

Power Play

This month I have some news that I'd just as soon be quiet about, but in the interest of spreading the word I'll sound off. It appears that there is a gentleman from this area who goes by several names, Col. W. W. Winthrop being one of them, who is in the business of mail fraud. His most recent venture (adventure ?) is with World Power Systems, of Tucson AZ. World Power Systems was evidently based in an apartment at 1161 N. Eldorado Place, "suite" 333.

They advertised heavily in the various hobby computing magazines that they were in the business of providing add-on disk drives, serial and parallel interfaces, S-100 converters and the like.

What they did do was cash a lot of checks. Total, complete fraud, folks. If you were contemplating sending them any money, save yourself the trouble. If you have sent them some, contact the Tucson District Attorney's office. At this point there is little that can be done except chalk up another one to experience. As a normal sort of thing, I tend to be very slow in criticizing other companies because that is a good way to do a lot of harm to a company that is trying their level best (besides, a lot of fair criticism can certainly be pointed our way). However, (you were expecting a 'however', weren't you?) the current state of trade for quite a few companies in the home computer peripheral market is such that C.O.D. terms are in order. If a company wants more than \$25.00 or so, and won't ship C.O.D, make sure they're an established company (like CLOAD magazine!).

Enough. The personal computing magazines will be carrying the World Power Systems story in in greater depth with higher accuracy, so I'll depart here and talk about Voice, another great one from the author of Zarbor. This is a machine language program, so you level II'ers will have to load it with a SYSTEM command. The sequence is: SYSTEM (computer responds with < *? >), V (for Voice), (computer responds with < *? >), then /17408. Level I tapes load with CLOAD as usual. This program looks at the cassette in port (the black plug) and puts a star on the screen if there is enough voltage there to trigger the circuit. It then advances the screen pointer and looks again. Sort of a raster scan oscilloscope, whatever that might be. Its formal name is "Voiceprint", which it got because if you press "Play" and "Record" down at the same time (you'll also have to insert a scratch tape), your recorder acts as a microphone and puts your voiceprint on the screen. We've tried to use it as a lie detector with little result, though it is kind of handy for looking at tape output. If you run Voice and then play data (just as if you were to load a program), the audio on the tape is visible on the screen. As you adjust the volume up from the low end, you'll notice that there are some diagonal stripes with intermittent spots between. The stripes are the framing pulses; they should form fairly straight, continuous lines. The spots between are data pulses; a spot corresponds to a binary "1". The optimum point seems to be the lowest volume where both framing pulses and data pulses are two stars wide. Dropouts in the tape show up as places where the stars go to single width (or disappear). Flutter and wow show up as wavy lines of framing pulses. The TRS-80 doesn't seem to be sensitive to this, though it looks (and sounds) strange. This program shows that ingenuity, combined with a working knowlege of the computer's innards, can create an interesting program with very little code (about two seconds of tape).



Labyrinth, a level II BASIC program, needs the memory size set to 31741 prior to loading (it contains a "user" function).

While we're loading programs, here's a note from Dave Lagerquist, our single-celled example of alien intelligence:

"There are some complaints coming in from the peanut gallery about my great idea of putting programs back - to - back on the cassettes. Why a genius programmer such as myself should be driven to explain my actions to the public is something I cannot understand, except for possibly getting paid next payday. There are actually three reasons, any of which is more than sufficient. One, some people are not aware that there are two recordings of each program on the tape, and the only way to remind them is to have them trip over the second copy every other CLOAD. Two, seeing as how I am the Quality Control Inspector around here, as well as editor, tape duplicator and janitor, I have set up the first of each pair of programs as "A" and the second as "B". This allows me to CLOAD "A" and then CLOAD "B" to doublecheck the tape. Running back and forth on the tape to find the "other" copy does not cut it when the duplicating machine is throwing new copies at me as fast as I can dodge. Three, the format appeals to my higher artistic instinct and basic human laziness in setting up the master tape."

Current negotiations with Dave have settled the issue with level I, however. We will go back to the "old" format of sequential copies, repeated on the second half of the level I side. I'd like to continue the subject of quality control for a moment, explaining the concept of a checksum.

