LINUX VOICE TUTORIAL

BREW PERFECT BEER WITH HELP FROM THE RASPBERRY PI

GRAHAM MORRISON

We love beer, we love the Raspberry Pi and we love the Arduino – so we're bringing them together for one awesome project.

7 STEPS TO BEER

- Brewing
- Cooling
- Fermenting
- Priming
- Bottling
- Ageing
- Drinking

GENERAL LINUX

HARDWARE

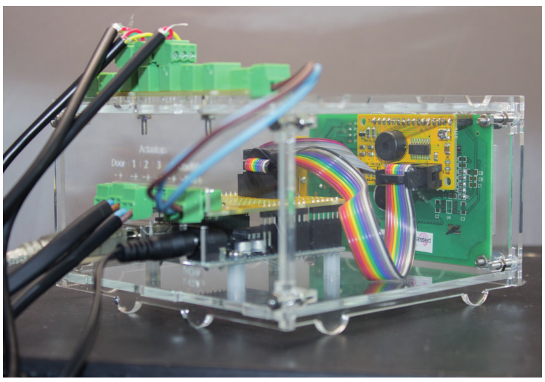
DIFFICULTY

DISCLAIMER

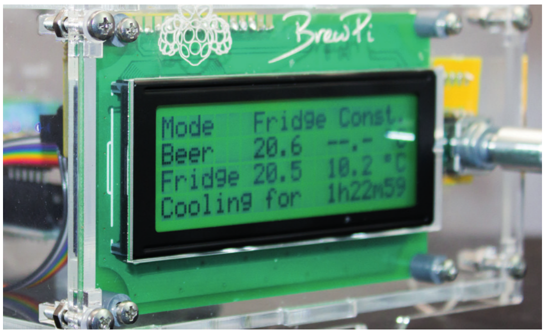
The following tutorial mixes liquid, electricity and DIY modifications, all of which can create a lethal cocktail of danger. Don't make any modifications yourself unless you're certain they're safe, and get a qualified electrician to check any modifications you do make.

Beer is lovely. But when you're making it at home, the biggest challenge (after discovering a way to boil vast quantities of water) is always finding somewhere to leave your brew to ferment. It's this stage of beer-making magic that turns what's known as wort into beer, creating alcohol and oodles of flavour. And for this stage to work well, you ideally need to be able to manage the temperature of the environment your beer is sitting in. In the UK, many amateur brewers resort to using an 'airing cupboard', normally situated next to the hot water tank and used for drying clothes. This isn't a bad place, because it's warmish – many beer kits like to ferment at around 20 degree centigrade – and the temperature doesn't fluctuate massively. But it still fluctuates, and it may even prove too warm. Many yeasts, especially for ale, prefer things a little cooler (18–20 degrees, ideally, but this depends on the beer). And lifting 25 litres of wort into a first-floor cupboard could break your back, and you've got a hygiene nightmare if it falls over, or falls through the flimsy shelf its sitting on.

BrewPi is the answer to this conundrum. It's a brilliant project that brings together a love of Linux, a little hardware hacking and plenty of beer into one fermenting barrel of hoppy goodness. It's essentially a device that controls the environment surrounding the fermenting bucket of beer, enabling you to make perfect beer every time, regardless of climate and house heating cycles. Many people use an old fridge or freezer as the surrounding container and connect the BrewPi to a cooling and heating mechanism to enable its clever algorithms to create the perfect environment for your beer. The BrewPi itself is a mixture of hardware, software and initiative. Not only



This shows the rear of the LCD connecting to the Arduino and the shield, with the OneWire connector above.



The various bits of the BrewPi give little indication that they can be put together to create something awesome.

has its creator, Elco Jacobs, built an incredibly effective system for fermenting beer, he's created an extremely helpful community of BrewPi enthusiasts, an online shop and an assembly system for easy access to all of the bits and pieces you'll need.

What you'll need

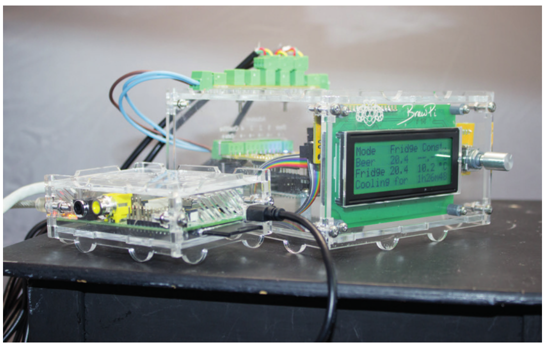
While you will need a fair bit of kit, it needn't cost very much. The fridge or freezer is the biggest consideration, as well as somewhere to put it. We asked the internet, and Mark Einon in Wales very generously obliged with a freezer he was going to give to the local freecycle initiative (thanks Mark!) Almost any fridge or freezer will do, as long as it's working, and you should be able to find someone willing to let an old model go for very little. You need enough space within the freezer to stand your fermenting bin, and as our freezer's shelves were made from coolant pipes, we had to bend these back before there was enough room. Fortunately, the pipes were easily pushed back. We then slotted in an old wooden shelf to stand the fermenting bucket on, as they can be very heavy when full of 25 litres of brewing beer.

