AUSTRALIAN OS9 NEWSLETTER

EDITOR: Gordon Bentzen   (07) 344-3881
SUB-EDITOR: Bob Devries   (07) 278-7209
TREASURER: Jean-Pierre Jacquet   (07) 372-4675
                   Fax Messages   (07) 372-8325
LIBRARIAN: Rod Holden   (07) 200-9870
CONSULTANT: Don Berrie   (079) 75-3537
SUPPORT: Brisbane OS9 Users Group

CONTENTS

OS9 OS9 OS9 OS9 OS9 OS9 OS9 OS9 OS9
OS9 OS9 OS9 OS9 OS9 OS9 OS9 OS9 OS9

OS9 OS9 OS9 OS9 OS9 OS9 OS9 OS9 OS9
OS9 OS9 OS9 OS9 OS9 OS9 OS9 OS9 OS9

EDITOR : Gordon Bentzen   **TREASURER** : Jean-Pierre Jacquet
SUBEDITOR : Bob Devries   **LIBRARIAN** : Rod Holden
                    SUPPORT : Brisbane OS9 Level 2 Users Group.

Hello! is anybody there? At the time of our last subsciption renewal we asked members to again return a Membership Application/Renewal form and we thank you again for your continued support. Also, if I recall correctly, we were pleased with the number of responses which indicated they would be able to contribute material for inclusion in the newsletter. Hey this was really encouraging! but where have you guys gone?

We are now half way through the current subscription year and I would not like to see anybody miss out on having their article published due to the rush in the last half of the year. Hint - please send your material as soon as possible :-)   <grin>

Does anybody use the source code which is published from time to time? Well I can report that I compiled the 'C' source for Bootsplit by Bob Devries from last months newsletter. I used the 'C' compiler from OS-9000 after only a couple of changes to the source and of course assistance from Bob. (see the details later in this newsletter).

Anyway the programme does work just fine on the OS-9000 Boot file and I also understand that Don Berrie had similar success with OSK and the MM One. Thanks Bob for this simple but very useful utility. I would be interested to know if anybody compliled a CoCo OS9/6809 version.

We do need to know what you think of the material we present in each newsletter as we want to cater to your needs and wants. This may make our job a little more difficult but we are willing to give it a go.

I will have to leave now to go stand by the mailbox.
                              Until next time, Cheers Gordon.

```
          __      /\
        /   \__/   \
       /   OZ - OS9  \
       \   RiBBs 2.1  /
        \___-----\_/
              o
```

The National OS9 Usergroup
(07)-805-5980   <====== New Phone number!
300/1200/2400/9600/14400 baud.
20:00 to 21:30 HRS.(AEST)
(8N1)

Co-ordinator: Bob Devries (07)-278-7209
Sysop: Rod Holden

This is (RiBBS).... A Tandy Coco Based BBS program.
This BBS is accessable to Usergroup Members ONLY!
Feel free to look around , and test out the options.

OS9 for Ever !!!!

