COMMODORE MAILINK

NEWSLETTER OF MEETING 64/128 USERS THROUGH THE MAIL

5-1-06

MEETING NEWS
May, 2006

Spring is in the air, May is here, and I'm in love. Yes, it's true. A few years ago I began corresponding with a nice man who is a member of this group and guess what happened? He introduced me to Pocket Planner! Pocket Planner 2 to be exact. It didn't take me long to create a number of different accounts under the broad headings of "INCOME" and "EXPENSE". And the varied types of calculations that can be performed on your numbers is mind-boggling. Another feature which works well for me is the printing. I use a Star NX1000 with Pocket Planner2 and for an experiment selected the Epson print driver and it works like a charm. Well, now you know--Pocket Planner 2 has me hooked, and the Commodore still rulez.

New people! We send a warm welcome to new member, **Yul Haasmann**, whose BIO is inside. And a new talent, **Jane M. Jones,** of Australia will begin her graphics-based column this year in these pages, and we are delighted to welcome Jane to our community. For those not "in the know", Jane authored a column called "Genie ReAppears" for many years in the Australian Commodore publication, "Melbourne Commodore Computer Club News" (MCCC News). Welcome aboard, new member Yul, and new writer, Jane!

The Slang Tutorial by **Stephen Judd** has now run its course, with the culmination through lesson #16 in this issue. It was intendeded for Slang to be on March MaiLinkOnDisk but is on May's disk instead. Plus, at press time, we discovered Judd's latest Slang v1.2 is now available. Visit http://www.ffd2.com/fridge/slang/ for more info.

"A Few Good People" is what we are looking for. Yep, we would be thrilled if a couple of members would volunteer to edit an issue of MaiLink. It isn't that difficult, and your volunteering a little will help inject some variety/variation into our editorial base. You can even volunteer now to edit an issue next year. You know where to find me.

One last item to mention is **Richard Savoy's** complimentary "TROUBLESHOOTING GUIDE" for the C64, which is enclosed with this issue. OK, now is your chance to relax a bit, lean back in that recliner and start reading about our favorite computer and what it can do.

--President, Linda Tanner

BUSINESS OFFICERS:(see addresses in BIO's)
PRESIDENT : Linda Tanner, tannerlj@yahoo.com
Handles group business, compliments, complaints, threats, etc.
TREASURER: Emil Volcheck, Jr. emilv@mercury.ccil.org
receives dues, donations, balances bank account, disburses monies;
PUBLISHER/MAILER: Richard Savoy;RSavoy5578@aol.com;
prints and mails CML; sends late reminders;
ARCHIVIST and MANAGING EDITOR: David Mohr,
lordronin@videocam.net.au; oversees receiving, maintaining and copying CML archives; backup editor;
MEMBER BIO EDITOR:Brian Vaughan:
frogissam@bluecity.org; edits member addresses and BIO's; denotes member as "Friendly Correspondent" in BIO's at member request;
E-DDRESS EDITOR: Joseph Fenn, jfenn@lava.net;
maintains email addresses of members; posts list online;
WEBMASTER: Andrew Schwartz, andyschwartz@hotmail.com;
maintains MaiLink website, http://mailink.videocam.net.au;
QUESTION & ANSWER MAN: Rolf Miller; answers C= related questions of wide variety; rolfmiller@aol.com;
YELLOW PAGES EDITOR: Jorge Pedreira, maintains resource lists of C= products and services; produces March/September Yellow Pages; edar43@abaconet.com.ar
MAILINK-ON-DISK EDITOR:Ken Barsky;creates disk version of CML for subscribers; KBarsky@msn.com.

THE EASY WAY TO CONTACT OFFICERS:

president@MAILINK.videocam.net.au
vicepresident@MAILINK.videocam.net.au
treasurer@MAILINK.videocam.net.au
bioed@MAILINK.videocam.net.au
emailed@MAILINK.videocam.net.au
webmaster@MAILINK.videocam.net.au
managingeditor@MAILINK.videocam.net.au
diskeditor@MAILINK.videocam.net.au
editor@MAILINK.videocam.net.au

COMMODORE MAILINK POLICIES and RULES:

Any and all opinions expressed in this publication are the views of the authors, and in no way necessarily reflect the viewpoints, attitudes or policies of Meeting 64/128 Users Through the Mail, unless so stated or indicated. Neither Commodore MaiLink nor Meeting 64/128 Users Through the Mail condones piracy of copyrighted software or other material. All programs published are with the permission of the author, or are to the best of our knowledge, in the public domain. Software offered for sale is said by the seller to be either public domain, or if commercial, is the original disk with original documentation. All manuscripts or any material for review or publication should be sent to the editor of the next issue. Commodore MaiLink reserves the right to edit submissions.

Members may place free Commodore BUY, SELL, TRADE ads in the MaiLink. Send diskfile or short note of ad to next editor.

EDITOR GUIDELINES: (abbreviated version))

Editing a newsletter involves collecting articles, programs, and other items, then placing these items in printed newsletter form. An editor may opt for the "cut and paste" approach, where artioles, once printed, are literally cut and pasted onto each "master page". Or, publishing software may be used.

Editor should where possible use the two-column per page and right and left page margins, except page 2, should be no less than 3/4 " . Most text should be 12 point or larger and a good rule of thumb is to use no more than three fonts per page.

Requirements for the Editor include, regular columns such as TWS, geoPublish Tutorial , BASIC, and Q&A DESK, and if submitted, ARCHIVES, BUY/SELL/TRADE ads, Treasurer's Report, Obituaries, Announcements, Address Changes, and New MemberBIO's. Also the Editor must detail the computer system, including software and hardware and printer used in the production of the current newsletter, as well as name, requirements, and deadlines of next editor. This could be in two separate columns: "The Editor's Desk" and "The Next Editor". If as Editor, you receive more than enough to fill pages 3-18, you will then forward to next editor the excess submissions.

The Editor should edit each item as it arrives, creating a diskfile and a backup diskfile, making certain a TWS version, or TRUE ASCII, of all disk files is (also) created. A good, clean "master copy " on single, unfolded sheets, printed on one side only and protected by cardboard should be sent to our Mailer, Richard Savoy, by the first day of the month you are editing. **NOTE #1** TWS or TRUE ASCII diskfiles should be sent CML Disk Editor, Ken Barsky. **NOTE #2:** to include a color page, send 100 completed sheets of that page, printed on both sides (ready for insertion into CML), along with the "masters" . If you are new at editing, a second copy of CML (copied on the back of junk mail is OK) should also be sent to the President for proofreading purposes.

Always keep your backup disk in a secure place until the MaiLink is in the hands of members. If you suddenly realize it is near the first of the month, and you are running late, notify the President who can help expedite matters.

HOW TO JOIN M.U.T.T.M.

Send completed application, with check or money order made payable to Emil Volcheck, Jr., to: President, R1 Box 120T, Black, MO 63625, USA. Dues are $15 USD for continental U.S., $17 US for Mexico and Canada, $25 US for all others. You will receive Commodore Mailink in January, March, May, July, September, and November, plus COMMODORE YELLOW PAGES and BIO's in March and September. Enjoy.

NEW MEMBERS BIOS

^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

YUL HAASMANN, 6944 CLEARWATER AVE. LAS VEGAS, NV 89147 Yul Hobbies: Archery, Medieval reenacting, Music. System: C-64, C128, Amiga 3000, 1541's & several Monitors, he also has a Macintosh Powerbook. Interests: Translating Programs and music composing. Email:(haasmann1@yahoo.com), Tel: 702 332 2101 Fax: 702 798 0648.