If the fridge or freezer has an inside light, this can be coerced into another essential task – heating up the inside environment. If not, you'll need some other kind of heating mechanism. Some people use a reptile mat wrapped around the fermenting bin, but we plumped for a 60W waterproof greenhouse heating bar, which cost us £15 new on eBay, and slotted nicely into the bottom of the freezer with plenty of room. You will also need both a Raspberry Pi, complete with a > 2GB SD card, and either an Arduino Duo or an Arduino Leonardo microcontroller. If you're anything like us, you've got an old Duo tucked away in a drawer somewhere and a Raspberry Pi going spare. And despite the name of the project, there's no specific reason for requiring a Raspberry Pi – any Linux device with a USB port capable of running the Apache web server and some Python scripts should be up to the job. You might want to try a NAS, for example, if you're running one already. But the Pi is well suited to being tucked away in the garage, and it's relatively cheap, so it's still a great option. Most of the hard work is done by the Arduino, as this interfaces with the various sensors and relays and runs the complex controlling algorithms that adjust the temperatures within your freezer. Your brew will even keep brewing if the Pi crashes, which is handy if there's a power failure and your Pi develops a read/write error. The Pi is really just logging and serving up the data for the web portal.

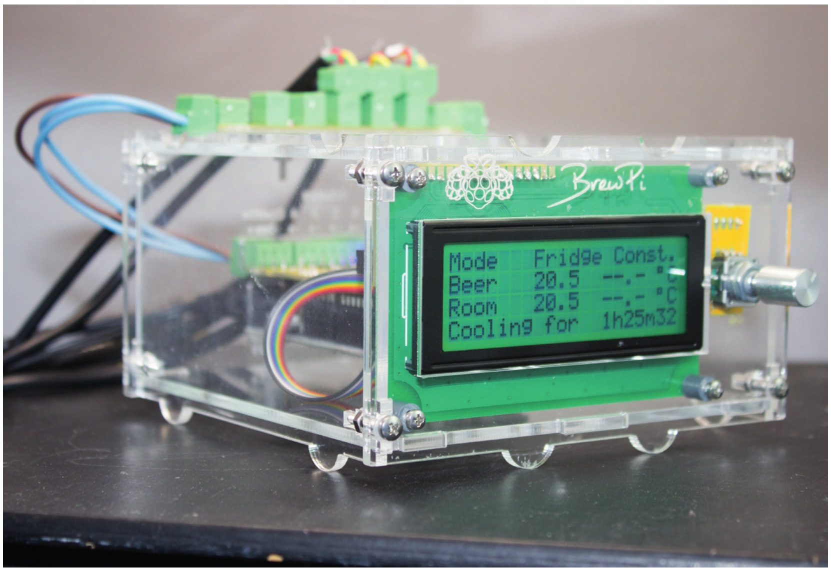
Unless you're an expert who's happy building circuits, you'll also need the BrewPi kit (**brewpi.com**). This includes everything you need to turn your Arduino into a sensor-wielding beer factory. It includes the shield, a PCB that slots onto the two compatible Arduino form factors, along with the LCD, the sensors, the actuators (more details later if none of this makes sense) and the other fiddly bits that may otherwise take an afternoon to source. It's even possible to buy the whole thing pre-constructed, but we think that's missing half the fun, especially when the build itself isn't that difficult.

We'd also highly recommend buying the case kits. These lasered bits of plastic encase both your Raspberry Pi and your Arduino to create a sleek, professional solution that looks great sitting atop your freezer. They also stop bits getting bashed about or falling off. Expect to pay around £70 for the shield and case kits together. You'll also need a miscellany of common tools to put the whole thing together; a soldering iron and solder, maybe a solder sucker, some tweezers, a range of differently sized screwdrivers and a steady hand.

Did we just say soldering iron? Yes! You'll need to solder the various components on to the Arduino shield. But it's straightforward, and this should make an ideal first project if you've not done any soldering before. All the components are large and there's no fiddly soldering required. Try watching a couple of YouTube soldering videos to familiarise yourself with



The Raspberry Pi can also fit on top of the BrewPi case, in a separate box or *au naturel*. Cases are good.



The BrewPi isn't an easier way of making beer. It's an easier way to make it perfect.