This is your Sysop giving you a directory listing of what type of software is available, feast your eyes on this lot;

```
DNLD                              DNLD/OS9_GAM/mixup.ar
DNLD/OS9_APP                      DNLD/OS9_GAM/nflschd.ar
DNLD/OS9_APP/ad9.ar               DNLD/OS9_GAM/nim.ar
DNLD/OS9_APP/areacode.ar          DNLD/OS9_GAM/pacos9.ar
DNLD/OS9_APP/astronl.ar           DNLD/OS9_GAM/psychic.ar
DNLD/OS9_APP/cal2.ar              DNLD/OS9_GAM/rescue.pak
DNLD/OS9_APP/calcl3.pak           DNLD/OS9_GAM/sokoban.ar
DNLD/OS9_APP/catlog.pak           DNLD/OS9_GAM/spy.pak
DNLD/OS9_APP/cclock.arc           DNLD/OS9_GAM/stones.pak
DNLD/OS9_APP/checks.pak           DNLD/OS9_GAM/strandl.arc
DNLD/OS9_APP/dbase.ar             DNLD/OS9_GAM/tetris.ar
DNLD/OS9_APP/dcmessag.arc         DNLD/OS9_GAM/trek.ar
DNLD/OS9_APP/dynafix.ar           DNLD/OS9_GAM/ttt.ar
DNLD/OS9_APP/dynapatc.ar          DNLD/OS9_GAM/valspeak.ar
DNLD/OS9_APP/ed.ar                DNLD/OS9_GAM/wrevenge.pak
DNLD/OS9_APP/exfind.ar            DNLD/OS9_GAM/yahtzee.ar
DNLD/OS9_APP/filer.ar             DNLD/OS9_GAM/zap.pak
DNLD/OS9_APP/hdline.ar            DNLD/OS9_GRA
DNLD/OS9_APP/jtdictl4.ar          DNLD/OS9_GRA/altfont.bin
DNLD/OS9_APP/label20l.pak         DNLD/OS9_GRA/ansishow.ar
DNLD/OS9_APP/ledger45.pak         DNLD/OS9_GRA/banner.pak
DNLD/OS9_APP/loans.ar             DNLD/OS9_GRA/block.ar
DNLD/OS9_APP/mail.ar              DNLD/OS9_GRA/bouncer.ar
DNLD/OS9_APP/man.ar               DNLD/OS9_GRA/brot.ar
DNLD/OS9_APP/number.ar            DNLD/OS9_GRA/cartog.ar
DNLD/OS9_APP/os9mail.pak          DNLD/OS9_GRA/cgp220.pak
DNLD/OS9_APP/paymster.ar          DNLD/OS9_GRA/cgpdump.ar
DNLD/OS9_APP/quiz.pak             DNLD/OS9_GRA/coco.ar
DNLD/OS9_APP/register.ar          DNLD/OS9_GRA/colordmp.pak
DNLD/OS9_APP/sc.pak               DNLD/OS9_GRA/cursive.ar
DNLD/OS9_APP/sculptor.ar          DNLD/OS9_GRA/cviewgf2.pak
DNLD/OS9_APP/sled.ar              DNLD/OS9_GRA/dragfont.ar
DNLD/OS9_APP/swise.pak            DNLD/OS9_GRA/fern.ar
DNLD/OS9_APP/tax90.ar             DNLD/OS9_GRA/fonted.ar
DNLD/OS9_APP/vi.ar                DNLD/OS9_GRA/fonts.ar
DNLD/OS9_APP/vull.pak             DNLD/OS9_GRA/fstgrf.ar
DNLD/OS9_APP/wp3l.ar              DNLD/OS9_GRA/gcomp.ar
DNLD/OS9_APP/xled22.pak           DNLD/OS9_GRA/gfx2.ar
DNLD/OS9_GAM                      DNLD/OS9_GRA/gfx2fix.ar
DNLD/OS9_GAM/adventur.pak         DNLD/OS9_GRA/gfxcmds.pak
DNLD/OS9_GAM/astrol.ar            DNLD/OS9_GRA/grab.ar
DNLD/OS9_GAM/batlship.ar          DNLD/OS9_GRA/ibm.fnt
DNLD/OS9_GAM/blackjac.pak         DNLD/OS9_GRA/macview.ar
DNLD/OS9_GAM/cribge.ar            DNLD/OS9_GRA/max9.ar
DNLD/OS9_GAM/dungeons.ar          DNLD/OS9_GRA/mge2vef.ar
DNLD/OS9_GAM/games.ar             DNLD/OS9_GRA/morewin.ar
DNLD/OS9_GAM/gameshow.pak         DNLD/OS9_GRA/ncolor.pak
DNLD/OS9_GAM/gemquest.pak         DNLD/OS9_GRA/orb.pak
DNLD/OS9_GAM/glife.ar             DNLD/OS9_GRA/palette.ar
DNLD/OS9_GAM/hordes.ar            DNLD/OS9_GRA/pci.pak
DNLD/OS9_GAM/kq3.pat              DNLD/OS9_GRA/prf.ar
DNLD/OS9_GAM/lander.ar            DNLD/OS9_GRA/rayboun2.arc
DNLD/OS9_GAM/lotto.ar             DNLD/OS9_GRA/raybounc.ar
DNLD/OS9_GAM/lsl2sol.pak          DNLD/OS9_GRA/rgb.ar
DNLD/OS9_GAM/mastmind.ar          DNLD/OS9_GRA/rlepic.pak
```

