**RFIDbee2 Manual v1 for Hardware v2021-04-13**

William H. Grover • [wgrover@engr.ucr.edu](mailto:wgrover@engr.ucr.edu)

A picture containing text

Description automatically generated

[Startup procedure 2](#_Toc71189802)

[When to use the Reset button 2](#_Toc71189803)

[When to change the batteries 3](#_Toc71189804)

[Checking batteries 4](#_Toc71189805)

[Recharging batteries 5](#_Toc71189806)

[Battery safety thoughts: 6](#_Toc71189807)

[Blink error codes 7](#_Toc71189808)

[Installing Python (needed to set the real-time clock) 7](#_Toc71189809)

[Setting the clock on an RFIDbee2 8](#_Toc71189810)

[Installing the Arduino IDE (needed to reprogram the RFIDbee2 reader) 9](#_Toc71189811)

[Reprogramming the RFIDbee2 11](#_Toc71189812)

[Reading debug messages from RFIDbee2 13](#_Toc71189813)

# Startup procedure

1. Confirm that there’s a rechargeable lithium-ion button-cell battery in the holder on the RFIDbee2’s real-time clock, with the positive side (the one with the writing) facing up / out of the holder. If the battery is missing or loose, you’ll need to set the RFIDbee2’s real-time clock (instructions below)

2. Confirm that there’s an SD card inserted in the card slot. These card slots are meant to be “clicky” like a ballpoint pen – you push the card all the way in once to insert it, and push it all the way in a second time to eject it. But the mechanism is a little creaky and you might want to push the card a few times to confirm that you’ve left it in the “inserted” position.

**3. Connect two Adafruit antenna cards to the RFIDbee2 via two ribbon cables.** There are two different types of antenna cards (the “breakout” and the “shield”) and it doesn’t matter which you use. Make sure the ribbon cable connectors are fully inserted. These cable connectors have a little pin that forces you to insert it in a certain orientation, and the two ends of the cables have connectors attached differently, so if one orientation isn’t convenient, try using the other end of the cable.

**4. Connect the RFIDbee2’s battery cables to a 6V battery.** Make sure you get the polarity of the connections right! Connecting them backward might kill the circuit. The red cable with the red connector goes to the red positive terminal on the battery. The black cable with the blue connector goes to the black negative terminal on the battery. Note that this cable unfortunately had a little color mixing – you might see a little red on the black cable, or a little black on the red cable.

**5. Press the little white Reset button on the RFIDbee2.** After about a second, the green status LED should flash once and then stay off. If you see something different, see error codes later on.

**6. Test the RFIDbee2 by holding an RFID tag against each of the two antennas**. You should see the green status LED blink rapidly while the card is near each reader.

That’s it, this RFIDbee2 should now be good to run for at least 3 days!

# When to use the Reset button

Press the Reset button any time that something about the RFIDbee2 has changed. This includes:

- Removing or replacing the SD Card (press Reset after the card is inserted)

- Disconnecting and reconnecting ribbon cable connectors (press Reset when everything is connected again)

- Swapping out the battery (press Reset after the new battery is connected)

# When to change the batteries

This plot is typical of my testing results, running a “stress test” that makes the RFIDbee2 read an RFID tag once every 5 seconds, 24 hours a day:

Chart, scatter chart

Description automatically generated

The RFIDbee2 needs 5 V to operate. The battery is nominally a 6V battery, but it actually puts out a little more that 6 V when fully charged; that’s OK and typical for these batteries. The plot shows that the battery provides >5 V for at least 3 days, but the voltage output dropped dramatically at about 3.5 days and could no longer power the RFIDbee2 right before the 4-day mark.

Chart, scatter chart

Description automatically generatedSo, how often should we change the batteries? You might need to experiment in the field and see. I hope that the batteries will last longer than 3 days with more realistic use (not the constant scanning that this “stress test” used), but probably the safest plan would be to just change the batteries every 3 days, before the voltage has a chance to drop.

Or, you might want to monitor the battery voltage (see next section) and change the batteries when you see the voltage approaching 5.5 V, which precedes the sudden voltage drop. The plot at right shows another “stress test” experiment, this time taking measurements just around the sudden voltage drop. Again, as soon as the voltage drops below 5.5 V, the sudden drop begins a couple hours later.

# A picture containing text, ground, device, meter Description automatically generatedChecking batteries

You can check the batteries any time, whether the battery is connected to the RFIDbee2 or just sitting there disconnected.

1. Confirm that the multimeter leads are connected to the multimeter jacks labeled **Common** (black) and **V**… (red) as shown at right. We don’t use the “A” jack for anything.

