

macOS Removable Media Auto Backup

This guide shows you how to set up **automatic backups** of any removable media whenever you connect it on macOS using **FreeFileSync** and a lightweight LaunchAgent.

The example uses the name usbtest for everything and can be customized by replacing the file names & paths to names and locations that suit your needs.

I use this method to trigger automatic backup syncs of my Rekordbox usb drives and the micro SD cards I use for my Polyend Tracker+, TrackerMini, Play+ & Synth to a folder within my iCloud drive. As soon as the drives mount it is scanned for changes since the last backup and any new or changed files are synced to the designated backup location. Since the location I chose is within my iCloud drive I get the added benefit of additional cloud based redundancy.

The attached zip includes this tutorial in PDF, example FreeFileSync batch files, and example install & uninstall script files you can customize with your your preferred names and paths.

How It Works

- A FreeFileSync batch job (`.ffs_batch`) mirrors your removable media to a chosen backup location.
- A LaunchAgent runs the job **only when the media mounts**.
- Logs are written to `~/Library/Logs/ffs_usbtest_sync.log`.

Prerequisites

- macOS (Intel or Apple Silicon)
- [FreeFileSync](#) installed
- Basic comfort using Terminal (copy/paste commands)

Step 1 — Choose Your Folders

For this example:

- **Removable Volume:** `/Volumes/USBTEST`
- **Backup Location:** `/Users/{yourusername}/bin/USBTEST-BACKUP`
- **.ffs_batch file:** `/Users/{yourusername}/bin/usbtest.ffs_batch`
- **Path Note:** In the scripts `/Users/{yourusername}/` is replaced with `~/` and will automatically use the current user as the path so you don't need to edit that part in the scripts. Either way will work.

Step 2 — Create and Save Your FreeFileSync Batch

1. Open **FreeFileSync**.
2. Set **Left folder** = `/Volumes/USBTEST`
3. Set **Right folder** = `/Users/{yourusername}/bin/USBTEST-BACKUP`
4. Set the **sync variant** = *Mirror* (so the backup matches the card, including deletions).
5. Review settings and adjust as needed.
6. In the menu bar, click File → Save as batch job...
7. Name it `usbtest.ffs_batch`
8. Save to: `/Users/{yourusername}/bin/`
9. In the batch settings, enable **Auto-close** and optionally *Run minimized*.

You now have a working batch job you can run manually.

Step 3 — Automate Sync on Mount (Script + LaunchAgent + Test)

Open **Terminal** (Applications → Utilities → Terminal). Copy and paste the following block into Terminal and press **Enter**. This will create the runner script, set up the LaunchAgent plist, validate it, load it, and then test it. Ensure your media is mounted and named appropriately.

```
mkdir -p "$HOME/bin"

cat > "$HOME/bin/ffs_usbtest_sync.sh" <<'EOF'
set -euo pipefail

VOLUME="/Volumes/USBTEST"
BATCH="$HOME/bin/usbtest.ffs_batch"
FFS="/Applications/FreeFileSync.app/Contents/MacOS/FreeFileSync"
LOG="$HOME/Library/Logs/ffs_usbtest_sync.log"
ERR="$HOME/Library/Logs/ffs_usbtest_sync.err"

LOCKDIR="/tmp/ffs_usbtest_sync.lock"
if ! mkdir "$LOCKDIR" 2>/dev/null; then
    echo "$(date) - Another sync already running; skipping." >> "$LOG"
    exit 0
fi
trap 'rmdir "$LOCKDIR" 2>/dev/null || true' EXIT

sleep 8

if [[ ! -d "$VOLUME" ]]; then
    echo "$(date) - Volume not mounted: $VOLUME" >> "$LOG"; exit 0; fi
if [[ ! -f "$BATCH" ]]; then
    echo "$(date) - Missing batch file: $BATCH" >> "$LOG"; exit 1; fi
if [[ ! -x "$FFS" ]]; then
```

```

    echo "$(date) - FreeFileSync not found: $FFS" >> "$LOG"; exit 1; fi

echo "$(date) - Starting FFS sync..." >> "$LOG"
"$FFS" "$BATCH" >> "$LOG" 2>> "$ERR" || STATUS=$? || true
STATUS="${STATUS:-0}"
if [[ "$STATUS" -eq 0 ]]; then
    echo "$(date) - Sync complete." >> "$LOG"
else
    echo "$(date) - Sync FAILED with status $STATUS" >> "$LOG"
fi
EOF

chmod +x "$HOME/bin/ffs_usbtest_sync.sh"

SCRIPT="$HOME/bin/ffs_usbtest_sync.sh"
LOGDIR="$HOME/Library/Logs"
PLIST="$HOME/Library/LaunchAgents/com.usbtest.ffs.sync.plist"

mkdir -p "$HOME/Library/LaunchAgents"

cat > "$PLIST" <<EOF
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN"
    "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
    <key>Label</key>
    <string>com.usbtest.ffs.sync</string>
    <key>ProgramArguments</key>
    <array>
        <string>/bin/bash</string>
        <string>-lc</string>
        <string>${SCRIPT}</string>
    </array>
    <key>StartOnMount</key>
    <true/>
    <key>StandardOutPath</key>
    <string>${LOGDIR}/ffs_usbtest_sync.log</string>
    <key>StandardErrorPath</key>
    <string>${LOGDIR}/ffs_usbtest_sync.err</string>
</dict>
</plist>
EOF

chmod 644 "$PLIST"
plutil -lint "$PLIST"
launchctl bootout "gui/$(id -u)" "$PLIST" 2>/dev/null || true
launchctl bootstrap "gui/$(id -u)" "$PLIST"

```

```
launchctl enable    "gui/$(id -u)/com.usbtest.ffs.sync"

diskutil unmount "/Volumes/USBTEST"
sleep 2
diskutil mount "USBTEST"
sleep 10
tail -f "$HOME/Library/Logs/ffs_usbtest_sync.log"
```

Expected output in the log:

- Starting FFS sync...
- Sync complete.

Adapt for Anything

- This works universally: any removable media → any backup destination.
- SD cards from photography, video & music equipment, usb sticks, external hard drives.
- To back up to Google Drive or Dropbox: set your **Right folder** inside those synced folders so the backups are also cloud-synced

Uninstall

Remove the agent and script:

```
launchctl bootout "gui/$(id -u)" "$HOME/Library/LaunchAgents/
com.usbtest.ffs.sync.plist"
rm "$HOME/Library/LaunchAgents/com.usbtest.ffs.sync.plist"
rm "$HOME/bin/ffs_usbtest_sync.sh"
```