DNLD/OS9_GRA/s.ar
DNLD/OS9_GRA/sea3d.pak
DNLD/OS9_GRA/shapes.ar
DNLD/OS9_GRA/show.ar
DNLD/OS9_GRA/spiro.ar
DNLD/OS9_GRA/vefprt.ar
DNLD/OS9_GRA/view44.ar
DNLD/OS9_GRA/waterfal.ar
DNLD/OS9_GRA/window.ar
DNLD/OS9_GRA/wload.ar
DNLD/OS9_GRA/worms.ar
DNLD/OS9_GRA/wreath.ar
DNLD/OS9_GRA/sview.pak
DNLD/OS9_MUL
DNLD/OS9_MUL/aif_mgr.ar
DNLD/OS9_MUL/aifbnc.ar
DNLD/OS9_MUL/bonk.pak
DNLD/OS9_MUL/card17a.ar
DNLD/OS9_MUL/carmen.ar
DNLD/OS9_MUL/cubist.ar
DNLD/OS9_MUL/deadic.ar
DNLD/OS9_MUL/ed30.ar
DNLD/OS9_MUL/edcon.ar
DNLD/OS9_MUL/edfont.ar
DNLD/OS9_MUL/edpat30a.ar
DNLD/OS9_MUL/edptr.ar
DNLD/OS9_MUL/envelope.pak
DNLD/OS9_MUL/fntman.pak
DNLD/OS9_MUL/fs2mv.ar
DNLD/OS9_MUL/gfrmat.ar
DNLD/OS9_MUL/glabel.ar
DNLD/OS9_MUL/gshell.ar
DNLD/OS9_MUL/gshell2.ar
DNLD/OS9_MUL/gshpat.ar
DNLD/OS9_MUL/guib.ar
DNLD/OS9_MUL/guib11.pak
DNLD/OS9_MUL/icon.pak
DNLD/OS9_MUL/iconedit.ar
DNLD/OS9_MUL/icons.ar
DNLD/OS9_MUL/icons.pak
DNLD/OS9_MUL/jbudget.pak
DNLD/OS9_MUL/kq3.ar
DNLD/OS9_MUL/manager2.pak
DNLD/OS9_MUL/multedit.ar
DNLD/OS9_MUL/multii.ipc
DNLD/OS9_MUL/multii.pak
DNLD/OS9_MUL/mv2pat.ar
DNLD/OS9_MUL/mvaifs.ar
DNLD/OS9_MUL/mvskel.ar
DNLD/OS9_MUL/mvtest.ar
DNLD/OS9_MUL/newdir.ar
DNLD/OS9_MUL/pacaif.ar
DNLD/OS9_MUL/phonbase.pak
DNLD/OS9_MUL/pointers.pak
DNLD/OS9_MUL/ratmaze.ar

