

AutoBrew - Build Notes for 10 litre machine

Components

Vessels

AutoBrew is built from two cheap 19 litre (5 US gallon) stainless steel pots.

<https://www.bigw.com.au/product/smart-value-19-litre-stock-pot/p/WCC10000000011544/>

A base could be made from a third cut down pot if needed.

Heating

The heating element is the same used for the Crown Industries 10 litre hot water urn.

<http://crownindustries.com.au/spares.php#urn>

LD001	Concealed element 1.5kw. 85507	\$37.00
-------	--------------------------------	---------

Control components

Particle Photon: <https://store.particle.io/collections/photon>

Particle Relay Shield: <https://store.particle.io/products/relay-shield>

A small 12V power supply was used to power the Particle Photon and Particle Relay Shield. A plug pack with USB output would be suitable if the Relay Shield was not used.

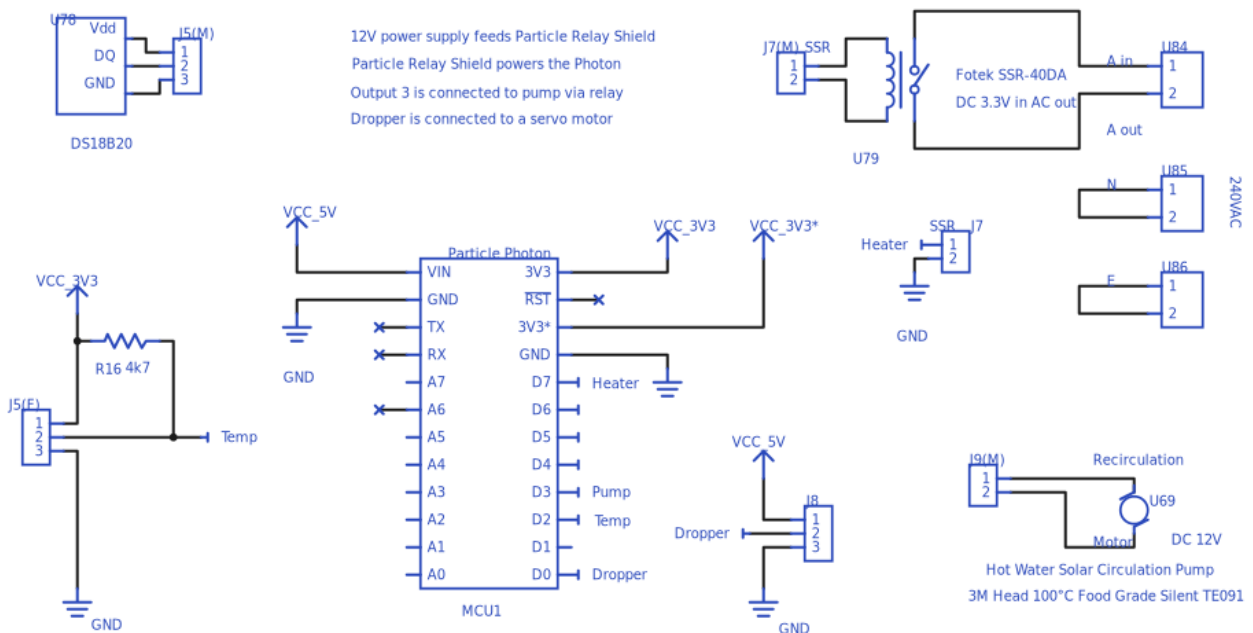
A Solid State Relay is used to power the heating element. Use one that matches the Photon (3.3V drive) and the heating element (in my case 1500W). Most Solid State Relays require a heatsink.

Circuit diagram

For the circuit on upverter, navigate to the following address.

<https://upverter.com/arniew1965/bf7f92c6be11514c/AutoBrew/>

Note that the circuit is not ready for pcb layout and may not contain all necessary component details. This is a working prototype, so a final design would include replacing the relay shield with a solid state switch for the pump, a simple power supply, and some LED status indicators.



