L.E. Systems, Inc.

Manual

ROBIN –
1974 BUCK ST.
EUGENE, OR. 97405

Installing the System

Installing is quite straightforward.

1. Remove the Atari cartridge lid.

2. Remove the Atari O.S. ROM cartridge.

3. Plug the board marked LEINIT into the lowest of the 6 expansion slots on the motherboard ("T-card"), loop the cable arond the side of the board.

4. Plug the "T-Card" into the O.S. Slots.
-- ENSURE -- that the "tabs" on each side of the T-card line up with the plastic tabs in the cartridge area. They will hold tightly around the Atari tabs when the board is vertical. This is critical to operation because the Atari connectors are very sensitive to any wobble whatsoever.

5. Plug in the drive(s) into the cable, with the connector off on the long end, by itself, headed for the Atari. You may have to do this twice for the 8 drive system, as it has two cables.

The BOTTOM edge of the drive connectors must connect to the RIGHT edge of the edge-card on the LE Systems board. Generally we do this by having the red stripe be on the right side at the T-board and at the bottom of the disks being connected to.

6. Plug in your drives; switch them on.

7. Plug the cable for drives 1-4 into the middle edge connector, and the cable for drives 5-8 into the left hand connector.

8. Plug in the power supply to the Atari-like connector for the T-card. Note that some LED's may turn on.

9. Plug the supplied Atari power supply into the T-card at the socket provided. Note, please don't push or bend on the board too hard during any of these steps--it is possible to break a trace.

10. Some LED's should now light, randomly.

11. Select SD/DD on the DIP switches. "ON" means double density; "OFF" means single density. You must select this for 1-4; 5-8 are always SD.

12. Turn Atari on.

An LED should light up, in the left hand column, for each drive you have online. If you have D1 as an LE drive, for instance, the topmost light will light up. The LED's on the rightmost row will reflect SD/DD status; a lit LED means DD.

If you have one of the LE drives assigned to D1, the system will now try to boot off of it. The drive will switch on. Put any

Atari disk that boots in it, close the door, and you're running.

NOTE: PERCOM mounts their drives upside down. With theirs, you insert a disk with the labels to the left. With ours, to the right.

You can check disk operation at any time by pressing the OPTION and SELECT keys at the same time. Displayed at the top of the screen will be:

1. Command #. The DOS command.
2. Unit #. Drive #.
3. DD/SS. Single or double density.
4. Sector #.
5. Track #, Sector # if we decode it for our drives.
6. Returned status, if our drive.

For instance, while booting, hold down those two keys. You'll get the idea.

Other than this, everything is pretty much the same. However, if you format a disk using the LE system, it will have a very fast sector pattern (for LE drives) and rather slow (for Atari drives). Use LE-formatted disks for your LE development, and things happen 8 times faster.

The LE double density format is around 10 times faster than the Percom DD format.

Trouble? Check:

All cables in? You'd be surprised..

T-card fully seated?

Selected # of drives with POKE 1802? (see DOS 2 manual)

Too many drives for your DOS? (>4 requires 2.0D, not 2.0S)

Anything obviously wrong? Smoke rising?

Call us for further help. We can troubleshoot a great deal on the phone.

General Info about the LEDS5

The LE Disk System is an extremely high speed, reliable Z-80 driven disk drive, meant exclusively for the Atari and for those needing access to disk-level functions, mostly for copy protection. It is a very limited production item.

It was designed with few, if any, compromises in mind. 8K of Static RAM is provided, along with a 4 Mhz Z-80. 6 expansion slots are provided for other Atari peripherals, such as the Bit-3 80 column card or Eprommers. It has its own power supply.

The system is absolutely compatible with Atari disks, with no patches or other trivia. Our primary design criteria was COMPATIBILITY -- plug the disks in and go. No changes, no patches to DOS, nothing.

We do this by intercepting all disk calls in the Atari operating system, at a very low level (SIO). We then determine if we would like to handle the disk call or let the Atari do it. If we do it, it happens very, very fast; if the Atari does it, it does it very, very slow. By catching the disk calls at this low point, we have maintained compatibility with nearly every program on the market; we know of two games, total, that our system cannot boot, and no other programs it can't. (The games donot boot because they rely on a slow transfer rate, which our system definitely does not have).

As a comparision, an Atari 810, using the "C" format, can read in all sectors in about 1 minute, 15 seconds. We can do the same thing in 17 seconds. We could do it in 9, but, the Atar itself cannot take data faster than we are feeding it; the DOS was not written efficiently enough to do so, and besides, the 6502 cannot be dedicated to data transfers, what with screen refresh going on, etc.