^^

CCCC Commodore Expo

(Cincinnati Expo June 3-4 2006)

———Original Message——

From: "Dragos"(mgladson@qmail.com)

CCCC is pleased to announce beginning on June 3rd, 2006, and continuing thru the 4th, we are hosting an expo for all Commodore enthusiasts. The event is taking place within greater Cincinnati, Ohio, at the Drawbridge Inn of Ft Mitchell KY. A huge room will be available, 3648 square feet, for demos and dealers. There will be a special room rate for C=4 attendees as well: $69/night + tax! Reserve a room today and help make C=4 a hit!
http://www.c4expo.org

The url for the hotel is:
http://www.drawbridgeinn.com/

^^

FREE GOODIES!

By: Brian Vaughan

Brian Vaughan is offering for free all issues of Compute's Gazette magazine from July, 1983, to June, 1990. These issues are in one box which weighs considerably. I can also send a second box with folders containing the best articles from the magazines Ahoy, Commodore, the old and new Compute!, and Run. I only ask that I be reimbursed for the cost to ship these boxes".
Brian

A LITTLE OF THIS & THAT!

By: Richard Savoy

This is the morning of April 21, and I have everything I need in house to now assemble, the May issue of the Commodore Mailink (CML) except to finish this page and page 18, neither is any big problem. The three items to the left had been planned to take up the page and any other last minute items, I hope I haven't forgotten anything, it's not like the old days when Commodore was "King"! I know we think it still is. In those days CML had material usually for the next issue.

Now-a-days, there isn't a great deal of current news to write about our beloved machines. So we have turned to more teaching series. A great new one to us is on page 4&5 of this issue and will be in that location in future issue. Welcome aboard Jane Jones.

My new series is well under way "**The Beginners Corner Lesson # 2**" is from the Commodore User's Guide, one comes with each new machine that was sold! If so, why are we re-copying it in our newsletter? Good question. First, many of our computers are hand me downs, and one thing often missing is the Manual. Second, how often when we get something new with a manual, we read just enough to get the item working and that's it. True we go back when we have trouble or want to learn something new. It sits in a draw and gathers dust. If you don't have a manual than these articles should be very helpful particularly if you don't have any other way of getting help. We have members who have never been to Commodore club meeting and others that can only get help through our group. More than once, when a member had passed on, a spouse would write and tell how the person who passed-on would look forward to the newsletter and sit down and read each issue from cover to cover. So I hope that that when you read the pages of this issue you learn something to improve the use of your Commodore. Keep Smiling. It's healthy.

GEOSGenie

by Jane M. Voskamp - Jones

An Introduction

Hello Commodore and GEOS users. Since I live in Australia, South Australia to be exact, in the state capital of Adelaide, I may well seem a long way away from the action, but when we use our wonderful Commodores, we are all still connected.

The last column I was writing was for Dale Sidebottom and UCUGA, unfortunately it ended abruptly due to circumstances beyond everyone's control - things happen you know, and it did not bother me one way or another. Luckily for me, Linda Tanner realized that I was then without a newsletter to write for and she has asked me to introduce myself to the members, and write a column for you. Linda has read some of my previous GEOSGenie columns which along with the brief set in UCUGA, have been published in the Melbourne Commodore Computer Club Inc's Newsletter {MCCC} in the state of Victoria Australia.

I do have a little spare time on my hands at the moment, and I don't mind - in fact I never mind writing about my Commodores and their GEOS connection.

Let's not beat around the bush. You all probably require a little history to indicate my knowledge of the Commodore and GEOS systems. I was a member of a local club here in Adelaide called the South Australian Commodore Computer Users Club Inc {SACCUG}. I had been writing various columns for this small club's newsletter since 1988 up to the end of 1995, when the club changed platforms. I have also written programs and I was lucky enough in 1992-1993 to have had them published in both RUN Magazine Disk and Loadstar128. If that gave you a headache to think about, then even I will agree that it was a long time ago. I have also written a suite of twelve interconnected HandBooks and published them all over the years, along with a whole bunch of ClipArt and graphics products. The HandBooks are not exclusively about GEOS either, three are specific to the Commodore 64, 128, and their disk drives.

Some of you probably remember the ill-fated GeoMetrix club - I was the Jane who won the Grand Prize Award for their 'Design a Flyer competition' in 1991. Then shortly after that I was the Jane invited to write the Art Gallery column for the equally ill-fated GeoVision International Magazine run by Grady Brown {ex GeoMetrix fame himself} in 1993. There were only three issues of that Magazine produced {I am purposefully going to ignore the strange pattern that is starting to appear - smile}

After GeoVision went down in a screaming heap {circa 1993}, I was invited in 1994 to write a column for another Magazine here in Australia - the Commodore Network based in Victoria, Australia. This Magazine was the launch of the GEOSGenie column. I contributed columns for a couple of years until it reached 22 issues. Then that magazine also closed down as the producer switched platform in 1996.

Why a GEOSGenie? I was given the name by the publisher of the Commodore Network Magazine back in 1993 after he personally saw

GEOSGenie, Continue from page 4

me working with GEOS in front of him, apparently I impressed him so much that thereafter he called me the GEOSGenie, and it just stuck. I never really thought about it that much myself, I simply enjoyed working within the GEOS environment and getting the programs to do what I wanted, and it seemed I had a knack for teaching these tricks to other users. This last part being most important with GEOS, making it more user friendly and fun.

After the Commodore Network vanished in 1996, Ivan Blitz, the president of the Melbourne Commodore Computer Club Inc {MCCC} hinted in 1997 that their publication could use a new column about GEOS, and so the GEOSGenie ReAppears column sprung into life. I contributed monthly columns to the MCCC Newsletter until in the end, I had written a massive 70 columns for them when the club finally disbanded in Feb2004.

And it was at this point that Dale invited me to participate in the Commodore Digest for UCUGA, which I was just as happy to do. And that brings us up to the present day in 2006. Instead of starting again from the beginning, I have a feeling that most of you supported Digest, so my thought is that I will pick up the threads of that column, and continue it in Commodore MaiLink. And naturally, there are quite a number of GEOSy things to delve into.

I always write my preliminary text for my columns in Busy Bee 128 v2 {an Eric Lee program} on my other C128D, this obviously will integrate well with MaiLink's disk. This way I can type faster and the program keeps up, plus throw in the spell checker in the 1750REU and we make a good team. I then import my text to GEOS using the TEXT GRABBER128 with C128 Generic1 program provided with the GEOWRITE WORKSHOP128 Pack.

I did have more humble beginnings on GEOS64, but my prodigious volume of material over took my first C64 with C1541 drive {purchased in 1983 and 1985 respectively}, and expanded into C128D GEOS128 v2.0 with 1750 REU {upgraded to 2Meg} with C1581 and C1571 drives hooked up - yes, four drives!. But that was as far as I got with equipment though, due to financial restrictions

I have always cherished the day that I started learning GEOS I could never have self--published twelve HandBooks without geoPublish, or been able to help as many people as I have over the years. And don't worry, I won't make you do anything with GEOS if you don't want too - smile. I remember how much it scared me in the very beginning, but as my understanding of the program grew, I was able to overtake the programs grumbles and tame it into a fun beast

Next issue, I will begin to explore GEOS with you, and I hope that we will have many good times together. I also look forward to meeting some of you through the mail, or even email if I'm lucky. So keep using your Commodores, and persevering with GEOS, and most importantly, having fun with the Commodores that what we already love and have. Always there for you, in GEOS land...

Readers Three Wishes And All That...

