

---

# **TiltBridge Documentation**

**John Beeler**

**Jun 19, 2020**



---

## Contents:

---

<b>1</b>	<b>What is TiltBridge?</b>	<b>3</b>
<b>2</b>	<b>TiltBridge Hardware</b>	<b>5</b>
2.1	LOLIN D32 Pro + TFT . . . . .	5
2.2	Additional Hardware Required . . . . .	6
2.3	Notes about Previous Versions/Other Hardware . . . . .	6
<b>3</b>	<b>Installing the Firmware on an ESP32</b>	<b>7</b>
3.1	Installation using BrewFlasher . . . . .	7
3.2	Installation using Fermentrack . . . . .	7
3.3	Installation using esptool.py . . . . .	7
3.4	Next Steps . . . . .	8
<b>4</b>	<b>User Guide</b>	<b>9</b>
4.1	Initial Setup . . . . .	9
4.2	Cloud Service Setup . . . . .	10
4.3	Other TiltBridge Features/Settings . . . . .	14
<b>5</b>	<b>Notes for Developers</b>	<b>17</b>





# TILTBRIDGE



---

## What is TiltBridge?

---

**TiltBridge** is a single-component hardware solution for bridging the gap between your **Tilt Hydrometer**'s Bluetooth signal and the internet. View your beer's gravity or transmit your Tilt's signal to the cloud simply, easily, and without needing a spare phone or Raspberry Pi. A TiltBridge can be built without any soldering or electrical know-how, and set up in 10 minutes or less.

A TiltBridge consists of an ESP32-based controller (the *TiltBridge Hardware*) and the TiltBridge Firmware (the software). The hardware can be purchased for as little as \$15 in many cases, and the software is available on GitHub for free.





---

## TiltBridge Hardware

---

The first step in a TiltBridge installation is sourcing the necessary hardware. The only component required for a TiltBridge is an ESP32-based board - however selecting the LOLIN D32 Pro and adding the compatible TFT screen will result in the best experience with the most features.

---

**Note:** While this page documents hardware that has ben demonstrated to work with TiltBridge in the past, the author of TiltBridge does not warrant that any of the hardware listed below continues to be operational. Additionally, although links to vendors are provided to show examples of the referenced hardware, no guarantees are made as to the vendors themselves. Buy hardware you trust from a vendor you trust.

---

### 2.1 LOLIN D32 Pro + TFT

The recommended board for this project is the LOLIN D32 Pro, along with its TFT screen and cable. This set of hardware enables the full suite of TiltBridge functionality without needing to solder anything. Simply add an enclosure and you're done!

#### 2.1.1 Sources

Although the only required component is the board, you will likely also want the TFT screen & cable:

- [AliExpress \(board\)](#)
- [AliExpress \(TFT Screen\)](#)
- [AliExpress \(Cable\)](#)

---

**Note:** For the board, you need one of the 16MB variants. Preference should go to the non-LED models.

---

## 2.2 Additional Hardware Required

In addition to the ESP32 board, you will also need a USB power source to power the ESP32 (such as a phone charger), a MicroUSB cable for flashing, and you will likely want some kind of enclosure (see below).

Once you have acquired the necessary hardware, you are ready to begin *Initial Setup*

### 2.2.1 Enclosures

Although not required, an enclosure is recommended. There are a number of designs out there depending on the board you choose.

#### D32 Pro Cases

- [D32 Pro + TFT Case by gromitdj](#)

#### OLED Cases

- [TiltBridge TTGO ESP32 OLED Enclosure by Thorrak](#)
- [Generic “OLED” Enclosure by Thorrak](#)
- [HiLetGo Wifi 32 Tiltbridge Case by calandryll](#)

## 2.3 Notes about Previous Versions/Other Hardware

TiltBridge is designed to run either with or without an attached screen, and therefore can be used with a variety of other boards. Although this functionality exists, due to the myriad issues experienced with OLED boards, non-LOLIN boards, etc. the use of boards other than the LOLIN D32 Pro are not currently recommended for new builds.

If you use another board, make sure to select the appropriate firmware:

- “OLED” boards, those with less than 16MB of flash, and those lacking a screen entirely should select the “OLED” firmware variant
- Any non-“OLED” board with 16MB of flash or greater using an attached ILI9341-driven TFT should select the “TFT” firmware variant

---

## Installing the Firmware on an ESP32

---

Installing TiltBridge on an ESP32 can be accomplished using either [BrewFlasher](#), [Fermentrack](#), or the Espressif `esptool.py` script.

