OS-9 USERS GROUP

"DEDICATED TO EXCELLENCE IN OS-9 COMPUTING"

M.O.T.D.

The OS-9 Users Group Newsletter

Issue #2, 1996

See

Page 15!

About the OS-9 Users Group, Inc.

General Information

The OS-9 Users Group, Inc.
6158 West 63rd St
CHICAGO IL 60638
USA

os9ug@sandv.chi.il.us

The OS-9 Users Group, Inc. (The OS-9 Users Group) is a non-profit organization, registered and incorporated in the state of Iowa, whose members share an interest in all forms of the OS-9 operating system.

The OS-9 Users Group is governed by the appropriate state and federal laws.

MOTD (Message Of The Day) is the OS-9 Users Group's official newsletter.

The OS-9 Users Group has no affiliation with Microware or any other organization.

Please remember that this is a self-help group, and as such depends on the generosity of its members for its existence. The OS-9 Users Group and this newsletter are run by volunteers -- please remember this when you deal with them.

Opinions expressed herein may not reflect those of anyone other than the author, nor do they necessarily reflect the policies of the OS-9 Users Group.

The **MOTD** will be printed a minimum of four times per year, ideally at least every three months.

Membership in the OS-9 Users Group is available to anyone with an interest in the OS-9 operating system.

Annual Dues are $25.00 in US funds for residents of the United States of America, and Canada.

Annual Dues are $30.00 in US funds for residents of other countries, to help offset the cost of Airmail delivery of this newsletter.

Dues should be made payable to The OS-9 Users Group, Inc., and sent to the above address.

Memberships run for one year from the date received.

Reprints

Reprints or back issues of the **MOTD** are available to members in good standing at the cost of $2.00 each plus $1.00 shipping (US Funds). Please send a SAE and a list of the issues you wish sent to you.

MOTD Contributions

Articles, editorials, "Letters to the Editor", personal ads, and columns are always welcome.

Submission of material does not guarantee publication. All material may be subject to review and editing by the BOD and the **MOTD** Editor, and must not be in conflict with the stated purpose of the OS-9 Users Group as defined in the Constitution and Bylaws of the Users Group. The BOD may also establish additional guidelines for acceptance before publication.

Suggestions or comments about the **MOTD** are welcome. Letters may be sent to the **MOTD** Editor.

MOTD Advertising

Commercial advertising is available in the **MOTD**. Please write for rates. All ads should be submitted as camera-ready copy. We reserve the right to limit the size and quantity of ads.

Free classified ads are available to individual members on a space-available basis.

Officers

The following individuals make up the current Board of Directors (BOD) of the OS-9 Users Group:

President	Colin McKay
Executive VP	David Graham
Director	Ken Scales
Director	Eddie Kuns
Director	VACANT
Secretary	Howard Luckey
Treasurer	Br. Jeremy

Notices

<EOF>

A few changes have been made here that will streamline (and hopefully hasten) the MOTD production cycle once and for all, resulting in more timely delivery of each issue. If it works, you should see the next issue shortly after this one, so cross your fingers!

Ed Gresick

I recently received this letter from Ed Gresick of Delmar Co., and knowing that Ed has many friends in the OS-9 community, I thought I would pass it along.

Colin McKay, President, OS-9 Users Group

Dear Colin,

It is with regret that I must tender my resignation as director-at-large of the OS-9 Users Group.

After my recent sojourn in the hospital and subsequent tests and procedures, my doctor is requiring me to reduce my activities and workload. This necessitates I completely reorganize my life. I will be phasing out many commercial activities which have placed pressure on me as well as ceasing other activities which lengthen my workday. There is a good side to this in that I hope to soon be able to spend some time on 'fun' OS-9 projects which I've had to shelve. I shall continue support of G-WINDOWS, the SYSTEM IV and SYSTEM V computers and OS-9000.

You have my best wishes and hopes for the OS-9 Users Group.

Ed Gresick

Our thanks and best wishes to Ed for a speedy recovery.

Chicago Fest

The Fifth Annual Last Chicago CoCoFest was held the weekend of 13-14 April 1996. The Sixth Annual Last Chicago CoCoFest is scheduled for the first weekend in April, 1997!

Mike Knudsen's Fest report is contained elsewhere in this issue.

Elvis (complete with sideburns and black leather jacket) was sighted by those Fest-goers who closed the bar Saturday night, somewhere around 2am. Elvis (his real name is Patrick) works nights at the Holiday Inn.

AGM and Elections

Our Annual General Meeting was held at the Chicago Fest on Saturday, 13 April. Ken Scales was elected by acclamation to the Director position. Further details about the meeting are contained in the AGM minutes in this issue.

Survey Time?

Not yet, but next issue will include a brief survey. We want to find out what you like and dislike about the Users Group and the MOTD, and what changes we can make, as well as your opinions about certain matters.

Submissions

A few articles of note this month, along with our usual material:

First is a comparison between OS-9 and OS-9000. This marks our first real OS-9000 article outside of SFTHM, and will hopefully be the first of many more.

The second is a report from the San Diego OS-9 Users Group about a demonstration they put on at the San Diego Computer Expo.

Sincere thanks to all the contributors for this issue -- it makes things much easier when we put an issue together and have material!

Submissions for the next issue, as always, are welcome. Just send it by email or snail mail to the MOTD Editor at the Users Group address.

Colin McKay
President

<EOF>

Minutes for BOD Meeting March 6, 1996

A regular meeting of the OS-9 Users Group Board of Directors (BOD) was called to order on March 6, 1996 on Delphi. Present: Colin McKay, president and chairman of the Board; David Graham, executive vice president and member of the Board; and directors-at-large Ed Gresick, Ken Scales, and Eddie Kuns.

Item 1. Approval of the minutes:
The first order of business was the approval of the minutes of the January 3, 1996 meeting. The minutes were approved as amended.

Item 2. Financial report:
Submitted as more of a bank balance rather than a financial statement with a promise of more details for 1995 at a later date.

OS-9 Users Group Account 1-01-96 to 1-31-96

Date	Activity	CR	DR	Balance
960101	Beginning Balance			$ 1,828.35
960131	Service Fee		5.89	1,822.46
960131	Ending Balance			1,822.46

(The following is unofficial pending receipt of the February 1996 statement.)

960201	Beginning Balance			1,822.46
960212	Membership Renewals	175.00		1,997.46

Respectfully submitted,
Brother Jeremy, CSJW, Treasurer

OLD BUSINESS:

Item 3. Scholarship:
The UG still needs a financial report so no action was taken.

Item 4. UG mail:
Arrangements have been made to cover the expenses of the mail service and to have the mail picked up and forwarded on a more regular basis including reports to keep the President and Board up to date.

Item 5. Chicago Fest:
Final arrangements were discussed. The UG will have a booth with extra copies of the MOTD to pass out, a time slot at the end of the Fest day on Saturday for the annual general meeting, and will probably rerun the Glenside Fest ad one more time.

Item 6. The MOTD:
The President reported that issue 1-1996 is about two thirds done and that it should be finished soon.

NEW BUSINESS:

Item 7. Nomination/Elections:
Only one position is open and that is for one of the director's slots. This will be announced in the MOTD and will be a part of the general meeting in April.

Item 8. Review of Appointed Positions:
The BOD discussed the need to review appointed positions for the purpose of injecting new blood into the leadership or for situations where some individual have taken on new responsibilities in their lives.

Item 9. Special Meeting:
The President called for an executive session of the BOD on Wednesday 3 April 1996 at 8PM EST for the purpose reviewing the various positions and for any final preparations for the Chicago Fest.

The meeting adjourned.

Howard Luckey, Secretary
Saturday April 20, 1996

These minutes were approved at the May 1, 1996 meeting of the Board of Directors.