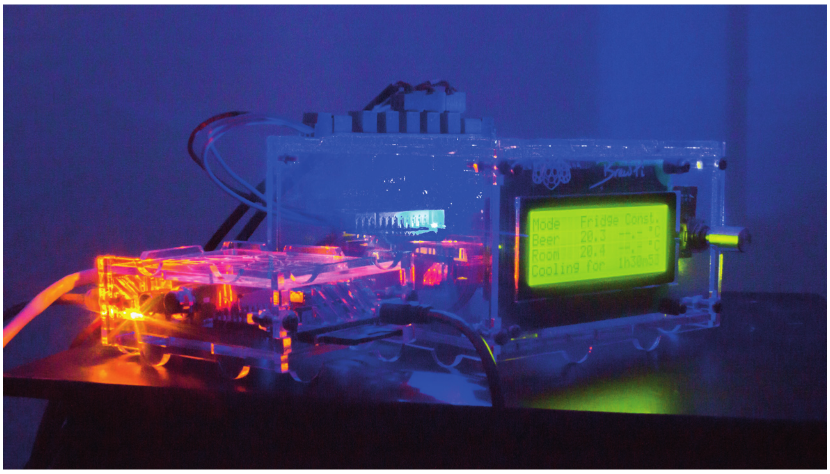
the process first, and then experiment a little with an old circuit board and some wire. You'll then be set for the main event.

The shield is the bit that attaches to the Arduino, and it's probably the most complex part of the whole assembly, so let's get this out of the way first. The main instructions can be found at **www.brewpi.com/brewpi-soldering-guide**, but we're going to cover the broad detail of the process, along with any particular notes we make along the way. The official instructions are made up of photos, and while they're great if you know what you're doing, we want to make the project as accessible as possible by making fewer assumptions about the builder than the official site.

Forging the shield

First, lay out all the components on a table top, grouping them together so you can check they're all there. This also makes it easier to install. Now start by being brave – you've got to snap the shield apart into four separate boards. It's a little like breaking bonfire toffee. The large board that breaks off (labelled with **www.brewpi.com**) connects directly to the Arduino. Then there's a long strip embedding seven columns of three holes, a medium-sized rectangle of a board with a surface mounted integrated circuit, and a tiny rectangle that will host the rotary encoder.

Break off the broken tabs remaining on the boards with a pair of pliers or a small pair of cable cutters so that the edges are as smooth as possible. Some of the pin arrays – the ones with the two collars of black plastic – are designed to fit on to your Arduino board so that it can connect to the holes on the shield. There are five of them, and you should find there's one for every header on the Arduino. These need to be connected to the Arduino first, before being soldered into the shield – this locks their orientation and connection. The longer pin goes into the Arduino, while the shorter piece goes into the shield. As we were using an ancient Arduino Uno, there were fewer power headers on the circuit board that pins allocated,



Red or blue LEDs on the shield indicate whether the BrewPi is currently heating or cooling your brew.

LV PRO TIP

Soldering tips; heat up the destination first, dab the solder onto the joint, make sure it flows into the joint naturally and try not to bridge any connections. If you do, heat and remove using a solder sucker.

but the eight-pin array still fitted over the power pins and the 10-pin header still fitted across the IO pins without getting in the way of everything coming together. Don't forget there's also smaller six-pin rectangular connector. Fortunately, the shield only fits one way. Start your soldering at the corners to make sure all the pins stay aligned.

Now solder the single green connector onto the ACT1–ACT 4 shield holes, with the component attached to the side with the website URL. Connect a three-pin green connector to one side, and one of the two-pin connectors to the other (they all offer ports at right angles to the board, and have the same connector form factor as the eight-pin one you've just connected). Ours wobbled slightly while fitting them, so it's best to solder one of the middle pins first and wiggle the connector into alignment, before soldering any remaining pins. Flip the shield over and solder one of the 10-pin block connectors to the header labelled "To the LCD backpack", and make sure you've got the gap in the right place (facing the edge).

That's all that needs to be done to the main board! Congratulations. Now might be a good time for a cup of tea before moving on to the LCD backpack itself.

Glowing electronic display

The LCD board is the one with the small integrated circuit already on it. The circular speaker fits into the middle with the upwards side on the same side as the chip, and after soldering, you need to cut the protruding pins from the other side. Another 10-pin header comes next, with the gap facing the integrated circuit. Flip this small board over (to the side without any components), and fit the 16-pin header into the holes. Solder from the other side.

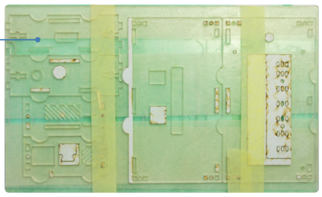
The tiny board for the rotary encoder is up next. The official instructions mention that the biggest two pins on the encoder need to be squeezed slightly to fit into the holes. We didn't need to do this, but we did need to use a fair amount of strength to get the encoder into position. Make sure the side with the handle is the one with the circle on the board, and solder the joints from the other side. A washer, a nut and then the handle can be slipped over the encoder when you've finished.

Next is what's known as the OneWire distribution board (the only board remaining). Sometimes it's written as '1-Wire', and it's a standard protocol for communicating with devices from Dallas Semiconductor (such as the temperature sensors we need for our BrewPi), using a single connector, hence its name. This needs seven of the three-pin green connectors – two shaped at right angles for the edge connectors, and the other five directly pointing up (you can see this illustrated on the board itself now you know what to look for, and that's the side they need to be connected to). Official instructions suggest starting with the two outer connectors, as these are oriented outwards lengthways. The other five all face upwards with their pins on the left when you're looking at the text on the board. The green 'AT-AT' connectors (for that is what they look like, not an official designation) then plug into these and the two end connectors.