### 3.1 Installation using BrewFlasher

The easiest way to install TiltBridge is using [BrewFlasher](#). BrewFlasher is a standalone application for Windows and Mac OS which allows flashing the same set of firmware that is available through Fermentrack without having to delve into the command line. It is free, easy to use, and is recommended for most users.

To install using BrewFlasher simply connect your ESP32 to your computer, download and open BrewFlasher, and follow the prompts on screen.

### 3.2 Installation using Fermentrack

Installation via [Fermentrack](#) is incredibly easy as it leverages the existing “firmware flash” workflow. To flash via this method simply log into your Fermentrack installation, choose “Flash Device” from the devices menu, select the “ESP32” device family, and follow the prompts. When prompted to choose a firmware to flash simply choose “TiltBridge”.

### 3.3 Installation using esptool.py

TiltBridge can also be installed using `esptool.py`. Although slightly more involved, this method does not require the use of a Raspberry Pi running [Fermentrack](#).

---

**Note:** The command below will automatically detect & flash any Espressif chips connected to the computer running `esptool.py`. Please leave only the device you are setting up connected to the computer to avoid the risk of misflashing.

---

1. Install esptool.py using the [instructions here](#).
2. Download all **five** of the appropriate TiltBridge firmware files for your controller from [GitHub](#) and rename them to the following - spiffs.bin, firmware.bin, partitions.bin, boot\_app0.bin, and bootloader\_dio\_40m.bin
3. Connect the ESP32 board to the computer you will be using to flash
4. Open a command prompt and run the appropriate flash command

The appropriate flash command for your build depends on the version firmware you are using. You may need to tweak the command, but sample flash commands are as follows:

- **“TFT” Firmware:**

```
esptool.py --chip esp32 --before default_reset --after hard_reset write_flash 0xe000 boot_app0.bin 0x1000 bootloader_dio_40m.bin 0x10000 firmware.bin 0x8000 partitions.bin 0x910000 spiffs.bin
```
- **“OLED” Firmware:**

```
esptool.py --chip esp32 --before default_reset --after hard_reset write_flash 0xe000 boot_app0.bin 0x1000 bootloader_dio_40m.bin 0x10000 firmware.bin 0x8000 partitions.bin 0x310000 spiffs.bin
```

### 3.4 Next Steps

Once your controller is flashed, secure it in its case/enclosure, and proceed to the *Initial Setup* process.

### 4.1 Initial Setup

Once you have acquired the necessary hardware, you are ready to flash the TiltBridge firmware to your controller and set it up for use on your network.

#### 4.1.1 Flashing the TiltBridge Firmware

The TiltBridge firmware can be downloaded from [GitHub](#) and flashed to your controller using either [BrewFlasher](#), [Fermentrack](#), or your favorite ESP32 flashing software. The easiest method to flash the firmware is using [BrewFlasher](#) - a desktop firmware flashing application maintained by the author of TiltBridge.

---

**Note:** Be sure to flash the right version of the firmware. For new builds using the LOLIN D32 Pro you will want the “TFT” firmware.

---

#### 4.1.2 Setting up TiltBridge

Once you have flashed the firmware to your ESP32, secured it in its enclosure, and powered it on, you are ready to connect it to your WiFi network. After the TiltBridge is powered on it will create a new WiFi access point with the SSID `TiltBridgeAP` and password `tiltbridge`. Connect to this AP using a phone, laptop, or other WiFi enabled device, and attempt to navigate to any website using a web browser in order to see the configuration panel.

1. Flash & power on the TiltBridge ESP32
2. Using a WiFi-enabled phone, laptop, or other device, connect to the wireless network with the SSID `TiltBridgeAP`
3. Once connected, the configuration screen may automatically appear. If it does not, open a web browser and attempt to navigate to any webpage.
4. When the configuration screen appears, click `Configure WiFi`

5. Select your wireless network from the list of available networks and enter the password in the `Password` field beneath the list of available networks
6. Choose a name to identify your TiltBridge and enter it in the `Device (mDNS) Name` field. The default name `-tiltbridge` can be used unless you have multiple TiltBridge devices in which case each must have a unique name.
7. Click `Save`
8. Reconnect your phone/laptop/etc to your regular wireless network

Once these steps are complete, your TiltBridge should connect to your wireless network and is ready to begin transmitting your Tilt's data over WiFi.