Annual General Meeting April 13, 1996

The meeting was called to order at 4:06PM at the Holiday Inn, Elgin, Il.

Minutes of the last AGM approved as published by the members present.

Elections

The only nominee was Ken Scales, and he was declared elected by acclamation.

Reports

President, Colin McKay

Our Group has been around for about four years and considering some of the division within the community and the fact that the group and its officers are apart by wide distances we have done fairly well. The Executive meets online every two months.

One problem is a need for a better division of labour. At present the President is doing the great majority of the work such as processing requests for information and memberships (as well as editing the MOTD). It was suggested that the Secretary take on more of this work. Over the last year there have been no major problems running the group, however there have been several small ones. Various projects and initiatives by the Executive are being done to help ensure the continued success of the Group. The primary need is for people who volunteer for a position to follow through and do that job.

The number of industrial members has increased. Coverage of OS-9/68000 and OS-9000, as well as OS-9/6809 will continue to increase and members are encouraged to submit material. It is hoped that the majority of industrial members will renew their membership, and to do this, the MOTD must increase relevant material.

Treasurer, Br. Jeremy

The UG is doing okay financially. We are paying our bills and have a positive balance. There are still some administrative items to be sorted out, but these will be dealt with by the Exec. (BOD Minutes May 1, 1996.)

Librarian, Dave Kelly

The merging of the UG and OS-9 Community Networks library is in progress. Some volunteer help for this job has been found. The Internet RTSI OS-9 library site was also mentioned.

Discussion

Ken Scales, Director, mentioned getting information about an OS-9 conference being presented in Europe by EFFO the European OS-9 group. He also expressed concern about fragmentation in the OS-9 community and the need for people to help out. He added that the UG seems to be over most of the rough spots.

There was a discussion about support for OS-9/6809, and Eddie Kuns mentioned that it would not be coming from Microware which officially dropped support for OS-9/6809 several years ago. Br. Jeremy mentioned his efforts to obtain some sources from Tandy, and from that there has been only some preliminary discussion.

Boisy Pitre discussed the proposed UG scholarship. The best possible world would be for the UG to have non-tax status so that donations by corporations and individuals could count as tax deductions. However, since the group has not reached that status yet (at the Federal level), Boisy proposed going ahead with the scholarship program anyhow. The membership agreed. There was a question about the qualifications for the scholarship. This matter is still under study.

The meeting adjourned at 4:31 PM.

Howard Luckey, Secretary

Subject to approval at 1997 Annual General Meeting.

Minutes for BOD Meeting May 1, 1996

A regular meeting of the OS-9 Users Group Board of Directors (BOD) was called to order at 8:05 PM CT on May 1, 1996 on Delphi. Present: Colin McKay, president and chairman of the Board; David Graham, executive vice president and member of the Board; and directors-at-large Ken Scales and Eddie Kuns.

Item 1. Minutes: The minutes from meeting on March 6, 1996 approved as amended.

Item 2. Financial report:

OS-9 Users Group Account 3-01-96 to 3-31-96

Date	Activity	CR	DR	Balance
960301	Opening Balance			1,990.94
960325	Mail Deposit	25.00		2,015.94
960329	Service Fee		5.63	2,010.31
960331	Ending Balance			2,010.31

OS-9 Users Group Account 4-01-96 to 4-30-96 (Prelim)

Date	Activity	CR	DR	Balance
960401	Opening Balance			2,010.31
960414	115 H Luckey (postage)		15.68	1,994.63
960414	116 J. Hegberg (MOTD)		605.36	1,389.27
960414	117 C. Boll (post fund)		50.00	1,339.27
960414	118 C. Boll (PO Box)		120.00	1,219.27
960414	119 Glenside (Booth)		30.00	1,189.27
960414	Memberships	550.00		1,739.27

The service fee will be approx $8.00 +/-. I will post a corrected statement when I receive the statement from the Bank. I believe that I have mentioned that as we no longer receive the Deposit Slips with the Monthly statement, it is vital that I receive email following any deposits to the account.

Respectfully submitted,

Brother Jeremy, CSJW, Treasurer

Report approved as presented.

OLD BUSINESS:

Item 3. Election results:
The Board congratulated Ken for his reelection as Director, and Ken expressed his thanks.

Item 4. Scholarship:
Member Boisy Pitre has the paperwork and will be handling the project as a separate entity in association with the UG. No further discussion.

Item 5. Chicago Fest:
The Board expressed appreciation for the good work done by the Glenside Club for putting on the Fest.

NEW BUSINESS:

Item 6. MOTD:
The next issue of the MOTD is in the process of being completed. So far there has been little or no suggestions or feedback about articles appearing in the MOTD. The Board and editor encourage contributions from members.

Item 7. Volunteers: The Board asks that all of us be on the lookout for new volunteers to help with articles and other activities necessary to make the UG successful.

Item 8. Old software: The Board suggested that Brother Jeremy get together with Bob Swoger of the Glenside Club to combine efforts to collect old software.

The meeting adjourned at 20:35

Howard Luckey, Secretary
Sunday June 16, 1996

These minutes were approved as read at the July 3, 1996 meeting of the Board of Directors.

The OS-9 Trolley Project

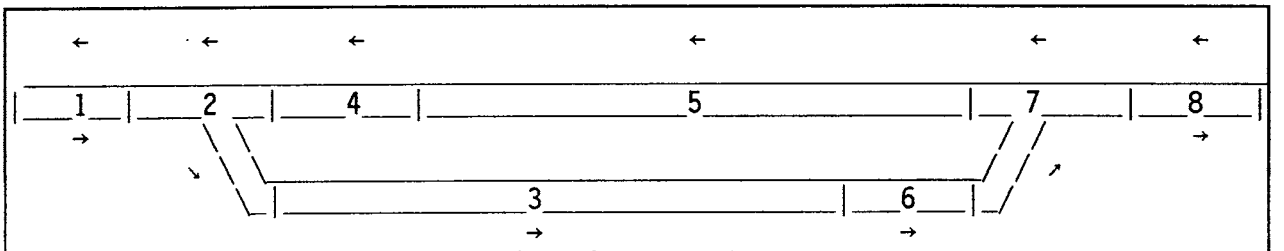
by Shawn Driscoll

Every year our club, The San Diego OS-9 Users Group, puts together a booth for display at the San Diego Computer Expo to illustrate the capabilities of OS-9. We would try our best to make OS-9 known to the general public. Our exhibit usually consisted of various computers running OS-9 of some kind. The visitors would look around a bit and then keep on walking because they saw nothing of interest to them. We looked like yet another booth with computers running some software.

For this year's Expo John Reece came up with the idea of hooking up a simple train layout to one of our CoCos (Tandy Color Computer 3) and have the computer control a couple trolley cars on it. With the system running OS-9 we could also have the CoCo running other programs at the same time and that would definitely attract some attention at our booth, we thought. Our group voted that the project was a great idea. John was to build the trolley layout and the hardware necessary to connect the CoCo to it and I was elected to write the software to control the two trolley cars.

Since my background is not in electronics I thought the trolley project was going to be rather tough. But luckily John had gotten his idea from a book called "Build Your Own Universal Computer Interface", by Dr. Bruce A. Chubb (ISBN 0-8306-3122-4, Publisher: TAB). The author explains in detail how to assemble the hardware for controlling various devices from just about any home computer. John used the schematic for the CoCo controller board to build an interface card that plugged into the Multi-Pak Interface. The card was addressed for the $FF70-FF76 area of memory on the CoCo 3. With Basic09 we could make POKEs and PEEKs to these addresses to control trolley direction, change switch-track positions, alter power voltages, and detect where the two trolley cars were on the tracks. The trolley layout for our booth exhibit is shown in the diagram on this page.

