

Gebruikersdag te Hilversum.

Zaterdag 12 april a.s. zal er weer een gebruikersdag worden georganiseerd, en wel van 11 tot 4 uur bij

Maban Nederland
Ruitersweg 60A
1211 KX Hilversum
tel. 035-233769

Op het programma staan o.a. (onder voorbehoud) :

- Een demonstratie van de uit Engeland overgekropen Muis.
- HIREG-Graphics-card. (Nederlandse vinding).
- Demonstratie laatste software.

Hopelijk tot ziens aldaar.

Gebruikersdag Noord-nederland.

Waarschijnlijk is het voor veel Enterprisers geen probleem om steeds naar Hilversum te reizen, maar om dat deze keer te voorkomen voor de vele Gebruikersdag in het noorden houden. Zover het ondergetekenden bekend is zal er ~~19~~ ²⁶ april een gebruikersdag in Hilversum zijn en wij hebben onze gebruikersdag op ~~19~~ ^{me} april geplaatst, deze vindt plaats aan het leekstermeer van 11 tot 4 uur, het volledige adres is:

Pav. "Meerzicht"
E.C. Van der Dong
Matsloot 3
9749 TJ Matsloot

Dit is te vinden Rijksweg A7 Groningen-Drachten afslag Peize - Roden - Hoogkerk, komend vanaf Groningen bij afslag rechtdoor en komend vanaf Drachten van afrit afkomen linksaf onder viaduct door weer linksaf en dan richting Roderwolde aanhouden.

Voor meer informatie kunt U zich wenden tot:

K.K.V.D. Dong
De Baander 4
9356 CM Tolbert
05945-15626

of

G. Idsardi
De Baander 57
9356 CL Tolbert
05945-16736

Bij voorbaat danken wij U voor uw komst.

Kees

Een foutje van BASIC?

Vanuit BASIC kunnen we een 'briefje' naar EXOS sturen om een andere systeem extensie een opdracht te geven. Een voorbeeld is een briefje met ":HELP WP". Als we dit onder BASIC intikken, zal het briefje bij de wordprocessor aankomen, die dan een boodschap op het scherm afdrukt. Willen we dit in een IS-BASIC programma opnemen, dan moeten we EXT (van "EXTension") gebruiken, met daarachter de tekst op het briefje in de vorm van een string. Dus bijvoorbeeld:

```
10 FOR I=1 TO 2
20 EXT "help wp"
30 NEXT I
40 PRINT "end of program"
```

Dit programma blijkt niet te werken! Het stuurt het briefje in lijn 20 weg, WP drukt de boodschap af, maar dan geeft BASIC een "invalid end of block" error. Nu is het niet erg waarschijnlijk dat we een boodschap zoals hier twee keer achter elkaar willen afdrukken, maar andere EXT functies willen we misschien wel een aantal keer achter elkaar gebruiken. Bovendien ontstaat deze error ook als we EXT in een DEF-block gebruiken. Maar gelukkig blijkt er iets tegen te doen: Verander bovenstaand programma als volgt:

```
5 WHEN EXCEPTION USE REPAIR
10 FOR I=1 TO 2
20 EXT "help wp"
30 NEXT I
40 PRINT "end of program"
45 END WHEN
50 HANDLER REPAIR
60 RETRY
70 END HANDLER
```

Dit werkt wel.

QUASAR PASCAL voor de ENTERPRISE

Quasar Pascal dat is geschreven door Roelof Horst. Het Pascal-systeem bestaat uit een compiler, een debugger en een full-screen editor. De compiler is in staat om standaard-Pascal programma's te vertalen, behalve de GOTO-statements, de SET-constructies en parameters van het type FUNCTION of PROCEDURE. Het eerste is echter eerder een voordeel dan een nadeel, omdat het gebruik ervan onherroepelijk leidt tot slecht programmeren en onoverzichtelijke programma's. Roelof heeft mij verteld dat het ontbreken van sets te maken heeft met het feit dat deze constructies niet efficiënt te programmeren zouden zijn op een Z80-machiner, omdat de woordlengte te klein is. Voor niet-Pascal-kenners; een set is een Pascal-constructie waarmee verzamelingen geprogrammeerd kunnen worden, je moet bij de declaratie opgeven welke "elementen" lid kunnen zijn van die verzameling, en vervolgens kunnen aan deze set (engels voor verzameling) elementen worden toegevoegd e.d. Zo'n set wordt in het geheugen opgeslagen door voor ieder mogelijk element een bit te reserveren in het geheugen. Als een bit "1" is betekent dat dat het betreffende element lid is van de verzameling. Als je nu lange woorden tot je beschikking hebt, kun je een complete set in een woord opslaan, maar de Z80 heeft maar een 8-bits woordlengte. Voor deze paar gebreken zijn echter wel een aantal andere dingen in de plaats gekomen, nl:

- Een OTHERWISE tak in het CASE-statement
- PLOT en ELLIPSE-statements
- De procedure PAGE (zelfde als WRITE(CHR(12)))
- enkele gonio-functies : ARCSIN, ARCCOS, TAN
- INLINE-statement: om machinetaal tussen te voegen
- Z80-registers zijn direct als variabelen beschikbaar
- EXOS-statement om EXOS-calls te geven
- SVAR-statement om EXOS-variabelen te veranderen
- ADDR-functie geeft het adres van een variabele
- SIZE-functie geeft de benodigde geheugenruimte van een variabele in bytes

Als er tijdens het vertalen van het programma een fout wordt ontdekt in het programma, dan kan de full-screen editor worden aangeroepen, waarna de cursor automatisch bij het foute statement wordt geplaatst. In de editor zijn de functie-toetsen voorzien van complete Pascal-constructies. Zo zal een druk op functietoets 2 bijvoorbeeld "FOR TO DO" op het scherm zetten, terwijl de cursor direct achter "FOR" staat te knippen. Dit werkt erg prettig, als je er eenmaal aan gewend bent.

De debug-mogelijkheden zijn ook erg handig om het foutzoeken te vergemakkelijken. Als i.p.v. RUN, het DEBUG-commando gegeven wordt, dan zal het programma stap voor stap worden uitgevoerd, terwijl het bij ieder assignment-statement een aantal gegevens afdrukt, zoals Program-Counter, Stack-pointer, adres van de variabele en zijn waarde. De debugger kan ook vanuit het programma zelf bestuurd worden m.b.v. de procedures TRON en TROFF.

Afgezien van een aantal kleine foutjes (waarvan Roelof mij verzekerde dat ze in de huidige versie zijn opgelost) kan ik concluderen dat QUASAR PASCAL een zeer compleet pakket is, zeer compacte code oplevert (ca. 15 to 50 keer zo snel als BASIC) en goede foutmeldingen geeft. Samen met de debugger is het foutzoeken hierdoor stukken gemakkelijker dan bv. HISOFT-PASCAL op de Spectrum.

QUASAR PASCAL kost f159,- en kan besteld worden door dit bedrag over te maken op giro nr 3793879 t.n.v. R.J.Horst,

molenstraat 29
7514 DL Enschede

U krijgt dan de cartridge samen met de nederlandse handleiding toegestuurd.

MACHINETAAL OP DE ENTERPRISE

Je kunt machinetaal op verschillende manieren in de enterprise opslaan; in BASIC met CODE of DATA of met DEVPAK (assembler/dis assembler/debugger).

In dit stuk wil ik deze drie vergelijken, voorbeelden en misschien onbekende mogelijkheden ervan geven.

1 CODE

De CODE-opdracht van IS-BASIC is een opdracht die een string met daarin de bijtjes van het machinetaal programma in het geheugen zet. Dit geheugen wordt verkregen door de ALLOCATE-opdracht die het BASIC-programma het gegeven aantal bijtjes opschuift naar een hoger adres.

Dit geheugen komt dan vrij voor de gebruiker. Als het programma wordt beëindigd, schuift het BASIC-programma weer terug naar zijn oude plaats, en is het machinetaalprogramma dus ook weer verdwenen.

In versie 2.0 van BASIC bevatte de ALLOCATE opdracht nog een aantal fouten:

- Hij schoof BASIC niet altijd het juiste aantal bijtjes op.
- In de BASIC -listing wilde hij nogal eens wat ongewenst veranderen.
- Een programma mocht geen DEF's hebben omdat hun oude plaats werd onthouden terwijl ze opgeschoven werden.

Gelukkig zijn al deze fouten verbeterd in versie 2.1.

Het is mogelijk met CODE labels te programmeren, alhoewel dat redelijk ingewikkeld is. In dit voorbeeldprogrammaatje wil ik de mogelijkheden met CODE laten zien.

```
100 ALLOCATE 40
110 CODE DATA="FABIAN "
120 CODE ROUT=HEXE("3E?0;1,7,0")
130 CODE=HEXE("11")& WORDE(DATA)
140 CODE=HEXE("F7;8,C3")&WORDE(ROUT)
150 CALL USR(ROUT,0)
```

Nadelen van CODE:

- Alleen korte programma's blijven overzichtelijk.
- Je kan zelf de geheugenplaats waar de code terecht komt niet bepalen.
- Je kunt er niet goed de source-tekst bij plaatsen (b.v.: !LD HL, 50)
- Je kunt geen labels maken die vooruit springen.
- Je moet zelf alles assembleren.
- Alleen notatie in hexadecimaal.
- Doordat je ALLOCATE moet gebruiken schuift de BASIC telkens weer terug, zodat je het machinetaalprogramma weer kwijt bent en daarin geen peeks en pokes kunt uitvoeren.

2 DATA

Om van de ALLOCATE-fouten af te zijn in versie 2.0, ben ik op zoek gegaan naar grote, vrije geheugendelen.

Die heb ik gevonden: een grote (b.v. voor het programma zelf) van adres 1470 tot 3200, en een kleine (b.v. voor variabelen) van adres 290 tot 480.

Deze gelden zowel voor versie 2.0 als voor 2.1.

Verder heb ik een methode bedacht om langere programma's te kunnen maken die nog overzichtelijk blijven, en gemakkelijk te veranderen zijn. Als je de bijtjes opslaat met DATA, kun je de source-tekst er gemakkelijk achter zetten, en zelf bepalen waar de code komt.

Hier is het programmaatje:

```
100 FOR X=1470 TO 3200
110 READ IF MISSENG EXIT FOR:R
120 POKE X,R
130 NEXT
```


Vorm van de data:

Labelplaats: .2,label(daarachter geen spaties)

```
B.V. DATA .2,ADDAB!-----  
      DATA 128          !ADDA,B  
      DATA 201          !RET
```

LABELCALL : .1,label (label+spaties minder als 6 characters)

```
B.V. DATA 205,.1,rout ! CALL ROUT
```

Beeindigen DATA van machine-code:

```
DATA .2,END (achter "END" geen spaties!)  
B.V. DATA 201          !RET  
      DATA .2,END!-----  
      PRINT "HELLO"
```

Je kunt ook negatieve waarden doorgeven:

```
b.v DATA 62,-1          !UDA,-1  
b.v DATA 32,-4          !JR NZ $-4
```

Nadelen van DATA(LANGE VERSIE):

- Je moet zelf alles assembleren.
- Beperkte geheugenruimte(2000 bijtes)
- Notatie alleen in decimaal.
- Het in het geheugen zetten van het programma kan bij lange programma's lang duren.
- Beperkte mogelijkheden met labels.

Het is misschien ook wel mogelijk om net als labels macro's op deze manier te doen, maar dat maakt het programma natuurlijk wel langer.

4 DEVPAC

DEVPAC bevat twee programma's uit gegeven door HISOFT, namelijk GEN, een assembler en MON, een debugger/disassembler.

Deze twee programma's zijn op elkaar afgestemd. MON/GEN maakt geen gebruik van BASIC en heeft ook geen voorgeprogrammeerde routine's of macro's dus moet men een programma maken dat hele maal op zichzelf staat. Ze zijn goed verzorgd en er zit een redevolgende goede engelse handleiding bij. Ze hebben veel mogelijkheden en hebben voordeel t.o.v. machinetaal in BASIC, zoals:

- Krachtige macro's.
- Relocatable programma's
- Conditioneel assembleren.
- Veel mogelijkheden met labels.
- ±Notatie in hex/dec/binair.
- Opzoeken van een string in de source-code of in het geheugen
- ±Stap voor stap uitvoeren van de instructies van een machinetaal-programma, waarbij de registers telkens met hun nieuwe waarde op het scherm komen.
- Bepaald geheugendeel vullen met een bepaalde character.

Omdat ik DEVPAC nog niet zolang heb en wat problemen heb met de TV en de cassetterecorder, heb ik nog niet alle mogelijkheden goed kunnen onderzoeken. Toch ben ik wel wat nadelen tegen gekomen.