DNLD/OS9_MUL/shelicon.pak
DNLD/OS9_MUL/smate.ar
DNLD/OS9_MUL/smenu.ar
DNLD/OS9_MUL/solitare.ar
DNLD/OS9_MUL/stereovu.ar
DNLD/OS9_MUL/submv.ar
DNLD/OS9_MUL/superike.ar
DNLD/OS9_MUL/vicon.pak
DNLD/OS9_MUL/windfix.ar
DNLD/OS9_MUL/yaicon.ar
DNLD/OS9_MUL/yatzee.ar
DNLD/OS9_MUS
DNLD/OS9_MUS/digitize.ar
DNLD/OS9_MUS/mfcvpl.ar
DNLD/OS9_MUS/play.ar
DNLD/OS9_MUS/play3.ar
DNLD/OS9_MUS/play4.ar
DNLD/OS9_MUS/playmus.pak
DNLD/OS9_MUS/playutil.pak
DNLD/OS9_MUS/sspak.ar
DNLD/OS9_MUS/ultibox.ar
DNLD/OS9_MUS/umuse.ar
DNLD/OS9_PRO
DNLD/OS9_PRO/alib.pak
DNLD/OS9_PRO/b09util.pak
DNLD/OS9_PRO/carray.ar
DNLD/OS9_PRO/cc2.pak
DNLD/OS9_PRO/clib.ar
DNLD/OS9_PRO/cpatch.ar
DNLD/OS9_PRO/devpakhp.ar
DNLD/OS9_PRO/devsys.pak
DNLD/OS9_PRO/disasm.ar
DNLD/OS9_PRO/header.ar
DNLD/OS9_PRO/os9p4.ar
DNLD/OS9_PRO/pilot1.ar
DNLD/OS9_PRO/pilot2.ar
DNLD/OS9_PRO/sorts_09.ar
DNLD/OS9_PRO/Secad.lha
DNLD/OS9_SYS
DNLD/OS9_SYS/cache.ar
DNLD/OS9_SYS/cc3dis.ar
DNLD/OS9_SYS/cc3disk.ar
DNLD/OS9_SYS/cc3go.ar
DNLD/OS9_SYS/chdisk.ar
DNLD/OS9_SYS/clockbb.ar
DNLD/OS9_SYS/copypatc.ar
DNLD/OS9_SYS/cpprint.ar
DNLD/OS9_SYS/dirosk.ar
DNLD/OS9_SYS/distsasi.ar
DNLD/OS9_SYS/distscsi.ar
DNLD/OS9_SYS/esw110.ar
DNLD/OS9_SYS/floppy35.ar
DNLD/OS9_SYS/kfutil.ar
DNLD/OS9_SYS/nubbclk.ar
DNLD/OS9_SYS/os9p2.ar

DNLD/OS9_SYS/os9p3.ar
DNLD/OS9_SYS/partgen.ar
DNLD/OS9_SYS/patches.ar
DNLD/OS9_SYS/ram_736k.asm
DNLD/OS9_SYS/ramdisk.pak
DNLD/OS9_SYS/rbf30.ar
DNLD/OS9_SYS/scfed2.ar
DNLD/OS9_SYS/spedup12.ar
DNLD/OS9_SYS/vdgdesc.ar
DNLD/OS9_SYS/vrn.ar
DNLD/OS9_SYS/widedirs.ar
DNLD/OS9_SYS/wx.ar

DNLD/OS9_SYS/xtbootfx.pak
DNLD/OS9_TEL

We are currently trying to obtain the rest of the OS9 Community Network files which is approximately 600 files from the States. As soon as we receive them a directory listing will be published, and from a list I have seen it looks real interesting. See you in the bit stream, Happy CoCoing.

Sysop
Rod Holden

oooooooooooOOOOOOOOOOOooooooooooo

BootSplit - OS9000 changes

With the help of an OS9000 user, I have been able to make the necessary modifications to my C programme BootSplit. The changes necessary are:

1. Change the name of the structure from **modhcom** to **mh_com**.
   So **struct modhcom module** becomes **struct mh_com module**.

2. Change the field names in the structure from **_m** to **m_**.
   So **module._msize** becomes **module.m_size**.

Regards, Bob Devries

oooooooooooOOOOOOOOOOOooooooooooo

Playing back sound samples

Recently, I had the chance to check out a CD-ROM which was filled with IBM files of all sorts. Of course, most of these were of no use to me, but one file, AUDIOFMT.ZIP caught my eye. After I unzipped it, I found that it was a collection of text files with details of quite a few different sampled sound file formats. I feel that these could be useful to the OS-9 world, so I am going to include them in the newsletter a few at a time, as space permits. Most of the files give the data break-down, so that a programme can be written to decode the sound sample file, and play it back on any computer. The file types include **8SVX**, **VOC**, **WAV**, and **AIFF**.