The numbers 1-8 represent the track blocks that have sensors connected to them. A trolley would travel from block 1 to block 2 and then on to block 3 and 6 then 7 and finally 8. The trolley would then travel in the opposite direction from block 8 to block 7 and then on to block 5 and 4 then 2 and finally 1, thus completing the route. This was all controlled by the CoCo which was detecting through the sensors where the trolley was and what it should do about it. John wrote up a program in DECB to control just one trolley car. This was done to test the hardware and wiring for every component. The computer was programmed to stop the trolley when it reached blocks 1 and 8 so that passengers would have time to get on and off.



Track and Sensor Layout

I was impressed with what I saw that day when I showed up to help out with the programming. Wow! John told me he was satisfied with how the trolley project was progressing and now it was up tome to write the OS-9 software to control two trolley cars.

It's one thing to have the computer control just one trolley. But to have it control two!

Yes, I did ask John a lot of questions as to how the sensors worked and how the computer controlled the track power and switch-tracks, etc. before I even began coding. I first came up with some basic safety rules for the layout:

1) Shut off the power at block 4 if a trolley is in block 1 or 2. Shut off the power at block 6 if a trolley is in block 7 or 8.

2) Make sure the switch-tracks are toggled the right way before letting a trolley leave block 1 or 8.

3) Don't power on block 4 until a trolley has left block 1 and 2. Don't power on block 6 until a trolley has left block 7 and 8.

These rules took care of any crash scenes on the layout. I used the POKE command to control parts of the layout, such as: turn on/off power on blocks 4 or 6, reverse power on blocks 1 and 2 or 7 and 8. Blocks 3, 4, 5 and 6 were hardwired so their power was always running in one direction whereas blocks 1, 2, 7, and 8 could have their power reversed in either direction. The PEEK command was used to find out which blocks had a trolley in them. Each block had a sensor connected to it. The controller would receive a value from the sensors and store it in an address the CoCo could look at. Eight bits would get sent. Each bit represented a sensor. A value of 1 would indicate that a trolley was detected and a value of 0 meant no detection. An 8-bit value of 00100001 would tell the software that a trolley was found in block 3 and that another one was found in block 8. A value of 00000000 would mean that there were no trolley cars on the layout and there was nothing to crash.

John felt I should program the trolley cars to wait at their stations (blocks 1 and 8) for a random amount of time so that they wouldn't be passing each other in the center of the layout all the time. We wanted the people visiting our booth to see a real-world environment. Sometimes the trolley cars would start on time at opposite ends of the track and pass each other in the center of the layout and sometimes a trolley would be late getting passengers boarded, and so the other trolley would have to wait (at block 4 or 6) until the trolley left its station before moving to the station to drop off passengers.

The computer had to keep track of these situations and operate accordingly. The visitors at our booth realized this right away. Some even suggested that we should use this idea to control real trolley cars. They were even more impressed when they saw the CoCo running other programs in other windows at the same time without affecting the trolley system. That's why we chose OS-9 as our programming environment. We didn't want to have the CoCo dedicated to just running one program. Our group would like to be doing other things on the CoCo while it is protecting many innocent lives on the railways.

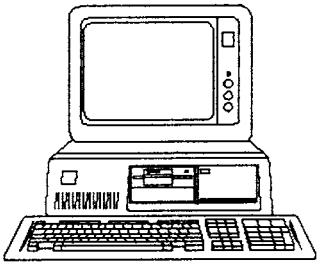
Overall, visitors left our booth with some knowledge about OS-9 who previously had never heard of it before. That was the intent of our exhibit and we feel that our project helped in that effect. <EOF>

OS-9 vs OS-9000

Historically, OS-9 has roots as far back as 1977. The environment we are now familiar with originated around 1980. Originally conceived as an operating system for the "next generation" Motorola 6809 processor, OS-9 was quickly ported when the 680X0 family of processors became available, giving capabilities never possible under the 8-bit architecture of the original chip.

In an attempt to make OS-9 more portable, it was later recoded in 'C' and released as OS-9000. This was made available for the Intel 386 and Motorola 68020 and above processors. (Editor's note: the '020 version is no longer produced.)

When the PowerPC family of processors became available, Microware made its operating system available for them as well. Instead of porting the assembly language-based OS-9 to the new RISC chips though, Microware took a more flexible approach and ported OS-9000.



This gives us OS-9, generally referring to the 68K version, and OS-9000 which can either be Intel or PPC. Actually, the current naming convention seems to be "OS-9 for 386" or "OS-9 for PPC". Don't let the name change back to OS-9 confuse you though - OS-9000 is not OS-9. While externally very similar, internal differences do exist.

NOTE: This article is based on experience with the 386 version of OS-9000 v1.4. Version 2.0 (the first version for PowerPC) has not been used by this author so there may be other differences not mentioned here.

OS-9000, being written essentially from scratch, didn't have any existing platforms to maintain compatibility with, and so enhancements were added while remaining faithful to the OS-9 philosophies of modularity, compact code, and relocatable modules. Notable changes from OS-9 include changes in RBF, process states, the shell, and SCF.

Simple changes were made like moving the bitmap location of the file structure (the RBF file manager) to improve performance, and adding extra fields to show file modification down to the second (OS-9 only tracks hours and minutes). Also, OS-9 has a limit for file attributes where only two sets exist: public and group/owner. This means any user in the same group as yourself can still read or write any file you create. The only way to prevent this is give each user in his own group. Under OS-9000, public, group, and owner have their own sets of attributes (read, write, execute) which eliminates this.

Directories have a "searchable" attribute allowing users to search a directory for files they may access, without actually being able to read or write to the directory itself (similar to Unix). These many improvements have the side effect of making OS-9000 disk formats incompatible with native OS-9.

Fortunately, Microware provides at least two methods of transferring files over - utilities that run under OS-9000 which will read OS-9 disks, or the PCF PC-DOS file manager, allowing the use of PC disk format which is compatible between any OS-9(000) system, PCs, MacIntosh, Atari ST, Amigas, and other systems that support it. (Though it is maddening to be required to shorten filenames to fit the 8 character filename limit of PC-DOS!)

Another interesting addition to OS-9000 is an extra "state" a process can be in. Modes such

as ACTIVE (currently running), SLEEPING (waiting for a time to elapse or signal to come in), and WAITING (for a child process to terminate) are familiar, but OS-9000 allows you to suspend any process temporarily, removing it from the active process queues, then resume it later on. Commands "suspend" and "activ" are provided to do this from the shell.

Speaking of the shell, OS-9000's stock shell has some enhancements of it's own, like the ability to not only pipe standard out of one program to the standard in of another ($prog1 ! prog2), but standard error as well ($prog1 !! prog2). Several other enhancements also exist which make recalling previous commands easier. Command line editing is also fancier thanks to additions to other i/o components such as the SCF sequential character file manager.

Instead of just being able to reprint or delete a previously typed line, OS-9000 allows you to do many things such as delete left word, delete right word, delete to end of line, insert/overstrike, and others. These features are automatically available when using the "readln" system call making fancy input routines a snap. All features can be disabled or toggled using a much more powerful version of the tmode utility.

Some performance enhancements have also been made. When OS-9 performs a "read", it calls the driver for each character. Thus, reading 80 bytes from a serial port invokes the driver 80 times. Under OS-9000, SCF maintains both input and output buffers so system level reads and writes are taken directly from that buffer. The driver simply stores incoming data in the SCF buffers, usually from within an interrupt service routine. This offers performance improvements as well as simplifying driver creation.

Since OS-9000 was written in 'C', it makes it much easier to talk to the kernel. With OS-9, you are expected to return certain status flags set in various registers when returning to the kernel (from a driver or interrupt service routine, for example). This requires you to either write your code entirely in assembly, or use "glue code" which takes something like "return SUCCESS" in C and then sets the appropriate carry bit of the status register and issues an RTS back to the kernel. Under OS-9000, drivers can be created without this glue code, making them easier to put together from scratch.