- Je moet DEVPAC kopen. (fl.99,-)
- Voordat je een programma kunt maken moet je eerst GEN laden van het cassettebandje, wat vrij lang duurt (Het programma is ongeveer 14 K lang) en machinetaal-programma's crashen snel dus dan moet je alles weer opnieuw laden.
- Je hebt geen direct-mode instructies zoals in BASIC (b.v. SET SPEAKER OFF; SET FKEY, CLEAR SCREEN, TEXT 80). Een paar daarvan kun je wel alvast van uit BASIC regelen alvorens je naar GEN gaat.
- Je kunt geen BASIC gebruiken naast je machinetaal-programma en er zijn ook geen voorgeprogrammeerde macro's dus moet je veel routines zelf maken, vooral routines die EXOS aan roept met RST en krijg je bij betrekkelijk eenvoudige programma's een grote source-code.
- Je verwisselt gemakkelijk de BASIC en de GEN editor met BASIC moet je b.v. LIST 100-200 doen, terwijl dat met GEN 1 100,200 is

-Als je programma's stap voor stap wilt testen, kan MON geen RST's uitvoeren.

OPMERKING:

Als DEVPAC op rompac of op disk uit komt, heb je niet zozeer een laad probleem meer.

Ik hoop dat ze dan ook voorgeprogrammeerde macro's of routines erbij zetten, wat de programmeertijd een stuk terug zal brengen.

CONCLUSIE

Mijn DATA-manier vind ik zelf een heel redelijk alternatief voor DEVPAC?, ik heb er lange programma's mee kunnen maken: ongeveer 24k aan BASIC-geheugen.

Ik ben aan het twijfelen welke manier ik moet kiezen, GEN en MON zijn goede programma's maar hebben ook hun onoverkomelijke nadelen. CODE 3 eigenlijk alleen maar geschikt voor hele kleine routines, die een hulpje zijn voor BASIC. Iedereen kan het beste zelf bepalen welke methode past bij zijn gebruik van machinetaal.

Als je problemen met of vragen over mijn programma of deze text hebt, bel me dan even.

FABIAN MOONEN
KRIJTERS LAAN 18
1609 EM ZWAAG
TEL. 02290-37122

A McIntyre Xmas

In a pre-Christmas advertising campaign, Enterprise, in conjunction with McIntyre Marketing Ltd. have announced two mail order package deals on Enterprise products.

The first package will include an Enterprise 128, a data recorder, joystick interface, joystick and five software titles all for £199.95. The normal retail price would be £338.55.

The second package includes a 128 computer, Fidelity TV/monitor, joystick interface and five software titles. Normal retail price would be £529.90. Through the McIntyre/Enterprise deal these products can be purchased for £299.95.

The software titles are Steve Davis Snooker, Nodes of Yesod, Machine Code for Beginners, Star Strike 3D and Beach Head. The advertising is being restricted to the national press and the packages are only available by mail order direct from McIntyre Marketing Ltd.

A spokesman for Enterprise could not give exact details on how long the deal would last but doubted whether it would be continued after Christmas.

No more hard times

A fair bit of news on the hardware front this month. Firstly, the disc interface you've all been waiting for should be available as you read this (the official release date is November 1st), priced £99.95. See the review later in this issue which should remove any doubts as to how brilliant EXDOS actually is.

Internal RAM expansions for 64K owners are available now, at cost price to those who paid the original £250 for

their machine. The external RAM expansions have unfortunately been delayed until after Christmas... Oh well, no 4 megabyte machines yet.

Subject to negotiations with the powers that be, we should soon be able to offer IEUG members a joystick with built-in interface to connect directly with the Enterprise (providing it passes the IEUG quality control process!). This will be manufactured by Aztec and will be based on the Vulcan Gunshot joystick, with auto-fire option. More news next issue.

Finally, the mouse and Speakeasy (conspicuously absent at the PCW show) should also be with us in the very near future. The mouse will be released with some software of the icon-driven type, although don't expect anything to rival the Macintosh just yet. The Speakeasy will have 2 text decoders - English and German, with more on the way including French and Danish. These will allow the unit to pronounce normal text, this being an improvement upon the standard Allophone system.

At last

There is a lot of movement on the software front at the moment, as can be seen from the reviews later on.

Software companies are at last making some use of the advanced features of the Enterprise, notably the sound chip, "Dave". The following titles are scheduled for release by the end of November :-

Lands of Havoc	(Microdeal)
Super Pipeline II	(Taskset)
Frank Bruno's Boxing	(Elite)
DEV PAC	(Hisoft)
Airwolf	(Elite)

Bargain drives

Cumana have started manufacturing 3.5" disc drives in distinctive 'Enterprise grey' at the very low price of £199.95 and £109.95 for the dual and single drives respectively. These drives are double sided, double density units with a built in power supply and offer real value for money (even cheaper if you choose to take advantage of our disc drive offer - see the Deals page for details!).

It is possible that Enterprise may be releasing a disc drive with built in EXDOS disc interface for the German market. However it is still in prototype form and it is uncertain whether a commercial version will eventually be made available.

En als voorbeeld een paar regels extra:

```
200 INPUT A
210 PRINT USR (1470,A)
220 GOTO 200
500 DATA 41          ! ADD HL,HL
510 DATA 1,50,0     ! LD BC,50
520 DATA 9          ! ADD HL,BC
530 DATA 201        ! RET
```

''''''''''
9 spaties

Nadelen van DATA(korte versie):

- Je kunt geen labels gebruiken, daarom is het niet mogelijk om alte lange programma's te maken.
- Je moet zelf alles assembleren.
- Notatie alleen in decimaal.
- Beperkte geheugen ruimte (2000 bijtes)

3 DATA (lange versie)

Om toch wat langere machinetaal-programma's te kunnen maken heb je labels nodig.

Dus heb ik geprobeerd daarvoor ook een programma te maken. Dat is me ook gelukt, hier is het resultaat:

```
8000 LET P=1470
8010 STRING L$=*150,C$(50)*5
8020 LET L$="1234"
8030 NUMERIC PLL(30),PLC(50)
8040 LET L=0:LET C=0:LET PLL(0)=99999
8050 DO
8060   DO
8070     READ @
8080     IF 0=.1 THEN LET C=C+1:LET PLC(C)=P:LET P=P+1:READ
           C$(C)
8090     IF 0=.2 THEN EXIT DO
8100     POKE P, 0:LET P=P+1
8110     LOOP
8120     LET L=L+1:READ L1$
8130     LET L$=L$&L1$&"      "(LEN(L1$):)
8140     LET PLL(L)=P
8150     PRINT L1$,P
8160 LOOP UNTIL L1$="END"
8500 FOR C2=1 TO C
8510   LET PLL2=PLC(POS (L$,C$(C2)))/5)
8520   IF PLL2=99999 THEN LET PLL2=PLL(POS(L$,C$(C2),90)/5)
8530   IF PLL2=99999 THEN PRINT "LABEL NIET GEVONDEN VCOOR AAN-
           ROEP" C$(C2):STOP
8540   POKE PLC(C2)+0,MOD(PLL2,256)
8550   POKE PLC(C2)+1,INT(PLL2/256)
8560 NEXT C2
```

Je kunt een label ook beschikbaar maken voor gebruik in BASIC (b.v. aan roep van een routine met CALL USR(START,0):

```
8700 LET START=PLL(POS(L$, "START")/5)
```

En ook een label een andere waarde geven dan hem in het programma gegeven werd:

```
8200 LET PLL(POS(L$, "DATA")/5)=300 ! LET OP HET REEDELNUMMER,
HET MOET IN HET PROGRAMMA KOMEN !!!!!
```

Of je kunt een nieuwe label maken (b.v. voor een constante die je gemakkelijk wilt kunnen veranderen):

```
8210 LET L=L+1:LET L$=L$&"CONST":LET PLL((L)=4000 ! Ook hier op
regelnummer letten.
```

Vorm van de data:

Labelplaats: .2,label(daarachter geen spaties)

```
B.V. DATA .2,ADDAB!-----  
      DATA 128          !ADDA,B  
      DATA 201          !RET
```

LABELCALL : .1,label (label+spaties minder als 6 characters

```
B.V. DATA 205,.1,rout ! CALL ROUT
```

Beeindigen DATA van machine-code:

```
DATA .2,END (achter "END" geen spaties!)  
B.V. DATA 201          !RET  
      DATA .2,END!-----  
      PRINT "HELLO"
```

Je kunt ook negatieve waarden doorgeven:

```
b.v DATA 62,-1          !UDA,-1
```

```
b.v DATA 32,-4          !JR NZ $-4
```

Nadelen van DATA(LANGE VERSIE):

-Je moet zelf alles assembleren.

-Beperkte geheugenruimte(2000 bijtes)

-Notatie alleen in decimaal.

-Het in het geheugen zetten van het programma kan bij lange programma's lang duren.

-Beperkte mogelijkheden met labels.

Het is misschien ook wel mogelijk om net als labels macro's op deze manier te doen,maar dat maakt het programma natuurlijk wel langer.

4 DEVPAC

DEVPAC bevat twee programma's uit gegeven door HISOFT, namelijk GEN, een assembler en MON, een debugger/disassembler.

Deze twee programma's zijn op elkaar afgestemd. MON/GEN maakt geen gebruik van BASIC en heeft ook geen voorgeprogrammeerde routine's of macro's dus moet men een programma maken dat hele maal op zichzelf staat. Ze zijn goed verzorgd en er zit een redevol goede engelse handleiding bij. Ze hebben veel mogelijkheden en hebben voordeel t.o.v. machinetaal in BASIC, zoals:

-Krachtige macro's.

-Relocatable programma's

-Conditioneel assembleren.

-Veel mogelijkheden met labels.

+Notatie in hex/dec/binair.

-Opzoeken van een string in de source-code of in het geheugen

+Stap voor stap uitvoeren van de instructies van een machinetaal-programma, waarbij de registers telkens met hun nieuwe waarde op het scherm komen.

-Bepaald geheugendeel vullen met een bepaalde character.

Omdat ik DEVPAC nog niet zolang heb en wat problemen heb met de TV en de cassetterecorder, heb ik nog niet alle mogelijkheden goed kunnen onderzoeken. Toch ben ik wel wat nadelen tegen gekomen.

-Je moet DEVPAC kopen. (fl.99,-)

-Voordat je een programma kunt maken moet je eerst GEN laden van het cassettebandje, wat vrij lang duurt (Het programma is ongeveer 14 K lang) en machinetaal-programma's crashen snel dus dan moet je alles weer opnieuw laden.

-Je hebt geen direct-mode instructies zoals in BASIC (b.v. SET SPEAKER OFF, SET FKEY, CLEAR SCREEN, TEXT 80). Een paar daarvan kun je wel alvast van uit BASIC regelen alvorens je naar GEN gaat.

-Je kunt geen BASIC gebruiken naast je machinetaal-programma en er zijn ook geen voorgeprogrammeerde macro's dus moet je veel routines zelf maken, vooral routines die EXOS aan roept met RST en krijg je bij betrekkelijk eenvoudige programma's een grote source-code.

-Je verwisselt gemakkelijk de BASIC en de GEN editor met BASIC moet je b.v. LIST 100-200 doen, terwijl dat met GEN 1 100,200 is

-Als je programma's stap voor stap wilt testen, kan MON geen RST's uitvoeren.

OPMERKING:

Als DEVPAC op rompac of op disk uit komt, heb je niet zozeer een laad probleem meer.

Ik hoop dat ze dan ook voorgeprogrammeerde macro's of routines erbij zetten, wat de programmeertijd een stuk terug zal brengen.

CONCLUSIE

Mijn DATA-manier vind ik zelf een heel redelijk alternatief voor DEVPAC?, ik heb er lange programma's mee kunnen maken: ongeveer 24k aan BASIC-geheugen.

Ik ben aan het twifelen welke manier ik moet kiezen, GEN en MON zijn goede programma's maar hebben ook hun onoverkomelijke nadelen. CODE 3 eigenlijk alleen maar geschikt voor hele kleine routines, die een hulpje zijn voor BASIC. Iedereen kan het beste zelf bepalen welke methode past bij zijn gebruik van machinetaal.

Als je problemen met of vragen over mijn programma of deze text hebt, bel me dan even.

FABIAN MOONEN
KRIJTERS LAAN 18
1609 EM ZWAAG
TEL. 02290-37122

A Mc Intyre Xmas

In a pre-Christmas advertising campaign, Enterprise, in conjunction with McIntyre Marketing Ltd. have announced two mail order package deals on Enterprise products.

The first package will include an Enterprise 128, a data recorder, joystick interface, joystick and five software titles all for £199.95. The normal retail price would be £338.55.

The second package includes a 128 computer, Fidelity TV/monitor, joystick interface and five software titles. Normal retail price would be £529.90. Through the McIntyre/Enterprise deal these products can be purchased for £299.95.

The software titles are Steve Davis Snooker, Nodes of Yesod, Machine Code for Beginners, Star Strike 3D and Beach Head. The advertising is being restricted to the national press and the packages are only available by mail order direct from McIntyre Marketing Ltd.

A spokesman for Enterprise could not give exact details on how long the deal would last but doubted whether it would be continued after Christmas.

