

Interleave Adjustment Utility

Version 1.8
July, 1988

(c) 1988 Radio KZ10
Dave Bushong

First Edition (June 1988)

Radio KZ10 provides this manual "as is," without warranty of any kind, either expressed or implied, including, but not limited to, the particular purpose. KZ10 may make improvements and/or changes in the product(s) and/or the program(s) described in this manual at any time.

This product could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication.

All specifications subject to change without notice.

(c) Radio KZ10 1988

Second Printing July, 1988

Manual rev: A

Notes:

Preface

This manual describes the operation of the software program IAU.EXE. It describes the technical details of operation and includes some general information about fixed disks.

IAU is a program which changes the physical arrangement of data on the computer's hard disk in order for the CPU to be able to access (read and write) the data in the least amount of time. This causes the computer system to operate in the fastest possible manner.

This manual describes some basics of hard disk operation. Then it explains what the "interleave factor" is, and why it might be changed. A description of the software algorithm follows.

A section is included containing frequently-asked questions and their answers about the program.

Appendix A contains application notes. Appendix B shows some of the display screens that you will see when running the program.

Contents

Page Subject

1	Computer Data Storage
2	Interleave
4	Read, Re-format, Write
5	Question and Answer
6	Pictorial display
11	Appendix A- Application notes
B-1	Appendix B- Figures

Computer Data Storage

Modern personal computers use rotating magnetic media for data storage. These include floppy disks (3.5, 5.25, 8 inch diameter) and fixed (5.25, 8 inch) disks. The data is stored and retrieved on the disk by means of flux changes in the magnetic coating of the surface of the media.

Floppy disks are typically made of Mylartm with a coating of ferro-magnetic (iron) particles on both sides. The disk retains the flexible quality of the Mylartm. This round disk is then sealed in a square plastic jacket, which keeps all but a small area of the media exposed.

The data is written and read analogously to the way a musical record is made, and later played. The information is contained in circular patterns of the disk. However, an LP has the information recorded in a spiral; the floppy disk has its data in concentric rings (like a dart board). The LP spins at 33 R.P.M.; the floppy disk spins at 300 R.P.M. (or in some systems, 360 R.P.M.) The floppy disk, depending on several factors, can save from 160 thousand bytes (called 160K) to over a million (a "megabyte").

An interesting experiment is to take one of these floppies out of its protective jacket and look at the media. Be sure that the disk that you do this to has no important information on it! Once you open the protective jacket, it and the disk shouldn't ever be put back into the drive!

Hard disks are very similar to the floppy disk counterpart. They are round, they have a coating of magnetic particles, and they have data on both sides of the media. However, they spin much faster in the drive (3600 R.P.M.), hold more information (20, 30, 40, or more megabytes), and do not bend. They are also not designed to be removed from the computer as with floppies.

The early (1980) personal computer hard disks had capacities of 5 megabytes. They were single-platter devices, with their 160 data tracks far apart by today's standards. However, progress in design and manufacturing has made thousand-track, multi-platter hard disk systems commonplace (and cheaper in today's dollars than the older ones!).

Thus, you can now have at your disposal, on your desk, a computer system with many "K" of RAM, many "meg" of hard

disk storage, rivaling the big systems of last decade in performance, cost, and storage.

Interleave

It helps to understand the following description if you can visualize what the data sectors "look" like on the disk. Imagine a pizza, cut into seventeen slices. Then, imagine you had some round cookie cutters in larger and larger diameters. When you have cut the pizza, first into slices (sectors) and then into cookie circles (cylinders, or tracks), you would end up with a bunch of pizza pieces that look like large fingernail clippings.

If you can bear to continue this analogy just a little longer, each of those clippings would compare to a sector of 512 bytes on the hard disk. You can easily determine the capacity of the hard disk using a calculator and the formula

$$\text{CAPACITY} = 512 \times \text{TOTAL SECTORS}$$

If, instead, you had five pizzas stacked up before you did the cutting, you would have ended up with more sectors. This is what happens with multi-platter hard disk units. If you had toppings on both sides of the dough on each pizza, you would have approximated the layout of modern PC hard disks. Each side with toppings relates to a data side, which is read by a separate head.

The formula above for disk capacity is, finally,

$$\text{CAPACITY} = 512 \times \text{SECTORS PER TRACK} \times \text{TRACKS PER PLATTER} \times \text{TOTAL HEADS}$$

The answer is in bytes.