All Atari disk commands are trapped at SIOV ($E459) and vectored to $C000, through the $C559. The $C000 was formerly unused; we now have a 2K ROM there. (Actually, 4K ROM, becase the 2732 4K ROMS are much faster than 27167s. But we only use 2K of it). The $C000 ROM ("Atari ROM") then sends the commands over to the disk.

The disk interfacing is handled by two three-port devices, memory mapped to $D700 (PORTA), $D701 (PORTB), and $D702(PORTC).

Port A is Z80 to 6502.
Port B is 6502 to Z80.
Port C is status & handshake:

Bit 7: data is waiting on Port B to be read (clears when B is read)
Bit 6: ditto for Port A.
Bit 0: When written to, shuts off the "SIO" trap..internal use only.

Bit 1: "Command Line": used to handshake with the Z80. When the Command line is high, the Z80 drops whatever it was doing and awaits a command, sent on Port B. Then, when the command line is dropped, Port B is read and that command executed.

Since there are all sorts of possible ways to foul up two-processor interfaces, including lockups, a generalized scheme to send the Z80 a command has been implemented, called GENCMD. Just put the desired command into the A register and JSR to GENCMD; it'll worry about the rest.

The $D700 area was used because no one uses it on the Atari; in fact, other O.S. boards don't even generate the signal and take it anywhere. (The $D500 and $D600 areas ARE used, by the way).

The system has its own power supply to avoid overloading the Atari buss. We now ship 40 watt supplies, but 32 watt (called VA on the package) is fine. You may have trouble with smaller supplies.

The disks are interfaced through the top of the DiskMaster board via parallel cable. 4 disks can be interfaced through each of two edge connectors. The drives are Tandon drives, modified rather extensively; we recommend you don't buy a Tandon and try to modify it, as it is very easy to burn out the system if you do it wrong. We do NOT use a standard interface, for speed and handshake reasons, and if you plug the Tandons into a standard interface, you'll fry some of the parts, guarenteed. Enough said?

We support up to 8 drives, as does Atari DOS. However, the Atari DOS MENU supports only up to 4, for some odd reason. Atari DOS was written with 8 drives in mind for the 815 disk, now dropped, but it'll handle 8.

If you get > 2 drives total, be sure to POKE 1802 with the right drive select codes, or your disks will return ERROR-160. See the DOS 2 manual about this.

The Tandon drives are exceptionally reliable. MTBF is 8,500 hours SPINNING with the heads lasting 15,000 hours. The RPM circuit, which is such a hassle on Atari drives, is exceptionally stable. (We set our drives to 290 RPM to maintain Atari compatibility).

Our drives use a slightly different "sector interleave pattern" to achieve their high speed. What it boils down to is that we store data on the disk in a slightly different position, in terms of rotation; data items that need to be read one after another are stored closer together, that's all. We are 8 times faster than the 810 drive because of this. Our disks can be read on the 810 and vice versa.

Our double density is brand new and compatible with PERCOM's, although they run at 300 RPM-- a possible compatibility problem. Again, Atari DOS can handle DD, but some applications programs may not; you'll have to see. Most will.

Double density is selected by a 4-position DIP switch on the upper right hand section of the board. "On" is DD, "Off" is SD. The right hand colum of lights indicate DD; on = DD. The left hand column indicates drive online; on=online, off=not there, flashing=on line and selected through SETDN (see SETDN).

A whole set of ROMS have been issued as the software has been improved. Release S/N 5 fixes ALL known bugs and adds double density for field testing. There may be some revisions past this if we find more bugs (please report any).

Copy System

The LE FDC-02,04, and 08 copy systems represent the state of the art in Atari disk duplication. No other system offers the speed and reliability of this system. The performance?

LEFDC-02: 1 copy every 17 seconds
LEFDC-04: 3 copies every 17 seconds
LEFDC-08: 7 copies every 17 seconds.

This compares to (on an Atari 810 or Percom drive):

30 sec formatting
90 sec copying
for 120 sec per diskette.
With an 8 drive copy system, in that same 120 seconds, we can produce 42 diskettes.

In addition, our system copies any Atari format possible. Bad sectors, duplicate sectors and such are no problem.

Verification takes a bit longer, as it does on the Atari 810. On the 810, 90 sec is required for verify. On the LE system, 17 seconds are required per disk to verify.

These speed estimates are rather conservative. For instance, they involve copying all 40 tracks of a disk. A typical machine language game is not 40 tracks in length; a reasonable number is 10 tracks. If only 10 tracks is copied, one quarter of the time (and still no pre-formatting) is needed; a copy will take around 5 sec, for 7 disks, as compared to about 60 sec for the Atari, for one disk.