No more hard times

A fair bit of news on the hardware front this month. Firstly, the disc interface you've all been waiting for should be available as you read this (the official release date is November 1st), priced £99.95. See the review later in this issue which should remove any doubts as to how brilliant EXDOS actually is.

Internal RAM expansions for 64K owners are available now, at cost price to those who paid the original £250 for

their machine. The external RAM expansions have unfortunately been delayed until after Christmas... Oh well, no 4 megabyte machines yet.

Subject to negotiations with the powers that be, we should soon be able to offer IEUG members a joystick with built-in interface to connect directly with the Enterprise (providing it passes the IEUG quality control process !). This will be manufactured by Aztec and will be based on the Vulcan Gunshot joystick, with auto-fire option. More news next issue.

Finally, the mouse and Speakeasy (conspicuously absent at the PCW show) should also be with us in the very near future. The mouse will be released with some software of the icon-driven type, although don't expect anything to rival the Macintosh just yet. The Speakeasy will have 2 text decoders - English and German, with more on the way including French and Danish. These will allow the unit to pronounce normal text, this being an improvement upon the standard Allophone system.

At last

There is a lot of movement on the software front at the moment, as can be seen from the reviews later on.

Software companies are at last making some use of the advanced features of the Enterprise, notably the sound chip, "Dave". The following titles are scheduled for release by the end of November :-

Lands of Havoc	(Microdeal)
Super Pipeline II	(Taskset)
Frank Bruno's Boxing (Elite)	
DEV PAC	(Hisoft)
Airwolf	(Elite)

Bargain drives

Cumana have started manufacturing 3.5" disc drives in distinctive 'Enterprise grey' at the very low price of £199.95 and £109.95 for the dual and single drives respectively. These drives are double sided, double density units with a built in power supply and offer real value for money (even cheaper if you choose to take advantage of our disc drive offer - see the Deals page for details !).

It is possible that Enterprise may be releasing a disc drive with built in EXDOS disc interface for the German market. However it* is still in prototype form and it is uncertain whether a commercial version will eventually be made available.

Private Correspondence

not only are you super men it might even make me write another superbly grovelling letter which will be even longer.

P.S The WP can't use all the memory of the 128 (and perhaps the 64) and it doesn't tell you when it runs out it just erases the document starting at the beginning. Nice one IS.

David Good
Derby

TB. Thanks for the praise - most appreciated. So, someone else noticed the odd mistake or twenty - hopefully there should be less from now on. Solving the line input problem took me a few minutes to solve, but at last I found the answer.

(1) Close $\{102$: and $\{0$.

(2) Open a text video page the size you want it (no smaller than 5 across and 3 down) and number it $\{102$.

(3) open $\{0$: "EDITOR:".

(4) Display $\{102$: From now on this is used for input etc.

All to much for Eric

I've been waiting for Issue Two for over a month now. I almost gave up hope, thinking it probably got lost in the post... until it arrived a few days ago, finally! Just seeing the large white envelope made my heart jump, opening it revealed another surprise : Issue 2 is all glossy! Smart isn't it? Who knows, the cover might be in full colour next time!

My favourite article in Issue One was Outside Connections - it permitted me to build my own printer and stereo cables, as well as a joystick interface. The programming feature in issue two is quite revealing - I haven't investigated all the "controls" and "escapes" yet. I decided to write this letter first.

I know the IEUG is crying out for contributions from members. So, I've been thinking about what I could contribute. Programs? Mine are usually long and not spectacular. Programming tips? I've always wondered "where do they get these tips?" Articles? I hardly know more about the Enterprise than the average user. However if there is still no article about "sound" in issue 3. I might write one in spite of my limited knowledge. Anyway, when I can afford to buy the Lisp cartridge I'll send you a few Lisp programs.

A few questions concerning the disk controller EXDOS: is CPM-80 included in the price? Will it read CPM disks for other machines such as Amstrad or Apple? In what size disk format will software be released?

Eric Lew.
London.

TB. Included in the price of EXDOS is IS-DOS, a CPM emulator (Enterprise will supply you with a disc containing this upon returning a card supplied with EXDOS). I have seen it read Amstrad discs, although it required a utility program to do so. As for Apple discs, so long as you have the right size disc drive. I cannot see any reason why it would not be possible read them. In answer to your last question, the Enterprise standard for disc-based software releases is 3.5", 40 track, single sided, although with EXDOS any 3.5" drive will be able to read this.

View from the fjords

I read about you in the latest issue of 'The Enterpriser' - a magazine for members of the "Official" Enterprise club in Norway.

Until a short time ago, I was seriously thinking of starting an independant user club here in Norway.

But I found out that it would demand too much time and effort working alone, so I gave up. Therefore, I'm very glad to see that somebody has got enough guts to start it.

Here in Norway we hve the great disadvantage of having greedy businessmen as the sole inporters of Enterprise equipment and software. It is terrifying to see how much we have to pay. I will give you some Examples: (compare them with your prices!)

Enterprise 64	£348
Enterprise 128	£417
Enterprise mouse	£80
Enterprise Speakeasy	£56
Enterprise memory expansion approx.	£100
Enterprise Printer	£348

	UK	NORWAY
Enterprise software	£7.50	£ 14
	£10	£ 17
eg Lisp		£ 43

Even if we have a two-year warranty, 20% VAT plus freight from England to Norway, there is no excuse for having such high prices. I have been looking for a company in England which can supply me and other Enterprise users here in Norway with hardware at cheap prices.

E.l.Rossebo
Norway.

TB. I'm sure the rest of our readers like myself will agree that £417 for a 128 is just a touch expensive (to say the least). But as they are the sole importers there doesn't seem to be much that can be done. If we come up with a supplier in this country we will let you know.

Who shot J.Jim?

Thanks for the user group and the 2nd issue of an excellent magazine. It's

Private Correspondence

good to know there's life out there, but is there life over here? I'm having great difficulty in finding a dealer stocking any of the limited software - this includes the list of stockists supplied by Enterprise none of whom even appear to stock the hardware either. So don't forget us over here when the cut price peripherals and software become a reality. I hope the same mistake isn't made again in distributing, as with the computer. I had to buy my Enterprise 64 through Dublin at inflated prices including 23% VAT.

I'm not a computing genius in fact I must admit a lot of it is over my head but I would appreciate an article on the sound chip Dave and how to get the most out of it in programming.

Now for a few gripes at Enterprise, where is all the advertised software such as Stud Poker, Jungle Jim, Supersonics and the Basic to Basic converters, also where is that basic overlay that was promised with the demo tape?

Also I have become one of the unlucky 5% who had to return a faulty machine. The fault was with the rom cartridge which was on Eprom and caused the machine to crash. I was also disappointed when it was returned with the old 2.0 version instead of the new 2.1 version.

M. Gallagher
Co Derry

TB. You too seem to be at the mercy of the distributor. Why not get a relative in England to send you the items you require - it must be cheaper than paying those prices. You bring up some good points about the software. Come on Enterprise, let's have some answers, we know you read this magazine. We won't forget you in our software deals, remember you are 'no longer alone' (All puns intended).

PS. The Basic overlay is in fact the INTERLACE program.

Transforming

After your plea for letters I'm writing this to 'ye all'. Firstly, I don't profess to being a proficient programmer. I would rate myself as slightly passable so that's why I didn't enclose anything.

Now, with that finished on with the show. I upgraded from the Spectrum and naturally discovered a few differences. On the Spectrum I had a simple accounts program running so I made an attempt on a conversion. This proved to be easy, until I ran it! On the ZX I loaded the info into arrays then broke in to make a few corrections. To run the program I pressed goto so as not to erase the data. However on the Enterprise I can't do that and it really annoys me.

Another thing that annoys me is the power transformer. I am onto my second power transformer costing me £24. Why?? The wire keeps breaking out where it leaves the transformer. Already I have had to wrap insulating tape round it. Has this happen to anyone else?

Leslie Aust.
Dublin.

TB. You're not the only one who's had trouble with the transformer. I myself found the same trouble but noticed it early and insulated it with glue gun plastic (fantastic stuff - I swear by it).

Pirates bashed

First of all YES there are people out there (here). How many I'm not sure as my member number appears to be 119. Only 119 of us or it could be 779.

Second PIRATES....I have an A???I (five letter words not allowed). Some

of the software for this machine has been particularly expensive i.e thirty to forty pounds in some case. Mule £28.95, ZORK set £35 each. JUMBO JET PILOT thirty pounds etc. Still, I did not write to complain to you about the expensive software for other machines. Anyway prices are coming down. The point is that you can't go out and buy much software like this because most of us can't afford it. But when you do and the quality is poor then you start to think about other means of obtaining it. I don't mind paying 10 to 20 pounds but that's me. What about people in the low income bracket (kids) seven or eight pounds might be a lot to them.

My conclusion is copying is relative. Relative to cost/quality. Relative to cost/own finances. Relative to media. By this I mean disk or tape.

This brings me to my next point disk drives and EXDOS. If Enterprise do release software on disk what format will it take? Will it be compatible? I can't see Enterprise or anybody else stocking a hoard of different formats.

Bob Tiffen,
Brentwood, Essex.

TB. I hate to tell you this, your number isn't either of those two, it's 219. We made a mistake in the numbers. Enterprise will only be stocking two disk types 3 1/2 40 track and 5 1/4 40 track. The reason for this is EXOS can read a 40 track disk even if its in an 80 track drive. You bring up some very valid points about piracy, some of which I have to agree with, but see what our next letter has to say about it.

...and slammed

Thank you for your prompt response to my application for membership of the I.E.U.G.

Private Correspondence

I especially appreciate the user magazine upon which I feel I must congratulate you. It seems to be a fine start - but, please, do not let it ever become a mere collection of listings of more and yet more computer games, as some commercially available magazines dedicated to lesser computers are. There are other things in life!

I am one of those apparently rare people who actually use a computer for computing. As a professional mathematician, I have used mostly APL on a minicomputer for statistical work, and although admiring the conciseness and flexibility of that language have become increasingly aware of its great limitation - an almost complete absence of 'structure'. Programs written by oneself appear to be so much incomprehensible gibberish after a very short while: understanding them becomes a work of art and amending them wellnigh impossible.

Becoming disenchanted with languages which lack structure features, I have waited for nearly two years for the Enterprise ne Elan: I even managed to wait long enough for the 128K model. It is, obviously, a recent purchase and, as yet I have no dedicated peripherals. I look forward to your reviewing these, as you have done for the EP80+ printer. With my background in mind, you will not be surprised at the following comments:-

1) Your correspondant who is so kinkily 'keen on software' seems to tipify the present age of craving something for nothing. Probably he has not written, is not willing to write, or is incapable of writing, acceptable programs of his own. It is fascinating but hard work - time-consuming, frustrating and infuriating hard work (I write from experience of seeking the eigenvalues of a matrix rather than the treasures in a cavern but I'm sure the principle is the same).

Anyone who wants the fruits of such labours for nothing should not be dignified by the romantic name of 'pirate'. Let us call a thief a thief.

2) Is it not just a little odd that your very first listing in your very first issue should contain a DIM, an ON GOSUB and three assorted unconditional GOTOs? Or was this a partial translation from one of the more primitive languages, e.g. Microsoft BASIC? (The author himself seems to have experienced some sort of conversion; his second program is mercifully free from such antiques).

3) The Enterprise manual is reasonably good, as manuals go. But can Enterprise be persuaded to fill in the glaring omissions with a supplement? In particular, where are tables giving the character set, the alternative characters and descriptions, however brief, of the 256 available colours?

John Smith,
Burton upon Trent.

NB. Many thanks for the praise - hopefully the quality of the magazine will continue to improve. I think you have nothing to fear as far as content is concerned - "Home Produce" seems more in danger of becoming a collection of graphics demonstrations than anything else at present! As far as structured programming is concerned, I think you may find the Pascal preview in this issue of interest (a series of articles concerned with structured programming techniques applicable to any language should be appearing soon). As for the other points you raise, I will take them in order :-

1) We at the IEUG fully agree with your attitude on software theft. Any misuse of the User Group in this area will not be tolerated.

2) The listings in Issue 1's "Home

Produce" section are written by ordinary members of IEUG. Many people have not yet come to grips with the power of IS-Basic, a situation which will be remedied in time. Instead of condemning the use of certain outdated structures, we should be educating people in the use of the better alternatives.

3) The Enterprise Advanced User manual will be available "early in 1986". However, we hope many items missing from the basic user guide will be covered in this magazine - in response to your request for a character set table, this will be included in Issue

4) As for the colours, it is a little difficult to describe 256 colours uniquely ("sort of light bluey-greeny grey..."), but maybe someone would like to attempt a cut-down version.

HINTS
and
Tips.

Our first offering of Hints and Tips comes from Andrew Richards of Yelford, if you have any drop us a line and tell the world about them. (Well it may not actually go all around the world but we do distribute to England, Ireland, Scotland, Wales, Norway, Holland, Arabia etc etc).