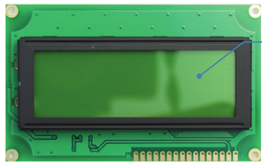
Now it's the turn of the rainbow-coloured ribbon cable, which we need to turn into something a little more civilised to enable it to connect to the ports we've been soldering. If you've ever made your own IDE cable for an ancient PC, this is very similar. The black plastic connectors that attach themselves to the ribbon cable have teeth that penetrate the insulation on the outside of the wire to make a connection without soldering anything. Just make sure the triangles on the connector align with the black wire in the flat cable. Push the cable through until it just protrudes from the other side, and taking the advice of the official instructions again, place the smaller edge on a table and use something flat to put considerable pressure onto the connector. It should just about come together, and in so doing, connect the pins to the cable. When this seems secure, fold the long end of the cable up and over the back of the connector before sliding the remaining black connector to hold the cable together. This needs to be done on both sides of the ribbon cable, and both connectors need to point the same way so that the cable won't twist. That last bit can be a little mind bending as you try to work

We had to bend one of the shelves in our freezer to make enough room for the fermenting bin.

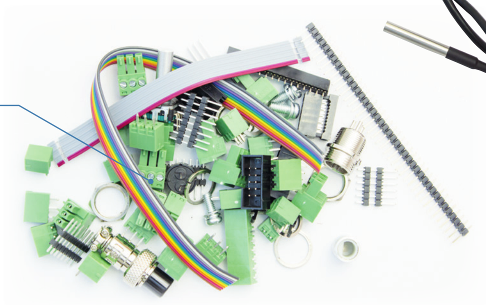
THE BREWPI SURVIVAL KIT



The flat packed Raspberry Pi and Arduino shield cases.

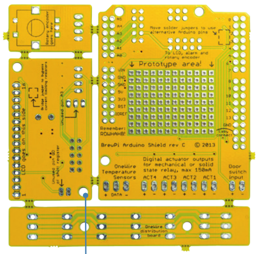


The LCD, which fits into the hole in one of the case panels.



Shield parts are mostly soldered onto the shield, but our kits had a few bits left over.

Temperature sensors are used to measure the beer temperature, the freezer temperature and the outside temperature.



The shield itself.

out which way to put the connector on so that the black cables stay in the same place and the connector is pointing in the same direction after you've twisted the cable back over the connector. You can now connect both of the boards with the correctly sized connector together with the cable, and we felt slightly more optimistic after testing the continuity of the connections to make sure we'd pushed through the connectors to the ribbon cable with enough pressure.

For the other ribbon cable, pull off the ends where they've been cut and wiggle this into the underside of the rotary controller board. Pin 4 should always be red. Then solder the pins to the board, The other end of this cable goes to the LCD board, parallel to the rainbow ribbon cable, and connected to the same side. Make sure pin 4 lines up and solder this as well.

The next stage is the LCD, and you first need to break off 16 pins for the LCD itself. The official guide has a great tip, where you connect the whole header to the female header on the other board and use this as a guide for snapping the 40-pin header at the right place with your hands. This didn't quite work for us, as we broke the header one pin short, but it was easy enough to solder the lone pin alongside the others. Solder these pins on the top surface (the same side as the LCD itself), and you can now attach the LCD to the female header.

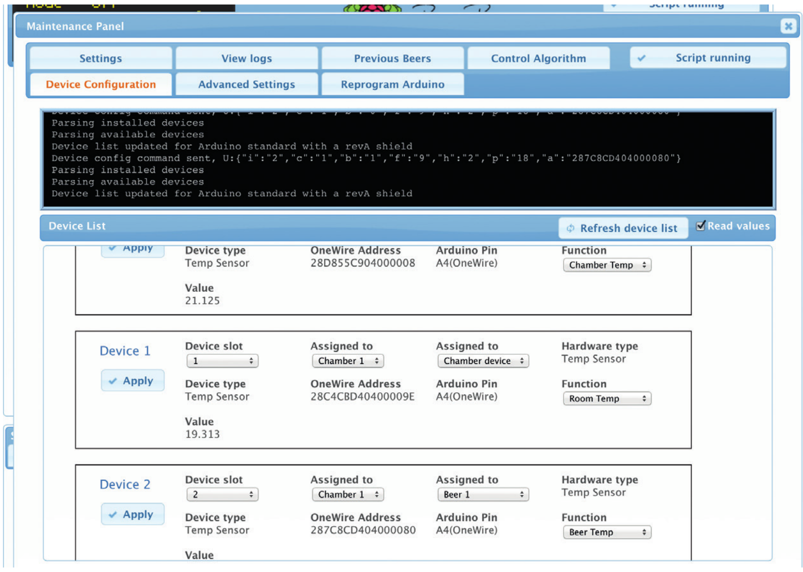
To make the sensors inside the fridge easily removeable, use a connector like this within a container.