Code for the Machine

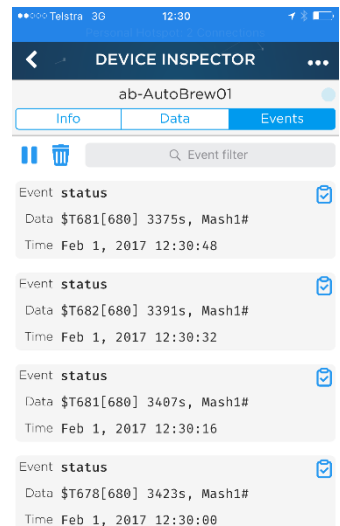
The machine is controlled by a Particle Photon. This is similar to an Arduino that is wi-fi connected. The Particle.io service includes in built cloud support with programming and testing tools. See them for details on getting up and running with the Photon.

The code is available via github: <https://github.com/arniew1965/AutoBrew-Machine>

Controlling the Machine

Currently the machine is controlled and monitored via functions and variables that belong to the Particle infrastructure. Various Particle tools are useful to control the machine. The Particle Dev IDE (built on top of Atom) is a favourite, as is the Particle iOS app.

A web app is planned that will use Angular to create an easy to use interface.



Monitoring and controlling the brew

AutoBrew publishes a human readable event to show brewing status.

The events at right show that the machine is in the first mash step. Over the four events, the temperature varies +/- 0.2 degrees C. The temperature and [target] are in degrees x 10. The number that ends with 's' shows how many seconds remain in this step.

The machine can be controlled via the 'control' function. For a full set of commands, see the section *Control reference*.

Automated Brewing

Once the machine has started brewing, it will complete all mash steps, boil and hop additions automatically. **Note that the hop dropper is not currently shown in this document as it is still in development.**



How it works

The machine pre-heats ready for a mash infusion. The infusion step also has a time component to allow for delayed starts if required.

For all stages, the machine monitors and acts on both temperature and time.

Once the liquor is heated, the pump moves liquor from the kettle to float the grain bed in the mash tun. Liquor is pushed up and through the grain and returns to the kettle via the *liquor return pipe*. The filters on top of the grain bed prevent grains from entering the kettle.

Once the mash is complete, the pump turns off and liquor in the *mash tun* drains to the kettle via the *pump* path. At the same time the kettle heats to boil.

Boiling is timed, including hop additions.

At the end of boiling, a valve is opened to fill a receiving vessel or optionally a chilling path to the receiving vessel.

Detailed notes

Kettle liquor volume indicator



The volume indicator pipe serves two purposes.

1. Show volume in kettle
2. The pipe connects the pump outlet to the mash tun which sits above.

The notches on the pipe indicate volume settings. The three notches in the image correspond to 15, 16 and 17 litres.

The thinner rubber gasket is moved to indicate volume required for the current brew recipe. The kettle is then filled to this 'line'.

The rubber gasket at the upper end acts as a seal on the bottom of the mash tun.

Control reference

The 'command' function can be used to control the machine in a human readable form. Note that a companion web app is under development. The web app will streamline control functions.

Command	Description	Notes
r	Start/end brewing	Starts the set recipe OR stops all brewing processes (toggle)
R	Get current recipe	Sends recipe string to ABmessage variable
Rt,mm;t,mm;...	set recipe	Use 'R' and then without spaces send number pairs 't,mm;' for - Infusion step - Mash steps 1-5 (use 0's for unused steps) - Where t is temp in deg x 10 and mm is time in minutes Then number for boil minutes 'mm;' Then numbers for three hop addition times 'mm,mm,mm,' Ie. R690,0;680,60;760,1;0,0;0,0;0,0;90;60,5,0,
cxxxx	set clock mins	Number from 0-9999
txxxx	set Target temperature	Number from 0-9999 (in degrees x 10)
d[num]	set servo to drop position [num]	Number from 0-4 - 0 is park or home position
d[num],xxx	set servo angle for drop [num]	Angle is from 0-180 degrees - This is for setting up only
v	Send version	Sends the current version number to ABmessage
u	Toggle unit	Degrees Celcius/Farenheit
bxxxx	Set boil trigger	Sets the trigger point for boil in degrees x 10 - Should be a value that is guaranteed to be exceeded
b	Return boil trigger	Sends value of trigger to ABmessage
V	Toggle verbose	Shows extra information to published event if needed

Build notes

Building and design notes are available in a separate document.

References

AutoBrew blog: arniew.wordpress.com

Particle: For reference material, hardware and software for the photon that controls AutoBrew
www.particle.io/

Contact me: Arnie Wierenga, on gmail – arniew1965@