Bob Devries

AIFF Format (Audio IFF) and AIFC

This format was developed by Apple for storing high-quality sampled sound and musical instrument info; it is also used by SGI and several professional audio packages (sorry, I know no names). An extension, called AIFC or AIFF-C, supports compression (see the last item below).

I've made a BinHex'ed MacWrite version of the AIFF spec (no idea if it's the same text as mentioned below) available by anonymous ftp from ftp.cwi.nl [192.16.184.180]; the file is /pub/audio/AudioIFF1.2.hqx. But you may be better off with the AIFF-C specs, see below.

Mike Brindley (brindley@ece.orst.edu) writes:

"The complete AIFF spec by Steve Milne, Matt Deatherage (Apple) is available in 'AMIGA ROM Kernal Reference Manual: Devices (3rd Edition)' 1991 by Commodore-Amiga, Inc.; Addison-Wesley Publishing Co.; ISBN 0-201-56775-X, starting on page 435 (this edition has a charcoal grey cover). It is available in most bookstores, and soon in many good librairies."

According to Mark Callow (msc@sgi.com):

A PostScript version of the AIFF-C specification is available via anonymous ftp on FTP.SGI.COM (192.48.153.1) as /sgi/aiff-c.9.26.91.ps.

RIFF WAVE (.WAV) file format

RIFF is a format by Microsoft and IBM which is similar in spirit and functionality as EA-IFF-85, but not compatible (and it's in little-endian byte order, of course :-). WAVE is RIFF's equivalent of AIFF, and its inclusion in Microsoft Windows 3.1 has suddenly made it important to know about.

Rob Ryan was kind enough to send me a description of the RIFF format. Unfortunately, it is too big to include here (27 k), but I've made it available for anonymous ftp as ftp.cwi.nl:/pub/audio/RIFF-format.

And here's a pointer to the official description from Matt Saettler, Microsoft Multimedia:

"The complete definition of the WAVE file format as defined by IBM/Microsoft is available for anon. FTP from ftp.uu.net in the vendor/microsoft/multimedia directory."

(Rob Ryan's version may actually be an extract from one of the files stored there.)

IFF/8SVX Format

Newsgroups: alt.binaries.sounds.d,alt.sex.sounds
Subject: Format of the IFF header (Amiga sounds)
Message-ID: <2509@tardis.Tymnet.COM>
From: jms@tardis.Tymnet.COM (Joe Smith)
Date: 23 Oct 91 23:54:38 GMT
Followup-To: alt.binaries.sounds.d
Organization: BT North America (Tymnet)

The first 12 bytes of an IFF file are used to distinguish between an Amiga picture (FORM-ILBM), an Amiga sound sample (FORM-8SVX), or other file conforming to the IFF specification. The middle 4 bytes is the count of bytes that follow the "FORM" and byte count longwords. (Numbers are stored in M68000 form, high order byte first.)

FutureSound audio file, 15000 samples at 10.000KHz, file is 15048 bytes long.

```
0000: 464F524D 00003AC0 38535658 56484452    FORM..:.8SVXVHDR
       F O R M    15040 8 S V X  V H D R
0010: 00000014 00003A98 00000000 00000000    ......:.........
         20     15000        0        0
0020: 27100100 00010000 424F4459 00003A98    '.......BODY..:.
      10000 1 0   1.0   B O D Y     15000
```

0000000..03 = "FORM", identifies this as an IFF format file.
FORM+00..03 (ULONG) = number of bytes that follow. (Unsigned long int.)
FORM+03..07 = "8SVX", identifies this as an 8-bit sampled voice.