Send in your comments, or great GEOS discoveries, and I will respond when I can in this column, unless you wish a private reply, in which case please send an overseas SSA{Business}E and I will write you back. You can even just let me know if you are enjoying the column

GEOSGenie PO Box 635
Enfield Plaza South Australia 5085
International Telephone 61 8 8260 4062
{just kidding, I don't expect you to phone this time zone}
email : jmvgrafix@yahoo.com.au

BASICALLY SPEAKING

by Linda Tanner

In this issue, we pave the way for using each of SID's three voices, separately, then simultaneously. You will notice in our program in statements 6,7, 8, and 9 (p.7) that we have defined our variables, including some we did not even use. It is much easier to define all related variables at once, with anticipation of eventually using them all, in a later, more complex, but related program. L1,L2,L3 and H1,H2,H3 represent the low-and high-bytes, respectively, of Frequency Control Registers 54272-54273 for voice1, 54279-54280 for voice2, and 54286-54287 for voice3. Since an 8-bit register cannot hold a number larger than 255, we need two registers to store our sound frequencies, all of which is detailed in the SID literature and requires more space than is available here.

Our program on page seven does not "do" much, as the purpose of this exercise is to demonstrate some basics of SID programming for the C64. One major difference between the sounds emitted by various systems, aside from the uniqueness of each and every SID, is whether SCPU is present and enabled. Due to speed discrepancies, it is advisable for those without SCPU to change the "counter" in lines 24, 34, and 44 from 4000 to 40 or so, as the FOR/NEXT loop is what gives pause between sounds in a given "set" of sounds such as ringing or buzzing.

Lines 10,12,13 and 14 activate voice 1with a "triangle" waveform, lines 20,22,23 and 24 activate voice 2 with a "sawtooth" waveform, lines 30,32,33 and 34 activate voice 3 with a "pulse" waveform, and lines 40,42,43, 44, 45 and 46 cause each of our three voices to sound as nearly as possible to simultaneously. Each voice is programmed similar to the others, except that each uses a different waveform. We'll examine voice 2 as an example.

FOR V2=1TO3 in line 20 allows us to sound voice 2 three times--once for each trip through the FOR/NEXT loop, for example, as three rings on a phone. Our FOR ZZ=1TO20 is what "extends" each of the three sounds. Each "ring" or "buzz" is actually composed of 20 distinct identical sounds emitted in rapid succession, giving the impression to the casual listener of one ring or one buzz.. We must clear our sound registers (54272-54296) before and after using our SID which is what our FORS=L1TO FM:POKES,0:NEXT does. POKEFM,15 sets volume on high. In line 22, our POKEW2,33 tells our Commodore to apply the sawtooth waveform and enable the "gatebit". In a nutshell, you POKE into the Waveform Registers for each voice, W1, W2, or W3, the numbers 17, 33, or 65 to set that voice for Triangle, Sawtooth, or Pulse waveform, respectively. To turn a waveform off, one would re-set with 16, 32, or 64. POKEW1,65, sets Pulse Waveform on for voice 1 and POKEW1, 64 sets Pulse Waveform off (disables gatebit) for Voice 1. Two very important chores are left: creating our ENVELOPE and setting Frequency.

H2 and L2 in line 22 are the high- and low-frequency control registers for voice 2. We determine frequency by adding our "low frequency number" to the quantity, 256* high frequency number. The best example is our own in line 12,22, or 32. Our low-frequency number, 48 is added to the quantity (256*28) to produce 48+7168=7216. Why the discrepancy between our listed frequency of 7217 and our calculated one of 7216? Pretend you don't have any of the Frequency Tables provided in all the SID programming books and chapters. How would you know what numbers to POKE into H2 and L2? Simple (almost). Select a frequency. Let's pick 7217. Now divide your desired frequency by 256, and your dividend is the number you will POKE into your High Frequency Register (H1, H2 or H3) which in this case is 28.. But we have a remainder of 49. Our remainder is the number we now POKE into our Low Frequency Register, L1, L2 or L3. Minute "discrepancies" may occasionally arise when one consults a Frequency Tables rather than manually calculating the numbers, but in our case it is OK, since we are not doing anything important yet. If you wanted to create a sound for voice1 with a frequency of 10,000 cycles per second (cps) or 10,000Hz, you would divide 10,000 by 256 to get 39 with a remainder of 16. Thus you would POKEH1,39 and POKEL1,16 and those two POKE's create the 10,000 Hz frequency for voice 1. It's that simple.

Our sound envelope of Attack/Decay/Sustain/Release, or ADSR, also requires two separate registers to hold our numbers, with Attack and Decay being held in E1, E3 or E5, and Sustain and Release stored in E2, E4 or E6, for voices 1,2 or 3, respectively. You can select values from ADSR Tables, or you can calculate your own. The A/D register is divided into the 4-bit "high nybble" for Attack and the 4-bit "low nybble" for Decay, and each can have a "value" from 0-15. The value for Attack must be multiplied by 16 and added to the value for Decay to produce your number to be POKE'd into an AD register such as E3. If we wanted an Attack of 1 and a Decay of 10, we would first calculate our value: ((1*16) + 10) =26, and then we would POKE E3,26. Likewise, Sustain and Release occupy the high nybble and low nybble of the SR register, and if we wanted a Sustain at volume of 11, and Release of 5, we first calculate ((11*16) + 5)=181, thus we would POKE E4,181. We continue "Sound" in July, 2006.

"SOUND64-2.0"

by Linda Tanner

```
6 L1=54272:L2=54279:L3=54286:H1=54273:H2=54280:H3=54287
7 P1=54274:P2=54275:P3=54281:P4=54282:P5=54288:P6=54289
8 W1=54276:W2=54283:W3=54290:E1=54277:E2=54278:E3=54284:E4=54285:E5=54291
9 E6=54292:FL=54293:FH=54294:RS=54295:FM=54296:OS=54299:EG=54300
10 FORV1=1TO3:FORZZ=1TO25:FORS=L1TOFM:POKES,0:NEXT:POKEFM,15
12 POKEH1,28:POKEL1,48:POKEW1,17:REM FREQ7217/TRIANGLE/GATEBIT=ON
13 POKEE1,10:POKEE2,245:GOSUB89:POKEW1,16:GOSUB89:REMADSR=0/10/240/5
14 NEXTZZ:FORB=1TO4000:NEXT:NEXTV1:GOSUB88
20 FORV2=1TO3:FORZZ=1TO20:FORS=L1TOFM:POKES,0:NEXT:POKEFM,15
22 POKEH2,28:POKEL2,48:POKEW2,33:REMFREQ7217/SAWTOOTH/GATEBIT=ON
23 POKEE3,10:POKEE4,245:GOSUB89:POKEW2,32:REM ADSR=0/10/240/5
24 NEXTZZ:FORB=1TO4000:NEXT:NEXTV2:GOSUB88
30 FORV3=1TO3:FORZZ=1TO10:FORS=L1TOFM:POKES,0:NEXT:POKEFM,15
32 POKEH3,28:POKEL3,48:POKEW3,65:REMFREQ7217/PULSEWAVE/GATEBIT=ON
33 POKEE5,10:POKEE6,245:POKEP5,248:POKEP6,15:GOSUB89:POKEW3,64
34 NEXTZZ:FORB=1TO4000:NEXT:NEXTV3:GOSUB88
40 FORZZ=1TO10:FORS=L1TOFM:POKES,0:NEXT:POKEFM,15
42 POKEH1,253:POKEH2,253:POKEH3,253:POKEL1,34:POKEL2,34:POKEL3,34
43 POKEW1,17:POKEWEW2,33:POKEW3,65:POKEP5,248:POKEP6,15
44 POKEE1,242:POKEE3,242:POKEE5,242:POKEE2,240:POKEE4,240:POKEE6,240
45 NEXTZZ:POKEW1,16:POKEW2,32:POKEW3,64
46 FORS=L1TOFM:POKES,0:NEXT:POKEFM,0
86 GOTO90
88 FORS=L1TOFM:POKES,0:NEXT:POKEFM,0:RETURN
89 FORA=1TO200:NEXT:RETURN
90 REMVOICE1 L1=FREQCTL-LOBYTE H1=FREQCTL-HIBYTE P1/P2=LO/HIBYTE
PULSWVFRM WDTH
91 REM VOICE1 WAVFRM CTL REG=W1 ENVELOPE GEN1 (A/D)=E1 ENVELOP GEN2 (S/R)=E2
92 REM VOICE2:L2/H2/P3/P4/W2/E3/E4
93 REM VOICE3 L3/H3/P5/P6/W3/E5/E6
94 REM FL=FILTERCUTOFFFREQ LOW NYBBLE(BITS2-0) FH=FILTCUTOFFFREQ
HINYBBLE(7-0)
95 REM RS=FILTER RESONANCE/VOICE INPUT CONTROL REGIST.  FM=FILTER
MODE/VOLUME
96 REM OS=OSCILLATOR3 RANDOM# GENERATOR/VOICE3 NUMERIC OUTPUT(BITS 0-7)
97 REM EG=VOICE3 ENVELOPE GENERATOR OUTPUT
99 REM NEED 5 ITEMS MINIMUM FOR 64
SOUND:VOL/ASDR/FREQ/WAVFORM/GATEBITENABLE
100 REM(SOUND64-2.0)3 VOICE PROGRAM FOR MAY 2006 CML BY LINDA TANNER
```