The final stage of shield forging is to take the sensors and strip the insulation off the end of the wires – a couple of millimeters will do. Each cable has three 'cores', and each core needs to be screwed into a three headed 'AT-AT' green connector, so that when these plug into the OneWire board, red is at the top (marked 5V – this is important), and yellow at the bottom. The official instructions note that the colour order of the yellow and green wires has changed, so it's worth making doubly sure if you're reading this in the distant future, as the sensors might not be able to take 5V going in the wrong cable. To make the ends of the wires easier to insert into the tiny screw holes, and to make them more resilient, it's worth dabbing them in a little molten solder.

Porter, Stout, IPA – and the case

You now have a choice. You can either keep the OneWire connector close to the rest of your BrewPi hardware, or place it closer to where the sensors are going to be. This might be useful if you wanted to position the OneWire board within the fridge, for example, but we decided to go with the official instructions and wire up a short three-core cable (maybe 20cm), with AT-AT connectors at either end, to connect the OneWire board to the BrewPi. We used an old power cable with earth for easy access to three cores with insulation attached. This cable eventually loops outside the case from the main board to the OneWire connector.

The cases are all made from various bits of lasered plastic, and it's never clear exactly what goes where. It's like a BrewPi 3D jigsaw puzzle. The Raspberry Pi case is a good place to start, as this is emblazoned



You can check your sensor devices are working by enabling the 'Read values' option before refreshing the device list.

with the Raspberry Pi logo flanked by some hops, and it's also obvious which way the pieces should go when you attempt to fit your Pi into the case. The feet of all the cases are half-circles, which is another good way of orienting yourself with the 13 or more pieces used to construct each of the cases.

As we we're using an early Pi, lacking holes on the PCB, there's no way of mounting the board inside the case. The official instructions show a couple of spacers and screws mounting the Pi to the lower case panel. Our case design didn't have a hole even if we did want to connect the Pi. But thanks to the various prominent ports and connectors on the Pi, it was held firmly in place regardless. One side has the video and audio connectors, the opposite just an HDMI connector. Lengthways, theres a micro USB at one end and USB and Ethernet at the other. It's also a good idea to push out any of the small bits of plastic that are used to create airflow through the case, as the Pi can be prone to overheating, but we couldn't remove some of these pieces as they weren't separated enough from the borders of the plastic. This may have been why two extra end pieces, with all the bits removed, were hidden away in one of the part bags.

It all goes together easily enough when you've worked out up and down and where each side fits. Be careful with the side containing the HDMI connector, as it's not immediately obvious when it aligns and you may not notice it's reversed until the end. When you've got everything held together, you've got to now use the long screws, two at each long end, to go through a washer, then into the case, and then through a nut you hold in the small vertical gap before tightening the whole thing up. It's fiddly and frustrating, so we'd suggest focusing on the beer.

Construction time again

This leaves you with significantly fewer bits to worry about for the other case, which is going to contain our BrewPi shield. Now, for some reason, our case is a hybrid of an earlier revision with a few differences between both the earlier version and the 2.0 cases, so there's no point telling you how to put the case together. In fact, the 3.0 case was announced in January, and is smaller again. We were able to make it up as we went along because it's much easier than building the shield, and mostly common sense. There

POWERING THE BREWPI AND UPDATING THE FIRMWARE

Before we move on to software, you need to give some consideration to how you're going to power both the Raspberry Pi and the Arduino. In theory, you could power the Arduino from the Raspberry Pi's USB, using only a single hub or adaptor. We tried this with as many milliamps as we could muster, but the LCD on the Arduino still dimmed when we did anything. Rather than take any risks with our beer, we decided to power both separately. As we all know, the Raspberry Pi is very susceptible to irregularities in power, so it's best not to take any risks – use a high amperage USB hub or adaptor for the PI, and an appropriate adaptor for the Arduino.

It's now time to test whether your soldering skills have been good enough, and to stretch a few of those Linux skills too! The first step is to get a working Raspberry Pi configuration, complete with your chosen method of network connection. This has been documented many times, so we won't go into the details – plus, downloading and installing NOOBS onto your Raspberry Pi makes the whole process easier than ever. Just make sure the Raspbian installation and the firmware is up to date, because there are some known issues with Raspberry Pi stability, especially with older versions. And stability is key when you're asking a Raspberry Pi to control temperatures for a week or two.To update Raspbian, type:

```
sudo apt-get update
sudo apt-get upgrade
```

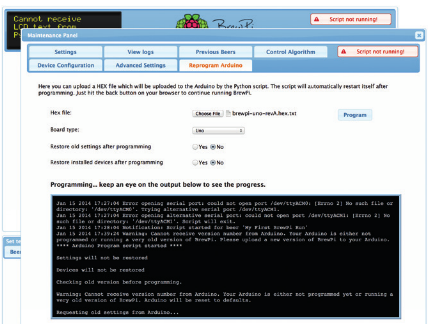
To update the firmware, type:

```
sudo apt-get install rpi-update
sudo rpi-update
```