To get round the ALLOCATE bug do the following. First OUT 177,254. This sets up addresses 16384 to 32767 as in segment 254. This should be repeated after an error. Then type POKE 540,0 and POKE 541,64, this sets the code pointer to address 16384 and gets rid of the need for ALLOCATE and keeps the variables safe.

LOOK [105:A waits for a key to be

Private Correspondence

pressed and puts the ASCII value in A.

PRINT £255:"X". Prints X on the top left of the status line.

PRINT AT will not work on the last column of an EDITOR channel because of word wrap and the margin. Use PRINT £102, AT X,Y: instead.

SET can also take a number in the form SET X,Y. X is the option number and Y is the value. There are 38 options. You will have to test to find the one you want! 1 is STOP. SET 1,1 simulates pressing the STOP key. 6 is SHIFT and 3 is ALT. All these system variables are stored in segment (memory page) 255 at address 16329 onwards. So SPOKE 255,16329+6,1 will turn CAPS lock on,

Control of segments is through OUT N, X. N is a number between 176 and 179, donating the 16K area to put the segment in. 176 is for addresses 0 to 16383 etc. X is the segment number. 0 is EXOS, 1 is BASIC functions, 4 is BASIC commands, 252 to 255 are RAM. IN (N) returns the segment number in that area.

In your article you recommend HEX.WORD\$ is easier.

PRINT £101: chr\$(27)"E" WORD\$ (50)
WORD\$ (50)

will draw a circle 30 units wide. Negative numbers may also be used for relative drawing. Escape sequences also work on different devices except different escape sequences.

LOOK and GET can be used on text channels (somebody said you can't). They return the ASCII value (for LOOK) or character (GET) under the cursor.

Address 562 points to the Command Table. 3 bytes after this comes the number of commands and the A list, 3 bytes per command. The first two bytes are the address of the command info, and after that comes the segment (

ignored). The address points to a table. 2 bytes for RUN address and 2 for INPUT address, 1 byte for the type of command (83 is normal) and 1 byte for name length and then the name itself. Use this to find out about the in-built commands. The run address is called when the command is executed, the input address checks if what comes afterward is correct. A disassembler is vital to understand this and to change the table.

Devices are stored in segment 255 at about address 11000. Before the device comes a byte containing 0, if device is on an address and segment of a table of addresses. The tables first two bytes contain the address to call on an interrupt and then a list of call addresses for each of the calls up to 8, shown on page 201 of the manual.

LD A,0, LD B,"A", RST 48, DB 7
will print an "A" on channel zero (machine code).

LD A,105, RST 48, DB 5
will wait for a key pressed and return it in B.

All of this information should be experimented with so you can understand it fully (it would take a long time to explain in detail). I hope it is of use to you.

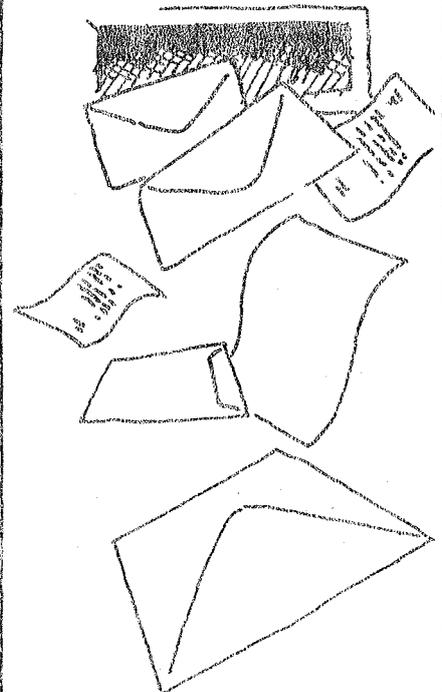
PS Who says Enterprise has BNM's image?

Andrew Richards,
age 14 1/2.
Telford.

TB. Thanks for your useful contribution - I'm sure many people will find the contents of your letter to be of great value. You touch upon some subjects many people are interested in but lack the technical expertise to explore fully - how about writing us a full-blown article?

To conclude this issue 'Private Correspondence', I would like to elaborate on what Phil Cohen said about getting ourselves heard. Enterprise have got themselves into a Catch 22 situation - people don't want to buy a machine they have never heard of, and if no machines are sold the Enterprise will remain unknown. Enterprise can only get out of this situation through heavy advertising, but while they get on with that, we can make our own contribution. Write to the press informing them of the virtues of the machine - you could even start up a 'Mines Better than Yours' slagging match if necessary (you know we would win!). In fact, stop at nothing to get the name of Enterprise known - it can only help us in the end!

P.S While you're at it, educate the people that the next best thing to Enterprise is the IEUG and Private Enterprise (plug plug).



Tim Box

Programming

```
100 RANDOMIZE
110 LET LOOK$=HEX$("1B,3D")
120 CLEAR SCREEN
130 FOR N=1 TO 152
140 PRINT CHR$(RND(26)+65);
150 NEXT
160 DO
170 INPUT AT 21,1:Y,X
180 PRINT [102:LOOK$;CHR$(X+32);
CHR$(Y+32)];
190 LOOK [102:A
200 PRINT AT 22,1:CHR$(A)
210 LOOP
```

The program sets up LOOK\$ to produce the position cursor escape code. Line 190 prints this escape code to the text channel, followed by the user entered cursor position. Note that the offset value of 32 is added to X and Y within the CHR\$ statement. The cursor is now positioned at the desired location on screen and the LOOK command that follows on line 200 puts the ascii code of the character at that position into A. This technique could be used equally well to look at a computer generated position.

The idea of user input control codes could also be used to produce a crude drawing program, in which the user enters the x and y beam displacements and relative plotting is used to move the beam by that amount:

```
100 LET PLOT$=HEX$("1B,52")
110 GRAPHICS
```

```
120 PLOT 650,350;
130 SET COLOUR 1,255
140 DO
150 INPUT PROMPT "X,Y ":X,Y
160 IF X<0 THEN LET X=X+65536
170 IF Y<0 THEN LET Y=Y+65536
180 LET X1=MOD(X,256)
190 LET X2=INT(X/256)
200 LET Y1=MOD(Y,256)
210 LET Y2=INT(Y/256)
220 PRINT [101:PLOT$;CHR$(X1);CHR$(
X2);CHR$(Y1);CHR$(Y2);
230 LOOP WHILE X<>0 OR Y<>0
```

In this example PLOT\$ is used to produce the start of the relative plotting escape sequence, X and Y are converted to LSB and MSB by lines 180-210 and the whole escape sequence is printed to the graphics screen in line 220. The semicolon at the end of line 220 is used to suppress the automatic carriage return.

As was mentioned earlier, using escape codes for relative plotting is far quicker than using turtle graphics. The last example demonstrates this fact. It draws a square, 100 units to each side, fifty times, firstly by turtle graphics then by using escape sequences. You should be able to notice the speed increase yourself.

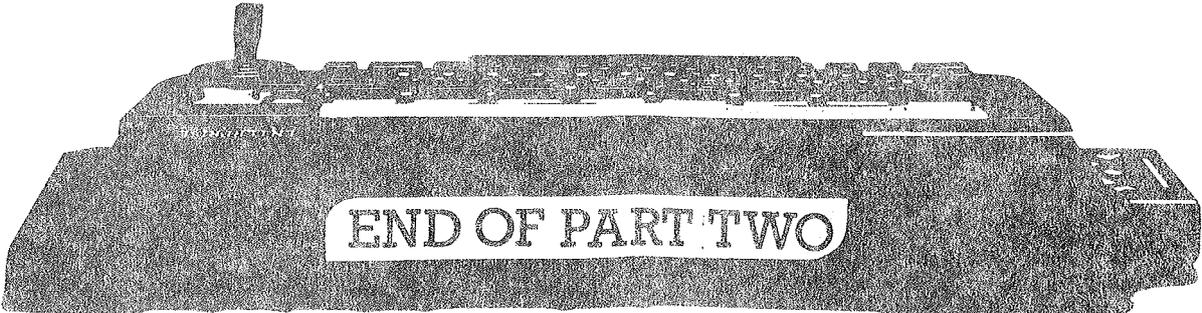
```
10 LET SQUARE$=HEX$("1B,52,64,0,0,
0,1B,52,0,0,64,0,1B,52,9C,FF,0,0,1B,
52,0,0,9C,FF,1B,73")
100 RANDOMIZE
```

```
110 GRAPHICS
120 SET PALETTE BLACK, GREEN, RED, CYAN
130 FOR N=1 TO 50
135 SET INK RND(3)+1
140 PLOT RND(1100),RND(600);
150 PLOT ANGLE 0;
160 FOR NN=1 TO 4
170 PLOT FORWARD 100,LEFT PI/2;
180 NEXT
190 SET BEAM OFF
200 NEXT
205 CLEAR GRAPHICS
210 FOR N=1 TO 50
215 SET INK RND(3)+1
220 PLOT RND(1100),RND(600);
230 PRINT [101:SQUARE$
240 NEXT
```

You should now have a good idea about what the various control codes and escape codes do. All that is left is to experiment with them yourselves and discover their advantages (and limitations). I will leave you with the short program I promised earlier to convert decimal values into hexadecimal LSB/MSB:

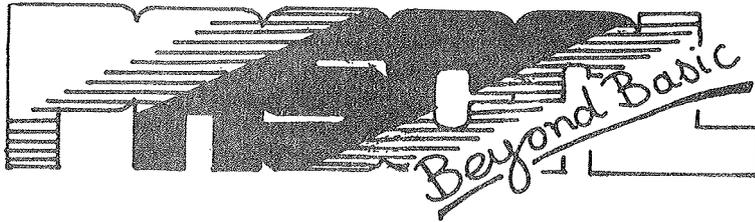
```
100 DO
110 INPUT PROMPT "DECIMAL VALUE:":
N
120 LET H$=WORD$(N)
130 PRINT "LSB";ORD(H$);"MSB";ORD(
H$(2;))
140 LOOP WHILE N<>0
```

DAVE
RACE



END OF PART TWO

Hi-soft



A question many people ask is "What is the natural progression in programming languages after BASIC?". Less informed individuals may propose machine code, pointing to its vast increase in speed. One of the main problems with machine code is ease of use. It takes a great deal of code to do even the most simple of tasks. Pascal, however, gives a large speed increase with powerful, high level structures which are easy to use. Machine code addicts are not left out though. Machine code can easily be directly embedded in Hisoft Pascal using the special `INLINE` function.

Hisoft Pascal is a 'compiled' language: that is, the program is first converted to machine code before it can be run. This gives a great increase over 'interpretive' languages like IS-BASIC, where each line of code is converted into machine code as it is found while the program is running.

The compilation time (time taken to convert Pascal into machine code) is very short, just a few seconds for even a large program. In normal compilation mode each line of the program is shown together with the memory address of the compiled version of the line. This slows the compiler down but can be turned on or off using a software switch within a comment. Other switches allow you to turn off various built in checking functions such as array boundary checking. This speeds up the program even more but makes debugging very difficult and I would recommend using this only after a program has been fully tested (some

hope!).

As a simple comparison of speeds, I wrote simple programs in IS-BASIC and Hisoft Pascal. Each of them adds together two values, 'b' and 'c', the result is then placed in 'a'. This is done a total of 32,000 times in a FOR loop.

IS-BASIC

```
100 PROGRAM simple
110 B=123
122 C=12345
130 FOR I=1 TO 32000
140 LET a=B+C
150 NEXT
```

Hisoft Pascal

```
PROGRAM simple;
VAR
  i,a,b,c : INTEGER;
BEGIN
  b:=123;
  c:=12345;
  FOR i:=1 TO 32000
  DO
    a:=b+c
  END.
```

Run time for IS-BASIC: 5 minutes 19 seconds.

Run time for Hisoft Pascal 2.23 seconds.

This was not a totally fair comparison as IS-BASIC handles all numeric values as REAL. Therefore I made a,b and c REALS and ran it again.

Run time for Hisoft Pascal using REAL values 10.1 seconds.

Slower, I admit but 10 seconds verses over 5 minutes! IS-BASIC still takes a pasting.

Pascal is not difficult to learn, especially if you have been making use of the advanced IS-BASIC features (wot, no GOTO's?). One of its attractive features is the control you have over variables. Pascal gives you a rich set of variable types, not just numeric and string variable types, you may also define your own variable types.

Hisoft Pascal contains a special function, EXOS, which gives the programmer access to the operating system. Pre-declared variables RA, RB, RC, RD, RE, RDE and RBC give access to the Z80 registers to pass parameters to EXOS. To make full advantage of this you need the Technical Manual (Hopefully there will be one commercially available soon - ED), although Hisoft Pascal does come with an example program (TURTLE.PAS) which contains routines for accessing some of the Enterprise's advanced graphics features. This is by no means a comprehensive library of routines, but IEUG will be printing a whole bunch of useful Pascal Procedures next issue which should give you access to just about everything that you may need. Once written, the Pascal program can be compiled and saved to tape. In this form the program can be run from tape without loading in the Pascal compiler.