As the disk spins, the sectors go by the read-write head. You can calculate the speed that the bits go by if you want to. With 17 sectors, 512 bytes per sector, and at 3600 R.P.M., that's 31,334,400 data bytes per minute, or 4,177,920 data bits per second. Since each sector has some "overhead", that is, extra data in the sector which allows the controller circuitry to access the data, the bits-per-second is a little higher than that. Typically you will see this parameter as 5 megabits per second.

4,177,920 bits per second means 522,240 bytes per second. If you attempted to read one track from the disk, you must be able to take it at that rate: the disk will not stop spinning while your CPU digests the data. (The IBM XT is not able to transfer the data via its DMA controller this

fast; in fact, it can only manage about 100K bytes per second. The reasons for this are beyond the scope of this description.) If the CPU isn't ready for the second sector when the disk drive is ready to send it, the disk spins, and 17 milliseconds later the sector shows up under the read/write head again. By now, the CPU is ready for it, and it accepts it. However, this will slow your data transfer rate down by a factor of 18, dropping it to 29,013 bytes per second! Poor, indeed!

What would happen if you only recorded four sectors on every track, with evenly-spaced gaps between them? There would be enough time between sectors for the CPU to transfer the data before the next one showed up. Now your transfer rate would be four sectors per spin, instead of one. The disadvantage is that your capacity just shrunk by a factor of more than four.

A compromise is available. The data can be placed on the disk in a way that allows enough time between sectors for the computer to read them as fast as they come by, and having the track full of sectors. If you write the four sectors, as described above, but then put sector number five immediately after sector number one, then sector number six immediately after number two, and so on. It would take four or five spins to get all seventeen sectors. This isn't 500K bytes per second, but it's four times faster than reading one sector per spin.

The process just described is called interleaving. This is the way that your data is placed on your hard disk.

The interleave factor is defined as the number of times the disk must spin in order to read all of the sectors on disk in order, from sector number one to sector number n (the highest number).

An interleave of one (or called "one-to-one") is the fastest possible: one whole track is read in one revolution. The sectors are stored on the disk as 1-2-3-4-5-6-7-8-9-10-11-12-13-14-15-16-17. An interleave of two (2:1) has the sectors stored on the disk as 1-10-2-11-3-12-4-13-5-14-6-15-7-16-8-17-9. The typical IBM AT comes to you with an interleave of three, and the typical XT comes to you with an interleave of five or six.

Who decides what interleave is right for your system? The person who installs the hard disk. If you bought the system

ready to go, it was your dealer who did it. If you built the system from subsystems (mail order, for example), it was probably you who did it (perhaps unknowingly). A generally safe value is determined for a certain type of system (processor type, speed in MHz, networking requirements, etc.), and this "magic number" is then used for all similar systems.

There is a big performance penalty if the interleave is too small -- in fact, if you are off by one in that direction it's like having an interleave of, say, 19 instead of 2, or 20 instead of 3. Since the controller misses the second sector of each multi-sector read (called "blowing a rev"), an extra revolution is required for each sector to be read.

So it's important to not have the interleave too small. But what about having the value too high? Not a terrible thing, but you still aren't getting the most performance out of your system. The CPU is saying "gimme" when the controller is saying "hang on, it's coming."

The best situation to have is to experimentally determine the best interleave for your specific computer, with all of your peripherals connected, and then change the interleave to that value.

That's what your new program does. And it does it without messing up the data that you already have stored there.
Read, Re-format, Write

The sectors are placed by an operation called "low-level format." This is distinguished from a "regular format" because it is not done by the FORMAT program on your DOS disk. That program really just arranges the DOS data on the already-formatted hard disk.

As most computer users know, formatting wipes out your data. You don't want to format a disk which has non-replaceable data on it. That's why the FORMAT program asks you, "Are you sure?" It's a drastic thing.

In order to change (or initially set) the hard disk interleave, each track must be formatted (low-level format) specifically for that value. As you may guess, this type of format also wipes out any data which is there. So if you want to re-format your hard disk, you save all your data onto floppies or tape backup; then re-format; then restore from your backup media. Ugh. That's so painful that most

people aren't interested in doing it. Besides, if anything goes wrong during the restore, and/or your backup media is altered or damaged, you will lose some or all of your data... forever. Double-uggh.

The IAU program saves you all of the headaches, and gives you all of the optimum performance you can get.

After determining the optimum interleave for your disk drive, the program proceeds to change it. It reads an entire track into memory, then formats the track to the new value, then writes the data back to the same track (but at the new interleave). This continues for the entire disk. When it's done, your disk will be better than new.

