```
   AAA   TTTTT  AAA   RRRR   III        ****           SSS   TTTTT
  A   A    T   A   A  R   R   I         ****         S     S    T
  A   A    T   A   A  R   R   I         ****         S          T
  AAAAA    T   AAAAA  RRRR    I        ******         SSS       T
  A   A    T   A   A  R   R   I       **  **  **         S      T
  A   A    T   A   A  R   R   I      **    **    **    S     S   T
  A   A    T   A   A  R   R  III    **     **     **   SSS      T

  DDDD  EEEEE  V   V  EEEEE  L       OOO   PPPP   EEEEE RRRR   SSS
  D   D E      V   V  E      L      O   O P    P E      R   R S     S
  D   D E      V   V  E      L      O   O P    P E      R   R S
  D   D EEEE   V   V  EEEE   L      O   O PPPP   EEEE   RRRR   SSS
  D   D E      V   V  E      L      O   O P      E      R   R      S
  D   D E       V V   E      L      O   O P      E      R   R S     S
  DDDD  EEEEE    V    EEEEE  LLLLL   OOO  P      EEEEE  R   R  SSS
```

Question and Answer Bulletin   April 1986
Copyright (C) 1986 by Atari Corp.   "all rights reserved"
1196 Borregas Ave.,   Sunnyvale, Ca.   94086

Here are the latest questions from the Atari developers mailbag
as answered by John Feagans, Director of Software Technology.
Leave questions on Compuserve for PIN 70007,1072 or GO PCS57 for
Atari developer SIG information.

This months contents

1. Corrections

In the March 1986 issue I mis-stated a description of the real VT-52 keyboard. This device actually does have a numeric pad and cursor keys. The difference between the numeric pad that it has and the VT-220 style that the ST has is that the cursor keys are located in the positions of the *, -, +, and Enter key on the numeric pad. The enter key itself is split into two smaller keys.

----

2. BIOS

*

Q: How do I make the buffer larger for communication over the MIDI port?

A: Use the xbios call Iorec(MIDI) to get the address of a structure which contains a pointer to the existing buffer. Change it to point to

your new and bigger buffer. Be sure to restore things when you exit your program for the benefit of the application that follows yours.

----

3. DOS

*

Q: How do I get the address of the DTA buffer for use with the Fsfirst( ) command?

A: There are two ways that you can do this: 1) Allocate a buffer yourself and use the Fsetdta(ptr) to tell the system about it, or 2) ptr=Fgetdta to use the one already defined.

*

Q: I am running out of stack space when I link with apstart.o. Where do I edit to increase the size?

A: Look for the label ustk in apstart.s. Directly preceding it you will find a statement ".ds.1 256". This statement allocates 256 * 4 words, or about 1k of stack. To increase the stack to 8k, make it ".ds.1 2048". Remember that declaring local arrays in C will eat more stack space-- allocate accordingly. To make the apstart.o you must assemble your new source by as68 -1 -u apstart.s.

----

4. VDI

*

Q: How do I transfer a color image from memory to the screen without copying the background around the image?

A: You need to construct a mono-plane mask with zeros where the image in memory will be transparent and ones where the image in memory will occupy space. Use the copy raster transparent (vrt_cpyfm) with transparent mode to cut a hole into the screen. Second use vro_cpyfm with "or" mode 7 to move the color image to the screen. See section 6 of the VDI manual for further details.

----

5. AES

*

Q: How do I hide and show the mouse?

A: Use the graf_mouse call. A useful thing to include in a header file are the following definitions:
```
#define HIDE_MOUSE
        graf_mouse(256,0xOL)

#define SHOW_MOUSE
        graf_mouse(257,0xOL)
```

*

Q: In the mentst.c example program, why are so many parameters in the evnt_multi call zeroed out?

A: The reason for this is that in this example program we are only interested in returning if there was mouse event. We are not interested in keyboard, timer, or button events, hence we have a dummy pointer as place holders in those parameter positions of the evnt_multi.

----

6. Desktop

*

Q: How does the AUTO folder work?

A: If you create a folder named AUTO, the system looks for it on power-up boot and will execute any programs it finds inside. If there is more than one program, the programs will be executed in the order that they were created in the folder. These programs are named *.prg but it is not possible to have an auto-starting GEM application. However, the programs may use the VDI, DOS, and BIOS. A good example to try is to place STWriter inside an AUTO folder and it will start running on power-up. When you exit STWriter, initialization resumes and execution brings up the desktop. AUTO folders are generally used for drivers such as those for the hard-disk and printers.

Q: How can you get the name of the data file that initiated a program with an installed document type?

A: For programs that have undergone the install application option at the desktop, it is easy to get the name of the document. The desktop program places the file name string in the command tail. The command tail starts 80 hex into the base page. You can get this information by accessing the base page directly, or by main(argv,argc) from a C program that uses gemstart.o.

----

7. Accessories

----

8. BASIC

*

Q: The accountants in my 50 million dollar company do not like my basic programs as large numbers will end up with a few pennies added to the end. What can I do?

A: You are seeing the effect of a single precision scientific math package. The maximum number of significant digits that can be stored is 6 to 7. Decimal fractions cannot be accurately represented by a binary floating point structure. To overcome the latter limitation, you can store all numbers as pennies, only inserting the decimal point when inputting or outputting them. To gain extra significant digits you can save each number as an integer whole and fractional part. To really gain significant digits you might consider writing some math routines which operate and store numbers in strings.

----

9. LOGO

*

Q: I keep getting an edit buffer full error. I have tried turning off buffered graphics and not loading desk accessories to get more space. Will getting ROMs help me?

A: The edit buffer is a fixed size. You need to break up your program into smaller

segments and load them in. Getting the ROMs would only give you more program storage and would not increase the size of the edit buffer.

----

10. Development Tools

*
Q: How do I get the C-Compiler to make a listing of my program with line numbers

A: The Alcyon C currently supplied in the developers kit does not have an option for generating a listing. Errors that are reported can be found by using the editor to step to that line in your source. I hope that someone out there will write a print/header and line number generating program to share with us in the data libraries in the developers SIG.

*
Q: I am including stdio.h and linking with gemlib and libf, but I still have undefined symbols etoa and ftoa in the link step when I use printf.

A: The order of the library files in the link are important. Make sure that libf is listed after gemlib. If you have any doubts you can also list it as libf gemlib libf.

*
Q: I have a RAM disk program. I would like to speed up my links by putting all the linker created temporary files on the RAM disk. How do I command the linker to do this?

A: For LO68, -F d: The -F

option specifies the drive on which temporary files are created. The variable d: is the drive designation. For LINK68 TEM[d:].

----

11. New On Compuserve

In data library 7 (for registered Atari Developers only) in the Atari Developers SIG on Compuserve, the following files are new this month:

WRITE.O	C RTL fix
LSTOUT.O	" "
CC.BAT	" "
GEMDOS.DOC	New improved.
BCD.DOC	Math package.
GETDOB.C	" "
MATH.OBJ	" "
PRINTD.C	" "
RELMOD.C	New improved.
SLDSET.PRG	Image service.
SLIDE.DOC	" "
SNPSHT.TOS	" "
SNPSVE.PRG	" "
TEMPLA.C	GEM skeleton.
FED.PRG	font editor.
FEDFIX.TTP	" "
FEDFIX.C	" "
FED.DOC	" "
FRDME.DOC	" "
FONTUSE.DOC	" "
FONTUSE.S	" "
STHDW.DOC	New document.
LARGE.S	Porting aids.
QA2.DOC	Previous Q/A.