

Backing Up Files on the Micro Journal Rev 2.1 (or Rev 2) using rclone and Dropbox

Revision 5, January 21, 2026

This revision is based on having micro Journal Linux image 2025.11+ or later on your Micro Journal. This is the version that comes with the newer Rev 2.1. For those who have the now out of production Rev 2, you will need to make sure you have updated to the newest firmware as this document will now assume important changes that were made, particularly around network availability. You will find the latest Micro Journal Linux image here:

<https://github.com/unkyulee/micro-journal-linux/releases>

The Micro Journal Rev 2.1 comes with two pretty nice ways of getting files onto your computer. First, by using your computer's browser as a file manager using WiFi. Second, on the Rev 2.1 (but not the Rev 2, even with the newer firmware), you can copy files to an external thumb drive. These are both covered in the user guide included as part of the quick start page. Both are excellent, one online and one off line. You may not feel a big need for anything else.

However, I sometimes would really like my files to be sent directly to Dropbox. If my drafts land in Dropbox, they are instantly available to any of my other devices I might edit on. That's really useful to me. A program called **rclone** can do this. And, although it can enable a full bi-directional sync, I have chosen to stick with just backing up (copying) my draft files I create on the Rev 2.1 to Dropbox because it fits my workflow. Bi-directional sync is not well sup-

ported and even the rclone dev considers it experimental and warns against using it for anything important. It's too risky and the need is too minor on this machine. If you want to upload a file to work on, just use the browser or external flash drive method.

A final note before plunging in. The instructions documented here were developed using a Linux computer. It turns out that the Windows version works just fine (in fact there is little difference). I did all my testing with my Linux computer despite being primarily a Windows user, but I will include the Windows instructions as well as I did test the Windows version. Presumably, this should work for all the other cloud services that rclone offers to connect to, but I only know for sure that it works with Dropbox.

Notation conventions in this document.

Anything you are asked to type in will be in bold, blue, and indented. If I say “type the command” it means you hit enter once it is typed in. Type everything precisely as shown in a line, including punctuation and spaces. Linux is very fussy on exactly how things are typed. 9 times out of 10, that will be the reason for any errors you receive.

If you already set this up following my instructions on the Rev 2, you can just copy the same rclone.conf file you created then to the documents directory on the Rev 2.1. Skip parts 1 and 2 of these instructions and go right to part 3 (Page 6).

Here is what I set up that now provides reliable back-ups of my files to Dropbox:

1. Install rclone on Computer.

1a: Linux.

In a terminal, issue the following command:

```
sudo -v ; curl https://rclone.org/install.sh | sudo bash
```

This will run and when it is done, rclone will be installed.

1b: Windows.

More steps, but pretty much the same. The main difference is that you first have to go out on the internet yourself and download the install file.

Go to <https://rclone.org/downloads/> and select the Intel/AMD - 64 Bit version for download. You will be downloading a zip file. Unzip the contents. Rename the unzipped folder to something easy to type (e.g. rclone) and move the folder to someplace easy to navigate to in a terminal window, meaning not buried too deeply in your folders. You are going to be doing most everything in a Windows terminal, not with a GUI, just like on Linux. Since both of the standard Windows terminals opens to the current user home (C:\Users\<username>), that's where I would put the rclone folder.

Open a terminal in Windows. I stubbornly use the legacy "Command Prompt" as my terminal, but Powershell works just as well.

Now, assuming you copied your rclone folder to where your terminal has opened, enter the command:

cd rclone

Your command line prompt should change to show that C:\Users\\rclone is your current location.

Type the **dir** command to confirm you are in the right place. There will be a small number of files. Rclone.exe is the important one. Rclone doesn't install on Windows. It's a portable app that runs in a terminal, just like the Linux version. In fact, even on Windows, it really is a Linux app.

So, just issue the command:

rclone.exe

You now have rclone up and running.

2. Setup a Dropbox Remote for rclone (Linux or Windows).

From this point on, the directions are identical for both Linux and Windows.

First, pick a name for your Remote. It doesn't matter what that name is but you will enter it into rclone when you do the setup. I called mine "Drop" for Dropbox and will use it in the examples below. You can call it anything you want. It is purely arbitrary, just a name for the remote connection you are making to Dropbox.

Second, pick or create a folder in Dropbox to use for backing up your files to. Just one folder with no sub-folders. I created a folder called "*00-drafts*" in Dropbox.

Now type the following command:

rclone config

This will start up an interactive setup script that looks like what is shown here:

<https://rclone.org/dropbox/> This script shows you the questions and the most common answers.

First item answer is *n* for new remote, second one is the name you are giving your remote, type of storage will be the number associated with Dropbox (which was 12 when I did it, but it could change). It's a long list, so you will have to scroll back to find Dropbox. During this script, it will have you use a link to Dropbox in your browser to get access to Dropbox authorized. Most of the rest of the questions I had no clue what the answer was. In those instances I either hit Enter or answered Yes, taking the default for each item. When you answer “y” to the final “Keep this remote?” question, it will be saved.