SLANG (Scpu LANGuage)

by Stephen Judd

Slang Tutorial Lesson #4:

Replace the for-loop in the example code with a) a while-loop, and b) a repeat-until loop, but make it print out the text five times instead of three. Compile and run each case to verify that it works.

Ending a program

You'll notice that the program ends with the line "done". In BASIC, putting an "END" in is optional. In Slang, it is required. It tells the C64 how and when to exit the program. If you'd like to see what happens when you leave it out, go ahead and try it! Just remember that "sys 54016" will restart Slang after you reset the computer.

We can now explain the last few lines as well. The command "waitchar" simply waits for a key to be pressed. Without this line, the program would immediately end and return you back to Slang. The "waitchar" would not be necessary if this program were being called from, say, BASIC -- in that case, you would probably want control to return to BASIC immediately. But when running programs directly from the Slang editor, you'll usually want to put a line like this at the end of the program.

Slang Tutorial Lesson #5:

As a test, go ahead and comment out that line (place a ";" before the waitchar), re-compile, and see what happens when you run the program!

Print

The little program above used the "sprint" command. There are actually two print commands in Slang: print, and sprint.

Sprint -- Simple Print, or String Print -- is a simple print command that only prints strings (not numbers). As we shall see shortly, the regular print command contains certain things in a _library_, which requires an extra step, and will make your programs larger. For the moment though we can focus on sprint.

Printing a string in Slang works much the same in BASIC, with just a few little twists. Unlike in BASIC, you cannot embed special characters inside the quotes, like: 10 PRINT"[ctrl-2][SHFT-CLR]nice white text on a clear screen." Instead in Slang, you have to use the character codes directly; in BASIC you could also do the above as:

```
10 PRINT CHR$(5)+CHR$(147)+"nice white text on a clear screen."
```

In Slang, you simply use a ! instead of chr$, so this would look like

```
sprint !5!147"nice white text on a clear screen."
```

The above command has one key difference from the BASIC version, however: in BASIC, PRINT normally prints a CHR$(13) at the end of the line, whereas sprint will not. It turns out that there's actually two more print commands in Slang, though: println and sprintln. These work exactly the same as print and sprint, but print an extra [RETURN] character at the end.

Slang Tutorial Lesson #6:

Go ahead and try replacing "sprint" with "sprintln" in the example program, and see what happens.

Slang Tutorial Lesson #7:

Modify the example program to print out the text in cyan. How about light green?

There's one more trick to print and sprint: you can specify where_ on the screen to print::

```
sprint (0,20) "some text"
```

will print the text at row=0, column=20

Slang Tutorial Lesson #8:

Modify the program to print the text to the middle of the screen. In contrast to sprint, print can print numbers and strings, and you can stick them all together on a single line. The catch is that many of the routines needed to do this are in a _library_. This library is a file on the disk, containing special routines. In general, libraries can be source code, object code, or a special kind of file (a relocatable file) for use by the linker, which I won't talk about here.

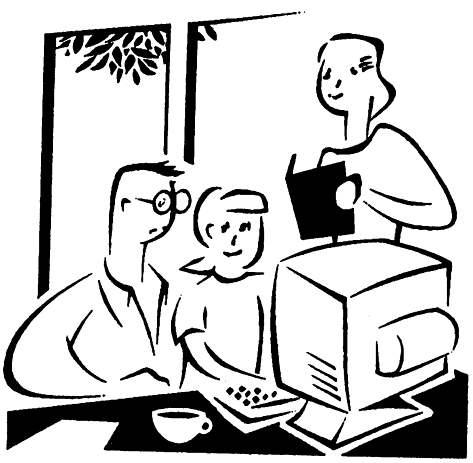
Slang Tutorial Lesson #9:

What we are going to use is a simple source code file. Try typing in the following code, and compile and run it:

```
byte b
for b=1:10
 println "b=" b
next
print "press any key..."
wait char
done
put 'putcore.e.s'
```

The two critical differences here are 1) we are now using print instead of sprint, and 2) the "put" command at the end of the file.

(continued on p.15)



THE BEGINNERS CORNER

LESSON # 2

Prepared By: Richard Savoy

****

Last time we got together we made sure we had all the equipment to get the Commodore 64 you and running. Today, we will connect the equipment.

Installation

Connections to your TV (as shown in the column.)

1. Attach one end of the TV cable to the phono type TV signal jack at the rear of the Commodore 64. Just push it in. Either end of the cable can be used.

2. Connect the other end of the cable to the antenna switchbox. Just push it in.

3. If you have a VHF antenna, disconnect it from your TV set.

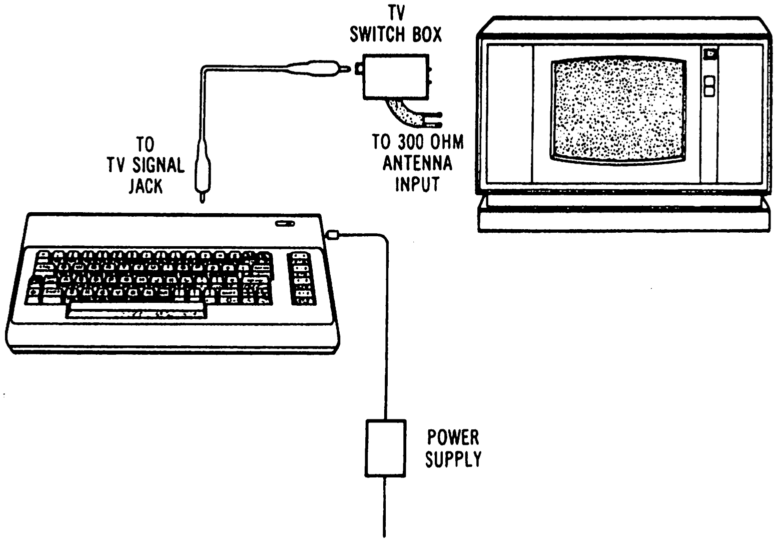
4. Connect your VHF antenna cable to the screw terminals labeled "antenna input" on the switchbox. If your antenna cable is the round 75-ohm coax type, use 75-ohm to 300-ohm adapter (not supplied) to attach your antenna cable to the switchbox.