## 4.2 Cloud Service Setup

Once your tilt has completed the *Initial Setup* process it is ready to begin scanning for [Tilt Hydrometers](#) and relay their gravity readings to the cloud. At the moment, TiltBridge supports three cloud service targets for this data:

- [Fermentrack](#)
- [Brewers Friend](#)
- [Google Sheets](#)
- [Brewfather](#)
- [BrewStatus](#)

### 4.2.1 Setting up TiltBridge for Fermentrack

The TiltBridge was designed by the primary author of [Fermentrack](#) and is designed with Fermentrack support in mind. In most cases, your TiltBridge can be linked to a single Fermentrack installation from directly within the Fermentrack app as part of the “Add Gravity Sensor” workflow. The configuration can also be re-sent to TiltBridge by clicking the

“Update TiltBridge Automatically” button from the “Manage Sensor” page for the Tilt Hydrometer linked to the TiltBridge.

### Adding the TiltBridge to Fermentrack

1. Connect to your Fermentrack installation and log in (if necessary)
2. From the `Device` menu choose `Add Gravity Sensor`
3. On the resulting page, choose `Tilt Hydrometer` and then select the connection type `TiltBridge`
4. Click the `Add New TiltBridge` link. Fermentrack will automatically scan for TiltBridge devices on your network.
5. If your TiltBridge device appears, click the “Add” button. Fermentrack will attempt to complete the remaining setup steps automatically. Proceed to Adding Tiltbridge-connected Tilts to Fermentrack
6. If your TiltBridge device **is not** in the list you will need to add it manually. In the form at the bottom of the page, enter the mDNS ID you selected during *Initial Setup* and enter a name by which Fermentrack will identify your TiltBridge.
7. Click the `Create TiltBridge` button. Fermentrack now knows to expect your TiltBridge, but you will need to finish the setup on the TiltBridge manually.

## If you need to manually configure the TiltBridge

If Fermentrack was not able to automatically configure your TiltBridge (either because the automatic configuration failed or because you created the TiltBridge manually) you will need to manually set the Fermentrack URL on the TiltBridge settings page.

1. Identify the IP address of your Fermentrack installation
2. On a device connected to the same network as the TiltBridge, navigate to <http://tiltbridge.local/> (replace tilt-bridge in this URL with the mDNS name you set during initial setup)
3. Click the `Settings` link at the top of the dashboard
4. In the `Fermentrack URL` field enter [http://<fermentrack\\_ip\\_address>/tiltbridge/](http://<fermentrack_ip_address>/tiltbridge/) and click `Update`

Fermentrack and TiltBridge are now configured to properly communicate. You can now set up Tilts connected using TiltBridge in Fermentrack.

## Adding TiltBridge-connected Tilts to Fermentrack

Once the TiltBridge and Fermentrack are configured to communicate, you can add the Tilt hydrometers to Fermentrack that will be pushed via the TiltBridge.

1. Connect to your Fermentrack installation and log in (if necessary)
2. From the `Device` menu choose `Add Gravity Sensor`
3. On the resulting page, choose `Tilt Hydrometer` and then select the connection type `TiltBridge`
4. Select the TiltBridge you just set up
5. Enter the name, temp format, and color, and click `Create Sensor`

Congratulations - your Tilt will now send gravity readings to Fermentrack.

## 4.2.2 Setting up TiltBridge for Brewers Friend

With a premium [Brewers Friend](#) account, you can store temperature and gravity readings from your Tilt Hydrometer as part of your brew logs. After you set up your TiltBridge to push to Brewers Friend, it will create stream devices for each of the detected Tilts which can then be subsequently attached to your brew logs. To comply with rate limits placed by Brewer's Friend, data is first sent about 1 minute after startup, and subsequently no more than once every 15 minutes.

---

**Note:** Due to ESP32 hardware limitations, all connections to Brewers Friend are done over HTTP

---

## Configuring the TiltBridge for Brewers Friend

To configure TiltBridge to use Brewer's Friend, you will need to obtain your Brewer's Friend API key and provide it to TiltBridge.