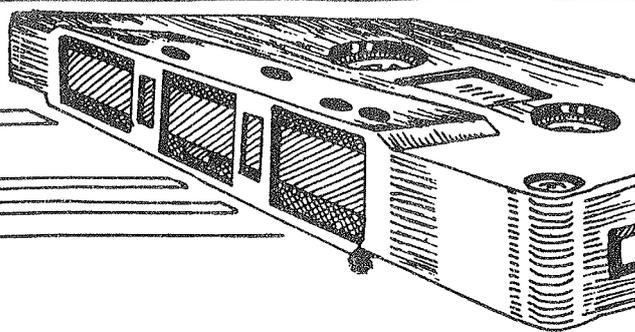
As I write, I have been given the latest version of Pascal, hopefully the compiler will be commercially available before Christmas. Hisoft Pascal will be available on tape and disc, but hopes for a ROM cartridge look slim. A full review of Hisoft Pascal will be given when we can get hold of the commercial version with manual.

P.S. Look out for some programs on the IEUG Christmas Bargain Mega Compilation Bonanza Tape written using Pascal.

Gary Thomson

Software

Update



KEY TO RATINGS;

ARCADE and ANIMATED ADVENTURES

- GAME CONTENT** - Variety of actions / screens
- PLAYABILITY** - Ease of use, addictive quality
- GRAPHICS** - Quality and use of graphics related to machine
- SOUND** - Use of stereo and tune / noise originality.
- VALUE FOR MONEY** - Overall impression when compared with price.

ADVENTURES

- GAME CONTENT** - Design of plot / background. Puzzle ingenuity.
- PRESENTATION** - Atmosphere, graphics (if any), text / screen layout.
- INTERACTION** - Parser quality, editing facilities
- VALUE FOR MONEY** - Overall impression when compared with price.

PERCENTAGES

- 0 - 25 - Yuk, Bleah !
- 26 - 50 - Bad to Mediocre
- 51 - 75 - Average to Good
- 75 - 100 - Excellent to completely Brilliant

Raid over...



Name : RAID
Producer : US Gold
Category : Arcade
Price : £9.95

In this strategy/arcade game you must destroy enemy missile silos to prevent missiles from landing in the good ole U.S. of A (yeeeee haaaaf ! Rustle them thaar steers boy!). This is achieved by flying your space fighters from the orbiting space station to the missile silos. After the silos' sites have been destroyed you can then attack the enemy defence centre. Although "the Enemy" is never mentioned by name, they are quite obviously the Russians (the defence centre being a blobby version of the Kremlin).

Raid is a Spectrum dump (copied over from a Spectrum including blocky attribute graphics on a reduced screen), though there is "stereo sound", well, stereo blips, whooshes and bangs actually. The game is split into 5 parts - in the first part you must fly your fighter out of the hanger (exciting stuff, huh? I think not). At first this is a bit of a challenge, but after a very short time it becomes annoying and tedious.

The second part involves flying the fighter over hostile territory. The instructions say that the controls for this scene are similar to the controls of a REAL JET! Now! My advice is: NEVER

let US Gold fly a jet.

The third part is an incredibly bad Invader/Galaxians type idea. You move left, right, up or down and can fire slow moving bullets. Meanwhile the excitement mounts as a lone defending plane enters stage left and ponderously advances on your fighter loosing off shots/bombs/bits of string or whatever. You must hit a coloured block in an amazingly detailed (I think not - ED) white tower, which would be quite exciting if it wasn't for the fact that you are told when you are on target - it might as well shoot the gun for you as well !

The fourth part is a sort of target shooting game with a few differences. One, the targets shoot back! Two, the targets come back to life after a short time. You control a little man who can move left or right and change the elevation of his gun/anti-tank weapon/ping pong pistol. The controls for this are too sensitive and we had great difficulty getting the same elevation twice in a row.

The fifth part involves throwing disc grenades/paper plates at a robot by bouncing them off a wall and catching them if they miss (overtones of Blockade here). We never actually saw this scene - something we will be eternally grateful for !

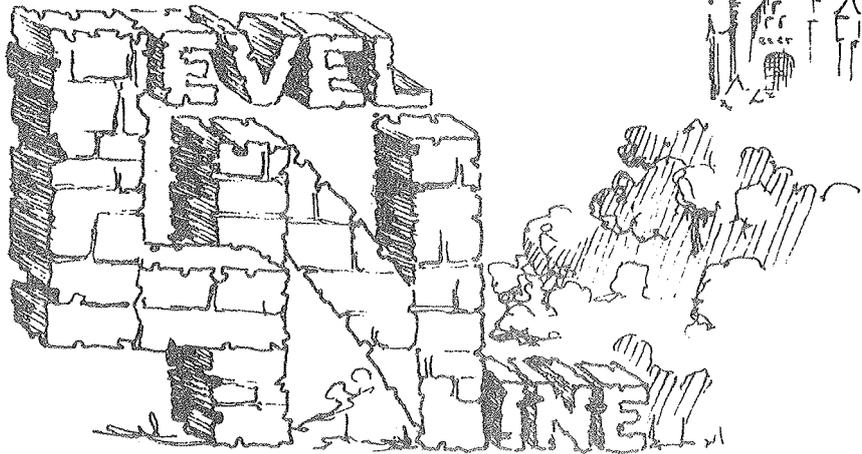
COMMENT:

GT. Almost as much fun as a cold bath!

≡Software Update

NB. This must be an anti-war game - it's so boring, anyone harbouring hostile intentions towards the Russians would be immediately converted to pacifism (or sent to sleep for an indefinite period!).

Game Content 65%
 Playability 55%
 Graphics 45%
 Sound 35%
 Value For Money 40%



Name : LORDS OF TIME
 Producer : Level 9 Computing
 Category : Adventure
 Price : £9.95

Those transdimensional villains, the Evil Timelords, have taken it into their heads to mess about with Earth's history in order to make the future very unpleasant indeed. You are made aware of their dastardly scheme via a wierd dream you have in which history seems to have gone bonkers. Your mission is to travel through nine time zones collecting a special item from each which together will destroy the despicably naughty (I would not buy a used car from these men) Timelords.

This must rank as one of the best adventures ever written, as it contains both a superb storyline and

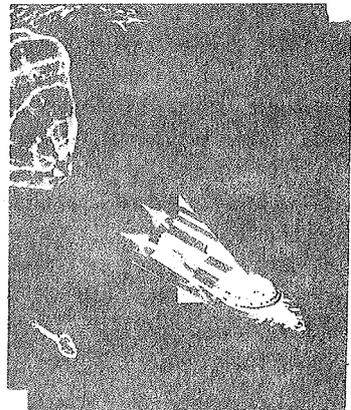
incredible atmosphere - to say nothing of the ingenuity of the puzzles! Level 9's offbeat humour/logic is present in large quantities, thus making Lords of Time challenging and hugely enjoyable. Most of the problems can be solved fairly easily, but some remain which require a lot more thought - the player must leave no avenue unexplored nor manure heap unturned. Death never seems to be too far away in this particular adventure, so saving the game regularly is highly advisable.

COMMENTS:

NB. I believe this to be THE BEST all round adventure I have ever played. Once started, I could not leave my keyboard until I had solved the last problem. If you are an adventure fan, then BUY IT !!!

GT. This is a game of outstanding quality. The parser is excellent, the descriptions are clear and easily visualised, and the manipulation of objects is extremely easy. One for the Christmas stocking (if you can wait that long!)

Presentation 90%
 Interaction 80%
 Game Content 85%
 Value For Money 85%



Name : SNOWBALL
 Producer : Level 9
 Category : Adventure
 Price : £9.95

In this science fiction adventure, you play the part of Kim Kimberley, a secret agent of dubious sex. Your mission is to safeguard the interstellar transport, Snowball 9. To achieve this you have been secretly placed on board in a special freezer coffin and will be awakened should anything go wrong. You begin the adventure waking inside your freezer coffin (I have often wondered what happend to the light in the fridge when I closed the door...now I know), something appears to be wrong, you must discover what it is and put it right before catastrophe strikes.

Software Update

The instructions/poster gives details of Kim's background as well as setting the scene for the adventure, including a Geopolitical summary for the year 2195. Very quickly you are confronted by the dreaded Nightingales, robot security guards who are not at all happy with passengers wandering about.

Unlike most other adventures, points are not gained for collecting items and placing them in a special location. Rather, points are scored by performing actions that help towards finding what is wrong and ultimately saving Snowball 9. You lose points for getting caught by the Nightingales (not a pleasant experience!) and getting killed in a variety of ways.

The atmosphere built by the game is very much like that of a first rate sci-fi novel. Tense in some parts, calculating in others and sprinkled with humour. The parser is good (though beware of typing an X at the start of a word!) but some of the descriptions can be a little hard to visualise in context.

COMMENT

GT. A classic adventure and the best sci-fi adventure I've played (though 2195 technology does seem to involve an enormous amount of button pushing!)
 • A definite favourite of mine.

NB. A very atmospheric adventure, containing loads of very logical problems which require a scientific approach (no "zap dragon with magic toothbrush" stuff here!). Sci-fi fans will love it.

Presentation 85%
 Interaction 80%
 Game Content 85%
 Value For Money 85%



Name : ADVENTURE QUEST
 Producer : Level 9 Computing
 Category : Adventure
 Price : £9.95

This is the second in Level 9's "Middle Earth" trilogy of adventures, and the sequel to "Colossal Adventure". It is set in the time long after Colossal, absolutely nothing having happened in the intervening period. However, just before terminal boredom threatens to set in, the Demon Lord Agaliarept livens things up a bit by stomping on the harvest and filling the place with orcs who spend their time headbutting the yokels. Your quest is to find the four Stones of the Elements which will enable you to enter Agaliarept's tower, and then use have to find) to give him the kicking he deserves.

We found this to be an extremely tough

adventure - after some initial exploration of the valley (the starting area) you are plunged into the hardest maze we have ever seen in an adventure - The Desert. Apart from containing loads of identical-looking locations, it also features a time limit (you will die of thirst if you get lost) and a marauding Sandworm which follows you around and will kill you if you let it get too close. You can lose loads of lives just trying to discover the routes through the maze to the other locations in the game, so using the saved game facility is an absolute must (even with the desert well behind you, the game remains wickedly difficult!) Hardened adventurers will find this a rewarding challenge - this is definitely not one for the novice.

COMMENTS :

NB. I found the storyline and atmosphere generated to be first rate, but the logical grouping of associated locations usually present in Level 9 offerings seems to be lacking here, making the exploration of the game a very heavy task.

GT. Just a Leeeetle bit hard this one! Definitely for adventure masochists everywhere.

Presentation 80%
 Interaction 60%
 Game Content 70%
 Value For Money 70%

Caverns of carnage

Name : DEVIL'S LAIR
 Producer : Loriciels
 Category : Arcade
 Price : £7.95

One stormy night, on the way home from the chip shop, you take a wrong turning and suddenly find yourself trapped in the mysterious and not at all nice Devil's Lair - a labyrinth of

sixteen connected caverns full (literally!) of traps, each cavern containing a key and a chest - every chest has to be opened in order to reveal the final exit. You also have to keep a close eye on your energy level - it decreases rapidly, and you have to constantly find and consume food and drink (present in the form of Big Macs and glasses of wine and only in

Software Update

extremely small quantities) to keep it from reaching zero (which spells death, sucker). You also die by colliding with the multitude of traps, which include flamethrowers, electric plates, landmines and iron spikes.

This is another platform game, in the same vein as "Jack's House of Cards" (see Issue 2) although visually it is far superior. The man is refreshingly easy to control, leaping nimbly from platform to platform and then into the middle of a disintegrator beam (which is appallingly easy to do). There are a variety of ways in which you can die, each of which is depicted in beautiful pastel graphics (its the only way to go...). Control is by joystick for left and right, with the space bar to jump.

To say this game is difficult would be an understatement of epic proportions - it is MEAN. Playing the game ourselves, we have only seen two of

the possible sixteen screens - and neither of us is bad at platform games ! However, once mastered, the elegance of the game unfolds. There are a large number of slides, the departure from which requires perfect timing if you

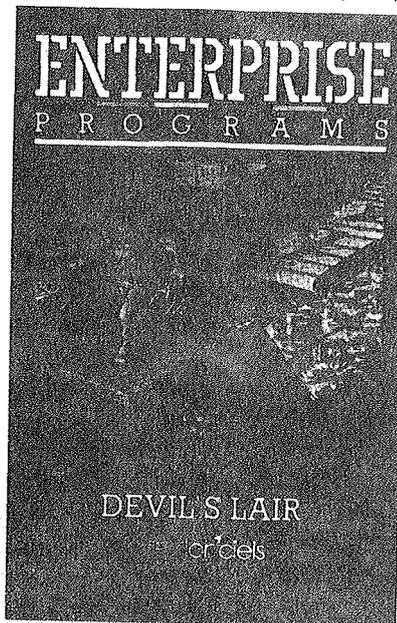
don't want to end up on the end of a spike! There is also a lift, but we've only seen that in the demo and so cannot comment on ease of use....

