

AMOS



NEWSLETTER

VOLUME 13, Summer 1993

Editorial

Rendering
Graphics
Page 2

Hi 'yall. I've been receiving interesting feedback lately from members and friends on what people are doing with AMOS, their gripes, and other comments. This is all in the new Letters/Forum page. This has been very popular and the most interesting of all the feedback I've received (good and bad) has been included.

Code Clinic
Pages 3 To 4

We've got a real mix this issue, however, that doesn't mean the usual stuff has been left out. Keep letting us know what you want to read. We'll cover it all eventually

Club Forum
Pages 7 To 8

I also have something rather special as well. Chris Maquire of Lawson has sent us an article on how to build your own scanner and write software in AMOS for it! Excellent stuff!

Making a
Scanner!
Pages 8 To 10

Those of you who get around on Bulletin Boards may have noticed the Public Domain Catalogue Disk in an AMOS file section. This is the most complete list of the AMOS PD Library. It will be continually updated and uploaded to various BBS's including Predators, Amiga File Server and Blade. More inside on that.

Also this issue is another Programmers competition. It was suggested by many that a regular competition would give everyone something to aim for. You'll find all the details on page 13.

There's been some interesting things happening in the programmer's world of AMOS over the last few months. We saw the release of AMOS Professional, a little bugged but alive. When the excitement died down, people began to ask me about compatibility with AMOS 3D and the Compiler. At the time, the new AMOS Pro couldn't accept any extensions from the

old AMOS 1.34. A bug fix should be here shortly and will be available as an updater. I received an interesting comment recently from someone who was a little annoyed about companies that release products (such as AMOS Pro) that are bugged. Unfortunately (and this will never change) a product as vast and complex as the AMOS language was programmed by one person. He in turn has had to construct over 700 all purpose commands and instructions to create what is our most dearly loved language, and to be bug free is almost impossible. Update v1.1 will soon be available and will fix a list of bugs longer than your arm.

I don't expect this to be our last updater (as has always been the way), but updates are cheap and readily available. We can be expecting (and are being worked on) extensions to support the AGA chipset, Workbench 3.X, CD-ROM, and so on. AMOS will always continue to follow the new trends of the Amiga.

We already have 24 bit support, DOS access, Printer/port access, Three dimensional modelling and manipulation, IFF Animation playback, Hypertext Support... and the list continues.

I feel that AMOS will be a major part of the Amiga for some time to come.

Wayne.

PROGRAMMER'S
CHALLENGE
Back By Popular Demand!
Page 13

Editor & technical advice:
Wayne Johnson
Assistance & PD:
Amanda Egan
Sponsors:
Pactronics Pty Ltd
Genetix Software
Proof Reading:
Max Walters
Jock McQue
Scanning & Printing
Neil Miller
Publishers:
Budget Instant Printing
Distribution:
Wendy Leach

AMOS 1.34
MIXED BAG
OF INFO
Pages 10-13

Adventure
Writing
Pages 5 To 6

NEW PD
Pages 19-20

AMOS ROLE PLAYERS
CLUB is Back! Page 16

Hi From
François
Page 8

Rendering GFX for games

We all use packages such as Dpaint to design Graphics and sprites for games. Very seldom do people render their BOBs in 3D! Tex shows you how to design amazing sprites for games.

Have you ever wondered how commercial games houses manage to make their graphics so "Visually Correct"? The graphics I am talking about are the spaceships, aliens and other Flying-Running objects that make up a game. If you get to see Games Mags, they occasionally show you screens of the animation frames that make up a "Ships" movements. If we take a vertical or horizontal shoot-em-up as an example, in some cases we have a single non-changing top view of the ship. If you move left, you simply move left with no change in the ship to reflect that. In the more advanced games we start to see that the ship actually banks to the left or right as you move, thus giving you a more realistic "Flight"!

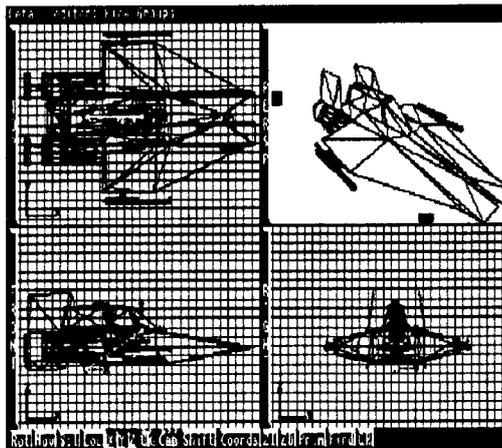
Have you ever sat down and tried to draw a ship that looks good in a 32 x 32 PIXEL square? Some of you with graphics talent will not find it too hard, just tedious. Then after you have your top view, sitting there and trying to draw the "Banking" frames becomes a nightmare! Trying to make sure the detail is correct and that everything is still to scale is not easy. That is if you do it the old fashioned way! You know, the way we just discussed using something like DPaint or even the Sprite Editor.

Well you will now find that quite a few games houses are now making their job easier by putting 3D modelling and rendering software to use. Programs like Imagine II, Real 3D and others allow you to "Build" your ship from the ground up with as much detail as you like with relative ease. You can see how your ship looks from 3 different angles and you can work on as close as you want. The other reason for using this type of software is you only have to "Build" one model, the software will look after all your different frames for you! You can rest assured that your frames will have great detail (As long as you put it there) and will be "Visually Correct" in every way!

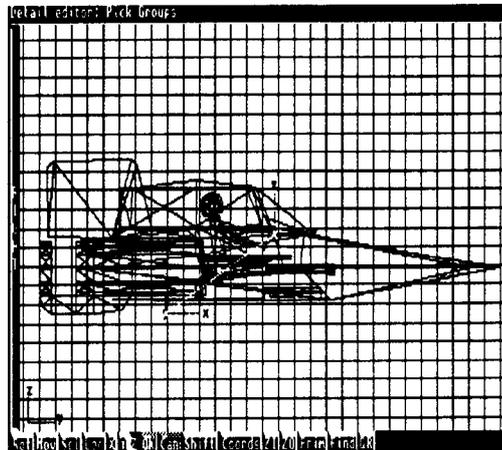
There is just one other piece of software that is needed to complete the process, that is software that can convert you 24-bit images back down to 16

colours. I use 16 colours so that I still have 16 other colours free for my other palette. Art Dept. Pro is one such program, but I'm sure there are PD programs around that will do just as good a job without the expense. Which brings me to the only drawback with this method, most of these powerful programs are quite expensive, but if you are serious about your graphics, then the expense is justified.

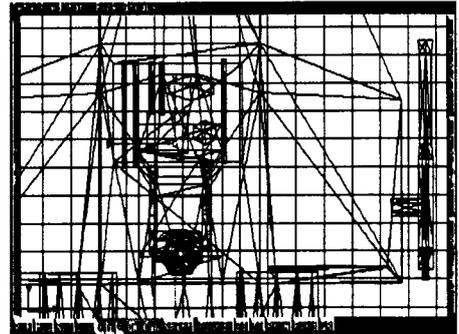
Below is a screen shot of the quad view used in Imagine II, the model displayed is an A Wing fighter out of Star Wars. I take no credit for this model, it is not one of mine. What you are looking at is...



TOP LEFT--TOP VIEW
BOTTOM LEFT--FRONT VIEW
BOTTOM RIGHT--RIGHT VIEW
TOP RIGHT--PERSPECTIVE VIEW
This shows you a magnified view of the

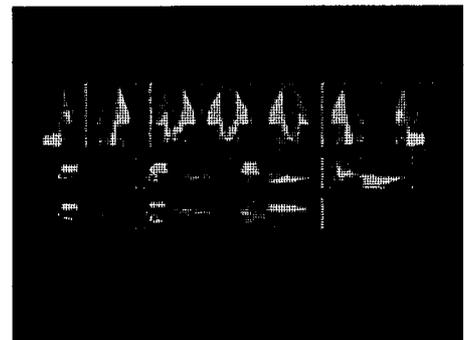


RIGHT VIEW, Imagine II allows you to fill the screen with just one of the views to zoom in and work in more detail. Here you can see that we even have a pilot at the controls, when this is rendered and animated, you can see the pilot quite clearly.



Another magnified view, this time from the TOP, here you can see the pilot easily and also one of the side cannons on the right.

OK, the picture below will show you what can be done, what I do is use 3 frames of animation to bank left or right, each rotated 15 degrees. Whilst this is OK for this size graphics, you may want to use 10 or even 5 degrees or rotation between each frame.



Once you have the final picture, convert it to 16 colours and load it into DPaint and do any final touching up that is needed. Remember, when you are converting down from a possible palette of 16.8 million colours to just 16, you will lose some of the really small detail. So just draw it back in! The only thing I have had to do is occasionally re-add a cannon or two.

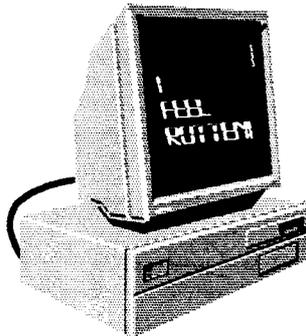
Now draw a box around your first frame which is the normal top view, then cut it off and stamp the box around the other ships. Make sure you pay attention to ensure that the frames align and there is no jerks to the left or right. Now resave and using the sprite editor, cut them off and save them out.

On this edition of the Newsletter Disk, I have included the frames for this A-wing fighter in both top view and side view, you are quite welcome to use them in your games.

Until next time, see you later,

....TEX!

CODE CLINIC



Greetings programmers! I hope you all had a great Christmas, and perhaps you got one or two presents for your Amiga (AMOS Pro perhaps?). Unfortunately Father Christmas didn't put any AMOS products under the tree for me.

Before I get this article underway, I would like to mention a great idea brought to my attention by one Chris Maguire. He has built a "digitising board" which allows you to easily trace images and transform them into coordinates for use within AMOS. He sent me some sample educational programs he has written that use this technique - one of them contained a fantastic map of Australia that really illustrated the power of this method. Hopefully you will find an article by Chris in this very issue, (and you will - Ed) and I suggest that you consider building a board of your own - there are many uses for one!

Anyway, I hope you all liked my four programs in the last issue. This time I have run out of ideas, and so I have only two programs to show you. These are:

1. A weird ripple effect that draws bobs on different bitplanes.
2. An evolutionary program using a recursive branching procedure.

In this article I am going to try to explain how these programs work in greater detail than I have done in previous issues. So, put your thinking caps on, boot up AMOS and do some finger exercises, here we go!

