

# Winning the Micro Game

---

*Dilettantism won't do when it comes to learning about microcomputers.*

---

Don Lancaster  
Synergetics

**H**ands on is everything. The only way to ever learn anything about computers is to jump in with both hands and feet, get on line and do some computing. Until you actually do and see what the micro world is about, you've accomplished nothing. You must do things yourself, on your own terms, in front of a working, real computer, alone.

It's both funny and sort of sad to hear a student say he just took a DP course but couldn't get any CPU time. *He* got taken, not the course.

You become computer literate by using computers, not by having someone tell you about them or by reading about them.

Understand a timing loop by writing one and watching it work. Do an interface by taking a triac, an optocoupler and a 100-Watt light bulb and shining light on the real world. Find out what an interrupt is by interrupting a computer. Do it—yourself.

*You have to make mistakes.* If you are learning micros or developing any new product, half your experiments should fail. A canned set of exercises on a micro trainer is next to worthless if everything falls into place and works perfectly the first time.

In the micro world, you make mistakes to learn and to progress. You should expect mistakes. Prepare for them. Welcome them. Aggressively seek them out.

Of course, it makes sense to never make the same mistake twice. Build on what you have. To expand your microcomputer universe, try new things that may fail. Find

out why they fail, and use this as a newer and bigger base to work from.

Usually, you are never anywhere near where you think you are in solving any hardware or software problem. Unexpected surprises and plain old stupidity are always between you and reality. If you think you have something working perfectly, you probably don't even understand the problem.

*You must mix hardware and software.* Some heads-in-the-clouds pure software people out there still believe that hardware is a mundane inconvenience standing between them and pure "computing." And there are technician types who do everything with bushel baskets full of integrated circuits.

Neither approach is good. Sometimes a simple and inexpensive hardware circuit can replace bunches of software. Other times and other places, a few lines of elegant software can eliminate the need for custom circuits or a special device.

Winning computer products will combine both hardware and software, using the best features of each to give you the simplest system and the lowest possible cost.

This means that if you are a hardware person, you should learn programming and learn it fast. If you have a software background, start soldering and wire-wrapping with a vengeance.

Synergy says that  $1 + 1 = 4$ . This is definitely the case when you get an optimum mix of hardware and software interacting with each other.

Neither can stand alone—not any longer.

*The real world is fuzzy.* Some textbooks and lab experiments work every time.

Everything is nice and clean, neatly tied up. You do exactly what you need to do the job, no more, no less. Unfortunately, reality doesn't work that way.

First, you must deal with people, and that will always mess things up. Key items will be missing or late. The magic chip may be a figment of an ad writer's dreams. Or a problem may have a simple and inexpensive technical fix that is politically or socially unacceptable. Goals conflict. So do egos.

Expect and accept fuzziness. As you get into a new computer area, things will start out completely confusing. Then they will become fuzzy. Then they will become, for a glorious instant, crystal clear. Then, of course, they get fuzzy again as you become more involved.

As you go to the bigger picture, expect more fuzziness. Also recognize that there really isn't much in the way of real-world beginnings and endings. Rather, things sort of dribble off into the great whatever.

Micros might—just might—be the missing link between people and intelligent life in the universe.

*Hit the basics hard.* Any 6502 micro freak can sit down and immediately "prove" that the 6502 is ten times better than any other micro in the world. The trouble is that you can do the same with any other micro family, as well.

For most micro uses, it makes no difference which micro from which family you use. Even if there temporarily was a "best" micro, other factors such as your own skills and attitude, the available software, the elegance of your competitors' programs and so on will reduce any advantage of the "best" micro to zilch.

If you don't happen to like the "best" micro, just wait a month or two, and it will get shot out of the saddle by something much more promising.

This all means that the micro you learn is not the micro you will use. Later on, there will be much better ones to work with, and they are sure to have completely different tech details.

To beat this, hit the basics hard. All known micros have address space and addressing modes. All have interrupts, subroutines, clocks, ports, memory and I/O. Use any micro you like to add tech details to the fundamentals. But get the essentials down solidly.

*Reach out and put the touch on someone.* The nickels in the micro world are now to be made in places where people are not yet using micros. Find these places and get involved with these areas and people on their own terms.