Speaking of the C language, many have said OS-9000 must be much larger and slower since it is written in a high level language. This is not entirely true. The OS-9000 kernel for 386 is about 50K, which, while it's nearly twice the size of the OS-9 kernel, is still very small. Even PC-DOS' COMMAND.COM is larger. Also, since OS-9000 is targeted on higher end processors such as 486/Pentium or PowerPC, optimizations at the chip level make it incredibly fast, needing something like a high-end 68040 or 060 to see similar performance from OS-9. (This is more speculation than anything, since no cross platform benchmarks have been done by this author.) Certainly a native assembly version of OS-9 for PPC would be very nice, but the extra convenience of being able to talk to the kernel directly in C (without extra glue code) is a welcome tradeoff. Besides, how many people do you know who code PowerPC RISC assembly? (The architecture was designed to be more compiler friendly than hand-assembler friendly.)

OS-9000 also can address hard drives larger than the 4-gig "limit" of OS-9. Perhaps not entirely important today, but certainly significant in the near future when these drives become commonplace (and as computers deal with more digital storage formats for audio and video).

There are many more internal differences

between OS-9 and OS-9000, yet from the C source code level most applications can be recompiled without any problems. Thanks to compatibility libraries, standard source code can be maintained without requiring significant recoding. Careful programming allows common sources to be compiled under OS-9 Level 2 (6809, pre-ANSI compiler), OS-9/68000 2.4 (K&R 3.2 compiler), as well as OS-9000/386 1.4 (Ultra-C). The compiler makes various #defines available to let your source code be "platform aware".

Some system calls are different though and do require changes (most of the time having to do with OS-9000 allowing many extra parameters or options). For instance, under OS-9 the _os9_sleep() call for Ultra-C takes one parameter - the pointer to an integer containing the amount of time to sleep for. Under OS-9000, there is an _os_sleep() that works similarly, but contains an extra pointer to a signal code which will contain a signal that woke you from the sleep, if one occurred. Many other "blocking" wait type calls also contain signal references. Another large difference is that OS-9000 allows you much greater control over what regions ("colors") of memory you allocate processes, events, modules, etc., from. Most all OS-9 calls, by comparison, are taken from general User Ram (primary system memory).

Another improvement which is great for multi-user systems, is the improved module directory structure. Under OS-9, all modules go in a flat directory meaning there can be only one copy of a module of any given name

in memory at a time. OS-9000 allows module sub-directories. This allows different modules to exist with the same name as long as they are in different module directories. There are utilities provided to make and delete module directories, and also a module PATH environment variable so different mdirs can be searched for commands. Attributes of these module subdirectories can also be changed.

Many system utilities now work differently or have more options. The format command lets you customize virtually any setting of the disk you are about to format without needing to reconfigure the drive descriptor. Xmode also seems to work quite differently under OS-9000 which may cause initial confusion to OS-9 users migrating over. Some other useful utilities, such as moded, are not available for OS-9000, but several others not available for OS-9 are, such as a command that prints out system information (from the init module).

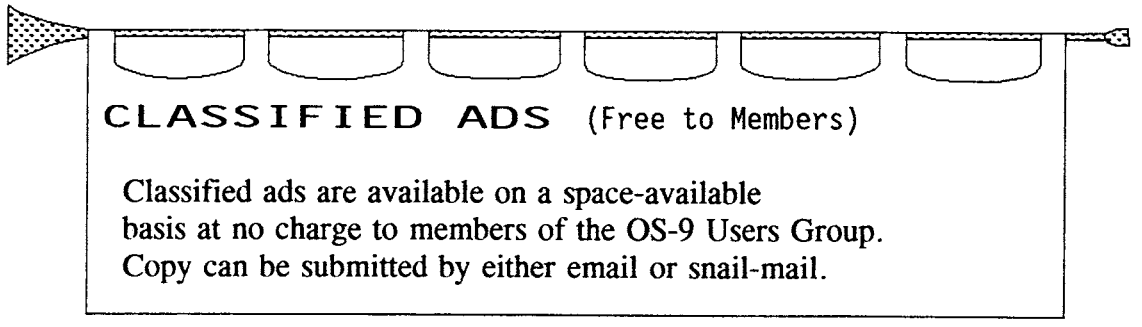
Many existing companies also provide OS-9000 support. Gespac makes tiny 386 boards with graphics that also run their 386 version of G-Windows, a GUI also available for OS-9/68K. The public domain market has also provided ports of utilities such as LHA (compression) and the popular dEd (disk editor). FTP sites such as os9archive.rtsi.com include subdirectories containing extra resources, and the OS-9 Users Group is currently compiling a library of shareware software which will be made available to group members as well.

A major upgrade for OS-9000 was also recently released. Version 2.0 includes even more enhancements as well as the introduction of the PowerPC version. It has been reported that the x86 version has been upgraded as well to be as similar as possible to the PPC. At the time this article as written, no further details of this were known (though Microware would probably make a great source to ask).

When it comes to product maturity, OS-9000 is still about half as old as it's older brother. It seems that in the past OS-9000 was more of a side product for Microware, but now much attention is being directed towards the PowerPC platform. If support or stability has been an issue with not using OS-9000 in the past, it may be time to look again. There are some exciting things going on. Also, a desktop PC makes a relatively low cost but powerful development system for code, including a great way to compile and test OS-9 programs. Let engineers use the computers they already work with, and then use a target resident compiler on the 68K machine to recompile the sources that were created and tested on the PCs. Or better yet, use Ultra-C's optional cross compiler to make the 68K executables directly under OS-9000.

In summary, OS-9000 is much more than just "OS-9 in C". It is enhanced internally while still allowing external operations to be functionally the same as OS-9. With more high end processors becoming available, and memory becoming cheaper, the days of the small assembly 29K kernels may no longer be necessary. So, watch the MOTD for future news on where OS-9000 is headed. The OS-9 Users Group is committed to bringing full OS-9000 support into the organization as well.

<EOF>



CLASSIFIED ADS (Free to Members)

Classified ads are available on a space-available basis at no charge to members of the OS-9 Users Group. Copy can be submitted by either email or snail-mail.

Chicago Fest Report

by Mike Knudsen

[This report is based on Mike Knudsen's Chicago Fest report posted on Internet for Bob Devries in Australia. It is used with Mike's permission.]

The 5th Annual Last Chicago CocoFest was a lot of fun, and pretty successful in terms of vendor and customer count. Not quite as big as last year, but still plenty of enthusiastic attendees. Next year's Fest will be April 4-6, if the glaciers have melted by then.

The "MusicMen"

Mike Carey and Brian Schubring (of Glenside Coco Club) formed a group along with myself called "MusicMen" and provided a humongous rolling rack of MIDI and sound gear with which we gave two seminars and serenaded the whole Fest. Booming bass and whistling piccolos, wall to wall.

The seminars re-awakened enough interest in music and MIDI that I sold several new copies of UltiMusE. Most important, Brian was selling (very cheap) brand new clones of the Coco MIDI Pak, along with Coco MIDI Pro RSDOS sequencer. That helped get the musical interest back up.

The Canucks

The Canadian Contingent arrived Saturday morning, contrary to original hopes of a Friday night arrival. Alan DeKok would've been there except for a catch-22 -- he had no passport to prove Canadian citizenship, and his birth certificate was at the passport office trying to get him one! There is indeed a new NitrOS-9, v1.22n.

Brian's Coco3 was running NitrOS-9 under UltiMusE, and it was at least two, maybe three or more times as fast! At least initial startup was that much quicker. For now, UltiMuse is available directly from me. A new disk manual is forthcoming.

The OSKers

Bill Wittman was there with some WCP-306 boxes, where most of the OSKers hung around. All sorts of our "graduates" to Microware were on hand. Carl Kreider is working to get MGR sped up enough to run well on the WCP-306.