1. This routine opens a screen with two bitplanes, and draws a large bob on each bitplane. Weird things happen when they combine. Listing first, explanation later...

```
Screen Open 0,320,256,4,Lowres
Curs Off : Flash Off : Hide
Palette $4,$4,$4,$F
Colour Back $8 Pen 3
Centre "PLEASE WAIT"
CIRC[1] : CIRC[2] : CIs 0
Double Buffer : Bob Update Off
Autoback 0
Amreg(0)=X Hard(160)
Amreg(1)=Y Hard(128)
A$="Loop: Move XS(0,RA-X),YS
(0,RB-Y),50; Jump Loop"
```

```
Set Bob 1,,1 : Bob 1,160,128,1
Channel 1 To Bob 1
Amal 1,A$ : Amal On 1
B$="Loop: Move XS(0,RC-X),YS
(0,RD-Y),50; Jump Loop"
Set Bob 2,,2 : Bob 2,160,128,2
Channel 2 To Bob 2
Amal 2,B$ : Amal On 2
Repeat
Amreg(0)=X Hard(Rnd(100)+110)
Amreg(1)=Y Hard(Rnd(100)+78)
Amreg(2)=X Hard(X Bob(1))
Amreg(3)=Y Hard(Y Bob(1))
Bob Clear : Bob Draw
Screen Swap : Wait Vbl
Until(Mouse Key) or(Fire(1))
or(Inkey$<"")
Erase 1 : Screen Close 0 : Edit
Procedure CIRC[N]
CIs 0,0,10 To 320,256 : Ink N
For T=30 To 100 Step N*4+5
For S=1 To N*2 : Circle
160,128,T+S
Next S
Next T
Get Bob N,60,28 To 260,228
Hot Spot N,$11
End Proc
```

Now for an explanation of what bitplanes are. Imagine a grid of one's and zero's, the size of the screen. There is a one or a zero for every pixel on the screen. This is known as a bitplane (as it is a plane of bits). Now, a single bit can represent two different colours. For four colours we need two bits, for eight colours we need three bits and so on. Hence in a four colour screen we have two bitplanes (one on top of the other). Similarly an eight colour screen has three overlaid bitplanes.

So if we have a four colour screen, and we want to make a particular pixel have colour 0, then we must put a 0 in bitplane one and a 0 in bitplane zero at that pixel's position. Likewise, a value of 0 in bitplane one and 1 in bitplane zero gives colour 1, a 1 in bitplane one and a 0 in bitplane zero gives colour 2, and a 1 in both bitplanes gives colour 3.

In the ripple demo, we clear the whole screen to colour zero. Hence both the bitplanes will have a zero at every pixel position. Now, a special bob drawing mode enables us to draw our bob on

one of the bitplanes while leaving the others unchanged. Hence we draw one of the bobs on bitplane zero, and the other one on bitplane one.

Hence the bob drawn on bitplane zero will have colour one, as it will put a 1 in bitplane zero and leave bitplane one unchanged. Similarly, the bob drawn on bitplane one will have colour two, as it will put a 1 in bitplane one and leave bitplane zero unchanged. Now, if the two bobs overlap, there will be a 1 in each bitplane, giving colour three.

Thus if we choose the palette such that colours 0, 1 and 2 are the same (dark blue for example), and colour 3 is different (say bright blue), then we will only be able to see the two bobs when they overlap. This is exactly what happens in this demo. I realise that this may all be a bit confusing, so try changing line 2 of the listing to

```
Palette $4,$8,$8,$F : Colour Back $8 :
Pen 3 : Centre "PLEASE WAIT"
```

to see what I mean.

Manipulating bitplanes in this way provides eye-catching graphical effects that are used in various games and demo's. Unfortunately, AMOS doesn't provide many commands to manipulate bitplanes with. I would like to see commands such as Screen Offset and Scroll changed so that you can choose which bitplanes they will effect.

2. This next listing is a bit of a departure for this column, as it is fairly long and complicated. However, I didn't have any other short programs to put in, and time was running out... (how about sending me your own graphical hacks?).

This program is based on a similar program developed by Richard Dawkins in his book *The Blind Watchmaker*. I strongly encourage you to have a read of this, as it contains many interesting concepts. Anyway, the program is based on a simple recursive branching procedure.

In my last column you may remember one of the programs drew a simple fractal map using a recursive procedure. This program has a recursive procedure too (called BRANCH), although it is a little more complex. This procedure draws tree-like shapes on the screen.

The procedure accepts seven different "genes" as parameters. These genes control how long the branches of the trees are going to be, how large the angle between the branches is, how many branches there are going to be, and so on. You will notice that the procedure branch recursively calls itself twice. This is because the branches fork and form two more branches. This is also the reason why the shapes drawn are always symmetric.

Anyway, when you run the program, you will see six boxes on the screen,

each containing a "Biomorph". The box in the bottom-centre of the screen is the parent Biomorph. The boxes surrounding it are children of the parent. Each of the children has one of their seven genes randomly mutated. Click on any of the six boxes with the left mouse button to select that Biomorph to be the parent of the next generation.

You will soon see that you are able to breed races of all kinds of Biomorphs as you choose the shapes which are most pleasing to your eye. To get a close up view of any of the Biomorphs, click on its square with the right mouse button.

Anyway, that's enough explanation of how the program works, so here is the listing!

```
Screen Open 0,640,512,2,Hires+Laced
Curs Off : Change Mouse 2
Palette $0,$444
Degree : Reserve Zone 6
Dim G$(7),M$(5,7)
For Y=0 To 200 Step 200
For X=0 To 400 Step 200
If(X>200) or (Y>200)
Inc ZN
Set Zone ZN,X,Y To X+200,Y+200
End If
Next X
Next Y
Set Zone 6,200,200 To 400,400
G$(1)=1.0 : G$(3)=10.0 : G$(5)=3.0 :
G$(7)=1.0
Do
Cis : Hide

DISPLAY[200,200,400,400,G$(1),G$(2),
G$(3),G$(4),G$(5),G$(6),G$(7)]
N=0
For Y=0 To 200 Step 200
For X=0 To 400 Step 200
If(X>200) or (Y>200)
Inc N : MUTATE[N]

DISPLAY[X,Y,X+200,Y+200,M$(N,1),M$(N,2),
M$(N,3),M$(N,4),M$(N,5),
M$(N,6),M$(N,7)]
End If
Next X
Next Y
Next Y
Clip : Show
Repeat
Repeat
Z=Mouse Zone : M=Mouse Click
Until(Z=0) and (M=0)
If M=1
If Z=6
Else
For T=1 To 7
G$(T)=M$(Z,T)
Next T
End If
End If
If M=2
Screen Open 1,320,256,2,Lowres
Curs Off : Palette $0,$444 : Hide
If Z=6
```

```
DISPLAY[0,0,319,255,G$(1),G$(2),G$(3),
G$(4),G$(5),G$(6),G$(7)]
Else
```

```
DISPLAY[0,0,319,255,M$(Z,1),M$(Z,2),
M$(Z,3),M$(Z,4),M$(Z,5),
M$(Z,6),M$(Z,7)]
End If
```

OK, so I know it's really long and complicated, but it's really worth it - believe me! I've personally spent hours breeding all types of weird new life forms. With a little patience and experimentation you will be hooked! I am currently working on a full-blown Biomorph application, so if you're interested don't hesitate to contact me!

Feel free to change this program in any way. Probably the best bit to play with is the MUTATE procedure. Change the line:

```
M$(N,GN)=M$(N,GN)+(Rnd(1000)-
Rnd(1000))/500.0
```

to (continues under this column)

```
Show
Repeat
Until Mouse Click
Screen Close 1
End If
Until M=1
Loop
Procedure
BRANCH[X#,Y#,XL#,XDL#,YL#,YDL#,
A#,DA#,N#]
Shared AA#
N#-N#-1.0
DX#=XL#*Sin(A#*15) :
DY#=YL#*Cos(A#*15)
Draw X#,Y# To X#+DX#,Y#+DY#
Draw X#,Y# To X#-DX#,Y#+DY#
If(XL#>0.0) and (YL#>0.0) and (N#>0.0)
BRANCH[X#+DX#,Y#+DY#,XL#+XDL#,
XDL#,YL#+YDL#,YDL#,A#+AA#,
+DA#,DA#,N#]
BRANCH[X#-DX#,Y#+DY#,XL#+XDL#,
XDL#,YL#+YDL#,YDL#,
A#+AA#+DA#,DA#,N#]
End If
End Proc
Procedure
DISPLAY[X1,Y1,X2,Y2,XL#,XDL#,YL#,
YDL#,A#,DA#,N#]
Shared AA#
AA#-A#
Clip X1,Y1 To X2+1,Y2+1
Box X1,Y1 To X2,Y2
X#=(X2+X1)/2.0 : Y#=(Y2+Y1)/2.0
```

```
BRANCH[X#,Y#,XL#,XDL#,YL#,YDL#,
A#,DA#,N#]
End Proc
Procedure MUTATE[N]
Shared G$(7),M$(7)
For T=1 To 7
M$(N,T)=G$(T)
Next T
GN=Rnd(6)+1
M$(N,GN)=M$(N,GN)+(Rnd(1000)-
Rnd(1000))/500.0
M$(N,1)=Abs(M$(N,1))
M$(N,3)=Abs(M$(N,3))
M$(N,7)=Abs(M$(N,7))
End Proc
```

```
M$(N,GN)=M$(N,GN)+(Rnd(1000)-
Rnd(1000))/100.0
```

...to see the Biomorphs mutate more rapidly. M\$(N,GN) holds the amount that the mutating gene is to change by, so dividing this with a smaller number will make it change more.

Anyway, that's all for this edition. Please send me any little programs that you may have, as I am rapidly running out of ideas for this article. I would also like to hear any suggestions you have for games or applications, and any problems you are having with AMOS. You may contact me as follows:

Snail Mail: Jason Hutchens
3 Giddens Crt,
NorthLake
Perth, W.A.
6163

Phone:(09) 337 4448

E-Mail:hutchens@swanee.ee.uwa.oz.au

See you next time! Until then, happy coding!

Adventure Writing

Welcome AMOS users to the new series of articles on how to program adventures in AMOS. As you all know, TEX has moved onto pastures greener and so I will be writing the articles from now on. I will not be continuing on from TEX's articles but instead I will be starting from scratch. Why, you may ask? Well I happen to have a different style for coding although some things I code in a way nearly the same as TEX, but lots of things I code in a different way which I find more flexible. I hope my articles help all you coders out there.

Before I begin the actual article, I would like to introduce myself. I am Franta Fulin, and if you have seen my shoot em up, then don't worry, I do adventures much better than I do shoot em ups.

Obviously, the first thing you need is a story line. I have found that the best way to code an adventure is to start with a very basic story line and develop it as you code. A map also helps and its often good if you write down some ideas on paper and choose spots to put them in as you go, or scrap the ideas totally.

Another very important thing is atmosphere and depth of gameplay. Graphics, music, sound effects and vivid descriptions help a lot. Some of you may not agree, but I believe that

graphics are the least important. If you are not going to include graphics, then at least make sure the descriptions are vivid and perhaps some sound effects can add to the atmosphere.