COMMENTS:

NB. This must rate as the most difficult platform game in the whole universe - after about three hours constant play I'd still only seen TWO of the screens ! Not a game you will lose interest in for a VERY LONG TIME.

GT. I challenge anyone to disprove my claim that it is impossible to complete (that is, open the chest on) more than about 3 screens of this game.

Game Content	80%
Playability	65%
Graphics	75%
Sound	40%
Value For Money	60%



Unrivalled

Name : SORCERY
 Producer : Virgin
 Category : Arcade
 Price : £7.95

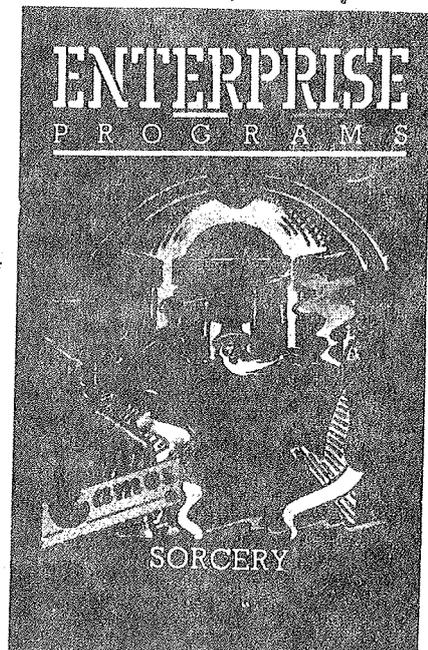
This is Ancient Britain, cloaked in the swirling mists of the Middle Ages. The evil Necromancer has imprisoned all but one of the nine goodly sorcerers in order to further his evil schemes and will soon kill them, thus making him undisputed Overlord and forcing the land to submit to his terrible rule. You play the part of the Last Free Sorcerer, and you must release all your colleagues in the fleeting minutes you have left before the nasty Necromancer zaps 'em into oblivion.

The sorcerer flies from screen to screen (there are 40 altogether) by use of the joystick - there is four-

way movement, and if the joystick is released the sorcerer will float slowly down to earth. Objects are picked up (or used if already held) by means of the space bar. The screens are the most beautifully illustrated we've yet seen on the Enterprise, with the impression of solid three dimensional figures portrayed by clever use of shading and colour. Game play is fast, furious and irresistibly addictive as you strive to discover what various objects' functions are, where the sorcerers are imprisoned and how to release them.

In addition to the time limit, you have to contend with quite a hefty smattering of nasties (two inhabit each screen) who are all out to drain your energy. Energy is displayed as a percentage - if this reaches zero then you snuff it ! You can also die by

coming into contact with any running water - evidently swimming wasn't



Software Update

considered an important wizardly discipline in those days ! There are a number of cauldrons scattered about... but you'll have to play the game to find out what they do !

All in all, "Sorcery" is a superbly implemented and hugely enjoyable game - at the time we first saw it (the PCW Show), it totally knocked the spots

off any other piece of Enterprise software available.

COMMENTS

NB. Forgive the pun, but this game is MAGIC ! It's superb fun to play (even if I do keep falling in the water !) and is a must for your games collection. Buy it !

GT. A colourful and amusing game filled with fast action and puzzles. Definitely one of the best games available for the Enterprise.

Game Content	80%
Playability	80%
Graphics	85%
Sound	55%
Value For Money	80%

Perilous potholing

Name : THE ABYSS
 Producer : AI Products
 Category : Arcade
 Price : £7.95

After the inevitable nuclear war (cynic ! - Ed.) the surface of the Earth was made uninhabitable. However, humanity still flourished deep below the surface due to an artificial environment they had created controlled by the RULER supercomputer (RULER is the Result of Ultimate Learning and Educational Refinement). I wonder if the people who reached this stage of ultimate learning were the same people who bombed the Earth ? Still, after a while RULER threw a major wobbler and wiped out most of the population, turning the undersurface world (now called "The Abyss") into a massive series of traps in order to kill the survivors. However, the last remaining bastion of humanity has made one final effort to survive - a sphere has been built which is capable of travelling through the dark vaults carrying one passenger - you. Your mission is to find and destroy RULER... however there is a long way to go.

The overriding aim of this game is survival. There are over 1,000 chambers which together form "The Abyss", each filled with homicidal robots (need I add that they're all programmed to kill you) and traps to negotiate. Some chambers are

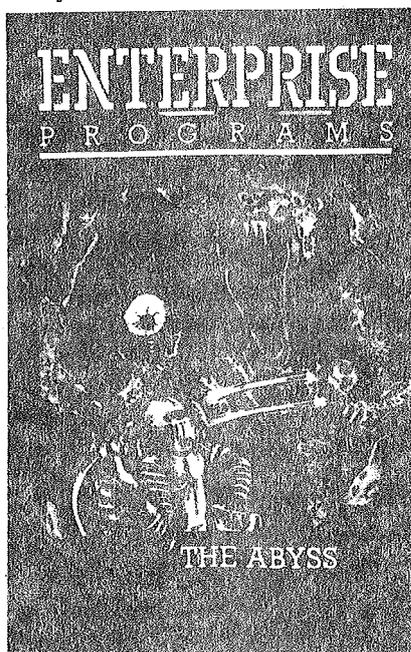
impassable until certain actions are performed - you must discover what is required, always avoiding the killer robots, photon thrusters and other assorted nastiness. Your sphere can operate in two modes - armed and defenceless (defenceless mode is better for negotiating the traps). You can alternate between these modes by use of "Swop" units which can be found scattered around in various chambers. Your "psyche", or energy level starts at the grand total of 888, but is soon reduced by colliding with the robots. It can be increased either by entering new rooms or by activating a "Psyche" unit - beware of over-energising though...

Control of the sphere is by internal joystick with the space bar to fire (when in armed mode). When in a room containing an object, a "Swop" unit or a "Psyche" unit, pressing the "I" key will do the relevant thing - pick up the object, increase your psyche (energy) or swop modes. If you pick up a bomb, it can be dropped by pressing the "U" key (but make sure you're not too close to the ground !).

The first feature of this game which immediately impressed us was the excellent music which plays throughout the game - yet another example of good use of the sound facilities available on the Enterprise. The graphics are fast and smooth, and although a bit gaudy are well up to the standard set by the top micro arcade games. We didn't get very far, but most chambers seemed fairly easy to negotiate with the marked exception of chambers containing photon thrusters. These are devilishly difficult, requiring precise timing (and a large degree of luck, considering how many of these chambers there must be between the start and the RULER computer). If you are an arcade fan, "The Abyss" is definitely a game to look out for.

COMMENTS

NB. I was too busy bopping to the music to be able to get very far ! It seems a bit rough that you die completely upon hitting a photon thruster beam. Overall very good



Software Update

GT. Killing the little alien robots appears to be rather pointless as another will immediately take its place! Otherwise it could be a great zapping game, as it is, it relies on fast reflexes and accurate positioning of your craft to get past the photon thrusters and other obstacles that bar your way. I found it fun and very

addictive, especially the music, even if it did drive some other members of the household mad (sorry Tim!).

Game Content	60%
Playability	65%
Graphics	70%
Sound	75%
Value For Money	70%

Enterprise 'Starter Software' release!
! "Eddie goes Adventuring"
...hmm...could be worth money to some crackpot...

Presentation	10%
Interaction	5%
Game Content	15%
Value For Money	25%

Bundle of bodes

Name : ADVENTURE PACK
Producer : Microdeal
Category : Adventure ????
Price : £6.95

This package contains not one but THREE text adventure games - Williamsburg, Ultimate and Castle Dracula. Superb value for £6.95 you might think... unfortunately not! These programs are the pits, man! Written in jolly old IS-BASIC and converted from the original Dragon 32 classics (ho ho!) they seem to have acquired some classic bugs and spelling mistakes along the way.

In Williamsburg, someone has hidden the fabled Golden Horseshoe somewhere in the colonial city. Your job is not only to find the treasure but also to find out which words the parser will actually recognise! Also, adventurers with any morals will find it impossible to complete this adventure as you are required to blow away defenceless inhabitants with a cannon in order to win, to say nothing of stealing almost anything you can find that will move!!! The puzzles in this game are so obvious it's untrue, the problem really being expressing the action you wish to perform in a manner the program will understand. In its defence, the program was written with 10 year olds in mind but the puzzles insult even their intelligence while the useless parser would make it nigh impossible for them to do much at all.

Neil Blaber
Gary Thomson

Castle Dracula is very similar in presentation to Williamsburg and suffers the same dreaded parser palsy. You are cast as Baron Von Helsing, who wakes up to find that his wife has done a runner and shackled up with old Draccie. This is another adventure in which you have to bump off either the local clergy or the shopkeeper in order to win - not really an example for ten year olds, is it?

Ultimate differs slightly in that there aren't really any puzzles to solve (good adventure, eh kids??). You start off in a market with 250 strength points and £300 - you can then buy various pieces of equipment, some of which will help you to bash up the nasties (such as the terrible "Bndit" - spelling mistake quoted from the game!) or allow you to enter otherwise inaccessible rooms. The nasties knock strength points off you unless you use the correct item against them - however once you get into a combat you are prompted for 'Which item?' and are unable to do an inventory to find out which items you have! I think I've said enough...

COMMENTS

NB. Blehhhhh!!! These are the worst excuses for adventure games I've ever seen. Take my advice, spend an extra three quid and get a decent (Level 9) adventure.

GT. If I hadn't seen otherwise, I could have sworn this was an

COMMENT

As we hoped, the quality of software has improved drastically (with a few exceptions) in recent months. Even more good news is that there will be no more Spectrum dumps financed by Entersoft. Programs in future will have to be at least Commodore and Amstrad quality before being financed. We see this as a good move as the comparisons with the Spectrum can only do this computer harm. It is not 'a good Speccy emulator', it is a machine in its own right, having much more to offer than any of its rivals. Enterprise specific software is still very thin on the ground (transparent in fact), but this should change as software houses learn more about the machine.

Recently we have had difficulty in obtaining production copies of Enterprise products for review purposes. Although we can see the problems associated with loaning large numbers of products to various outlets (press, distributors etc.), we feel somewhat annoyed that the User Group seems to have been grouped alongside these and is thus getting the 'cold shoulder' from Enterprise. After all, we do represent their customers!

It seems that the policy of giving us the two-week loan of new products to review no longer exists, thus rendering it impossible for us to perform this function (it will take longer than a two hour session in the Enterprise offices to properly review the Speakeasy, Enterprise!).

Of the Enterprise's many outstanding features, graphics are bound to be at the top of most peoples' lists. Screen coordinates, however, can pose some problems.

The Enterprise can support screen resolutions varying from 672 by 245 down to 2 by 9 (from Basic that is). The way screen coordinates are defined in IS-Basic is dependent on the size of the screen being used and not upon the number of pixels across and down. This means it is very simple to change the colour resolution of a picture without having to change any of the screen coordinates. This can make it difficult to identify which coordinates are associated with different pixels (for example, several screen coordinates may refer to the same pixel).

All character spaces contain 9 real vertical pixels (except in interlace mode when you have 18). Screen Y coordinates are 36 (9x4) high which gives a ratio of one vertical pixel to four screen Y coordinates.

The horizontal characters spaces vary in pixel content depending on the resolution and colour mode, but if you remember that each increase in colour halves the pixel resolution (2 to 4, 4 to 16 and 16 to 256), you can work out the pixels per screen for any colour resolution combination. In high resolution 2 colour mode there are 16 pixels and in low resolution two colour mode there are 8 pixels across. In order to simplify this, here are some formulas to work out the pixel and coordinate resolution, using the screen size in character spaces together with the resolution and colour mode.