But before you run the program, run one of the popular performance analyzer programs (such as Peter's SI program, or one of the Core tests). Then run IAU. Then run the performance test again. You will be pleasantly surprised with the increase in performance.

The greatest part of all this is that you really are getting something for nothing! All of that performance potential was always there; you simply are getting your money's worth now.

If you run the program, and you agree with me -- that your disk performance is really better -- please read the section about ShareWare, at the end of the manual.

Pictorial Display

With the release of version 1.7, you can observe a pictorial of your hard disk system on the display. The only requirement is that you have an E.G.A. or V.G.A. adapter installed, with either monochrome or color capability.

The pictorial will show you what the inside of the hard disk looks like (platters and heads), and will show you what the data would look like if you could see it on the disk. The interleave measured by the program is what is used for this picture. So you must select the "M"easure option before you are shown any graphics. When you have measured the current value, then select "I"nformation (see figure 9 in appendix B). The graphic display will look like figure 10 in the appendix.

The program which does this is MAKEPIC.EXE. This was also written in Turbo 'C', using the graphics routines available in version 1.5.

If you want to run MAKEPIC directly from the command line, the calling parameters are:

```
MAKEPIC drive interleave rpm
```

The third parameter is optional; if you leave it off, MAKEPIC will measure the R.P.M. at run time.

We are interested in your comments about this program.

Question and Answer

This section addresses some of the frequently-asked questions about interleave and the IAU program.

Q Will IAU work with my computer system?

A: Yes, if you have an IBM or compatible system with one or two hard disks, at least 256K of RAM, an IBM-supported video system (MDA, CGA, EGA, CGA) or de-facto standard (Hercules, Plantronics) video system.

Q: What about two hard disks? Which one will the program work with?

A: The program detects that you have more than one, and asks you which one it should use. Once you answer the question, that is the only drive which the program will access until you exit the program. You may specify the drive from the command line, also (see below).

Q: What if I have more than one partition on my hard disk? Can the program deal with that? How about if it's not a DOS partition?

A: The program operates at the physical device level. That means that if you have your physical drive split up into two or more logical drives, it makes no difference. Logical partitions do not change the physical structure of the hard disk. If you want to change just one partition of your hard disk, you can select a cylinder range for the program to adjust.

Q: Does the program work with R.L.L. drives?

A: No, if they don't have 17 sectors per track. We have found that RLL drive manufacturers use a different method of setting the interleave during a format track call, and have chosen not to support R.L.L. drives, except on a case-by-case manufacturer-supplied driver basis.

Q: I'm concerned about Trojan Horse, or virus, programs. How do I know if someone has tampered with IAU.EXE?

A: Good question. A file compression program called PKARC (and its companion PKXARC) can be instructed to give information about the archive file, like this:

```
C>PKXARC /V ILEAVE16
```

You should see this type of information:

Filename Time CRC	Length	Method	Size	Ratio	Date
----- -----	-----	-----	-----	-----	-----
IAU.EXE 16:32:46 6A4E	66118	Squashed	52270	21%	06-16-88
MAILER.EXE 14:51:20 5677	12330	Crunched	9059	27%	06-08-88
PARK.COM 10:15:40 5FD9	9690	Crunched	8257	15%	06-14-88
----- 0003	----- 88138		----- 69586	----- 22%	

The PKXARC program prints its sign-on information just before that is displayed. The file size of ILEAVE16.ARC is 69,747 bytes, and should have a date of 17-Jun-88, and a time of 4:51 p.m. Later versions of the program will have different values. Since the author has no control over the integrity of downloaded programs, it is a good idea to get a floppy direct from the source.

The CRC value will change if any part of the program is altered by unauthorized persons. If these values aren't the same as when you got the program, something is wrong.

If you have gotten a self-extracting archive file, or if some programs have been added to the archive, there might be more than three files shown - that is OK. Just be sure that the CRC values are the same as when you got the program from me.

Q: Even if this is the real version of the program, how do I know whether or not it will trash my hard disk?

A: Another good question. This type of program is potentially destructive to your data. Some manufacturers of hard disk controllers do not follow the IBM method of performing read, write, and/or format commands. It is possible that some of your data could be lost. You should try the program on the "test" cylinder, and be satisfied with the results, before you do it to the whole disk.

Q: What is the "test cylinder?"