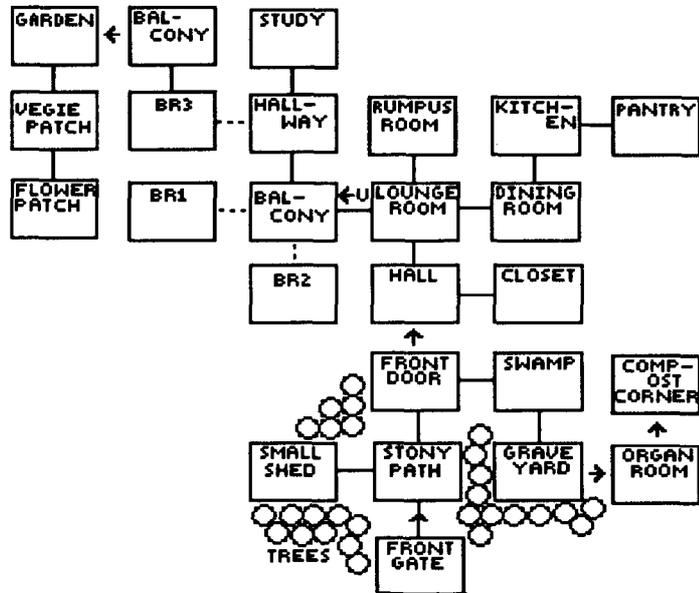
The plot through this series of articles will be that you are an undercover cop, sent on an assignment to find out what is going on in the strange castle on top of the hill. Hopefully this will turn out to be a very involving plot with lots of humour thrown in. The map of our adventure is below. You may notice that not all the locations on the map are in the code, we will put the rest in next time.

Now for some code:

```
Dim LOC$(30),MOV(30,6),OBJ$(
    (20,3),OBJ(20)VERB$(30)
Global LOC$( ),MOV(),OBJ$( ),
    POS,NO,NN,NL,OBJ( )
Global VERB,OBJECT,VERB$( )
POS=1 : NO=4 : NN=10 : NL=7 :
VERB=0 : OBJECT=0
Proc DAT
.
.
.
Procedure DAT
For LOP=1 To NL
Read LOC$(LOP)
For LOP2=1 To 6
Read MOV(LOP,LOP2)
Next LOP2
Next LOP
Data "The Front Gate",2,0,0,0,0,0
Data "Stony Path",4,0,0,3,0,0
Data "Garden Shed",0,2,0,0,0,0
Data "Front Door",0,5,2,0,0,0
Data "Swamp",0,0,6,4,0,0
Data "Grave Yard",5,0,0,0,0,0
Data "Organ Room",0,0,0,0,0,0
.
For LOP=1 To NO
Read OBJ(LOP)
Read
OBJ$(LOP,1),OBJ$(LOP,2),OBJ$(L
    OP,3)
Next LOP
Data 2,"Strange Stone","STONE",
    "A strange stone is here."
Data 4,"Pot Plant", "POT
    PLANT","A brown pot holds a
    small plant."
Data 3,"Rusty Pole","RUSTY
    POLE","You notice a pole
    here."
Data 5,"Swamp Mud","MUD",
    "There is thick brown mud
    here."
.
For LOP=1 To NN
```

```
Read VERB$(LOP)
Next LOP
Data "LOOK","READ","EAT",
    "SMELL","SIT"
Data "PICK UP","DROP","GO",
    "OPEN","CLOSE"
End Proc
```

This is the main code which sets up the locations, objects and nouns. With the objects you may wonder why I have dimensioned the array as OBJ\$(x,3). The three strings for each object are as follows: The first is the name of the object to be used when the player is told 'You pick up the a\$.' where a\$ would be the first string. The second is the way the program is to recognize it in the users



commands. Finally, the third is for when you look. If you have ever seen an adventure game that gives you a description and then says 'You can see:' and tells you all the objects at that location? Well if you have the third string is used to tell you what you can see without making it obvious. For example, Sierra adventure games blend the objects in with the description so that it is not plainly obvious what you should do. This system works by loading the description of disk (more on that later) and then printing all the third strings of the objects that are at that location. So take location number two, the stony path. If you typed look the computer would say:

The stony path comes to a fork here, leading to the north is the house and

to the west is a small shed. A strange stone is here.

This fits together better, I think. Obviously if you pick it up and drop it, the description would change to something like: 'The stone you dropped is here.' because some descriptions like the moonlight glints on an Erie object would work fine outside, but if you took it into the closet and dropped it, there would no longer be moonlight glinting on it.

If you are still a bit confused, write to me, my address is at the end of this article. You should see how it works when it is up and running.

Now that some basic data is set up, its time for a parser. A simple two word parser should suffice as we don't want to go too overboard, and I don't want to write a sixteen page article on how to do a multi-noun, adjective, adverb, pronoun understanding parser. I doubt I could pull one off anyway. A dual objective parser with pointers (under etc.) I could but there's no point.

After the call to the DAT procedure and in between the two ' comment marks, insert the below:

```
Do
Proc INP
Proc PARS
Loop
```

Now at the end of the code, you can

fold the DAT procedure if you want, add the below procedure.

```

Procedure INP
ERE:
Input a$
if a$=" "-"" then goto ERE
INP$=Upper$(A$)
End Proc

```

We are not too concerned about this procedure at the moment but we will fix it up later.

The most important procedure of our adventure has to be the parser. If the parser is unforgiving and unhelpful, often it will turn the player off the game. The parser that is below should be put near the end of the code.

```

Procedure PARS
OBJECT=0 : VERB=0
For LOP=1 To NN
If Instr(INP$,VERB$(LOP))=True
VERB=LOP
LOP=NN
End If
Next LOP
For LOP=1 To NO
If Instr(INP$,OBJ$(LOP,2))=True
OBJECT=LOP
LOP=NO
End If
Next LOP

```

```

If VERB=0 And OBJECT=0
? "I am not sure what you wish to
do."
Goto PAR_END
End If
If VERB=0 And OBJECT<>0
If OBJ(OBJECT)<>1 and
OBJ(OBJECT)<>POS
? "I am not sure what you wish to
do."

```

```

Else
? "What do you want to do to the
";obj$(LOP,1);"."
End If
Goto PAR_END
End If
If VERB<6 Then Goto S1
If VERB>5 Then Goto S2
S1:
On VERB Proc
LOOK,_READ,EAT,SMELL,SLEEP
S2:
On VERB-5 Proc
PICK_UP,DROP,GO,_OPEN,
_CLOSE

```

```

PAR_END:
End Proc

```

This is our basic parser and I'm sure you'll agree its fairly simple but effective. The first loop in the procedure checks for the VERB or the action in the sentence. The second loop checks for the OBJECT in the sentence. If there is no object and no verb found, an appropriate message is displayed. If there is a valid object in the sentence, but no noun then one of two messages is displayed. If the object is not being held and the object is not at the current location then the same message is displayed as if there was no object in the sentence. If the computer said that object is not here it would make it too easy as you would know it is somewhere, while otherwise you don't. It works well in practice, you'll have to trust me. If the object is present and there is no VERB in the sentence then the computer asks you what you wish to do with that object.

To make sure you don't get undefined procedure errors, make all the procedures at the end of the program, but don't put anything in them. So you have the following at the end of your code.

```

Procedure LOOK
End Proc
Procedure _READ
End Proc

```

Do this until you have your procedures at the end of your code. You can fold them once you have finished coding them. Not much room left in this article so I'll just put in one procedure, the GET procedure. This has to be one of the easiest procedures to do, so here we go.

```

Procedure GET
If OBJ(OBJECT)<>1 And
OBJ(OBJECT)<>POS

```

```

? "I'm not sure what you wish to
pick up."

```

```

Goto G99
End If
If OBJ(OBJECT)=1
? "You are already carrying the
";OBJ$(OBJECT,1);"."
Goto G99
End If
If OBJ(OBJECT)=POS
? "You pick up the
";OBJ$(OBJECT,1);"."
OBJ(OBJECT)=1
Goto G99
End If
G99:
End Proc

```

This procedure will work although you cannot move yet so the adventure isn't up and running yet. This first article was designed to be a structure building/explaining article so hopefully by the next article we will have the following:

- The ability to move around
- The ability to look
- Drop an item
- look at an item
- display an inventory
- our first puzzle or two
- anything else I throw in

I hope you all enjoyed my article, they will get better once we get it up and running a bit. If you want to write to me with complaints, suggestions, or money then you can contact me at:

**Wild Ware Amiga Adventure
Productions
27 Spaans Crescent
Salisbury North 5108
South Australia**

I cannot answer all letters, but I will try to answer them in the article. Good luck.

Franta Fullin

BBS Update

Telecommunications freaks will probably be familiar with our two main BBS's, Predators BBS and Blade BBS. Now we have a third BBS:

Amiga File Server (02) 876 8965

This BBS will shortly be linking a FIDONET up with Predators and Blade to create a large news/information base.

Club Forum

Due to the obvious success of the forum last issue, the section has been expanded and will continue to expand. thank you to all for your enthusiasm towards such a forum.

I WOULD LIKE TO SEE THE MAGAZINE AVAILABLE ON DISK AT TIME OF RELEASE.

M.McKENZIE,
ARCHERFIELD.

Unfortunately, that sort of thing eventually ends up of the BBS and national PD circuit. You, as a member pay to receive the Newsletter EXCLUSIVELY and therefore we should not have a situation where Newsletters are able to be copied.

I LIKE WRITING UTILITIES LIKE "DIRMASTER" AND I WOULD LIKE TO SEE BETTER DIRECTORY READING THAN "DIR FIRST\$", ETC. BECAUSE IT CANNOT READ DIRECTORIES AND FILES WITH SPACES. A BIGGER FORUM WOULD BE GOOD.

FRANTA,
SALISBURY.

Check out the new ones appearing in the AMOS Public Domain. DCAT have just finished the latest version of DISK ACTION with a further version to follow. As for the Forum; here it is!

AN ARTICLE ON HOW TO MAKE THE MOST OF THE OBJECT MODELLER WOULD BE GOOD.

ANDREW,
PRAHRAN EAST.

Anyone who is heavily into 3D programming, please get in touch with me (02) 748 4700.

THANK YOU FOR THE TIME AND EFFORT EVERYONE'S PUT INTO SUPPORTING AND ENCOURAGING AMOS USERS.

GARRY,
BROADFORD.

Thanks, Garry. The AMOS Club has been set up to do just that, and because of it, we have seen a lot of good local talent. Congratulations to everyone who has submitted articles and PD disks over the last two and a half years (yes it has been that long).

WHAT'S SO BAD ABOUT GOTO?

HAPPY TO SEE CODE CLINIC EXPANDING.

SCOTT,
MURRUMBATEMAN.

The GOTO statement is quite OK to use and if you feel it necessary to use it, then don't feel you can't. The point of that article was to show that you don't need it 99% of the time when you follow the "New Style" way of coding. This does not mean, of course, that you should change what style feels comfortable to you.

MORE EDUCATIONAL & PLATFORM GAMES AND LESS SHOOT 'EM UPS.

COLIN,
YOUNG.

MORE OF: THE NEWSLETTER! WHY NOT PUT OUT SPECIAL EDITIONS BETWEEN REGULARS? OR IS THAT IMPRACTICAL?

DAVID,
CANBERRA.

This is the very reason why there is a monthly column in the Australian Commodore and Amiga Review Magazine. It's to fill the gap between Newsletter releases. Keep you eye out in MegaDisc from now on, and also for a new Magazine that is planned to go on the market in a month or two.

COULD YOU DO MORE AMAL PROGRAMMING HELP IN THE AMOS NEWSLETTER BECAUSE I STILL CAN'T GET THE HANG OF IT. TYSON,
BELMONT.

THE NEWSLETTER IS GETTING BETTER, BUT NEEDS TO BE BIGGER. A SERIES ON HOW TO WRITE GAMES, SHOOT-EM-UPS, WARGAMES WOULD BE HANDY. CRAIG,
MURRAY BRIDGE.