5. Connect the twin lead output cable to the antenna switchbox to the VHF antenna terminals of your TV set. If your set is one of the newer types with a round 75-ohm VHF connector, you will need a 300-ohm to 75-ohm converter (not supplied) to connect the switchbox to the 75-ohm VHF antenna input on the set.

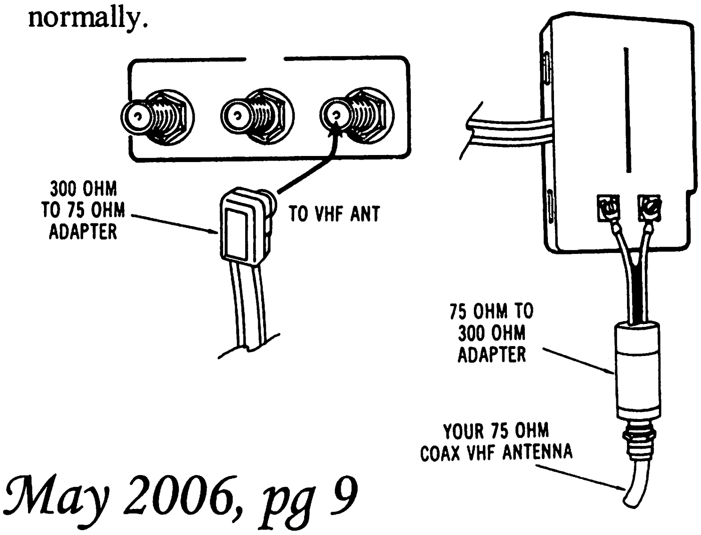
6. Set the TV's VHF tuner to the channel number indicated on the computer's channel selector switch (channel 3 move the switch to the left, channel 4 move to the right).

If a strong local TV signal is present on one these channels, select the other channel to avoid possible interference.

8. Plug the power supply cable into the power socket on the side of the Commodore 64 (C64). Just push it in. It is "keyed" to allow insertion in only one direction, so you can't connect the power cord the wrong way. The power supply converts household current into the form the computer uses,



The C64 is now correctly connected, No additional connections are required to use the computer with your TV. The antenna switchbox will connect the computer to the TV when the slide switch is in the "computer" position. When the switch is in the TY position your set will operate normally.


300 OHM TO 75 OHM ADAPTER
TO VHF ANT
75 OHM TO 300 OHM ADAPTER
YOUR 75 OHM COAX VHF ANTENNA

GEOPUBLISH TUTORIAL PART 16

Drawing Tools Part 1

By: Bruce Thomas

^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

Now that we have our Article Titles, Captions and Pull-Quotes in place we can add some graphic elements to our pages. In this first look at the drawing tools available in Page Graphics (P.G.) Mode we are going to concentrate on Lines and Boxes.

Another Blueprint

Once again, our publishing efforts require some planning. If you have already printed off Thumbnail pages of our project up to this point, that is great. If you haven't, then now is the time to do so. Using PostPrint or geoPubLaser it is as simple as checking a box to print a page of Thumbnail images. If you don't have access to a PS Printer at the moment you could print each geoPublish (gP) Page to a geoPaint file using the Paint Pages Printer Driver and then use the Thumbnail program from the RUN GEOS Power Pak Disk to place up to 16 gP pages into a single geoPaint proofing file to print off.

With our Thumbnail printout and Red Pen it is a simple process to mark the location of our graphics on each page to ensure that we don't miss anything.

For this project we will draw a line down the left guideline. On pages where an article begins we are going to draw a line across the page below the title. On subsequent article pages we are going to place a shaded box around our titles. We will also add some highlights around our Pull-Quotes on Pages 2 and 4.

Better than Grade 1

We are going to start off drawing simple line. Unlike using Crayola Crayons, however, gP provides some very handy features to make drawing lines easy and impressive.

Start gP and OPEN the TutorialPart1 document. GOTO Page 1 and change to P.G. Preview. To make drawing our lines easier we want to turn on ratchet (C= R) under the options menu. Ratchet is similar to snap in that it controls the pointer placement but it does so in a different manner. Where snap attaches the pointer to the Master Page (M.P.) Guidelines, ratchet uses a grid system. Under the options menu you will see a choice to `set ratchet'. Choose this and you will see that we can use a `two pixel ratchet' or a `1/2 inch ratchet' as our grid. Choose the `1/2 inch ratchet' and press OK.

We are going to draw our vertical line down the left side of the page first so select the line tool [gP Page 3-5] and then open the attributes for it. We want to choose `square' and `transparent', solid black and the narrowest line width. Press OK and then place the crosshairs at the top left corner of our page (1V) and click and drag the line down to the 10" mark. You will see how ratcheting makes this easy to do.

Our next line will be under the title so place the crosshairs at the 3V location, click and drag across to 3Z.

This will be our basic page configuration for graphics. With the M.P. line across the bottom of the text we have enclosed our text on 3 sides. The gap between where the left line ends at the 10" mark and the M.P. bottom line leaves the page a little less confined than if we had joined the two lines and ratcheting made it easy to do.

Goto Pages 3, 5, 7, 10, 13 and 14 and add these two lines to each page in the same way we did on Page 1. We'll come back to Pages 10 and 14 to finish them up later. **Continue on page 11**

++

GeoPublish Continue from page 10:

And now a Box

Goto Page 2 and draw the left edge line down the page to the 10" mark. Select the `Rectangle/ Square' [gP Page 3-6] tool and then open the attributes for it. We want our headlines to be visible but we also want to add a little flair to our pages. Select Square, the narrowest frame width, Transparent, and then click the up arrow beside the pattern box (which should be empty by default) 7 times to get a lightly shaded pattern. This pattern translates into a 12.5% gray shade when laser printed which gives a nice background for our titles. Click OK to close the attributes box.

Click the crosshairs pointer at 1A and drag across and down and click a second time at 2E to make a shaded box for our headline. The attributes for the graphic objects are retained until you change them so it is a simple process to goto Pages 4, 6, 8, 9, 11, 12, 15 and 16 and add the line and box to each page.

Once you have completed adding these elements toall of these pages goto Page 10. We continued an article onto Page 10 so we must draw a box around its' headline in the middle of the page.

You should be able to see the headline text just below our lower regions. This makes it so that our 1/2 inch pixel ratchet won't line up our box exactly where we want it. We have 2 options. We could use the 2 pixel ratchet or we could stick with the 1/2 inch ratchet and just adjust the box in Zoom view after initial placement.

I am going to stick with the 1/2 inch ratchet since the 2 pixel ratchet doesn't line up where I want it to and I'll have to adjust the box afterwards anyhow. The 2 pixel ratchet seems to move 4 pixels in the horizontal direction and 8 pixels vertically so it won't produce the box I want.

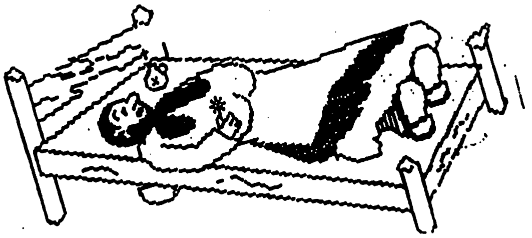
With the 1/2 inch ratchet selected place your crosshairs at 5 1/2"A then click and drag to 6"E. Select the Pointer tool and select Zoom in the disp menu. Drag the Zoom box down the left edge of the page to the 5" mark and click to place it.