Finally, you want to test the new remote you just set up. In either a terminal in Linux or in the rclone app in Windows, type the following command:

```
rclone ls <remote name>:<folder name>
```

In my case, that would be:

```
rclone ls Drop:00-drafts
```

This command tells rclone to look in the 00-drafts folder on Dropbox and list the files there. If it succeeds, you are good to go (if there are no files in that folder, it won't list anything, so you might want to stick a file in there for the test).

The final step for part 2 is to get a copy of the rclone configuration file that you just finished setting up on Linux or Windows and copy it to the Rev 2.1.

In either the Linux Terminal or rclone on Windows, type the command:

```
rclone config file
```

It will return the path to (location of) your rclone.conf configuration file.

Start up the Rev 2.1 and select the “file_share.sh” script in ranger. Open the browser on

your computer and enter the provided URL. Choose upload and navigate to the location where your rclone.conf file is. Select that file and upload it. It will now be located in the documents folder on your Rev 2.1 Also save a copy of your rclone.config file somewhere safe for future use.

You are finally ready to begin Step 3. Everything from this point is being done on the Rev 2.

3. Install Rclone on the Rev 2.1

Starting at the launcher, to install rclone type **q** to drop out of ranger and execute the following set of commands one at a time:

```
cd
wget https://rclone.org/install.sh
chmod +x install.sh
sudo ./install.sh
```

You will see, after a pause, text being displayed on the screen that should indicate rclone is being installed. When it is finished, you can check that rclone is installed by typing the command:

```
rclone -v
```

You should see the help file. That confirms the installation.

4. Set up rclone to connect to Dropbox

First, you need to find the location of the rclone.config file it created when it installed rclone. Type the command:

```
rclone config file
```

This will return the path to the file. It will likely look like mine, which is:

/home/microjournal/.config/rclone/rclone.conf

It may indicate there is no rclone.conf file but will give you the path where it would be located, which will still look like the above path. That's fine. That doesn't change any of the steps that follow.

Now you are going to use the copy of the rclone.conf file that was created on your computer, and that you copied to your documents file. to overwrite (or write) the new rclone.conf file that was created when you installed rclone on the Rev 2.1. The rclone.conf file from your computer has all the information about the Dropbox remote. Type the following command:

mv microjournal/documents/rclone.conf .config/rclone/rclone.conf

Now rclone should be able to use the Dropbox remote. Test it just as you did on your computer. Again, using my remote name and Dropbox folder name as an example, I would type the command:

rclone ls Drop:00-drafts

If I get a file listing for the contents of my 00-drafts folder on Dropbox, I'll know it works.

5. Create script to back up files to Dropbox folder

We are almost there. Type the command:

cd microjournal

This will return you to the directory where all the scripts you see in ranger are stored. You can confirm that by executing a **dir** command.

You are now going to create a script to copy the files you have in documents to your

Dropbox folder. Script files always have the extension `.sh` You can name the file anything you want, but I recommend you keep it short and simple and something that reminds you what the action will be that it executes. I will use mine as an example. I called mine ***dropbox.sh*** (mostly to get it to group alphabetically with the other file management scripts in ranger).

To create the script, type the following command:

```
nano dropbox.sh
```

Nano will open a blank new file with the filename I gave it. You will type the following lines into it:

```
#!/bin/bash
```

```
#Copy files from the documents directory to a folder in Dropbox  
#This is backup/ copy, not a bi-directional sync
```

```
rclone copy $1 documents/ Drop:00-drafts
```

The first line is required for all scripts. The next two lines are comments (Linux will ignore lines beginning with a hash tag). Even for something as simple as this script, it is good to have comments that remind you six months later what the script does. The final line tells rclone to copy all files in the documents directory to my 00-drafts folder on Dropbox. Obviously, when you do this, use your remote name and folder name. Save and exit nano.

Next you have to make the script executable. Type the command:

```
chmod +x dropbox.sh
```

The final step is we need to test that rclone can copy the files to Dropbox. You are going to run the script from the command line but with an option that tells rclone to log whether the file copy operation would be successful, but without actually copying the files. Execute the following command:

`./dropbox.sh --dry-run`

The period-forward slash is how you execute a script file from the command line. The option “--dry-run” tells rclone to run but not actually copy files. You should see output on the screen indicating that it determined that all your files in documents would have been transferred but it didn't do it because it was a dry run. You should only have to do this once unless you start getting errors. When next you go into ranger, dropbox.sh (or whatever you called yours) will appear as a selection and you can just select it and hit enter anytime.

If you use Dropbox, this is such an easy and reliable method of getting your drafts to your computer or any device you have that uses Dropbox. A lot of steps in setting it up, but after that, for a Dropbox user, it is very simple. Not as good as a real Dropbox client like Dropsync that can do everything in the background without you even having to think about it, but still quite useful.

Any questions, feel free to contact me or stop by the Alphasmart - Smart Tools forum where I hang out. It's linked on the page where you found this.

Hook