We'll see what we can do. Bigger newsletters require more support, and that starts with you.

NEWSLETTER 12 WAS GREAT!

KEEP THE CODE CLINIC COMING. CAN ANYONE HELP ON HOW TO USE THE GRAPHICS.LIBRARY FROM AMOS?

PAUL,
RAYMOND TERRACE.

Over to you, Jason...

WHAT I WOULD LIKE TO SEE IS (1) A VIDEO ON HOW TO GET STARTED LIKE THE DPAINT VIDEOS, AND (2) WITH NEARLY 2000 MEMBERS, HOW ABOUT A WORK SHOP OR SEMINAR?

R.MOYLAN,
BLACKTOWN.

A video is rumoured for both AMOS and AMOS PRO. Also, a book is in the works and may be released later this year. All this is just rumour at the moment but keep a watch in the magazines. See pages 8,11 and 13. As for a workshop, it's a good idea. We'll see what can be done and there will be some details next issue.

CONGRATULATIONS ON A GREAT NEWSLETTER. MAYBE YOU COULD INCLUDE SOME SCREEN SHOTS OF PD GAMES ETC.

ROBERTO,
MT LAWLEY.

Screen shots in a Black and White Magazine may not be the best idea, however on each newsletter disk will be sample programs from the latest PD disks.

THE ONLY PROBLEM I HAVE WITH AMOS 3D IS THE OBJECT MODELLER WHICH WONT WORK WITH KICKSTART 2.04.

ADRIAN,
PERTH.

All fixed now. Version 11.3 is now ready and is compatible with Workbench 2.

I WOULD LIKE TO SEE EXAMPLES OF HOW TO PROGRAM VECTOR GRAPHICS WITH AMOS 3D.

SCOTT,
STRATHFIELD.

IN THE NEXT AMOS UPDATE, COULD YOU INCLUDE A COMMAND TO REBOOT THE COMPUTER? CHRIS,
TASMANIA.

This can be done with the EXEC procedure found in AMOS to call a small utility on the PD circuit called "REBOOT". On the other hand, check out an appropriate Hardware Reference Manual and make AMOS

call the Hardware register to trip the machine.

**WHAT HAPPENED TO THE ARCH?
ADRIAN,
OCEAN REEF.**

The Arch is now run as a new club by Andrew Suttor. See page 16.

**I WOULD LIKE TO SEE AN
ARTICLE ON WRITING MUSIC
USING G.M.C (GAMES MUSIC
CREATOR DISK BA-1).**

**MONTE,
CREMORNE.**

GMC is a dinosaur. New trackers exist like Protracker which is on disk AA-130 and is much more professional while being easier to use

**IS THERE AN UPGRADE OFFER TO
CHANGE FROM AMOS TO AMOS
PROFESSIONAL?**

**PHILIP,
CROYDON.**

Yes there is: \$106 + the front cover of your AMOS Manual. You will receive the 6 disk set and a brand new manual completely re-written.

**I WOULD VERY MUCH LIKE TO
JOIN THE ARCH CLUB. I AM
DESPERATELY TRYING TO WRITE
MY OWN ADVENTURE.**

**GREG,
ADELAIDE.**

See above.

Introducing The Digitboard

Are you like me? You want to get some nice drawings into your great AMOS program, or into DPAINT. But you can't draw for toffee, can you? And that clumsy mouse doesn't help. In DPAINT you've tried that little gadget stuck on the side of the mouse. The result looks like the remains of a melting candle. Proportional squaring has worked roughly, but not well enough to do that nice World Map you'd like.

What you need is a digitiser, eh? But \$600 for something decent!! Anyway - sour grapes - you don't need THAT good a result. Just a nice, honest drawing. One you can beautify yourself in DPAINT and put back into AMOS.

So clean up the rusty saw and screwdriver, and at the cost of about \$30 and a bit of sweat, get back to basics and build one. I've included the specifications for the one I built. Crude but flexible and very effective. Also, a (necessarily) short listing to demonstrate the method I use, culled from my childrens' geography program.

No doubt other readers already have similar but improved versions or methods. I'd like to hear about them. I hope there is somebody out there who can benefit from these few words.

A DIGITISING BOARD

MATERIALS -

- Tongue & Groove ordinary Floorboards - 7 lengths x 1 metre BALTIC PINE (hardboard or Cypress Pine are not soft enough to accept drawing pins)

- Two further lengths 580mm to screw under the boards as battens.

- 1 metre of thick (cartridge) paper, 630mm wide

Welcome back François

You'll be pleased to know that François' busiest period is now over and he will be with us in Fax again from now on. Here's a taste of what's to come...

Thanks very much for your Fax. Sorry to answer so late, but I have been partying a lot, eating and drinking! Time for me to wish you a very happy new year, and to all the 2000 members of the AMOS Club. Amazing, 2000 members. I cannot believe this! You are by far the world's biggest AMOS club.

I have been working really mad to get AMOSPro out before Christmas. That also explains why AMOSPro has a few bugs left. We have been lucky that no big bug was in the product.

All right, '93 on the way, some info and bug fixes for AMOSPro:

Version 1.1 of AMOSPro will work on an AGA machine. There will also be an update for AMOS 1.34 to work on

an AGA machine, but I will never implement the new resolutions for this product.

Everything is now ready for what I intend to do after the compiler, a new instruction "Intuition On/Off". With Intuition On, each time you open a screen, AMOS will open a ViewPort of the same bitmaps. Result, you will be able to create Workbench based applications!!!

You will not have, of course, all the nice hard- scrolling and all copper-based effects, but at least you will now be able to create serious applications with AMOS. Such a program will work on ANY machine, whatever the system changes.

Kind Regards,
François!

Southern Amiga Magazine

Watch out for the regular column in this new magazine.
Out in April!

- 1 metre of good quality tracing paper, 630mm wide
- 1 pad - A4 size graph-paper, 1/10 inch squares
- Misc - tracing paper to make drawings on : drawing pins
- Large spring-type paper clips (min. 6) : Paper paste
- 1 Pantograph - from any good supplier of drawing materials

TO MAKE THE BOARD

Cramp the 7 boards together, and screw the battens under, about 100mm in from each end. You should now have board 1 metre long and 630mm appr. high. Plane or sandpaper the top surface of the board to make it smooth. Stick the thick paper onto the board with strong adhesive tape. (Cartridge paper preferred)

TO MAKE THE DRAWING SURFACE

You need nine sheets of the squared graph paper, turned sideways so that, when joined three high and three across, they will cover the area of the board. Carefully trim the white margin off one side and one end of each sheet. (You can judge which edges. Use a metal edge and art knife for accuracy).

The remaining margins can now be glued with paper paste, and the sheets joined together so that the 1/10 inch squares line up precisely.

This is the only part which requires care. When they have been satisfactorily joined, go along each joint with a wooden rule and firmly press down. (If you are fortunate enough to have 1 metre of proper 1/10 inch squared drafting paper 630mm wide, then, of course, all that gluing can be avoided). Whichever way, you should now have one metre of squared graph paper which should now be pinned down onto the board.

MARKING OUT

Pin down the sheet of good-quality tracing paper onto the board. (Eventually the two top corners should be strongly secured, but the two bottom corners of this sheet will need to be lifted so your drawings can be inserted, thus they will be

secured with spring paper clips. However, to proceed:-

You can now see the graph paper squares below, and you need to set out your 'screen' on the top sheet of tracing paper - Four inches in from the left edge of the board, draw a vertical line, full length. One inch down from the top, draw a horizontal line.

Downwards, mark the line at each inch and, starting from the top corner as 0,0 number each mark 10,20 . . to 200. In the same way mark the top line to the right, up to 320. Complete the rectangle along the bottom and the right side and there is your 320 x 200 screen. Each 1/10 inch square represents one pixel; however, by calling each 1/10 inch square 2 pixels, the board could represent a 640 x 200 screen.

DRAWING

This depends on your imagination. To start, take a separate piece of tracing paper, say, as big as the top left quarter of the board. Now, unless you are an artist, trace some pictures from a magazine or whatever sample - a few simple line drawings, but sufficient to cover most of the your piece of tracing paper.

When complete, place the sheet of drawings under your top cover sheet (the tracing paper 'screen'), but above the graph-paper sheet. Place it in any position, keeping in mind that this is the screen. Clip down the top sheet to 'sandwich' the drawings. Pick a strategic starting point on each separate drawing and mark it lightly on the top sheet as a reminder. (Except for these temporary marks and your screen outline, the top sheet of tracing paper should never be drawn on).

Read off the X,Y coordinates of the starting point, then choose another appropriate point along the drawing and do the same. Jot these down on a sheet of paper as the start of your DATA list - they will be x1,y1 and x2,y2 in your first DRAW command in your program. The next command in the program will be DRAW TO x,y, so these will and so on until you have traversed the lines of your drawing. (On your written list, keep in mind to put a comment to pin-point the commencement of each new line, in

case you need to retrace your data for errors). So now your drawing has been 'traced' as a DATA list of coordinates.

Your first DRAW command will be kept separate in your program, but let's say that after that first set of coordinates (x1,y1 to x2,y2) you have 30 more DRAW TO pairs (x,y). These will be incorporated in a FOR r=1 to 30. . NEXT loop in which you will READ and DRAW TO each pair, x,y.

NOTES

If your chosen screen is, say, 640 x 200, simply keep in mind to double your x coordinate value when writing down your x,y data pairs. If you have drawn, say, a map and it looks 'long and skinny' on the screen, top to bottom, then in your program DRAW command adjust the Y coordinate by multiplying the variable (for example Y*0.9 or Y*1.2).

For those who are not artists, a simple Pantograph will achieve excellent results, especially for detail such as on maps. After being programmed onto the AMOS screen, they can be transported to Deluxe Paint or similar for processing, then transferred back for use in AMOS. It's all much easier to do than it is to read about. Once the plotting and listing of the coordinates has become familiar, the whole process becomes quite easy, and worth the effort for the result.

Oh, and don't remove your drawing from the board until you have checked the result on screen.

GREYSCALE, PRINTERS, and DPAINT

DOT MATRIX v DPAINT

My printer is a 9-pin dot-matrix Citizen 200GX. Not a bad printer at all. But it IS frustrating having a picture on your DPaint screen which might look nice framed. Let's face it - at this level, colour is OUT. Even outlines and hand- painting can look better than the result from a low-level colour kit. And I've yet to find an article on a reasonably-priced camera that would do the job. (How DO they produce those great colour screen shots in the magazines?)

So the alternative is black and white.

Resigned to this fact, black and white has its own special attraction. (To illustrate this, peruse a copy of 'Zen and the Art of the Macintosh' - excuse the expletive - by Michael Green.

Mine was from the local Library.) By first setting up a stable of line/dot brushes, nice effects are possible.

I tried gray-scale prints from DPaint, but they were not good. On examining these prints, though, I saw some parts were just dots where others were mixtures of lines and dots. Now, newspaper pictures of some 50 years ago (I believe!) showed the ranges of dots very clearly. That was the technology of the time.