????+00..03 = "VHDR", Voice8Header, describes the parameters for the BODY.
VHDR+00..03 (ULONG) = number of bytes to follow.
VHDR+04..07 (ULONG) = samples in the high octave 1-shot part.
VHDR+08..0B (ULONG) = samples in the high octave repeat part.
VHDR+0C..0F (ULONG) = samples per cycle in high octave (if repeating), else 0.
VHDR+10..11 (UWORD) = samples per second. (Unsigned 16-bit quantity.)
VHDR+12   (UBYTE) = number of octaves of waveforms in sample.
VHDR+13   (UBYTE) = data compression (0=none, 1=Fibonacci-delta encoding).
VHDR+14..17 (FIXED) = volume. (The number 65536 means 1.0 or full volume.)

????+00..03 = "BODY", identifies the start of the audio data.
BODY+00..03 (ULONG) = number of bytes to follow.
BODY+04..NNNNN   = Data, signed bytes, from -128 to +127.

```
0030: 04030201 02030303 04050605 05060605
0040: 06080806 07060505 04020202 01FF0000
0050: 00000000 FF00FFFF FFFEFDFD FDFEFFFF
0060: FDFDFF00 00FFFFFF 00000000 00FFFF00
0070: 00000000 00FF0000 00FFFEFF 00000000
```

```
0080: 00010000 000101FF FF0000FE FEFFFFFE
0090: FDFDFEFD FDFFFFFC FDFEFDFD FEFFFEFE
00A0: FFFEFEFE FEFEFEFF FFFFFEFF 00FFFF01
```

This small section of the audio sample shows the number ranging from -5 (0xFD) to +8 (0x08). Warning: Do not assume that the BODY starts 48 bytes into the file. In addition to "VHDR", chunks labeled "NAME", "AUTH", "ANNO", or "(c)" may be present, and may be in any order. You will have to check the byte count in each chunk to determine how many bytes to skip.

ooooooooooOOOOOOOOOOOooooooooooo

Scientific Calculator for OS9

In among the large quantity of public domain software we have, I found a scientific calculator programme for OS9. It is in the OCN library, and is called SC.AR. It is in the OS9_APP subdirectory. What follows is the comments from the front of the source code, which tells how it is used, and which functions are available. The source code should compile on either OS9/6809, or OS9/68000, and probably on OS9000 as well.

```
/* sc.c

   Scientific Calculator, Version 2.1 for OS9 and OS9/68000

   Copyright 1989 John M Semler

   NOTE: You must use good mathematical common sense when interpreting the
   results returned by "sc".

   ********************************************************************

   Revision History -

     11/03/89  Modified the source code so that it can be compiled under
               OS9 and OS9/68000.

     11/03/89  Corrected memory management bug involving variable storage.

   ********************************************************************

   General Overview:

     Input into "sc" consist of a series of statements of which there are
     two types. Expression statements are mathematical strings that will
     evaluate to a single result. Command statements are strings that
     will perform some action but return no results. All statements must
     terminate with the character ';'.

   Binary operators:

     Operands can either be real numbers, real matrices, complex numbers,
     complex matrices, named constants ("e", "i", and "pi"), variable
     references, and or valid expressions unless otherwise restricted by
     mathematical convention.
```

Precedence and associativity of the binary and unary operators:

Operator	Precedence	Associativity	Comments
+	5 (high)	right-to-left	Unary plus
-	5	right-to-left	Unary minus
^	4	right-to-left	Power operator
/	3	left-to-right	Division
*	3	left-to-right	Multiplication
-	2	left-to-right	Subtraction
+	2	left-to-right	Addition
=	1 (low)	right-to-left	Assignment **

The evaluation order of expressions can be modified through the use of parenthesis.

** Note: The assignment operator can appear anywhere in the expression. However, the left operand must be a variable reference.

