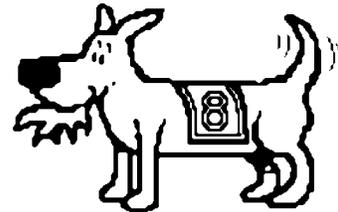


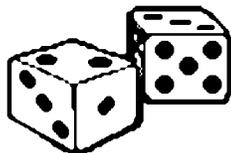
The Zero Page

*The newsletter of the Commodore Users of Wichita
For the sharing, learning, and love of Commodore computers"*



Eight-bit keeps takin' a byte out of Commodore computing

Number 57, July 1998



Random Access

by Dale Lutes

You often hear Commodore users referring to their systems as 8-bit machines. Our club mascot is a little

dog named Eight-Bit who "keeps taking a byte out of Commodore computing." What are these bits and does the number of them really mean anything? I'll try to explain.

A bit is the smallest and simplest piece of information that a computer can deal with. A bit represents one of two values and we can think of a bit as being off or on, false or true, clear or set. In the earliest digital computers, a bit was represented by the state of a relay or switch that was open or closed. In today's computers, a bit is represented by a voltage in a digital circuit that is either low or high.

A bit by itself is of limited value. If you wanted to use one to represent numbers, you could only count from zero to one. But by grouping bits together, the number of values that we can represent increases at an extraordinary rate. In the 1960s and 1970s, most computers operated on bits in groups of eight and the term "byte" was used to refer to a group of this size. A byte allows us now to represent numbers from 0 to 255. By associating numbers with printable characters, a byte can be used to represent all of the letters of the alphabet (both capitals and lowercase), the digits from 0 to 9, thirty odd symbols for punctuation and math, plus a dozen or so special codes. What I've just described is the ASCII character set and there are still 128 codes left over that can be used for graphic or foreign-language characters.

Our Commodores are designed process data one byte at a time. The most basic computer operations (moving data from one location to another, adding & subtracting, testing bit values, and others) all operate on bytes. Remember how a byte can only represent numbers as high as 255? When our 8-bit computers need to add

larger numbers, they must do it in several steps: The least significant bytes are added first, then carrying part of the result into the addition of the next bytes if necessary. This isn't as complex as it sounds. In fact, we humans learn to do it in elementary school mathematics. Consider the following addition problem:

$$\begin{array}{r} 17 \\ +15 \\ \hline \end{array}$$

We are taught to start with the least significant digits, adding the 7 and the 5. The result is 12. We write down the 2 and carry the 1

into the next step. This is exactly what happens in an 8-bit computer that is adding two 2-byte numbers. What does this imply when we compare our 8-bit systems to computers that use 16, 32, or even 64 bits? If you think about it you'll see that an operation such as addition can be performed in fewer steps on 16/32/64-bit machines which means everything runs much faster.

16, 32, and 64 bit computers are common nowadays. Those of you who are good with math will realize that these numbers are all powers of 2 (and multiples of the 8-bit length of a byte). While this is convenient, it is not absolutely necessary for a computer to be designed this way. In fact, one of the first computer systems I worked with was a 36-bit

design. But that's another story and another column...

Bylaws Changes Approved

by Dale Lutes

The members in attendance at the June meeting unanimously approved the three bylaws changes that were outlined in last month's newsletter. The updated bylaws will be reprinted in full next month.

I want to express a special thank you to Susan Bales for assembling this newsletter during my absence. I am looking forward to seeing all of you again in August. --Dale

Stupid Programming Tricks

by Dale Lutes

Last month, I presented a FORTRAN program written by Brian Pendleton of Edinburgh University and asked you to predict the output. No, I didn't get it either. But with the help of my friendly debugger, I was able to figure out why it does what it does. For the most part, rambo uses spaces to disguise variable names as well-known FORTRAN keywords. It also takes advantage of the fact that words such as END and READ are not reserved in FORTRAN as they are in some other languages. Here is the program again, this time with extraneous spaces removed. All comments are mine to help make it clearer (I hope):

```
program rambo

C   Some variables that LOOK like keyword combinations.
C   Yes, it's legal to define a variable named END.
real ifinteger
integer ifreal, end
character*7 stopdimension

C   In the equivalence, it is important to realize that
C   IF7 is an integer and YGOTO100 is a real.
equivalence (if7, ygoto100)
data ygoto100 /-100/

C   The next 4 statements may look like logical tests,
C   but each is a simple assignment statement.
do20i = 1.20
if7 = 6                !Also makes YGOTO100 = 0.0
then5 = 4
20  ifx = ygoto100     !Now IFX = 0
stopdimension = '(f16.6)'
```