A: This is the track which is closest to the middle of the disk (and it has a higher number than any tracks which have your data on them). It is used by diagnostic programs to perform tests, and is a guaranteed safe place to perform testing. It is guaranteed to have no media defects on it,

and is guaranteed to be OK if a program erases whatever data happens to be on it.

Q: What should I do before running the program?

A: YOU SHOULD ALWAYS HAVE A CURRENT BACKUP OF ALL THE DATA YOU WANT TO KEEP. THIS IS TRUE WHETHER OR NOT YOU CHOOSE TO RUN THE IAU PROGRAM. BACKUPS SHOULD BE A REGULAR PART OF YOUR COMPUTER OPERATION!

Q: How do I work the program?

A: The program looks like spreadsheet programs look. See figure 1 in Appendix B. You press the cursor keys (located to the right of the Enter key on most PCs) to select the option that you want. Each selection will be shown highlighted as you pass by. Underneath the option is a brief description of what will happen if you press enter. Most options have "sub-options", that is, you will be shown a new list of options when you choose. This method of option entry is easy for most people to get used to.

Q: What is the proper sequence to use for the first time user?

A: Measure the test track (see Figure 2). Then determine the optimum value for the test track (Figure 4). Change the interleave value on the test track to the optimum value determined (Figures 5 & 6). Then re-measure the test track, to see if the change actually took place.

Q: What about "on-line help?"

A: The main menu (the first screen with options) has a "Help" selection which is a summary of this manual. You may refer to it whenever you forgot some of the basics about interleave. Registered users can get more help by pressing the F1 key when a menu is up. At that time they will be able to view the file IAU.HLP which is provided with ILEAVE17.EXE for registered users. You can also run the program with a command line parameter of "?" and get a quick description of the program, that is, you type

IAU ? (and press Enter).

Q: What if the program has trouble reading data from the disk?

A: If it can't read a track, it skips it, and leaves it at the old interleave value. In this way, it never makes the disk worse than it was. If you select a track for "Optimize" which is unreadable, the program will tell you to

select another one. Any read, format, or write errors which occur during the interleave operation will be displayed on the screen. Some errors are expected and you should not panic.

Q: Can I get this error log to print out? How about to a disk file?

A: Yes. When you invoke the program, use the command line argument logfile=xxx, where xxx is the name of a file (LOGFILE.PRN, etc.), or a device (such as PRN, COM1, etc.)

Q: When I run the program with my (older) XT, it starts off great, but then takes longer and longer for each track. At this rate, it will take all night!

A: That's OK, though, since you only run this program once, but there is a way around this. Stop the current operation, and exit the program. Then, run it again with the "RECAL" command line option.

Q: How do I stop the program once it gets started changing the interleave?

A: If you press the Escape key during the re-interleave operation, you will be returned to the menu. Do not press CTRL-C or CTRL-BREAK while this program is running. You have been warned.

Q: Ouch. What if I have some other kind of error which causes an immediate drop to DOS? Like the infamous "Abort, Retry, Ignore" message? And I have to choose "Abort" for some reason?

A: That's not so bad, except for this: BE SURE THAT IN THAT CASE, YOU REBOOT THE SYSTEM BEFORE YOU READ OR WRITE THE HARD DISK. THIS IS VERY IMPORTANT IN CASES WHERE YOU ABNORMALLY EXIT THE PROGRAM.

Q: What is a "command line argument?" What do they do? How do I use them? Why should I?

A: Command line arguments are options that you can give to a program. For example, when you type DIR A: you are telling the system that you want a directory, but for drive "A". And when you tell the FORMAT program to put the operating system on the disk, you give the /S command line argument. IAU has six command line

arguments. They are:

Argument	Function
/ALTER	allows you to change the drive table
/CACHE	allows operation of the program when using a cache program
LOGFILE=	specifies that errors will be sent to this device or file
/RECAL	allows better operation with controllers which recalibrate before each format-track operation
/VAL=	allows you to select a default interleave value from the command line
/COMPAT	Skips the IBM compatibility byte check at the beginning of the program
/DRIVE=	Allows you to specify either drive 0 or 1 from the command line.
/? or ?	Display a brief description of the program and the revision level.

Most of the time, you won't use any of these. ALTER is dangerous for those who don't know what they're doing. CACHE might be required in some systems, and the program will tell you if yours is one of them. LOGFILE was discussed earlier. Note that LOGFILE does not use a switch character (slash) as all the others do.