Put micros to work feeding cattle, treating sewage, gambling on Wall Street, designing looms, mixing cement, baking calzones, milking goats, hulling pecans, questing tinajas, animating video, co-oping groceries, hybridizing sinsemilla, improving wood stoves, redesigning bicycles, restoring steam calliopes, monitoring steam gauges, selling paper clips, cutting dress patterns and teaching trumpets.

When you do reach out, always work in the other person's terms and language, bending the micro info to fit as you can. If they are smart enough to learn micros, they won't need you for anything.

Find places where they don't yet know that micros can help. Then jump in.

*Don't reinvent the wheel—steal the plans instead.* Much of the needed and obvious micro-related information has already been done and is readily available for your use. For instance, if you want to drive a Teletype or another printer, use someone else's driver routine. Don't stop what you are doing and invent your own—unless you truly want to know how a driver program works.

Scads of Morse code trainer programs are out there. Why write another? The same goes for sorts and word justification subroutines. And there are more versions of Lunar Lander than there are moons in the solar system. How many Hangman, Hex-pawn or NIM games have you seen?

Now, if you want to learn these programs, that's fine. but, if your goal is using something, rather than creating something, find out what has already been done and go with it, or improve it and then go with it. Refer to monitor listings, user software exchanges, micro magazines, application notes, club newsletters, program books and micro information exchanges for programs to use.

*Better wrong now than right later.* In

anything you do in the micro world, your first attempt will be wrong and will have to be reworked. So, immediately kludge up your first attempt and let your mistakes show you the way to go. Often you don't even understand what the real problem is until you are inside a program or a wire-wrap board looking at it.

Try a simple, quick and dirty tactic that at least sends you in roughly the direction you want to go. Make some guesses. Take a stab at it.

In your early attempts, if it works, use it. Start your project flying more or less right-side up. Later on, you can go back and add structure to your programs, elegance to your methods, convenience to your user and simplicity to your hardware.

Add the final spit-and-polish on the way out the door, and not early in the game.

*Write it down.* And not on the back of an old envelope, either. Documentation is the

aren't supposed to smile while you are playing their games.

Simon says don't smile. It's still a game. Have fun.

*You will never get enough.* No matter how far you have gone in microcomputing and no matter how much of what kind of hardware and software you have on hand, you will always "need" more of something.

More memory? Start with a 1K trainer, then 4K, then a 16K micro. Then overfill the 16 megawords of an extended micro space. Need hard copy? Start with Excedrin headache number ASR-33, then on to thermal, a Selectric and finally a daisywheel.

Now, if only daisy was intelligent and had its internal word processing.

From plain-jane video, go on to graphics, color graphics, hi-res and then super-resolution color with gray scale. From cassettes, it's on to floppy, dual floppies, quad density and then a Winchester.

---

---

## You can learn far more about micros watching fourth graders zap Klingons than you ever will in a university COBOL course.

---

---

password to avoid self-destruct modes. You don't record only final programs and schematics. Instead, keep track of what you did and why you did it, neatly, and in some semblance of order.

Keep accurate records of where you have gone and where you are heading. Put together the files of literature, instructions, op codes and program bits and pieces.

Software is worthless if you can't show someone else how to use it. Hardware has no way to operate if there's no way to connect it or fix it. You must be able to go back and reuse or modify what you did a week or a year ago.

Documentation is not just a hex dump; it's a micro way of life. You cannot survive without it.

*Don't separate work from play.* Which of these is more important: Designing an efficient sort algorithm for a business general ledger program or figuring out what to do with the oily slime in Adventure?

In the long run, the oily slime is far more important because that's what is stimulating interest in micros and making people computer-literate.

Any program run on a computer is a game! However, stuffy institutions, banks, bureaucrats and other so-called "serious" computer users have rules that say you

There never is, nor will there be, a time when you have "enough" of anything. What looks like a light at the end of the tunnel is a train speeding towards you.

You will find only one way out of the "more" syndrome. Always go with what you have. Make it work. Live with it as long as you can. Force it to pay its own way.

Make it do. Use it up. Wear it out.

*If it's old line, stomp on it.* Some pre-micro people and institutions are still kicking around the lunatic fringe of the new micro world. They persist with large, bureaucratic, centralized, insanely priced and unavailable megacomputers run by an elite priesthood singing the incantations of an arcane language. They completely fail to recognize the power of the micro as a highly personal, one-to-one, decentralized, inexpensive, interactive and individual convivial tool.