Could I achieve this effect? Well, not quite. But I did get a result which is presentable. Here's how:-

On a 16-colour gray-scale screen, including B & W of course, make 16 rectangles in a 'ladder', fill each with a shade and print the value alongside it. Save the picture. Quit DPaint and open its Preference screen. Set the printer to EpsonX[CBM_MPS1250] because, for me anyway, EpsonXOld doesn't give a good result. On the Graphic 1 screen set Gray Scale. Use this (or Save it).

Back in DPaint3, load the Gray-scale picture and Print it, (in the Requester set [Gray] and [%]). On the print-out, examine which of the blocks are printed only in DOTS. These are the values to use in your pictures. For me, they were FFF,EEE,DDD, BBB,999,777,555 and 000. Conveniently, an eight-colour scale. This helps to save precious Chip ram, especially in interlace mode, and is sufficient for good prints.

The judicious use of black, and outline greys adds to the effect. As well, give some thought to a new printer ribbon.

Bear in mind all the above settings suited my Citizen. You may require different ones. Also, it will be quite obvious to some readers that a better knowledge of my printer's workings could have produced the same, or better, results by different means. Please tell me how.

Chris Maguire

DigiBoard Program Example

```
Rem ** MAP OF AUSTRALIA **
Screen Open 1,640,240,8,Hires
Screen Display 1,,,,
Hide
Curs Off
Rem
Ink 2
Draw 150,28 To 142,27
For R=1 To 160
  Read X,Y
  Draw To X,Int(Y*0.9)
Next R
AUST:
Data 138,31,136,31,136,26,127,24,114,
25,106,28,108,30,104,31,106,33,103,32
Data 99,35,100,38,92,39,95,43,92,45,
88,41,80,46,81,51,76,54,73,58
Data 56,63,50,65,44,67,32,69,10,76,10,
81,6,77,4,84,6,85,5,89
Data 7,95,18,103,14,101,11,102,16,
107,22,108,34,123,36,127,44,135,42,
139
Data 38,141,39,145,49,148,60,148,67,
145,72,144,77,141,94,137,98,138,102,
137
```

```
Data 108,137,118,132,132,126,137,
127,149,125,158,122,176,121,186,124,
190,123,198,124
Data 204,130,210,131,213,139,220,
138,224,134,229,132,230,130,235,126,
236,131,230,136
Data 230,138,226,140,226,142,238,
141,240,146,245,148,239,143,242,144,
248,145,255,148
Data 255,151,253,152,262,157,265,
158,269,160,275,158,283,161,290,157,
293,160,297,160
Data 298,163,313,162,320,159,338,
159,342,156,342,151,348,145,353,143,
353,140,360,136
Data 360,134,366,133,373,127,376,
120,380,116,386,110,386,108,382,105,
380,100,382,93
Data 374,85,365,81,366,76,358,73,355,
74,353,71,353,68,343,60,324,52,320,44
Data 316,39,313,26,310,25,308,27,300,
24,300,18,297,15,294,7,287,5,287,12
Data 284,15,284,17,280,18,280,21,278,
23,278,34,272,42,263,44,242,47,213,27
Data 220,22,220,19,222,18,230,12,225,
11,222,13,220,12,204,11,178,5,172,8
Data 180,8,180,12,168,12,162,15,155,
17,155,21,150,22,149,25,154,26,150,28
```

1.34 BITS AND INFO

Miscellaneous AMOS 1.34 info

- In Direct Mode, using [Shift]+[Up] or [Shift]+[Down] will alter the size of the Direct Mode screen. (This also works in AMOS Professional V1.00)

- In the Editor, typing a "tilde" character (~), on its own line and then pressing return, will allow the insertion of blank lines in your program listing.

- The Colour Back command will only appear to work when the copper is initialized, so this command will only seem to work once for every screen opened (unless other screen functions are performed). To get around this, so that true full screen fades are possible, use the Screen To Front n command after the Colour Back command (where n is the current screen number).

NB. Only one screen has to be opened for this to work, so no excess memory is wasted.

```
Colour Back Colour(1)
Screen To Front 0
Wait Key
Fade 2
For I=0 To 30
  Colour Back Colour(1)
  Screen To Front 0
  Wait Vbl
Next
```

Undocumented AMOS 3D Commands

Two commands which are not to be found within the documentation of this Extension, and are reasonably important (particularly the latter), are:

● Td Priority n,b,c

Td Set Colour is used to change the colours found on the faces of a 3D Object, and works in a similar way to the functions for assigning colours for Objects within the Object Modeller itself.

The letters following the command represent the following:

n is the Object number

b is the Block number

c is the Colour combination code of the block.

Valid Colour Combinations range from 0-15, where numbers 0 to 12 are the same as within the OM, and numbers 13 to 15 are new.

Td Priority is used to alter the priority at which a specific Object is shown in relation to other Objects. In a way, it is similar to the Bob Priority command, but allows far more control over the objects.

Priority values range from -100 (making that Object always be behind all other Objects), to 100 (making that Object always be shown in front of all other objects). A value of 0 is the default value for all Objects, and arranges the Objects by depth.

The letters following the command represent the following:

n is the Object number

p is the Object priority

Values of p:

p<0

Draw the object behind all other objects with higher priorities.

p=0

Draw the Object in the normal way (by depth).

p>0

Draw the object in front of all other objects with a lower priority.

This command is of great importance to people who wish to write "flight sims" etc, as they will now be able to create a small flat are, which can be moved around with Object 0 as a ground, but allow other objects to appear above it, no matter how far the aircraft Object is away from the viewport (by having the ground with a priority of -100)!!

More Problems Answered

● On page 7 of the Volume 12 (Spring) newsletter, in the "More Problems" section, Bill Jordan had difficulties with the Limit Mouse command on PAL screens. A solution to this was suggested, but a better and more efficient way is:

Wait Vbl : Limit Mouse

● Another problem involved the use

of Sam Raw on a 1 Meg machine. Hopefully!! this should help. To use Sam Raw the bank that holds the sounds to be played should be located in Chip Ram, defined by using a Reserve as Chip Data command, and then using the standard Sam Raw command.

Inactivity in Menus

For ages I have been wondering how to use the "inactive" string in AMOS Menus. According to the manual, the following structure is used:

```
Menu$(1)="Project"
Menu$(1,1)="Normal","Select","Inactive"
Menu On
Menu Inactive(1,1)
Direct
```

If this is used, according to the manual, when the "Project" menu is selected, the item shown should read "Inactive". However, this is not the case, as "Normal" is still displayed, yet is not selectable! (NB in the program above the "background" string has been left off the end of the Menu\$(1,1) definition string, as this is legal according to the manual.)

To allow your inactive string to be displayed, the "background" string must be included and contain SOMETHING. If nothing is desired as a background, or if something is, make sure that the last command within this string is:

(LO 0,0)

which locates the graphic cursor back to 0,0. The Menu Inactive command never seems to allow the default italics to occur (which is suggested by the manual; both AMOS 1.34 and AMOS Pro V1.00), so if this is the desired effect, in the "inactive" string, have the following:

```
"(SS 4)" + Normal string +
"(SS 0)"
```

Therefore, to make the initial program above work correctly, the following should be entered:

```
Menu$(1)="Project"
Menu$(1,1)="Normal","Select","(SS4)
Inactive(SS0)",
"(LO 0,0)"
Menu On
Menu Inactive(1,1)
Direct
(NB. If this routine is entered into AMOS Pro,

```

place a Wait Vbl command before the Direct, as there doesn't seem to be enough time for it to work properly; the Editor is returned to instead.)



AMOS TOME Series IV (V4.23) Review

AMOS TOME IV has finally arrived, and is certainly well worth a look. The Extension can be obtained from Allen Computer Supplies in Croydon, Vic (see advert in ACAR for address etc), and is available as either an upgrade or new package (same product, different value. By returning the original cover of the TOME V3 pack, the upgrade only costs \$20, when compared to a more substantial £29.99).

This time round, the package includes an 80 page manual explaining all of the commands and programs, instead of having a *hypertext* style guide on disk, and is better because of it.

On disk are 19 example programs, demonstrating various aspects and commands of TOME, as well as 5 games which exploit the TOME command set.

If you have not used or heard of TOME before, it stands for "Total Map Editor", and allows easy creation and manipulation of maps within the AMOS environment. These maps are built up using standard AMOS icons, and can be scrolled in any direction easily.

TOME IV now offers about 67 commands which is a vast improvement over V3.0 and V3.1, and now allows things such as tile animation (have animated backgrounds) and zone detection (to see if the player have entered a specific region of the map).

However, not only have you got 67 new commands for use within your programs, but also included within the package, is the "Shuffle" or "AMOS Club Extension V2.6". This gives about another 39 miscellaneous commands which allow such actions as the reading of analogue joysticks, renaming of AMOS Banks, changing of Bob colours, displaying of all bobs in a bank on screen, word wrapping and many more.

The TOME Editor is greatly

MegaDisc
Disk-Magazine & PD
... for the best in Amiga info and entertainment
 Megadisc has been around for a number of years, and like ACAR (see page 13), there will be a new AMOS column featured in there every issue. Call Tim for details on (02) 959 3525

enhanced, yet still has the same general feel to it, so after upgrading, you should still feel at home. Some new items within the Editor are:

The Tile Animation Controller and

MaPLe

MaPLe is a Map Programming Language, and allows you to automate some tedious functions with 19 macro-styled commands.

Overall, TOME Series IV is a great improvement over V3.1, and if you used the previous version a great deal, you would be crazy to miss this upgrade for only \$20.

If you do not own the previous version of TOME, but are interested in writing games which could use this map technique (such as Shoot 'em Ups, Adventures, Arcade games etc), then TOME is certainly worth the money, when compared to the time and memory which would otherwise be wasted using any other method.

When you also consider that you are getting two Extensions for the price of one, and are therefore expanding AMOS by about another 100 commands, you are certainly receiving value for money!

Bank Namer Proc

One of the commands available in the "Shuffle Extension", is Bank Name, and allows the naming of your own reserved AMOS Banks to whatever the you desire. This can still be done without this extension, with the following procedures:

(see BankNamer.Amos on Newsletter disk 4)

The first of the procedures is for use within AMOS Pro, as it utilizes the new command Poke\$, which allows you to poke a string into memory at any address.

The second procedure is for use within any AMOS (including AMOSPro but excluding Easy AMOS).

To test out the procedues, type the following code into the Editor, and replace BANKNAME[] with BANKNAMEPRO[] if using AMOS Pro.

```
Reserve As Data 10,100
BANKNAME[10,"Baldric"]
List Bank
Save "ram:stuff.abk",10
Wait Key
Erase 10
Load "ram:stuff.abk"
List Bank
```

End

(Again, in BankNamer.Amos on Newsletter disk 4)

This program demonstrates that the bank name is saved and loaded with the reserved bank.

(NB. The names of banks containing Sprites, Icons, Music etc can be changed, but will not be saved when using AMOSPro!)

Kickstart Tester

For those of you who would like their program to know what Kickstart it is being run under, the following procedure may come in useful:

(See KickTester2.Amos on Newsletter Disk 4)

The procedure can be used and analysed as follows:

Proc KICKVERSION

```
KICKSTART#-Param#
Print KICKSTART#
```

The KickVesion procedure above, will only scan for kickstart 1.3 or 2.04, but other versions can be included by increasing the first value in the Data statement to the number of kickstarts to be analyzed, and adding the version number to the list. The version number is determined by typing Version in CLI, and reading the number to the left of the decimal point for the "Kickstart" value; ie for Kick2, Version would output the following:

```
Kickstart version 37.175.
Workbench version 37.67
```

thereby giving a result of "37". Once this is done, insert an If Check=n clause to allow the output from the procedure to return the kickstart value.