We now need to grab the latest installation tools. To do that, just enter the following and leave all the answers at their default values:

```
git clone https://github.com/BrewPi/brewpi-tools.
git ~/brewpi-tools
sudo ~/brewpi-tools/install.sh
```

After this has completed, reboot your Pi. You will now be able to point a web browser on your LAN to the IP address of your BrewPi. Don't (yet) get distracted by the blinking lights, as they're not doing anything meaningful. Instead, you need to upload the BrewPi firmware to the Arduino before anything can happen. First download the firmware file itself (here's the link: **http://dl.brewpi.com/brewpi-avr/stable**), and make sure you get the correct file. The file depends on your Arduino type and revision – ours is an Arduino Uno Rev A, for instance. To upload this to your BrewPi, click on the 'Maintenance Panel' button on the right of the web interface, then click on 'Reprogram Arduino'. Select your Arduino from the drop-down menu, then select the downloaded hex file. Make sure 'No' is answered for both the 'Restore Old Settings After Programming' and 'Restore Installed Devices After Programming' options and click on the 'Program' button. You'll see the output of what's happening in the black box below, but with a bit of luck, the BrewPi will beep a couple of times and a few minutes later, you'll have a programmed BrewPi.

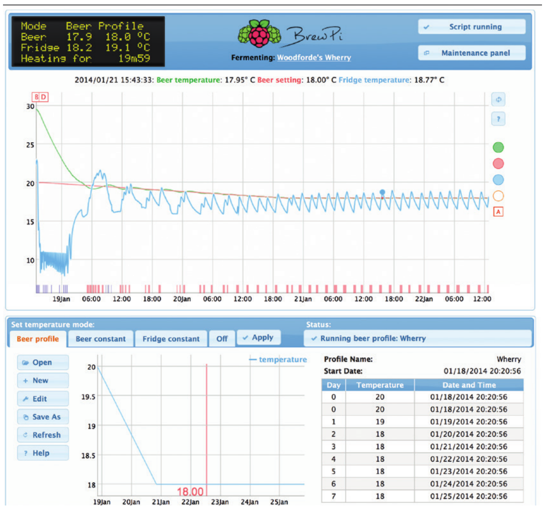


When you update the firmware of the BrewPi, the output console keeps you updated on progress. It only takes a couple of minutes.

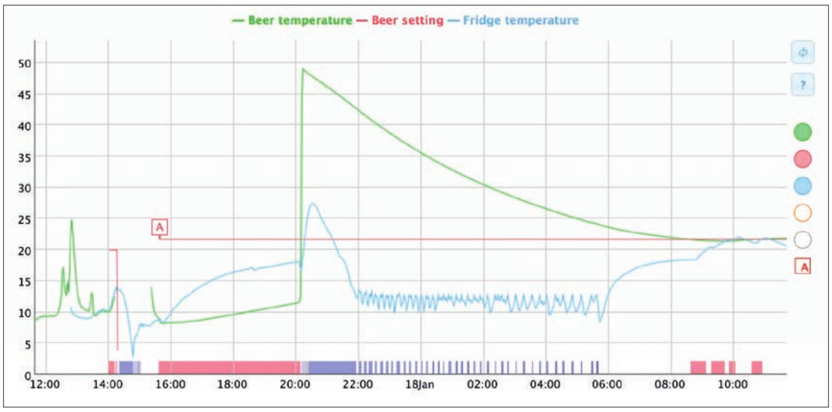
are three different kinds of bolt – two of identical length but slightly different widths, which you'll find out when you try to squeeze a larger one into the smaller holes, but you might notice the other way around, so it's still worth laying everything out before you start, Similarly, there are two different kinds of nut, although on first glance they all look identical, and the case building consists of two separate small phases – connecting the Arduino to the case followed by the LCD panel we built into the shield earlier. The grey threadless spacers are used to distance the LCD from the edge of the case, while the threaded white spacers are used for the Arduino. The position of the holes through the Arduino PCB mean that it can only be fitted onto the case one way – with the power and USB connector along the rear edge.

As we mentioned earlier, you also have the choice of whether to mount the OneWire board to the top panel or mount this inside your freezer cabinet so that the sensors plug directly into this within the freezer. As we opted to mount it to the case, and you need to use the provided small plastic panel (with OneWire embossed onto its top surface, along with numbers for each input). Two of the narrow bolts go through the PCB, through the small plastic panel, through the case, through a washer and finally onto a nut to make this happen.

After connecting the Arduino to the case and making a decision about the OneWire connection, we now need to put everything together like a simple 3D jigsaw puzzle. The half-circle plastic nodules are the feet, and to get ours together, we first fitted the rear panel. This is the one with the holes for power, USB and the controller connectors, and after you've placed it over the Arduino ports, you can hold it in by plugging in the green 'AT-AT' connectors to the outside of the case. They fit in pairs with the exception of the single three-pin connection on one edge. The two side panels then slid into the rear panel, followed by the top and finally the LCD, which slid onto those to all of the other panels to make the front. Don't forget that many



Our first brew started at 20 degrees and lowed to 18 after 48 hours, to create the best temperature for the beer.