1. Go to [Brewers Friend](#) and log in to your account
2. Click the `Profile` button in the upper right, and click `Account`
3. Copy the API key (not one of the URLs) to your clipboard

4. On a device connected to the same network as the TiltBridge, navigate to <http://tiltbridge.local/> (replace tilt-bridge in this URL with the mDNS name you set during initial setup)
5. Click the `Settings` link at the top of the dashboard
6. In the `Brewers Friend API Key` field paste the API key you copied earlier and click `Update`

### Testing Brewers Friend

Once the Brewer's Friend API key is provided, the TiltBridge will begin transmitting gravity data once every 15 minutes. Following the first data transmission, you can easily check in Brewer's Friend to see if the data was received.

1. Go to [Brewers Friend](#) and log in to your account
2. Click the `Profile` button in the upper right, and click `Account`
3. Click the `Device Settings` button on the right
4. In the `Device Settings` pane click `Show All` (to the right of `All Brew Sessions`)
5. All Tilt devices should appear in the list as `Stream` devices and will be identified by their color

---

**Note:** Per Brewer's Friend guidelines, data is only pushed once every 15 minutes.

---

### 4.2.3 Setting up TiltBridge for Google Sheets

Similar to TiltPi or the Tilt Hydrometer app, TiltBridge supports logging to Google Sheets. Setting up Google Sheets is more involved than either [Fermentrack](#) or [Brewers Friend](#), but provides a free, easily accessible cloud data service.

#### Preparing Google Sheets to Receive Data

The first step in Google Sheets integration is preparing Google Sheets to receive the data. Google Sheets requires the use of a Gmail or Google Apps for Domains account. This preparation step only needs to be done once per Google Account and once complete can be used with multiple Tilts, TiltBridges, and beer logs.

1. Open the [TiltBridge Master Sheet](#) on Google Sheets. If prompted to login, log into your Gmail or Google Apps for Domains account.
2. As prompted, go to the `File Menu`, and select `Make a Copy`
3. Enter a name for this sheet (it will be the master sheet to coordinate all your beer logs) and click `OK`
4. The copy you just made will open in a new window. Go to the `Tools` menu and select `Script Editor`

This will then open the Google Script editor. Before you can begin logging to Google Sheets, you need to “publish” the script that will receive the incoming data. To do so, you can follow these instructions (which are the same as those at the top of the window that appears):

1. Go to the `Publish` menu and select `Deploy as web app...`
2. In the dialog box that appears, set `Who has access to the app:` to `Anyone`, even anonymous and click `Deploy`
3. A dialog box will appear. Select `Review Permissions`. Another dialog box will appear. Select your Google Account.
4. A dialog box with `This app isn't verified` will appear. Select `Advanced` then select `Go to Tilt Cloud Template for TiltBridge (unsafe)`



5. A dialog box with permission requests will appear. Select “Allow”.
6. A dialog box confirming the app has been published will appear. Click OK, then Close Script Editor and return to the Google Sheet.
7. Use the new “Tilt” menu in the Google Sheet to view the cloud URL or email it to yourself.

### Configuring the TiltBridge for Google Sheets

Once you have prepared Google Sheets to receive data pushed by the TiltBridge, you will need to update a handful of settings on the TiltBridge itself so that it knows where to send the data.

1. On a device connected to the same network as the TiltBridge, navigate to <http://tiltbridge.local/> (replace tilt-bridge in this URL with the mDNS name you set during initial setup)
2. Click the `Settings` link at the top of the dashboard
3. In the `Google Script URL` field paste the Google Script URL you made note of during the preparation step above and click `Update`
4. After the URL is updated, enter your Gmail (or Google Apps) email address in the `Google Script Email` field and click `Update`

Your TiltBridge should now be configured to send data to Google Sheets. To begin logging a Tilt you will need to enter a sheet name for the data to be logged to.

### Logging a Beer with TiltBridge and Google Sheets

After configuring the TiltBridge each Tilt your TiltBridge detects can be individually logged to its own sheet on Google Sheets. To enable logging, you will need to specify a sheet name in TiltBridge. If this sheet does not exist on Google Sheets it will be created. If the sheet does exist, new data points will be appended to it.