As quite a few of you are painfully aware, the most likely source of errors in computers involves the process of saving and reloading the contents of memory. This is true of everything from mag - card calculators to Cyber 176's. There are two places where these errors can occur. The first is in saving the data, where the reason might be dust on the magnetic recording head or a bad section of tape. These are called "hard errors", because they represent an absolute loss of data (if it isn't there, there is no technique that will recover it). The main defense (for cassettes) against hard errors is to record the data twice, and trust that the same thing doesn't happen again. The second place for errors to creep in is in reading the data back. These are "soft errors", and are (fortunately) the more prevalent of the two. Soft errors can be fought with tape head demagnetizers, head cleaners, volume control jockeying, and sincere cursing.

Whether a load error is hard or soft is immaterial - if it is undetected, it becomes a bug (usually, though not always, obvious). Several techniques have evolved for verifying that a given load may have been successful. The first is that if a program runs, it is probably good. The problem is that an error in a little-used section of code can go undetected for some time, creating invidious bugs in software which has previously been shown to be reliable. Another technique, much more appealing, is the checksum. A checksum is an old tool of bookkeepers, which is best illustrated by example: the checksum (base ten) of 6, 7, 5, 8, and 5 is 1. The true sum is 31, but with checksums the "carry" is thrown away when adding. There is no single number that can be changed to any other (single digit) value without affecting the checksum (including the checksum!). If one digit in the sequence is in error, there is a one - in - ten chance that another error will cancel it out (again, including the checksum itself). If we treat a program or data file as a sequence of numbers (base 256), and use a tolerably reliable recording technique, the chances of an undetected error get very close to zero - one chance in several million for a short program or data file (buzzword aside: some larger computer systems use a "CRC" - cyclic redundancy check. It's sort of a superchecksum).

All level I format recordings end with a checksum, and the CLOAD command computes a checksum when loading. If the checksum on the tape does not agree with the checksum computed during the load, the CLOAD command returns with a WHAT? and lets you know about the problem. All so nice. Oh, if level II only did likewise! Level II BASIC programs have no checksum, and the recording technique is more touchy to boot. The way to verify a good load in level II is by the CLOAD? command. What this does is read another copy (or the same copy rewound to the beginning) and compare it with the previously loaded copy. Obviously, if the CLOAD? is used with the same copy, a hard error will pass through undetected (identical loads - both bad). Fortunately, all the other formats allow a checksum, as does the disk system - level II cassette BASIC appears to be the only one short of this feature.

As we "print" the magazine, we simultaneously pull sample cassettes and test them. If the amount of time we spend testing each sample is halved, the number of samples is doubled. As it is, the duplicator has us beat by an easy 100 to 1 - thus the push to verify them as fast as possible. If we were verifying by listing/running, the ratio would be more like 1000 to 1. The only way to check level II programs, therefore, is the CLOAD / CLOAD? sequence on separate copies, and the fastest way to do that is to have them back - to - back on the cassette. If we get a number of letters which extol the inherent reliability of our product and request the return of the previous format, I'll re-open negotiations with Dave. As for now, when the subject is raised, he crawls back further into his cell, snarls, bares his teeth, and brandishes his rusty knife.

Bug time - a reminder that some level II ROMs have a bug that RESTORES the DATA pointer after every READ statement. One of the programs that finds this bug is our front cover, and it it finds it (naturally) by not working. A POKE 16553, 255 will fix everything, and it will be in our cover program starting with the June issue.

For you high rollers out there with the disk systems, our Sketch program (December 78 issue) has a USER(0) command that is overwritten by the disk operating system. We're looking into whether or not there's a simple patch that will take care of this.