2. Turn the dial to the **V with straight and dotted lines over it** (meaning DC voltage) as shown below.

3. Touch the probe tips to the battery terminals (red probe to red positive terminal, black probe to black negative terminal) and read the voltage from the meter as shown below:

A picture containing ground, meter, device

Description automatically generated

# Recharging batteries

To recharge the batteries, disconnect them from the RFIDbee2 and follow the instructions for 1 battery (using the 6V setting) or 2 batteries (using the 12V setting):

A picture containing ground, outdoor, person

Description automatically generated**If you’re charging just one battery**, plug the charger into the wall, then clip the leads to the battery like this (red clamp to red positive connector, black clamp to black negative connector).

Confirm that **6V** lights up on the charger as shown above. If it doesn’t, you can manually select it by repeatedly pressing the **Mode** button (you might have to hold the button to enter 6V mode). If everything is working OK, you should see the first red light on the charger blinking slowly. The battery should be charged in a few hours. Disconnect everything and you’re done!

**A picture containing ground

Description automatically generatedIf you’re charging two batteries on one connection simultaneously**, first place the batteries side by side as shown here, arranged the same way (reds on the left, blacks on the right).

Then connect the black terminal of the left battery to the red terminal of the right battery using one of the short wires.

Finally, plug the charger in to the wall, and connect the clamps to the remaining (outer) battery terminals: red clamp on the red terminal on left, and black clamp on the black terminal on right.

Confirm that **12V** lights up on the charger (NOTE: we want the plain “12V” light, not the “AGM” or “Lithium” ones). If 12V doesn’t light up, you can manually select it by repeatedly pressing the **Mode** button (you might have to hold the button to enter 12V mode). If everything is working OK, you should see the first red light on the charger blinking slowly. The batteries should be charged in a few hours. Disconnect everything and you’re good to go!

**When the charger is finished charging the battery/batteries**, all four of the lights will be lit up including the big green one on the right:

**A picture containing text, indoor

Description automatically generated**

# Battery safety thoughts:

It’s safe to touch the terminals on these batteries, but I’d avoid touching both at the same time if your hands are wet.

Don’t store the batteries anywhere that they can bump into a metal object that might connect the terminals and short circuit the battery; it might cause a fire.

Supposedly lead-acid batteries can generate hydrogen gas when they’re charging, so people are warned not to charge them in confined spaces like sealed containers.

# Blink error codes

When you power the RFIDbee2 and press the Reset button to start it up, it will flash the green LED to tell you the status of the reader. The different blinking patterns have different meanings:

**1 blink, then stays off:** The reader started up OK. Always confirm correct operation by scanning an RFID tag on each antenna! The green LED should flash rapidly when it reads the tag.

**2 blinks, repeating constantly:** The real-time clock doesn’t know what time it is. Set it using the instructions below.

**3 blinks, repeating constantly:** Problem communicating with the antennas. Confirm that the ribbon cables are plugged in securely.

**4 blinks, repeating constantly**: Problem with SD card (confirm that there’s an SD card inserted).

# Installing Python (needed to set the real-time clock)

If you need to set the clock on an RFIDbee2 (because the button battery was removed, or some other problem), you need to first set up Python on your computer:

1. Download the latest version of Python from <https://www.python.org/downloads/> and run the installer. On the “Installation Type” step, click Custom and confirm that everything is checked:

Graphical user interface, text, application

Description automatically generated

Then complete the installation.

2. Open the **Terminal** app on your Mac and type which python3 and hit Return.

If you see something like /Library/Frameworks/Python.framework/Versions/3.9/bin/python3, that’s great, move on to step 3.

If you see something like /usr/bin/python3 instead, type /Applications/Python\ 3.9/Update\ Shell\ Profile.command and hit Return. Note: you can type a few letters and hit the **Tab** button to autocomplete these commands to save yourself typing the whole thing. After the program finishes, quit **Terminal**, then reopen **Terminal**, then type which python3 and hit Return, and this time you should see something like /Library/Frameworks/Python.framework/Versions/3.9/bin/python3.

3. In the **Terminal** app, type pip3 install pyserial and hit Return. The program should finish without any error messages.

4. You should now have Python and all necessary dependencies installed on your computer and you can now set the clock on an RFIDbee2 (see next section).

# Setting the clock on an RFIDbee2

Before performing these steps, you must have Python installed using the steps above, and the RFIDbee2 must be showing the 2-blink code.

1. Connect the RFIDbee2 to a USB port on your computer, using a cable that you know can carry data as well as power (the cable I supplied is one such cable).