1. On a device connected to the same network as the TiltBridge, navigate to <http://tiltbridge.local/> (replace tilt-bridge in this URL with the mDNS name you set during initial setup)
2. Click the `Settings` link at the top of the dashboard
3. Type a name for the Google Sheet to which you want to log data points in the appropriate `Tilt Sheet Name` field and click `Update`

Once this is complete, your Tilt will begin logging data points to the sheet name you specified. If the sheet does not exist, it will automatically be created.

---

**Note:** Points are only pushed to Google Sheets once every 10 minutes. As a result, it may take up to 10 minutes for the sheet to be created or for the first points to appear.

---

## 4.2.4 Setting up TiltBridge for Brewfather

With a premium [Brewfather](#) account, you can store temperature and gravity readings from your Tilt Hydrometer as part of your brew logs.

### Configuring your TiltBridge for Brewfather

TiltBridge will need your Brewfather stream ID in order to post to your Brewfather account.

1. Go to [Brewfather](#) and log in to your account
2. Click the `Settings` option in the menu on the left
3. Under “Power-ups” in the lower left corner, click the “switch” next to “Custom Stream” if it is not already toggled
4. Copy just the string of letters/numbers that appears after the start of the URL (<http://log.brewfather.net/stream?id=>) to your clipboard
5. On a device connected to the same network as the TiltBridge, navigate to <http://tiltbridge.local/> (replace tilt-bridge in this URL with the mDNS name you set during initial setup)
6. Click the `Settings` link at the top of the dashboard
7. In the `Brewfather Stream key` field paste the string you copied earlier and click `Update`

### Testing Brewfather

Once the Brewfather Stream key is provided, the TiltBridge will begin transmitting gravity data once every 15 minutes. Following the first data transmission, you can easily check in Brewfather to see if the data was received.

1. Go to [Brewfather](#) and log in to your account
2. Click the `Devices` option in the menu on the left
3. All Tilt devices should appear in the list as `Custom Stream` devices and will be identified by their color

---

**Note:** Per Brewfather’s guidelines, data is only pushed once every 15 minutes.

---

### 4.2.5 Setting up TiltBridge for Brewstatus

TiltBridge is designed to allow for data to be pushed to [Brewstatus](#). This functionality was helpfully added by contributors to the project on [GitHub](#) and has not yet been documented.

---

**Todo:** Document Brewstatus configuration

---

## 4.3 Other TiltBridge Features/Settings

TiltBridge also has features to simplify both keeping your TiltBridge up to date as well as resetting the WiFi network settings.

### 4.3.1 Updating your TiltBridge Firmware

Updates to TiltBridge can be easily performed by flashing the latest available firmware from [GitHub](#) using either [BrewFlasher](#), [Fermentrack](#), or your favorite ESP32 flashing software.

### 4.3.2 Resetting your WiFi Configuration Settings

There are two options for resetting your TiltBridge's WiFi configuration settings.

If you are able to log into your TiltBridge using a web browser, do so, and then click the “x” option at the top of the page, then click the “Reset WiFi Settings” button. This will cause your TiltBridge to delete its network configuration & allow it to be configured using the TiltBridge configuration AP.

---

**Todo:** replace “x” in the above

---

If you are not able to log into your TiltBridge using a web browser, you can also reset the WiFi settings using the “boot” button (if available) on your TiltBridge device. To use this, simply push the “boot” button on the TiltBridge while the device is scanning for Tilt Hydrometers. After you've pushed the button once, press it a second time within several seconds to confirm that you want to reset your WiFi settings.

---

**Todo:** Add note about simulating the boot button

---



---

## Notes for Developers

---

Interested in extending TiltBridge or integrating it into your project? Awesome!

While TiltBridge was developed with a certain feature set in mind, TiltBridge is open source for a reason. There are plenty of features that I couldn't imagine which would make the project even better than it is. That said, prior to developing a new feature and submitting a pull request, I strongly recommend reaching out. Due to the current state of the Arduino framework for ESP32, TiltBridge currently uses 99.2% of the *expanded* program partition as well as the majority of the ESP32's available RAM and as such new features are likely to face considerable space constraints. My hope is that as the ESP32's Bluetooth drivers are optimized, both the RAM usage and the size of the compiled binaries will shrink allowing for additional code to be added.