

RAM Drive Sync Batch

For Windows and RAMdisk

By Richard Alexander Hall, 2014-12-31. I release this documentation and the batch file that accompanies it into the Public Domain. This .pdf file has an embedded Open Document Text (.odt) file and fonts (which are also open-source/free) for easy edits in e.g. LibreOffice.

What and Why

If you are, for example, an artist who works with extremely large files (maybe 7,500MB+), things can slow down a lot when you save and copy.

This slowdown can be dramatically alleviated with a RAM drive, which is a virtual hard drive that resides in your computer's RAM (memory). This batch file solves a caveat of that approach.

You probably don't need to read the rest of this "What and Why" section—you can just skip to the "Setup" section, or if you've already got it set up, skip to "Usage" for reference.

The drawback of a RAMdisk is that if you forget to copy files off of the RAMdisk to permanent storage before you turn off the computer (or if a power blip or outage turns off the computer), you lose your data.

That drawback can be dealt with by using a RAM drive with a batch that runs in the background to synchronize the RAM drive with permanent storage (like a dedicated hard drive folder). With such a setup, you can save gigantic files virtually instantly, and keep working on them, while the batch file migrates the changes more slowly to the permanent folder. **BUT.**

Note the USAGE WARNING under "Usage."

This batch file:

RAMdriveSync.bat

—can help easily accomplish such a setup.

It relies on the freeware program RAMdisk, available from:

<http://www.dataram.com/>

It also relies on the ROBOCOPY command which is built into many versions of Windows.

Setup

An outline of how to use this batch with RAMdisk—apologetically bypassing greater detail:

1. Obtain a copy of RAMdisk, and install it.
2. Run the installed RAMdisk program, and create a RAM drive (disk) to accomodate the largest possible file size group you're comfortable working with (factoring that the RAM disk will be loaded into your computer's main memory or RAM). You may want to

create it as an unformatted partition, and after it's created, go into Windows' Disk Management utility and format it as NTFS with a drive letter of e.g. "Z:\".

3. Create a folder which you will want to regularly and automatically sync your RAMdisk files to (to preserve them permanently on a hard disk) as you work with them, e.g.:

```
C:\Users\%username%\Documents\_ZsavedFiles__StaticMirror
```

(Where %username% is a Windows environment variable that means "your user folder." If you type %username% into Windows explorer and press <ENTER>, it will jump to your user folder.)

Note that you may set the source (RAM drive) and destination (permanent hard drive storage) folders for this batch file in the first lines of code of *RAMdriveSync.bat*, e.g.:

```
SET SYNC_FROM_DIR=Z:\
SET SYNC_TO_DIR=_ZsavedFiles__StaticMirror
```

—where *SYNC_FROM_DIR* will be the RAM drive (source) and *SYNC_TO_DIR* will be the permanent hard drive storage folder (destination).

Usage

First make sure that the drives and directories referenced in the SET commands of the batch file exist. Then, with everything set up (as described above), simply place your working files in the Z:\ RAMdrive, then double-click *RAMdriveSync.bat*, and leave it running.

Work with your files from the Z:\ RAMdrive (source), and once per minute, the batch will copy all of their changes to e.g. the *_ZsavedFiles__StaticMirror* (destination) directory. This target directory is where you will move files from—to whatever other permanent storage folder of your choice—after you're done working with them via this RAM drive sync setup.

Depending, files probably won't stay in the source (Z:\) directory between power downs—or power outs—depending. They will however remain in the target (*_ZsavedFiles__StaticMirror*) directory.

USAGE WARNING: Because specially huge files (e.g. ~7,500MB+) can sometimes take at least a few minutes to save from your computer's memory to your hard drive, you should take short breaks every long now and then (which is good working sense to avoid fatigue anyway) so that the batch can take the time, if necessary, to actually get your data copied from the RAM drive to permanent storage.

Also, see "NOTE" in the comments of the batch file itself.

This is because if it takes five minutes to copy a huge file you're working with, but you save it every minute, the batch may start the copy over every minute, so that it might not ever get to take the full five minutes necessary to complete the copy.

But this provides a data-dependent incentive to work smarter (without overworking), don't you think? ;)