Q: When I am finished running the program, it tells me that I should register the copy! I already did! How do I get rid of that message?

A: If it really bothers you that much, you can give this command to DOS after you boot your system: SET IAU=Q. Then when you run the program, the message will be suppressed.

Q: What about the other files in the archive? What are they for?

A: PARK is a program which will place the read/write heads at a safe place for transportation. If you don't run this (or a similar program) before physically moving the computer, a bump could cause the disk to be permanently damaged. This program puts the heads in what the manufacturer calls the "landing zone," where no data is stored. MAILER is a program which allows you to contact the

author, either for the registration fee, or for comments about the program. It prints out an envelope on your printer which can be used for mailing. Other public domain programs may be tossed in from time to time which might be useful for IAU users.

Q: How many copies of the program are there?

A: It's hard to know how many copies of a shareware program are being used. There have been about 900 people (as of July, 1988) who have expressed interest in one form or another.

Q: What is "Share Ware?"

A: You know, don't you? It's where you get a program and try it out to see if you like it. Then you like it, and keep using it. Then you say, "That's a nice program! I'll bet a commercial version of this would cost hundreds of dollars! So I don't feel too bad sending the author the paltry sum that he/she suggests. That way I know I'll get on the update list and be able to ask questions over the phone!"

Since you are reading this manual, it means that you have requested the manual with a donation. Thank you. This makes it all worthwhile for me.

Appendix A
Application Notes

Application notes describe changes or other items of interest to users of the program. Minor program bugs are also described, with their possible solutions. The application notes begin on the next page.

APPLICATION NOTE 1 - FLOATING POINT SUPPORT

PROBLEM:

During the actual "Change interleave" operation, the percent complete indicator jumps all over the place, instead of showing a steady increase as it progresses.

CAUSE:

The Auto-87 detect logic of the run-time package in Turbo C doesn't always work - or - your SETUP parameters indicate that you have a numeric co-processor when in fact you don't (or it's not responding).

SOLUTION:

Before you run the program, enter this at the DOS prompt:

```
SET 87=N
```

This will tell the run-time math package that there is no numeric co-processor.

AFFECTING:

This problem affects the display only, and has no effect on the hard disk. No noticeable effect on program speed is noticed. The problem is eliminated in version 1.7

APPLICATION NOTE 2 - DISK ERRORS DURING HARD DISK READ/WRITE

PROBLEM:

During the re-formatting operation, error codes appear in the ERROR CODE window on the screen. Some users are concerned about these errors, and their meaning.

CAUSE:

Some errors are caused by previously-unknown bad sectors; however, the majority are caused by tracks that are marked bad and will not ever be used by the operating system.

SOLUTION:

Use the list below to determine if the error was previously encountered (error code 0A). If so, there is no chance for danger; if not, you should run one of the utilities which can update your FAT (Norton's DISKTEST, DOS' DETECT, etc.).

Error	Meaning
CC	Write fault
BB	Undefined error
AA	Drive not ready
40	Seek failed
20	Bad controller
11	ECC corrected data error
10	Bad ECC on disk read
0A	Bad sector flag detected
05	Reset failed
04	Sector not found
02	Address mark not found
01	Bad command from program

There are a few more error codes possible; however, these cover the majority of types that you will see. In these cases, the IAU program will not change anything on that track.

AFFECTING:

XT, AT. Users can verify that there was no change to the interleave using the (M) easure option for the track in error.

APPLICATION NOTE 3 - INTERNAL ERROR CODES

PROBLEM:

Some users have reported error codes generated during the running of the program.

CAUSE:

Run-time errors which can not be handled or determined.

SOLUTION:

Varied - depending on the cause of the problem(s).
Version 1.7
of IAU has the following error codes:

- 1 No hard drives detected in system
- 99 Controller RAM diagnostic fails
- 98 Drive diagnostic fails
- 97 Controller diagnostic fails
- 45 Not enough system RAM to execute (check tsr's, ramdisks, etc.)
- 33 Same as 45

Those are all "fatal" errors, meaning that the program cannot recover from the condition. The message is in the format of:
Fatal error: xx
where xx is a number from the list above.

Another type of problem can occur which will be displayed as

Error # xx

And the number will be a two-digit (hex) value. This means that the BIOS compatibility byte is not IBM-compatible. You may bypass this check by entering the program with the /COMPAT command line argument.

Appendix B
Figures

The figures on the following pages are referred to in the preceding text. They consist of "screen dumps" from the program while running on a typical system.