You can learn far more about micros in twenty minutes watching a pair of fourth graders zap Klingons than you ever will in a university COBOL course.

Old-line conventions include IBM, batch processing, COBOL, decollators, Honeywell, key-to-disk, FORTRAN, keypunches, centralized billing and data encryption. They are without any redeeming social value. They had their chance and blew it. We

Designed for your . . .  
**TRS-80™**  
 the **PHOTOPOINT™** Light Pen\*  
 a whole NEW concept in  
 computer application

- Just plugs into your TRS-80 with disk or without! (Does not void warranties.)
- Programs with 3 lines in Basic!
- Comes with two Programs
- Complete Instructions!!
- Just point to play!
- Often eliminates confusing keyboard from games, education, or multiple choice.

All you need to get up and running the same day you receive your PhotoPoint is included. For only . . .

**\$19.95** Order NOW from  
**Complete** MICRO MATRIX  
 P.O. Box 938  
 Pacifica, CA 94044

\*Dealer inquiries welcomed!

✓ 66

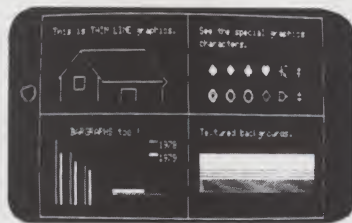
## VERONA™

If you are an 8080 Assembly Language programmer, VERONA will help you write better programs in less time. VERONA is a terminal oriented, dictionary based, user extensible assembly language programming utility. Use VERONA to quickly produce programs for software design checking or for hardware troubleshooting. VERONA puts the full power of the processor at your fingertips, giving you an ability to interact with your system that you never had before. For ISIS, CP/M, and others. From \$40.

FBE Research Company Inc. ✓ 206

P.O. Box 68234 • Seattle, WA 98168

## LOWER CASE & GRAPHIC SYMBOLS GENERATOR KIT FOR TRS-80™ CG 16 ..... \$94.50



TRUE 2 LINE DECENDER LOWER CASE. ELECTRONIC SYMBOLS. THIN LINE GRAPHICS. GAME SYMBOLS. TEXTURED BACKGROUNDS AND MANY MORE DEMO CASSETTE IS INCLUDED FULLY ASSEMBLED WITH DETAIL INSTRUCTION FOR EASY INSTALLATION REQUIRES ELECT PENCIL TYPE LC MOD OR ORDER MEMORY AND SWITCH KIT. SMK. FOR \$18 50

ADD \$2.50 FOR S. & H. CALIF. RES. ADD 6% SALES TAX SEND CHECK OR MONEY ORDER TO

**G.P. ASSOCIATES** ✓ 187  
 P.O. BOX 22822, SACRAMENTO,  
 CA 95822



### RS232C Computer compatible Paper Tape Transmitter/Model 612

Stops and starts on character at all speeds, uses manual control or X-on, X-off 90-260 volt, 50-60 Hz power. 50-9600 baud, up to 150 char/sec synchronous or asynchronous; gated internal or external clock, RS 232C, current loop or parallel output, reads 5-8 level tape, 7-11 frames per character, even or odd parity. Desk top or rack mount.

**Addmaster Corporation, 415 Junipero Serra Drive, San Gabriel, CA 91776, (213) 285-1121, Telex 674770 Addmaster SGAB** ✓ 261

tried it their way and it didn't work.

Old line not only fails to see the problem; they are the problem.

Always ask, "Why are you telling me this?" The useful products and ideas in the micro world are not heavily advertised. In fact, anything genuinely useful takes a lot of time and trouble to nail down.

If a micro is widely or heavily advertised, it more than likely means that something much better is available elsewhere. If someone is radically trying to convert you to his microprocessor or his way of doing things, the chances are he has drifted into right field and become snookered into a bad scene. He is looking for converts to ease the pain when he is shot out of the saddle.

When anyone tries to tell you about micros, always ask, "What is the real reason you are telling me this?" Find out the motives involved. Then get a second opinion, check out another choice or find a different viewpoint before you plunge ahead.