NB. The kickversions in the Data statement MUST be in desending order, otherwise the procedure will send up false readings.

(eg. If Kickstart 2.1 was to be included, the Data statement would read: Data 3,38,37,13 and the If Check=n values would be increase by one.)

60Hz in AMOS

For those of you out there wanting to utilize the extra speed given by proper NTSC running at 60Hz, the following procedure will allow you to do this:

```
Proc SIXTYHZ[True]
Procedure SIXTYHZ[HZ]
```

```
If HZ=True Then HZ=0 Else HZ=32
Poke $DFF1DC,HZ
End Proc
```

This program works by simply poking in the necessary value into a display register, to flip between 60 and 50Hz. (NB. This will only work with Amigas installed with the ECS Agnus chip.)

GadTool Style Buttons in AMOS

In Volume 12 of the AMOS Newsletter, Bill Jordan gave a listing for a button to be used in AMOS. Below is a completely different routine which I had already developed before seeing his, and is more like the buttons found within GadTools style applications in Workbench 2. Not only does it select the gadget if the mouse is pressed AND released within the button, and invert the colours, but it also allows the option to include a Hotkey for gadget.

The button definer and button selection detector are given as two separate procedures.

(See Newsletter disk 4 - Button.Amos)

The button definer procedure has the following template and options:

```
BUTTON[BX,BY,BW,BH,ZN,TC,BC,
HILITE,SHADOW,TXT$,UNDER]
```

BX and BY - Top left corner of button.

W - Width of button.

H - Height of button.

ZN - Zone number to be set for button.

TC - Text colour.

BC - Background colour of gadget.

HILITE - Bas-relief hilite colour on button.

SHADOW - Bas-relief shadow colour on button.

TXT\$ - Text to be found within gadget.

UNDER - Flag for underscore hotkey.

Basically, the program draws the button and defines a zone for it, which the mouse must enter and click, to select the button. (For further details, see description in Newsletter 12).

However, a major enhancement, is

the use of a hotkey, which is defined by having the "Under" flag set to true. The program then looks at the command line, and button name, and searches for an underscore character "_". If it is found, the key following it, will be used for the hotkey to activate the button.

That is what all the Data statements are used for, and consequently the array SCN\$(); they determine the Key State() value to be used in the BUTCHECK procedure.

The result from the BUTCHECK[] procedure is OK, and this tells you the number of the button selected, so numerous buttons can be used. The reason arrays have been used with the co-ordinates and hotkeys of the buttons, is to allow various buttons to be on- screen at once (10 to be precise).

For any queries on AMOS, I can be reached on Guru Meditation BBS in Melbourne, on the number (03) 326 0440, or can be net- mailed (E-mailed). The addresses are:

Apana: U.U.C.P
DAVIS_COLE@GURU.APANA.ORG.
AU

Fido: 3:635/542.0

David Cole

There should definately be something there for everyone. Thanks, David. Contact David on the FIDO above.

Ed.

local hangout

Roberto Parlavecchio has got some projects coming along slowly and would like some help from other programmers that might be interested in co-writing some games, etc. Write to him at:

Roberto Parlavecchio
10 Clarence Street
Mt Lawley
WA 6050

Monte Boyd wants to get in touch with other programmers to team up and write games from Shoot-em-ups to Adventures. Monte promises to answer ALL letters. His address is:

Monte Boyd
52 Sutherland Street
Cremorne
NSW 2090

Apologies to Chris Good. I got his street name a little incorrect. His address is:

Chris Good
2 Halcyon Drive
Waterview Heights
Via Grafton

The Australian COMMODORE & AMIGA *Review*

If you're sick of only receiving AMOS information every three months, and you have never heard of this magazine, The Australian Commodore And Amiga Review features the "AMOS Column" every month. It costs \$3.50 per issue and is available from any local newsagent. You'll find all the latest information on AMOS Basic from all around the world every month.

Programmer's Autumn Challenge

As requested by the majority, A new 3 monthly competition will be held. It was suggested that a regular comp rather than an annual one would give programmers something to aim for. The third category is a little different than usual. Winners and prizes next issue!

OK! Categories the Autumn season are:

- A game, either Shoot'em up, adventure, or simulator (ANY type, not just plane, etc. How about a heart vessel simulator???)
- A Demo. Any type of demo. Winner will be judged on speed and smoothness. Try to make it fast moving and attention grabbing!
- Or a Strange section. Try to make the most professional-looking, but most useless program, eg. A fully icon driven BOB wave maker. Anything weird.

AMOS

Angy's Angles on Amos

Oh noble editor, please get the headline right, please, PLEASE get it right, GERRITRIGHT! *Oops Sorry...Ed.* Thank you. *No Problem...Ed.*

Hi, and Happy New Year! First, a little tale. I was in my favourite computer store asking the staff some curly questions about graphics hardware, and stood aside to allow a middle-aged gent. to pay for his purchases - among which was AMOS Professional. I couldn't resist chipping in to tell him what a great programming language AMOS is, and putting in a plug for the AMOS Newsletter, and the Hotline. He seemed never to have heard of the Newsletter etc. but said he knew AMOS was a great programme because he owned the first version. Now, either he didn't read the documents in the box, and so didn't join the club and didn't register as an owner - in which case he's missed out on the excellent upgrade offer and paid far too much for his copy of AMOS Professional - OR his copy of AMOS is one of those "permanently borrowed from a friend" copies that so annoy the anti-Piracy people. I'm aware of many people whose first copy of a productivity programme is - um - shall we say "grey"- but who have rushed out to legitimately buy the next upgrade (at full price) as soon as it's been released. For this reason I don't agree with some of the manufacturers that every pirate copy represents a lost sale. The problem for the customer is that while ten minutes in the store may be enough to test out a game - in those stores that will let you do this - but it's not long enough to test out a complex

package like AMOS, particularly with three small boys clutching "Zool" in the queue behind you. Some stores, of course, still take the attitude that the customer who wishes to buy a microwave oven is entitled to the undivided attention of a salesman who knows the various models well enough to explain the differences, but the customer wanting to spend the same amount of money on a word processor is expected to buy one off the shelf in a sealed pack. I'm all for easily available demos of productivity programmes, so I'm quite pleased that the original AMOS has been released as the Amiga Format Christmas cover disc - I'm sure it will result in lots of new converts busily buying their upgrades. (To comfort those of us who bought ours in the shops, the magazine buyers still have to pay for their manuals.) And I hope we can now welcome to the fold one respectable gentleman with his brand new paid-for copy of AMOS Professional.

On the subject of paying, I hope people are taking note of requests for shareware fees. I'm beginning to read annoyed diatribes from shareware authors who are threatening to GO COMMERCIAL if fees aren't paid. (I'm not sure how they actually know how much their programmes are being used regularly by people.) Trouble is, I thought one of the ideas behind PD and Shareware was to allow programmers to hone their skills until they reached a professional level. As far as I'm concerned, if you're good enough, by all means go commercial and the best of luck.

Public Domain Review

That said, there's been a general improvement in quality with this batch of AMOS discs, more properly finished work and more care to ensure all necessary files are included. Some presentations are really slick and professional. The British Library has a new librarian in Anne Tucker (now Sandra Sharkey is running Deja Vu software), and an attractive new title screen by Dicon Peeke - check out that mouse! Maybe some one could design a new front-end for the Australian discs? (Should we make it a competition, Wayne?) On the downside I've found one or two demos which run out of memory halfway through even with 3 meg. I presume these have been written on machines with 1 meg. of Chip Ram, but it would be nice if they told us this. AMOS doesn't have a command allowing you to free up memory as you go. And a small note to programmers: do you think you could avoid lines which tell the computer to look in a particular drive? It's a nuisance to have a programme which only runs in df1 or df0, and some people out there may still only have one drive.

I've continued working with AmosBase (AA120), which, simple though it is, I still find clear and very user-friendly - the keystrokes are easy to learn and quick to use. (At least, there is "B for Amend", but just think "boo-boo" and you'll be right.) I had another look at PFDB3.0 (BA132) as promised, but I now can't find the access for the printer support, and I don't find it as easy to use as AmosBase, so the Aussies win that one.

The two outstanding discs in the new batch, at least for a puzzle game addict such as me, are without a doubt "Tetren" and "Quadrix", superb, fully finished, addictive games which have kept me away from housework for weeks. Both have a new GA prefix - not for GAME, but for General

Amos, and are programmes with NO source code on the disc. I'll talk about them first.

GA131 Quadrix by Geoffrey Sparks.

Autoboos. Happy without extra memory.

A very professional product, with elegant presentation, a catchy tune which doesn't become maddening, and satisfying thunky sound effects. You are set a series of problems to solve against time, by firing blocks with different properties at each other from four sides of the playfield, and you need quick wits and a fast trigger finger. Read the instructions via Workbench first, they help a little. Some levels are maddening as you struggle to work out what on earth you are meant to do - and then light dawns and of course, it's all so simple... Sacrifice a life or two to draw some of the levels and work them out with pencil and paper, it's worth it. Every ten levels there is a password (I would have liked them every five), and a falling-blocks subgame - make a line of three in any direction to score. This eventually reaches speeds too fast for the eye to follow.

When asked for a password at the beginning, enter anything you like to start - you don't have a password yet. Entering letters is laboriously, with the joystick, and I do wish that when you wish to restart after playing a higher level you were returned to the last password entered, instead of having to enter it all over again.

One major snag with the gameplay is that far too often it interprets one click of the fire button as two clicks, or one move of the joystick as two. I tried three different joysticks, and eventually anchored one firmly to a board and tapped, rather than pushed. Wayne tells me this is because Geoffrey has told the computer to check for "Firebutton on" but not to check afterwards for "off", and I presume something similar for the joystick position. This is where not being able to get at the source code is a real nuisance.

That said, I love the game and I just have to beat level 23, it shouldn't take more than an hour or three.

Geoffrey lives in Castlemaine, Victoria, and for a Shareware fee of \$20.00 will send you 100 more levels and all the source code.

GA119 Tetren by Andy Whiteley

Autoboos. Needs extra memory. One or two player. Prefers to be played un-write-protected. Can be installed to hard drive.

Andy's a Brit. who wrote this for his Mum who was hooked on Tetris. (I approve of people who write games for their Mums - are you listening, Adam?) There's a pleasant opening sequence with animated drawings of the faces of Andy, his Mum and the friends who helped with the programme.

This is NOT for someone who has never played Tetris, for although there are three easy levels, they are over far too soon, and you are playing against all kinds of obstacles. Ever played Tetris in the dark, or backwards? Now's your chancel Read the level titles and their passwords for a glimpse of Andy's sense of humour, and listen to Mike Kingham's tunes for more.