The BrewPi is brilliant at controlling temperature. Here's the sensor output after we put a bin of 50°C water into the fridge and asked the BrewPi to take the temperature down to 21°C.

of these panels have a thin layer of plastic that can be removed, along with a few squares for the joints that may not have fallen out with the laser cutting.

Eight of the remaining screws now pull the case together, in the same way that they did for the Raspberry Pi case. The official instructions suggest using a magnet to hold the nut in place, but we we found it easier to push the bolt in until it reaches the gap for the nut, then ease the nut into place using the nut to make sure it doesn't go too far and drop inside the case (which is going to happen with the last one anyway – stay calm and think of beer). A quick tip if one does fall in, you can play an amusing game with yourself and attempt to bounce the nut back out of the same hole - it's not that difficult but looks a little deranged. Sensible people will loosen the bolts at one end to separate the box enough, which is also a good way of taking the top of the case without removing any of the bolts. And don't forget the washers on the outside. They're needed to make the bolt fit.

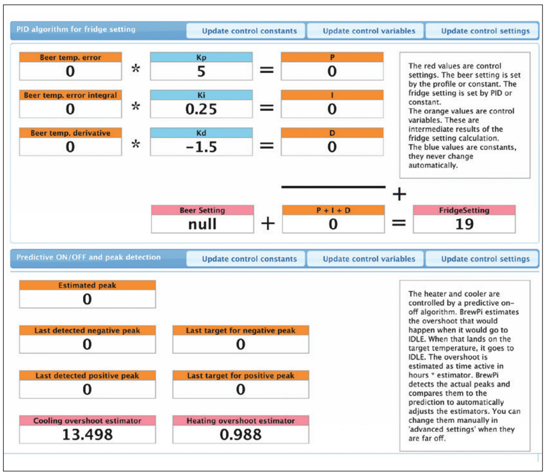
Loose fit

But we'd suggest maybe loosely taping the case together for now, until you've been able to test out your BrewPi with the software to ensure that everything works. That way you don't get doubly frustrated by something not working and having to go through the whole unscrewing process again. You now need to connect the two SSR blocks to the outputs on the shield, making sure you get the positive cable going to the positive input and the negative cable going t the negative input on the SSR. These solid state relays perform a simple job, turning the power going through the other two points either on or off. This is used by the BrewPi to automatically turn on refrigeration or heating. Some BrewPiers have reverse engineered their refrigeration units and heaters to splice these connections into the most efficient place. We cut open the power cables to both the freezer and the heater, took out and cut the negative wire, and used this on other side of the power output on both SSRs. The power output was on the top of our SSRs, while the control inputs were in the bottom. Make sure you get this correct and that your wiring is safe, because you could easily create a hazard at this step. You should also consider the

LV PRO TIP

Although not essential, a cheap multimeter can make testing much easier – especially if it makes a sound when the two contacts connect. This is called testing for continuity, and it's a great way to make sure dodgy soldering is working.

The algorithm that controls the BrewPi is complex, but you can even fine tune this from the Maintenance panel if you so desire.



location of the SSRs, as they're usually exposed and obviously shouldn't go anywhere near liquid.

Back on the BrewPi shield, one output to the SSR triggers a red LED while the other triggers a blue LED, so it's worth getting them correctly connected as you can then see when your device is heating or cooling. These connections are on the backside of the shield, not on the OneWire connector – that's just used for the sensors at the moment, although there's talk of adding a hydrometer reader to measure the alcohol content, which is something we'd love to see.

Now stop. It's time to admire your work. The tough bit is over with, as the BrewPi is now built, waiting only for a little Linux magic to bring it life. And you know all those holes in the top of the BrewPi case? And the weird semi circle feet on the Raspberry Pi case? They fit together! Your Raspberry Pi should sit snugly to the top of the case like the Boeing 747 of brewing.

Configuring devices

The very final step (we promise!), is to tell your BrewPi exactly what you've got connected, and we found it easier to start with a blank canvas. Click on 'Device Configuration' button from the Maintenance panel and you'll see a list of devices your BrewPi thinks are connected. The devices are the switches to control the heating and cooling, plus the two or three sensors you've got connected. If any devices appear in the Installed Devices list, set their function (a drop-down list on the right of each entry) to 'None' and click Apply. This will move them from the 'Installed Devices' box to the 'Detected Devices' box, from where we can now add them as we need to. Enable 'Read Values' and click on Refresh Devices. Click on the 'Refresh Device List' button and enable the 'Read Values' check box. This will list connected devices along with a number to indicate what the switch or sensor is reading. You can easily detect and check your sensors are functioning in this way. OneWire works with unique identifiers embedded within each device, so the device ID is unique for each sensor, not for the BrewPi configuration. That means if you identify which sensor you're going to use within your fermenting bin, you can plus this into any of the OneWire connectors. We checked sensor was working by plugging each in turn and refreshing the device list to make sure a temperature value was being read. We also identified each sensor by heating or cooling the sensor and wrote down which one was which.