*Nail down all resources.* It is easy to assume that formal courses and expensive, hardbound textbooks are the only way to "learn" microcomputers. In fact, these are two of the worst possible ways to become computer literate. Most of these learning aids are stillborn, hopelessly obsolete and misdirected.

Anything you can relate to that involves micros is a resource. Your first, and foremost, resource is yourself, through hands-on experience.

Other resources include micro magazines, clubs, game playing, Dungeons and Dragons sessions, micro trainers, computer stores, used wire-wrap boards, tech journals, funky books, reader-service cards, benchmarks, students, teachers, trade shows, surplus stores, computer fairs, rap sessions and swap meets.

*And most important of all, go on your own vibes.* There is no right or wrong direction in the micro world. In fact, 99.9 percent of the micro world remains unknown, unexplored and uncharted. So, if "they" insist on something, most often "they" don't know what they are talking about.

If you are interested in something and want to go in that direction, fine. Do it!

Your surest bet for long-term winning is to roll with your own vibes. Explore what you want to. Ignore the herd thundering the other way. Get off the beaten path.

Make yourself your own best customer. Satisfy your own needs and your own curiosity. Put as much psychic energy and personal value as you can in the routes that you pick, and you are certain to win the micro game.

You are, by definition, the center and the most important part of the micro universe. Don't ever forget it. ■

Send SASE for free information

FORTH is an advanced language/system for advanced programmers. MMSFORTH is a professional version tailored to the Radio Shack TRS-80 Model 1 and professionally supported by Miller Microcomputing Services.

Structured & Modular Programming Interpreter, Compiler, Assembler, Full-Screen Editor

Expandable Instruction Set w/Source Code

Graphics, Strings, Arrays, Double Precision

Very Fast and Compact, Virtual Memory

Includes 5 Demo Programs w/Source Code

MMSFORTH System Diskette (16K, 1 drive) \$64.95  
 MMSFORTH System Cassette (16K, Level II) \$44.95  
 MMSFORTH PRIMER (required for above) \$15.00  
 microFORTH PRIMER in MMSFORTH (32K, 1 drive) - an exceptionally fast, flexible and easy to use Data Base Management System \$49.95  
 THE DATAHANDLER (required for DATAHANDLER) \$9.95  
 P.I.M.S. Manual (required for DATAHANDLER) \$29.95  
 FLOATING POINT MATH and FULL Z80 ASSEMBLER \$25.00  
 Diskette (16K, 1 drive) \$19.95  
 Other FORTH literature - more detailed and advanced introduction \$6.95  
 USING FORTH PRIMER - very readable introduction \$19.95  
 microFORTH PRIMER \$9.95  
 FORTH TUTORIAL MANUAL - good on FORTH internal structure \$6.95  
 CALTECH FORTH MANUAL - good on FORTH internal structure \$19.95  
 Shipping is \$2 plus \$1 per extra book (Mass. orders add 5% tax, foreign plus 15%)

**mmsFORTH**  
 MILLER MICROCOMPUTING SERVICES  
 61 Lake Shore Rd., Natick, MA 01760  
 617.653.6136

✓ 235

**COMPUTER FORMS** ✓ 340

Distributor of Computer Paper Products

All paper products are white, blank, tractor feed, (pinfeed) fanfold continuous stock.

6" x 4" Postcard Stock (7" width with 1/2" margins) Use as is or trim for 3 1/2" x 5 1/2"  
 Pkg. 1000 cards ... \$17.95 Pkg. 2000 ... \$29.95  
 Box 4000 ... \$49.95

Standard 9 1/2" x 11" Computer Paper (8 1/2" x 11" sheet)  
 Pkg. 500 sheets ... \$5.95 Box 3500 sheets ... \$27.95  
 Box shipping weight ... 31 lbs.

Try our mini-paper! 6" x 8 1/2" sheet size (7" width includes 1/2" tractor margins)  
 Box 3200 sheets ... \$23.95 sh. wt. only 17 lbs.

Cash Order: Include \$2 for shipping, excess will be billed with your order.  
 Credit card Order: Shipping will be added to your order. Include all credit card information.

Send for FREE CATALOG of paper products. Postcard stock, address labels, many sizes & types of paper.

**Computer Forms. (616) 429-7922**  
 5588 Caribou, Stevensville, MI 49127

CHECK OR MONEY ORDER