2. Confirm that the RFIDbee2 is showing the 2-blink pattern on the green LED, meaning that it is ready to have its clock set.

3. In the **Terminal** app, cd to the folder where you have the RFIDbee2 code on your computer. For example, if the code is in your Downloads folder, you might type something like

cd /Users/username/Downloads/RFIDbee

substituting your username for username. You can type a few letters and then hit tab to autocomplete this command.

4. In the **Terminal** app, type python3 clocksetter.py and hit Return.

# Installing the Arduino IDE (needed to reprogram the RFIDbee2 reader)

Hopefully you won’t need to do this, but if we need to reprogram the RFIDbee2s in the field, you first need to install the Arduino integrated development environment (IDE) using these steps:

1. Download the latest version of the Arduino IDE for MacOS from <https://www.arduino.cc/en/software> (click “just download” when asked for a donation). In the finder, move the downloaded **Arduino** app to the **Applications** folder on your Mac.

2. Open the **Arduino** app and you should see a window like this:

Graphical user interface, text, application, chat or text message

Description automatically generated

In the **Arduino** app menu, select **Tools / Manage Libraries**. This window should appear:

Graphical user interface, text, application, email

Description automatically generated

Search for and install each of the three following libraries. Also install any dependencies that the program recommends installing. Since there are some similarly named libraries, I’m showing screenshots of each correct libraries to install (note the version numbers may differ, but the author and description of the library should match):

**Adafruit PN532:**

Graphical user interface, text, application, Word

Description automatically generated

**RTClib:**

Graphical user interface, text, application, Word

Description automatically generated

**SdFat:**

Graphical user interface, text, application

Description automatically generated

When you’re done installing those libraries and their recommended dependencies, close the Library Manager.

You’ve now got the Arduino IDE installed and are ready to reprogram the RFIDbee2 in the next section.

# Reprogramming the RFIDbee2

If you need to reprogram the RFIDbee2 (maybe using new code that fixes a bug or something), install the Arduino IDE using the instructions above, then follow these steps:

1. Connect your RFIDbee2 to your computer using a USB cable that can transfer data as well as power (the one I gave you will work).

2. Open up the Arduino IDE app.

3. Select **File** / **Open** and navigate to the “RFIDbee2.ino” file that you want to use. You should see the file open like this:

Graphical user interface, text, application, chat or text message

Description automatically generated

4. Under **Tools** / **Board**, make sure **Arduino Nano** is selected.

5. Under **Tools** / **Port**, select the entry that mentions an Arduino, like this:

Graphical user interface, text, application, chat or text message

Description automatically generated

You might find that you have other ports listed, but there should be only one that mentions an Arduino; click that one. After you do it, there should be a check mark beside it:

Graphical user interface, text, application, chat or text message

Description automatically generated

6. Back in the Arduino IDE window, click the little check mark button (“Verify”) in the upper-left corner of the window:

Graphical user interface, application

Description automatically generated

After a few seconds, you should get a “Done compiling” message at the bottom of the window like this:

Graphical user interface, text, application, website

Description automatically generated

7. Now click the right-facing arrow (“Upload”):

Qr code

Description automatically generated

After a few seconds, you should get a “Done uploading” message at the bottom of the window like this:

Graphical user interface, text, application, chat or text message

Description automatically generated

At this point, the RFIDbee2 has been reprogrammed with the selected code. You can unplug it and start it up as usual (described earlier) and it should be running the new code.

# Reading debug messages from RFIDbee2

During normal operation, the RFIDbee2 sends messages via the serial (USB) connection that can be helpful for debugging problems. Here’s how to read those messages on your computer:

1. Connect your RFIDbee2 to your computer using a USB cable that can transfer data as well as power (the one I gave you will work).

2. Open up the Arduino IDE app.

3. Under **Tools** / **Port**, select the entry that mentions an Arduino, like this:

Graphical user interface, text, application, chat or text message

Description automatically generated

You might find that you have other ports listed, but there should be only one that mentions an Arduino; click that one. After you do it, there should be a check mark beside it:

Graphical user interface, text, application, chat or text message

Description automatically generated

4. Back in the Arduino IDE window, click the little magnifying glass (“Serial Monitor”) in the upper-right corner of the window:

Graphical user interface, text, application

Description automatically generated

This window should appear:

Graphical user interface, text, application

Description automatically generated

Confirm that “9600 baud” is selected in the dropdown box at the bottom.

5. Press the Reset button on the RFIDbee2 and after a second or two, you should see messages appear on the Serial Monitor window. A normal startup should send messages that look like this:

FIXME I need to paste this in.