You need two sensors for the BrewPi to work properly. One measures the ambient temperature within your fridge or freezer, while the other measures the temperature within the beer. For the beer measurement, it's recommended you use a 'thermowell' to keep the sensor separate from your beer. You also need to solve the problem of getting the sensor cables into the fridge or freezer cavity. Some users piggyback their wires onto any wires they can already find going into fridge. Our approach was to butcher an Ethernet cable – there are more than enough cores within one of these for 2 of the sensors – and drill a tight-fitting hole for both this cable and the power cable for the heating unit, into the side of the freezer. This has worked with no problems so far, and not affected the insulation of the freezer.

Brewing your first beer

With sensors in place and the software running on your BrewPi, you're ready to brew. Despite the slightly intimidating appearance of the web interface, it's very straightforward to use. Click on the 'fermenting' link just below the BrewPi logo and you'll be given the option of starting a new brew. You can do this to log the details of each brew, as well as clear the data for the start of a new fermentation cycle. The main display area is taken up by a graph showing the changes in beer temperature (green) and freezer temperature (blue), as well as the temperature outside the fridge, although this isn't used by its algorithms. At the bottom, along the timeline, blue and red blocks show when the cooling and heating was engaged.

There are three modes for fermenting your brew; Beer Constant, Fridge Constant and Beer Profile. Beer Constant simply keeps the beer at a specific temperature, which you dial into the large number bar at the bottom of the screen. Expanding on this, the Beer Profile setting enables you to set a desired beer temperature for each day. This is useful if you want to try a slightly warmer environment at the beginning and end of the fermenting cycle. When either of these beer profiles are active, the LCD display shows the absolute temperature as well as the temperature for the profile. This is the target temperature for the algorithm, and you'll find the BrewPi will cool or heat to nudge the temperature closer to the desired value.

The Fridge Constant setting does what it says, keeping the temperature of the fridge at a specific value. This might be useful for the couple of days after you've bottled your beer, or put it in a cask, as you usually have a couple of days of secondary fermentation. But it could be equally useful for cooling your final product for the final, essential step of brewing beer – keeping your home-brew ready to drink at a perfect temperature, all year round.

BREWING YOUR OWN BEER: A BRIEF ENCOUNTER

The world of homebrew will feel familiar – it's full of people who obsess over details and argue endlessly about packages.

Homebrew forums across the internet are full of enthusiasts arguing over every detail of the brewing process. And we mean every detail. Fermentation temperature is a dark art of its own, as is the amount of priming sugar to use – we've seen simpler algorithms explain Bézier curves in OpenGL!

As with Linux, all this data and debate can be totally overwhelming to the beginner. But again like Linux, it's worth struggling through to the other side. Just think of the beer.

We also see no shame in starting small. Beer kits are perfect for this. They can be a little pricey, but they'll take the pain out of your first brew. To get started, you'll need some simple pieces of kit. Here's what we recommend:

1 A 25-litre fermentation bin

This doesn't need to be absolutely airtight, as the brewing process will create CO_2, which sits on the top to create an airlock. We drilled a hole in the top to encase one of our BrewPi sensors within its own well.

2 A similar sized pressure barrel

The pressure part is important for the secondary fermentation preocess, because it's what carbonates your beer and keeps your beer fresh. We'd recommend a pressure valve with a connector for a CO_2 canister. These are relatively cheap, and they're used to create a CO_2 buffer when the pressure gets too low to push the beer out effectively. If you don't want to use a pressure barrel, you can use bottles with caps.

Just like open source software, you can create your own recipe or you can stand on the shoulders of giants. Image credit http://superflex.net

3 Sanitiser

Everything that comes into contact with your developmental beer has to be free of any harmful bacteria. Bacteria and wild yeast kill beer over the period it is stored, leading to feelings similar to a hard drive failure.

4 A syphon and hydrometer.

The syphon is to transfer your beer from the fermentation bin to the pressure barrel or bottles, while the hydrometer is to calculate how much alcohol is in your brew. You must measure the gravity at the beginning and the end of the process for this to work – taking a measurement at the end isn't enough.

The biggest threats to your beer are sanitisation, as we've already mentioned, and temperature fluctuation, which is solved with the BrewPi. Another tip we've found helpful is to cover all threads (such as those for the tap, the top and the valve on the pressure barrel) with Vaseline, as this helps to keep them airtight.

After you've whetted your appetite with a beer kit or two, it's time to move up to replacing the kit with your own. There are thousands of years of experience on the subject, and to be honest, we've only just started. But a good place to look for your first brew is a recipe that is itself open source.

Free Beer

This is exactly what is offered at **FreeBeer.org**, a tested and refined recipe for making excellent beer that's been released CC-BY-SA.

The ingredients list five different types of malt, Guaraná beans for added spice and energy and London ale yeast. This is followed by step-by-step instructions that will take your beer from mash to wort to fermentation to beer in as little as three weeks, all in the name of Free Beer. If you do get around to making some, and you have a bottle left over, you know where to send them.