HIGH RESOLUTION PIXELS WIDE

CX = CHARACTERS WIDE
CY = CHARACTERS HIGH

Co-ordinates vs PIXELS

2 COLOUR MODE = CX*16
4 COLOUR MODE = CX*8
16 COLOUR MODE = CX*4
256 COLOUR MODE = CX*2

(Low resolution have half this number of pixels)

COORDINATES RESOLUTION WIDE

COORDINATES WIDE = CX*32 (for all resolutions & modes)

PIXELS HIGH

PIXELS HIGH = CY*9 NON INTERLACE
PIXELS HIGH = CY*18 INTERLACE

COORDINATE RESOLUTION HIGH

PIXELS HIGH = CY*36

Below is a program which will calculate the number of pixels and coordinates for any size video page.

```

100 PROGRAM EXAMPLE
105 CLEAR TEXT
107 PRINT
108 PRINT "Type in the details of
the screen that you want (use
values from the machine options)
:"
109 PRINT
110 INPUT PROMPT "Characters wide: "
:CHAR X
120 INPUT PROMPT "Characters High: "
:CHAR Y
130 INPUT PROMPT "Video Mode: ":VID_
MODE
140 INPUT PROMPT "Video Colour: ":
VID COL
145 PRINT
170 SELECT VID_COL
190 CASE 0
190 LET PIXEL_X=CHAR_X*16
200 CASE 1
210 LET PIXEL_X=CHAR_X*8
220 CASE 2
230 LET PIXEL_X=CHAR_X*4
240 CASE 3
250 LET PIXEL_X=CHAR_X*2

```

```

270 END SELECT
290 SELECT VID_MODE
300 CASE 0,2
310 PRINT "These are text modes!"
320 CASE 1
340 CASE 5
350 LET PIXEL_X=PIXEL_X DIV 2
360 CASE 15
370 LET PIXEL_X=CHAR_X*8
380 END SELECT
400 LET PIXEL_Y=CHAR_Y*9
420 LET COORD_X=CHAR_X*32
430 LET COORD_Y=CHAR_Y*36
450 PRINT "Horizontal pixels (X):";
PIXEL_X
460 PRINT "Vertical pixels (Y):";
PIXEL_Y
470 PRINT
480 PRINT "Screen coordinates:"
490 PRINT " X: 0 -";COORD_X-1
500 PRINT " Y: 0 -";COORD_Y-1

```

HORIZONTAL COORDINATE STEPS

High Resolution:

2 COLOUR = 2
4 COLOUR = 4
16 COLOUR = 8
256 COLOUR = 16

Low Resolution:

2 COLOUR = 4
4 COLOUR = 8
16 COLOUR = 16
256 COLOUR = 32

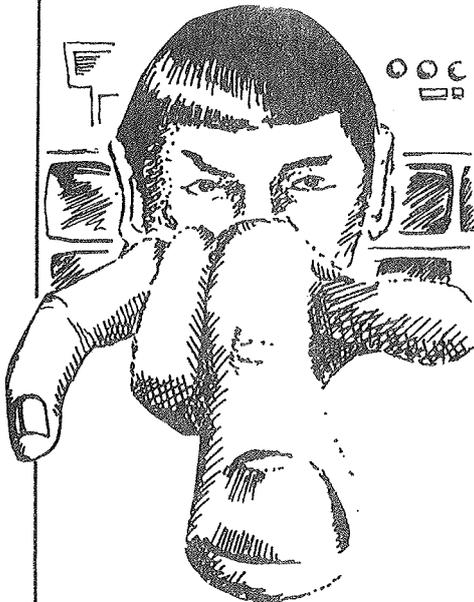
VERTICAL COORDINATE STEPS

All resolution & colour modes = 4
All resolution & colour modes
(INTERLACE) = 2

NOTE. The numbering system starts from 0 - therefore although the maximum number of coordinate positions is 1280, the highest coordinate is only 1279.

The ATTRIBUTE screen uses the same formula as the high resolution 4-colour mode.

TIM BOX



One of the Enterprise computer's main selling points must be its superior graphics capabilities, especially in comparison with other 8 bit microcomputers. It manages to achieve considerable flexibility in this area by using a custom designed graphics chip 'NICK'. It is the aim of this article to introduce the reader to this chip, and show ways of manipulating the display which are simply not possible using standard IS BASIC commands.

The actual display that is generated by NICK depends upon two areas of memory. The first area is where a table of values that determine such things as margins, colours etc. is located. This table is known as the 'Line Parameter Table'. The second area of memory is where the screen data is stored - characters, coloured pixels etc, the 'Information Area'.

The Line Parameter Table, or LPT, consists of groups of 16 bytes that have the following significance.

BYTE0: The number here represents how many pixel lines (scanlines) are to be affected by the following 15 bytes. The actual number of scanlines is given in a two's complement form,

The Line Parameter Table

but bit 7 of the byte being a magnitude bit, not a sign bit!

E.g. 247 here would represent 9 pixel lines (the usual setting).

BYTE1: This is the modebyte and defines the video mode of the scanlines.

BYTE2: Left margin byte. The bottom 6 bits define the left hand margin for the lines. Usually these bits are set to produce 11.

BYTE3: Right margin. As BYTE2 but for the right hand margin (usually set to 51).

BYTE4, BYTE5: Define where in video memory the data for the scanlines is, i.e. the information area for the lines.

BYTE6, BYTE7: Used to point to the character font for the text modes, or a pointer to pixel information when using the attribute mode.

BYTE8 to BYTE15: Define palette colours 0 to 7 for the scanlines.

For the purposes of this article I shall not make any use of BYTE1 to BYTE3, or BYTE6 and BYTE7.

The LPT has at default 34 groups of these 16 bytes, only 28 may be used for display purposes. The other 6 groups describe blank lines that define the border and end of the video frame. It should be noted here that the total number of scanlines is always 312, a number greater or less than this will cause the display to become unsteady (an exception to this is when interlace is being used).

To manipulate the display, we need to know where the LPT is stored in memory. To find this out we need to have a basic understanding of the internal architecture of the Enterprise. The Z-80 microprocessor at the heart of the computer can only address 64K bytes, so how can the Enterprise address 4M bytes? The

answer to this is that the computer uses 256 blocks, or segments, of 16K, with the Z-80 using any 4 segments at a time (internal memory decoding being achieved by another custom chip 'DAVE').

The NICK chip can only use segments 252 to 255, so called video memory. (this is why the Enterprise 64 is slower than the Enterprise 128, accesses to these segments are slower due to the interaction of NICK with the Z-80). Thus all graphics data, i.e. the LPT and information area must reside in these segments.

Each segment has 16K of memory, thus actual memory can be described by a segment number (0 to 255), and a segment memory address (0 to 16K). A rather special segment is number 255, the so called system segment. In this 16K block reside several 'system variables'. At address 255,16372 and 255,16373 is one such variable called LP_POINTER. This contains the address of the start of the LPT in segment 255. Thus by using

```
100 LET LPT_ADDR=REM(SPEEK(255,16372)+
256*SPEEK(255,16373),16384)
```

LPT_ADDR will equal the address of the first byte of the LPT. We now have enough information to start experimenting with the display.

LISTING 1 is a demonstration of how to move screen memory about. Run the program and press 'enter' when the question mark appears. You will see line 5 on the screen replaced with line 1!! The program uses a text screen, but the same would apply if we had used a graphics screen.

LISTING1:

```
100 PRINT AT 1,15:"HELLO THERE"
110 PRINT AT 5,15:"I'M LINE 5"
```

Advanced Programming

```

120 LET LPT_ADDR=REN(SPEEK(255,16372)
+256*SPEEK(255,16373),16384)
130 INPUT A$
140 LET D1=SPEEK(255,LPT_ADDR+16+4)
150 LET D2=SPEEK(255,LPT_ADDR+16+5)
160 SPOKE 255,LPT_ADDR+5*16+4,D1
170 SPOKE 255,LPT_ADDR+5*16+5,D2

```

Now try LISTING 2. Here a graphics screen is used to demonstrate how we can change the colour of the inks on separate lines. So although we are using a 4 colour mode, it is possible to have 60 colours on screen at once. Although at the moment the inks can be different only between character lines, we shall see how we can overcome this in the next listing.

```

100 GRAPHICS HIRES 4
110 FOR T=1 TO 120
120 SET INK RND(3)+1
130 PLOT RND(1279),RND(719);
140 NEXT T
150 LET LPT_ADDR=REN(SPEEK(255,
16372)+256*SPEEK(255,16373),16384)
160 DO
170 FOR T=1 TO 20
180 SPOKE 255,LPT_ADDR+T*16+9,
RND(255)
190 SPOKE 255,LPT_ADDR+T*16+10,
RND(255)
200 SPOKE 255,LPT_ADDR+T*16+11,
RND(255)
210 NEXT T
220 LOOP

```

We know that BYTE0 refers to the number of scanlines to be affected by BYTE1 to BYTE15. Thus if we alter the value of BYTE0, but ensure that the total number of scanlines is 312, we will be able to introduce colour differences between pixel lines, not only character lines as before.

Using this method I have managed to produce 11 background colours on a 4 colour display (i.e. 3 foreground colours may still be used) to create the effect of a sunset. A very important part of the program occurs between lines 250 and 320. This part of the program ensures that BYTE4 and BYTE5 of each group of 16 bytes in the LPT are pointing to the correct area of video memory. This is achieved as follows:-

Take the first group of 16

bytes in the LPT that refer to the graphics screen, i.e. LPT_ADDR+16 (LPT_ADDR refers to the status line). Then BYTE4, BYTE5 point to the first byte in video memory of the graphics screen, call this ADDR (graphics information is usually held in a continuous block of memory). Each scanline on a default hires graphics screen corresponds to 80 bytes of data. Thus to display the information described by the 16 byte area in the LPT requires 80*number of scanlines (BYTE0). Hence for the next group of 16 bytes in the LPT, BYTE4, BYTE5 must point to ADDR+80*number of scanlines of previous group. This process must be repeated for all groups of 16 bytes that describe the graphics screen.

LISTING3:

```

100 GRAPHICS HIRES 4
110 SET COLOUR 1,RED
120 RANDOMIZE
130 LET LPT_ADDR=REN(SPEEK(255,
16372)+256*SPEEK(255,16373),16384)
140 !
150 ! *** Change number of scanlines
to be affected by a ***
160 ! *** particular background
colour ***
170 !
180 FOR NUM=1 TO 10
190 READ NUM_LINES
200 SPOKE 255,LPT_ADDR+NUM*16,256-
NUM_LINES
210 NEXT NUM
220 !
230 ! *** Ensure that BYTE4, BYTE5
point to correct memory area ***
240 !
250 RESTORE 480
260 FOR NUM=2 TO 20
270 LET ADDR=SPEEK(255,LPT_ADDR+(
NUM-1)*16+4)+256*SPEEK(255,LPT_
ADDR+(NUM-1)*16+5)
280 READ NUM_LINES
290 LET NEW_ADDR=ADDR+NUM_LINES*80
300 SPOKE 255,LPT_ADDR+NUM*16+4,
REN(NEW_ADDR,256)
310 SPOKE 255,LPT_ADDR+NUM*16+5,
INT(NEW_ADDR/256)
320 NEXT NUM
330 !
340 ! *** Run through colours to
show sunset effect ***
350 !
360 CALL MOUNTAINS
370 FOR J=1 TO 7
380 FOR NUM=1 TO 10

```

```

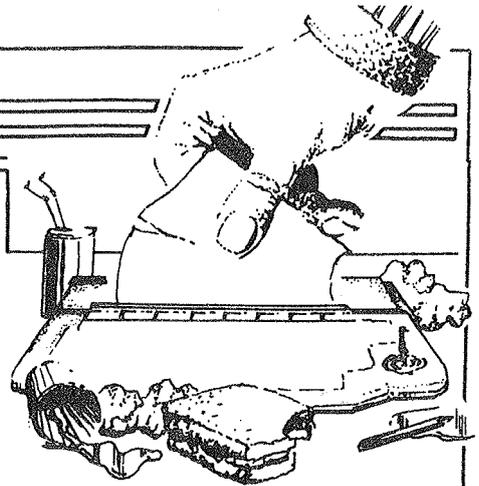
390 READ COLOUR
400 SPOKE 255,LPT_ADDR+NUM*16+8,
COLOUR
410 NEXT NUM
420 WAIT 1
430 NEXT J
440 END
450 !
460 ! *** Number of scanlines
affected by particular colour ***
470 !
480 DATA 22,17,13,11,9,6,5,3,2,1,9,
9,9,9,9,9,9,9
490 !
500 ! *** Colours to be used as
background colours ***
510 !
520 DATA 72,1,65,9,73,201,89,217,75,
255
530 DATA 8,72,1,65,9,73,201,89,217,
75
540 DATA 64,8,72,1,65,9,73,201,89,
217
550 DATA 0,64,8,72,1,65,9,73,201,89
560 DATA 0,0,64,8,72,1,65,9,73,201
570 DATA 0,0,0,64,8,72,1,65,9,73
580 DATA 0,0,0,0,64,8,72,1,65,9
590 !
600 ! *** Draw some 'mountains',
shows that standard plot works
***
610 ! *** Note, do not use any !!
colour changes unless via a ***
620 ! *** similar routine as lines
720 to 740 ***
630 !
640 DEF MOUNTAINS
650 SET INK 2
660 FOR T=0 TO 1279 STEP 30
670 PLOT T,RND(400)+260;
680 NEXT T
690 PLOT 1279,RND(400)+260
700 PLOT 500,200,
710 PLOT PAINT
720 FOR NUM=1 TO 20
730 SPOKE 255,LPT_ADDR+NUM*16+
10,0
740 NEXT NUM
750 END DEF

```

We have seen that various effects may be created by manipulating the LPT, and even more effects are hinted at. For example, instead of using the plot command, you could poke data directly to the screen. Horizontal pixel scrolls may be induced (adjust the values of all BYTE4, BYTE5 pairs) etc. Experiment with the display, and write in to PRIVATE ENTERPRISE with your discoveries.