Several vendors of generic computer supplies, plus resellers of used Coco hard and software. Lots of Cocos went unsold. HawkSoft had a good presence with OSK and OS9 software.

After Hours

Saturday night we looked around the bar, and everyone seemed to be an OS9er; guess that's a sign of the times. People I remember talking with include Boisy Pitre, James Jones, Kent Myers, Carl Kreider, Joel Hegberg, Paul Jerkatis, Bill Wittman, Colin McKay, Ken Scales, Chris & Nancy Hawks, Al Dages, RC Smith, and my apologies to anyone I forgot.

Next year we'll try to get a keynote speaker, and make sure he shows up.

Wheelin' & Dealin'

I ran the no-min-bid auction both days. As usual, good stuff went cheap and Glenside made some money to put on next year's Fest.

I wish folks had tried harder to buy up some of the "orphanware."

There were terrific deals in used hard and floppy drives and other gear. A Disto no-halt floppy controller went for a piddly $5 at the auction. If you need something, it would be worth the drive or train fare to come next year.

<EOF>

Wittman Computer Products

HARDWARE * SOFTWARE * CONSULTING

We offer Gift Certificates

Software

Game Pack – Sea Battle, Minefield, KnightsBridge, Othello & Yahtzee	$30 MM/1 $25 CoCo
Variations of Solitaire – Pyramid, Klondike, Spider, Poker & Canfield	$30 MM/1 $25 CoCo
Gold Runner 2000 – KWindows game	$35 MM/1
KChess – GNU Chess with KWindows GUI	$25 MM/1
GNU Chess Sources Disk	$ 5 OS-9/68000
CirCad – Circuitry CAD program with EPS output	$80 MM/1
DeskTamer v2.0 – a premier personal information manager	$65 MM/1
LaTerm / LaDial – GUI terminal package with script auto-dialing	$65 MM/1
LaPhone / LaFax – GUI voice, fax & address database manager	$65 MM/1
LaFax – stand-alone fax sending utility	$40 OS-9/68000
InfoXpress – automates the message gathering process to save the user on-line time & money	$75 OS-9/68000 $55 OS-9/6809
X-10 Home Controller – program and monitor your X-10 system	$40 MM/1
M6809 – utility to emulate the OS-9 Level II operating system	$65 OS-9/68000
TVP Point-Of-Sale - fully integrated multi-user accounting, inventory, ordering, ↪ and check-out (cash drawer) POS software all in one package	$CALL OS-9/68000

Hardware

WCP306 single board computer with 16 bit PCAT I/O Bus — 3 slot board $400
↪ MC68306 CPU at 16.67 MHz, code compatible with 68000. — 5 slot board $425
↪ Running OS-9 V3.0 including many utilities.

"The WCP306 is a good product to use. There are certainly opportunities in using it in the manufacturing environment."
quote by : "Sam T.C. Kwok of Motorola Semiconductors H.K. Ltd."

Point-Of-Sale Equipment
Hard Drives & Floppy Disk Drives
Cases & Power Supplies
Data / Fax Modems
Keyboards, Memory & Mice

For a more detailed description, write or call us for a catalog.

We have moved to a larger location!

Wittman Computer Products
William L. Wittman, Jr.
39 South Lake Avenue
Bergen, NY 14416
(716) 494–1506

Reprinted from Sculptor News, Vol 5, No 4, with permission of The KD Consulting Group, Inc.

Ryegate Show Services

Speed and power. A century before computer ad writers used them, those two words epitomized the regal sport of equestrian show jumping. For Ryegate Show Services, speed and power apply two ways -- horse and rider exploding over a lofty fence, and Sculptor 4GL+SQL quietly running the show.

Located in east central Pennsylvania, Ryegate is the leading horse show services provider in the U.S. With its database of more than 15,000 horses and 20,000 exhibitors, however, its influence reaches around the world. Using Sculptor programming, Ryegate verifies show results that help determine which horses and riders qualify for the United States Equestrian Team (USET). The USET represents the United States in international competitions.

Ryegate's original application, designed to process entries, payments, and show results, was written in 1979 using the RMS database on OS-9. At that time, RMS was the only database available for multi-user applications on a small business machine. The initial test involved 1400 entries at one show and duplicated the way entries had been handled manually. Entering this initial data required four people working six days, about 50 entries per day per person.

Sculptor features expand, simplify processing

When support for RMS disappeared in the mid-1980s, Ryegate decided to rewrite in Sculptor. According to Randy Morgan of Ryegate, "It was soon discovered that Sculptor had a much wider range of features, and allowed us to expand and simplify the processing." Rewrites in Sculptor created master files, and now one operator can enter up to 300 entries per day. Originally designed for the specific needs of one show, the system has been modified to take advantage of its inherent capabilities and streamline the detailed data every show generates.

Ryegate continued to expand its Sculptor applications as more shows switched from manual to computer systems, and as the sport has grown. Lloyd Longenecker, founder of Ryegate Show Services, mentions Sculptor's ease of programming as a factor in the decisions both to choose Sculptor and to stay with it. Specifically, he values the ability to link separate databases, and Sculptor's easy generation of custom screen and report forms. Because Sculptor is neither unnecessarily complex nor overly sensitive, the programs run reliably. Mr. Longenecker cites accessible technical support as a plus for Sculptor, but as Mr. Morgan said, "The programs work, so we don't have to call very often."

Easy portability opens vertical market

Today, Ryegate writes and tests on a GMX box running OS-9 68030. The system supports multiple users as well as remote printers and information terminals. Customers who have purchased the system use DOS, either single-user or networked, and Sun Unix, a versatility guaranteed by Sculptor's easy portability.

Ryegate currently processes six major shows from its offices in Annville, going on-site for one week to two months depending on the

show. Each show is a separate entity, with its own suite of programs and data which are linked to master files and previous years' data from the same show. Each potential exhibitor submits an entry including name, address, membership information, and tax ID of the owner, trainer, and rider. Information about the horse also is entered, including name, recording information, description, and classes entered. This data is then compared to the specific requirements for various classes. If more information is needed, custom-generated letters and invoices go out to each exhibitor.

For certain shows, horses must qualify by earning points over the previous year. In these cases, a list of entries by class and point totals is generated and sent to the American Horse Shows Association (AHSA) for verification. After this governing body verifies the entries, custom letters go out to exhibitors indicating the status of each entry. Entry lists are exported to PageMaker, from which Ryegate prints catalogs, geographical lists of exhibitors, and press releases. Maintaining a clean, accurate database is crucial, as rankings can be separated by just fractions of seconds, and entrants must meet annually updated stringent entry rules.

Grand -- and lucrative -- traditions

Rooted in grand traditions, show jumping is a sport of grace, speed, precision, and money. The horses, often corporate-owned or -sponsored, are divided into Hunter and Jumper classes. Course obstacles, called fences, can be stone or wooden gates, hedges, water jumps, or combinations of these. Scores start at zero, and the fewer points accumulated, the higher the standing. Specified points are added for rails down or other missteps, and the course clock counts fractions of seconds.

Jumper judging is precisely objective. Fences average 4' to 5'6", and horses must demonstrate skill over an average of twelve jumps per course. If more than one horse jumps "clean," speed determines the winner. Puissance Jumper classes, run at only a few U.S. shows, are judged on height alone. In the first round, puissance jumpers clear two "get ready" fences, and then go for the wall -- a fence that averages 5'9" to 7'. The indoor puissance record is a towering 7'10". At that height, power is everything.

Sometimes grace counts, too

Power is less an issue for hunters, whose highest jump is about 4'. Judges assess the horses subjectively on grace and form as well as soundness. In addition to the tests of skill, hunters must display walk, trot, and canter. Judges look for flow and motion of the horse and rider combination -- what Annette Longenecker, daughter of Ryegate founder Lloyd Longenecker, describes as "how the horse floats over the ground."