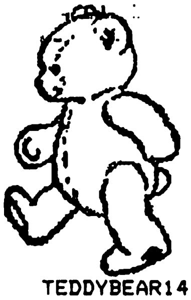
After the screen redraws lightly bump your pointer against the right edge of the screen and quickly move the pointer back to the left. Wait for the screen to redraw and then do this again. These two bumps will move the screen over enough so that our left guideline (A) is almost at the edge of the screen where our ruler is. This will make it easier to judge where we want to place our box. Hide the Toolbox (C= T or use the options menu toggle) and then click on the shaded box. Click on the `move button' and drag the box down so that the top of it is 1 pixel below the 5 5/8" mark (the first notch below the original position) and click to place the box. Unhide the Toolbox (C= T) and click on the Pointer tool to remove the bounding box from our shaded box. Next, click the Update Tool [gP Page 3-5] to redraw our page. The shaded box is positioned just right to have our headline centered.

Now, go to Page 14 and then change to Preview under the disp menu. By changing pages first we don't have to wait for Page 10 to redraw in Preview before changing to Page 14 and having to wait for it to redraw. Select the Rectangle/ Square tool and draw a box from 6"A to 6 1/2" E. Change to Zoom view and move the Zoom box down to the 4 1/2" mark. Select the Pointer Tool and then `bump' the right edge of the screen twice. Click on the shaded box and, using the move button, move the shaded box up to 1 pixel below the first ruler mark above the 6" mark. Click to place the box. Select the Pointer Tool and then select the Update Tool to redraw the page.

Continued on Page 12

SPARROW 23x22





TEDDYBEAR14X18

GeoPublish Continue from page 11:

Highlighting

Our two pull-quotes on Pages 2 and 4 need a little help to make them stick out from the page the way we want. Remain in Zoom view and goto Page 2. You should see our pull-quote in the lower portion of the screen

Select the line tool and draw a vertical line down the page from the 1" ruler mark. Draw a parallel line down the page from the 3 1/2" ruler mark. These lines will ensure that our next two lines are drawn correctly.

Our next 2 lines will be drawn, one above and one below the Special Text, centered between our Special Text and our article text, across the page from the 1" vertical line to the 3 /12" vertical line. After drawing these 2 lines, select the Pointer, click on the vertical line at the 1" mark and then click on the Delete Tool [gP Page 3-6]. Next, delete the vertical line at the 3 1/2" mark and then press the Update Tool to redraw the screen.

Goto Page 4 and then we have to scroll the page to the right and down some to locate our Pull-Quote at the 7" vertical mark in the center column. We want to draw our vertical lines about 1/4" inside the column guidelines. Then, we want to draw our horizontal lines on top and bottom of the Special Text and between our vertical lines. Once you have done that Delete the vertical lines and Update the page.

If you have been crossing off the graphics on our Thumbnail page you will see that we have completed adding our finishing touches to TutorialPart1. Close this file and open TutorialPart2.

More of the same

Use whatever method you have to and create a Thumbnail page for TutorialPart2. Go over this page with your red pen and mark the places where we need our lines and boxes.

With TutorialPart2 open, goto Page 17 if it isn't open there, select the line tool and then the attributes. Attributes don't carry over from one document to the next so we have to ensure that the chosen options are Square, narrowest frame, transparent and solid black pattern. Turn on Ratchet under the options menu and make sure the setting is for 1/2 inch ratchet. In Preview add our Left Edge line and our line under the title. Add the same two lines to Page 18, 20, 22, 23, 24 and 28.

Goto Page 19 and set the attributes for the Rectangle/Square Tool to Square, narrow frame, transparent and click the up arrow 7 times from the blank pattern to select the 12.5% shading pattern. Draw the Left Edge line and shaded title box on this page and Pages 25, 26, 27, 29, 30 and 31.

Goto Page 18, select the Rectangle/Square Tool and draw a shaded box from 5"A to 5 1/2"E. Change to Zoom View, place the zoom box at the left edge of the page at the 4" mark and `bump' the screen to the right twice. Select the Pointer Tool, hide the Toolbox (C=T), click on the Shaded box and use the Move Button to move the box up by 4 pixels. Unhide the Toolbox, click the Pointer tool to remove the bounding box and the press the Update Tool to redraw the screen.

We have a different Page Layout in this gP document so let's goto Page 21 and then change to Preview. This page has two articles on it. We need to draw a line down the left edge of the page as well as one down the middle guideline (X) - extend both to the 10" mark. The left column is a continuation article so it requires a shaded box at the top. Draw a shaded box from 1V to 2X. The right column is a new column so it needs a line under the title from 3V to 3X.

We have to resize the shaded box so it isn't up against the center guideline. Change to Zoom view and move the box to the middle of the page.

Continue on page 13

GeoPublish Continue page 13:

Make sure the Pointer Tool is selected and click on the shaded box. Use the Resize Button to move the box to the left 1/4", click to place it, click the Pointer Tool to remove the bounding box and then press the Update Tool to redraw the screen.

Verification

In order to verify that we have placed all of the graphics that we wanted to it would be good to print out another Thumbnail page for each geoPublish file. Compare these pages with the previous Thumbnail pages and make any required adjustments.

And that is all we have to do to add all of the graphics we need to up to this point in our Tutorial Project. From now on the directions on what to do for adding graphics will be included in the format file that accompanies the geoWrite files on the web site.

In a future column we will look at an improved geoPublish program (Todd Elliott's V1.1) and see how an added feature makes it easier to add similar graphics to multiple pages as we have just done.

Next time we will look at more of the drawing tools we have at our disposal in geoPublish.

Until then, enGEOy your Commodore!
Bruce Thomas

++++++++++++++++++++++++++++++++++++

First report of Commodore at CES
Part II

By: Robert Bernardo

Introduction

Robert's article on the International CES Expo started on page 14 of the March CML and continued to page 15 column one, but for the lack of space we had to continue, to this issue. In our conversion I had said how much I personally enjoyed his articles as I had been to Las Vegas in the past. His reply was, "Some people don't understand that the article is meant to be a feature article (in other words, they don't understand why I ramble and not get straight to the interview.") But he thanks us all for our support.

So here we go where we left off!

I felt like s small fry in a pond of big fish. Not sure about whether photographs were permitted, I wandered and looked, eventually stopping by the Commodore 64, 1541, 1701 monitor, and Competition Pro joystick sitting on a stand in the center right of the area. Everything was turned on, and the Commodore was running The game, "Way of the Exploding Fist" (but not the original disk). I got to talking with the man playing with the game, his name Frank Cifaldi. Frank joked, "This is the best thing here." Finding a friend there, I followed him over to the table which was surrounded by benches. On the table were 6-8 Navigator Combos - a black-cased, handheld device with a 3.5 LCD color screen. He showed me one of the Navigator-Combos, went a 3.5 LCD color screen. He showed me one of the Navigator Combos, went through the menu screens, and got to a platformer game which I did not recognize.

"It's running a NES game," he remarked.

"Well, that's not right," I replied. "A Commodore device should run Commodore games."

"Yeah. That's the only emulator that they have in it." He went back to gaming with the device, complaining that the game forced you to use in the right button in 2 different directions at the same time. As he played, he went on to speak to an associate of his, and I walked over to the long information counter in the back, picked up brochures for the 3 new Commodore devices and started studying the prototypes that were on display.

The Commodore MediaTower kiosk was on the left end of the Commodore area. Standing next to it was Ian Matthews, Canadian > webmaster of http://www.commodore.ca and employed by Commodore as a representative in this show.