Peter Walton

Home Produce



The following program allows experimentation with envelopes with the minimum of fuss. The operator first sets the volume pattern of the envelope using the joystick to draw with (the cursor moves in increments of ten units if the space bar is held down at the same time). When a phase is completed press the "N" key - if the volume part of the envelope is finished press "F".

The program will then move on to allow the pitch change to be set for each phase of the envelope; again when the pitch for any phase is correct press "

"N" to move on to the next phase. The program will automatically move on when the pitch for the last phase has been defined.

The user will then be given the option of playing the envelope he has just drawn, designing a new envelope or redefining the old one.

If the user chooses to play the envelope, the initial pitch and duration of the sound will have to be given, as well as the value of the units which define pitch changes. There is also an option to print out the phases of the envelope as they would appear in an envelope command.

NOTE: The units of duration are "ticks", where 50 ticks = 1 second. Pitch definition is in semitones, with 37 being equivalent to middle C.

The only lines which may cause some confusion are 670 to 700 which actually set up the envelope by sending an escape sequence to the sound driver. This method is necessary due to line length limits within IS-Basic. The sequence takes the form:

```
esc E [1] [2] [3] [4] [5] [6] [7]
```

Where [1] is the envelope number 0-254

[2] is the total number of phases in the envelope
1-21

[3] is the number of phases before the first release phase. This should be set to 254 if there are to be no release phases.

[4] is the pitch change. A signed 16-bit number between -32768 and 32767, this represents units of 1/512 semitones.

[5] is the left volume change from -63 to 63.

[6] is the right volume change.

[7] is the phase duration 1 to 255.

Obviously [4] to [7] must be repeated as many times as there are phases to the envelope. Another escape sequence which may be of interest allows a sound to be produced, and takes the form:

```
esc S [1] [2] [3] [4] [5] [6] [7] [8]
```

Where [1] is the envelope number to be used. 255 represents the built-in envelope, producing constant pitch and volume.

[2] is the initial pitch, 0-65535, in 1/512 semitones.

[3] is the overall left volume, 0-255.

[4] is the overall right volume.

[5] is the sound style, 0-255.

[6] is the sound source, 0-3.

[7] is the duration, 0-65535, given in 1/50 seconds.

[8] is a flags byte. The first 2 bits show the SYNC value, and the last bit is set to allow the sound to override any other sounds in the sound queue. All other bits should be reset to 0.

```
100 PROGRAM "ENVELOPE_GENERATOR"
110 !
120 ! ENVELOPE GENERATOR
130 ! (c) D.M.RACE
150 !
160 SET SOUND BUFFER 21
```

Home Produce

```
170 SET KEY CLICK OFF
180 STRING DUMMY$
190 NUMERIC P(1 TO 21),V(1 TO 21),D(1 TO 21),GRAPH(1 TO 3,0 TO 21)
200 LET GRAPH(3,0)=96:LET GRAPH(1,0)=105
210 !
220 ! MAIN BODY OF PROGRAM
230 !
240 DO
250 CALL DRAW
260 LET XX,X=96:LET YY,Y=105:LET NN=21
270 CALL DEFINE(0)
280 DO
290 CLEAR TEXT
300 PRINT AT 1,1:"(P)lay envelope"
310 PRINT AT 2,1:"(C)hange envelope"
320 PRINT AT 3,1:"(D)efine new envelope"
330 LOOK [105:DUMMY
340 LET DUMMY$=UCASE$(CHR$(DUMMY))
350 SELECT CASE DUMMY$
360 CASE "P"
370 CALL PLAY
380 CASE "C"
390 CALL DEFINE(-1)
400 CASE "D"
420 CASE ELSE
430 END SELECT
440 LOOP UNTIL DUMMY$="D"
450 LOOP
460 !
470 ! PLAYS ENVELOPE
480 !
490 DEF PLAY
500 CLEAR TEXT
510 INPUT PROMPT "Initial Pitch: ":PIT
520 INPUT PROMPT "Duration (Ticks): ":DUR
530 INPUT PROMPT "Pitch Change Units: ":P_CH
540 LET V(1)=(GRAPH(1,1)-105)/4
550 LET P(1)=(GRAPH(2,1)-411)*32
560 LET D(1)=(GRAPH(3,1)-96)*(DUR/(GRAPH(3,NN)-96))
570 FOR N=2 TO NN
580 LET V(N)=(GRAPH(1,N)-GRAPH(1,N-1))/4
590 LET P(N)=(GRAPH(2,N)-GRAPH(2,N-1))*32*P_CH
600 LET D(N)=(GRAPH(3,N)-GRAPH(3,N-1))*(DUR/(GRAPH(3,NN)-96))
610 NEXT
620 FOR N=1 TO NN
630 LET D(N)=CEIL(D(N))
640 NEXT
650 CLEAR SOUND
660 CLEAR ENVELOPE
663 !
664 ! THE FOLLOWING LINES DEFINE
665 ! AN ENVELOPE BY PRINTING TO
666 ! THE SOUND CHANNEL.
667 !
670 PRINT [103:HEX$("1B,45,1")&CHR$(NN)&CHR$(255);
680 FOR N=1 TO NN
690 PRINT [103:WORD$(P(N)) WORD$(V(N))(1) WORD$(V(N))(1) WORD$(D(N));
700 NEXT
710 SOUND DURATION DUR,ENVELOPE 1,PITCH PIT
```

Home Produce

```
720 CLEAR TEXT
730 PRINT AT 1,1:"DO YOU WANT A LISTING? "
740 LOOK [105:DUMMY
750 LET DUMMY%=UCASE$(CHR$(DUMMY))
760 IF DUMMY%="Y" THEN CALL LIST_ENV
770 END DEF
780 !
790 ! DEFINES ENVELOPE
800 ! CHANGE=0 FOR NEW ENVELOPE
810 !     =-1 TO REDIFINE OLD ONE
820 !
830 DEF DEFINE(CHANGE)
840 CLEAR TEXT
850 SET INK 3+CHANGE
860 LET X_POS=96:LET Y_POS=105
870 PRINT AT 1,1:"Total Duration="
880 PRINT AT 2,1:"Phase Duration="
890 PRINT AT 3,1:"     Volume="
900 FOR N=1 TO NN
910 IF CHANGE THEN
920 LET X,XX=GRAPH(3,N):LET Y,YY=GRAPH(1,N)
930 PLOT X_POS,Y_POS;XX,YY
940 END IF
950 CALL PLOT_VOL
960 IF X_POS=1264 OR UCASE$(DUMMY%)="F" THEN LET NN=N:LET N=21
970 NEXT
980 CLEAR TEXT
990 LET X_POS=96:LET Y_POS,Y,YY,GRAPH(2,0)=411
1000 SET INK 2-CHANGE
1010 PRINT AT 1,1:"Pitch Change Overall="
1020 PRINT AT 2,1:"Pitch Change This Phase="
1030 FOR N=1 TO NN
1040 LET X,XX=GRAPH(3,N)
1050 IF CHANGE THEN LET Y,YY=GRAPH(2,N)
1060 CALL PLOT PIT
1070 NEXT
1074 !
1075 ! REDRAWS ENVELOPE IF OLD ONE
1076 ! HAS BEEN REDEFINED.
1078 !
1080 IF CHANGE THEN
1090 CALL DRAW
1100 SET LINE MODE 0
1110 FOR C=1 TO 2
1120 SET INK 4-C
1130 FOR CC=1 TO NN
1140 PLOT GRAPH(3,CC-1),GRAPH(C,CC-1);GRAPH(3,CC),GRAPH(C,CC)
1150 SET LINE MODE 3
1160 NEXT
1170 NEXT
1180 END IF
1190 END DEF
1200 !
1210 ! PLOTS VOLUME PART OF ENVELOPE
1220 !
1230 DEF PLOT_VOL
1240 LET DUMMY%=""
1250 DO UNTIL XX>X_POS AND DUMMY%="F"
1260 IF JOY(0) BAND 16 THEN
```

Home Produce

```

1270 LET M=10
1280 ELSE
1290 LET M=1
1300 END IF
1310 LET XX=XX+M*4*((JOY(0) BAND 3)=2)-((JOY(0) BAND 3)=1))
1320 LET YY=YY+M*4*((JOY(0) BAND 12)=4)-((JOY(0) BAND 12)=8))
1330 IF XX<X_POS THEN LET XX=X_POS
1340 IF XX>1264 THEN LET XX=1264
1350 IF YY<105 THEN LET YY=105
1360 IF YY>357 THEN LET YY=357
1370 PLOT X_POS,Y_POS;X,Y
1380 PLOT X_POS,Y_POS;XX,YY
1390 PRINT AT 1,16:(XX-96)/4
1400 PRINT AT 2,16:(XX-X_POS)/4
1410 PRINT AT 3,16:(YY-105)/4
1420 LET X=XX:LET Y=YY
1430 LET DUMMY%=INKEY$
1435 IF DUMMY%<>" " THEN LET DUMMY%=UCASE$(DUMMY%)
1440 LOOP UNTIL XX>X_POS AND DUMMY%="N"
1450 PING
1460 SET LINE MODE 0
1470 PLOT X_POS,Y_POS;XX,YY
1480 SET LINE MODE 3
1490 LET X_POS=XX:LET Y_POS=YY
1500 LET GRAPH(1,N)=Y_POS:LET GRAPH(3,N)=X_POS
1510 END DEF
1520 !
1530 ! PLOTS PITCH PART OF ENVELOPE
1540 !
1550 DEF PLOT_PIT
1560 PLOT X_POS,Y_POS;X,Y
1570 DO UNTIL UCASE$(INKEY%)="N"
1580 IF JOY(0) BAND 16 THEN
1590 LET M=5
1600 ELSE
1610 LET M=1
1620 END IF
1630 IF JOY(0) BAND 4 THEN LET YY=YY-4*M
1640 IF JOY(0) BAND 8 THEN LET YY=YY+4*M
1650 IF YY>715 THEN LET YY=715
1660 IF YY<107 THEN LET YY=107
1670 PLOT X_POS,Y_POS;X,Y
1680 PLOT X_POS,Y_POS;XX,YY
1690 LET X=XX:LET Y=YY
1700 PRINT AT 1,22:(YY-411)/16;" "
1710 PRINT AT 2,25:(YY-Y_POS)/16;" "
1720 LOOP
1730 PING
1740 SET LINE MODE 0
1750 PLOT X_POS,Y_POS;XX,YY
1760 SET LINE MODE 3
1770 LET X_POS=XX:LET Y_POS=YY
1780 LET GRAPH(2,N)=Y_POS
1790 END DEF
1800 !
1810 ! PLOTS GRAPH BACKGROUND
1820 !
1830 DEF DRAW
1840 GRAPHICS HIRES 4
1850 SET LINE MODE 3
1860 SET PALETTE 0,WHITE,CYAN,GREEN
1870 PLOT 96,18;96,715,16,105;1264,105
1880 FOR N=105 TO 716 STEP 16
1890 PLOT 88,N;96,N
1900 NEXT
1910 SET LINE STYLE 5
1920 PLOT 88,411;1279,411
1930 SET LINE STYLE 9
1940 PLOT 88,357;1279,357
1950 SET LINE STYLE 1
1960 END DEF
1970 !
1980 ! LISTS ENVELOPE DESCRIPTION
1990 !
2000 DEF LIST_ENV
2010 CLEAR TEXT
2020 PRINT AT 1,1:"PITCH","LEFT","RIGHT","DURATION"
2030 FOR N=1 TO NN
2040 PRINT AT 2,1:"
2050 PRINT AT 2,1:ROUND(P(N)/512,2),V(N),V(N),D(N)
2060 PRINT AT 3,1:"PRESS ANY KEY TO CONTINUE"
2070 LOOK [105:DUMMY
2080 PRINT AT 3,1:"
2090 NEXT
2100 END DEF

```

**DAVE
RACE**

Here at IEUG headquarters (Viva el Presidente!) we are trying to set some kind of file naming standard. It is a little bit difficult determining what a file actually is when it is called "dubrie_1". The standard we are adopting is that of a filename extension after the filename consisting of a full stop, followed by an extension code. The codes are as follows:

```

filename.BAS - BASIC
filename.PAS - Pascal source code
filename.LSP - Lisp environment
filename.FTH - Forth source code
filename.WP \ Word Processor files
filename.DOC / (Saved using SAVE function)
filename.ASM - Assembler source
filename.YXT - ASCII file
                (eg. WP files saved using PRINT option)
filename.EXE \ Executable code
filename / (eg game/applications program)
filename.REL \ Exos relocatable module
filename.X /

```

L. VAN VEEN
VELDSPANTSTRAAT 278
9743 XS GRONINGEN

100
The Orange
of Holland

POST
23. IV. 86
P. O. BOX 8
4164 ZA

160
SCHEFF
FR 1955

ROOGERHEIDE
NEDERLAND