Whether concentrating on precision, power, or grace, owners can choose from 200 to 300 classes in which to enter their horses. The season begins December 1 with winter circuits in Florida and Southern California. Those who place high enough are invited to the renowned, 100-year-old Devon Show, held near Philadelphia in the late spring. After Devon, summer circuits qualify horses for the prestigious indoor circuit.

Meticulous record-keeping guarantees accuracy

When a show begins, Ryegate verifies each entry and marks pertinent information in the database. Stabling lists are generated showing stall locations based on the layout of the individual show grounds. Entry lists per class are provided to announcers and reporters, and jumping orders can be generated for specific classes. Ryegate staff enter the results of each class, and prize money is credited to the account of each horse. Payments are posted as received. After the show, prize checks are written and mailed, and final results and standings are submitted to various sanctioning organizations.

High jumps, high stakes

To count as qualifiers for Nations Cup and other international events in the U.S., Canada, and Europe, shows must be AHSA-sanctioned, offer at least $25,000 in prize money, and enjoy a certain level of prestige. Prestige generally ups the stakes; some shows offer $50,000, $100,000, or even a quarter-million dollars in prize money and other winnings. Corporate sponsorship helps maintain these lucrative prizes, high visibility, and an intense level of competition. On the Volvo World Cup circuit, for example, winners receive a Volvo automobile as well as prize money. And for some events, status outweighs prizes altogether; the Olympic Games this year will draw top riders and horses for gold medals.

From its in-house and on-site show system, Ryegate has developed several subsets to control various ranking activities. These include the ranking system used by the USET to select horses and riders to represent the U.S. in international competition. The American Hunter Jumper Foundation depends on Ryegate data as well to determine year-end awards and qualifiers for their annual show. In addition, Ryegate systems calculate membership and winnings for the National Children's League, National Adult League, the Washington International Horse Show equitation classes, the National Children's and Adults' Medal equitation classes, and the Show Jumping Hall of Fame Jumper Classic Series. Each of these organizations relies on Ryegate to qualify participants for annual show jumping finals.

Sculptor: Power behind the pageantry

Show jumping combines tradition, speed, power, and grace like no other sport. With stakes as high as the jumps, information has to be dependable. Ryegate Show Services and Sculptor work behind the pageantry to provide the reliable, meticulous accounting that gives a show meaning.

Ryegate Show Services 405 South Spruce Street Annville, Pennsylvania 17003 (717) 867-5643

Randy Morgan is Head of Programming and Data Processing for Ryegate. Randy was graduated in 1990 from Lebanon Valley College with a degree in programming, He first joined Ryegate in 1988 as a college intern. Annette Longenecker, recently graduated from West Chester University, is pursuing a career in elementary education. She has been involved in the hunter and jumper horse show industry her entire life, showing both jumpers and hunters from early childhood through college.

<EOF>

ROFFIX Here is a little snippet from Bob van der Poel Software you might find helpful... it works to strip out the underlining and bolding done in *roff generated files. There are some things someone with time might want to play with:

- the program just converts LFs to CRs. Properly, it should eliminate LFs followed by or preceeded by a CR.

- BS/Char sequences which are split over two reads aren't handled properly.

```
/* roffix.c

        A very quick program to delete $08 xx sequences in files generated by
nroff, etc. and as a bonus, converts LFs to CRs. This was written to post-process
files formatted on a remote system by nroff for easy reading with my editor.

        Based on a similar program, crlf.c, which does LF to CR conversions which
appeared in the MOTD Summer/92 by Bob Devries.

        Bob van der Poel     95/07/02
*/

#include <stdio.h>
#include <modes.h>

main(argc,argv)
int argc;
char **argv;
{
        int i;
        int inpath,outpath;

        /* no filename, show help */

        if(argc==1) showhelp();

        /* if arg starts with -, help */

        for(i=1; i<argc;i++)
        {
                if(argv[i][0]=='-') showhelp();
        }

        for(i=1;i<argc;i++)
        {
                if((inpath=open(argv[i],S_IREAD))==-1 ||
                  (outpath=open(argv[i],S_IWRITE))==-1)
                        fprintf(stderr,"roffix: can't open %s\n",argv[i]);
                else dofix(inpath,outpath);

                if(inpath!=-1) close(inpath);
                if(outpath!=-1) close(outpath);
        }
}

char hunk[10*1024];             /* 10k buffer */

dofix(inpath, outpath)
register int inpath, outpath;
{
        register char *p1, *p2, *end;
        register int count;

        while((count=read(inpath, hunk, sizeof(hunk))) > 0)
        {
                for(p1=p2=hunk, end=&hunk[count]; p1<end; )
                {
                        if(*p1=='\l')
                        {
                                *p2++='\n';
                                p1++;
                        }
                        else if(*p1=='\b')
                        {
                                if(p2>hunk) p2--;
                                p1++;
                                count-=2;
                        }
                        else *p2++=*p1++;
                }
                write(outpath, hunk, count);
        }
        _ss_size(outpath, _gs_pos(outpath));
}

showhelp()
{
     fprintf(stderr,"roffix - Strips LFs and BackspaceXX from RBF files\n");
     fprintf(stderr,"  Usage: roffix <path> [<path>]\n");
     exit(0);
}
```

<EOF>

Part 6 of 14

Basic09 has always been a much misunderstood language. I suppose that is because most Basic09 users come from Disk Basic, and the change is quite marked, especially the lack of line numbers. It does, however, have a few quite powerful looping commands which can be used to make up for the lack(?). These looping commands are: FOR / NEXT / STEP...aha, I see you recognize that one...IF / THEN / ELSE / ENDIF ...you almost know that one too... LOOP / ENDLOOP ...a new one.. REPEAT / UNTIL ..and.. WHILE / DO / ENDWHILE ..and..EXITIF / ENDEXIT.

The FOR/NEXT loop is much the same as the RSBASIC one. The loop variable may be either an INTEGER or REAL, and the STEP value may be positive or negative.

Now comes the IF/THEN loop. The new word here is the ENDIF statement. Because Basic09 does not (usually) use line numbers, there must be a way to tell the interpreter where the IF/THEN loop finishes. This is where it is different from RSBASIC. With Basic09, the IF/THEN loop can be, and usually is, split over multiple lines. Here's a sample of programme to show you.

```
IF a$="2" THEN
   RUN gfx2("bell")
   print "You pressed '2'."
ELSE
   RUN gfx2("bell")
   print "You pressed something else."
ENDIF
```

So you can see that the ENDIF is required, so that Basic09 can see where to start again after the THEN part has been done. In RSBASIC, this would all have to be on one line! So you can see how much more powerful it is!

The next loop command is called just that: LOOP and ENDLOOP. This forms an unconditional branch back from the ENDLOOP to the LOOP. The only way out of this loop is to use the EXITIF/ENDEXIT command. This allows you to exit any loop prematurely, and is particularly useful here. Here's how it loops... er, looks:

```
LOOP
   INPUT ">",a$
   EXITIF a$="exit" THEN
   ENDEXIT
   PRINT a$
 ENDLOOP
```

The reason the LOOP/ENDLOOP is used here is because the other looping commands all go through the loop at least once before they terminate, or require that the test condition is known beforehand. As this is not the case here, I have used LOOP/ENDLOOP.

If you already have the starting value of the 'condition' variable, then the WHILE/DO/ENDWHILE command is useful:

```
DIM a,b:STRING

WHILE a="" DO
   RUN inkey(a)
   print a;
   b=b+a
ENDWHILE
```

Here the loop may not be used at all, depending on the value of the string variable 'a'. If 'a' is a NULL string (no characters), then the loop contents are executed. Otherwise, they are not, and the programme starts again after the ENDWHILE command. If you must do the contents of a loop at least once, use this method:

```
REPEAT
   a=PEEK($FF40)
UNTIL LAND(a,8)=0
```

This way, the value of a is the result of a PEEK command, and is not known until at

least once around the loop. If the value makes the UNTIL argument equal zero, then the loop is only executed once, otherwise it goes round again.