Built-in constants:

```
pi = 3.141592653589796
e  = 2.718281828450945
i  = <0,1>                 (The imaginary unit)
```

Complex numbers:

Two different forms are accepted: Angle bracket pairs, or expressions involving the imaginary unit. For instance, the complex expression $X = 3 + 4i$ can be entered as:

```
OS9:sc              OS9:sc              OS9:sc
X=3+4*i;    or      X=<3,4>;    or      X=5*e^(.927295218*i);
<3,4>               <3,4>               <3,4>
```

Matrices (n x m where n,m dictated by the size of the input buffer):

Matrices are entered using two levels of curly brackets. The inner-most levels of curly brackets will group row elements together. For instance, the complex matrix expression

$$
\begin{vmatrix} -1 & 0 & 3 \\ 7 & 1 & -1 \\ 2+i & 3 & 0 \end{vmatrix}^{-1} * \begin{vmatrix} 3 \\ 2 \\ 5 \end{vmatrix}
$$

can be entered as:

```
OS9:sc
{{-1,0,3},{7,1,-1},{2+i,3,0}}^-1*{{3},{2},{5}};
```

or for more readability

```
OS9:sc
{{-1,  0,  3},
```

```
{ 7,   1,  -1},
{2+i,  3,   0}}^-1 * {{3}, {2}, {5}};
```

Display register reference:

You can access the display register through the variable name "disp".

Variable references:

Variables may consist of a single letter followed by zero or more
occurrences of letters and digits terminating with zero or more
occurrences of dinks. Names that are reserved by "sc" may not be
used as variable names. The list of reserve names are as follows:
"clear", "list", "eng", "fix", "polar", "rect", "pi", "e", "i",
"disp", and the built-in function names.

Built-in functions (all functions operate with scalar operands unless
otherwise noted):

```
sin(x)      Returns the sine of x,
cos(x)      Returns the cosine of x,
tan(x)      Returns the tangent of x,
asin(x)     Returns the arc sine of x,
acos(x)     Returns the arc cosine of x,
atan(x)     Returns the arc tangent of x,

csc(x)      Returns the cosecant of x,
sec(x)      Returns the secant of x,
cot(x)      Returns the cotangent of x,
acsc(x)     Returns the arc cosecant of x,
asec(x)     Returns the arc secant of x,
acot(x)     Returns the arc cotangent of x,

sinh(x)     Returns the hyperbolic sine of x,
cosh(x)     Returns the hyperbolic cosine of x,
tanh(x)     Returns the hyperbolic tangent of x,
asinh(x)    Returns the arc hyperbolic sine of x,
acosh(x)    Returns the arc hyperbolic cosine of x,
atanh(x)    Returns the arc hyperbolic tangent of x,

csch(x)     Returns the hyperbolic cosecant of x,
sech(x)     Returns the hyperbolic secant of x,
coth(x)     Returns the hyperbolic cotangent of x,
acsch(x)    Returns the arc hyperbolic cosecant of x,
asech(x)    Returns the arc hyperbolic secant of x,
acoth(x)    Returns the arc hyperbolic cotangent of x,

log(x)      Returns the logarithm base 10 of x,
ln(x)       Returns the logarithm base E of x,
antilog(x)  Returns the anti logarithm base 10 of x,
exp(x)      Returns the anti logarithm base E of x,

sqrt(x)     Returns the square root of x,
sqr(x)      Returns the square of x,
abs(x)      Returns the absolute value of x,
```

```
det(x)    Returns the determinant of the matrix x.
conj(x)   Returns the conjugate of the scalar x (or matrix x).
trans(x)  Returns the transpose of the matrix x.
canon(x)  Returns the row canonical form of the matrix x.
```

Commands:

```
list  - Lists out the memory locations (variable names, values).
clear - Clears out the memory locations.
eng   - Changes the display mode to the engineering format.
fix   - Changes the display mode to the fix point format.
polar - Changes the display mode to polar format "<r, theta>".
rect  - Changes the display mode to rectangular format "<Re, Im>".
```

Notes concerning the display mode:

1) The polar display mode will display theta in the range from $-PI/2$
   to $3*PI/2$. If the magnitude of theta is less than 1.0e-11 radians,
   theta will not be printed out (no brackets). This guards against
   small roundoff error causing real numbers to be printed in polar
   form.