The double playfield is clearly presented with a fairytale background complete with a gnome who dances for you as you play. Watch what he does if you dawdle too much and bore him! In the two player version you each have your own personal gnome. In the one player version the second playfield is taken up with a barchart showing what proportion of blocks have fallen of each colour during the game. Interesting if you have time to look! Note: write down the passwords fast, they don't stay on screen for long, and if you don't you'll have to replay and win that level all over again to reach the next one.

This game is very memory intensive - it keeps switching off the music to conserve memory. It is also a very full disc and prone to lock up on the least excuse. I strongly recommend you make a backup (I didn't, and now I may never beat level 18) and that you DON'T save your high scores.

Clever, fun and I like it.

Now for this month's British discs.

BA404 Alpha Demo by Vex

Autoboos. Needs extra memory. Short, and quite fun. Written in three days as his first AMOS programme - so you see, it can be done. It has some good pictures, scrolly greets and some cute animation plus an appeal for a music writer to join him (he's Welsh). His post code is very difficult to interpret due to the fancy typeface. OK, but not a red-hot must-buy.

BA405 Adventure Bank Creator v1.2 by Chris Beeson.

Autoboos. PAL only, prefers extra memory though Chris says it should run on 1/2 meg. Certainly runs out of memory if you try to run it from AMOS. Designed to allow you to create a databank to include in your own programme with all the information for a click-on graphic style adventure. You design the background for your dungeon or whatever, plus objects like swords and potions for your players to collect. Load in the picture, load in the bobs, and click on the picture to put them where you want. Define zones and exits in the graphics and attach scripts to tell the computer what to say when the player enters "Get Sword" or "Examine Cupboard" etc.

It's not quite as user friendly as it seems to be at first glance. For example, to design graphics within the programme you absolutely need DPaint III (not II or IV). Also, however much you click on "load" you won't be able to load in your graphic - this command is for loading Adventure Bank Creator files only. To load in your drawing click on "background". It's also rather crash-prone. With patience and perseverance, though, it could prove very worthwhile.

BA413 Assorted programmes

Load through AMOS. NIALL, Zone Editor, Icon Utility and Intro AMOS need extra memory. the disc is called "AMOS fractal ham" and features a group of colour fractal generated

pictures as well as lots of bits. To take the pictures first:

Cauchyham: Set two parameters and watch mesmerised as line by line and very, very slowly a complex pattern in full colour emerges - it can be printed out. There's a certain soothing fascination, as in watching a fish tank, but in the end the fish have brighter personalities. Set lower numbers for a smaller picture drawn faster.

There are also a whole group of pictures with names like "Normtolorentzhamc2" and "yakawareggeham": These are more Ham fractals, the pictures slowly forming over a watercolour wash effect.

In the same vein is **Crazy IFF viewer:** shows piccies all swirled up. Tell your friends it's been done by the latest most expensive image manipulator!

Also on the disc:

Eventlogger: a utility for timing such things as video tapes, adding comments and printing them out. The high-tech answer to the stopwatch, pencil and pad.

3D Requester by Mike Foord - a new requester design with "raised" buttons - why bother?

AMOSdir: should give you the disc's directory. Unfortunately it needs "vol.rdd". Deleting the reference in the code lets it read your disc, and then the "dir" command in direct mode will list them, with luck. Useful for the kind of data disc the "dir" command pretends it can't read - DPIII data, for example. But I found it twitchy.

Convert AMOS by Michael Foord: turns hex, decimal or binary strings to decimal numbers and returns them as a param#. I think. Does that make sense to you? It doesn't to me. Also in the source code is a Crunch/Decrunch routine and Crypt routine.

Icon info: running this does zilch. I think it's a bit of a routine and it's uncommented.

Discol: you must put in the references to your own music and

picture banks, then I think you get a slide show to music. It would have been nice if this had been commented.

Disk Utility by Kenneth Kowolewski: Delete the reference to an accessory disc in line 30 (proc: copyright). The utility tells you how much space is left on the disc, deletes files, creates directories and so on. O.K. if you need a directory utility and don't want to quit AMOS to run it.

Edit Object by Shock Wave (Nick) of Hanissis 5 (Truly): Make a drawing and it will tell you the co-ordinates for each point for you to include in your data - very neat and saves a lot of guesswork if you're like me and know that the top left hand corner of the screen is x0 y0, but have no idea what the bottom right is, let alone halfway down and right a bit. If you're clever enough to work out the z co-ordinate yourself you can use it for 3D objects too.

Icon Convert: converts bobs to icons, but there's no viewing facility.

Icon Utility by D. E. Lewis: to grab icons, sprites and bobs from IFF files. Instructions on disc. The "rubber band" for boxing your icon is white, so if your picture is black and white you may need to peer a bit.

Instrument Converter: Francois' soundtracker to IFF converter.

Intro.Amos: I don't understand what this is, but it seems to need the compiler! Please, people, put comments in your listings!

NIALL by Matthew Peck: a conversion of a programme that's been around a while, the Non-Intelligent Language Learner. Type in sentences and gradually the computer will learn the words and phrases and start to reply with sense - the longer you persevere, the more it learns and the more sensible (sort of) the replies. Fun. The dictionary can print out.

SCBM - SinCosBank Maker by Yazoo of Bitmap: this is probably excellent but I don't understand the math. The tune's great, the instructions are very badly spelt.

Sprite swapper: icon to sprite and back - two procedures to include in your programmes.

Starfield: a parallax scrolling starfield, and-

Starrotate: a rotating starfield.

Vectorcreator by Mark Peyton: a

very simple black and white animation frame creator, with onion skin. Don't make your picture with more than 19 points - at 20 it crashes.

Wavy1: balls looping the loop, as it were.

Wavy2: A well drawn, well animated sprite of a man who runs when you press a key, and-

Wavy3: a whole screen full of the same man running.

AND finally-

Zone Editor by Mark Peyton: lets you load an IFF file and select zones. Useful for graphic adventures, but try Adventure Bank Creator first, maybe? A white box outline, again, sometimes tricky to see. As you can

NOTICE!

The old Arch has been revived by Andrew Suttor of Wanniasa and has been renamed **AMOS ROLE PLAYING MANIA**. Those wanted the Arch back, and that was a lot of you, write to Andrew at:

AMOS ROLEPLAYING MANIA

4 Holden Crescent

Wanniasa ACT 2903

Contact Andrew with code or ideas. He tells me that you don't have to be a programmer to be part of it. If you want to do Graphics or just design quests then get in touch and design some killer Roleplaying games!

BA415 Uncle Simes 4:

Load through AMOS. Needs Extra memory.

A small collection of programmes on what seems to be a one-man magazine disc, mostly very well commented.

Starfield: a very fast horizontally scrolling starfield which makes me a bit queasy. Adam points out that in real space the stars shouldn't appear to move at all, except when you're turning. So there.

Maze: self-solving mazes to demonstrate recursion. Very neat; you set the start and finish and it experiments and finds the way out.

Menu demo: part of a database. Click on screen to activate; a good-looking example.

Recursion 1 & 2: 1 reverses a text string for you, 2 does the same for numbers. Follow the instructions in the text to see what happens slowly, line by line. By taking out the apostrophe before the "follow" command the cursor brings the Follow Mode up a bit.

Parallax scroll: read the note about CText extensions, and delete other references to Ext.1. if you don't have CText. Also change "Blood" to "Load" as errors pop up. Gives a mad, over fast double scroll, background going one way, foreground going the other, with, of course, no text.

Vectors: A 3D object revolves in a black ground with the object making a "window" through to a coloured background. Good tune.

Dynamic: needs CText. Deleting "extension" references and changing "Blood" to "load" as above helps. Delete "Break off" to be able to test. You now have a textless demo.

A well presented disc with easily understandable notes on the routines.

BA419 Bounder - Gory Story:

Autoboos to Gory Story. Load through AMOS for CAT (Creative Adventure Toolkit) demo. Needs extra memory.

Gory Story is a CAT created adventure, pleasantly legible, and as usual I am hopelessly stuck right at the beginning: I've climbed down into the cave and been killed by bats. The beauty of the system is that you can use multiple commands: "Open the door, wear the shield and use the sword on the monster". The limitation, as usual, is that what you can use as a command is what the programmer thinks appropriate - now, you'd think you'd be allowed to "open" a rucksack, wouldn't you? But you can't. The CAT demo is a promo for the programme which allows you a quick intriguing look at it. Great for adventurers, if you're a text-only fan. But I resent those bats. P.S. Got'em!

BA424 Card Games by D. A.Lorner

Autoboos. OK on 1/2 meg except that it will jam up after you've read the instructions to each game, and require rebooting. If you don't need

the instructions there's no problem.

This is a great disc with eleven different varieties of Patience on it, plus two other games. The cards in the early games are rather crudely drawn and might become muddly on a TV rather than a monitor. The pips can be hard to identify when the cards overlap. Later games have a clearer deck. There are easy games, hard games, maddening games and boring games, but there's sure to be something you like. And if you don't go for cards, or the little "ping" gets too annoying, there are two other games on the disc: "Tray", a sliding tile game - remember you're moving the hole, not the tile, and a delightful gem in "Scoreline"

"Scoreline" is a well presented board game in which you are making lines of five vertically or horizontally on a five by five board, on a random throw of the dice: and of course, a dice has six sides, not five! Easy to learn, addictive to play, this disc kept disappearing off my desk and migrating to Adam's machine. David says: "This is MAD! It's clever," and Adam says "It's a cool game. I wish it remembered high scores. Hey guys - come and look at this - 1300!" And he doesn't go for puzzle games! Don't waste your free goes, there's a bit of strategy involved there.

The disc is very good value. You might like to add a Quickmouse or Zoom mouse utility called from the startup sequence.

BA425 AMOS Painting Package 2 by M. Burbidge.

Autoboos. When faced with a blank screen click for the icons and colour palette.

Includes a Convert Format utility to pack your IFF files. Docs on disc, and if you like it please send M. Burbidge some PD discs. Note that the Undo option is not implemented, so be warned! A few useful and fun options - you can add a grid, rotate colours and make pompoms with a starspray; you can filter out all the blue, or red, or green; you can mix the neighbouring colours in a spray pattern. There's a blue-and-white AMOS logo brush which acts like a paint roller and is quite fun, and all the AMOS fill patterns. The palette

control is awkward - you don't drag the sliders to their next location, you click on where you want them to go instead, - and there's a very weird icon like an upside-down snail; the icons aren't easy to see - and you must load the fonts before trying to use text or you'll crash the programme. It's quite good, but I do need that Undo function!

Now some Aussie discs:

AA131 Laser Bikes by Jason Hutchens of Owl Moon Software:

Autoboos. Needs extra memory. A variant of Light Cycles, of course. Great music, which Adam reliably identifies as a rip-off sample of Red Hot Chilli Peppers. Adds new obstacles to an old favourite, but I'm still not rapt. I really like to be represented by something more substantial than a one pixel wide line, especially in a two player game when I'm elbow to elbow fighting for space in front of the monitor. This version is so oversensitive to the joystick that the joystick recentring can cause backtracking (hence, Death), and if you are playing against the computer your opponent can bounce off obstacles which wipe you out, and can even cross his own line at right angles, which you seem to be unable to do. Not for me.

AA132 Utility Compilation by Ivor King.

Autoboos. Docs on disc. Three useful graphic utilities:

Co-ordinate and ASCII monitor: Load in screens, click on a point and it tells you what the co-ordinates are. Also type in a character to find out its ASCII code.