Loops may be nested, of course, and different loops may be used within each other to make the code more like spaghetti. At least that is what it might look like to a beginner. At least Basic09, when asked to list the programme, inserts indents for the various levels of nesting so that you can see at a glance what is going on. And of course, if you forget an ENDIF, or other end-of-loop statement, it tells you in no uncertain terms that you have erred! Error 69 - Unmatched control structure, means you have left out an ENDIF or ENDLOOP or something.

Of course, GOTO (yuk) can be used if you have used a line number to go to. I guess it is also a form of loop, and can be used within the loop structure to by-pass bits of code. Line numbers are also used if you want to use GOSUBs. Basic09 does not use text labels, but uses line numbers instead. Other forms of loop or control structures are ON...GOTO, ON...GOSUB, and ON ERROR. These all need line numbers to refer to. The ON ERROR is particularly useful in trapping user errors, but don't use them for trapping programme errors (bugs), or you may never see them! Always REM them out until you have thoroughly debugged your code, and only unREM them to trap unwary users of your programme.

Well, I hope that gave you something to go on with. Until next time...

Regards, Bob

Part 7 of 14
Tokenised Files

Here is a programme which will read a RSDos BASIC tokenised file and print it out as ASCII text. You must of course have some way of moving the file from a RSDos disk to OS9, such as the 'RSDos' utility which is available from pretty much any good OS-9 BBS or Internet file site.

This programme expects the filename on the command line, and prints the ASCII to STDOUT, so you can redirect it to a file, or the printer. If you don't give it a name, it prints a usage message. This is done by setting up an ON ERROR GOTO line first up, and then using the PARAM line, which reads in the variables from the command line. If none exist, an error 56 would occur, but as it is trapped, the programme quits cleanly.

After opening the file, the programme jumps over the first five bytes, and reads the next two to get the first line number. It then enters a loop to keep reading characters until a zero value is read, which is the end of line character, and prints a carriage return. After skipping two bytes (the address pointer to the next line), it reads two bytes for the next line number. This goes on until three zero bytes are read in a row, which means the end of the programme has been reached. The file is then closed and the programme ends.

The source is on the next page...

```
PROCEDURE translate
 0000        ON ERROR GOTO 1000
 0006        PARAM file:STRING[100]
 0012        DIM token:BYTE
 0019        DIM command:STRING[20]
 0025        DIM path:INTEGER
 002C        DIM x:INTEGER
 0033        DIM flag,ernum:INTEGER
 003E        DIM line:INTEGER

 0045        flag=0
 004C        OPEN #path,file:READ
 0058        FOR x=1 TO 5
 0068          GET #path,token
 0072        NEXT x
 007D        GET #path,token
 0087        line=token*256
 0093        GET #path,token
 009D        line=line+token
 00A9        PRINT line; " ";

 00B3        WHILE NOT(EOF(#path)) DO
 00BE          GET #path,token
 00C8          IF flag=1 THEN
 00D4            line=token*256
 00E0            GET #path,token
 00EA            line=line+token
 00F6            IF line=0 THEN
 0102              END
 0104            ENDIF
 0106            GET #path,token
 0110            line=token*256
 011C            GET #path,token
 0126            line=line+token
 0132            PRINT line; " ";
 013C            flag=0
 0143            GET #path,token
 014D          ENDIF
 014F          IF token<>0 THEN
 015B            flag=0
 0162            IF token>127 THEN
 016E              token=token-127
 0179              IF token=128 THEN
 0185                RESTORE 1020
 018A                GET #path,token
 0194                token=token-127
 019F              ELSE
 01A3                RESTORE 1010
 01A8              ENDIF
 01AA              FOR x=1 TO token
 01BB                READ command
 01C0              NEXT x
 01CB              PRINT command;
 01D1            ELSE
 01D5              PRINT CHR$(token);
 01DC            ENDIF
 01DE          ELSE
 01E2            PRINT
 01E4            flag=1
 01EB          ENDIF
 01ED        ENDWHILE

 01F1        CLOSE #path
 01F7        END
 01F9 1000   ernum=ERR
 0202        IF ernum=56 THEN
 020E          PRINT "usage:translate <filename>"
 022C          PRINT "       converts RSDos basic tokenized programme to ASCII"
 0267          END
 0269        ENDIF
 026B        PRINT "OS9 error "; ernum; " has occurred!"
 028D        END

 028F 1010   DATA "FOR","GO","REM","'","ELSE","IF","DATA","PRINT","ON","INPUT"
 02D3        DATA "END","NEXT","DIM","READ","RUN","RESTORE","RETURN","STOP"
 0311        DATA "POKE","CONT","LIST","CLEAR","NEW","CLOAD","CSAVE","OPEN"
 034F        DATA "CLOSE","LLIST","SET","RESET","CLS","MOTOR","SOUND","AUDIO"
 038F        DATA "EXEC","SKIPF","TAB(","TO","SUB","THEN","NOT","STEP","OFF"
 03CE        DATA "+","-","*","/","^","AND","OR",">","=","<"

 03FD        (* commands for ECB follow *)
 041A        DATA "DEL","EDIT","TRON","TROFF","DEF","LET","LINE","PCLS"
 0454        DATA "PSET","PRESET","SCREEN","PCLEAR","COLOR","CIRCLE"
 048B        DATA "PAINT","GET","PUT","DRAW","PCOPY","PMODE","PLAY","DLOAD"
 04C9        DATA "RENUM","FN","USING"

 04E2        (* commands for DECB *)
 04F9        DATA "DIR","DRIVE","FIELD","FILES","KILL","LOAD","LSET","MERGE"
 0538        DATA "RENAME","RSET","SAVE","WRITE","VERIFY","UNLOAD"
 056D        DATA "DSKINI","BACKUP","COPY","DSKI$","DSKO$","DOS"

 05A0        (* commands for SECB (CC3) *)
 05BD        DATA "WIDTH","PALETTE","HSCREEN","LPOKE","HCLS","HCOLOR"
 05F5        DATA "HPAINT","HCIRCLE","HLINE","HGET","HPUT","HBUFF"
 062A        DATA "HPRINT","ERR","BRK","LOCATE","HSTAT","HSET","HRESET"
 0664        DATA "HDRAW","CMP","RGB","ATTR"

 0683        (* function commands CB *)
 069D 1020   DATA "SGN","INT","ABS","USR","RND","SIN","PEEK","LEN","STR$"
 06DC        DATA "VAL","ASC","CHR$","EOF","JOYSTK","LEFT$","RIGHT$","MID$"
 071A        DATA "POINT","INKEY$","MEM"

 0735        (* function command for ECB *)
 0753        DATA "ATN","COS","TAN","EXP","FIX","LOG","POS","SQR","HEX$"
 078E        DATA "VARPTR","INSTR","TIMER","PPOINT","STRING$"

 07BE        (* function commands for DECB *)
 07DE        DATA "CVN","FREE","LOC","LOF","MKN$","AS"

 0807        (* function commands for SECB (CC3) *)
 082D        DATA "" \\(* empty one for CC3 bug *)

 084F        DATA "LPEEK","BUTTON","HPOINT","ERNO","ERLIN"
```

<EOF>

ATTENTION!

WHO? YOU ARE INVITED!