C Here's a long variable name disguised as a sequence
C of FORTRAN keywords.
thencallgotowrite = 20
elseformat = 7.2 / do20i !ELSEFORMAT = 7.2 / 1.2
end = 6

C The next line is equivalent to:
C read = 0 + 6.0 - 4.0 + 1.2
100 read = ifx + elseformat - then5 + do20i

C Write the answer to unit 6 using the F16.6 format
write (end, stopdimension) read
end

And the result is: 3.200000.

I hope you had fun with this program and the C trick that I shared with you in April/May. If you know of any other "stupid programming tricks" please send them to me. My paper and e-mail addresses are listed elsewhere in this newsletter.

In natural science, Nature has given us a world and we're just to discover its laws. In computers, we can stuff laws into it and create a world.

--Alan Kay, quoted in *Out of Their Minds* by Dennis Shasha & Cathy Lazere, (c) 1995, Springer-Verlag New York, Inc.



CUW Member Phone List

This list is presented for the convenience of CUW members. It is not to be reprinted in any other publication and is not to be used for commercial purposes. It is not a complete CUW membership list. Only those who have agreed to allow us to use their phone numbers are included. The numbers listed below are subject to change without notice.

831-0008	Bales	Robert
531-2560	Benoit	Virginia
943-6725	Frisbie	Bob
721-0835	Lutes	Dale
265-2560	McManamey	Don
777-4634	Phillips	Galen
776-0735	Poston	Carl
838-6819	Salts	Charles
776-2683	Shook	Jerry
776-2683	Shook	Mike
529-0039	Shouse	Moe

The Zero Page is a monthly publication of the Commodore Users of Wichita. The opinions expressed here are those of the authors and do not necessarily reflect the views of the CUW. Unless otherwise stated, articles in this newsletter may be reprinted by other Commodore user groups without permission provided that the author and the CUW are given proper credit.

Meetings of the CUW are scheduled for the second Saturday of each month. The deadline for articles is 14 days prior to the meeting day. Submissions are accepted on 1541, 1571, or 1581 formatted floppy disks. geoWrite, ASCII, or PETASCII files are preferred. Microsoft Word documents (Mac or IBM compatible) on 1.44MB floppies are also accepted. In a pinch, paper hard copy will work. Call Dale at 721-0835 or mail your articles to:

Commodore Users of Wichita
c/o Dale Lutes
11102 W. 17th Street
Wichita, KS 67212-1187

E-mail: lutes@feist.com



Don's Digest

by Don McManamey

It was just about time for morning coffee break on Monday, 8 June when an explosion ripped through the DeBruce grain elevator just outside the Wichita

city limits. Radios are against the rules where I work at Raytheon Aircraft but nevertheless, word of the disaster quickly filtered through the plant. Information was sketchy and sometimes contradictory. All we really knew was that it was not good. The next morning I happened to pass by a newspaper lying on a work bench. The paper told of the explosion on the front page. Realizing the time needed to put together a paper and then distribute it, I wondered what the latest news was.

Toward the end of the day I passed by the same paper and contemplated what might have transpired since that morning. With that, a thought entered my mind. It was not a new thought but one in which I wonder, has anyone really given any serious consideration. Is anyone working on it? In the age of radio and TV it is a wonder that any newspapers can survive. The fact that radio and TV are limited as to what they can give is certainly in the favor of the print industry. Just one hour of picking and choosing can give most of us what we want from the paper. With broadcast, a moment of inattention might mean we lose, perhaps forever, the one piece of information we were looking for. Fact is, I listened to the radio for forty five minutes trying to gain the latest on the explosion.