2) For the rectangular mode, if the real component is greater than
   1.0e11 times the imaginary component, the imaginary component will
   not be printed out (no brackets). If the imaginary componet is
   greater than 1.0e11 times the real component, the real component
   will be zeroed out. This was done to help reduce the clutter.

To compile and link this program under OS9 (6809 version) you will
need the math enhanced library "CLIBT.L" and the defines from the file
"HEADER.AR", both of which can be found in data library 3 of the OS9
forum on CompuServe.

****

SAMPLE PROBLEMS IN LINEAR ALGEBRA:

Problem #1 (m equations in n unknowns, infinite number of solutions)

Solve for x, y, z, s, t given:

```
        x + 2y - 3z - 2s + 4t = 1
       2x + 5y - 8z -  s + 6t = 4
        x + 4y - 7z + 5s + 2t = 8
```

Enter the coefficient matrix and the constants as follows:

```
OS9:sc
canon({{ 1, 2,-3,-2, 4, 1},
       { 2, 5,-8,-1, 6, 4},
       { 1, 4,-7, 5, 2, 8}});
```

"sc" will respond with:

```
{{1,0,1,0,24,21},{0,1,-2,0,-8,-7},{0,0,0,1,2,3}}
```

Rewrite as:

```
    x + 0y +  z + 0s + 24t = 21        x = 21 -  z - 24t
    0x +  y - 2z + 0s -  8t = -7   or  y = -7 + 2z +  8t
    0x + 0y + 0z + 1s +  2t =  3       s =  3 - 2t
```

The parametric solution vector is (let z=a and t=b)

```
  (x=21-a-24b, y=-7+2a+8b, z=a, s=3-2b, t=b)
```

If a=1 and b=2 does the solution vector (x=-28,y=11,z=1,s=-1,t=2)
satisfy the set of linear equations given above?

Yes! Work it out in the following manner:

```
OS9:sc
{{ 1,  2,-3,-2,  4},
 { 2,  5,-8,-1,  6},
 { 1,  4,-7, 5,  2}} * {{-28},{11},{1},{-1},{2}};
```

"sc" will respond with:

```
{{1},{4},{8}}
```

which is the constants for the set of equations given above.

Problem #2 (m equations in n unknowns, no solution)

Solve for x, y, z, s, t given:

```
    x + 2y -  3z - 2s +  4t = 1
    2x + 5y -  8z -  s +  6t = 4
    4x +10y -16z - 2s +12t = 9
```

Enter the coefficient matrix and the constants as follows:

```
OS9:sc
canon({{ 1,  2, -3, -2,  4,  1},
       { 2,  5, -8, -1,  6,  4},
       { 4, 10,-16, -2, 12,  9}});
```

"sc" will respond with:

```
{{1,0,1,-8,8,0},{0,1,-2,3,-2,0},{0,0,0,0,0,1}}
```

Rewrite as:

```
    x + 0y +  z + 0s -  8t = 8          x = 8 - z + 8t
    0x +  y - 2z + 3s -  2t = 0    or   y = 2z - 3s + 2t
    0x + 0y + 0z + 0s +  0t = 1          0 = 1
```

There is no solution for this set of equations since 0 is
not equal to 1!

Problem #3 (n equations in n unknowns solved using the inverse

operator)

Solve for x, y, z given:

    x + 2y - z = -1
    x - y + 2z = 8
    2x - y - z = 5

Enter the coefficient matrix and the constants as follows:

```
OS9:sc
{{ 1, 2,-1},
 { 1,-1, 2},
 { 2,-1,-1}}^-1 * {{-1},{8},{5}};
```

"sc" will respond with:

{{3},{-1},{2}}

Check the answer:

```
{{ 1, 2,-1},
 { 1,-1, 2},
 { 2,-1,-1}} * disp;
```

"sc" will respond with:

{{-1},{8},{5}}

which is the constants for the set of equations given above.

**/

ooooooooooOOOOOOOOOOOooooooooooo