Continued on page 14

Continue from page 13 Commodore at CES:

He was busy talking to another business-type; I wanted to talk to Ian, because he had been Recommended > to me by Brian Bagnall, author of the book, "On the Edge: the Spectacular Rise and Fall of Commodore." I waited, but he was oblivious to everything else, as were the "suits" talking to another business types.

Behind the MediaTower was the Commodore MediaBox, shiny in brushed metal (stainless steel? aluminum?) with a C= symbol that was backlit with a blue glow. (Speaking of C= logos, I noticed that almost all the logos in the Commodore exhibit area had reverted to their non-italicized form but in monochromatic black or silver instead of the multi-colored red and blue). In a vertical display case to the righ of the information counter were eVICs, mPETs, NavigatorCombos, and C64 DTV's with pyramidal red boxes and hexagonal white boxes (but no Hummer DTV). To the right of the display case was a wall of 3 large-screen plasma t.v.'s, displaying the latest t.v. commercials for the new Commodore products.

Frank and others were taking digital shots, but I remembered the CES warning to ask permission. As Frank departed, he whispered, "I don't think they care about you taking pictures." To be sure, I went to American Commodore rep, George, and asked for permission. "Are you with the press? "No, I'm with the Fresno Commodore User Group," and I handed over my business card.
"That's great!"
"And I have a bunch of questions, but I know it's the end of the day. I'll be back tomorrow afternoon."

With about 20 minutes before CES closed for the day, I pulled out my film SLR and started snapping shots. After I took my photos, I noticed that Ian and his business client from Germany had sat down at the NavigatorCombo table. I sat across from them and tried to listen in.

Ian was explaining about Commodore's marketing plans to the client. As he did so, I pulled out the On The Edge flyers from my camera bag. Other attendees to the side of me noticed the flyers, and smiling, they started picking them up and passing them around. One attendee asked the "suits", "Have you read this book?" All the "suits" looked, and one responded, "Yes, a fine book." Another attendee asked me, "Are you the author of the book?"
No, but the author, Brian Bagnall, asked me if I could possibly leave these here for distribution."
Ian gently cautioned, "This is a business function. It's not for marketing."

"Hi, Ian, I'm Robert Bernardo from the Fresno Commodore User Group," and I handed over my business card. "I was wondering if I could leave these here so that passers-by could pick them up."

"And how do you know Brian?" "We sold his books at the Vintage Computer Festival."
"Oh, yes," Ian replied in recognition.
"The Dutch won't like it if you start passing that, eh?"

"Oh, no, we don't want to get them angry. We want them to be on our side." He smiled, and I put the flyers away. Continuing with the talk of the marketing plan, I said, "I thought Commodore was going to go global, except for North America."

Ian responded, "We have Europe and Africa, and we have deals in South America... Chile... "Argentina?" He nodded. He then said that there were plans of going into North America 6 months after that. 6 p.m. came. The "suits" wanted to continue, but the other booth workers were leaving, CES security was walking in, and the lights were being turned off. Time to go. I packed my camera case and notepad and headed for the exit. As I left, one of the "suits", a longer-haired, brunette gentleman waved. I waved back. **TO BE CONTINUE!**

Sorry that all the space we have this issue, we will continue in July issue Richard Savoy

SLANG Tutorial, continued from p. 8

For now,you need the "PUT" at the _end_ of your code -- after the done statement -- if you want to use the full print command.

Notice that we have put both a string and a number on the print line, similar to BASIC. You can also separate these with commas, if you like:

```
print "b=",b
```

It just depends on which way you think is clearer. You can also print to any part of the screen:

```
print(b,20) "this is row "b
```

Slang Tutorial lesson #10:

Replace the println statement with the above "this is row" statement, and see what happens.

As you can see, print is much more powerful and flexible than sprint, but the downside is that you need to PUT the core library in there, which increases compile time and makes the program much larger. Experience will help you figure out when you want to use one or the other.

The core library

Above we used the core library, using the line put "putcore.e.s" at the end of the program (whether you use ' or " quotes doesn't matter, incidentally). This is a very important library and is needed for more than just print.

Slang Tutorial Lesson #11:

Multiplication and division of bytes/ints requires this library, and you will get an error if you don't include the core library. (Print will also generate an error.) Try the following program:

```
int b
b=10
b=b*2
done
```

Now compile it, and you will get an error at the multiplication. Then add the line

```
put 'putcore.e.s'
```

to the end of the program, compile... and it will work. And since you're including the core library anyways, you can go ahead and print out b if you want to.

Why use a library? If you know BASIC, you know that a command like "print" actually calls a routine in the BASIC ROMs to do its thing. The BAISC ROMs are, for the most part, one big library, but they are in ROM instead of in a diskfile. Sometimes machine language programmers will call these BASIC routines instead of writing their own. Similarly, in a compiled language like Slang, sometimes it makes sense to call a common routine to perform some task. A library is nothing more than a collection of useful routines.

Arrays and Strings

We're just about done here. I'm not going to cover _all_ of the available commands; the goal here is to get across the major commands, and then you can browse the slangref.txt document to check out other commands. So I'll just touch on the remaining topics at this point.

Arrays in Slang work much the same as arrays in BASIC. You declare them very similarly to other variables:

```
int b ; regular variable
int c(20) ; array
```

This is similar to using the DIM statement, if that helps. There's just one thing to remember: array indices start at 0. In the above declaration, you can address c(0), c(1), c(2)....c(19), like

```
c(19) = 1000
```

but

```
c(20) = 1000
```

will be incorrect (and may cause a crash). There are 20 elements total: 0 through 19. If it's too confusing, then one thing you can do is to add one extra element to any array -- like, use "int c(21)" -- and not worry about it.

Strings are also really similar to BASIC, but you're going to have to get one thing straight in your head: strings are really just bytes. There is no "string" type, like byte/int/float; strings are just bytes.

You already know this, from BASIC:

```
d$="a" ; treating letter "a" as a string
d$=chr$(65) ; treating letter "a" as the number 65
```

The letter "a" is really just a number -- 65 in this case. This works the same way in Slang:

```
ubyte d
d="a" ; a "string"
d= 65 ; a number
```

That's fine for just one character, but that's not a string. In Slang, a string is just a byte array:

```
ubyte d(20)
d$="hello" ; treat as a string
d(0) = "y" ; change string to "yello""
d(0) = 67 ; change string to "cello"
```

In this example, d is a byte array. You can treat it as a string, or as a list of numbers. As in BASIC, the "$" tells Slang to treat the variable as a string -- I won't go into the details about this, but I think the meaning is pretty clear from the above example. **(continued on p.16)**

Slang Tutorial Lesson #12: (cont.from p.15)

Here's a simple program using strings to try out:

```
ubyte d(20),i
d$="hello"
println "d$=" d$
d(0) = 67
println "d$=" d$
for i=0:5
 println d(1)
endfor
waitchar
done
```

Go ahead and play around with it -- try adding 1 to each element of the string, etc. One important thing to note: when you run the program, you'll notice that the last number printed in the for-loop is a 00. With strings, the very last element will be zero -- this is what tells Slang where the end of the string is. If you overwrite this ending zero, you'll generally get a whole bunch of garbage. So...

Slang Tutorial Lesson #13:

Change the line

```
d(0) = 67
```

to

```
d(5) = 67
```

and see what happens when the string is printed. This overwrites the ending zero byte, and should in general print out a bunch of garbage. Sometimes, if you're lucky, there will be another 00 somewhere in the array!

<u>If-elseif-endif</u>

The if-then structure:

```
if a=10
 do something
elseif a=11
 do something else
endif
```

It's similar to the BASIC command, except that you have an elseif or endif command to mark the block of code to be executed, just like while and repeat.

