match
THEN MATCH_FOUND
END
```

Did anyone notice that the illustration of the Apple IIc on page 3 of May's issue was the wrong way round?

On page 66 (Feedback) of the same issue Max Parrott asks whether the ITT has DRAW and XDRAW. The answer is yes. DRAW and XDRAW seem to work in exactly the same way as the Applesoft commands (except that the hi-res screen is larger). — **Dave Miller, Finchley, London.**

## Bootless

I WONDER if you can help me. I have an Apple IIe which, at the

gramophone record collection (2,600 discs) and try for a business program to run the pub stock.

However, four of my games (NOT pirates) have suddenly started not to boot.

I went to Apple '84 at Slough — a good day out — and subsequently was advised that my Apple disc drive might be misaligned.

I sent it to Brit-teck (Apple User, July 84, Page 26) and paid £30 to have this done. The

games in question are:

Old Ironsides and Stickybear Bop (Optimum Resource), Spectre (Datamost), Type Attack (Sirius).

I can't send them back as they all did work but they still won't boot. Have you any suggestions as I am worried about starting a program of my own in case the same thing happens. — **Dave Parker, Wingrove, Bucks.**

● It is possible that your faulty drive wrote to the discs in question and destroyed the data on the tracks used in the booting process. Presumably this shouldn't happen again as the drive is now okay.

I think that these discs should be replaced by the software houses concerned because they made them uncopyable in the first place.

Any reputable software house will replace them — unfortunately usually for a small fee and the return of the faulty disc. **Max Parrott**

## Make life difficult . .

I READ Jonathan Lewis's encoding program and found that it was very easy to crack as he was only using EOR to encode the data.

As we all know this only allows 256 different combinations, two being rendered useless (0 and the encoding byte).

Why not change this one byte into two and so that any onlooker trying to decode it will have to know that two bytes

moment, is used for games, but I am hoping to get a second disc drive and some software to run a huge file program for my

## Shifted letter

I RECENTLY received my lower case generator and adaptor board for my very early Apple II.

I was very interested in the article by J.P. Lewis in the October 1982 edition of *Windfall*, entitled "Lowercase on power-up".

I use a suitably modified Applesoft in my 128k RAMcard.

I have got around the problem mentioned in the article regarding a shifted letter (such as \$) by reversing the Ctrl and Shift key connections to the 558 IC.

This has an added advantage for me in that this configuration follows the convention used in the Zardax program for shifting between capitals and lower case. — **D.S. Gallanders, Hong Kong.**

## Slotting in graphics

FIRST let me congratulate you on the much improved format of *Apple User*, which seems to get better month by month. Keep up the good work!

I recently took the opportunity to catch up on a few back issues missed while I was

abroad and came across the problem of a couple of your readers in the excellent *Feedback* pages of the May edition.

May I offer an alternative solution to those outlined by Max Parrott and Peter Gorry in their detailed and concise answer to the difficulty described by I.G. Dalgleish and Terence Rowbottom?

Both readers were faced with a problem involving the insertion of graphics within documents produced by their word processors.

One of the facilities offered by my company's product, *Snapshot*, is the ability to interrupt a running program, dump the contents of the screen to a printer and then resume running the program from the exact point of interruption.

I personally make use of this ability virtually every day to insert graphs and charts generated by *Apple Business Graphics*, under *Pascal*, into reports written on *Wordstar*, under *CP/M*.

There's no reason why your readers could not do the same with their own graphics packages and word processors using any peripheral card which generates a hardware interrupt (*Snapshot*, *Wildcard* + and some printer cards).

When you arrive at that point in your document where you wish to insert some graphics, use the w/p commands which

designate page length or "kill" the bottom margin, save it and print it.

Now use your graphics software to generate the screen you want to print and dump it to your printer with *Snapshot*.

Then you simply return to your word processor, this time killing the top margin and inserting a page break at the appropriate point.

Dumping graphics between blocks of text in this manner produces clean, presentable documents without the need to cut and paste or, incidentally, buy one of the limited number of printers which are supported by the dump routines of the best graphics programs. I use a *Taxan/Kaga KP-810* which is not supported by either *Apple Business Graphics* or *Visitrend/Visiplot*. — **David Bridson, Marketing Manager, Dark Star Systems.**

## Missing field

SOME months ago I completed a suite of programs designed to monitor jobs in our office. It is composed of several programs, written in *Applesoft*, and five text files.

The basis is a time sheet program which records time spent on a job by the people involved.

Unfortunately, after running

for two months, it was discovered that when the limits of the record lengths were reached, one field was missing from the end.

After investigation, it transpired that this was, in fact, missing from every record. When writing a record the length was 249 (which is correct) and when reading back a record the length was 239 (which is nine bytes short).

I proceeded to write and run three 'Noddy' programs (one of which I have listed here) and the results, apart from confirming where the bug is, were reasonably enigmatic.

```
10 D$ = CHR$(4)
20 FOR X = 1 TO 248: A$ = A$ +
  "X": NEXT X
30 PRINT D$; "OPEN
  TOD, S6, D1, L249"
40 PRINT D$; "WRITE TOD, R1":
  PRINT A$
50 PRINT D$; "CLOSE TOD"
60 PRINT D$; "OPEN
  TOD, S6, D1, L249"
70 PRINT D$; "READ TOD, R1":
  INPUT A$: PRINT D$
80 PRINT "LENGTH OF A$="; LEN
  (A$)
```

```
JRUN
LENGTH OF A$=239
```

I would, therefore, be grateful to receive enlightenment. — **T. Shreeve, Norwich.**

● The *Apple Basic* manual says that strings can have a maximum length of 255 but doesn't point out that *INPUT* only allows up to 239 characters to be entered at the keyboard or via a disc.

Do not, however, despair; your 248 characters are safely on the disc, you just cannot read them with *INPUT*. One answer is to delete line 70 in your last example and replace it with:

```
65 PRINT D$; "READ TOD, R1"
70 A$ = ""
75 FOR X = 1 TO 248: GET
  T$: A$ = A$ + T$: NEXT :
  PRINT D$
```

**Max Parrott**

## Apple Raid problem

THE program *Apple Raid* in the May 1984 *Apple User* can only be started on our *Apple IIc* by typing *RUN 740* as the high-score system does not work.

This also means that you cannot get instructions on how to play, although this can be achieved with *RUN 1080*.

Please could you check the printout of the program I am sending you and also inform me of any adjustments I might have to make. — **David Hallett, Pettistree, Suffolk.**

I HAVE a problem on my *IIc* with the game *Apple Raid* (May

1984 *Apple User*). Having just received the June edition, I see that there is no correction made to the published listing and instructions for this game, so I assume it must be me — and hope you can help.

When the game is loaded from disc I follow the instructions on page 58 re what is necessary to run the game and type in *HGR2: GOTO 1510 Return*.

The screen goes blank and the drive does not move, as apparently it should (so this must be the problem), and when I press *Ctrl Reset* I get the

following addition up on the screen:

```
B$AVE ARHSC,A16477,L2
LOCK ARHSC
```

Then by typing *Run Return* I get:

```
FILE NOT FOUND
BREAK IN 20
```

I have triple checked my listing, but not being a buff, I am not sure what the so called initialising the high score routine is all about. — **J. Scott, Loddon, Norfolk.**

● Sorry, this is our fault. Before typing in *HGR2: GOTO 1510 Return* type *D\$ = CHR\$(4) Return*. Then carry on.