And before we get off the subject of software, let me preview a fantastic upcoming attraction from the folks who brought your personal computer its own personal magazine. Yes folks, the Best of Cload (Volume 1) is about to hit the duplicator - a compendium of all the best software in our first six months of trying. Appropriately enough, it will be ready to go well after the second six months of trying has taken place, and well into the third six months (still trying...). What we will have is a cassette tape (you expected an 8-track cartridge, maybe?) with both level I and level II versions of the best - a Flags cover, Break, Sand Castle, Pool, $Y=mx+b$, Knight's tour, Pinball, Yield to Maturity, and Jukebox. Included with the package will be hardcopy listings of all the programs along with commentary on some of the programming techniques used in each. This is a service to those of you who just joined us, and a disservice to those of you who have been waiting for us to reprint these in the current issues (most of the original versions were in level I only). That's OK, though. The real stinger is that we are going to charge \$10.00 a copy for this package (plus 60 cents in CA), partly because it is a more substantial package to put together (we're even putting it in a binder) but mainly because we're getting greedy in our old age. It's coming out in June, so come one, come all, line forms to the left, have your money ready. Those who want early copies please send early money and trust that we don't skip to Brazil with it (move over, Colonel W.W.).

For this month's hardware talk let's cover a device called a relay. A relay is a remotely actuated electromechanical switch. Fine. What's a switch? Well, when you turn the room lights on, the traditional method is to walk over to the wall switch and flip the little lever up. What happens then is that there are two pieces of metal which come into contact with each other. All metals conduct electricity, so when there are two pieces of metal touching, electricity can flow from one piece to the other. Flipping the wall lever down does the opposite. The two pieces of metal separate, and electricity will no longer flow from one piece to the other (except for the power switch on this computer I'm typing on - it went up in smoke 15 minutes ago). Controlling a switch, therefore, controls the flow of electricity to a device (a light bulb, television, or if you must, a coffeepot). Which leads us to a special switch which can be controlled by our computer. A relay is a switch which is controlled by an electromagnet. There is a coil of wire wound around a bobbin inside the unit, called - logically - the coil. If a small amount of current flows through this coil, a magnetic field builds up and attracts a piece of metal towards it. This piece of metal, in turn, contacts another piece of metal and we're back to the same point as before. Radio Shack sells such a device, catalog number 275 - 003, which is appropriate for our use. Its coil is set up for 12 volt operation and draws about 10 milliamps when on (within the driving capabilities of an ordinary chip). Its switch portion can control up to one amp of current, not enough for a coffeepot or heater, but plenty for a radio or small lamp, or for controlling another relay large enough to control just about anything. I'm out of space for this issue, but next issue there will be a complete schematic of everything set up so far, complete with enough circuitry and software to input a switch position and control a relay with a digital output.

Relay

.....
 FULLER ELECTRONICS IS PROUD TO ANOUNCE THE RF-III

In the tradition of a quality product at a reasonable price Fuller Electronics introduces the RF-III. In short, the RF-III will allow you to connect your TRS-80 to the outside world without endangering you computer.

WHAT WILL IT DO?

These are only some of the uses you can put the RF-III to:

>Control lights at random times, to look like you're home when you're not.*
 >Turn on and off you video recorder at the precise second you desire, and set it up months in advance, if you wish.* >Control photographic enlargers to the second.* >Control your microwave oven to start at the moment you desire, then stop if you don't make it home in time.* >Use the internal beeper to signal the end of a long sort. >It will signal you if an error has occurred in your program. >Lets you know with one beep, or two or what ever, exactly what part of the program your computer is in at any given moment. >Check your RS-232 output without turning on the printer. >Use your TRS-80 as an automatic telephone dialer. (subject to local regulations)
 >The RF-III actually protects you computer's internal relay with its own relay. This means your computer only sees 1/10 the current it normally does. If you are using the CTR-21 or an "off brand" recorder you may be overloading the internal relay right now.

Complete and full documentation is supplied with each RF-III. This will give you instructions and diagrams on how to use the RF-III to perform each of these applications.

*NOTE Each application marked with a star requires additional parts available at your local Radio Shack store. These parts are listed in the documentation with stock numbers.

The RF-III is limited only by the imagination of the user.

FULLER ELECTRONICS
 7455 HOLLISTER AVE. STE 232
 GOLETA, CA 93017

VISA & MC ACCEPTED
 \$39.95 PLUS \$2 SHIPPING
 7-DAYS/24-HOURS PHONE ORDER NO. 800-638-2000