<u>Subroutines</u>

In BASIC, you are familiar with GOSUB -- it calls a subroutine, and then RETURN returns back to the place at the subroutine call. Slang subroutines are more sophisticated, but the idea is the same and overall they are pretty similar.

A subroutine is like a mini-program: you can define variable, have a bunch of statements, etc. The neat thing is that these variables and statements are local to the subroutine -- they "belong" to the subroutine, and do not interact with other subroutines. This allows you to organize your programs efficiently.

Just like a variable, you have to define a subroutine. And you need to end the routine using "endsub".

With a subroutine, you can pass __parameters__ to the routine. For example, the graphics library has a routine called GrPlot, to plot a point. And it requires two parameters: the x- and y- position of the point to be plotted:

```
GrPlot (x1,y1)
```

Subroutines can also pass parameters _back_ to the calling routine. Once you "get" the idea of parameter passing, and of local variables, the rest is a cinch.

Slang Tutorial Lesson #14:

Type in and run the following program, which demonstrates the idea of local variables within a subroutine:

```
byte b
b=1
TestRoutine() ; Call the subroutine
println "but in the main code, b is still "b
waitchar
done
sub TestRoutine() ; Define the subroutine
 byte b ; create the _local_ variable b
 b=10
 println "in the subroutine, b="b
endsub  ; end subroutine
put "putcore.e.s"
```

You can see what I mean about a subroutine being a mini-program. Within the subroutine you declare variables, have statements, and end it with the endsub. You don't really worry about what other routines or the main program does; the subroutine doesn't "see" them.

So, a few observations: the "sub" keyword defines a subroutine. We _call_the subroutine by simply typing the subroutine name in the main program. Once the routine finishes, the calling program continues executing where it left off.

The subroutine declares its own variable b. Even though it has the same name as the variable in the main program, this is a _local_ variable; it "belongs" to the subroutine, not to the main program.

Slang Tutorial Lesson #15:

In this lesson, we'll try parameter passing.

```
int x1,y1
x1=12
```

(continued on p.17)

(SLANG continued from p.16)

```
y1=20
AddEm(x1,y1)
println "the result was " AddEm<-result
waitchar
done
sub AddEm(int a, int b)<-int result
 result = a+b
endsub
put "putcore.e.s"
```

(As before, <- is the backarrow key).

In this example, the subroutine AddEm takes two input_ parameters, a and b, and has one _output parameter, result. As before, all three of these parameters are local to the subroutine, so they could have the same name as variables in the main program.

With the return parameter, what we are really doing is making a subroutine variable _visible_ to outside routines. Normally all variables and such inside a subroutine belong to the subroutine, and are not available outside of the routine; this is how to make specific ones available outside of the subroutine. Just as with input variables, there can be a whole list of output variables.

Back in the main program, this is just another variable called AddEm<-result. You can use it in expressions, etc. just like any other variable:

```
x1 = x1 + 2 + AddEm<-result
```

So in summary, a subroutine is like a mini-program, with its own variables and statements. But, you can pass parameters into the subroutine, and you can retrieve variables out of the subroutine, as needed. Subroutines are really helpful in organizing and simplifying programs, so it's well worth taking the time to understand them if you don't already!

Saving object code

Once you've written a program, what do you do with it? What we're going to do here is save the object_ code, and then load and run it.

Slang Tutorial Lesson #16: (always need some multiple of 16, right?)

Compile one of the example programs above (one that works!). Once it compiles successfully, press F7 to enter the disk menu.

First, save the source code by pressing "s". Go ahead and enter some filename; the program will automatically append a ".s" to the filename.

Second, save the object code by pressing "o". You'll see the same filename as default, so go ahead and press return. If you now list the directory, you should see two new files: one with a .s, the other with a .o. You can now load and run the object code totally independently from Slang.

Exit back to the editor. We are now going to exit to BASIC: press shift-ctrl-<- (backarrow) to the left of the '1' key), and you should be back at the BASIC prompt. If you'd like to try your program from here, you can type "sys4096".

Now reset the machine. Go to the disk directory, and load your object file,8,1. Your object code is a machine language file, just like other programs you may own or download. Once it's loaded, type "sys 4096" to run it. (By default, Slang programs are located at 4096, but it is easy to locate them elsewhere, such as 32768 or 49152, or even make it so you can RUN them from the BASIC prompt.)

When you're done, type "sys 54016" to return to Slang.

And that's it! You should now be able to write, run, and save Slang files and be well on your way to writing new programs. If you have any problems or questions, don't hesitate to write me or to post to the forum!Good luck!

(Editor's Note: Stephen Judd noted Slang probably wouldn't work with RAMLink.)

Stephen Judd sjudd@ffd2.com

COMMVEX, July 2006

The July 29-30 Commodore Vegas EXPO v2 is already in the planning stages. The Plaza hotel has a block of 10 rooms set aside for the cause, and is offering discounts with rates at $61 per night Friday and Saturday, with $31 for Sunday night (all prices include room tax).

To reserve your room, call the Plaza Hotel at 1-800-634-6575 and be sure to mention you are attending the event COMMVEx in order to receive the discount. If the rooms are not booked by at least one month in advance of CommVEx, then they are released back into the "market". The Plaza Hotel has very nice, comfortable rooms; their website is http://www.plazahotelcasino.com.

Shuttle buses and taxis are available for those who need transportation from the airport. For those who cannot afford the shuttle bus or taxi, we will try to pick you up at the airport, as long as you give us plenty of advance notice.

More details, if available, will follow on hotel, airport, and transportation in our July CML. Stay tuned.

(Information from Robert Bernardo rbernardo@iglou.com).

FROM THE EDITOR:

By: Linda Tanner

Editing this issue was interesting to say the least. Beause of miscalculations and miscommunication on my part, this issue ended up being co-edited by myself, and **Richard Savoy. Andrew Schwartz,** who was originally slated to edit May, and will edit our July CML instead.

Pages 1,2,6,7,8,15,16, and 17 were created using a C128 system ,WHEELS, PostPrint3, and Lexmark Optra 40 printer.

Pages 4 and 5 were created
on a C128 system by **Jane M.,Jones.**

Pages 3, 9, 10, 11, 12, 13, 14 and 18 were created by **Richard Savoy** on his system consisting of C128, Panasonic KX-P2135 HP Desk Jet printer. And his new toy H/P Office Jet 600 Combination Copier, Text Scanner, and other goodies.

STILL AVAILABLE TWO FREE BONUS DISKS WITH EACH ISSUE OF THE NEWSLETTER YOU GET ONE OF TWO DIFFERENT CLUB DISKS

Just send your check for $ 3.00 made payable to: Emil Volcheck and make a note that the check is for postage for bonus disks. Mail to our Treasurer:

Emil Volcheck Jr.
1046 General Allen Ln.
West Chester PA 19382-8030

(You will get a total of 12disk this year)

TABLE of CONTENTS:

By: Richard Savoy

The Next Editor

Our July 2006 Editor will be Andrew (Andy) Schwartz who will accept and Commodore format files. If file is short, it may be hard copy. Andy's deadline for accepting articles and other submissions is June 15. Send to 6300 King Louis Drive, Alexandria, VA 22312, or to andyschwartz@cox.net.

Who Said That?

"A drama critic is a man who leaves no turn unstoned, 11
George Bernard Shaw

"/ have spent a lot of time searching through *the Bible for loopholes."*
W. C. Fields

"/ would rather sit on a pumpkin and have it all *to myself than to be crowded on a velvet cushion. "*
Henry David Thoreau