WHAT? COLOR COMPUTER FEST

WHEN? August 1 & 2, 1997 (Sat. 10am-5pm; Sun. 10am-3:30pm)

WHERE? EMBERS INN 1700 Harrisburg Pike CARLISLE PA

Exit 16 off of the PA Turnpike I-76, turn to Harrisburg,
go 1.3 miles, it's on the right
OR I-81 Exit 17, turn left, go 1/10th mile, on the right!

Overnight room rate: $60 - Be sure to ask for the "FEST" rate! Call (717)243-1717 for reservations! A limited supply of rooms is blocked out for the fest, so reserve early. These rooms will be released for regular reservations on July 1, 1997 and will NOT be available at the fest rate!

ADMISSION: $5.00 per person per day OR $7.00 for both days (paid in advance). Children accompanied by a responsible adult are FREE!

For further information, general or exhibitor, contact:

Ron Bull ron.bull@paonline.com
115 Ann Street ronbull@aol.com
DUNCANNON PA 17020 (717)834-4314

Programming with Solder

Many people have been replacing the 6809 CPU on their Color Computer with the Hitachi 6309 lately. In addition to the lower power requirements (and reduced heat generation) of the CMOS-based 6309, some newer software such as John Kowalski's .mod music file player for Disk Basic, and Northern Xposure's NitrOS-9, require the extra instructions in the 6309.

For those of you contemplating this change, the obvious route involves desoldering the 6809 and replacing it with a socket. A slightly simpler method of installing the 6309 though, is to install it on top of the existing 6809, and disable the 6809, with either a Piggyback installation, or with a Switchable Piggyback installation:

Piggyback Installation

1. Take a 40 pin IC socket and cut off pins 5, 6, 33, 36, and 38. Bend pin 39 inwards.

2. Solder a length of 30 gauge wire across the bottom of the socket between pins 1 and 39.

3. Cut pin 39 of the 6809 cpu on the motherboard (IC1).

4. Solder a piece of 30 gauge wire between pins 7 (+5vdc) and 39 (Tri-State) of the 6809.

5. Place the socket on top of the 6809 (IC1). Make sure the socket pins make firm contact with the corresponding 6809 pins.

6. Join each pair of corresponding pins with solder. (Except 5, 6, 33, 36, 38 and 39.)

7. Plug in the 6309.

Switchable Piggyback Installation

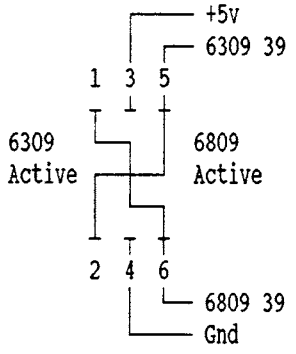
Adding a DPDT switch makes it possible to switch between the 6809 and the 6309. This can be used to confirm the operation of software with systems not using a 6309.

1. Take a 40 pin IC socket and cut off pins 5, 6, 33, 36, and 38. Bend pin 39 outwards, so that it can later be connected to the switch.

2. Cut pin 39 of the 6809 cpu on the motherboard (IC1). Bend the pin upwards, and solder a piece of wire to it. Connect the wire now, as the pin is easier to access now.

3. Place the socket on top of the 6809 (IC1). Make sure the socket pins make firm contact with the corresponding 6809 pins.

4. Join each pair of corresponding pins with solder. (Except 5, 6, 33, 36, 38 and 39.)



DPDT Switch Wiring

5. Connect Pins 1 and 6 of a Double-Pole, Double-Throw (DPDT) switch to pin 39 of the 6809 on the motherboard as shown above.

6. Connect Pins 2 and 5 of the switch to pin 39 of the socket.

7. Connect Pin 3 of the switch to +5 volts (6809 Pin 7), and Pin 4 of the switch to Ground (6809 Pin 1).

8. Finally, plug in the 6309.

<EOF>

Ever wonder why the Color Computer keyboard layout differs from other 'standard' keyboards? The following information is taken from the Jargon File, a listing of computer terms and anecdotes available on Internet.

bit-paired keyboard: n. obs. (alt. 'bit-shift keyboard') A non-standard keyboard layout that seems to have originated with the Teletype ASR-33 and remained common for several years on early computer equipment. The ASR-33 was a mechanical device (see {EOU}), so the only way to generate the character codes from keystrokes was by some physical linkage. The design of the ASR-33 assigned each character key a basic pattern that could be modified by flipping bits if the SHIFT or the CTRL key was pressed. In order to avoid making the thing more of a Rube Goldberg kluge than it already was, the design had to group characters that shared the same basic bit pattern on one key.

high bits	low bits									
	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001
0010		!	"	#	$	%	&	'	(	)
0011	0	1	2	3	4	5	6	7	8	9

Looking at the ASCII chart above, we can see why the characters !"#$%&'() appear where they do on a Teletype (thankfully, they didn't use shift-0 for space). This was *not* the weirdest variant of the {QWERTY} layout widely seen, by the way; that prize should probably go to one of several (differing) arrangements on IBM's even clunkier 026 and 029 card punches.

When electronic terminals became popular, in the early 1970s, there was no agreement in the industry over how the keyboards should be laid out. Some vendors opted to emulate the Teletype keyboard, while others used the flexibility of electronic circuitry to make their product look like an office typewriter. These alternatives became known as 'bit-paired' and 'typewriter-paired' keyboards. To a hacker, the bit-paired keyboard seemed far more logical --- and because most hackers in those days had never learned to touch-type, there was little pressure from the pioneering users to adapt keyboards to the typewriter standard.

The doom of the bit-paired keyboard was the large-scale introduction of the computer terminal into the normal office environment, where out-and-out technophobes were expected to use the equipment. The 'typewriter-paired' standard became universal, 'bit-paired' hardware was quickly junked or relegated to dusty corners, and both terms passed into disuse.

<EOF>

For superior OS-9 performance, the

SYSTEM V

Provides a 68020 running at 25 MHz, up to 128 MBytes of 0 wait-state memory, SCSI and IDE interfaces, 4 serial and 2 parallel ports, 5 16-bit and 2 8-bit ISA slots and much more. The **SYSTEM V** builds on the design concepts proven in the **SYSTEM IV** providing maximum flexibility and inexpensive expandability. Optionally available at 33 MHz.

AN OS-9 FIRST - the MICROPROCESSOR is mounted on a daughter board which plugs onto the motherboard. This will permit inexpensive upgrades in the future when even greater performance is required.

G-WINDOWS benchmark performance index of a **SYSTEM V** running at 25 MHZ with a standard VGA board is 0.15 seconds faster than a 68030 running at 30 MHz with an ACRTC video board (85.90 seconds vs 86.05 seconds).

For less demanding requirements, the

SYSTEM IV

The perfect, low cost, high-quality and high performance OS-9 computer serving customers world-wide. Designed for and accepted by industry. Ideal low-cost work-station, development platform or just plain fun machine. Powerful, flexible and expandable inexpensively. Uses a 68000 microprocessor running at 16 MHz.

G-WINDOWS - a PROVEN WINNER

FOR OS-9

Available for the **SYSTEM IV** and **SYSTEM V** computers, the PT68K4 board from Peripheral Technology and the CD68X20 board from Computer Design Services. Resolutions from 640 x 480 x 256 to 1024 x 768 x 256. Support for multiple VGA cards running different processes, different portions of the same process or both.

G-WINDOWS is now AVAILABLE for OS-9000 from *delmar co*

Support for generic VGA boards and SUPER VGA boards including ET4000 (Tseng Labs), OTI067 (Oak), CT452 and CT453 (Chips and Technology), GENOA, WD90C11 (Paradise) and S3 (S3 Inc.) for modes from 640 x 480 to 1280 x 1024 depending on board.

Distributor of MICROWARE SYSTEMS CORPORATION Software

This ad was prepared and printed using QuickEd under OS-9.

delmar co

PO Box 78 - 727 North Broad Street - Middletown, DE 19709

302-378-2555 FAX 302-378-2556