Colour Monitor: to test colour values and registers. This will go through all 4096 colours for you, subtle change by subtle change, showing you the values for each. The colours are displayed large, on most of the screen, which I like.

Pixel Reader: load an IFF piccie, click on any pixel, read its colour and register.

Also included: Mini-utilities, comprising List directory, Create directory, Rename file, Show free memory, Quit, show free disc space,

Delete file, List disc info to printer, close workbench. This group can be made into an AMOS accessory.

AA134 Moonbase by James Barker:

Autoboos. Needs extra memory.
Problem is, I don't know if this is a duff disc, or if I'm too dumb to play this game. This is a big, big game in "Sim City" style, but set on the moon. you are to set up and run a space colony. There are an awful lot of complicated and forgettable instructions and a stack of menus. For some reason it won't load at all on Adam's machine (3 meg, hard drive, 1.3) but didn't mind mine (2 meg, 1.2). When I played it I could build a limited number of initial buildings, but couldn't do much of anything else and couldn't make out what was happening. So I called in the "Sim City" expert and we figured out we had to go surveying to go any further, but we didn't seem to be allowed to do that. By which time we were very, bored. So EITHER there's a bug in it OR it takes far too much getting into before it starts to be interesting. Either way it's a bold attempt that seems to have failed. If you have better luck with it, let me know - I want very much to like this one, but I don't.

AA136 Dragnet by the Shadow of Darkness:

Autoboos. Needs extra memory.
A first game, and commendable as such, a lot of thought has gone into designing the screens and not so much into the gameplay. A vertically scrolling space shoot'em-up, you only have one life, and on being wiped out you are facing an attractive but s-l-o-w "game over" message before being sent back through the long intro sequence and high score tables. Patience wears thin long before the game gets interesting.

AA138 Lies by the Hitman and Golden Fleece of Karma productions:

Autoboos. Needs extra memory.
A long demo, with a quirky, fun opening, that then becomes a tedious history of the scrolling demo. There is also a game preview with a long, slow, hard to read story, in red on black, concluding "Damn, out of

buffer space again."

AA139 Disk Action by David & Coralie Tucker of DCat:

Autoboos. Needs extra memory.
The first screen up is a scanned in photo of a one-eyed tabby - presumably de catl. A nicely presented directory utility. Instructions on disc - click on the docs file and then on textdump to view them. Note the limited uses for squash and unsquash, and note that P stands for Parent (directory) NOT Parnet as stated. Click on the asterisk attached to a directory name to view the contents of the directory. There's a good IFF file viewer. Ideally I like to see both df0 and df1 directories at once, but this is clear to look at and efficient.

Look for the new AMOS Pro PD Listing in the next newsletter which will contain only AMOS Pro compatible programs.

Contributions of AMOS Pro programs are urgently needed.

AA140 Various programmes:

Autoboos. Launcher needs extra memory, though I can't think why.
Isplex's compilation with:

Hello: messages, instructions and details.

Slowprint: a routine for printing on screen at a very readable speed.

Sin-wave: a very pretty mouse led caterpillar of big delayed dots - great for a snakebyte game or something.

Fireworks: fairly minor white dots in various fireworks patterns - sweet, really.

Launcher: to run through AMOS change "Get disc fonts" to "Get fonts". A robot is approaching your missile launcher from the far right - starting off screen, but you can scroll to find him. As he tromps towards you, he is squashing your men. Use the mouse to set the trajectory of your missiles and fire to destroy the robot, preferably without hitting your

own men in the process. Simple and quite fun.

Also lurking on the disc, to be loaded through AMOS (needs extra memory), a little programme called Trip, which gives you a psychedelic picture with changing colours. The sixties live again, yeah!

AA142 Harvey Demo by Jason Harvey:

Run through AMOS with 3D installed.
Hooray, an interesting demo at last! This demo is very memory intensive indeed, so you may need to unplug peripherals before you can go very far. I found it ran happily to about half way. Then I had to go back to the editor and delete a few of the procedures I'd already seen, and run again so that the memory they'd taken up wasn't used, and it went further. (Keep the disc protected and DON'T save your new version of the programme unless you want to lose those procedures permanently!) The coding is neat so it's easy to identify the procedures. Even so I couldn't gain access to any of bits that needed the "Spin" routine. That said, this is a well presented series of routines and ideas, including some 3D animations, and Jason has a sense of humour, too. Jason could easily have presented these routines on another "Mixed routines" disc, but he's taken the trouble to put them together in an entertaining way. I'm impressed, though I would like to see those "Spin" bits.

AA143 The Book by S.T. (the Saint) Poole; and Multisave by Perry Mowbray.

Load through AMOS. The Book is a nice looking, small database for addresses only, limited to 200 entries including a space for a "buisness" phone number. Why doesn't somebody out there write an AMOS compatible spell checker?

Multisave doesn't seem to run properly. Whatever I try it crashes my system.

AA144 by Graffiti Dog John K. Rutterford:

Autoboos. Needs extra memory for Balloon.

Two games. Balloon is a two player game in which shooting balloons will

stop the monster catching you and make him catch your opponent first. Don't let a stray arrow hit your opponent, though, or you die. David thinks that's unfair. Fun, but not very polished.

KMFH (Kamikaze Martins From Hell) at least has a wonderful title! A very fast, pretty shoot'em-up with great animated aliens which can come up from underneath you! To add to it, you score not by shooting the alien ship, but by catching the little green men that then fall out.

Also, to be loaded from AMOS, a rather tricky file copier.

And finally:

AA145 Check this out #3 by Jason Chan:

Autoboost to RAAP. Needs extra memory.

What would we do without an offering from Jason Chan! **RAAP** (Run All AMOS Programmes) is a utility which loads and runs an AMOS programme from any disc directly, so that you don't have to load AMOS and load the code into the Editor first. It pulls up the disc's directory, and away you go. It's a very good idea for running those nifty programmes which don't autoboot, like *The Book* (above).

Unfortunately you have to reboot after the end of each programme, so you'll still probably want to go through AMOS when first investigating a compilation disc. Note that if you try to load the instructions (which aren't very useful) through RAAP they will probably crash - This is because they are looking for some music which isn't where it's supposed to be. Load through AMOS and delete the line calling the music to read them.

HepB: of all things, an information screen on hepatitis. I can't think why it's on the disc, and I'm afraid anything about viruses, human or otherwise, has me reaching for my faithful copy of *NoVirus*, just to be on the safe side. I'm glad to report that *NoVirus* thinks I'm paranoid in this case.

Clowns: a pair of jolly animated clowns jumping for lollipops - Jason

suggests you might like to write the game to use them!

Viewscenes: a good slide show viewer currently set up to show 3 Vista-style landscapes which are also on the disc.

STConverter: a timing fix for Francois Soundtracker to AMOS music converter.

Pepsi Amal demo: an animated Pepsi can.

Improved AMOS ideas: nicely presented alternate looks for new AMOS screens and requesters, in much prettier colours.

So there you are. No prizes for guessing that my favourite discs are *Quadrix* and *Tetren*. But it's nice to see some more imaginative demos, and some generally more ambitious and better presented work. Amble on

And as usual, the new stuff from England. There's a quite a few new disks lined up, although the new North American disks have not arrived as yet. They should be available on the catalogue disk in a few weeks.

BRITISH PD

- BA 431... KARATE WORM II. Game
- BA 432... ALLIANCE II. Game
- BA 433... SPRITE 600. Art
- BA 434... GRAPHER/LIFE. Utility
- BA 435... DRIVER. Game
- BA 436... AMOS MENUS. Utility
- BA 437... ZIRUS MEGADEMO. Demo
- BA 438... MATCH WITH HUMPTY. Education
- BA 439... CLASSIC COLLECTION. Games & Source Code
- BA 440... SCUM. Game
- BA 441... QUIZ CHALLENGE. Game
- BA 442... SLIDE SHOW CONSTRUCTION. Utility
- BA 443... PICTURES. Game
- BA 444... BRAIN DAMANGE. Demo
- BA 445... WEEDIE. Game
- BA 446... UNCLE SIMES 3. Needs C-Text. Source Code
- BA 447... AMOS Programs 33. Source Code
- BA 448... CARD GAMES 2. Game
- BA 449... LUKE MILLER'S MUSIC 7. Music
- BA 450... TOTALLY AMOS PROGS 1. Source Code
- BA 451... NODDY's PLAYTIME. Demo
- BA 452... RAMOS PRO. Utility
- BA 453... AMOS PROGRAMS 34. Source Code
- BA 454... U-TILITIES 1. Utility
- BA 455... SHORT OUT. Game
- BA 456... DIGITAL ORGASM. Demo
- BA 457... VOLBER'S UTILITIES.

- Utility
- BA 458... TOP OF THE LEAGUE. Game
- BA 459... PICK'N'MIX 2. Source Code
- BA 460... TOTALLY AMOS PROGS 2. Source Code
- BA 461... FRUST. Game
- BA 462... FOOTBALL/SPEEDY REEDY. Game
- BA 463... AMOS PROGRAMS 35. Source Code
- BA 464... JAK ROUTINES 1. Source Code

GENERAL BRITISH PD

These disks contain Compiled AMOS Programs but no Source Code.

- GA 144... STARCATZ. Utility
- GA 145... TURBOTEXT. Utility
- GA 146... DAISY PAINTER. Utility
- GA 148... CYCLIC. Demo
- GA 149... SYSTEM DISORDER. Demo
- GA 150... MAGIC TURN & UTES. Game
- GA 151... MONSTRAK. Game
- GA 153/4. FUMBLE. Demo
- GA 156... ZIRUS WONDERLAND. Demo
- GA 158... ACCOUNT MASTER. Utility
- GA 159... SUPER LEAGUE '93. Game
- GA 160... QUIZ TIME. Game
- GA 161... VECTORS R US. Demo
- GA 162... SPACED OUT. Game
- GA 163... BLITZ. Game
- GA 164... MUSIC DEMO 1. Music
- GA 165... MUSIC DEMO 2. Music
- GA 166... MUSIC DEMO 3. Music
- GA 167... 3 GAMES. Game
- GA 168... SPACED OUT 2. Demo
- GA 169... HYSTERIA. Demo
- GA 175... ROACH MOTEL I Game
- GA 180... TABMASTER/HECTIC IV DEAR DIARY. Game/Utility
- GA 181... ADDRESSPRINT. Utility
- GA 183... DERBY DAY/ DRAWMORE/

NEW AUSTRALIAN DISKS

- AA 144 - BALLOON/BATCH FILE COPIER/BLAST IT 2 by J. Rutterford.
- AA 144 - MAKKA MANIA - A wild and wacky demo from Peter McMurray.
- AA 145 - CHECH THIS OUT #3 - Another mix from Jason Chan: CLOWNS, the old 2 clowns on a see-saw and bust the balloons game, AMOSFILE-EXECUTER, loads your programs via the requester, ST-ConverterV2_2, Yet another Module converter; also fixes timing problems, PEPSI DEMO plus more...
- AA 146 - CLOSE QUARTERS - Choose your weapon, your character and go in and give 'em hell! Two players required.
- AA 147 - HORSE RACING - By D. Hare. I don't even like horse racing but