What might the newspaper of the future look like? To answer this question, let us look at the technology available today no matter what the cost. I envision a device only slightly larger than a standard sheet of paper (8.5 X 11). My reason for choosing this size is that it is small enough to carry around with out being cumbersome but large enough to carry a fair amount of information including pictures. I also see it as being about an inch thick. Nearly the entire front surface would be a color LCD display. The data could be received on a small cartridge like a Game Boy uses or via a phone link.

The idea I like best is more like a pager or cell phone. It could be ground based or from a satellite. Your paperless paper might be purchased, leased/rented or some other arrangement. However this is done you could carry your paper around all day long and have it continuously updated with time and date stamps and perhaps a flag to make you aware of a change. You could mark articles to be preserved so the update would not erase them. The paper might also know that you looked at an article and tell you that it had been updated. Information on breaking news could be as timely as the radio or TV with the time to type the article being the biggest factor on speedy transmission. Who knows, the way things are going, a reporter may talk into his/her cell phone and a computer on the other end immediately transcribe the information and transmit it out to you.

Now some might argue that you can't clip an LCD screen and paste it in your scrap book. With a download to your PC or to Kinkos, you could hold in your hot little hands a print out of any article or page for your scrap book or to mail off to your sis' in Georgia. With the virtual elimination of the printing press, typesetters, paper, and the distribution process, the cost of the paper might get so low that advertising alone could pay for your paper and you could receive it

without cost in the same way nuclear reactors were going to reduce the cost of electricity to the point where everyone would pay a low, flat fee (OK, so that prediction didn't come true and your paper will probably always cost you something). Now, what if the paper wants to provide internet service? This same device might be able to handle this as well.

One might argue that a device of this size cannot provide that much memory or that the power requirements would be too great for the batteries it could carry. Keep in mind that these are two areas where things seem to change every day. Memory chips with greater capacity and lower voltage drain. New battery types along with solar cells. I remember my first solar calculator. If it wasn't in bright light it just wouldn't work. Today I use one at work that can operate well when left in my tool box drawer and the solar cell covered by the drawer above. At one time it was thought that man could not fly. Then we thought the speed of sound was a barrier we could not break. We seem to be able to do the impossible when we set our minds to it.

P.S. About three weeks after making the rough draft of this article I saw a program on TV where a device was shown that resembled what I have described. It was a "Soft Book" which was supposed to be used in place of a printed book. With new titles downloaded from the internet the choices could include books that are considered too risky for the printed page. Perhaps more than ever before, we are limited only by our imagination and perhaps our pocket books.



Moe's Soap Box

by Moe Shouse

In my last column, I said that a new source of info on the newer stuff for the Commodores was a new newsletter called CPU. Well it's not really new. Mr. Tom Gosser in Shelton WA has done other issues, but he had trouble. Someone broke into his office to steal his very

expensive laser printer, and then set fire to the place. That was bad enough, but the local enforcement confiscated all that was salvageable to help in the investigation. To make a long story shorter he got it back in about a year and a half. But he didn't have his subscribers list, so he had no way to tell them. He did get his stuff back, but still thought he didn't have his subscribers list. Then he found it, and in 9 weeks put out the magazine. The postmark on my copy is May 25, 1998.

This issue is number 4 volume 1. It is a very good newsletter. On page 4 is a long and good review of NovaTerm 9.6 by Remington Steele. He even covers some minor problems, but he thinks it's more of a hardware thing. Also on page 8 is a list of how some different programs work with the Super CPU-128. So here is a run down on just a few: Maverick will boot only with CPU disabled or in normal mode. PrintShop must be in normal mode to boot. Switch to turbo after it's up and running. PrintMaster+ is turbo compatible, even during boot up. Fun Graphics Machine boots up in turbo mode. Disable FGM fast load if JiffyDOS is on. FGM runs super fast! A great performer with the SuperCPU! I don't have room here to list his chart but with GEOS this SuperCPU really screams. A RamLink to Hard drive copy at 1 MHz -- 9 minutes 31 seconds; at 20 MHz -- 59.33 seconds.