Tandon disk drives are used; they are renowned through the industry as the most reliable available. They are all set to 290 RPM; other "copy services" frequently copy at 300 RPM, causing problems.

Copy software is provided along with the system in the form of a machine-language, automatic booting program. To use:

1. Prepare a master on our system.

2. Put the copy system disk in drive 1.

3. Switch on the system. The program will autoload.

4. The system will give you a welcome page. Put in 1,23, or 6 slave disks into drives 2,3,4,5,6,7, and the master disk into drive 1. The master disk MUST be write protected; the software will check for this, so put a write protect tab on it. Note that as an option we will ship the copy system so that the destination disks can be copied without write-protect notches.

5. Press START as the screen directions indicate.

The copy will now commence. A row of "+"'s will appear across the screen, as each trak is copied. A "+" indicates a successful track copy. After 40 +'s have appeared, the copy is done, and the verify will begin.

Note: the verify may be switched off optionally with a very short software patch. If you're using quality disks, your return rate may be so low that verifying will not be necessary.

D

Drive #2 will switch on, and each sector will be read in on the disk, with the read process optimized to cut down on the time spent (around 17 sec) per disk. If an error is encountered, a retry will occur; if the retry also fails, the system will stop and notify the oprator of the error. This generally means defective media; don't ship that disk! A good suggestion is to try recopying it.

If all goes well, the system will move to the next drive, and so on. While one drive is verifying, it would be a good idea to take the disk just verified and switch it for a blank disk, thus speeding the throughput.

When all 1/3/7 drives are done verifying, the copy is done.

ERRORS:
If instead of a "+", you get a mode inverse "1" or "2" on a given track, then the copy on that track cannot be guarenteed because no synchronizing sector could be found. This may not be an error; is that supposed to be an empty track? Depending on your disk copy protection scheme, it may be. You may also not have a sync sector on that track; we'll get into that in a moment. But if you get this on a normal track, you have a problem. Try re-seating the master disk and retrying the copy; NONE of the slave disks are good in this case.

If you are copying an empty track, why not make the software just skip that track? It will speed up your copies and you'll still have an empty track on the destination disk. Just a suggestion, but the whole copy program is table driven for this very reason .. to make it very easy to customize.

Next, about sync sectors. We switch the track copy on and off using the end of a "sync sector", which is any legit sector number on the track. The default sync sector # is 18. (decimal). Should you not have a sector #18 on one of your tracks, you will have to change the sync sector for that track number to a sector that does exists AND that is (preferably) unique. (While we could handle a doubled sector as a sync sector, it is a bit dangerous). For instance, if you have no sector 18, but only one sector 7 on the track, make sector 7 the sync sector. To change it, just go modify the "SYNC" table in the copy program; the table is quite well documented.

To alter which tracks to copy, alter the TRACK table. This will enable you to only copy the tracks that are relevant, and will greatly speed up your copy process. Again, the TRACK table is easy to understand; just list the tracks you want copied, and end with $FF. This is all internally documented.

Operating the copy system is a snap. About five minutes training should suffice to have someone operating it, once someone has set it up. The default system, as shipped, copies normal disks for 40 tracks; this needs to be optimized for strange copy protection formats. Just let us know, and we will step you through the process if you get confused. The copy software is written in Macro Assembler format, and may soon be available in Synassembler format; we seem to be switching to it regardless. Feel free to edit it however you like; we will do custom consulting for extensive changes. We strongly recommend making one copy system disk for each different form of copy protection; i.e., "This Disk Copies MunchMaster", "This Disk Copies BlitherBall", and so on, to help keep (relatively) untrained operators from being confused.

On making masters:

You MUST make your master disks on our system, which is also a development sustem. This is because we write the disks very cleanly which is required by some of the older 810's to read them; they had very bad data seperators in those drives, and clones won't work unless they are clean to begin with. Generally, making a master on our system is even easier than making it on an 810. First, our system is faster. Second, it gives you access to all sorts of powerful copy protection schemes. But in general, you must format and copy all sectors using our system for your master disks.

You may not make a backup master by cloning the original master. The masters must all be individually sector copied, duplicated, or whatever.

We are always willing to help you make your masters, but in general, remember, the disk works just like an 810, just faster and more reliabily. When you write, you can access every other sector (not quite every sector), not every ninth as in the 810; this will greatly solve your timing problems. Also, you can do things like address mark reads or track dumps to verify that all is well.

The copy systems can be expanded at any time by adding more drives, rewiring, and changing ROMs. We do not recommend you try this yourself! We have extensively modified the Tandon drives and they no longer use the "standard" floppy disk 34-pin interface. If you plug these drives into another system using Tandons, prepare yourself for smoke and optical effects -- they are THAT modified.